



Universidad Autónoma
de Baja California

"2026, Año de la Educación para la Construcción de la Paz"

FACULTAD DE CIENCIAS

Oficio No. 522/2026-1

DR. LUIS ENRIQUE PALAFOX MAESTRE

Rector de la Universidad Autónoma de Baja California

Presente

Estimado Sr. Rector:

Anteponiendo un cordial saludo, por medio del presente y en respuesta al Oficio No. 692/2026-1 emitido por la Secretaría General de nuestra máxima casa de estudios, me permito someter a su consideración la Propuesta de Modificación del Plan de Estudios del Programa Educativo de Ciencias Computacionales, que se oferta en la Facultad de Ciencias, con la finalidad de que sea incluida en el orden del día de la Sesión Ordinaria del H. Consejo Universitario, a efecto de que sea turnada a la Comisión correspondientes para su revisión y posterior dictamen.

En ese sentido, sírvase encontrar adjunto; el Acta de la Sesión Ordinaria de Consejo Técnico de la Facultad de Ciencias, el documento de propuesta y el oficio de validación de la Coordinación General de Formación Profesional.

Sin otro particular, le agradezco la fina atención y le envío un afectuoso saludo.

ATENTAMENTE

Ensenada, Baja California, 24 de abril de 2026

"POR LA REALIZACIÓN PLENA DEL SER"

DIRECTOR

DR. ALBERTO LEOPOLDO MORÁN Y SOLARES



Universidad Autónoma
de Baja California

24 ABR 2026

RECTORÍA

RECIBIDO

C.c.p. Archivo.
ALMYS/ljmp



Universidad Autónoma
de Baja California

"2026, Año de la Educación para la Construcción de la Paz"

COORDINACIÓN GENERAL DE FORMACIÓN PROFESIONAL

Oficio núm. 127/2026-1

Asunto: Validación de Programa Educativo

DR. ALBERTO LEOPOLDO MORÁN Y SOLARES

Director de la Facultad de Ciencias

Ensenada, B.C.

Presente. -

En atención a la solicitud para la creación del Programa Educativo de **Licenciatura en Ciencias Computacionales**, y derivado de la revisión técnica realizada por esta Coordinación General a mi cargo, me permito comunicarle lo siguiente:

Una vez analizada la propuesta y habiéndose verificado el cumplimiento de los requisitos académicos y curriculares, así como la pertinencia del plan de estudios propuesto, esta **Coordinación General de Formación Profesional otorga su VALIDACIÓN** al citado programa.

Lo anterior, en cumplimiento de lo instruido por la Secretaría General mediante el oficio circular **OF.CIR. 692-2026**, relativo a la integración del orden del día de la segunda sesión ordinaria del H. Consejo Universitario del presente año.

Se le recuerda que, para la integración del expediente definitivo que deberá remitirse a la Secretaría General a más tardar el **24 de abril**, es necesario adjuntar a este oficio la totalidad de la documentación probatoria, incluyendo el Acta de Consejo Técnico de su unidad académica y la propuesta curricular completa.

Sin más por el momento, agradezco su compromiso con la excelencia académica de la Facultad a su digno cargo y aprovecho la ocasión para enviarle un cordial saludo.

ATENTAMENTE

Mexicali, Baja California, a 24 de abril de 2026

"POR LA REALIZACIÓN PLENA DEL SER"

COORDINADORA GENERAL

DRA. YESSICA ESPINOSA DÍAZ

Universidad Autónoma
de Baja California

24 de abril de 2026

Coordinación General
de Formación Profesional

DESPACHADO



Universidad Autónoma
de Baja California

RECTORÍA
COORDINACIÓN GENERAL
DE FORMACIÓN PROFESIONAL

C.c.p. Dr. Luis Enrique Palafox Maestre. Rector de la UABC.
C.c.p. M.I. Edith Montiel Ayala. Secretaria General.
C.c.p. Minutario
YED/liliana

Recibido:
24-04-2026

SESIÓN ORDINARIA DE CONSEJO TÉCNICO DE LA FACULTAD DE CIENCIAS
15 de abril de 2026

Siendo las 12:00 horas del miércoles 15 de abril de 2026, las y los integrantes del Consejo Técnico de la Facultad de Ciencias se reunieron en sesión ordinaria en el Audiovisual A de la Facultad de Ciencias para atender la convocatoria emitida por el Presidente, Dr. Alberto Leopoldo Morán y Solares, bajo el siguiente:

ORDEN DEL DÍA

1. Apertura de la sesión.
2. Lista de asistencia y declaración del quórum legal.
3. Elección de dos escrutadores, uno por el personal académico y otro por el alumnado.
4. Lectura y aprobación del orden del día.
5. Lectura y aprobación del Acta de la sesión anterior.
6. Presentación, discusión y, en su caso aprobación, de la propuesta de modificación del Plan de Estudios del Programa Educativo de Biología.
7. Presentación, discusión y, en su caso aprobación, de la propuesta de modificación del Plan de Estudios del Programa Educativo de Física.
8. Presentación, discusión y, en su caso aprobación, de la propuesta de modificación del Plan de Estudios del Programa de Ciencias Computacionales.
9. Presentación del informe de la Comisión Especial sobre la propuesta del Reglamento para la Colección de Vertebrados de la Facultad de Ciencias.
10. Asuntos generales.
11. Clausura de la sesión.

----- DESAHOGO DEL ORDEN DEL DÍA -----

1. Apertura de la sesión

El Presidente del Consejo, Dr. Alberto Leopoldo Morán y Solares, da la bienvenida a la sesión del Consejo Técnico de la Facultad de Ciencias y explica el proceder de los principales puntos a tratar en la sesión.

2. Lista de asistencia y declaración del quórum legal

Se cuenta con la presencia de nueve propietarios y tres suplentes que fungen como propietarios, por lo que se declara el quórum legal.

3. Elección de dos escrutadores, uno por el personal académico y otro por el alumnado

El Presidente solicita propuestas para la elección de las o los escrutadores para la sesión, recordando que su conformación debe incluir un representante del personal académico y uno por parte del alumnado. Por parte del alumnado se propone a Laura Esther Oropeza Guerrero, mientras que por parte del personal académico se propone al Dr. Raúl Casillas Figueroa. Al no haber más propuestas, se somete a votación y se aprueba por unanimidad.

4. Lectura y aprobación del orden del día

El Presidente le solicita al Secretario dé lectura al orden del día. Se somete a votación el orden del día y se aprueba por unanimidad.

5. Lectura y aprobación del Acta de la sesión anterior

El Presidente del Consejo recuerda que, con base en un acuerdo previo, se envió con antelación el Acta de la sesión anterior para su revisión, procediendo a preguntar si existe alguna observación o comentario. Al no haber comentarios, el Presidente somete a votación la aprobación del Acta de la sesión anterior y se aprueba por unanimidad.

6. Presentación, discusión y, en su caso aprobación, de la propuesta de modificación del Plan de Estudios del Programa Educativo de Biología

Para el desahogo de este punto, el Presidente del Consejo solicitó al pleno la autorización para la integración a la sesión del Dr. Ulises Gregorio Pacheco Bardullas y del Dr. Manuel Iván Ocegueda Miramontes, con el propósito de presentar las propuestas de modificación de los programas educativos de Biología y Física, respectivamente. La solicitud fue aprobada por unanimidad. Se hace constar que la Dra. María Victoria Meza Kubo participó en su carácter de integrante del Consejo Técnico y como responsable de presentar la propuesta de Ciencias Computacionales.

Acto seguido, el Presidente solicitó al Dr. Pacheco la presentación de la propuesta de modificación del plan de estudios de la Licenciatura en Biología. Durante su intervención, se destacó que la propuesta tiene como objetivo consolidar un perfil de egreso acorde con las exigencias actuales en áreas como conservación, biología molecular, biotecnología, biodiversidad, ecología, así como la gestión y manejo de recursos naturales. En la fase de discusión, se abordó la pertinencia de eliminar el tronco común, señalando que la modificación busca fortalecer una estructura curricular sólida y actualizada, que permita al alumnado adquirir dominio de diversas metodologías desde el inicio de su formación. Asimismo, se reflexionó sobre los momentos adecuados para incorporar certificaciones adicionales al plan de estudios.

Concluida la presentación y discusión, la propuesta de modificación del plan de estudios de la Licenciatura en Biología fue sometida a votación, siendo aprobada con diez votos a favor y dos abstenciones.

7. Presentación, discusión y, en su caso aprobación, de la propuesta de modificación del Plan de Estudios del Programa Educativo de Física

Para el desahogo de este punto, el Presidente del Consejo solicitó al Dr. Manuel Iván Ocegueda Miramontes la presentación de la propuesta de modificación del plan de

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

estudios de la Licenciatura en Física. Durante su intervención, se expuso que la propuesta se fundamenta en los resultados de evaluaciones internas y externas del programa educativo, así como en su pertinencia social y contexto profesional, incorporando además la retroalimentación de personas empleadoras y egresadas. En la discusión, se analizó la selección de certificaciones y la posibilidad de su actualización o incorporación en etapas posteriores del plan de estudios.

Concluido el análisis, la propuesta de modificación del plan de estudios de la Licenciatura en Física fue sometida a votación, siendo aprobada con once votos a favor y una abstención.

8. Presentación, discusión y, en su caso aprobación, de la propuesta de modificación del Plan de Estudios del Programa Educativo de Ciencias Computacionales

Para el desahogo de este punto, el Presidente del Consejo solicitó a la Dra. María Victoria Meza Kubo la presentación de la propuesta de modificación del plan de estudios de la Licenciatura en Ciencias Computacionales. Durante su intervención, se señaló que la propuesta se sustenta en un diagnóstico interno y externo del programa educativo, orientado al fortalecimiento de habilidades técnicas y habilidades blandas, así como a la vinculación mediante proyectos interdisciplinarios y al desarrollo de competencias transversales. Con base en estos elementos, se planteó una actualización disciplinar del programa.

Concluido el análisis, la propuesta de modificación del plan de estudios de la Licenciatura en Ciencias Computacionales fue sometida a votación, siendo aprobada con ocho votos a favor, un voto en contra y tres abstenciones.

9. Presentación del informe de la Comisión Especial sobre la propuesta del Reglamento para la Colección de Vertebrados de la Facultad de Ciencias.

El Presidente solicita propuestas para abordar los puntos faltantes del orden del día, con el propósito de respetar el acuerdo de que el tiempo de la sesión no exceda dos horas. Se propusieron las siguientes opciones:

- Dejar la sesión abierta para concluir los puntos faltantes del orden del día.
- Cerrar la sesión actual y convocar a una nueva sesión ordinaria el día 22 de abril de 2026 para concluir los puntos faltantes del orden del día.

Se someten a votación las propuestas y se aprueba por unanimidad cerrar la sesión actual y convocar a una nueva sesión ordinaria el día 22 de abril de 2026 para concluir los puntos faltantes del orden del día.

10. Asuntos generales.

No se presentaron temas para asuntos generales.

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]
[Handwritten signature]
[Handwritten signature]
[Handwritten signature]
[Handwritten signature]
[Handwritten signature]
[Handwritten signature]

[Handwritten signature] *[Handwritten signature]* *[Handwritten signature]* *[Handwritten signature]* *[Handwritten signature]*

11. Clausura de la sesión

Siendo las 14:00 horas se da por concluida la sesión. Se anexa lista de asistencia.



Dr. Alberto Leopoldo Morán y Solares
Presidente del Consejo Técnico



Dr. José Israel Paez Ornelas
Secretario del Consejo Técnico



Dr. Raúl Casillas Figueroa
Escrutador



Laura Esther Oropeza Guerrero
Escrutadora



Unicef de la Rosa

R. Paez Ornelas

Apelú

luis



Jan



CYK



She





UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

"2026, Año de la Educación para la Construcción de la Paz"

Licenciatura en Ciencias Computacionales

Propuesta de modificación del plan de estudios y cambio de denominación que presenta la Facultad de Ciencias, Ensenada.

Fondo de un carnero con cuernos grandes, sobre un fondo verde oscuro.

Ensenada, Baja California, México. Mayo de 2026

**Propuesta de modificación del Plan de Estudios 2017-1 y
cambio de denominación de Licenciado en Ciencias
Computacionales a Licenciatura en Ciencias
Computacionales**

Facultad de Ciencias



UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

DIRECTORIO

Dr. Luis Enrique Palafox Maestre

Rector

M.I. Edith Montiel Ayala

Secretaria General

Dra. Lus Mercedes López Acuña

Vicerrectora campus Ensenada

Dra. Haydeé Gómez Llanos Juárez

Vicerrectora campus Tijuana

Dr. Jesús Adolfo Soto Curiel

Vicerrector campus Mexicali

Dra. Yessica Espinosa Díaz

Coordinadora General de Formación Profesional

Dr. Alberto Leopoldo Moran Y Solares

Director de la Facultad de Ciencias

Dr. Antelmo Castro López

Jefe del Departamento de Diseño Curricular



PROYECTO DE MODIFICACIÓN

Coordinadores

Dra. María Victoria Meza Kubo

Comité responsable

Dra. Alma Rocío Cavazos Marín

M.I. Adrián Enciso Almanza

Dra. Eloisa del Carmen García Canseco

Dr. José Angel González Fraga

Dr. Everardo Gutierrez López

Dr. Evelio Martínez Martínez

MGTIC. Gerardo Tovar Ramos

Coordinación metodológica y técnica del diseño curricular

Dr. Antelmo Castro López

Mtra. Melissa Zuno Bolaños

Diseño de programas de unidad de aprendizaje

Dr. Jorge Alberto Villavicencio Aguilar

Dr. Roberto Romo Martínez

M.I. Adrián Enciso Almanza

Dra. María Victoria Meza Kubo

Dr. Raúl Casillas Figueroa

Dr. Julian Román Durán Camacho

Dra. Eloísa del Carmen García Canseco

M.C. Adina Jordán Arámbulo

M.C. Claudia Lara Silva

Dr. Juan Crisóstomo Tapia Mercado

Dra. Brenda Leticia de la Rosa Navarro

Dr. Carlos Yee Romero

Dr. Jonathan Verdugo Olachea

Dr. José Ariel Camacho Gutiérrez

M.C. Evelio Martínez Martínez

Dr. Everardo Gutiérrez López

Dra. Priscila Elizabeth Iglesias Vázquez

Dr. Ramón Carrillo Bastos
M.G.T.I.C. Gerardo Tovar Ramos
Dr. Jose Ángel González Fraga
Dra. Alma Rocío Cabazos Marín
Dr. Elmer Cruz Mendoza
Dr. Duilio Valdespino Padilla
Lic. Ivan Valencia Santiago
Dr. Gustavo Enrique Ramos Alcaraz
M.C. Roberto Adolfo Romero Hernández
M.C. Pedro Pérez Ruiz
Dr. Alberto Leopoldo Morán y Solares
Dra. Selene Solorza Calderón
M.C. Antonio de Jesús García Chávez
M.T.I.C. Mary Sol Aguirre Arzate
Dr. Sergio Omar Infante Prieto
M.G.T.I.C. Omar Christian Aparicio Ortiz
Dr. Manuel Ivan Ocegueda Miramontes
Dr. José Eleno Lozano Risk
Dra. Angelina González Peralta
Dr. Claudio Ismael Valencia
M.T.I.C. Gabriel Aparicio Hernández
Dr. Miguel Angel Alonso Arévalo
Dr. Julio Enrique Valencia Suarez

Conducción y acompañamiento en el diseño de programas de unidad de aprendizaje

Lic. Verónica Elizabeth Rosas Rojas
Mtra. Brianda Guadalupe García Guerrero
Mtra. Melissa Zuno Bolaños
Mtra. Itzel Ashanty Moreno Heras
Dra. Ananda Aracely Navarro Barrera

Índice

1. Introducción	9
2. Justificación	12
2.1. Fundamentación social	12
2.2. Fundamentación de la profesión	14
2.3. Fundamentación institucional	17
3. Filosofía y enfoque educativo	22
3.1. Articulación institucional y del programa educativo	26
4. Plan de estudios	31
4.1. Perfil de ingreso	31
4.2. Perfil de egreso	32
4.3. Campo profesional	33
4.4. Mapa curricular	35
4.5. Distribución de créditos y de unidades de aprendizaje	36
4.6. Características de las unidades de aprendizaje por etapas de formación	37
4.7. Características de las unidades de aprendizaje por áreas de conocimiento	40
4.8. Tipología de las unidades de aprendizaje	43
4.9. Equivalencias de las unidades de aprendizaje entre planes de estudio	47
5. Estructura y componentes formativos del plan de estudios	50
5.1. Etapas de formación	51
5.1.1. Etapa básica	51
5.1.2. Etapa disciplinaria	52
5.1.3. Etapa terminal	53
5.2. Promoción de competencia en el idioma inglés	54
5.3. Promoción del uso de la TI e IA para el desarrollo de la profesión	57
5.4. Promoción de habilidades blandas para el ejercicio de la profesión	60
5.5. Propuesta de certificados	64
5.6. Bienestar estudiantil	65
6. Modalidades de aprendizaje y obtención de créditos	67
6.1. Modalidades curriculares (ruta estructurada del plan de estudios)	67
6.1.1. Unidades de aprendizaje obligatorias	68

6.1.2. Unidades de aprendizaje optativas.....	69
6.1.3. Otros cursos optativos	69
6.2. Otras Modalidades de Aprendizaje y Obtención de Créditos (OMAS).....	70
6.2.1. Estudios independientes.....	71
6.2.2. Ayudantía docente.....	72
6.2.3. Ayudantía de investigación.....	73
6.2.4. Ejercicio investigativo	73
6.2.5. Ayudantía de laboratorio.....	74
6.2.6. Apoyo a actividades de extensión y vinculación.....	75
6.2.7. Proyectos de vinculación con valor en créditos (PVVC).....	76
6.2.8. Programa de emprendedores universitarios.....	81
6.2.9. Unidades de aprendizaje por asesoría académica.....	81
6.2.10. Actividades para la formación en valores	82
6.2.11. Movilidad e intercambio estudiantil.....	84
6.2.12. Actividades artísticas, culturales y deportivas	88
6.2.13. Cursos intersemestrales	89
6.3. Modalidades obligatorias y requisitos institucionales de egreso	89
6.3.1. Servicio social comunitario y profesional.....	90
6.3.2. Prácticas profesionales.....	91
6.3.3. Lengua extranjera.....	92
6.3.4. Modalidades de titulación	93
7. Condiciones institucionales para la operación del plan de estudios	96
7.1. Difusión del programa educativo	97
7.2. Planta académica.....	99
7.3. Cuerpos académicos.....	101
7.4. Infraestructura, materiales y equipamiento	103
7.5. Recursos bibliográficos y de información.....	108
7.6. Programa de Tutoría Académica	111
7.7. Servicios de apoyo al estudiantado.....	114
8. Sistema de evaluación y aseguramiento de la calidad	117
8.1. Evaluación del plan de estudios.....	118
8.2. Evaluación del aprendizaje	119
8.3. Evaluación colegiada del aprendizaje	121

8.4. Uso de resultados y mejora continua	122
9. Proceso de validación curricular del plan de estudios	123
9.1. Evaluación por pares externos	125
9.2. Aprobación del Consejo Técnico.....	129
9.4. Validación por la Coordinación General de Formación Profesional	133
10. Referencias	134
11. Anexos	139
11.1. Anexo 1. Formatos metodológicos	139
11.2. Anexo 2. Programas de unidad de aprendizaje	173
11.3. Anexo 3. Evaluación externa e interna del programa educativo	913

1.Introducción

La Licenciatura en Ciencias Computacionales ha transitado de un enfoque centrado en la programación y el desarrollo de software hacia una disciplina estratégica, interdisciplinaria y orientada a la solución de problemas complejos mediante el uso intensivo de tecnologías digitales. Esta evolución responde a la acelerada transformación digital que caracteriza al entorno contemporáneo, impulsada por avances como la inteligencia artificial, la ciencia de datos, la computación en la nube, la ciberseguridad y la automatización inteligente. En este contexto, se demandan profesionales capaces no solo de desarrollar tecnología, sino de integrarla de manera crítica, ética y pertinente en diversos ámbitos sociales y productivos.

Las tendencias que actualmente redefinen el campo de las Ciencias Computacionales se articulan con la consolidación de la economía digital y la sociedad del conocimiento. Entre ellas destacan: el desarrollo de sistemas inteligentes basados en aprendizaje automático; la analítica avanzada de datos para la toma de decisiones; la ingeniería de software orientada a arquitecturas modernas, escalables y seguras; la protección de la información en entornos cada vez más interconectados; y el diseño de experiencias digitales centradas en el usuario. De manera transversal, cobra relevancia la incorporación de principios éticos, de privacidad, equidad y sostenibilidad tecnológica, ante los impactos sociales, económicos y culturales derivados del uso de la tecnología.

La relevancia de las Ciencias Computacionales en el escenario actual es estratégica, debido a su carácter transversal y habilitador en prácticamente todos los sectores. Su impacto se manifiesta en ámbitos como la industria, la salud, la educación, las finanzas, el comercio, la seguridad y la gestión pública. En el caso de Baja California, una región con alta dinámica económica, vocación tecnológica y vinculación con ecosistemas de innovación internacionales, los profesionales en esta área desempeñan un papel clave en la transformación digital de las organizaciones, el desarrollo de soluciones tecnológicas de alto valor y el fortalecimiento de la competitividad regional. Su contribución resulta fundamental para la generación de conocimiento, la innovación y el desarrollo sostenible.

En su proceso de consolidación, la disciplina ha ampliado significativamente su campo de acción. Actualmente, los egresados no solo dominan fundamentos teóricos y técnicos en computación, sino que también participan en el diseño, desarrollo e implementación de sistemas complejos, lideran proyectos de innovación tecnológica, colaboran en equipos multidisciplinarios y contribuyen al desarrollo de investigación aplicada. La naturaleza dinámica de las Ciencias Computacionales favorece su articulación con múltiples campos del conocimiento, lo que permite abordar problemáticas desde una perspectiva integral y orientada a resultados.

En este marco, la actualización del plan de estudios de la Licenciatura en Ciencias Computacionales respondió a la necesidad de alinear la formación profesional con las demandas actuales y prospectivas del entorno tecnológico y social. En este sentido, se buscó consolidar la formación de egresados con competencias sólidas en el ámbito técnico, con capacidad de análisis y resolución de problemas, pensamiento crítico, habilidades de innovación y un compromiso ético con el uso de la tecnología. De esta manera, el programa se posiciona como un eje estratégico para el desarrollo científico, tecnológico y social, contribuyendo de manera significativa a la construcción de una sociedad más equitativa, sostenible y basada en el conocimiento.

En atención a este propósito, el Plan de Estudios 2017-1, después de siete años de operación, fue sometido a un proceso de evaluación externa e interna con el objetivo de determinar su pertinencia (Serna y Castro, 2018). Los resultados obtenidos permitieron identificar fortalezas, áreas de oportunidad y necesidades emergentes del contexto, lo que condujo a la definición de una propuesta de **modificación curricular** orientada a fortalecer la coherencia, pertinencia y flexibilidad del programa educativo, en congruencia con la evolución de la ciencia, la tecnología y las dinámicas del entorno.

En este documento se presenta la propuesta de modificación del plan de estudios de la Licenciatura en Ciencias Computacionales, diseñada bajo la filosofía institucional, el enfoque basado en competencias y la flexibilidad curricular que orientan el desarrollo de programas educativos de la Universidad Autónoma de Baja California (2018), y sustentada en una estructura académica y administrativa pertinente y suficiente para su operación.

Para su análisis y valoración, el documento se organiza en diez apartados que integran de manera articulada los fundamentos, componentes y condiciones de operación del plan de estudios. En el primero, se presenta la introducción. En el segundo, se desarrolla la justificación de la propuesta de modificación del Plan de Estudios 2017-1, sustentada en los resultados de la evaluación externa e interna del programa educativo. En el tercero, se expone la filosofía y el enfoque pedagógico, así como su articulación institucional.

En el cuarto apartado se describe el plan de estudios, incluyendo el perfil de ingreso, el perfil de egreso, el campo profesional, el mapa curricular, la organización por etapas de formación y áreas de conocimiento, así como la distribución de créditos y la tipología de las unidades de aprendizaje. En el quinto, se detallan la estructura y los componentes formativos del programa, incluyendo la promoción de competencias transversales. En el sexto apartado se presentan las modalidades de aprendizaje y obtención de créditos, así como los requisitos institucionales de egreso.

En el séptimo apartado se describen las condiciones institucionales para la operación del plan de estudios, incluyendo la planta académica, la infraestructura, los recursos y los servicios de apoyo al estudiantado. En el octavo, se define el sistema de evaluación y aseguramiento de la calidad, tanto del plan de estudios como del aprendizaje. En el noveno apartado se integra el proceso de validación curricular. Finalmente, en el décimo apartado se incluyen las referencias y los anexos que documentan y respaldan el desarrollo metodológico, académico y normativo de la propuesta.

2. Justificación

Los diagnósticos que han motivado la modificación del plan de estudios de la Licenciatura en Ciencias Computacionales, realizados en los años 2018 y 2024, sustentan que el plan de estudios actual ha sido pertinente para la formación de competencias en el desarrollo de soluciones computacionales; sin embargo, el contexto tecnológico, social y laboral ha evolucionado de manera acelerada, generando nuevas demandas que requieren una actualización curricular integral. En el diagnóstico más reciente, se identificó que, si bien persisten necesidades previamente detectadas, estas se han intensificado y complejizado, particularmente en los siguientes aspectos:

1. La transformación digital se ha acelerado de manera significativa, incorporando nuevas tecnologías emergentes como la inteligencia artificial, el análisis de grandes volúmenes de datos, la computación en la nube y la ciberseguridad, lo que demanda profesionistas con capacidades técnicas especializadas y actualización continua.
2. Se ha transitado de una formación centrada en fundamentos teóricos hacia la necesidad de integrar enfoques prácticos, interdisciplinarios y orientados a la solución de problemas reales, en donde el egresado no solo comprenda la tecnología, sino que la aplique estratégicamente en distintos sectores productivos y sociales.
3. Existe una demanda creciente por fortalecer no solo las competencias técnicas, sino también las habilidades transversales, tales como la comunicación, el trabajo en equipo, la adaptabilidad y la capacidad de aprendizaje continuo, necesarias para el desempeño profesional en entornos dinámicos y altamente competitivos.

2.1. Fundamentación social

Los resultados de la evaluación externa e interna evidencian que el egresado de la Licenciatura en Ciencias Computacionales debe responder a problemáticas sociales vinculadas con la transformación digital, la automatización de procesos, la gestión de la información y la innovación tecnológica en distintos sectores productivos y sociales. Este contexto se enmarca en una realidad global caracterizada por una población superior a

los 8,000 millones de personas y por el crecimiento acelerado del uso de tecnologías digitales, así como del aprendizaje en línea y la capacitación en habilidades tecnológicas (Organización de las Naciones Unidas [ONU], 2019; Organisation for Economic Co-operation and Development [OECD], 2022).

A nivel social, se identifican tendencias relevantes como el incremento en la capacitación digital, particularmente en áreas como análisis de datos y tecnologías de la información, así como nuevas expectativas laborales en los jóvenes, quienes buscan equilibrio entre vida personal y profesional, estabilidad financiera y compromiso con el entorno social y ambiental (Deloitte, 2022). En México, con una población de más de 126 millones de habitantes y una edad media de 29 años, existe una base demográfica joven que demanda mayores oportunidades de formación y empleo en sectores tecnológicos (Instituto Nacional de Estadística y Geografía [INEGI], 2020). Asimismo, se prevé un envejecimiento progresivo de la población hacia 2050, lo que incrementa la necesidad de soluciones tecnológicas orientadas a distintos grupos etarios (Consejo Nacional de Población [CONAPO], 2019).

El contexto económico y social también evidencia desigualdades, como la persistencia de condiciones de pobreza en diversas regiones del país, así como una brecha digital significativa, donde aproximadamente el 50% de la población aún carece de acceso pleno a tecnologías digitales (Consejo Nacional de Evaluación de la Política de Desarrollo Social [CONEVAL], 2020; ONU, 2019). Esta situación resalta la necesidad de desarrollar soluciones tecnológicas accesibles, seguras y contextualizadas, así como fortalecer la inclusión digital.

En este sentido, el mercado laboral presenta una alta demanda actual y futura de profesionistas en el área, con requerimientos crecientes en conocimientos especializados y habilidades prácticas. Los empleadores señalan que, si bien la formación es pertinente, es necesario fortalecer la aplicación del conocimiento en contextos reales, el desarrollo de habilidades blandas y la especialización en áreas como el desarrollo de software (OECD, 2022). Asimismo, se reconoce que muchas organizaciones aún enfrentan retos en su transformación digital y requieren profesionales capacitados en ciberseguridad y adopción tecnológica (Dell Technologies, 2018).

La evaluación consideró una muestra de empleadores predominantemente del sector privado, quienes destacan como principales demandas laborales el dominio técnico, la capacidad de resolución de problemas, el trabajo en equipo y la adaptabilidad. En este sentido, los requerimientos del mercado se estructuran en conocimientos tecnológicos actualizados, habilidades prácticas y analíticas, actitudes proactivas y valores como la ética y la responsabilidad profesional. Entre sus recomendaciones se enfatiza el fortalecimiento del enfoque práctico, la integración de proyectos reales y la formación en competencias transversales.

Por su parte, los egresados presentan una alta inserción laboral, mayoritariamente en áreas afines a su formación, lo que confirma la pertinencia del programa. No obstante, coinciden en la necesidad de mayor especialización, actualización tecnológica y fortalecimiento de la vinculación con el sector productivo. Asimismo, identifican como competencias clave el dominio técnico, la comunicación efectiva, el trabajo colaborativo y la capacidad de aprendizaje continuo (World Economic Forum [WEF], 2023).

Adicionalmente, el contexto nacional e internacional impulsa estrategias orientadas a la transformación digital, como la Estrategia Nacional Digital en México y el Plan Estatal de Desarrollo de Baja California 2022–2027, que enfatizan la necesidad de fortalecer la infraestructura tecnológica, la educación digital y la vinculación con el sector productivo. De igual forma, los Objetivos de Desarrollo Sostenible destacan la relevancia de la educación de calidad, el trabajo decente y la innovación, ámbitos en los que el egresado puede contribuir de manera significativa (ONU, 2019).

En conjunto, estos elementos confirman la pertinencia social del programa educativo y orientan su actualización hacia un enfoque más práctico, especializado y alineado con las demandas actuales y futuras del entorno laboral, incorporando tanto competencias técnicas como habilidades transversales que fortalezcan el desempeño profesional del egresado.

2.2. Fundamentación de la profesión

La profesión en Ciencias Computacionales se desarrolla en un entorno global caracterizado por la transformación digital, la automatización de procesos y la creciente integración de tecnologías de la información en todos los sectores productivos y sociales.

Este contexto se intensificará hacia el 2030, donde se prevé un progreso tecnológico sin precedentes, acompañado de una rápida innovación, colaboración internacional y movilidad laboral, así como de nuevos desafíos éticos y sociales a escala global (Institute of Electrical and Electronics Engineers [IEEE], 2023).

Los avances científicos y tecnológicos que impactan la profesión incluyen el desarrollo de la inteligencia artificial, el análisis de grandes volúmenes de datos, la computación en la nube, la ciberseguridad y el internet de las cosas (IEEE, 2022). Asimismo, se identifican tendencias emergentes como la atención médica remota, los dispositivos inteligentes, la digitalización de transacciones, la realidad aumentada y las tecnologías para la sostenibilidad, lo que amplía el alcance de la profesión hacia sectores como la salud, la industria, el gobierno y la investigación científica (IEEE, 2023). Estos avances han transformado las formas de interacción, producción y gestión de la información, generando la necesidad de profesionales con capacidades para adaptarse a entornos dinámicos y en constante evolución.

Los campos de acción del egresado abarcan el desarrollo de software, la administración de sistemas y redes, la ciencia de datos, la ciberseguridad, la consultoría tecnológica y la investigación aplicada. En términos operativos, las prácticas profesionales se agrupan en funciones como el desarrollo y análisis de software y multimedia, así como la administración de bases de datos y redes de computadoras (Instituto Nacional de Estadística y Geografía [INEGI], 2019). Estas actividades se desarrollan mediante la resolución de problemas complejos, el trabajo interdisciplinario, la gestión de proyectos tecnológicos y la aplicación de metodologías ágiles centradas en el usuario.

La profesión comparte su ejercicio con disciplinas afines como la Ingeniería en Software, la Ingeniería en Ciencias Computacionales, la Ingeniería en Computación y la Licenciatura en Informática, lo que evidencia un campo profesional amplio e interrelacionado (Consejo Nacional de Acreditación en Informática y Computación [CONAIC], 2022).

En cuanto a su evolución, la profesión en México tiene antecedentes desde la década de 1960 con la creación de programas técnicos y de posgrado en computación, consolidándose posteriormente con la oferta de licenciaturas especializadas (Centro de

Investigación y de Estudios Avanzados [CINVESTAV], s.f.). Su crecimiento ha estado estrechamente vinculado al desarrollo industrial y tecnológico del país, con incrementos sostenidos en la formación de profesionistas y en su inserción laboral en áreas estratégicas (Hernández-Laos et al., 2012). Actualmente, la oferta educativa en el área continúa en expansión, con programas impartidos en diversas instituciones de educación superior y una matrícula en crecimiento (Asociación Nacional de Universidades e Instituciones de Educación Superior [ANUIES], 2023).

En términos prospectivos, la profesión se orienta hacia la consolidación de la revolución digital, con impactos directos en la vida cotidiana, la industria, el medio ambiente y la gestión gubernamental (Oropeza-García, 2019). En este escenario, se prevé un crecimiento significativo en ocupaciones como analistas y científicos de datos, especialistas en Big Data, ingenieros de datos y profesionales en redes y bases de datos, con incrementos estimados entre el 30% y 35% hacia 2027 (World Economic Forum [WEF], 2020).

Asimismo, las habilidades requeridas evolucionan hacia un enfoque integral que combina competencias cognitivas (pensamiento analítico y creativo), tecnológicas (alfabetización digital), socioemocionales (resiliencia, adaptabilidad, liderazgo) y de gestión (control de calidad y toma de decisiones) (WEF, 2020). A ello se suma la importancia del aprendizaje a lo largo de la vida, en un contexto donde la capacitación continua en tecnologías de la información es cada vez más demandada (Organisation for Economic Co-operation and Development [OECD], 2022).

Finalmente, la profesión adquiere un compromiso creciente con el desarrollo sostenible, en concordancia con los Objetivos de Desarrollo Sostenible, particularmente en ámbitos como la educación de calidad, el trabajo decente y la innovación tecnológica (Organización de las Naciones Unidas [ONU], 2015).

En este sentido, la formación en Ciencias Computacionales resulta fundamental para responder a los retos actuales y futuros de la sociedad del conocimiento, mediante la preparación de profesionales capaces de innovar, adaptarse y contribuir al desarrollo social, económico y tecnológico.

2.3. Fundamentación institucional

La evaluación interna del currículo permitió identificar fortalezas y áreas de mejora del programa educativo a partir de la participación de docentes y estudiantes, cuyos aportes resultan fundamentales para su actualización y mejora continua. Asimismo, evidenció que la estructura organizacional del programa se encuentra alineada con los propósitos institucionales, lo que favorece su operación; no obstante, se identifican áreas de oportunidad en la gestión de recursos y en el fortalecimiento de las condiciones para el desarrollo de las actividades académicas (Universidad Autónoma de Baja California [UABC], 2024).

En el caso del profesorado, la evaluación consideró una muestra representativa de docentes con experiencia en las distintas áreas del programa, lo que permitió obtener una visión integral sobre la pertinencia y operación del plan de estudios. Las y los docentes reconocen como fortaleza la solidez teórica del programa, así como el desarrollo de competencias relacionadas con el diseño y desarrollo de soluciones computacionales. Sin embargo, señalan la necesidad de fortalecer la articulación curricular, incrementar el enfoque práctico y reforzar otras competencias establecidas en el perfil de egreso. Asimismo, destacan la importancia de incorporar metodologías activas de aprendizaje, fortalecer la vinculación con el sector productivo y promover la formación pedagógica del profesorado y su desarrollo en estudios de posgrado (UABC, 2024).

En el ámbito curricular, se identifica que la etapa disciplinar es la de mayor pertinencia, al promover el pensamiento analítico, lógico y crítico en las y los estudiantes. No obstante, se reconoce la necesidad de fortalecer la aplicación práctica de los conocimientos, el uso de métodos y técnicas propias del campo profesional, así como ampliar los espacios curriculares para el uso de tecnologías de la información. También se destaca la importancia de dar seguimiento a la actualización y cumplimiento de los contenidos temáticos, así como de fortalecer la formación en idioma inglés, particularmente en habilidades de comunicación oral y escrita (UABC, 2024).

Por su parte, la evaluación de estudiantes incluyó una muestra representativa de la población inscrita en diferentes etapas de formación, lo que permitió conocer su experiencia académica desde diversas perspectivas. Las y los estudiantes valoran positivamente la formación recibida, especialmente en los fundamentos teóricos y en el

desarrollo del pensamiento crítico; sin embargo, manifiestan la necesidad de incrementar las experiencias prácticas, los proyectos aplicados y la actualización en herramientas tecnológicas. Asimismo, destacan la importancia de fortalecer habilidades blandas, el trabajo colaborativo y la vinculación con contextos reales de desempeño profesional (UABC, 2024).

En cuanto a la trayectoria escolar, se identifican áreas de atención relacionadas con el rezago académico en determinadas unidades de aprendizaje, lo que requiere revisar las estrategias pedagógicas y fortalecer los mecanismos de apoyo académico. Si bien las actividades de asesoría y tutoría son valoradas como adecuadas, se sugiere ampliar las estrategias de atención a estudiantes en condición de vulnerabilidad, así como promover la movilidad estudiantil nacional e internacional y fortalecer el seguimiento de prácticas profesionales, las cuales contribuyen significativamente a la formación del estudiante. Asimismo, el programa destaca por su desempeño en evaluaciones externas como el EGEL-CENEVAL, lo que evidencia su calidad académica (UABC, 2024).

En relación con el personal académico, la infraestructura y los servicios, se identifican oportunidades de mejora en el fortalecimiento de la formación docente, la actualización de recursos tecnológicos, la idoneidad de los espacios educativos y la accesibilidad para personas con discapacidad. De igual forma, se señala la necesidad de ampliar el equipamiento y los recursos didácticos para mejorar el proceso de enseñanza-aprendizaje (UABC, 2024).

En conjunto, los resultados de la evaluación interna evidencian la pertinencia del programa educativo, al tiempo que orientan su actualización hacia un modelo más flexible, práctico y alineado con las demandas actuales del entorno académico y profesional. Estas mejoras se centran en la integración curricular, la innovación educativa, el fortalecimiento de las competencias prácticas y la formación integral del estudiante. En la tabla 1. Se muestra la comparativa entre el plan vigente y la propuesta de ajustes para el plan en modificación.

Tabla 1

Comparación entre el plan vigente 2017-1 y la propuesta de mapa curricular.

Plan 2027-1	Plan 2017-2
Se comparte un tronco común de 6 unidades de aprendizaje obligatorias: • Comunicación Oral y Escrita, • Diseño de Algoritmos, • Calculo Diferencial, • Geomería Vectorial y • Algebra Superior, con los programas de Física y Matemáticas Aplicadas.	Se comparte un tronco común de 5 unidades de aprendizaje obligatorias: • Principios de la física, • Diseño de Algoritmos, • Calculo Diferencial, • Matemáticas Superiores y • Tecnología y Sociedad, con los programas de Física, Matemáticas Aplicadas y Ciencia de Datos.
350 créditos distribuidos como: • 255 créditos obligatorios • 85 créditos optativos • 10 práctica profesional	350 créditos distribuidos como: • 298 créditos obligatorios • 42 créditos optativos • 10 práctica profesional
Se cuenta con la asignatura optativa Metodología de la Investigación, ubicada en la etapa disciplinaria, la cuál no garantizaba el desarrollo de proyectos de investigación y no se daba continuidad para una titulación por trabajo terminal.	Se fortalece el área de investigación integrando transversalmente la elaboración de reportes de prácticas e informes técnicos, así como la lectura y discusión de artículos científicos. Además, se incluye una asignatura: Proyecto Científico de Cómputo Inteligente en la etapa terminal que permitirá que las y los estudiantes integren los conocimientos recibidos a lo largo de la profesión para proveer de soluciones innovadoras con apoyo de tecnologías de la IA.
La acreditación de la segunda lengua se hace bajo las siguientes opciones: a) Al quedar asignados al menos en el cuarto nivel del examen diagnóstico de idioma extranjero aplicado por la Facultad de Idiomas. b) Constancia de haber obtenido, en cualquier tiempo y lugar, por lo menos 300 puntos en el examen TOEFL, para el caso del idioma inglés o su equivalente en el caso de otros idiomas.	La acreditación de la segunda lengua se hace bajo las siguientes opciones: a) Obtener al menos 65 puntos (B1+) del examen de diagnóstico de la CGSECE de la UABC. b) Constancia de haber obtenido por lo menos 56 puntos en el examen TOEFL-iBT; o por lo menos 500 puntos en el examen TOEFL-iTP, o el examen TOEIC; o haber obtenido al menos el nivel B1 del CENNI (nivel 10), o de la Certificación Excellence SDSU, o del Curso LEL

Plan 2027-1	Plan 2017-2
c) La acreditación del examen de egreso del idioma extranjero, que se aplica en la Facultad de Idiomas de la UABC. d) La acreditación de por lo menos dos unidades de aprendizaje de un idioma extranjero, impartidos por las propias unidades académicas. e) Acreditar una unidad de aprendizaje de Lectura y Comprensión técnico avanzado del área del conocimiento, de un segundo idioma impartido por la unidad académica correspondiente. f) Estancias internacionales autorizadas por la unidad académica, con duración mínima de tres meses en un país con idioma oficial distinto al español. g) Haber acreditado estudios formales en idioma extranjero.	(Lengua Extranjera en Línea DEXWAY), o del Marco Común Europeo de referencia (MCER) (B1); o al menos el nivel 4.6 de IELTS, o su equivalente. En todos los casos, con una vigencia no mayor a 2 años. c) La acreditación del 5to nivel de idioma de los cursos del CEC (antes UNISER). d) Quedar asignado al menos en el sexto nivel del examen diagnóstico de lengua extranjera aplicado por la Facultad de Idiomas de la UABC. e) La acreditación del examen de egreso de la lengua extranjera, que se aplica en la Facultad de Idiomas de la UABC. f) La acreditación de las unidades de aprendizaje English Level 1 y English Level 2, o de por lo menos dos unidades de aprendizaje disciplinarias obligatorias del plan de estudios impartidas en inglés en la Facultad de Ciencias. g) Estancias internacionales autorizadas por la Facultad de Ciencias, con duración mínima de tres meses en un país con lengua oficial distinta al español. h) Haber acreditado estudios formales en lengua extranjera en instituciones educativas en México o en el extranjero, donde presente certificados de diplomados o estudios de media superior o superior.
Se cuenta con dos unidades de aprendizaje optativas en inglés	Se cuenta con una unidad de aprendizaje obligatoria y una unidad de aprendizaje optativa en inglés
Se oferta la unidad de aprendizaje Emprendedores, como optativa con valor curricular en la etapa terminal, para fomentar la creatividad e iniciativa en las y los estudiantes en esta área.	Además de la unidad de aprendizaje Emprendedores, como optativa con valor curricular en la etapa terminal, se promueve la participación de las y los estudiantes en diversas actividades de emprendedurismo, tales como: • CIMATANK • BOOTCAMP

Plan 2027-1	Plan 2017-2
	• CIMARRONES EMPRENDEDORES - TALLERES DE EMPRENDIMIENTO Y CREATIVIDAD • PROGRAMA DE PREINCUBACIÓN CIMARRONES EMPRENDEDORES
No se especifica de forma explícita la formación de competencias transversales	Se incluye un apartado donde se describen las estrategias a lo largo de la currícula para la adquisición de las 10 competencias transversales recomendadas por el organismo acreditador: CONAIC

Fuente. Elaboración propia

3. Filosofía y enfoque educativo

El nuevo plan de estudios de la Licenciatura en Ciencias Computacionales se sustenta en la filosofía educativa de la Universidad Autónoma de Baja California (UABC), la cual concibe la educación superior como un proceso formativo integral orientado al desarrollo pleno de las personas, la generación y aplicación del conocimiento, y el compromiso con la transformación social. En este marco, la educación universitaria se entiende como un proceso intencional, sistemático y contextualizado que promueve la formación de profesionistas con capacidad crítica, responsabilidad ética y vocación de servicio.

La propuesta curricular se alinea con el Modelo Educativo institucional, que establece los principios, atributos y componentes que orientan la formación universitaria en la UABC (UABC, 2018). Este modelo se fundamenta en corrientes pedagógicas contemporáneas como el humanismo, el constructivismo y la educación a lo largo de la vida, reconociendo al estudiantado como el centro del proceso educativo y como sujeto activo en la construcción de su aprendizaje (UABC, 2018). Desde esta perspectiva, el aprendizaje no se limita a la adquisición de conocimientos, sino que implica la integración de saberes, el desarrollo de habilidades, la formación de valores y la capacidad de actuar de manera pertinente en contextos diversos.

El enfoque humanista orienta la formación hacia el reconocimiento de la dignidad humana, la equidad, la inclusión y el respeto a la diversidad, promoviendo una educación que atiende tanto las dimensiones cognitivas como socioemocionales y éticas del estudiantado. En el diseño del nuevo plan de estudios, este enfoque se traduce en la incorporación de contenidos, estrategias y experiencias formativas que favorecen la responsabilidad social, la sensibilidad ante problemáticas contemporáneas y la formación en valores, así como en la integración de espacios curriculares y modalidades de aprendizaje que promueven la participación activa del estudiantado en contextos reales y diversos. Asimismo, se refleja en el énfasis en el bienestar estudiantil, la atención a trayectorias académicas diferenciadas y la generación de ambientes educativos inclusivos y respetuosos.

Por su parte, el constructivismo fundamenta la organización de experiencias de aprendizaje en las que el conocimiento se construye de manera activa, a partir de la

interacción con el entorno, la reflexión crítica y la resolución de problemas. Este principio se concreta en el plan de estudios mediante la adopción de estrategias como las metodologías activas, el trabajo interdisciplinario, el análisis de casos y la vinculación con escenarios reales, que permiten al estudiantado integrar saberes y desarrollar competencias en situaciones auténticas. De igual forma, se refleja en la estructura progresiva del currículo, en la cual las unidades de aprendizaje están diseñadas para favorecer la construcción gradual del conocimiento, transitando desde fundamentos teóricos hacia su aplicación en contextos complejos.

La educación a lo largo de la vida, como principio transversal, impulsa el desarrollo de capacidades para el aprendizaje autónomo, la actualización continua y la adaptación a contextos cambiantes (UABC, 2018). En el nuevo plan de estudios, este principio se materializa a través de la flexibilidad curricular, la incorporación de unidades de aprendizaje optativas, la diversificación de las modalidades de aprendizaje con obtención de créditos (OMAS) y la promoción de competencias como el autoaprendizaje, la gestión del conocimiento y el uso crítico de tecnologías de la información. Asimismo, se fortalece mediante la integración de experiencias como la movilidad académica, la vinculación con el entorno profesional y el desarrollo de proyectos, que preparan al estudiantado para enfrentar los retos de un contexto dinámico y en constante transformación.

En congruencia con estos fundamentos, el programa educativo adopta un enfoque por competencias, entendido como un modelo formativo que integra de manera articulada conocimientos, habilidades, actitudes y valores en contextos de desempeño, orientando la formación hacia la capacidad de intervenir de manera pertinente en la solución de problemáticas complejas. Este enfoque no se limita a la adquisición de contenidos, sino que privilegia la aplicación del conocimiento, la toma de decisiones fundamentada y la actuación profesional en escenarios reales, favoreciendo la pertinencia y la vinculación con los ámbitos social, productivo y científico.

En el diseño del plan de estudios, este enfoque se traduce en una organización curricular intencionada, en la que cada unidad de aprendizaje contribuye al desarrollo progresivo de las competencias del perfil de egreso. Esto implica una secuencia formativa que transita desde la construcción de fundamentos conceptuales hacia su

integración y aplicación en contextos de mayor complejidad, asegurando la coherencia entre los distintos momentos de la trayectoria académica.

Asimismo, la operacionalización del enfoque por competencias se refleja en la definición de resultados de aprendizaje, en la alineación entre contenidos, actividades formativas y procesos de evaluación, y en la incorporación de experiencias que demandan del estudiantado la movilización integrada de saberes para resolver situaciones auténticas. En este sentido, la evaluación del aprendizaje se orienta a valorar el desempeño en contextos relevantes, mediante evidencias de aprendizaje que permitan identificar el nivel de logro de las competencias, más allá de la memorización de contenidos.

De manera complementaria, este enfoque se articula con la diversificación de escenarios formativos contemplados en el plan de estudios, incluyendo espacios áulicos, de laboratorio, de campo y de vinculación, así como otras modalidades de aprendizaje con obtención de créditos. Estos escenarios favorecen la transferencia del conocimiento, la interdisciplinariedad y la integración teoría-práctica, elementos esenciales para el desarrollo de competencias profesionales

El modelo educativo institucional establece, además, atributos fundamentales que se incorporan de manera transversal en el diseño curricular: la flexibilidad, la formación integral y el sistema de créditos. La flexibilidad curricular permite al estudiantado construir trayectorias formativas acordes con sus intereses, ritmos y necesidades, mediante la combinación de unidades de aprendizaje obligatorias, optativas y otras modalidades de aprendizaje con obtención de créditos. La formación integral favorece el desarrollo equilibrado de las dimensiones académica, ética, social, cultural y profesional. Por su parte, el sistema de créditos reconoce el trabajo académico del estudiantado en diversos contextos formativos, promoviendo la movilidad, la interdisciplinariedad y la diversificación de experiencias de aprendizaje.

El rol del personal docente es fundamental para la implementación efectiva de esta filosofía educativa. En el marco del Modelo Educativo de la UABC, el profesorado se concibe como facilitador, mediador y gestor del aprendizaje, responsable de diseñar experiencias educativas pertinentes, innovadoras y alineadas con las competencias del perfil de egreso (UABC, 2018). En el nuevo plan de estudios, esta función se traduce en

la planeación intencionada de unidades de aprendizaje orientadas al desarrollo de competencias, en las que el docente estructura actividades que integran saberes teóricos y prácticos, promueve el análisis interdisciplinario y vincula el aprendizaje con problemáticas reales del entorno.

Asimismo, el profesorado genera ambientes de aprendizaje activos y colaborativos mediante la implementación de estrategias de enseñanza y de aprendizaje, así como de metodologías activas como el Aprendizaje Basado en Proyectos y Problemas, el análisis de casos, el trabajo en equipo y la integración de experiencias en laboratorio, campo y contextos de vinculación. En este sentido, el docente orienta, acompaña y retroalimenta de manera continua el proceso formativo, favoreciendo la reflexión crítica, la toma de decisiones informada y la aplicación del conocimiento en situaciones auténticas. Su labor también implica el uso pedagógico de tecnologías de la información e inteligencia artificial, la promoción del aprendizaje autónomo y el desarrollo de habilidades transversales en el estudiantado.

De igual forma, el personal docente participa activamente en los procesos de evaluación del aprendizaje desde un enfoque formativo, diseñando instrumentos que permitan valorar el logro de competencias en distintos niveles de complejidad, así como en la evaluación y mejora continua del plan de estudios. Esto incluye su integración en cuerpos académicos, proyectos de investigación, actividades de vinculación y espacios colegiados, desde donde contribuye a la actualización disciplinar, la innovación educativa y la pertinencia del programa frente a las necesidades del entorno.

Por su parte, el estudiantado es concebido como un agente activo, autónomo y corresponsable de su proceso formativo, capaz de gestionar su aprendizaje, tomar decisiones informadas y participar de manera crítica en la construcción del conocimiento. En la operación del nuevo plan de estudios, este papel se materializa en su participación activa en experiencias de aprendizaje que demandan análisis, reflexión, colaboración y aplicación del conocimiento en contextos reales, tales como proyectos interdisciplinarios, prácticas profesionales, actividades de vinculación y otras modalidades de aprendizaje con obtención de créditos.

Esta concepción implica que el estudiantado asume un rol protagónico en la construcción de su trayectoria académica, aprovechando la flexibilidad curricular para

orientar su formación de acuerdo con sus intereses y metas profesionales. Asimismo, desarrolla habilidades para aprender a aprender, trabajar en equipo, comunicarse de manera efectiva en distintos contextos y utilizar de manera crítica herramientas tecnológicas para la gestión del conocimiento.

De manera complementaria, el estudiantado participa en procesos de autoevaluación y coevaluación que fortalecen su capacidad reflexiva y su compromiso con la mejora continua, al tiempo que se involucra en experiencias de movilidad, investigación, emprendimiento y vinculación que amplían su formación y favorecen su inserción en el ámbito profesional. En este sentido, su formación no se limita al aula, sino que se construye a partir de una interacción constante con el entorno social, académico y productivo.

En este sentido, el perfil de egreso del plan de estudios no solo enuncia las competencias a desarrollar, sino que orienta de manera explícita la organización curricular, la selección de contenidos, las estrategias de enseñanza y los mecanismos de evaluación implementados a lo largo de la trayectoria formativa. De esta manera, los fundamentos filosóficos y el modelo educativo institucional se traducen en experiencias de aprendizaje concretas que permiten al estudiantado analizar, intervenir y proponer soluciones en contextos reales, asegurando la coherencia entre la intencionalidad formativa y los resultados del proceso educativo.

3.1. Articulación institucional y del programa educativo

En congruencia con este marco filosófico y educativo, el programa educativo se articula con los referentes institucionales y organizacionales que orientan el quehacer académico de la Universidad y de la unidad académica, asegurando la coherencia entre los principios educativos y la práctica formativa.

Misión y visión de la Universidad Autónoma de Baja California

Misión

Contribuir al desarrollo inclusivo y sostenible, al bienestar de la sociedad bajacaliforniana, la nación y del planeta, a través de la formación integral de profesionistas, de investigadoras, investigadores y ciudadanía comprometida con

una cultura democrática; así como a la generación y difusión de la cultura, del conocimiento y de las tecnologías (UABC, 2023, p. 20).

Visión 2040

La UABC es una universidad líder e innovadora con reconocimiento a nivel nacional y global por ser un agente transformador de progreso social que contribuye a la realización plena del ser humano, al desarrollo incluyente y sostenible, a la profundización de la democracia y a la justicia social a través de la formación en licenciatura, posgrado y a lo largo de la vida, y de sus avances científicos, tecnológicos y culturales (UABC, 2023, p. 21).

Misión y visión de la unidad académica

Misión

La misión de la Facultad de Ciencias de la Universidad Autónoma de Baja California es formar profesionistas de excelencia y de alto nivel competitivo, capaces de aplicar sus conocimientos y habilidades para enfrentar y resolver los retos propios al entorno científico, actual y futuro. La Facultad de Ciencias impulsa la investigación científica en sus diferentes áreas. Además, busca generar conocimiento y extenderlo a la comunidad, llevándolo a su aplicación en el ámbito científico, académico y social. Todo esto con la finalidad de mejorar la calidad de vida en el entorno local, regional, nacional e internacional. Finalmente, fomenta a cada paso los valores culturales, el sentido ético, la responsabilidad social y el respeto al medio ambiente, a través de la extensión de la cultura, la difusión y divulgación del conocimiento.

Visión

La visión de la Facultad de Ciencias es ser una institución reconocida por la competitividad y formación integral de sus egresados, destacada por la calidad y vigencia de sus planes de estudio que responden a la demanda del ámbito científico de los diversos sectores de la sociedad y de la industria en el entorno local y nacional. En la actividad docente, busca tener niveles elevados de formación del personal académico, con un incremento del número de docentes

con grado de maestría y doctorado, aprovechando la vinculación y convenios con sectores dedicados a la ciencia para la formación y actualización constante de los académicos, lo que tiene un impacto favorable en las y los estudiantes, contribuyendo así a la producción científica e intelectual con base en el nuevo modelo educativo.

Al estudiante de la Facultad de Ciencias, además de su valor competitivo, lo destaca su valor que le asigna a la conducta ética, el esfuerzo permanente y el compromiso para responder a las cambiantes realidades de su entorno.

Con esta visión, comprometida a contar con la acreditación de los programas de estudio de las carreras y de laboratorios certificados, se pretende la definición colectiva de programas estratégicos, en el marco de referencia de los nuevos paradigmas y considerando el estado actual de oportunidades, fortalezas y debilidades, para encauzarlas a la realización de su misión.

Misión, visión y objetivos del programa educativo

Misión

El programa de Licenciado en Ciencias Computacionales tiene la misión de formar profesionales con un alto nivel de conocimientos científicos, tecnológicos y sentido de responsabilidad social, comprometido con su entorno, capaz de identificar problemas y ofrecer soluciones, promoviendo la aplicación ética y responsable de las tecnologías de la información y comunicación en beneficio del Estado y del País.

Visión

El programa de Licenciado en Ciencias Computacionales tiene la visión de ser un programa acreditado, atendido por un conjunto de académicos asociados a Cuerpos Académicos consolidados, los cuales están permanentemente actualizando los planes y programas de estudio, integrando en las diferentes asignaturas los nuevos conocimientos y estrategias de aprendizaje resultado de investigaciones científicas y tecnológicas. Así, académicos, estudiantes y egresados conforman profesionales multi e interdisciplinarios que promueven el uso eficaz de recursos humanos y materiales existentes e impulsan y participan en programas de movilidad interinstitucional. Todo lo

anterior les otorga a los grupos así integrados y organizados una visión holística sobre los retos, alcances y oportunidades de su profesión en un contexto regional, nacional e internacional.

Objetivos del programa educativo

Objetivo general

Consolidar un programa educativo en Ciencias Computacionales que articule la formación académica de calidad, la generación y aplicación del conocimiento, y la vinculación con el entorno, para contribuir al desarrollo científico, tecnológico y social, mediante la preparación integral de profesionales, el impulso a la innovación y la atención de problemáticas relevantes en los ámbitos local, nacional e internacional, con responsabilidad ética y compromiso social.

Objetivos específicos

1. Ofrecer una formación académica sólida y actualizada, sustentada en fundamentos científicos y tecnológicos de las ciencias computacionales, alineada con estándares de calidad y procesos de acreditación.
2. Promover la participación estudiantil en actividades académicas, científicas y de movilidad, que fortalezcan su formación integral y visión global.
3. Desarrollar un entorno académico inclusivo, colaborativo y multidisciplinario, que favorezca la innovación, el pensamiento crítico y el trabajo en equipo e interdisciplinario.
4. Promover el uso ético, responsable y sostenible de las tecnologías, considerando su impacto social, económico y ambiental.
5. Fortalecer la generación y aplicación del conocimiento, mediante el desarrollo de proyectos de investigación e innovación en áreas estratégicas como algoritmia, software, sistemas e inteligencia artificial.
6. Impulsar la vinculación con los sectores público, social y productivo, para atender problemáticas reales a través de proyectos de desarrollo tecnológico, prácticas profesionales y colaboración institucional.

7. Fomentar la actualización y superación académica del profesorado, promoviendo su participación en cuerpos académicos, redes de investigación y programas de formación continua.
8. Impulsar la difusión y divulgación del conocimiento, contribuyendo al acceso y apropiación social de la ciencia y la tecnología.

En resumen, la filosofía y el enfoque educativo que sustenta el programa educativo articula los principios del modelo educativo institucional con las necesidades del entorno y las características del campo profesional, consolidando una propuesta curricular pertinente, flexible e innovadora. Esta integración promueve que la formación del estudiantado no solo responda a estándares académicos de calidad, sino que también contribuya al desarrollo sostenible, la equidad social y la transformación responsable de la realidad.

4. Plan de estudios

La estructura del plan de estudios comprende los siguientes apartados: perfil de ingreso, perfil de egreso, campo profesional, características de las unidades de aprendizaje por etapas de formación, características de las unidades de aprendizaje por áreas de conocimiento, mapa curricular, descripción cuantitativa del plan de estudios y tipología de las unidades de aprendizaje.

4.1. Perfil de ingreso

Quien desee ingresar a la Licenciatura en Ciencias Computacionales deberá poseer las siguientes características:

Conocimientos básicos en:

- Fundamentos matemáticos del nivel medio superior: Comprensión de conceptos matemáticos fundamentales y procedimientos aritméticos, algebraicos y geométricos.
- Lenguaje Formal: Conocimiento de las reglas de sintaxis y léxico necesarias para la comunicación académica.

Habilidades:

- Pensamiento lógico-matemático
- Expresión escrita
- Comprensión lectora y analítica
- Manejo básico de hardware y software

Actitudes y valores

- Compromiso
- Ética y responsabilidad
- Trabajo en equipo
- Apertura al aprendizaje autónomo
- Pensamiento crítico

4.2. Perfil de egreso

Las licenciadas y los licenciados en Ciencias Computacionales son profesionales capaces de diseñar y desarrollar soluciones tecnológicas innovadoras mediante el análisis, modelado, implementación y evaluación de proyectos que automatizan procesos y resuelven problemas reales. Se forman en áreas emergentes como inteligencia artificial, ciencia de datos, ciberseguridad y desarrollo de aplicaciones, asegurando la integridad y seguridad de la información. Su trabajo combina pensamiento lógico, creatividad y compromiso ético para responder a las necesidades de la sociedad y de los distintos sectores productivos.

Quien egrese de la Licenciatura en Ciencias Computacionales posee las siguientes competencias:

1. Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.
2. Desarrollar software de base eficiente y robusto mediante la integración de conocimientos en arquitectura y redes de computadoras y los fundamentos de compiladores y sistemas operativos, para optimizar el rendimiento, la gestión de recursos y la comunicación entre sistemas, asegurando aplicaciones funcionales, seguras, escalables y sostenibles, con una actitud ética, responsable y visionaria.
3. Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social.
4. Diseñar y evaluar soluciones de cómputo inteligente aplicando principios de inteligencia artificial, minería de datos, e integración de herramientas computacionales, con el propósito de apoyar la toma de decisiones en situaciones complejas propias de contextos científicos, sociales, industriales o tecnológicos,

actuando con responsabilidad ética, pensamiento crítico, y colaboración interdisciplinaria.

4.3. Campo profesional

Quien egrese del programa educativo de Licenciatura en Ciencias Computacionales pueden desempeñarse en los sectores público, privado y de manera independiente, en actividades relacionadas con el desarrollo de soluciones tecnológicas, la gestión de sistemas y la innovación digital.

Sector público:

- Dependencias gubernamentales y organismos públicos:
 - Análisis y automatización de procesos organizacionales.
 - Desarrollo de sistemas de software para la gestión de información.
 - Administración y mantenimiento de infraestructura tecnológica y redes.
 - Implementación de estrategias de seguridad de la información.
- Instituciones de investigación:
 - Colaboración en equipos multidisciplinarios.
 - Diseño y aplicación de modelos computacionales.
 - Participación en proyectos científicos en áreas como biología, oceanografía, astronomía y meteorología.

Sector privado:

- Empresas dedicadas al desarrollo de software de base o comercial:
 - Desarrollo de aplicaciones y sistemas de software.
 - Análisis, diseño e implementación de soluciones tecnológicas.
 - Gestión de proyectos de desarrollo de software.
 - Implementación de soluciones en inteligencia artificial, ciencia de datos y ciberseguridad.
 - Consultoría, soporte y mantenimiento de sistemas tecnológicos.

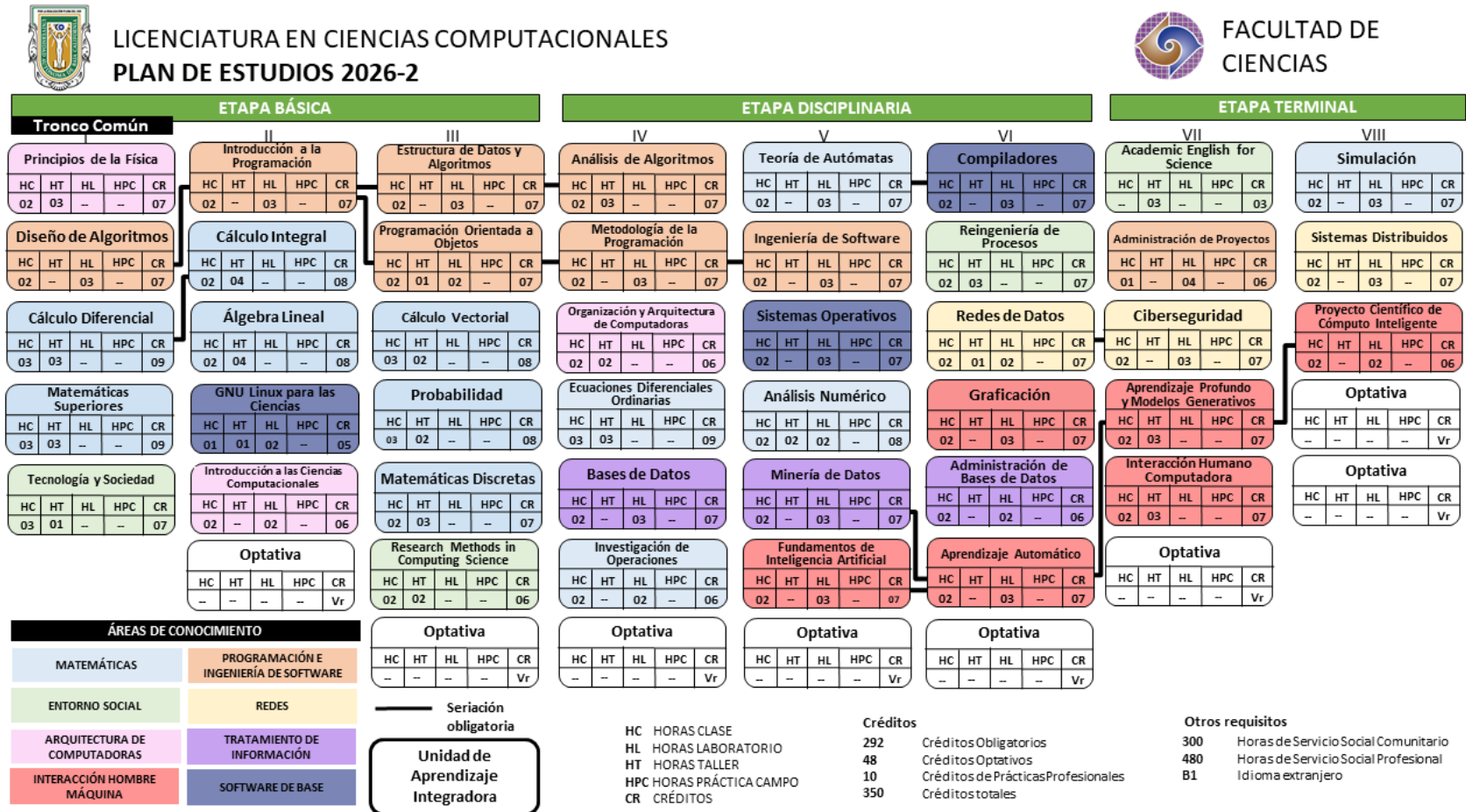
Profesional independiente

- Emprendimiento de proyectos y empresas de base tecnológica.
- Consultoría en tecnologías de la información y desarrollo de software.
- Diseño de soluciones digitales innovadoras para distintos sectores.

4.4. Mapa curricular

Figura 1

Mapa curricular de la Licenciatura en Ciencias Computacionales



Fuente: Elaboración propia

4.5. Distribución de créditos y de unidades de aprendizaje

Distribución de créditos por etapa de formación

Etapas	Obligatorios	Optativos	Total	Porcentajes
Básica	116	12	128	37%
Disciplinaria	126	18	144	41%
Terminal	50	18	68	20%
Prácticas Profesionales	10		10	2%
Total	302	48		100%
Porcentajes	86%	14%	100%	

Distribución de créditos obligatorios por área de conocimiento

Área	Básica	Disciplinaria	Terminal	Total	%
Matemáticas	57	30	7	94	32%
Entorno Social	13	7	3	23	8%
Arquitectura de Computadoras	13	6	0	19	6.5%
Interacción Hombre Máquina	0	21	20	41	14%
Programación e Ingeniería de Software	28	21	6	55	19%
Redes	0	7	14	21	7%
Tratamiento de Información	0	20	0	20	7%
Software de Base	5	14	0	19	6.5%

Distribución de unidades de aprendizaje por etapas de formación

Etapa	Obligatorias	Optativas	Total
Básica	17	2	19
Disciplinaria	18	3	21
Terminal	9	3	12
Total	44	8	52

*Para promover flexibilidad y brindar opciones de formación al estudiantado, se integran en esta propuesta 08 unidades de aprendizaje optativas.

4.6. Características de las unidades de aprendizaje por etapas de formación

Unidad académica: Facultad de Ciencias, Ensenada
Programa educativo: Licenciatura en Ciencias Computacionales
Grado académico: Licenciatura
Plan de estudio: 2027-1

Clave*	Unidad de aprendizaje	HC	HT	HL	HPC	HE	CR	RQ**
<i>Etapa Básica Obligatoria</i>								
01	Principios de la Física	02	03	--	--	02	07	
02	Diseño de Algoritmos	02	--	03	--	02	07	
03	Cálculo Diferencial	03	03	--	--	03	09	
04	Matemáticas Superiores	03	03	--	--	03	09	
05	Tecnología y Sociedad	03	01	--	--	03	07	
06	Introducción a la Programación	02	--	03	--	02	07	02
07	Cálculo Integral	02	04	--	--	02	08	03
08	Álgebra Lineal	02	04	--	--	02	08	
09	GNU Linux para las Ciencias	01	01	02	--	01	05	
10	Introducción a las Ciencias Computacionales	02	--	02	--	02	06	
11	Estructura de Datos y Algoritmos	02	--	03	--	02	07	06
12	Programación Orientada a Objetos	02	01	02	--	02	07	06
13	Cálculo Vectorial	03	02	--	--	03	08	
14	Probabilidad	03	02	--	--	03	08	
15	Matemáticas Discretas	02	03	--	--	02	07	
16	Research Methods in Computing Science	02	02	--	--	02	06	
	<i>Optativa</i>	--	--	--	--	--	Vr	
	<i>Optativa</i>	--	--	--	--	--	Vr	
<i>Etapa Disciplinaria Obligatoria</i>								
17	Análisis de Algoritmos	02	03	--	--	02	07	11
18	Metodología de la Programación	02	--	03	--	02	07	12
19	Organización y Arquitectura de Computadoras	02	02	--	--	02	06	
20	Ecuaciones Diferenciales Ordinarias	03	03	--	--	03	09	
21	Bases de Datos	02	--	03	--	02	07	
22	Investigación de Operaciones	02	--	02	--	02	06	
23	Teoría de Autómatas	02	--	03	--	02	07	
24	Ingeniería de Software	02	--	03	--	02	07	
25	Sistemas Operativos	02	--	03	--	02	07	

Clave*	Unidad de aprendizaje	HC	HT	HL	HPC	HE	CR	RQ**
26	Análisis Numérico	02	02	02	--	02	08	
27	Minería de Datos	02	--	03	--	02	07	
28	Fundamentos de Inteligencia Artificial	02	--	03	--	02	07	
29	Compiladores	02	--	03	--	02	07	23
30	Reingeniería de Procesos	02	03	--	--	02	07	
31	Redes de Datos	02	01	02	--	02	07	
32	Graficación	02	--	03	--	02	07	
33	Administración de Bases de Datos	02	--	02	--	02	06	
34	Aprendizaje Automático	02	--	03	--	02	07	27,28
	<i>Optativa</i>	--	--	--	--	--	Vr	
	<i>Optativa</i>	--	--	--	--	--	Vr	
	<i>Optativa</i>	--	--	--	--	--	Vr	
Etapa Terminal Obligatoria								
35	Academic English for Science	--	03	--	--	--	03	
36	Administración de Proyectos	01	--	04	--	01	06	
37	Ciberseguridad	02	--	03	--	02	07	
38	Aprendizaje Profundo y Modelos Generativos	02	03	--	--	02	07	34
39	Interacción Humano Computadora	02	03	--	--	02	07	
40	Simulación	02	--	03	--	02	07	
41	Sistemas Distribuidos	02	--	03	--	02	07	
42	Proyecto Científico de Cómputo Inteligente	02	--	02	--	02	06	38
	Prácticas Profesionales	--	--	--	--	--	10	
	<i>Optativa</i>	--	--	--	--	--	Vr	
	<i>Optativa</i>	--	--	--	--	--	Vr	
	<i>Optativa</i>	--	--	--	--	--	Vr	
Etapa Optativa								
Etapa Básica Optativa								
43	Geometría Vectorial	01	03	--	--	01	05	
44	Pensamiento Crítico, Ciencia y Humanidades	02	02	--	--	02	06	
45	Comunicación para la Divulgación de las Ciencias	02	02	--	--	02	06	
46	Paradigmas y Lenguajes de Programación	02	--	02	--	02	06	
47	Desarrollo Web	02	--	02	--	02	06	

Clave*	Unidad de aprendizaje	HC	HT	HL	HPC	HE	CR	RQ**
<i>Etapas Disciplinarias Optativas</i>								
48	Programación de Sistemas Empotrados	02	--	03	--	02	07	
49	Desarrollo de Aplicaciones Móviles	01	--	04	--	01	06	
50	Física para videojuegos	02	01	02	--	02	07	
51	Procesamiento Digital de Imágenes	02	--	04	--	02	08	
<i>Etapas Terminales Optativas</i>								
52	Emprendedores	02	--	02	--	02	06	
53	Visión por Computadora	02	--	04	--	02	08	
54	Diseño de Interacción	01	--	04	--	01	06	
55	Algoritmos Bioinspirados	02	--	04	--	02	08	
56	Calidad y Pruebas de Software	02	--	02	--	02	06	
57	Reconocimiento de Patrones	02	--	04	--	02	08	
58	Desarrollo de Aplicaciones Low-Code	02	--	02	--	02	06	

*No es la clave oficial, es una numeración consecutiva asignada para el control, orden y organización de las asignaturas. Cuando el plan de estudios se apruebe por el H. Consejo Universitario, se procede al registro oficial en el Sistema Integral de Planes y Programas de Estudio y Autoevaluación y se le asigna la clave.

** Nomenclatura:

HC: Horas Clase

HL: Horas Laboratorio

HT: Horas Taller

HPC: Horas Prácticas de Campo

HE: Horas Extra clase

CR: Créditos

RQ: Requisitos.

Vr. Variable

4.7. Características de las unidades de aprendizaje por áreas de conocimiento

Unidad académica: Facultad de Ciencias, Ensenada
Programa educativo: Licenciatura en Ciencias Computacionales
Grado académico: Licenciatura
Plan de estudio: 2027-1

Área: Matemáticas								
Clave	Unidad de aprendizaje	HC	HT	HL	HPC	HE	CR	RQ
03	Cálculo Diferencial	03	03	--	--	03	09	
04	Matemáticas Superiores	03	03	--	--	03	09	
07	Cálculo Integral	02	04	--	--	02	08	03
08	Álgebra Lineal	02	04	--	--	02	08	
13	Cálculo Vectorial	03	02	--	--	03	08	
14	Probabilidad	03	02	--	--	03	08	
15	Matemáticas Discretas	02	03	--	--	02	07	
20	Ecuaciones Diferenciales Ordinarias	03	03	--	--	03	09	
22	Investigación de Operaciones	02	--	02	--	02	06	
23	Teoría de Autómatas	02	--	03	--	02	07	
26	Análisis Numérico	02	02	02	--	02	08	
40	Simulación	02	--	03	--	02	07	
Unidades de Aprendizaje Optativas								
43	Geometría Vectorial	01	03	--	--	--	05	

Área: Entorno Social								
Clave	Unidad de aprendizaje	HC	HT	HL	HPC	HE	CR	RQ
05	Tecnología y Sociedad	03	01	--	--	03	07	
16	Research Methods in Computing Science	02	02	--	--	02	06	
30	Reingeniería de Procesos	02	03	--	--	02	07	
35	Academic English for Science	--	03	--	--	--	03	
Unidades de Aprendizaje Optativas								
44	Pensamiento Crítico, Ciencia y Humanidades	02	02	00	00	02	06	
45	Comunicación para la Divulgación de las Ciencias	02	02	00	00	02	06	
52	Emprendedores	02	--	02	--	02	06	

Área: Arquitectura de Computadoras								
Clave	Unidad de aprendizaje	HC	HT	HL	HPC	HE	CR	RQ
01	Principios de la Física	02	03	--	--	02	07	
10	Introducción a las Ciencias Computacionales	02	--	02	--	02	06	
19	Organización y Arquitectura de Computadoras	02	02	--	--	--	06	
Unidades de Aprendizaje Optativas								
48	Programación de Sistemas Empotrados	02	--	03	--	02	07	
50	Física para videojuegos	02	01	02	--	02	07	

Área: Interacción Hombre Máquina								
Clave	Unidad de aprendizaje	HC	HT	HL	HPC	HE	CR	RQ
28	Fundamentos de Inteligencia Artificial	02	--	03	--	02	07	
32	Graficación	02	--	03	--	02	07	
34	Aprendizaje Automático	02	--	03	--	02	07	27,28
38	Aprendizaje Profundo y Modelos Generativos	02	03	--	--	02	07	34
39	Interacción Humano Computadora	02	03	--	--	02	07	
42	Proyecto Científico de Cómputo Inteligente	02	--	02	--	02	06	38
Unidades de Aprendizaje Optativas								
51	Procesamiento Digital de Imágenes	02	--	04	--	02	08	
53	Visión por Computadora	02	--	04	--	02	08	
54	Diseño de Interacción	01	--	04	--	01	06	
55	Algoritmos Bioinspirados	02	--	04	--	02	08	
57	Reconocimiento de Patrones	02	--	04	--	02	08	

Área: Programación e Ingeniería de Software								
Clave	Unidad de aprendizaje	HC	HT	HL	HPC	HE	CR	RQ
02	Diseño de Algoritmos	02	--	03	--	02	07	
06	Introducción a la Programación	02	--	03	--	02	07	02
11	Estructura de Datos y Algoritmos	02	--	03	--	02	07	06
12	Programación Orientada a Objetos	02	01	02	--	02	07	06
17	Análisis de Algoritmos	02	03	--	--	02	07	11
18	Metodología de la Programación	02	--	03	--	02	07	12
24	Ingeniería de Software	02	--	03	--	02	07	
36	Administración de Proyectos	01	--	04	--	01	06	

Área: Programación e Ingeniería de Software								
Clave	Unidad de aprendizaje	HC	HT	HL	HPC	HE	CR	RQ
Unidades de Aprendizaje Optativas								
46	Paradigmas y Lenguajes de Programación	02	--	02	--	02	06	
47	Desarrollo Web	02	--	02	--	02	06	
49	Desarrollo de aplicaciones móviles	01	--	04	--	01	06	
56	Calidad y Pruebas de Software	02	--	02	--	02	06	
58	Desarrollo de Aplicaciones Low-Code	02	--	02	--	02	06	

Área: Redes								
Clave	Unidad de aprendizaje	HC	HT	HL	HPC	HE	CR	RQ
31	Redes de Datos	02	01	02	--	02	07	
37	Ciberseguridad	02	--	03	--	02	07	
41	Sistemas Distribuidos	02	--	03	--	02	07	

Área: Tratamiento de Información								
Clave	Unidad de aprendizaje	HC	HT	HL	HPC	HE	CR	RQ
21	Bases de Datos	02	--	03	--	02	07	
27	Minería de Datos	02	--	03	--	02	07	
33	Administración de Bases de Datos	02	--	02	--	02	06	

Área: Software de Base								
Clave	Unidad de aprendizaje	HC	HT	HL	HPC	HE	CR	RQ
09	GNU Linux para las Ciencias	01	01	02	--	01	05	
25	Sistemas Operativos	02	--	03	--	02	07	
29	Compiladores	02	--	03	--	02	07	23

4.8. Tipología de las unidades de aprendizaje

Unidad académica: Facultad de Ciencias, Ensenada
Programa educativo: Licenciatura en Ciencias Computacionales
Grado académico: Licenciatura
Plan de estudio: 2027-1

Clave	Nombre de la unidad de aprendizaje	Tipo	Observaciones
<i>Etapa Básica Obligatoria</i>			
01	Principios de la Física	3	
	Laboratorio de Principios de Física	2	
02	Diseño de Algoritmos	3	
	Laboratorio de Diseño de Algoritmos	2	
03	Cálculo Diferencial	3	
	Taller de Cálculo Diferencial	2	
04	Matemáticas Superiores	3	
	Taller de Matemáticas Superiores	2	
05	Tecnología y Sociedad	3	
	Taller de Tecnología y Sociedad	2	
06	Introducción a la Programación	3	
	Laboratorio de Introducción a la Programación	2	
07	Cálculo Integral	3	
	Laboratorio de Cálculo integral	2	
08	Álgebra Lineal	3	
	Laboratorio de Álgebra Lineal	2	
09	GNU Linux para las Ciencias	3	
	Taller de GNU Linux para las Ciencias	2	
	Laboratorio de GNU Linux para las Ciencias	2	
10	Introducción a las Ciencias Computacionales	3	
	Laboratorio de Introducción a las Ciencias Computacionales	2	
11	Estructura de Datos y Algoritmos	3	
	Laboratorio de Datos y Algoritmos	2	
12	Programación Orientada a Objetos	3	
	Taller de Programación Orientada a Objetos	2	
	Laboratorio de Programación Orientada a Objetos	2	
13	Cálculo Vectorial	3	
	Taller de Cálculo Vectorial	2	
14	Probabilidad	3	
	Taller de Probabilidad	2	
15	Matemáticas Discretas	3	
	Taller de Matemáticas Discretas	2	

Clave	Nombre de la unidad de aprendizaje	Tipo	Observaciones
16	Research Methods in Computing Science	2	
	Workshop of Research Methods in Computing Science	2	
<i>Etapas Disciplinarias Obligatorias</i>			
17	Análisis de Algoritmos	3	
	Taller de Análisis de Algoritmos	2	
18	Metodología de la Programación	3	
	Laboratorio de Metodología de la Programación	2	
19	Organización y Arquitectura de Computadoras	3	
	Taller de Organización y Arquitectura de Computadoras	2	
20	Ecuaciones Diferenciales	3	
	Taller de Ecuaciones Diferenciales	2	
21	Bases de Datos	3	
	Laboratorio de Bases de Datos	2	
22	Investigación de Operaciones	3	
	Laboratorio de Investigación de Operaciones	2	
23	Teoría de Autómatas	3	
	Laboratorio de Autómatas	2	
24	Ingeniería de Software	3	
	Laboratorio Ingeniería de Software	2	
25	Sistema Operativos	3	
	Laboratorio de Sistema Operativos	2	
26	Análisis Numérico	3	
	Taller de Métodos Numéricos	2	
	Laboratorio de Métodos Numéricos	2	
27	Minería de Datos	3	
	Laboratorio de Minería de Datos	2	
28	Fundamentos de Inteligencia Artificial	3	
	Laboratorio de Fundamentos de Inteligencia Artificial	2	
29	Compiladores	3	
	Laboratorio de Compiladores	2	
30	Reingeniería de Procesos	3	
	Taller de Reingeniería de Procesos	2	
31	Redes de Datos	3	
	Taller de Redes de Datos	2	
	Laboratorio de Redes de Datos	2	
32	Graficación	3	
	Laboratorio de Graficación	2	
33	Administración de Bases de Datos	2	
	Laboratorio de Administración de Bases de Datos	2	

Clave	Nombre de la unidad de aprendizaje	Tipo	Observaciones
34	Aprendizaje Automático	3	
	Laboratorio de Aprendizaje Automático	2	
<i>Etapas Terminal Obligatoria</i>			
35	Academic English for Science	--	No tiene HC
	Workshop of Academic English for Science	2	
36	Administración de Proyectos	3	
	Laboratorio de Administración de Proyectos	2	
37	Ciberseguridad	3	
	Laboratorio de Ciberseguridad	2	
38	Aprendizaje Profundo y Modelos Generativos	3	
	Laboratorio de Aprendizaje Profundo y Modelos Generativos	2	
39	Interacción Humano Computadora	3	
	Laboratorio de Interacción Humano Computadora	2	
40	Simulación	3	
	Laboratorio de Simulación	2	
41	Sistemas Distribuidos	3	
	Laboratorio de Sistemas Distribuidos	2	
42	Proyecto Científico de Cómputo Inteligente	3	
	Laboratorio de Proyecto Científico de Cómputo Inteligente	2	

La tipología de las asignaturas se refiere a los parámetros que se toman en cuenta para la realización eficiente del proceso de aprendizaje integral, tomando en consideración la forma en cómo ésta se desarrolla de acuerdo a sus características, es decir, teóricas o prácticas (laboratorio, taller, clínica o práctica de campo etc.), el equipo necesario, material requerido y espacios físicos en los que se deberá desarrollar el curso, todo ello determinará la cantidad de estudiantes que podrán atenderse por grupo.

De acuerdo con la Guía Metodológica para la Creación, Modificación y Actualización de los Programas Educativos de la Universidad Autónoma de Baja California (UABC, 2010), existen tres tipologías y es importante precisar, que será el rango normal el que deberá predominar para la formación de los grupos; los casos de límite superior e inferior sólo deberán considerarse cuando la situación así lo amerite por las características propias de la asignatura. Así mismo, se deberá considerar la infraestructura de la unidad académica, evitando asignar un tipo 3 (grupo numeroso) a un laboratorio con capacidad de 10 a 12 estudiantes cuya característica es horas clase

(HC) y horas laboratorio (HL). La tipología se designará tomando en cuenta los siguientes criterios:

- Tipo 1. Está considerado para aquellas actividades de la enseñanza en las que se requiere la manipulación de instrumentos, animales o personas, en donde la responsabilidad de asegurar el adecuado manejo de los elementos es del o la docente y donde, además, es indispensable la supervisión de la ejecución del estudiantado de manera directa y continua (clínica y práctica). El rango correspondiente a este tipo es: rango normal = 6 a 10 estudiantes.
- Tipo 2. Está diseñado para cumplir con una amplia gama de actividades de enseñanza aprendizaje, en donde se requiere una relación estrecha para la supervisión o asesoría docente. Presupone una actividad predominante del estudiantado y un seguimiento vigilante e instrucción correctiva docente (talleres, laboratorios): rango normal = 12 a 20 estudiantes.
- Tipo 3. Son asignaturas básicamente teóricas en las cuales predominan las técnicas expositivas; la actividad se lleva a cabo dentro del aula y requiere un seguimiento grupal en el proceso de aprendizaje integral: rango normal = 24 a 40 estudiantes.

4.9. Equivalencias de las unidades de aprendizaje entre planes de estudio

Unidad académica: Facultad de Ciencias, Ensenada
Programa educativo: Licenciatura en Ciencias Computacionales
Grado académico: Licenciatura
Plan de estudio: 2027-1

Clave*	Unidad de aprendizaje Plan de Estudios 2027-1	Clave*	Unidad de aprendizaje Plan de Estudios 2017-2
<i>Etapla Básica Obligatoria</i>			
01	Principios de la física	--	Sin equivalencia
02	Diseño de Algoritmos	23815	Diseño de Algoritmos
03	Cálculo Diferencial	23816	Cálculo Diferencial
04	Matemáticas Superiores	23818	Algebra Superior
05	Tecnología y Sociedad	23851	Aspectos Legales, Sociales y Éticos de la Computación
06	Introducción a la Programación	23821	Introducción a la Programación
07	Cálculo Integral	23822	Cálculo Integral
08	Álgebra Lineal	23823	Álgebra Lineal
09	GNU Linux para las Ciencias	41466	Sistemas Operativos tipo UNIX
10	Introducción a las Ciencias Computacionales	23853	Introducción a las Ciencias Computacionales
11	Estructura de Datos y Algoritmos	23826	Estructura de Datos y Algoritmos
12	Programación Orientada a Objetos	23827	Programación Orientada a Objetos
13	Cálculo Vectorial	23828	Cálculo Vectorial
14	Probabilidad	23829	Probabilidad
15	Matemáticas Discretas	23830	Matemáticas Discretas
16	Research Methods in Computing Science	--	Sin equivalencia
<i>Etapla Disciplinaria Obligatoria</i>			
17	Análisis de Algoritmos	23831	Análisis de Algoritmos
18	Metodología de la Programación	23832	Metodología de la Programación
19	Ecuaciones Diferenciales Ordinarias	23834	Ecuaciones Diferenciales Ordinarias
20	Organización y Arquitectura de Computadoras	23833	Organización y Arquitectura de Computadoras
21	Bases de Datos	23835	Bases de Datos
22	Investigación de Operaciones	23836	Investigación de Operaciones
23	Teoría de Autómatas	23837	Teoría de Autómatas
24	Ingeniería de Software	23838	Ingeniería de Software

Clave*	Unidad de aprendizaje Plan de Estudios 2027-1	Clave*	Unidad de aprendizaje Plan de Estudios 2017-2
25	Sistemas Operativos	23839	Sistemas Operativos
26	Análisis Numéricos	23841	Métodos Numéricos
27	Minería de Datos	23846	Minería de Datos
28	Fundamentos de Inteligencia Artificial	23845	Inteligencia Artificial
29	Compiladores	23842	Compiladores
30	Reingeniería de Procesos	23848	Reingeniería de Procesos
31	Redes de Datos	23844	Redes de Datos
32	Graficación	23840	Graficación
33	Administración de Bases de Datos	--	Sin equivalencia
34	Aprendizaje Automático	33606	Aprendizaje Automático
<i>Etapa Terminal Obligatoria</i>			
35	Academic English for Science	--	Sin equivalencia
36	Administración de Proyectos	23843	Administración de Proyectos
37	Ciberseguridad	23849	Seguridad en Cómputo
38	Aprendizaje Profundo y Modelos Generativos	--	Sin equivalencia
39	Interacción Humano Computadora	23862	Interacción Humano Computadora
40	Simulación	23850	Simulación
41	Sistemas Distribuidos	23847	Sistemas Distribuidos
42	Proyecto Científico de Cómputo Inteligente	--	Sin equivalencia
<i>Etapa Básica Optativa</i>			
43	Geometría Vectorial	23817	Geometría Vectorial
44	Pensamiento Crítico, Ciencia y Humanidades	--	Sin equivalencia
45	Comunicación para la divulgación de la ciencia	--	Sin equivalencia
46	Paradigmas y Lenguajes de Programación	23854	Paradigmas y Lenguajes de Programación
47	Desarrollo Web	23855	Introducción a la Programación en Internet
<i>Etapa Disciplinaria Optativa</i>			
48	Programación de Sistemas Empotrados	23864	Programación de Sistemas Empotrados
49	Desarrollo de Aplicaciones Móviles	23859	Desarrollo Aplicaciones para Dispositivos Móviles
50	Física para Videojuegos	23863	Física para Programadores de Videojuegos
51	Procesamiento Digital de Imágenes	23865	Procesamiento Digital de Imágenes

Clave*	Unidad de aprendizaje Plan de Estudios 2027-1	Clave*	Unidad de aprendizaje Plan de Estudios 2017-2
<i>Etapas Terminal Optativa</i>			
52	Emprendedores	23873	Emprendedores
53	Visión por Computadora	23881	Visión por Computadora
54	Diseño de Interacción	23874	Diseño de Interacción
55	Algoritmos Bioinspirados	23870	Algoritmos Bioinspirados
56	Calidad y Pruebas de Software	23872	Calidad del Software
57	Reconocimiento de Patrones		
58	Desarrollo de Aplicaciones Low-Code	--	Sin equivalencia

5. Estructura y componentes formativos del plan de estudios

El presente capítulo expone la estructura y los componentes formativos del plan de estudios, los cuales han sido diseñados estratégicamente para garantizar una formación pertinente, integral y alineada con las exigencias actuales del entorno social, profesional y disciplinar. La propuesta curricular responde a un enfoque por competencias, orientado a la formación de profesionistas capaces de analizar, intervenir y transformar su contexto mediante la integración de conocimientos, habilidades, actitudes y valores.

La organización del plan de estudios en etapas de formación —básica, disciplinaria y terminal— permite estructurar trayectorias académicas coherentes y progresivas, que transitan desde la adquisición de fundamentos hasta la aplicación del conocimiento en escenarios reales. Esta estructura favorece el desarrollo gradual de competencias, fortaleciendo la vinculación con el entorno y facilitando una transición efectiva hacia el ejercicio profesional.

De manera estratégica, el plan incorpora componentes formativos transversales que responden a las tendencias contemporáneas de la educación superior y a las demandas del campo profesional, tales como el fortalecimiento del idioma inglés, el uso crítico y pertinente de las tecnologías de la información y la inteligencia artificial, así como el desarrollo de habilidades blandas orientadas al liderazgo, la comunicación y la toma de decisiones en contextos complejos.

Asimismo, la propuesta integra mecanismos de flexibilidad curricular, mediante la optatividad orientada a la obtención de certificados y la diversificación de experiencias de aprendizaje, lo que permite al estudiantado construir trayectorias académicas pertinentes y diferenciadas. De igual forma, se incorpora el bienestar estudiantil como un eje central, reconociendo su impacto en la permanencia, el rendimiento académico y la formación integral.

5.1. Etapas de formación

El plan de estudios está compuesto de tres etapas de formación a través de las cuales se forma al estudiantado con enfoque interdisciplinario de manera gradual mediante las unidades de aprendizaje y contenidos buscando desarrollar y proporcionar las competencias propias de la Licenciatura en Ciencias Computacionales, las cuales serán verificables y aplicables a la práctica profesional real que se gesta en el entorno, mismas que podrán ser adecuadas de acuerdo con la evolución y desarrollo de la ciencia y tecnología de su disciplina.

5.1.1. Etapa básica

La etapa de formación básica abarca los tres primeros periodos escolares del plan de estudios y está conformada por 16 unidades de aprendizaje obligatorias y 2 unidades de aprendizaje optativas. Estas asignaturas contribuyen a una formación inicial sólida e integral, con énfasis en el desarrollo del pensamiento lógico, matemático y computacional, así como en la comprensión del contexto social y tecnológico en el que se insertan las soluciones informáticas. Durante esta etapa, el estudiantado adquiere conocimientos fundamentales en programación, estructuras matemáticas, modelación y análisis de problemas, al tiempo que desarrolla habilidades instrumentales para la resolución estructurada de problemas y el uso inicial de herramientas tecnológicas. Asimismo, se promueve una visión crítica sobre el impacto de la tecnología en la sociedad, favoreciendo una formación equilibrada entre lo técnico y lo contextual. En esta etapa, el estudiantado deberá completar 116 créditos obligatorios y 12 créditos optativos. Desde esta etapa, el estudiantado tendrá la oportunidad de participar en cursos y actividades complementarias en áreas de deportes y cultura que fortalezcan su formación integral. Antes de finalizar la etapa básica, el estudiantado deberá acreditar 300 horas de servicio social comunitario.

Competencia de la etapa básica

Desarrollar soluciones computacionales a partir del análisis y modelado de problemas, mediante el uso de fundamentos matemáticos, programación y razonamiento lógico, que permitan la construcción de algoritmos correctos y eficientes, con pensamiento crítico,

disciplina metodológica y creatividad, considerando el impacto de la tecnología en la sociedad.

5.1.2. Etapa disciplinaria

En la etapa disciplinaria, el estudiantado profundiza y consolida los conocimientos teóricos, metodológicos y técnicos de las Ciencias Computacionales, orientándose al diseño y desarrollo de soluciones de complejidad media. Esta etapa promueve la integración de saberes mediante el análisis y optimización de algoritmos, la aplicación de metodologías de programación y la adopción de prácticas de ingeniería de software, en un entorno formativo que articula distintos componentes del área computacional.

A lo largo de los periodos intermedios, se abordan contenidos de mayor complejidad relacionados con la gestión de datos, sistemas y redes, así como la incorporación de fundamentos de inteligencia artificial para el análisis y procesamiento de información. Esto permite al estudiantado fortalecer su capacidad para construir soluciones eficientes, confiables y seguras, enfrentando problemáticas desde una perspectiva estructurada e integral. Se compone de 21 unidades de aprendizaje: 18 obligatorias y 3 optativas, dividiendo sus créditos en 126 créditos obligatorios y 18 optativos.

El estudiantado, habiendo acreditado el servicio social comunitario o primera etapa, podrá iniciar su servicio social profesional al haber cubierto el 60% de avance en los créditos totales del plan de estudios y concluirlo en la etapa terminal de acuerdo con lo que establece el Reglamento de Servicio Social vigente.

Competencia de la etapa disciplinaria

Diseñar y desarrollar soluciones computacionales de complejidad media a partir del análisis y optimización de algoritmos, la aplicación de metodologías de programación y prácticas de ingeniería de software, así como la gestión de datos, sistemas y redes, incorporando fundamentos de inteligencia artificial para el análisis y procesamiento de información, asegurando la eficiencia, confiabilidad y seguridad de los sistemas.

5.1.3. Etapa terminal

La etapa terminal se desarrolla en los dos últimos periodos del programa educativo, en los cuales el estudiantado integra y aplica los conocimientos teóricos y técnicos adquiridos a lo largo de su formación. En esta etapa se incrementa el énfasis en el trabajo práctico, la resolución de problemáticas reales y la participación activa en proyectos que permiten consolidar su perfil profesional en distintos contextos de aplicación. Durante esta fase, se promueve el uso de tecnologías avanzadas de las Ciencias Computacionales para el diseño, implementación y gestión de soluciones innovadoras, incorporando elementos como sistemas distribuidos, simulación, ciberseguridad y diseño centrado en el usuario. Asimismo, mediante el desarrollo de proyectos de cómputo inteligente, el estudiantado fortalece su capacidad para integrar conocimientos, contribuir a la toma de decisiones y atender necesidades reales de organizaciones, actuando con liderazgo, ética profesional y en colaboración con equipos interdisciplinarios.

La etapa se compone de dos períodos que integran 11 unidades de aprendizaje: 8 obligatorias y 3 unidades de aprendizaje optativas con un total de 68 créditos, de los cuales 50 son obligatorios y 18 son optativos, además de 10 créditos obligatorios de las Prácticas Profesionales que el estudiantado debe realizar cuando haya cubierto el 70% de los créditos del plan de estudios, según lo establecido en el Reglamento General para la Prestación de Prácticas Profesionales vigente de la UABC. En esta etapa, el estudiantado podrá realizar hasta dos proyectos de vinculación con valor en créditos con un mínimo de dos créditos optativos cada uno.

Competencia de la etapa terminal

Integrar y aplicar tecnologías avanzadas de las ciencias computacionales para el diseño, implementación y gestión de soluciones innovadoras en entornos reales, mediante el uso de sistemas distribuidos, simulación, ciberseguridad y el diseño centrado en el usuario, así como el desarrollo de proyectos de cómputo inteligente, contribuyendo a la toma de decisiones y a la transformación digital de las organizaciones, con liderazgo, ética profesional y trabajo interdisciplinario.

5.2. Promoción de competencia en el idioma inglés

Las habilidades específicas para el nivel de inglés que solicita el programa educativo, equivalente a B1+ comprenden:

1. Comprender.

- **Comprensión Auditiva:** El estudiante es capaz de comprender las ideas principales cuando el discurso es claro y normal, especialmente sobre asuntos cotidianos en el trabajo, la escuela o el tiempo de ocio. También comprende la idea principal de programas de radio o televisión sobre temas actuales o de interés personal/profesional, siempre que la articulación sea relativamente lenta y clara.
- **Comprensión de Lectura:** Capacidad para comprender textos redactados en una lengua de uso habitual, cotidiano o relacionada con el trabajo. Puede entender descripciones de acontecimientos, sentimientos y deseos en cartas personales.

2. Hablar.

- **Interacción Oral:** El usuario sabe desenvolverse en casi todas las situaciones que se presentan al viajar a lugares donde se habla la lengua. Puede participar espontáneamente en conversaciones sobre temas cotidianos (familia, aficiones, trabajo, viajes).
- **Expresión Oral:** Capacidad para enlazar frases de forma sencilla para describir experiencias, hechos, sueños, esperanzas y ambiciones. Puede explicar y justificar brevemente sus opiniones y proyectos, además de narrar historias o tramas de libros y películas.

3. Escribir.

- **Producción de Textos:** Es capaz de escribir textos sencillos y bien enlazados sobre temas conocidos o de interés personal.
- **Correspondencia:** Puede escribir cartas personales que describen experiencias e impresiones particulares.

Para lograr estas competencias, el programa de Licenciatura en Ciencias Computacionales promueve el fortalecimiento del inglés como segunda lengua de forma progresiva a través de su mapa curricular. El proceso está adaptado a las necesidades del profesional del área de la computación mediante la integración del idioma en diversas unidades de aprendizaje, como se indica a continuación.

En total, se identifican 19 unidades de aprendizaje obligatorias que promueven el desarrollo de competencias en el idioma inglés.

Etapas básicas:

- Introducción a las Ciencias Computacionales
- GNU/Linux para las Ciencias
- Research Methods in Computing Science

Etapas disciplinarias:

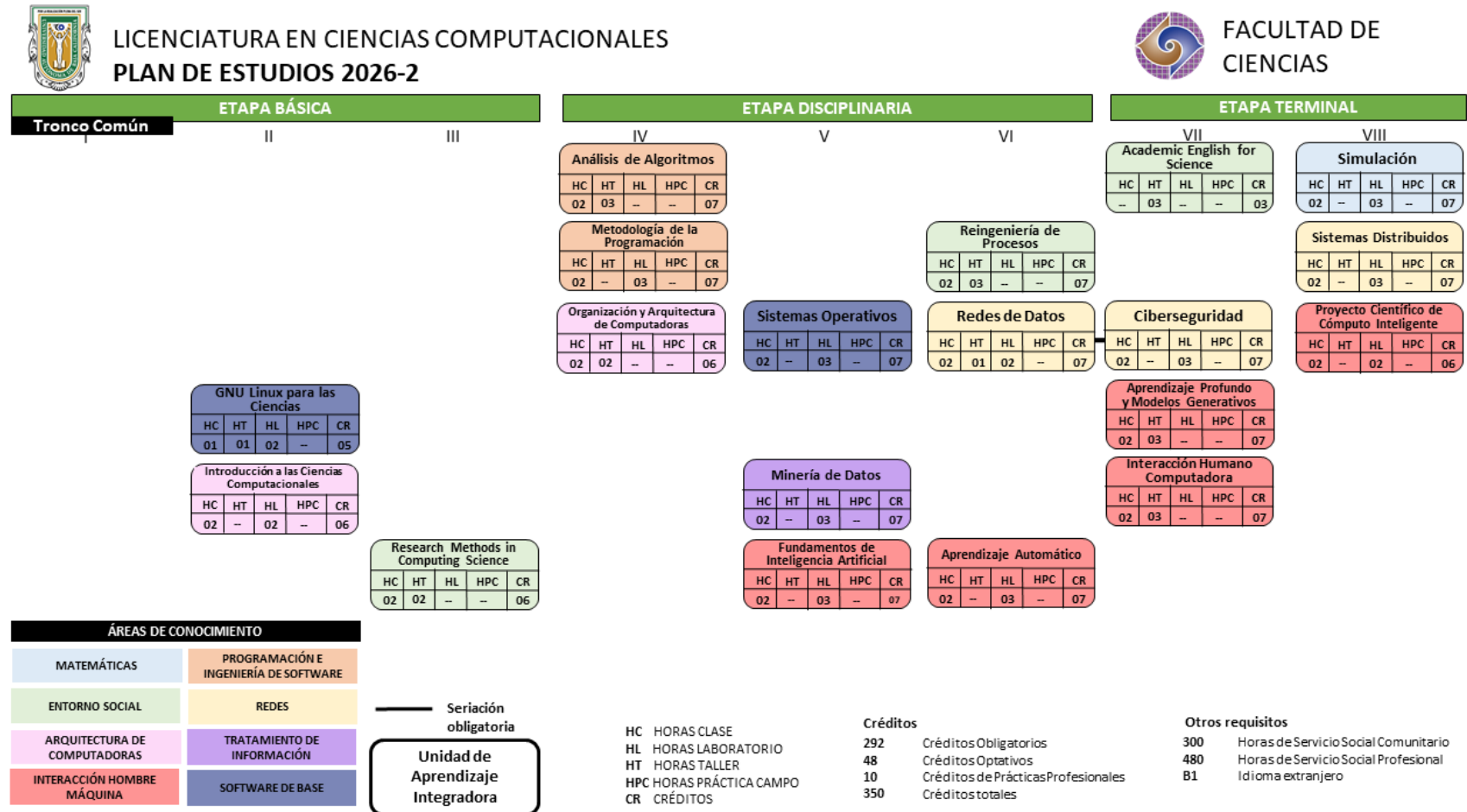
- Análisis de Algoritmos
- Metodología de la Programación
- Organización y Arquitectura de Computadoras
- Sistemas Operativos
- Redes de Datos
- Minería de Datos
- Fundamentos de Inteligencia Artificial
- Aprendizaje Automático
- Reingeniería de Procesos

Etapas terminales:

- Academic English for Science
- Aprendizaje Profundo y Modelos Generativos
- Ciberseguridad
- Interacción Humano-Computadora
- Sistemas Distribuidos
- Simulación
- Proyecto Científico de Cómputo Inteligente

Figura 2

Unidades de aprendizaje que promueven la competencia del idioma inglés



Fuente: Elaboración propia

5.3. Promoción del uso de la TI e IA para el desarrollo de la profesión

A partir de la estructura curricular del programa de Licenciatura en Ciencias Computacionales, se promueve de manera progresiva el uso de Tecnologías de la Información (TI) y, en etapas avanzadas, de Inteligencia Artificial (IA), mediante un conjunto de unidades de aprendizaje distribuidas en las etapas básica, disciplinar y terminal. Estas asignaturas integran experiencias prácticas, desarrollo de software, análisis de datos y diseño de sistemas inteligentes, permitiendo al estudiante evolucionar desde el uso instrumental de herramientas digitales hasta la construcción de soluciones tecnológicas complejas.

En total, se identifican 18 unidades de aprendizaje (obligatorias y optativas) que incorporan directamente el uso de TI e IA.

Etapas básicas:

- Principios de la Física
- Diseño de Algoritmos
- Matemáticas Superiores
- Introducción a la Programación
- Programación Orientada a Objetos
- Estructura de Datos y Algoritmos
- Matemáticas Discretas
- GNU/Linux para las Ciencias
- Research Methods in Computing Science

Etapas disciplinarias:

- Metodología de la Programación
- Ingeniería de Software
- Bases de Datos
- Sistemas Operativos
- Redes de Datos
- Minería de Datos
- Fundamentos de Inteligencia Artificial
- Aprendizaje Automático

- Análisis Numérico
- Reingeniería de Procesos
- Administración de Bases de Datos
- Graficación

Etapas terminal:

- Academic English for Science
- Administración de Proyectos
- Aprendizaje Profundo y Modelos Generativos
- Ciberseguridad
- Interacción Humano-Computadora
- Sistemas Distribuidos
- Simulación
- Proyecto Científico de Cómputo Inteligente

Unidades de aprendizaje que promueven el uso de tecnología para el desarrollo de la profesión



5.4. Promoción de habilidades blandas para el ejercicio de la profesión

Las habilidades blandas, también denominadas competencias socioemocionales y transversales, constituyen un componente esencial en la formación integral del estudiantado de la Licenciatura en Ciencias Computacionales, ya que favorecen su desempeño efectivo en entornos académicos, profesionales y multidisciplinarios. Estas competencias comprenden capacidades como la comunicación, el pensamiento analítico, la resolución de problemas, el trabajo colaborativo, la toma de decisiones y la responsabilidad profesional, las cuales resultan indispensables en el contexto actual de la computación, caracterizado por la complejidad de los sistemas, la rápida evolución tecnológica y la interacción con diversos dominios de aplicación.

En este programa educativo, la promoción de habilidades blandas se alinea con las recomendaciones del organismo acreditador CONAIC y se desarrolla de manera transversal a lo largo de las etapas de formación: básica, disciplinaria y terminal. A diferencia de un enfoque aislado, estas competencias se integran de manera articulada en actividades académicas, prácticas de laboratorio, proyectos y procesos de evaluación, permitiendo su desarrollo progresivo en contextos reales y propios del ámbito computacional.

Si bien el programa tiene un fuerte énfasis en la formación científica, técnica y disciplinar, se reconoce que el ejercicio profesional en Ciencias Computacionales demanda, además, la capacidad de comunicar ideas de manera efectiva, analizar y sintetizar información compleja, modelar soluciones, trabajar en equipo y actuar con responsabilidad ética. Por ello, estas habilidades se incorporan tanto de forma explícita como implícita en distintas unidades de aprendizaje, asegurando su continuidad y transversalidad en el currículo.

En particular, se prioriza el desarrollo de las siguientes competencias:

- Comunicación oral y escrita
- Análisis y síntesis de la información
- Planteamiento y resolución de problemas
- Modelación de soluciones
- Aprendizaje autónomo
- Trabajo en equipo intra e interdisciplinario

- Toma de decisiones
- Uso efectivo de herramientas de TIC
- Responsabilidad profesional y ética
- Visión del impacto de las soluciones tecnológicas

A continuación, se presenta su integración por etapa formativa y su organización dentro del mapa curricular:

Etapla básica:

- Tecnología y Sociedad
- Introducción a la Programación
- Programación Orientada a Objetos
- Matemáticas Superiores
- Diseño de Algoritmos
- Research Methods in Computing Science

Etapla disciplinaria:

- Estructura de Datos
- Análisis de Algoritmos
- Teoría de Autómatas
- Bases de Datos
- Ingeniería de Software
- Redes de Datos
- Sistemas Operativos
- Organización y Arquitectura de Computadoras
- Minería de Datos
- Aprendizaje Automático
- Investigación de Operaciones
- Reingeniería de Procesos

Etapla terminal:

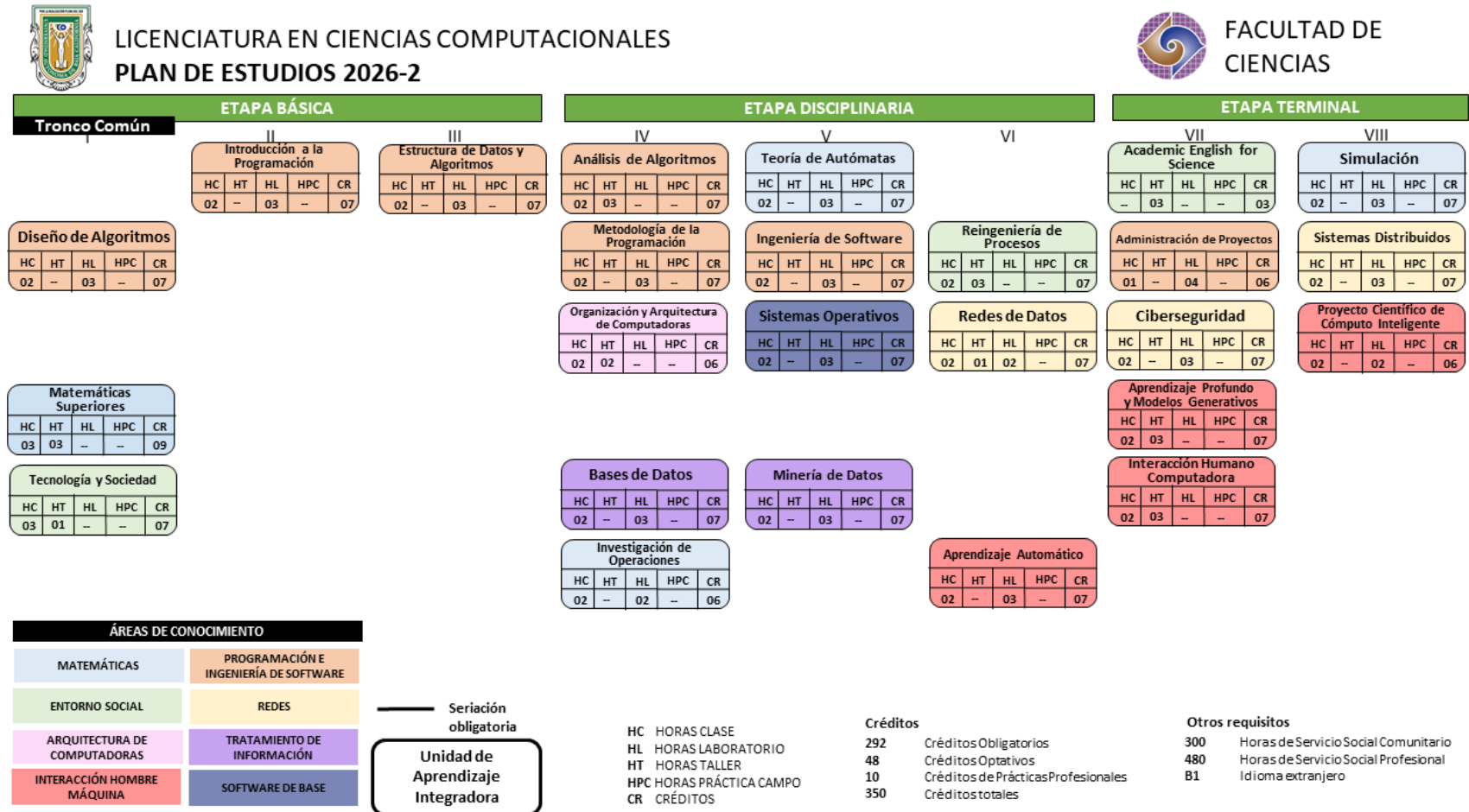
- Administración de Proyectos
- Sistemas Distribuidos

- Simulación
- Interacción Humano Computadora
- Academic English for Science
- Aprendizaje Profundo y Modelos Generativos
- Ciberseguridad
- Proyecto Científico de Cómputo Inteligente

En conjunto, el plan de estudios asegura una progresión coherente en el desarrollo de habilidades blandas, transitando desde la adquisición de fundamentos en comunicación, análisis y aprendizaje autónomo, hacia la integración de estas competencias en la solución de problemas complejos, y finalmente hacia su aplicación en contextos profesionales reales. Esto permite que las personas egresadas no solo posean sólidos conocimientos técnicos, sino también las capacidades necesarias para adaptarse, innovar y colaborar eficazmente en entornos altamente dinámicos.

Figura 4

Unidades de aprendizaje que promueven el uso de las habilidades blandas



Fuente: Elaboración propia

5.5. Propuesta de certificados

La presente propuesta curricular mantiene el principio de flexibilidad del programa educativo, incorporando un esquema estratégico de optatividad orientado al fortalecimiento de áreas clave en las Ciencias Computacionales. Este enfoque tiene como propósito favorecer el desarrollo de competencias especializadas y emergentes, en congruencia con las tendencias actuales del sector tecnológico, así como con las demandas del entorno profesional a nivel nacional e internacional.

Los certificados se conciben como ciclos formativos estructurados y reconocibles, que permiten al estudiantado profundizar en ámbitos específicos de la disciplina y obtener un valor agregado a su formación profesional. Estos ciclos se integran a partir de unidades de aprendizaje optativas y funcionan como énfasis de especialización, promoviendo la diferenciación profesional, la empleabilidad y la vinculación con contextos reales de aplicación tecnológica.

Cada certificado está conformado por un conjunto de al menos tres unidades de aprendizaje que favorecen el desarrollo progresivo de competencias en un área determinada. Para su obtención, el estudiantado deberá cursar y acreditar la totalidad de las asignaturas que integran cada ciclo, lo que permitirá reconocer formalmente su dominio en dicho campo. Asimismo, en función de su trayectoria académica e intereses, el estudiantado podrá obtener más de un certificado, aprovechando las ventajas de la flexibilidad curricular.

La actualización y pertinencia de estos certificados será responsabilidad de la unidad académica, asegurando su alineación con la evolución de la disciplina, las necesidades del sector productivo y las áreas emergentes del conocimiento en computación. En esta propuesta curricular se contemplan los siguientes certificados:

Certificación en Desarrollo de Soluciones Multiplataforma e IoT

- Desarrollo Web
- Desarrollo de Aplicaciones Móviles
- Programación de Sistemas Embebidos

Certificación en Visión por Computadora y Reconocimiento de Patrones

- Procesamiento Digital de Imágenes
- Visión por Computadora
- Reconocimiento de Patrones

5.6. Bienestar estudiantil

El bienestar estudiantil se asume en el presente plan de estudios como un eje estratégico para favorecer la permanencia, el rendimiento académico y la formación integral del estudiantado. En congruencia con el Modelo Educativo de la Universidad Autónoma de Baja California, se promueve una experiencia formativa equilibrada que atiende dimensiones académicas, socioemocionales y de salud, reconociendo que las condiciones en que se desarrolla el aprendizaje inciden directamente en el logro del perfil de egreso.

En este sentido, el diseño curricular considera de manera explícita el equilibrio de la carga académica a lo largo de la trayectoria escolar, con el propósito de evitar la sobrecarga y favorecer procesos de aprendizaje significativos. Para ello, la organización del plan de estudios y de las unidades de aprendizaje toma en cuenta tanto las horas frente a docente como las horas de trabajo independiente (extraclase), asegurando que la dedicación semanal del estudiantado sea razonable, distribuida de forma progresiva y acorde con el nivel de complejidad de cada etapa de formación.

De manera operativa, la asignación de créditos y la planeación de las unidades de aprendizaje se fundamentan en una estimación del tiempo total de dedicación del estudiantado, considerando actividades como estudio independiente, desarrollo de proyectos, prácticas de laboratorio y campo, elaboración de evidencias de aprendizaje y participación en actividades colaborativas. Esta distribución permite que el estudiantado disponga de tiempo suficiente para el cumplimiento de sus actividades académicas, así como para su desarrollo personal, descanso y participación en actividades complementarias.

Asimismo, la secuencia de unidades de aprendizaje en cada semestre se ha diseñado procurando una articulación equilibrada entre carga teórica y práctica, evitando concentraciones excesivas de actividades de alta demanda en periodos específicos. En

la etapa básica se prioriza la adaptación al ritmo universitario y el desarrollo de hábitos de estudio; en la etapa disciplinaria se incrementa gradualmente la complejidad y la integración de saberes; y en la etapa terminal se favorece la dedicación a experiencias formativas en contextos reales, como prácticas profesionales y proyectos de vinculación, con una carga académica flexible que facilita la transición al ámbito laboral.

De manera complementaria, el programa educativo promueve el bienestar del estudiantado a través de estrategias institucionales como el acompañamiento tutorial, las asesorías académicas, la orientación educativa y psicopedagógica, así como el acceso a servicios de salud, actividades culturales, deportivas y de formación en valores. Estas acciones contribuyen a la atención oportuna de necesidades académicas y personales, fortaleciendo la permanencia y el éxito escolar.

Adicionalmente, se fomenta el desarrollo de habilidades para la gestión del tiempo, la autorregulación del aprendizaje y el equilibrio entre la vida académica y personal, mediante la incorporación de actividades formativas y el acompañamiento docente. El profesorado, en su función de facilitador del aprendizaje, considera la carga global del estudiantado al diseñar actividades y evidencias, promoviendo una distribución equilibrada del trabajo académico y evitando la saturación de tareas en periodos críticos.

En conjunto, el bienestar estudiantil se integra como un componente transversal del plan de estudios, que no solo busca reducir factores de riesgo asociados al rezago o abandono escolar, sino también generar condiciones favorables para el aprendizaje, el desarrollo integral y la formación de profesionistas

6. Modalidades de aprendizaje y obtención de créditos

El plan de estudios de la Licenciatura en Ciencias Computaciones, integra diversas modalidades de aprendizaje orientadas a la obtención de créditos y al desarrollo de competencias del perfil de egreso, las cuales se organizan en función de su papel dentro de la trayectoria formativa del estudiantado.

En primer lugar, se encuentran las modalidades curriculares estructuradas, las cuales corresponden a la ruta formativa definida en el mapa curricular y están constituidas por las unidades de aprendizaje obligatorias, optativas y otros cursos optativos. Estas modalidades aseguran la formación disciplinar, metodológica y profesional del estudiantado, y representan el núcleo académico del programa educativo.

En segundo término, se incorporan las Otras Modalidades de Aprendizaje y Obtención de Créditos (OMAS), las cuales permiten la obtención de créditos optativos mediante experiencias formativas flexibles. Estas modalidades favorecen la diversificación de la trayectoria académica, la interdisciplinariedad, la vinculación con el entorno y el desarrollo de competencias en contextos reales, fortaleciendo la formación integral del estudiantado.

Finalmente, el plan de estudios contempla un conjunto de modalidades obligatorias y requisitos institucionales de egreso, tales como las prácticas profesionales, el servicio social en sus dos etapas y la acreditación de una lengua extranjera. Estas experiencias constituyen componentes formativos esenciales para la consolidación del perfil profesional, al articular la formación académica con el ejercicio práctico, la responsabilidad social y la proyección en contextos globales.

6.1. Modalidades curriculares (ruta estructurada del plan de estudios)

Las modalidades curriculares constituyen el eje estructural del plan de estudios, al definir la ruta formativa que el estudiantado sigue a lo largo de su trayectoria académica conforme al mapa curricular. Estas modalidades están integradas por las unidades de aprendizaje obligatorias, optativas y otros cursos optativos, las cuales, en conjunto, garantizan el desarrollo progresivo de las competencias del perfil de egreso.

A través de estas modalidades, se articulan los fundamentos teóricos, metodológicos y aplicados del campo disciplinar, permitiendo la construcción de conocimientos y el desarrollo de habilidades en contextos académicos formales. Asimismo, ofrecen un equilibrio entre la formación común requerida y la posibilidad de orientar la trayectoria académica hacia intereses específicos, mediante la selección de espacios optativos. En este sentido, las modalidades curriculares aseguran la coherencia, secuencialidad y pertinencia del proceso formativo, constituyendo la base sobre la cual se integran las experiencias de aprendizaje complementarias y las estrategias de flexibilidad curricular del programa educativo.

6.1.1. Unidades de aprendizaje obligatorias

Las unidades de aprendizaje obligatorias se distribuyen en las tres etapas de formación que integran el plan de estudios de la Licenciatura en Ciencias Computacionales y constituyen el núcleo estructural de la formación académica. Su diseño responde de manera directa y explícita a las competencias profesionales y específicas definidas en el perfil de egreso, asegurando la coherencia entre la intención formativa del programa y las experiencias de aprendizaje que el estudiantado desarrolla a lo largo de su trayectoria.

En este sentido, cada unidad de aprendizaje ha sido concebida como un espacio formativo orientado al desarrollo progresivo de dichas competencias, integrando conocimientos, habilidades y actitudes que permiten al estudiantado avanzar de manera articulada desde los fundamentos disciplinares hasta la aplicación en contextos complejos. Esta alineación curricular garantiza que, en su conjunto, las unidades de aprendizaje obligatorias constituyan el principal mecanismo para el logro del perfil de egreso.

Asimismo, estas unidades deben ser cursadas y acreditadas de manera obligatoria por el estudiantado (UABC, 2018a), dado que representan los contenidos y experiencias esenciales para la formación profesional. Para este programa educativo, se integran 42 unidades de aprendizaje obligatorias, mediante las cuales el estudiantado obtiene 292 créditos de los 350 que conforman el nuevo plan de estudios, lo que

evidencia su carácter central en el cumplimiento de la formación académica y profesional establecida.

6.1.2. Unidades de aprendizaje optativas

Las unidades de aprendizaje optativas han sido diseñadas en correspondencia con las competencias profesionales y específicas del perfil de egreso, permitiendo al estudiantado complementar, profundizar y orientar su formación hacia áreas de interés particular dentro del campo de las Ciencias Computacionales. En este sentido, estas unidades constituyen espacios curriculares estratégicos que favorecen la especialización, la integración de saberes y la construcción de trayectorias académicas pertinentes.

Asimismo, estas unidades se caracterizan por su flexibilidad, al adaptarse al proyecto académico del estudiantado con el acompañamiento de la tutoría, ofreciendo experiencias de aprendizaje que fortalecen su desempeño profesional en contextos diversos (UABC, 2018b). A través de ellas, el estudiantado puede articular conocimientos disciplinares con intereses específicos, contribuyendo al desarrollo integral de las competencias establecidas en el perfil de egreso.

En esta propuesta curricular se integran ocho espacios optativos en el mapa curricular, distribuidos en las etapas básica, disciplinaria y terminal, lo que permite una progresión formativa coherente y alineada con los distintos niveles de complejidad del proceso de aprendizaje. Estos espacios optativos permiten curricularmente la obtención de los 48 créditos optativos que debe cumplir de acuerdo al plan de estudio, mismo que también podrán ser acreditados con las OMAS descritas en este apartado.

6.1.3. Otros cursos optativos

Los otros cursos optativos constituyen una modalidad curricular orientada a la incorporación de contenidos emergentes, innovaciones disciplinares y temas de relevancia contextual que complementan la formación del estudiantado (UABC, 2018). Estos cursos permiten mantener actualizado el plan de estudios frente a los avances científicos, tecnológicos y sociales, así como responder de manera oportuna a las demandas del entorno y del mercado laboral.

En este sentido, los otros cursos optativos amplían las posibilidades de formación al ofrecer experiencias de aprendizaje adicionales que fortalecen las competencias del perfil de egreso, particularmente en áreas de especialización, formación integral o contextualización disciplinar. Su diseño e incorporación se orienta a enriquecer la trayectoria académica del estudiantado, favoreciendo la pertinencia y actualidad del programa educativo.

Estos cursos podrán integrarse al plan de estudios conforme a los mecanismos institucionales vigentes de evaluación y aprobación académica, asegurando su calidad, pertinencia y viabilidad. Asimismo, se adscribirán a una etapa de formación específica y sus créditos serán considerados como parte de los créditos optativos correspondientes, manteniendo la coherencia curricular del programa educativo. Estos cursos optativos se deberán registrar ante el Departamento de Apoyo a la Docencia y la Investigación del campus.

6.2. Otras Modalidades de Aprendizaje y Obtención de Créditos (OMAS)

Las Otras Modalidades de Aprendizaje y Obtención de Créditos (OMAS) constituyen un conjunto de experiencias formativas flexibles que complementan la ruta curricular estructurada, permitiendo al estudiantado diversificar su trayectoria académica mediante la obtención de créditos optativos en contextos distintos al aula tradicional. Estas modalidades se orientan al desarrollo de competencias del perfil de egreso a través de actividades académicas, de investigación, de vinculación y de formación integral.

Las OMAS favorecen la interdisciplinariedad, la autonomía en el aprendizaje y la vinculación con el entorno social y profesional, al posibilitar la participación del estudiantado en proyectos, actividades institucionales y experiencias en escenarios reales. Su implementación responde a los principios de flexibilidad curricular establecidos en el modelo educativo institucional, ofreciendo alternativas formativas que atienden intereses específicos, ritmos de aprendizaje y necesidades del contexto; fortalecen la formación integral del estudiantado, al complementar la formación disciplinar con experiencias que promueven el pensamiento crítico, la responsabilidad social, la innovación y la empleabilidad.

La operación de estas modalidades se rige por los **Criterios para la operación de diversas modalidades de aprendizaje con obtención de créditos**, emitidos por la Coordinación General de Vinculación y Cooperación Académica (CGVCA), los cuales establecen los lineamientos, mecanismos, procedimientos, temporalidades y condiciones académicas para su registro, desarrollo, seguimiento, evaluación y acreditación. En este sentido, la mayoría¹ de las OMAS deberán registrarse formalmente en el Sistema de Modalidades de Aprendizaje (SIMA), cumpliendo con los requisitos, formatos y procedimientos institucionales establecidos, siendo este registro una condición indispensable para su reconocimiento académico. Asimismo, la operación de estas modalidades es apoyada a nivel campus por el Departamento de Apoyo a la Extensión de la Cultura y la Vinculación (DAECV), instancia responsable de brindar acompañamiento en los procesos de registro, validación, seguimiento y evaluación, en coordinación con las unidades académicas y las instancias institucionales correspondientes.

6.2.1. Estudios independientes

Su propósito es fortalecer la capacidad del estudiantado para estudiar, analizar y aplicar de manera autónoma los fundamentos teóricos y prácticos de un área específica del conocimiento relacionada con su disciplina, mediante la elaboración y ejecución de un plan de trabajo supervisado que contribuya al desarrollo de competencias propias de su perfil profesional.

Es una alternativa de realizar estudios de interés disciplinario no sujeto a la asistencia a clases ni a un programa oficial de unidad de aprendizaje. En esta modalidad, el estudiantado se responsabiliza de manera personal de realizar las actividades descritas en un plan de trabajo que elabora bajo la supervisión y visto bueno de un docente titular que fungirá como asesor. El plan de trabajo debe ser coherente y contribuir a alguna de las competencias específicas del plan de estudios en una temática en

¹ Se excluyen la Movilidad e intercambio estudiantil, las Actividades artísticas, culturales y deportivas y los Cursos intersemestrales.

particular; las actividades contenidas en el plan de trabajo deben contribuir al logro de competencias y los conocimientos teórico-prácticos de la temática especificada.

El estudiantado podrá cursar un Estudio Independiente por periodo escolar, hasta dos a lo largo de su programa educativo. Cada Estudio Independiente tendrá un valor máximo de 6 créditos. En caso de que el estudiante no acredite el Estudio Independiente, deberá inscribirse nuevamente en el mismo estudio, registrándose como parte de su carga académica.

6.2.2. Ayudantía docente

El propósito es fortalecer en el estudiantado el desarrollo de habilidades, conocimientos y herramientas teórico-metodológicas relacionadas con la labor docente, mediante la participación en actividades de apoyo académico que contribuyan al logro de las competencias profesionales del perfil de egreso y a las competencias específicas de su plan de estudios.

La Ayudantía Docente es una modalidad de aprendizaje que permite al estudiantado participar en actividades de apoyo académico dentro de una unidad de aprendizaje que ya hayan cursado y en la cual hayan demostrado un desempeño satisfactorio, con una calificación mínima de 80. Esta modalidad brinda al estudiante la oportunidad de desarrollar habilidades vinculadas al quehacer docente mediante acciones supervisadas, tales como brindar asesorías al grupo, apoyar en la organización y distribución de materiales, colaborar en actividades de clase y acompañar procesos académicos definidos por el docente titular. Las actividades asignadas no sustituyen las funciones del personal docente, sino que constituyen un medio formativo para que el estudiante fortalezca su aprendizaje, desarrolle competencias profesionales y adquiera experiencia en ambientes educativos de manera responsable y guiada.

El estudiantado podrá cursar una Ayudantía Docente por periodo escolar, hasta dos a lo largo de su programa educativo. Cada ayudantía tendrá un valor máximo de 6 créditos cuando la unidad de aprendizaje exceda esta cantidad.

6.2.3. Ayudantía de investigación

Su propósito es fortalecer en el estudiantado el desarrollo de habilidades, conocimientos y herramientas teórico-metodológicas propias del quehacer científico, mediante su participación en actividades de investigación que promuevan el análisis crítico de información, el manejo riguroso de fuentes bibliográficas y la organización autónoma del trabajo académico. Esta modalidad contribuye directamente al perfil de egreso y al logro de las competencias profesionales y específicas definidas en el plan de estudios.

La Ayudantía de Investigación es una modalidad de aprendizaje orientada a proporcionar experiencias formativas en el ámbito de la investigación científica. A través de su participación en proyectos de investigación registrados y dirigidos por personal académico de la Universidad —o por investigadores de otras instituciones con las que exista colaboración formal—, el estudiantado se integra a procesos de análisis, diseño, ejecución o sistematización propios del trabajo científico.

Esta modalidad permite que las y los estudiantes desarrollen competencias investigativas fundamentales, fortalezcan su pensamiento crítico y amplíen su comprensión de los métodos y prácticas que sustentan la generación de conocimiento científico disciplinario y multidisciplinario.

La Ayudantía de Investigación tendrá un valor máximo de 6 créditos, conforme a lo establecido en el plan de estudios del programa educativo. El estudiantado podrá cursar una ayudantía por periodo escolar y acumular hasta dos ayudantías de este tipo en su trayectoria académica.

6.2.4. Ejercicio investigativo

El propósito es favorecer en el estudiantado el desarrollo de experiencias formativas que impulsen la iniciativa, la creatividad y el pensamiento crítico mediante la aplicación de conocimientos, habilidades y actitudes disciplinares en el campo de la investigación a través del aprendizaje autogestivo. Esta modalidad contribuye directamente al perfil de egreso y al fortalecimiento de las competencias profesionales y específicas establecidas en el plan de estudios.

El Ejercicio Investigativo es una modalidad de aprendizaje orientada a promover la creatividad, la capacidad analítica y el interés por la investigación científica a través de

la elaboración de una propuesta formal de investigación. En este ejercicio, el estudiantado integra los fundamentos teóricos, metodológicos y técnicos propios de su área de formación para diseñar un proyecto que responda a una problemática pertinente de su interés; se desarrolla bajo la orientación de una o un profesor-investigador, quien supervisa y valida el cumplimiento de los objetivos académicos y metodológicos establecidos.

El Ejercicio Investigativo tendrá un valor de hasta 6 créditos, de acuerdo con la carga de trabajo académico establecida en el plan de estudios y los productos requeridos. El estudiantado podrá cursar un ejercicio investigativo por periodo escolar y un máximo de dos a lo largo de su programa educativo. En caso de no acreditar la modalidad, deberá reinscribirse al mismo Ejercicio Investigativo.

6.2.5. Ayudantía de laboratorio

El propósito es fortalecer en el estudiantado el desarrollo de habilidades, conocimientos y herramientas técnico-metodológicas propias del trabajo experimental, mediante su participación en actividades de apoyo en espacios de laboratorio que contribuyan al logro de las competencias profesionales del perfil de egreso y de las competencias específicas del plan de estudios.

La Ayudantía de Laboratorio es una modalidad de aprendizaje que permite al estudiantado participar en actividades de apoyo académico en unidades de aprendizaje que presentan un predominio de horas laboratorio respecto a las horas clase, y que previamente hayan cursado y acreditado con un desempeño satisfactorio, con una calificación mínima de 80. Esta modalidad brinda al estudiantado la oportunidad de desarrollar habilidades vinculadas al trabajo experimental y técnico mediante acciones supervisadas, tales como la preparación de materiales y equipos, apoyo en la organización de prácticas, acompañamiento en la ejecución de actividades experimentales, seguimiento de protocolos, y apoyo en el manejo adecuado de insumos, instrumentos y normas de seguridad en laboratorio. Las actividades asignadas no sustituyen las funciones del personal docente, sino que constituyen un medio formativo para que el estudiantado fortalezca su aprendizaje, desarrolle competencias técnicas y

metodológicas, y adquiera experiencia en ambientes de laboratorio de manera responsable, segura y guiada.

La Ayudantía de Laboratorio tendrá un valor máximo de hasta 6 créditos, conforme a lo establecido en la normatividad institucional vigente. El estudiantado podrá cursar esta modalidad conforme a las disposiciones institucionales, considerando que:

- Se podrá registrar una ayudantía por cada unidad de aprendizaje con predominio de horas laboratorio.
- Se permitirá la participación de un estudiante por grupo atendido en la unidad de aprendizaje correspondiente.
- Durante su trayectoria académica, el estudiantado podrá cursar hasta dos ayudantías, ya sea en modalidad docente, de laboratorio o una combinación de ambas, de acuerdo con la normatividad vigente.

6.2.6. Apoyo a actividades de extensión y vinculación

El propósito es promover en el estudiantado experiencias formativas que fortalezcan el desarrollo de habilidades, conocimientos y herramientas teórico-metodológicas en los ámbitos de la extensión y la vinculación universitaria. Esta modalidad busca contribuir al perfil de egreso mediante la participación en actividades orientadas a la atención de la comunidad, la comunicación oral y escrita, el trabajo colaborativo en equipos multidisciplinarios, la organización y planeación de eventos, así como otras acciones que favorezcan su integración al entorno social y profesional.

La modalidad Apoyo a Actividades de Extensión y Vinculación comprende la participación del estudiantado en acciones institucionales dirigidas a acercar el conocimiento científico, tecnológico, humanístico y cultural a los sectores social y productivo. Dichas acciones pueden involucrar actividades diversas relacionadas con la identificación de necesidades, la elaboración de propuestas, la asistencia en programas o proyectos de extensión, la interacción con comunidades y la promoción de la relación Universidad–sociedad. Estas actividades se desarrollan bajo la supervisión de una o un docente responsable y pueden estar integradas a un programa formal de vinculación, asegurando su pertinencia académica y su contribución al desarrollo de competencias profesionales.

La modalidad de Apoyo a Actividades de Extensión y Vinculación tendrá un valor de hasta 6 créditos, de acuerdo con la carga académica y los productos establecidos en el plan de trabajo. El estudiantado podrá cursar una modalidad de extensión y vinculación por periodo escolar y un máximo de dos a lo largo de su trayectoria académica, salvo disposición distinta del plan de estudios. En caso de no acreditar la modalidad, el estudiante deberá reinscribirse en el mismo proyecto registrado durante el periodo inmediato siguiente.

6.2.7. Proyectos de vinculación con valor en créditos (PVVC)

Los Proyectos de Vinculación con Valor en Créditos (PVVC) constituyen una modalidad de aprendizaje orientada a la aplicación, integración y generación de conocimientos en contextos reales, mediante la atención de problemáticas específicas del sector social, productivo o gubernamental. Su propósito es fortalecer el desarrollo de competencias del perfil de egreso a través de la participación del estudiantado en procesos de intervención, investigación aplicada, asistencia técnica o extensión de servicios, en estrecha vinculación con actores externos (UABC, 2018a).

Esta modalidad se caracteriza por su naturaleza integradora, ya que permite articular, de manera simultánea, diversas modalidades de aprendizaje asociadas al currículo, favoreciendo la consolidación de saberes teóricos, metodológicos y prácticos en escenarios auténticos. En este sentido, los PVVC se desarrollan preferentemente en la etapa terminal del plan de estudios, cuando el estudiantado cuenta con los fundamentos disciplinares necesarios para enfrentar situaciones complejas de su campo profesional.

Los proyectos se implementan en colaboración con unidades receptoras pertenecientes a los sectores social y productivo, constituyéndose como espacios formativos donde el estudiantado participa en la solución de problemas reales o en la mejora de procesos, bajo la asesoría, supervisión y evaluación conjunta de un profesor o profesora de tiempo completo o medio tiempo de la unidad académica, así como de un profesionista designado por la instancia receptora (UABC, 2018b).

En términos de su integración académica, los PVVC podrán conformarse por al menos una modalidad de aprendizaje vinculada al currículo, tales como prácticas

profesionales, ejercicio investigativo, apoyo a actividades de extensión o estudios independientes, entre otras. El valor en créditos del proyecto corresponderá a la suma de los créditos de las modalidades que lo integran, más dos créditos adicionales asignados al registro del propio PVVC.

La operación de los PVVC se regirá por un conjunto de criterios y mecanismos institucionales que aseguran su pertinencia, calidad y viabilidad. Para su registro, el estudiantado deberá haber cubierto la totalidad de los créditos obligatorios de la etapa disciplinaria y contar con el servicio social profesional acreditado, o bien encontrarse inscrito en un programa de servicio social profesional con avances comprobables al momento de solicitar su incorporación al proyecto.

El desarrollo de los PVVC se realizará durante la etapa terminal del programa educativo, permitiéndose la inscripción de un proyecto por periodo escolar. Su registro deberá gestionarse en los periodos establecidos ante la instancia responsable de formación profesional y vinculación en el campus correspondiente, previa revisión y aprobación del responsable del programa educativo y de la coordinación correspondiente en la unidad académica.

Cada PVVC contará con la designación de un profesor o profesora responsable, quien tendrá a su cargo la supervisión académica y el seguimiento del proyecto. La evaluación del desempeño del estudiantado será de carácter integral, considerando las valoraciones emitidas tanto por la persona supervisora en la unidad receptora como por el profesorado responsable, así como los criterios adicionales definidos por la unidad académica.

En cuanto a la capacidad operativa, se establece que el profesorado de tiempo completo podrá fungir como responsable de hasta cinco proyectos, atendiendo un máximo de quince estudiantes en total, mientras que el profesorado de medio tiempo podrá coordinar hasta dos proyectos con un máximo de ocho estudiantes. En caso de que un proyecto exceda el número máximo de participantes, podrá asignarse más de un docente responsable. Asimismo, se recomienda formalizar convenios de vinculación con las unidades receptoras, con el fin de asegurar condiciones adecuadas para el desarrollo de las actividades formativas.

El programa contempla la posibilidad de que el estudiantado curse un segundo PVVC, siempre que haya cumplido satisfactoriamente con el primero y que, en sus dos periodos escolares previos, no presente asignaturas reprobadas y mantenga un desempeño académico mínimo de 80 en evaluación ordinaria. Este segundo proyecto deberá registrarse en un periodo posterior a la conclusión del primero, privilegiando aquellos de mayor complejidad e integración (nivel III).

Para efectos de organización y seguimiento, los PVVC se clasifican en tres niveles, definidos en función del número de créditos, horas de dedicación y grado de integración curricular. Esta clasificación permite diferenciar el nivel de complejidad y profundidad de la experiencia formativa, favoreciendo una progresión en el desarrollo de competencias del estudiantado.

Finalmente, los Proyectos de Vinculación con Valor en Créditos se constituyen como un componente estratégico del plan de estudios, al favorecer la transición del ámbito académico al profesional, fortalecer la vinculación con el entorno y consolidar el desarrollo de competencias en contextos reales, contribuyendo de manera significativa al logro del perfil de egreso.

A continuación, se presentan un ejemplo de PVVC para el programa educativo Licenciatura en Ciencias Computacionales.

Ejemplo PVVC Nivel 1

Nombre del Proyecto: Proyecto de desarrollo de aplicaciones móviles y sistemas web.

Descripción: Consiste en el análisis de requerimientos y análisis y desarrollo de aplicaciones web y móviles implementados con modelo MVC y tecnologías como Spring Framework, Spring Boot, Bootstrap, Node JS, PHP, bases de datos relacionales y noSQL, así como la implementación de aplicaciones móviles con tecnologías React Native y otras tecnologías multiplataformas.

Competencia General del Proyecto: Desarrollar aplicaciones web y móviles considerando los estándares y lineamientos de desarrollo del Departamento de Sistemas de la UABC, para garantizar la funcionalidad y la calidad del producto final, con responsabilidad e iniciativa.

Duración: 2 meses

Tabla 4.

Ejemplo del PVVC Proyecto de desarrollo de aplicaciones móviles y sistemas web.

Modalidades de Aprendizaje:	Créditos	Carácter
Unidad de Aprendizaje: Desarrollo Web	6	Optativo
Unidad de Aprendizaje: Desarrollo de Aplicaciones para Dispositivos Móviles	6	Optativo
PVVC: Proyecto de desarrollo de aplicaciones móviles y sistemas web.	2	Optativo
Total:	14	

Fuente: Elaboración propia.

Ejemplo PVVC Nivel 2

Nombre del Proyecto: Proyecto de Estructuración y Desarrollo UNESOFT.

Descripción: Realizar un proyecto estratégico para la empresa COMPU IR MARKETING en conjunción con su afiliada UNEX para estructuración y desarrollo inicial de una área de desarrollo de software Full-Stack para proyectos internos y de clientes para un mejor desempeño en el área. Se llevará a cabo de manera no presencial.

Competencia General del Proyecto: Desarrollar una aplicación Full-Stack, aplicando las mejores prácticas de UX y UI que facilite la coordinación entre desarrolladores y clientes, con responsabilidad e iniciativa.

Duración: 3 meses.

Tabla 3.

Ejemplo del PVVC Proyecto de Estructuración y Desarrollo UNESOFT.

Modalidades de Aprendizaje:	Créditos	Carácter
Unidad de Aprendizaje: Diseño de interacción	6	Optativo
Unidad de Aprendizaje: Desarrollo de Aplicaciones para Dispositivos Móviles	6	Optativo
Unidad de Aprendizaje: Calidad del Software	6	Optativo
PVVC: Proyecto de Estructuración y Desarrollo UNESOFT.	2	Optativo
Total:	20	

Fuente: Elaboración propia.

Ejemplo PVVC Nivel 3

Nombre del Proyecto: Proyecto de Desarrollador de Software Móvil.

Descripción: Durante un periodo de 4 meses, el estudiante formará parte del equipo de desarrollo de la empresa Trimble, contribuyendo al diseño y creación de una aplicación móvil. este proyecto estará orientado al desarrollo front-end utilizando tecnologías modernas como .NET MAUI y el lenguaje de programación C#. El desarrollo se llevará a cabo bajo la metodología ágil SCRUM, lo que implicará una dinámica de trabajo colaborativo con reuniones diarias a través de Google Meet para la actualización del estado del proyecto, la planificación de Sprints y la resolución de problemas. Además, se hará uso de herramientas como GIT para el control de versiones y la colaboración en equipo. A lo largo del proyecto, el practicante no solo aplicará sus conocimientos técnicos, sino que también fortalecerá sus habilidades en trabajo en equipo, comunicación efectiva y resolución de problemas, contribuyendo al éxito del proyecto y adquiriendo experiencia significativa en un entorno profesional de desarrollo de software.

Competencia General del Proyecto: Desarrollar una aplicación móvil utilizando tecnologías modernas y una metodología ágil para la resolución de problemas, colaborando en equipo de forma responsable y creativa.

Duración: 4 meses.

Tabla 5.

Ejemplo del PVVC Proyecto de Desarrollador de Software Móvil.

Modalidades de Aprendizaje:	Créditos	Carácter
Unidad de Aprendizaje: Calidad del Software	6	Optativo
Unidad de Aprendizaje: Desarrollo de Aplicaciones para Dispositivos Móviles	6	Optativo
Unidad de Aprendizaje: Diseño de Interacción	6	Optativo
Práctica Profesional	10	Optativo
PVVC: Proyecto de Desarrollador de Software Móvil.	2	Optativo
Total:	30	

Fuente: Elaboración propia.

6.2.8. Programa de emprendedores universitarios

El propósito es fortalecer en el estudiantado las habilidades de emprendimiento y liderazgo para crear y establecer una empresa. Con esta modalidad se busca apoyar a las y los estudiantes que manifiesten inquietudes con proyectos innovadores por medio de un análisis del perfil emprendedor, la formulación de un plan de negocios, orientación para apoyo financiero y su validación académica, entre otros. El producto es la creación de una empresa que cumpla con tres criterios: (1) demostrar la viabilidad del negocio mediante el Business Canvas Model e incluir el Plan de Negocios elaborado con metodología de la Incubadora Cimarrones Emprendedores UABC; (2) realizarse con la participación de hasta 4 estudiantes adicionales que fungirán como socios, con proporción no menor a 20% cada uno y; (3) establecer la empresa mediante el acta constitutiva certificada por notario público y registro ante SAT.

La modalidad de Actividades para Emprendimiento podrá otorgar hasta seis créditos, de acuerdo con lo establecido en el plan de estudios y en la normatividad vigente. Dado que el proceso de creación y constitución formal de una empresa puede extenderse por más de un semestre, se recomienda que el proyecto sea registrado en SIMA únicamente cuando exista la certeza de que se completará el proceso, o bien cuando la empresa ya se encuentre establecida. Esto con el fin de evitar retrasos en la acreditación o la reprobación de la modalidad.

La acreditación estará sujeta a la entrega y validación de todos los productos requeridos, incluyendo el Business Model Canvas, el Plan de Negocios, la constitución legal de la empresa y el registro fiscal correspondiente, así como el cumplimiento de las actividades académicas y asesorías establecidas.

6.2.9. Unidades de aprendizaje por asesoría académica

El propósito es brindar al estudiantado una alternativa formativa que garantice la continuidad de su trayectoria académica mediante el acompañamiento docente en el desarrollo de unidades de aprendizaje obligatorias u optativas que, por condiciones de oferta académica, no se encuentren disponibles en el periodo correspondiente, contribuyendo al logro oportuno de las competencias del perfil de egreso.

Las Unidades de Aprendizaje por Asesoría Académica constituyen una modalidad de aprendizaje orientada a atender situaciones excepcionales en las que el estudiantado requiere cursar una unidad de aprendizaje obligatoria u optativa del plan de estudios que ha sido previamente autorizada por la unidad académica y que no se encuentra ofertada en el semestre en que se solicita.

Esta modalidad se desarrolla bajo un esquema de acompañamiento académico personalizado, en el cual el profesorado responsable orienta, supervisa y evalúa el proceso de aprendizaje del estudiantado, asegurando el cumplimiento de los contenidos, objetivos y competencias establecidos en el programa de la unidad de aprendizaje correspondiente. Su implementación permite garantizar la continuidad académica, evitar rezagos en la trayectoria escolar y favorecer el avance oportuno hacia el egreso, manteniendo la calidad y rigor académico del proceso formativo.

Las unidades de aprendizaje cursadas bajo esta modalidad conservarán el valor en créditos establecido en el plan de estudios para cada unidad de aprendizaje correspondiente, ya sea obligatoria u optativa; cada estudiante podrá darse de alta en un máximo de dos unidades de aprendizaje por asesoría académica por semestre. Esta modalidad deberá solicitarse mediante oficio utilizando el formato vigente, con la información capturada por el profesorado responsable (nunca por el estudiantado), dirigido al Departamento de Apoyo a la Extensión de la Cultura y la Vinculación (DAECV) del campus correspondiente.

6.2.10. Actividades para la formación en valores

El propósito es fortalecer en el estudiantado la reflexión ética y axiológica mediante la participación en actividades que promuevan la vivencia y práctica de los valores universales y de la comunidad universitaria, contribuyendo así a su formación integral como personas, ciudadanos y profesionistas con responsabilidad social y sentido ético.

Esta modalidad busca propiciar espacios de aprendizaje y convivencia en los que las y los estudiantes reflexionen sobre los valores éticos y universales que orientan la conducta humana y la vida social. A través de experiencias formativas, de servicio, diálogo y trabajo colaborativo, se fomenta la práctica de los valores fundamentales de la Universidad Autónoma de Baja California —confianza, democracia, honestidad,

humildad, justicia, lealtad, libertad, perseverancia, respeto, responsabilidad y solidaridad—, promoviendo así el desarrollo de una conciencia ética que fortalezca su compromiso ciudadano y profesional.

La modalidad de Actividades para la Formación en Valores otorga un valor de 6 créditos, los cuales podrán acreditarse exclusivamente durante la etapa básica del programa educativo. El otorgamiento de estos créditos no se rige por el esquema tradicional de “ocho actividades por un crédito”, debido a que esta modalidad corresponde a un proceso formativo integral, orientado a la reflexión ética, la vivencia de valores y la participación en experiencias comunitarias y colaborativas.

Los 6 créditos se acreditarán una vez que el estudiante: (1) participe en las actividades programadas por la unidad académica; (2) cumpla con la asistencia y los productos formativos o evidencias establecidas; y (3) · Sea evaluado conforme a los criterios definidos para la modalidad.

La formación integral del estudiantado en la Licenciatura en Ciencias Computacionales incorpora de manera explícita el desarrollo de valores éticos, sociales y profesionales, reconociendo que el ejercicio de la computación tiene implicaciones directas en la sociedad. En este sentido, el programa educativo promueve un enfoque axiológico que favorece la construcción de una identidad profesional basada en principios como la honestidad, la responsabilidad, el respeto, la justicia, la solidaridad y el compromiso social, en concordancia con los valores institucionales de la UABC. Esta formación se articula tanto en el ámbito curricular como en el extracurricular. En el plano curricular, se integra de manera transversal en diversas unidades de aprendizaje, destacando particularmente **Tecnología y Sociedad**, donde el estudiantado analiza de forma crítica las implicaciones éticas, legales y sociales del desarrollo tecnológico. Asimismo, en distintas asignaturas se promueve el actuar responsable, la toma de decisiones fundamentada y la reflexión sobre el impacto de las soluciones computacionales.

De acuerdo con la normativa institucional, el plan de estudios contempla actividades con valor en créditos dentro de la etapa básica, orientadas a fortalecer la formación en valores. Estas actividades generan espacios de reflexión y participación que contribuyen al desarrollo personal y social del estudiantado, favoreciendo su

formación como ciudadanos responsables y profesionistas con sentido ético. De manera complementaria, la institución ofrece programas, normativas y mecanismos que fortalecen este eje formativo, entre los que destacan el **Código de Ética institucional**, la **Defensoría de los Derechos Universitarios**, la **Unidad de Género, Diversidad e Inclusión Educativa**, así como protocolos y campañas orientadas a la prevención de la violencia, la promoción de los derechos humanos, la equidad y la cultura de paz. Estas instancias brindan un marco de referencia y acompañamiento para garantizar el respeto a la dignidad humana y la convivencia armónica dentro de la comunidad universitaria.

Adicionalmente, se promueve la participación del estudiantado en actividades formativas y de vinculación que fortalecen los valores en la práctica, tales como:

- Semana de valores y semana cultural
- Actividades de atención a la comunidad y a grupos en situación de vulnerabilidad
- Actividades deportivas y de bienestar integral
- Concursos y competencias académicas
- Actividades de divulgación científica

En conjunto, estas acciones permiten consolidar una formación ética y socialmente responsable, asegurando que las personas egresadas no solo cuenten con competencias técnicas sólidas, sino también con los valores necesarios para contribuir de manera positiva a la sociedad y al desarrollo tecnológico responsable.

6.2.11. Movilidad e intercambio estudiantil

El propósito es fortalecer la formación integral del estudiantado mediante su participación en experiencias académicas en otras instituciones de educación superior, nacionales o extranjeras, que contribuyan al desarrollo de competencias profesionales, interculturales y de adaptación a contextos diversos, en correspondencia con el perfil de egreso del programa educativo.

La movilidad e intercambio estudiantil es una modalidad de aprendizaje que permite al estudiantado cursar unidades de aprendizaje, realizar prácticas profesionales u otras actividades académicas en instituciones de educación superior distintas a su unidad académica de origen, ya sea en el ámbito intrainstitucional (entre programas,

unidades académicas o dependencias de la UABC) o interinstitucional (en instituciones nacionales o extranjeras).

Esta modalidad incluye tanto esquemas de movilidad como de intercambio estudiantil, entendiéndose este último como una forma de movilidad que implica, de manera esencial, una acción recíproca entre instituciones. A través de estas experiencias, el estudiantado tiene la oportunidad de integrarse a contextos académicos, culturales y profesionales distintos, favoreciendo la adquisición de nuevas competencias, el fortalecimiento de la autonomía, la capacidad de adaptación y la maduración personal y académica (UABC, 2018b).

La movilidad intrauniversitaria permite al estudiantado cursar unidades de aprendizaje en otras unidades académicas de la propia Universidad, así como participar en proyectos académicos o de investigación, aprovechando los recursos institucionales disponibles. Por su parte, la movilidad interuniversitaria se desarrolla mediante convenios de colaboración establecidos con instituciones de educación superior nacionales e internacionales, lo que posibilita el reconocimiento académico de las actividades realizadas a través de equivalencias, conversión o transferencia de créditos.

La operación de esta modalidad se sustenta en los mecanismos institucionales de vinculación y cooperación académica, a través de los cuales se promueven convocatorias, programas de intercambio y esquemas de acompañamiento al estudiantado. En este sentido, las unidades académicas fomentan activamente la participación en programas de movilidad, estableciendo estrategias que favorezcan el acceso, permanencia y aprovechamiento de estas experiencias formativas.

Las actividades realizadas en el marco de la movilidad e intercambio estudiantil podrán ser acreditadas mediante equivalencias, conversión o transferencia de créditos, conforme a la normatividad institucional vigente y a los dictámenes académicos correspondientes.

El número de créditos a otorgar dependerá de las unidades de aprendizaje cursadas, las actividades realizadas y su correspondencia con el plan de estudios del programa educativo. La acreditación estará sujeta a la validación por parte de la unidad académica, asegurando la pertinencia, equivalencia y calidad académica de las experiencias formativas desarrolladas.

Tabla 7

Universidades de países extranjeros con las que se establecen convenios para movilidad.

País	Institución
Alemania	Mainz University of Applied Sciences
Chile	Universidad Tecnológica Metropolitana
Colombia	Universidad Autónoma Latinoamericana
Costa Rica	Universidad Interamericana de las Américas
Corea del Sur	Seoul National University of Science & Technology
España	Universidad de la Coruña, Universidad de Vigo, Universidad Politécnica de Catalunya
Estados Unidos	San Diego State University, Southwestern College City University of New York – Lehman College
Francia	Université de Technologie Tarbes Occitanie Pyrénées – Institut Universitaire de Technologie, Université de Technologie Tarbes Occitanie Pyrénées – École National d'Ingénieurs de Tarbes
Hungría	Kodolanyi Janos University of Applied Sciences
Italia	Università Degli Studi Di Perugia
Japón	Meijo University
Perú	Universidad Peruana de los Andes, Universidad Científica del Sur
Paraguay	Universidad Autónoma de Asunción, Universidad Nacional de Asunción
Portugal	Instituto Politécnico de Bragança
Rusia	ITMO University
Taiwán	Macau University of Science and Technology
Turquía	Istanbul Aydin University
Uruguay	Universidad de la República de Uruguay, Universidad de Montevideo

Fuente: Elaboración propia a partir de la información de la Coordinación General de Vinculación y Cooperación Académica de la UABC.

Tabla 8*Universidades de México con las que se establecen convenios para movilidad*

Estado	Institución
Aguascalientes	Instituto Tecnológico de Aguascalientes, Universidad Autónoma de Aguascalientes
Baja California Sur	Universidad Autónoma de Baja California Sur
Campeche	Instituto Tecnológico de Campeche Universidad Autónoma de Campeche Universidad Autónoma del Carmen
Chiapas	Universidad Autónoma de Chiapas Universidad de Ciencias y Artes de Chiapas
Chihuahua	Universidad Autónoma de Chihuahua Universidad Autónoma de Ciudad Juárez
Coahuila	Universidad Autónoma de Coahuila
Colima	Instituto Tecnológico de Colima, Universidad de Colima
Durango	Universidad Juárez del Estado de Durango
Estado de México	Instituto Tecnológico de Toluca Tecnológico de Estudios Superiores de Ecatepec Universidad Autónoma del Estado de México
Guerrero	Universidad Autónoma de Guerrero
Hidalgo	Universidad Autónoma del Estado de Hidalgo Universidad Politécnica de Francisco I. Madero Universidad Tecnológica de Tula Tepeji
Jalisco	Universidad de Guadalajara
Michoacán	El Colegio de Michoacán, A. C. Universidad Michoacana de San Nicolás de Hidalgo.
Morelos	Universidad Autónoma del Estado de Morelos
Nayarit	Universidad Autónoma de Nayarit
Nuevo León	Universidad Autónoma de Nuevo León
Oaxaca	Instituto Tecnológico de Oaxaca Universidad Autónoma "Benito Juárez" de Oaxaca
Puebla	Benemérita Universidad Autónoma de Puebla Universidad Popular Autónoma del Estado de Puebla Universidad Valle de Puebla
San Luis Potosí	Universidad Autónoma de San Luis Potosí Universidad Cuauhtémoc Campus San Luis Potosí
Sinaloa	Universidad Autónoma de Sinaloa Universidad Politécnica de Sinaloa
Sonora	Centro de Estudios Superiores del Estado de Sonora, El Colegio de Sonora, Instituto Tecnológico de Sonora, Universidad de Sonora, Instituto Tecnológico Superior de Cajeme
Querétaro	Universidad Autónoma de Querétaro
Quintana Roo	Universidad de Quintana Roo Universidad del Caribe
Tabasco	Universidad "Juárez" Autónoma de Tabasco
Tamaulipas	Universidad Autónoma de Tamaulipas
Veracruz	Universidad Veracruzana

Yucatán	Centro de Investigación Científica de Yucatán, A. C.
Zacatecas	Universidad Autónoma de Zacatecas

Fuente: Elaboración propia a partir de la información de la Coordinación General de Vinculación y Cooperación Académica de la UABC.

6.2.12. Actividades artísticas, culturales y deportivas

El propósito es brindar al estudiantado experiencias formativas que fortalezcan su desarrollo integral mediante la participación en actividades artísticas, culturales y deportivas. Esta modalidad busca fomentar la creatividad, la apreciación estética, la expresión artística, la identidad cultural, el trabajo colaborativo y la formación de hábitos de vida saludable, contribuyendo al desarrollo de competencias personales y profesionales acordes al perfil de egreso.

Las Actividades Artísticas, Culturales y Deportivas constituyen una modalidad formativa orientada a promover la participación del estudiantado en acciones relacionadas con el arte, la cultura y el deporte. Estas actividades incluyen talleres, grupos artísticos, programas de promoción cultural, prácticas deportivas y otros espacios que fortalecen las capacidades creativas, expresivas y físicas de quienes participan. El estudiantado puede cursar esta modalidad en su unidad académica de adscripción o en otras unidades académicas de la UABC, de acuerdo con la oferta institucional y la programación de actividades curriculares establecidas.

La obtención de créditos dependerá del tipo de actividad cursada.

- a. Actividades Complementarias de Formación Integral (I, II y III):
 - Se otorga 1 crédito por cada ocho actividades complementarias registradas en el carnet correspondiente.
 - Se pueden obtener hasta 2 créditos por periodo escolar mediante esta vía.
- b. Actividad Deportiva I y II / Actividad Cultural I y II:
 - Se podrá obtener crédito siempre y cuando:
 - La participación sea individual.
 - La actividad no haya sido acreditada previamente por otra modalidad.
 - Exista aprobación del comité de la unidad académica.
 - La actividad esté debidamente registrada y avalada por la UABC.

- c. Cursos acreditables de la Facultad de Artes y la Facultad de Deportes:
 - El estudiantado puede cursar actividades ofertadas por estas facultades para la obtención de créditos, conforme a la normatividad vigente.

6.2.13. Cursos intersemestrales

Su objetivo es ofrecer al estudiantado la oportunidad de cursar unidades de aprendizaje obligatorias u optativas en un periodo intersemestral, con el propósito de favorecer el avance en su trayectoria académica y la regularización de su plan de estudios, conforme a la normatividad universitaria vigente.

Esta modalidad de aprendizaje se desarrolla en periodos intersemestrales y permite al estudiantado cursar unidades de aprendizaje con una carga intensiva y concentrada; está orientada a facilitar la continuidad académica, el aprovechamiento del tiempo entre ciclos escolares y la optimización del avance curricular. Por sus características, los cursos intersemestrales se imparten con un máximo de cinco horas presenciales por día, incluyendo actividades de clase, laboratorio o taller; no aplican para unidades de aprendizaje que contemplen prácticas de campo, y se ofertan bajo un esquema autofinanciable, conforme a lo establecido en el Estatuto Escolar de la UABC.

El estudiantado interesado deberá cumplir con los requisitos académicos y administrativos determinados por la unidad académica correspondiente, pudiendo cursar hasta dos unidades de aprendizaje por periodo intersemestral, de acuerdo con las disposiciones institucionales; y obtendrá el número de créditos correspondiente a cada unidad de aprendizaje inscrita, conforme al plan de estudios de su programa educativo.

La acreditación de los créditos se otorgará una vez que el o la estudiante cumpla con los criterios de evaluación establecidos por el docente y apruebe el curso intersemestral. Los cursos intersemestrales no modifican el valor crediticio de las unidades de aprendizaje, dado que se rigen por los mismos programas, cargas horarias y criterios académicos establecidos para los periodos semestrales ordinarios.

6.3. Modalidades obligatorias y requisitos institucionales de egreso

El plan de estudios incorpora, además de las modalidades curriculares y flexibles, un conjunto de modalidades obligatorias y requisitos institucionales de egreso que

constituyen componentes esenciales para la formación profesional del estudiantado. Estos incluyen las prácticas profesionales, el servicio social en sus etapas comunitaria y profesional, la acreditación de una lengua extranjera y mecanismos de titulación.

Estas modalidades tienen como propósito articular la formación académica con la experiencia en contextos reales, fortalecer la responsabilidad social universitaria y favorecer el desarrollo de competencias profesionales en escenarios laborales y globales. A través de su cumplimiento, el estudiantado consolida aprendizajes adquiridos a lo largo de su trayectoria, aplica conocimientos en situaciones auténticas y desarrolla habilidades clave para su inserción en el campo profesional.

En conjunto, estas modalidades aseguran que las personas egresadas cuenten con una formación integral, pertinente y alineada con las demandas del entorno, contribuyendo a su desempeño eficaz y ético en el ámbito socioambiental.

6.3.1. Servicio social comunitario y profesional

La Universidad Autónoma de Baja California, con fundamento en su Reglamento de Servicio Social vigente, establece como requisito para el estudiantado de licenciatura la realización del servicio social en dos etapas: comunitaria y profesional. Estas etapas tienen como propósito contribuir a la formación integral del estudiantado mediante la aplicación de conocimientos en beneficio de la sociedad, particularmente de los sectores en condición de vulnerabilidad.

La primera etapa, denominada servicio social comunitario, comprende un mínimo de 300 horas y está orientada al fortalecimiento de la formación valoral y el compromiso social del estudiantado. De acuerdo con el reglamento, esta etapa puede iniciarse desde el ingreso a la universidad y se recomienda su realización durante la etapa básica del programa educativo. Asimismo, se establece que, al alcanzar el 40% de los créditos del plan de estudios sin haber acreditado esta etapa, la carga académica se verá limitada conforme a la normatividad institucional.

La segunda etapa, denominada servicio social profesional, comprende un mínimo de 480 horas, con una duración no menor a seis meses, y está orientada a la aplicación de conocimientos, habilidades y competencias propias del perfil profesional. Esta etapa podrá iniciarse una vez que el estudiantado haya acreditado el servicio social

comunitario y cubierto al menos el 60% de los créditos del plan de estudios, en concordancia con lo establecido en el reglamento vigente.

La operación del servicio social en ambas etapas será responsabilidad de la unidad académica, la cual se encargará de la planeación, asignación, supervisión, evaluación y acreditación de las actividades realizadas, en coordinación con las unidades receptoras de los sectores público, social y productivo. Para su realización, el estudiantado deberá cumplir con los requisitos institucionales, incluyendo la acreditación del taller de inducción, la asignación formal a un programa registrado y la entrega de informes de actividades debidamente validados.

6.3.2. Prácticas profesionales

Las prácticas profesionales constituyen una estrategia formativa orientada a la aplicación de conocimientos, habilidades y actitudes en contextos reales del ámbito laboral, contribuyendo al fortalecimiento de la formación integral del estudiantado y a su vinculación con los sectores social, productivo y gubernamental.

De acuerdo con el Reglamento General para la Prestación de Prácticas Profesionales de la Universidad Autónoma de Baja California, estas actividades podrán realizarse en unidades receptoras debidamente registradas, tales como organismos públicos, privados o sociales, cuyas funciones se relacionen con el campo disciplinar del programa educativo. Asimismo, podrán iniciarse una vez que el estudiantado haya cubierto aproximadamente el 70% de los créditos del plan de estudios, en concordancia con la normatividad institucional vigente.

Para el caso específico de la Licenciatura en Ciencias Computacionales, las prácticas profesionales contemplan un total de 320 horas, cuya acreditación otorga 10 créditos de carácter obligatorio dentro del plan de estudios. Estas se desarrollan bajo esquemas de supervisión y evaluación conjunta entre la unidad académica y la unidad receptora, mediante el cumplimiento de actividades previamente definidas, así como la entrega de informes y evidencias de desempeño debidamente validadas.

Desde una perspectiva formativa y de vinculación, las prácticas profesionales representan una oportunidad estratégica para la inserción temprana del estudiantado en el ámbito laboral, al permitirle integrarse a equipos de trabajo, conocer dinámicas

institucionales, aplicar conocimientos en situaciones reales y establecer redes de contacto profesional. En este sentido, el programa educativo promoverá su realización en sectores estratégicos del ámbito ambiental, favoreciendo el desarrollo de experiencia práctica, el fortalecimiento del perfil profesional y una transición progresiva hacia el ejercicio laboral.

La planeación, asignación, seguimiento, evaluación y acreditación de las prácticas profesionales será responsabilidad de la unidad académica, en coordinación con las unidades receptoras, asegurando que las actividades realizadas contribuyan al desarrollo de competencias profesionales y respondan a las necesidades del entorno. Las prácticas profesionales consolidan un componente formativo clave del plan de estudios, al articular la formación académica con la experiencia laboral, fortalecer la empleabilidad del estudiantado y contribuir a su desempeño eficaz en contextos socioambientales complejos.

6.3.3. Lengua extranjera

Los criterios establecidos para acreditar el dominio de una segunda lengua se fundamentan en la visión de la UABC, que reconoce la necesidad del bilingüismo para competir en un contexto global. Por ello, y en consonancia con lo estipulado en los planes de estudio de la Facultad de Ciencias, se ha definido un nivel de competencia específico que garantice que los futuros profesionistas posean las herramientas lingüísticas necesarias. Este estándar no solo busca cumplir con un requisito curricular, sino que es un elemento clave para robustecer su formación, asegurando una movilidad académica y profesional efectiva, y favoreciendo su incorporación al mercado laboral con una visión amplia y competitiva, tal como lo establece el modelo educativo institucional. Las opciones definidas son las siguientes:

- a) Obtener al menos 65 puntos (B1+) del examen de diagnóstico de la CGSECE de la UABC.
- b) Constancia de haber obtenido por lo menos 56 puntos en el examen TOEFL-iBT; o por lo menos 500 puntos en el examen TOEFL-iTP, o el examen TOEIC; o haber

obtenido al menos el nivel B1 del CENNI (nivel 10), o de la Certificación Excellence SDSU, o del Curso LEL (Lengua Extranjera en Línea DEXWAY), o del Marco Común Europeo de referencia (MCER) (B1); o al menos el nivel 4.6 de IELTS, o su equivalente. En todos los casos, con una vigencia no mayor a 2 años.

- c) La acreditación del 5to nivel de idioma de los cursos del CEC (antes UNISER).
- d) Quedar asignado al menos en el sexto nivel del examen diagnóstico de lengua extranjera aplicado por la Facultad de Idiomas de la UABC.
- e) La acreditación del examen de egreso de la lengua extranjera, que se aplica en la Facultad de Idiomas de la UABC.
- f) La acreditación de al menos dos unidades de aprendizaje ofertadas en la Facultad de Ciencias, tales como: English Level 1, English Level 2, o unidades de aprendizaje disciplinarias del plan de estudios impartidas en inglés.
- g) Estancias internacionales autorizadas por la Facultad de Ciencias, con duración mínima de tres meses en un país con lengua oficial distinta al español.
- h) Haber acreditado estudios formales en lengua extranjera en instituciones educativas en México o en el extranjero, donde presente certificados de diplomados o estudios de media superior o superior.

El cumplimiento por parte del estudiante en alguna de las opciones señaladas anteriormente dará lugar a la expedición de una constancia de acreditación de lengua extranjera emitida por la Facultad de Ciencias o la Facultad de Idiomas de la UABC.

6.3.4. Modalidades de titulación

La obtención del título profesional en la Universidad Autónoma de Baja California se rige por lo establecido en el Estatuto Escolar vigente, el cual regula el proceso de otorgamiento de títulos a quienes han concluido la totalidad de los créditos del plan de estudios y cumplido con los demás requisitos académicos e institucionales aplicables. Asimismo, el proceso de titulación se sujeta a lo dispuesto en el Reglamento General de Exámenes Profesionales y Estudios de Posgrado, que establece las condiciones, procedimientos y modalidades mediante las cuales el estudiantado podrá demostrar el logro de las competencias adquiridas durante su formación profesional.

En este marco normativo, las personas egresadas de la Licenciatura en Ciencias Computacionales deberán cumplir con los requisitos establecidos en la normatividad universitaria vigente, así como con los lineamientos específicos definidos por la unidad académica correspondiente, a fin de obtener el título profesional.

Con el propósito de favorecer la eficiencia terminal y ofrecer alternativas pertinentes para la culminación del proceso formativo, la Universidad contempla diversas modalidades de titulación, las cuales permiten al estudiantado acreditar, desde distintos enfoques, el dominio de conocimientos, habilidades y competencias profesionales desarrolladas a lo largo de su trayectoria académica. Entre dichas modalidades se encuentran:

- Obtener la constancia de presentación y acreditación del Examen General de Egreso de Licenciatura (EGEL) aplicado por el Centro Nacional de Evaluación para la Educación Superior (CENEVAL), o su equivalente autorizado por el H. Consejo Universitario.
- Haber alcanzado un promedio general mínimo de 90 al concluir los estudios profesionales.
- Haber cubierto el total de los créditos de una especialidad o al menos el 50% de los créditos de una maestría en un área del conocimiento afín.
- Comprobar el desempeño del ejercicio o práctica profesional por un periodo mínimo acumulado de dos años, conforme a los criterios establecidos por la unidad académica.
- Aprobar el informe o memoria de la prestación del servicio social profesional.
- Presentar y aprobar una tesis profesional, desarrollada con base en el método científico y conforme a los lineamientos metodológicos establecidos.
- Titulación por proyecto, mediante la presentación de un informe derivado de actividades de vinculación con la sociedad, particularmente aquellas desarrolladas en el marco de Proyectos de Vinculación con Valor en Créditos (PVVC).
- Titulación automática para egresados y egresadas de programas educativos reconocidos como de calidad por organismos acreditadores o evaluadores externos, como COPAES o CIEES, conforme a la normatividad vigente.

Estas modalidades de titulación aseguran que las personas egresadas demuestren de manera formal el logro del perfil de egreso, al tiempo que ofrecen flexibilidad para que el proceso de titulación responda a sus trayectorias académicas y experiencias profesionales, fortaleciendo su inserción en el ámbito laboral y su contribución al desarrollo sustentable. De acuerdo con el Reglamento General de Exámenes Profesionales, el proceso de titulación podrá incluir, según la modalidad elegida, la presentación de un examen profesional, el cual tiene como finalidad valorar de manera integral los conocimientos generales del sustentante y su capacidad para aplicarlos en la resolución de problemáticas propias de su campo disciplinar.

7. Condiciones institucionales para la operación del plan de estudios

La viabilidad y sostenibilidad de un plan de estudios no dependen exclusivamente de su solidez académica o de la pertinencia de su diseño curricular, sino de las condiciones institucionales que garantizan su adecuada implementación, seguimiento y mejora continua. En este sentido, el presente apartado establece el conjunto de capacidades organizacionales, recursos y dispositivos de gestión académica con los que cuenta la institución para asegurar la operación efectiva del programa educativo, en congruencia con los principios de calidad, pertinencia, equidad e innovación educativa que orientan a la universidad.

Estas condiciones institucionales se estructuran de manera sistémica, considerando tanto los insumos estratégicos como los procesos de apoyo que inciden directamente en la formación integral del estudiantado. Se abordan, en primer término, los mecanismos de difusión del programa educativo, los cuales permiten su posicionamiento y adecuada comunicación hacia los distintos públicos de interés. Posteriormente, se describen las características de la planta académica y la organización de los cuerpos académicos, elementos clave para el desarrollo de funciones sustantivas como la docencia, la investigación y la vinculación.

Asimismo, se detallan las condiciones de infraestructura física y tecnológica, así como la disponibilidad de materiales y equipamiento especializado, indispensables para el desarrollo de experiencias de aprendizaje pertinentes y contextualizadas. De igual forma, se consideran los recursos bibliográficos y de información que respaldan el acceso al conocimiento actualizado y a fuentes especializadas, en consonancia con las exigencias del campo disciplinar.

Finalmente, se incorporan los dispositivos institucionales orientados al acompañamiento y bienestar del estudiantado, tales como el Programa de Tutoría Académica y los servicios de apoyo proporcionados desde la estructura organizacional, los cuales contribuyen a fortalecer la trayectoria escolar, la permanencia, el egreso oportuno y la formación integral.

7.1. Difusión del programa educativo

Mecanismo de difusión:

La Universidad Autónoma de Baja California cuenta con mecanismos robustos de promoción externa diseñados para orientar a los aspirantes y difundir el quehacer académico. Estas acciones se fundamentan en la misión institucional de promover y difundir la cultura, así como en la divulgación del conocimiento para enriquecer a la sociedad. Asimismo, el modelo educativo establece que la información y orientación proporcionada a las y los estudiantes antes del ingreso es un componente esencial para facilitar su incorporación a la vida universitaria.

A continuación, se describen las actividades de promoción en las que participamos las y los profesores del programa académico, las cuales sirven como plataforma para la captación de talento y el posicionamiento social de la carrera:

1. Mecanismos de Orientación Vocacional y Captación Directa

Estas actividades buscan un contacto cercano con las y los estudiantes de nivel medio superior para resolver dudas específicas sobre el plan de estudios y el perfil profesional:

- **Expoprofesiones UABC:** Evento institucional masivo donde el programa académico instala módulos informativos. Es el principal escaparate para mostrar la oferta educativa a miles de bachilleres de la región.
- **Ferias Vocacionales en Preparatorias:** Participación activa en eventos organizados por instituciones externas (públicas y privadas), donde personal docente y estudiantes del programa presentan las ventajas competitivas de la carrera.
- **Pláticas por invitación:** Charlas técnicas y motivacionales impartidas directamente en los planteles de nivel medio superior, enfocadas en las áreas de aplicación profesional del programa.
- **Seminarios para Preparatorias en la Facultad:** Actividades de "puertas abiertas" donde grupos de bachilleres visitan nuestras instalaciones para recibir talleres o seminarios académicos, permitiéndoles experimentar el entorno universitario de primera mano.

2. Mecanismos de Divulgación Científica y Extensión

Estas acciones posicionan el programa a través de la comunicación masiva y eventos de impacto comunitario:

- Programa Radiofónico "Hablemos de Ciencias": Espacio de difusión en medios universitarios donde se abordan temas de actualidad vinculados al programa, permitiendo que la sociedad conozca la relevancia científica y social de nuestra disciplina.
- Eventos de Divulgación Semana de Ciencias y Expo Ciencia y Tecnología: Jornadas de puertas abiertas dirigidas al público general y estudiantes, donde se realizan experimentos, exhibiciones y demostraciones de los proyectos desarrollados en el programa.
- Noche de las Estrellas y Noche de las Ciencias: Eventos de divulgación científica de gran alcance que permite la vinculación del programa con la comunidad astronómica y científica y el público interesado en la ciencia, fomentando vocaciones científicas tempranas.

3. Fomento al Talento y Pensamiento Computacional (Vinculación Temprana)

El programa lidera iniciativas estratégicas que permiten un acercamiento progresivo a la disciplina desde las etapas iniciales de formación:

- Organización de la Olimpiada Mexicana de Informática en Baja California (OMIBC): El programa educativo es el responsable estatal de coordinar este certamen, el cual está dirigido a estudiantes de primaria, secundaria y preparatoria. Esta competencia permite identificar y potenciar el talento en resolución de problemas y algoritmos, vinculando a la Facultad con el sistema educativo básico y media superior.
- Reto Bebras para Nivel Preescolar: Como parte de la estrategia de inclusión y fomento al pensamiento lógico desde la infancia temprana, se coordina la participación de niños y niñas de preescolar en el Reto Bebras. Esta iniciativa

introduce conceptos fundamentales de informática de manera lúdica, sembrando la semilla de la curiosidad científica desde los primeros años.

- Medios digitales para la divulgación y comunicación con las y los estudiantes y la comunidad. La información del programa se encuentra alojada en diversos sitios web dentro de la Universidad.
- Páginas Web:
 - Página del programa: <https://ciencias.ens.uabc.mx/computacion/>
 - Página de la Facultad de Ciencias: <https://ciencias.ens.uabc.mx/>
 - Página de la Coordinación de Formación Profesional:
<https://cgfp.uabc.mx/licenciatura-en-ciencias-computacionales/>
- Redes sociales de la Facultad de Ciencias y del Programa LCC:
 - Facebook: <https://www.facebook.com/FC.UABC>
 - Facebook: <https://www.facebook.com/LCC.FC.UABC/>
 - Instagram: <https://www.instagram.com/lcc.fc.uabc>
 - X: https://x.com/LCC_FCC_UABC
 - Threads: <https://www.threads.com/@lcc.fc.uabc>
- Google Classroom

Espacio para la comunicación más oficial entre las y los estudiantes del programa y sus profesores.

- <https://classroom.google.com/c/NjMxNjY2NjczOTkz>

7.2. Planta académica

La planta académica que atiende el programa educativo está conformada por 32 profesores, de los cuales 7 son Profesores de Tiempo Completo (PTC) adscritos al programa, 2 Técnicos Académicos, 9 son PTC de otros programas educativos y 14 son Profesores de Asignatura. De los PTC adscritos al programa, el 71 % (5) cuenta con reconocimiento en el Sistema Nacional de Investigadoras e Investigadores (SNII) y el 85% (6) cuentan con el Reconocimiento a Profesores de Tiempo Completo con Perfil Deseable (PRODEP). El número y grado académico de las y los profesores, se muestra en las Tablas 8 y 9.

Tabla 9.*Grado de los PTC adscritos al programa educativo.*

Grado	Cantidad
Doctorado	6
Maestría	1
Licenciatura	0
Total	7

Fuente: Elaboración propia.

Tabla 10.*Perfil de la planta docente de tiempo completo.*

Empleado	Nombre	Grados académicos	Institución de egreso del último grado
24473	Eloísa del Carmen García Canseco	Ingeniería Electrónica, Maestría en Ciencias en Electrónica y Telecomunicaciones, Doctorado en Física	Université Paris XI, France
22578	Everardo Gutiérrez López	Licenciado en Informática, M. en Ciencias en Ciencias de la Computación, Doctorado en Ciencias en Ciencias de la Computación	CICESE
20699	José Angel González Fraga	Ingeniería en Electrónica, M. en Ciencias en Ciencias de la Computación, Doctorado en Ciencias en Ciencias de la Computación	CICESE
16107	Alma Rocío Cabazos Marín	Licenciatura en Ciencias Computacionales, M. en Ingeniería Óptica de Software, Doctorado en Ciencias	Universidad de Sonora
15439	María Victoria Meza Kubo	Licenciatura en Ciencias Computacionales, M. en Ciencias en Computación Doctorado en Ciencias en Computación	UABC
13653	Evello Martínez Martínez	Licenciatura en Ciencias Computacionales,	Fundación TELEDDES, A.C.

Empleado	Nombre	Grados académicos	Institución de egreso del último grado
		Maestría Corporativa en telecomunicaciones y redes de Información	
13399	Alberto L. Morán y Solares	Licenciatura en Ciencias Computacionales, M. en Ciencias en Computación Doctorado en Ciencias en Computación	INPG, Francia

Fuente: Elaboración propia.

7.3. Cuerpos académicos

En la Facultad de Ciencias, las y los profesores de tiempo completo que conforman la academia de la Licenciatura en Ciencias Computacionales se encuentran organizados en tres Cuerpos Académicos (CA), los cuales aportan de manera sustantiva al fortalecimiento del programa educativo. Esta estructura permite articular las funciones de docencia, investigación y vinculación, favoreciendo una formación académica sólida y pertinente.

De las y los docentes de tiempo completo adscritos al programa, un grupo significativo forma parte de dos cuerpos académicos consolidados: Tecnologías de Información y Visualización (UABC-CA-171) y Tecnologías para Ambientes Inteligentes (UABC-CA-113), mientras que otro docente participa en el Cuerpo Académico Enseñanza y Divulgación de las Ciencias (UABC-CA-205), actualmente en proceso de consolidación. Esta integración refleja un equilibrio entre capacidades consolidadas de investigación y áreas en desarrollo orientadas al fortalecimiento de la innovación educativa.

El Cuerpo Académico Tecnologías de Información y Visualización, con grado de consolidado, cuenta con tres líneas de generación y aplicación del conocimiento estrechamente vinculadas con el campo disciplinar del programa. La LGAC de Aplicaciones de la Inteligencia Artificial, genera aplicaciones en los diferentes campos de estudio de la inteligencia artificial: visión por computadora, aprendizaje automático, procesamiento del lenguaje natural, robótica, entre otros. Promoviendo la investigación en IA para su aplicación en resolución de tareas y problemáticas de las ciencias. La

LGAC Interdisciplina en la Ciencia, se enfoca en la solución de problemas complejos desde un enfoque holístico, bajo la perspectiva de diferentes áreas del conocimiento, incluyendo aspectos biológicos, ecológicos, tecnológicos, de visualización y de manejo de recursos. Con lo anterior se promueve la flexibilidad cognitiva y enfoques innovadores. Finalmente, la LGAC de TICs, educación y divulgación de la ciencia, promueve el acceso universal al conocimiento, así como al pensamiento crítico y la comprensión de la información novedosa generada a través de las investigaciones de frontera, a través de la docencia, talleres, publicaciones, conferencias, simposios, redes sociales, etc. Estas líneas se encuentran alineadas con las áreas de conocimiento y asignaturas del programa educativo, contribuyendo al desarrollo de competencias técnicas y analíticas en las y los estudiantes.

Por otro lado, el Cuerpo Académico Tecnologías para Ambientes Inteligentes, también con grado de consolidado, desarrolla una línea de generación y aplicación del conocimiento enfocada en el análisis, diseño y desarrollo de infraestructuras tecnológicas para ambientes inteligentes, capaces de ofrecer servicios oportunos a los usuarios. Esta línea responde a tendencias actuales en el ámbito de las ciencias computacionales, particularmente en áreas como sistemas inteligentes, automatización y desarrollo de soluciones centradas en el usuario, fortaleciendo la pertinencia del programa frente a los retos de la transformación digital.

Asimismo, el Cuerpo Académico Enseñanza y Divulgación de las Ciencias, en proceso de consolidación, aporta desde una perspectiva pedagógica mediante el diseño y aplicación de materiales didácticos orientados a mejorar los procesos de enseñanza, aprendizaje y divulgación científica. Su participación representa una oportunidad estratégica para fortalecer la innovación educativa y el desarrollo de prácticas docentes más efectivas dentro del programa.

Las y los profesores adscritos a estos cuerpos académicos cuentan con experiencia en el desarrollo de proyectos de investigación, así como en la aplicación de tecnologías en distintos contextos, lo que permite integrar al proceso formativo experiencias actualizadas y vinculadas con las necesidades del entorno. Esta actividad académica y científica favorece la actualización constante del plan de estudios, la incorporación de avances tecnológicos y la generación de conocimiento pertinente.

La pertenencia de las y los docentes del programa educativo a estos cuerpos académicos implica que la Licenciatura en Ciencias Computacionales se encuentra respaldada por una planta académica con actividad científica estructurada y colaborativa, lo que contribuye a elevar la calidad de la enseñanza, fortalecer la vinculación con otros sectores, mejorar la proyección académica del programa y ofrecer mayores oportunidades de formación para el estudiantado.

En este sentido, la integración de los cuerpos académicos dentro de la Facultad de Ciencias representa un beneficio estratégico para la calidad educativa, al asegurar que el programa se sustente en líneas de investigación vigentes, alineadas con las tendencias del campo disciplinar, y orientadas a la solución de problemáticas reales. Esta sinergia institucional contribuye a la consolidación y crecimiento del programa educativo en el mediano y largo plazo.

Tabla 11. Integrantes del cuerpo académico que incide en el programa educativo.

Líneas de Generación y Aplicación del Conocimiento	Integrantes
Aplicaciones de la Inteligencia Artificial, Interdisciplina en la Ciencia, TICs, educación y divulgación de la ciencia.	González Fraga Jose Angel Martínez Martínez Evelio Gutiérrez López Everardo
Tecnologías para Ambientes Inteligentes	Morán Y Solares Alberto Leopoldo García Canseco Eloisa del Carmen Meza Kubo María Victoria
Enseñanza y Divulgación de las Ciencias	Alma Rocío Cabazos Marín

Fuente. Elaboración propia.

7.4. Infraestructura, materiales y equipamiento

Facultad de Ciencias

La infraestructura de la Facultad de Ciencias está diseñada para apoyar el desarrollo académico, la investigación y la vinculación con la comunidad, mediante la disponibilidad de espacios y recursos que garantizan la calidad en los procesos de enseñanza-aprendizaje y el fortalecimiento del quehacer científico. Esta infraestructura es de uso compartido entre los programas educativos de licenciatura que se imparten en la unidad académica, entre los que se encuentran Física, Biología, Ciencias Computacionales,

Matemáticas Aplicadas y Ciencia de Datos, lo que favorece la interacción interdisciplinaria y el aprovechamiento eficiente de los recursos institucionales.

La Facultad de Ciencias cuenta con una infraestructura conformada por 10 edificios académicos, en los cuales se distribuyen aulas, laboratorios, cubículos para el profesorado, áreas administrativas y espacios de apoyo académico. Esta organización permite atender de manera adecuada las actividades docentes, de investigación y de vinculación, proporcionando condiciones favorables para el desarrollo integral de los programas educativos.

Asimismo, la distribución de los espacios facilita la implementación de actividades prácticas, el trabajo colaborativo y el desarrollo de proyectos académicos, lo que contribuye a fortalecer la formación del estudiantado y a mantener la pertinencia del programa educativo frente a las demandas actuales del entorno científico y tecnológico.

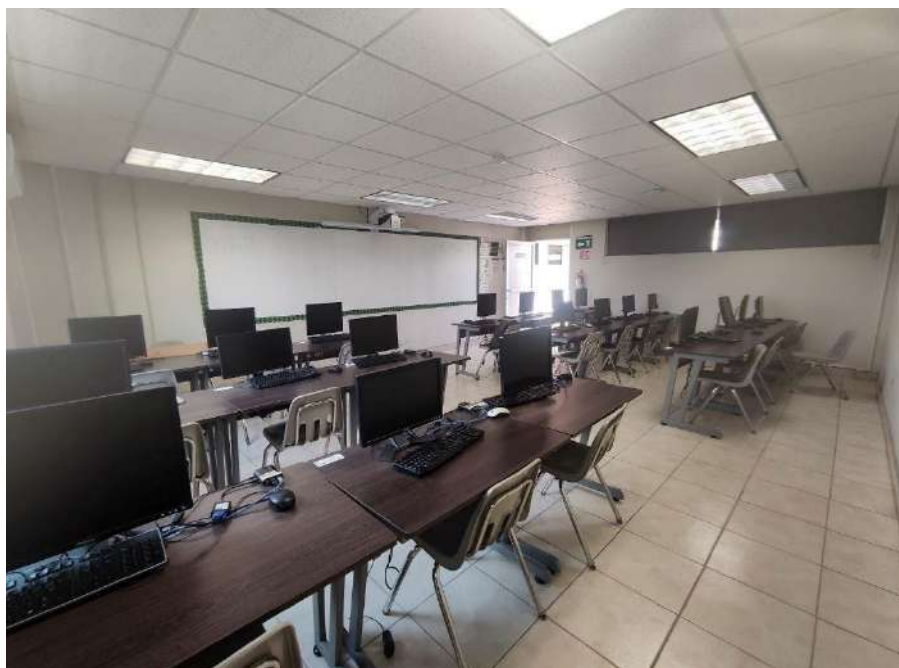
Aulas

Se cuenta con un total de 18 aulas destinadas a la docencia, las cuales están distribuidas en distintos edificios. En el edificio E-2 se dispone de 2 aulas; en el edificio E-3, un aula completamente equipada; en el edificio E-4 se cuentan con 6 aulas en planta baja (D1 a D4 y D6 a D7) y 2 salas de posgrado en planta alta; y en el edificio E-9 se localizan 7 aulas (A1 a A7).

Destaca que los salones D1 a D4 cuentan con proyector, lo cual fortalece el uso de recursos digitales en el proceso de enseñanza-aprendizaje. En conjunto, estos espacios permiten atender adecuadamente las actividades académicas del programa.

Laboratorios

Existen cinco laboratorios de cómputo ubicados en el edificio E-57, cuatro de ellos con capacidades de 21 estudiantes y uno con capacidad para 42 estudiantes. Estos espacios están equipados con mobiliario audiovisual y están destinados para el desarrollo de actividades prácticas, cuentan con estaciones de trabajo y condiciones adecuadas de seguridad y operación, lo que garantiza el soporte tecnológico necesario para el programa educativo.



Fuente. Elaboración propia

Figura 5

Laboratorio de Cómputo LD1. (Raspberry Pi 4 - GNU/Linux)



Figura 6

Laboratorio de Cómputo LD3 (PC - Windows)



Fuente. Elaboración propia

Figura 7

Laboratorio de Cómputo LD4 (PC - Windows)



Fuente. Elaboración propia

Figura 8

Laboratorio de Cómputo LD5 (PC – Windows



Fuente. Elaboración propia

Figura 9

Laboratorio de Inteligencia Artificial



Fuente. Elaboración propia

Cubículos para profesores de carrera y su equipamiento

La facultad dispone de aproximadamente 39 cubículos distribuidos en diversos edificios (E-4, E-7, E-8 y E-6), destinados al trabajo académico de las y los profesores de tiempo completo. Estos espacios están estratégicamente ubicados en cercanía con aulas y laboratorios, lo que facilita la atención a estudiantes, el desarrollo de tutorías y actividades de investigación. Asimismo, los cubículos cuentan con el equipamiento necesario para garantizar el desempeño adecuado de las funciones docentes.

Los dos cubículos destinados a profesores de asignatura cuentan con el equipamiento básico necesario para el desarrollo de sus actividades, tales como mobiliario adecuado (escritorio y sillas), acceso a energía eléctrica y conectividad a la red institucional, lo que permite la preparación de clases y la atención a estudiantes en condiciones apropiadas.

Auditorios, salas audiovisuales y de teleconferencias

Se tienen a disposición dos salas audiovisuales (Audiovisual A y Audiovisual B) y una sala de usos múltiples (sala de juntas), las cuales están equipadas con televisiones y/o proyectores fijos. Estos espacios permiten la realización de clases, presentaciones, reuniones académicas y actividades de videoconferencia, contribuyendo al desarrollo de actividades docentes y de vinculación.

7.5. Recursos bibliográficos y de información

Biblioteca

La Facultad de Ciencias se apoya en los servicios de la Biblioteca Central del campus Ensenada, ubicada a menos de 100 metros de la unidad académica, lo que garantiza un acceso cercano y oportuno para estudiantes y profesores. La biblioteca cuenta con un acervo superior a 70,000 volúmenes, así como acceso a recursos digitales, bases de datos especializadas y libros electrónicos.

Entre los servicios disponibles se incluyen préstamo interno y externo, préstamo interbibliotecario, consulta en línea, formación de usuarios, acceso a bases de datos, biblioteca digital, hemeroteca, materiales audiovisuales, cubículos y espacios de estudio, así como red inalámbrica y herramientas de apoyo académico como metabuscadores y

detección de plagio. Estos servicios garantizan el acceso a información actualizada y pertinente para el desarrollo de las actividades académicas del programa.

Equipo de cómputo para uso de estudiantes

Además de los laboratorios de cómputo, las y los estudiantes tienen acceso a servicios institucionales como red inalámbrica en el campus, red alámbrica de alta velocidad en laboratorios, cuentas de usuario para uso de equipos y acceso a software y recursos tecnológicos necesarios para su formación.

Equipo de cómputo para uso de las y los docentes

Todas y todos los profesores de tiempo completo cuentan con al menos un equipo de cómputo con conexión a la red universitaria. Asimismo, disponen de acceso a un área de impresión equipada con computadora e impresora/multifuncional, así como a la infraestructura tecnológica institucional necesaria para el desarrollo de sus actividades académicas.

Equipo de apoyo para estudiantes y docentes

Se cuenta con equipo de apoyo que incluye servicio de impresión láser y de gran formato (plotter), así como un sistema de préstamo de laptops y proyectores para su uso en actividades académicas. Asimismo, se dispone de salas audiovisuales y espacios con proyectores y televisiones para el desarrollo de clases, presentaciones y eventos académicos.

Figura 10

Laboratorio de Investigación I



Fuente. Elaboración propia

Figura 11

Laboratorio de Investigación II



Fuente. Elaboración propia

Figura 12

Laboratorio de Prácticas Avanzadas 2



Fuente. Elaboración propia

7.6. Programa de Tutoría Académica

El Programa de Tutoría Académica constituye un dispositivo institucional estratégico para el acompañamiento integral del estudiantado a lo largo de su trayectoria escolar, orientado a favorecer la permanencia, el desempeño académico, el egreso oportuno y la consolidación del proyecto profesional. Su enfoque se sustenta en una lógica formativa centrada en el estudiante, en la cual la tutoría trasciende la asesoría administrativa para convertirse en un proceso sistemático de orientación académica, personal y vocacional.

El propósito central del programa es potencializar las capacidades, habilidades y toma de decisiones del estudiantado, promoviendo una participación activa, reflexiva y responsable en su proceso de formación. En este sentido, la tutoría se configura como un espacio de mediación pedagógica que permite identificar oportunamente riesgos académicos, necesidades de apoyo y áreas de oportunidad, así como canalizar al

estudiantado hacia los servicios institucionales pertinentes. Su impacto se expresa en múltiples dimensiones del desarrollo estudiantil:

- Académica: mejora en la planeación de la carga académica, el rendimiento escolar y la continuidad en los estudios.
- Personal: fortalecimiento de la autonomía, la autorregulación y el sentido de pertenencia institucional.
- Profesional: orientación para la construcción del perfil de egreso, elección de trayectorias formativas y toma de decisiones vinculadas al campo laboral.

El tutor o tutora desempeña un papel clave como agente de acompañamiento y orientación, cuya función principal es guiar al estudiantado en la comprensión y gestión de su trayectoria académica. Esta función implica:

- Brindar asesoría académica personalizada sobre la selección de unidades de aprendizaje, avance curricular y cumplimiento de requisitos.
- Detectar de manera oportuna situaciones de riesgo académico o personal, proponiendo estrategias de atención o canalización.
- Promover el desarrollo de habilidades de aprendizaje autónomo y toma de decisiones informadas.
- Dar seguimiento continuo al progreso del estudiante mediante registros sistemáticos y evidencia de intervención.

La función tutorial se reconoce como una actividad académica formal, respaldada por procesos de capacitación y actualización docente, lo que garantiza la calidad y homogeneidad en su implementación.

El programa opera de manera estructurada en la unidad académica desde la Coordinación de Formación Profesional, en articulación con la subdirección, instancia responsable de la asignación de tutores, el seguimiento operativo y la evaluación del programa. A cada estudiante se le asigna un tutor o tutora desde su ingreso hasta el egreso, manteniendo un acompañamiento continuo, con posibilidad de ajuste en casos justificados.

La atención tutorial se organiza mediante:

- Sesiones grupales e individuales obligatorias, con un mínimo de cuatro por ciclo escolar (inicio, seguimiento intermedio, cierre y reinscripción).

- Planes de trabajo tutorial definidos institucionalmente, que orientan las actividades a desarrollar en cada sesión.
- Capacitación permanente a tutores y tutorados para asegurar el adecuado funcionamiento del programa.

El Sistema Institucional de Tutorías (SIT) es la plataforma tecnológica que sustenta la operación, seguimiento y evaluación del programa. Su función es sistematizar la información derivada del proceso tutorial, permitiendo:

- Registrar las sesiones de tutoría y las actividades realizadas.
- Dar seguimiento individualizado a la trayectoria del estudiante.
- Generar reportes académicos y evidencias de intervención tutorial.
- Facilitar la evaluación del programa mediante instrumentos aplicados a tutores y tutorados.

El programa incorpora mecanismos formales de evaluación y retroalimentación continua. Al cierre de cada periodo escolar:

- El tutor elabora un reporte de resultados, documentando avances, problemáticas detectadas y acciones implementadas.
- El estudiantado participa en la evaluación del desempeño tutorial, aportando información clave sobre la calidad del acompañamiento.
- La coordinación del programa integra un informe institucional, que permite identificar áreas de mejora y orientar decisiones académicas y administrativas.

La tutoría académica es un eje articulador del trayecto formativo, ya que acompaña al estudiante en momentos críticos como el ingreso, la adaptación al entorno universitario, la definición de trayectorias, la resolución de dificultades académicas y la transición al egreso. El Programa de Tutoría Académica no solo opera como un mecanismo de apoyo, sino como una estrategia institucional integral que fortalece la formación del estudiantado, contribuyendo a la calidad y pertinencia del programa educativos; su adecuada implementación incide directamente en indicadores institucionales clave como la retención, eficiencia terminal y satisfacción estudiantil.

7.7. Servicios de apoyo al estudiantado

Los servicios de apoyo al estudiantado del programa educativo se articulan a partir de la estructura organizacional (ver figura 15) de la Facultad de Ciencias, lo que permite garantizar una atención integral, sistemática y oportuna a lo largo de la trayectoria académica. Esta organización institucional favorece la coordinación entre áreas académicas, administrativas y de vinculación, asegurando que el estudiantado cuente con acompañamiento permanente en los ámbitos académico, formativo, administrativo y de bienestar. En este sentido, la atención al estudiantado no recae en una sola instancia, sino que se distribuye estratégicamente entre diversas figuras y unidades responsables que, de manera coordinada, contribuyen al desarrollo integral del alumnado y al logro del perfil de egreso.

En el ámbito académico, el programa cuenta con la participación del Responsable de Programa Educativo de Licenciatura, quien coordina la operación del plan de estudios, da seguimiento a la trayectoria escolar y atiende necesidades específicas del estudiantado relacionadas con su formación. De manera complementaria, el Responsable de Tutorías y el Responsable de Asesorías brindan acompañamiento académico personalizado, orientado a fortalecer el desempeño escolar, prevenir el rezago y apoyar la permanencia y egreso oportuno.

Asimismo, el Responsable de Trayectoria Estudiantil y el Responsable de Seguimiento de Egresados permiten dar continuidad al proceso formativo más allá del aula, mediante el monitoreo de indicadores de avance académico y la retroalimentación sobre la inserción laboral y el impacto del programa en el campo profesional.

En el ámbito de formación integral, se cuenta con el apoyo de la Coordinación de Formación Profesional, así como del Responsable de Formación y Seguimiento Docente, quienes contribuyen a fortalecer la calidad de la enseñanza y la pertinencia de las experiencias de aprendizaje. De igual forma, el Responsable de Educación Continua y el Responsable de Emprendedores promueven oportunidades para el desarrollo de competencias complementarias, la actualización disciplinar y la vinculación con el entorno productivo y social.

En materia de vinculación y experiencias formativas en contextos reales, el estudiantado es apoyado por el Responsable de Servicio Social, el Responsable de

Prácticas Profesionales y el Responsable de Movilidad e Intercambio, quienes gestionan y supervisan estas modalidades, asegurando su pertinencia, calidad y alineación con el perfil de egreso. Estas instancias facilitan la transición del estudiantado hacia escenarios profesionales, fortaleciendo su inserción laboral.

De manera complementaria, el programa educativo incorpora el acompañamiento del área de Orientación Educativa y Psicopedagógica, la cual brinda atención especializada para favorecer el bienestar integral del estudiantado. Este servicio contribuye a la atención de necesidades relacionadas con la adaptación a la vida universitaria, el desarrollo de habilidades socioemocionales, la gestión del tiempo y del aprendizaje, así como la identificación y atención de factores que puedan incidir en el desempeño académico, tales como situaciones de riesgo psicoeducativo o dificultades de aprendizaje. Asimismo, promueve acciones preventivas, de acompañamiento y canalización a instancias institucionales cuando se requiere atención específica, fortaleciendo la permanencia, el rendimiento académico y el desarrollo personal del estudiantado.

Adicionalmente, el programa cuenta con el respaldo de áreas de apoyo académico y técnico como el Responsable de Laboratorios, el Encargado de Fotocopiado, el Operador de Transporte y los Conserjes, quienes contribuyen a la operación de las actividades formativas en aula, laboratorio y campo, garantizando condiciones adecuadas para el aprendizaje.

En el ámbito administrativo, el Subdirector, la Secretaría, el Administrador y el Responsable de Informática aseguran el funcionamiento eficiente de los procesos escolares, el acceso a recursos tecnológicos y la atención a trámites académicos, contribuyendo a una gestión institucional eficaz.

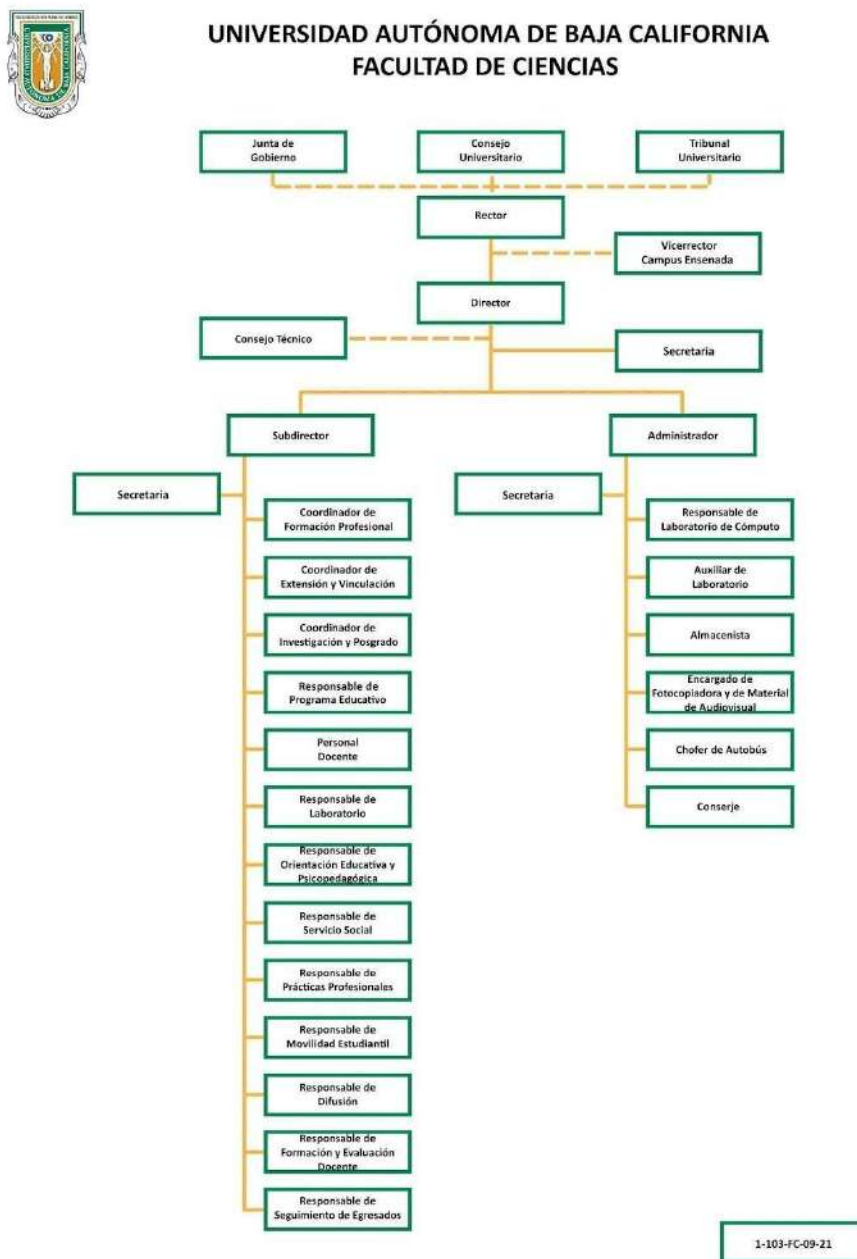
De manera transversal, estas instancias trabajan de forma coordinada mediante mecanismos de comunicación institucional, trabajo colegiado y seguimiento de indicadores académicos, lo que permite identificar oportunamente necesidades del estudiantado, implementar acciones de apoyo y fortalecer la calidad del proceso educativo, por lo tanto, la estructura organizacional de la Facultad de Ciencias constituye un sistema de apoyo integral que favorece el desarrollo académico, profesional y

personal del estudiantado, asegurando trayectorias escolares exitosas, el logro del perfil de egreso y una formación pertinente con las demandas del entorno.

Figura 15

Estructura organizacional de la Facultad de Ciencias

<https://ciencias.ens.uabc.mx/organigrama/>



8. Sistema de evaluación y aseguramiento de la calidad

Para garantizar la calidad, pertinencia y efectividad del plan de estudios, se establece un sistema de evaluación integral, permanente, sistemático y orientado a la mejora continua, que permite generar información relevante para la toma de decisiones académicas, curriculares e institucionales.

La evaluación del programa educativo se concibe como un proceso articulado que trasciende la medición de resultados, al integrar el análisis del diseño curricular, su implementación en contextos reales, el logro de las competencias del perfil de egreso y la pertinencia de la formación frente a las demandas del entorno social, profesional y disciplinar. En este sentido, la evaluación no se limita a la revisión interna de la unidad académica, sino que incorpora su relación con el contexto, la vinculación con sectores externos y el impacto del programa en la formación de profesionistas.

En congruencia con el enfoque por competencias adoptado en el plan de estudios, el sistema de evaluación se orienta a valorar el desempeño del estudiantado mediante evidencias que reflejen la integración de conocimientos, habilidades, actitudes y valores en situaciones auténticas de aprendizaje. Asimismo, se sustenta en metodologías contemporáneas de evaluación, tales como la evaluación formativa, la evaluación auténtica, el uso de rúbricas analíticas, el portafolio de evidencias, la autoevaluación y la coevaluación, las cuales permiten acompañar el proceso formativo y retroalimentar de manera continua el aprendizaje.

El sistema de evaluación del programa educativo se estructura en cuatro dimensiones interrelacionadas: la evaluación del plan de estudios, la evaluación del aprendizaje, la evaluación colegiada del aprendizaje y el uso de resultados para la mejora continua. Estas dimensiones permiten asegurar la coherencia entre el perfil de egreso, las competencias, las experiencias de aprendizaje y los resultados formativos, consolidando una cultura de evaluación basada en evidencia.

De esta manera, la evaluación se constituye como un mecanismo estratégico para el aseguramiento de la calidad educativa, al permitir identificar fortalezas, detectar áreas

de oportunidad y orientar la mejora continua del programa educativo, en correspondencia con los principios del Modelo Educativo.

8.1. Evaluación del plan de estudios

La evaluación del plan de estudios tiene como propósito valorar de manera integral su pertinencia, coherencia interna, vigencia disciplinar y efectividad en la formación del estudiantado, con el fin de asegurar que la propuesta curricular responda de manera adecuada al perfil de egreso y a las demandas del entorno social, profesional y académico.

En el marco de este programa educativo, la evaluación del plan de estudios se concibe como un proceso sistemático y permanente que considera tanto el análisis del diseño curricular como su implementación en la práctica. Esto implica revisar el plan de estudios como documento normativo —en cuanto a sus fundamentos, perfil de egreso, competencias, estructura, organización de créditos y modalidades de aprendizaje—, así como su concreción en los procesos de enseñanza, evaluación, trayectoria escolar, vinculación y operación institucional.

Para tal efecto, la evaluación del plan de estudios se llevará a cabo mediante un esquema temporal que permita valorar tanto su instrumentación inicial como su impacto formativo. En una primera fase, se realizará una evaluación de seguimiento a los dos años de su implementación, con el objetivo de identificar posibles áreas de ajuste relacionadas con la operación del plan, tales como la coherencia entre el diseño y la práctica, el comportamiento de la trayectoria escolar del estudiantado, la pertinencia de las unidades de aprendizaje, el funcionamiento de las etapas de formación y la disponibilidad de recursos académicos e institucionales.

En una segunda fase, se desarrollará una evaluación interna y externa una vez que se hayan cumplido dos años de egreso de las primeras cohortes formadas bajo el nuevo plan de estudios. Esta evaluación permitirá analizar el impacto del programa en términos del logro del perfil de egreso, la inserción laboral de las personas egresadas, la pertinencia de la formación profesional y la congruencia del currículo con las necesidades del campo disciplinar. Los resultados obtenidos constituirán insumos fundamentales para la actualización o modificación del plan de estudios.

Para el desarrollo de este proceso se emplearán diversas fuentes de información, tales como indicadores institucionales de trayectoria escolar, resultados de evaluación del aprendizaje, evaluación docente, estudios de seguimiento de egresados, opinión de empleadores, así como informes generados por academias y cuerpos colegiados, entre otros. El análisis de esta información se realizará de manera colegiada, con el propósito de garantizar una interpretación integral y fundamentada de los resultados reconociendo que la evaluación del plan de estudios se constituye como un mecanismo estratégico que permite asegurar la calidad y pertinencia de la formación, al tiempo que orienta la toma de decisiones para la mejora continua del programa educativo.

8.2. Evaluación del aprendizaje

De acuerdo con el Estatuto Escolar vigente, la evaluación de los procesos de aprendizaje tiene como propósito que las autoridades universitarias, el personal docente y el estudiantado dispongan de información pertinente para valorar los resultados del proceso educativo y propiciar su mejora continua; que el estudiantado conozca el nivel de aprovechamiento académico alcanzado y, en su caso, obtenga la promoción correspondiente; y evidenciar el desarrollo de las competencias adquiridas durante su formación.

En el marco del presente plan de estudios, la evaluación del aprendizaje se concibe como un proceso integral, continuo, sistemático y formativo, orientado a valorar el desarrollo progresivo de las competencias del perfil de egreso. En este sentido, trasciende la medición de conocimientos teóricos aislados, para centrarse en la capacidad del estudiantado de integrar y aplicar conocimientos, habilidades, actitudes y valores en contextos académicos y profesionales propios de su disciplina.

Bajo el enfoque por competencias, la evaluación se orienta al desempeño del estudiantado en situaciones de aprendizaje significativas, por lo que se sustenta en la generación de evidencias que permitan valorar la aplicación del conocimiento en la resolución de problemas, el análisis de situaciones complejas y la construcción de propuestas pertinentes. En congruencia con ello, cada unidad de aprendizaje establecerá de manera explícita las competencias a desarrollar, los resultados de aprendizaje esperados y los criterios de evaluación, los cuales orientarán tanto las actividades

formativas como las evidencias que se considerarán para valorar el logro de los aprendizajes.

La evaluación del aprendizaje se desarrollará a lo largo del periodo escolar en distintos momentos que permiten acompañar el proceso formativo del estudiantado. En un primer momento, se realizarán evaluaciones diagnósticas que permitan identificar los conocimientos previos y las condiciones de entrada del alumnado; posteriormente, se llevarán a cabo evaluaciones formativas que faciliten la retroalimentación continua, la identificación de áreas de mejora y el fortalecimiento del aprendizaje; y finalmente, se implementarán evaluaciones sumativas que permitan valorar el nivel de logro de las competencias al cierre de las unidades de aprendizaje o de experiencias formativas específicas.

En correspondencia con las metodologías de enseñanza adoptadas en el programa educativo, la evaluación del aprendizaje incorporará estrategias contemporáneas acordes con el enfoque por competencias, tales como la evaluación basada en el desempeño, la evaluación auténtica en situaciones reales o simuladas, la evaluación por proyectos, el análisis de casos, así como el uso de portafolios de evidencias. De igual forma, se promoverá la participación activa del estudiantado en su propio proceso de evaluación mediante ejercicios de autoevaluación y coevaluación, con el propósito de fortalecer su capacidad de reflexión, autorregulación y mejora continua.

Las evidencias de aprendizaje estarán constituidas por productos académicos y profesionales que reflejen el desempeño del estudiantado, tales como proyectos integradores, reportes técnicos, prácticas de laboratorio y campo, análisis de casos, presentaciones orales, propuestas de intervención, entre otros. La selección de estas evidencias será congruente con las competencias a desarrollar y con el nivel de complejidad de cada etapa de formación.

Para asegurar la objetividad, transparencia y consistencia en la evaluación, se emplearán instrumentos como rúbricas analíticas, listas de cotejo y guías de evaluación por competencias, los cuales permitirán establecer criterios claros y compartidos entre el personal docente, así como facilitar la retroalimentación al estudiantado.

Asimismo, la evaluación docente institucional se integra como un mecanismo complementario que permite retroalimentar la práctica educativa, identificar áreas de

mejora en el proceso de enseñanza y promover la actualización pedagógica del profesorado. En caso de detectarse áreas de rezago en el estudiantado, se podrán implementar estrategias de apoyo académico, tales como cursos de nivelación, asesorías especializadas o acompañamiento académico, con el propósito de favorecer su permanencia y éxito escolar.

8.3. Evaluación colegiada del aprendizaje

La evaluación colegiada del aprendizaje constituye un mecanismo institucional orientado a garantizar la objetividad, consistencia y calidad en la valoración del logro de las competencias del estudiantado, mediante la participación organizada del profesorado en espacios académicos colegiados.

En el contexto del programa educativo, esta evaluación se concibe como un proceso complementario a la evaluación desarrollada en cada unidad de aprendizaje, cuyo propósito es generar información integral sobre el nivel de avance del estudiantado en distintos momentos de su trayectoria formativa, así como identificar fortalezas y áreas de oportunidad en la implementación del plan de estudios.

La evaluación colegiada se implementará a través de diversos instrumentos y mecanismos institucionales, tales como exámenes departamentales, exámenes de trayecto y exámenes de egreso, así como otras estrategias que las academias y cuerpos académicos determinen pertinentes. Estos instrumentos serán diseñados de manera colegiada, con base en criterios de desempeño alineados con las competencias del perfil de egreso y con los contenidos fundamentales de las unidades de aprendizaje.

Los exámenes departamentales permitirán verificar la homogeneidad de los aprendizajes en unidades de aprendizaje específicas, asegurando que los contenidos y competencias se estén abordando de manera consistente entre distintos grupos y docentes. Por su parte, los exámenes de trayecto estarán orientados a valorar el logro de competencias al cierre de una etapa de formación, mientras que los exámenes de egreso permitirán identificar el nivel de dominio de conocimientos y habilidades esenciales para el inicio del ejercicio profesional.

El análisis de los resultados de la evaluación colegiada se realizará en espacios académicos formales, como academias y cuerpos colegiados, lo que permitirá identificar

tendencias, detectar áreas críticas del currículo y generar acuerdos orientados a la mejora de los procesos de enseñanza y evaluación.

8.4. Uso de resultados y mejora continua

El sistema de evaluación del programa educativo se articula con un proceso permanente de mejora continua, en el cual los resultados obtenidos en las distintas evaluaciones se utilizan como insumo para la toma de decisiones académicas, curriculares y organizacionales, con el propósito de fortalecer la calidad y pertinencia de la formación.

En este sentido, la evaluación no se limita a la generación de información, sino que implica su análisis, interpretación y uso sistemático en la implementación de acciones de mejora. Para ello, la unidad académica establecerá un proceso estructurado que permita integrar los resultados de la evaluación del plan de estudios, la evaluación del aprendizaje, la evaluación colegiada, la evaluación docente, así como la información derivada del seguimiento de egresados y la opinión de empleadores.

El análisis de esta información se llevará a cabo en espacios colegiados, donde participarán academias, cuerpos académicos, responsables del programa educativo y autoridades académicas, con el fin de identificar tendencias, áreas de oportunidad, problemáticas recurrentes y buenas prácticas. Este análisis permitirá distinguir entre aspectos relacionados con el diseño curricular y aquellos vinculados con la operación del programa, facilitando la definición de acciones pertinentes.

Con base en este proceso, se establecerán acciones de mejora orientadas a fortalecer el plan de estudios, tales como la actualización de contenidos, la revisión de secuencias formativas, el ajuste de estrategias de enseñanza y evaluación, la capacitación docente, el fortalecimiento de los servicios de apoyo al estudiantado y la mejora de recursos e infraestructura.

Estas acciones serán implementadas con base en una planeación que contemple responsables, plazos e indicadores de seguimiento, lo que permitirá dar continuidad a los procesos de mejora y verificar su efectividad. Asimismo, los resultados del seguimiento se incorporarán a nuevos ciclos de evaluación, generando un proceso continuo de retroalimentación.

9. Proceso de validación curricular del plan de estudios

El proceso de validación curricular del presente plan de estudios constituye una etapa fundamental para garantizar su calidad, pertinencia y congruencia con las necesidades del entorno social, profesional y disciplinar, así como con los lineamientos institucionales de la Universidad Autónoma de Baja California. Este proceso permite someter la propuesta curricular a la revisión, análisis y dictamen de distintos órganos colegiados e instancias académicas, con el propósito de asegurar que su diseño, estructura y orientación formativa cumplan con los criterios establecidos para la formación profesional de calidad.

En este sentido, la validación curricular se desarrolla mediante un proceso sistemático y colegiado que integra la participación de actores internos y externos a la universidad, lo que favorece una valoración integral del plan de estudios desde múltiples perspectivas. Este proceso contempla la revisión por pares externos, quienes aportan una visión especializada y actualizada del campo disciplinar; el análisis del Consejo de Vinculación, que permite contrastar la pertinencia del programa con las necesidades del sector social, productivo y profesional; así como la revisión y aprobación por parte del Consejo Técnico de la unidad académica, órgano responsable de garantizar la calidad académica del programa.

De manera complementaria, la propuesta curricular es sometida al aval de instancias institucionales como la Coordinación General de Formación Profesional, la cual verifica su congruencia con el modelo educativo, los lineamientos curriculares y la normatividad universitaria. En el caso de programas en modalidad semipresencial o a distancia, se incorpora además la evaluación del Centro de Investigación para el Aprendizaje Digital, con el fin de asegurar la calidad pedagógica y tecnológica de la propuesta en entornos virtuales.

Se reconoce que este proceso de validación curricular garantiza que el plan de estudios sea resultado de un ejercicio riguroso, colegiado y fundamentado, que integra distintas perspectivas y niveles de análisis, fortaleciendo su pertinencia, viabilidad operativa y alineación con los principios institucionales. Asimismo, permite consolidar

una propuesta formativa sólida, orientada al logro del perfil de egreso y a la formación de profesionistas capaces de responder a los desafíos contemporáneos de su campo disciplinar.

9.1. Evaluación por pares externos



Facultad Interdisciplinaria de Ciencias Exactas y Naturales
Universidad de Sonora

Hermosillo Sonora, a 17 de abril de 2026

Dr. Alberto Leopoldo Morán y Solares

Director de la Facultad de Ciencias
Universidad Autónoma de Baja California
P r e s e n t e

Estimado Dr. Morán:

Reciba un cordial saludo. En atención a la invitación que se me extendió para participar como evaluador externo de la propuesta de modificación del Plan de Estudios del Programa Educativo de Licenciatura en Ciencias Computacionales, adscrito a la Facultad que usted dignamente dirige, me permito emitir la siguiente opinión:

Derivado de una revisión exhaustiva de la propuesta presentada, se considera que las modificaciones planteadas contribuyen de manera significativa al fortalecimiento del perfil académico del programa, al alinearse de forma pertinente con las exigencias actuales del entorno tecnológico y educativo. De igual manera, se identifica una orientación clara hacia la consolidación de una formación integral de los estudiantes, en congruencia con las dinámicas contemporáneas de la sociedad y con los avances científicos y tecnológicos que caracterizan el campo de las Ciencias Computacionales.

La propuesta curricular evidencia un enfoque actualizado y pertinente, particularmente en la incorporación y fortalecimiento de áreas estratégicas como la Inteligencia Artificial, la Ingeniería de Software, la Ciencia de Datos, la Ciberseguridad y las Redes de Datos. Asimismo, se valora positivamente la inclusión de asignaturas orientadas al aprendizaje automático, aprendizaje profundo y modelos generativos, entre otras, las cuales se encuentran alineadas con las tendencias y estándares internacionales del sector.

Se identifica una adecuada correspondencia con las directrices y recomendaciones internacionales de referencia, como la IEEE y la ACM, en un contexto global caracterizado por la transformación digital y la integración de tecnologías emergentes.

En concordancia con lo anterior, la propuesta responde de manera pertinente a la necesidad de fortalecer un enfoque basado en competencias, promoviendo la integración de elementos prácticos, interdisciplinarios y orientados a la solución de problemáticas reales. En este sentido, se favorece la

Edificio 3K-1 planta baja, Blvd. Luis Encinas y Rosales, s/n,
Colonia Centro, C.P. 83000 Hermosillo, Sonora, México
(662) 259 21 53 y 259 21 54, extensión 8153 y 8154
f.cen@unison.mx
<https://i-cen.unison.mx>





Facultad Interdisciplinaria de Ciencias Exactas y Naturales

Universidad de Sonora

formación de egresados con sólidas capacidades técnicas, pensamiento crítico, habilidades de innovación y un alto sentido de responsabilidad ética en el uso de la tecnología.

Cabe destacar que el diseño curricular incorpora estrategias para el desarrollo de competencias transversales, en concordancia con los lineamientos de organismos acreditadores como el CONAIC, fortaleciendo habilidades como la comunicación, el trabajo en equipo, la adaptabilidad y el aprendizaje continuo.

Con base en lo anterior, emito una opinión favorable y avalo la implementación de la propuesta de modificación del Plan de Estudios del Programa de Licenciatura en Ciencias Computacionales adscrito a la Facultad de Ciencias de la Universidad Autónoma de Baja California, al considerar que cumple con los criterios de pertinencia, actualidad, coherencia académica y alineación con estándares nacionales e internacionales.

Atentamente,
"El saber de mis hijos hará mi grandeza"

Dr. Juan Pablo Soto Barrera
Coordinador General

Facultad Interdisciplinaria de Ciencias Exactas y Naturales
Universidad de Sonora



C.c.p. Archivo

Edificio 34-I planta baja, Blvd. Luis Encinas y Rosales s/n,
Colonia Centro, C.P. 83000 Hermosillo, Sonora, México
(662) 259 21 53 y 259 21 54, extensión 8153 y 8154
ficen@unison.mx
<https://fien.unison.mx>





San Luis Potosí, S.L.P. a 16 de abril de 2026

Estimada Dra. María Victoria Meza Kubo
Coordinadora de la Lic. en Ciencias Computacionales de la UABC

Antes que nada, le envío un cordial saludo desde la Universidad Autónoma de San Luis Potosí (UASLP). Por este medio doy respuesta de la invitación realizada a una servidora, a través de la dirección de la Facultad de Ciencias, para revisar la propuesta de modificación del Plan de Estudios del programa educativo **Licenciatura en Ciencias Computacionales**. Mis comentarios son los siguientes:

1. En la fundamentación de la profesión (2.2), se sugiere revisar si se pueden actualizar las referencias de la IEEE y la CONAIC (Ej. "Aseguramiento de competencias blancas para la calidad a nivel internacional de Programas Educativos en TIC", libro publicado por la CONAIC).
2. En la fundamentación institucional, se reconoce la estrategia del nivel de inglés y que sean dos cursos obligatorios en inglés.
3. UA por área de conocimiento (3.5), una inquietud ¿en qué materia(s) se trabaja el tema del Cómputo en la Nube?...puesto que es uno de los temas que se mencionan en la Justificación.
4. Titulación (5.3), la última viñeta causa confusión, se sugiere indicar si el programa académico tiene las acreditaciones mencionadas.

A manera de conclusiones: La propuesta de actualización curricular es pertinente en concordancia de la disciplina que ofrece, las características del Plan de Estudios están alineadas con el Perfil de Egreso, con respecto al plan de estudios anterior no se generan más semestres para el estudiante. De acuerdo a lo expuesto, la propuesta también está alineada a los estatutos de la Institución y a su Modelo Educativo. La flexibilidad para la obtención de créditos optativos es muy amplia, lo que le permite al estudiante realizar las actividades de acuerdo a sus preferencias.

No habiendo más comentarios, me despido de usted.

Atentamente,


Dra. Sandra Edith Nava Muñoz
Coordinadora Académica del Posgrado en Computación
de la Facultad de Ingeniería de la UASLP



www.uaslp.mx

Av. Manuel Nava 8
Zona Universitaria • CP 78290
San Luis Potosí, S.L.P.
tel. (444) 826 2330 al 39
fax (444) 826 2336

Respuesta a las observaciones de la Dra. Sandra Nava Muñoz.

1. En la fundamentación de la profesión (2.2), se sugiere revisar si se pueden actualizar las referencias de la IEEE y la CONAIC (Ej. "Aseguramiento de competencias blancas para la calidad a nivel internacional de Programas Educativos en TIC", libro publicado por la CONAIC).

R: Se actualizó el párrafo y referencia de la publicación de la IEEE 2026; sin embargo, la referencia de CONAIC sugerida no pudo ser actualizada, ya que se refiere a los Modelos Curriculares, documento del cual no ha tenido una actualización.

2. En la fundamentación institucional, se reconoce la estrategia del nivel de inglés y que sean dos cursos obligatorios en inglés.

R: Gracias.

3. UA por área de conocimiento (3.5), una inquietud ¿en qué materia(s) se trabaja el tema del Cómputo en la Nube?...puesto que es uno de los temas que se mencionan en la Justificación.

R: La propuesta integra como curso obligatorio Sistemas Distribuidos. El estudio de sistemas distribuidos es más amplio y fundamental que el cómputo en la nube, ya que proporciona los principios que sustentan su funcionamiento. Por ello, el tema de nube se incorpora como una aplicación moderna dentro del curso, garantizando una formación sólida y no limitada a tecnologías específicas. Este enfoque se complementa con los cursos de Redes y Ciberseguridad, que proporcionan los fundamentos de comunicación y seguridad necesarios, así como con Bases de Datos, donde se incorporan tecnologías modernas como bases de datos NoSQL, contenedores y servicios de bases de datos en la nube, fortaleciendo la formación del estudiante en el desarrollo y gestión de soluciones en entornos cloud.

4. Titulación (5.3), la última viñeta causa confusión, se sugiere indicar si el programa académico tiene las acreditaciones mencionadas.

R: La última viñeta corresponde a otra forma de titulación. Considerando su recomendación se incluyó el texto "Para el caso específico de Ciencias Computacionales, actualmente el Programa está acreditado por el organismo CONAIC".

9.2. Aprobación del Consejo Técnico

SESIÓN ORDINARIA DE CONSEJO TÉCNICO DE LA FACULTAD DE CIENCIAS 15 de abril de 2026

Siendo las 12:00 horas del miércoles 15 de abril de 2026, las y los integrantes del Consejo Técnico de la Facultad de Ciencias se reunieron en sesión ordinaria en el Audiovisual A de la Facultad de Ciencias para atender la convocatoria emitida por el Presidente, Dr. Alberto Leopoldo Morán y Solares, bajo el siguiente:

ORDEN DEL DÍA

1. Apertura de la sesión.
2. Lista de asistencia y declaración del quórum legal.
3. Elección de dos escrutadores, uno por el personal académico y otro por el alumnado.
4. Lectura y aprobación del orden del día.
5. Lectura y aprobación del Acta de la sesión anterior.
6. Presentación, discusión y, en su caso aprobación, de la propuesta de modificación del Plan de Estudios del Programa Educativo de Biología.
7. Presentación, discusión y, en su caso aprobación, de la propuesta de modificación del Plan de Estudios del Programa Educativo de Física.
8. Presentación, discusión y, en su caso aprobación, de la propuesta de modificación del Plan de Estudios del Programa de Ciencias Computacionales.
9. Presentación del informe de la Comisión Especial sobre la propuesta del Reglamento para la Colección de Vertebrados de la Facultad de Ciencias.
10. Asuntos generales.
11. Clausura de la sesión.

----- DESAHOGO DEL ORDEN DEL DÍA -----

1. Apertura de la sesión

El Presidente del Consejo, Dr. Alberto Leopoldo Morán y Solares, da la bienvenida a la sesión del Consejo Técnico de la Facultad de Ciencias y explica el proceder de los principales puntos a tratar en la sesión.

2. Lista de asistencia y declaración del quórum legal

Se cuenta con la presencia de nueve propietarios y tres suplentes que fungen como propietarios, por lo que se declara el quórum legal.

3. Elección de dos escrutadores, uno por el personal académico y otro por el alumnado

El Presidente solicita propuestas para la elección de las o los escrutadores para la sesión, recordando que su conformación debe incluir un representante del personal académico y uno por parte del alumnado. Por parte del alumnado se propone a Laura Esther Oropeza Guerrero, mientras que por parte del personal académico se propone al Dr. Raúl Casillas Figueroa. Al no haber más propuestas, se somete a votación y se aprueba por unanimidad.

4. Lectura y aprobación del orden del día

El Presidente le solicita al Secretario dé lectura al orden del día. Se somete a votación el orden del día y se aprueba por unanimidad.

5. Lectura y aprobación del Acta de la sesión anterior

El Presidente del Consejo recuerda que, con base en un acuerdo previo, se envió con antelación el Acta de la sesión anterior para su revisión, procediendo a preguntar si existe alguna observación o comentario. Al no haber comentarios, el Presidente somete a votación la aprobación del Acta de la sesión anterior y se aprueba por unanimidad.

6. Presentación, discusión y, en su caso aprobación, de la propuesta de modificación del Plan de Estudios del Programa Educativo de Biología

Para el desahogo de este punto, el Presidente del Consejo solicitó al pleno la autorización para la integración a la sesión del Dr. Ulises Gregorio Pacheco Bardullas y del Dr. Manuel Iván Ocegueda Miramontes, con el propósito de presentar las propuestas de modificación de los programas educativos de Biología y Física, respectivamente. La solicitud fue aprobada por unanimidad. Se hace constar que la Dra. María Victoria Meza Kubo participó en su carácter de integrante del Consejo Técnico y como responsable de presentar la propuesta de Ciencias Computacionales.

Acto seguido, el Presidente solicitó al Dr. Pacheco la presentación de la propuesta de modificación del plan de estudios de la Licenciatura en Biología. Durante su intervención, se destacó que la propuesta tiene como objetivo consolidar un perfil de egreso acorde con las exigencias actuales en áreas como conservación, biología molecular, biotecnología, biodiversidad, ecología, así como la gestión y manejo de recursos naturales. En la fase de discusión, se abordó la pertinencia de eliminar el tronco común, señalando que la modificación busca fortalecer una estructura curricular sólida y actualizada, que permita al alumnado adquirir dominio de diversas metodologías desde el inicio de su formación. Asimismo, se reflexionó sobre los momentos adecuados para incorporar certificaciones adicionales al plan de estudios.

Concluida la presentación y discusión, la propuesta de modificación del plan de estudios de la Licenciatura en Biología fue sometida a votación, siendo aprobada con diez votos a favor y dos abstenciones.

7. Presentación, discusión y, en su caso aprobación, de la propuesta de modificación del Plan de Estudios del Programa Educativo de Física

Para el desahogo de este punto, el Presidente del Consejo solicitó al Dr. Manuel Iván Ocegueda Miramontes la presentación de la propuesta de modificación del plan de

estudios de la Licenciatura en Física. Durante su intervención, se expuso que la propuesta se fundamenta en los resultados de evaluaciones internas y externas del programa educativo, así como en su pertinencia social y contexto profesional, incorporando además la retroalimentación de personas empleadoras y egresadas. En la discusión, se analizó la selección de certificaciones y la posibilidad de su actualización o incorporación en etapas posteriores del plan de estudios.

Concluido el análisis, la propuesta de modificación del plan de estudios de la Licenciatura en Física fue sometida a votación, siendo aprobada con once votos a favor y una abstención.

8. Presentación, discusión y, en su caso aprobación, de la propuesta de modificación del Plan de Estudios del Programa Educativo de Ciencias Computacionales

Para el desahogo de este punto, el Presidente del Consejo solicitó a la Dra. María Victoria Meza Kubo la presentación de la propuesta de modificación del plan de estudios de la Licenciatura en Ciencias Computacionales. Durante su intervención, se señaló que la propuesta se sustenta en un diagnóstico interno y externo del programa educativo, orientado al fortalecimiento de habilidades técnicas y habilidades blandas, así como a la vinculación mediante proyectos interdisciplinarios y al desarrollo de competencias transversales. Con base en estos elementos, se planteó una actualización disciplinar del programa.

Concluido el análisis, la propuesta de modificación del plan de estudios de la Licenciatura en Ciencias Computacionales fue sometida a votación, siendo aprobada con ocho votos a favor, un voto en contra y tres abstenciones.

9. Presentación del informe de la Comisión Especial sobre la propuesta del Reglamento para la Colección de Vertebrados de la Facultad de Ciencias.

El Presidente solicita propuestas para abordar los puntos faltantes del orden del día, con el propósito de respetar el acuerdo de que el tiempo de la sesión no exceda dos horas. Se propusieron las siguientes opciones:

- Dejar la sesión abierta para concluir los puntos faltantes del orden del día.
- Cerrar la sesión actual y convocar a una nueva sesión ordinaria el día 22 de abril de 2026 para concluir los puntos faltantes del orden del día.

Se someten a votación las propuestas y se aprueba por unanimidad cerrar la sesión actual y convocar a una nueva sesión ordinaria el día 22 de abril de 2026 para concluir los puntos faltantes del orden del día.

10. Asuntos generales.

No se presentaron temas para asuntos generales.

11. Clausura de la sesión

Siendo las 14:00 horas se da por concluida la sesión. Se anexa lista de asistencia.



Dr. Alberto Leopoldo Morán y Solares
Presidente del Consejo Técnico



Dr. José Israel Paez Ornelas
Secretario del Consejo Técnico



Dr. Raúl Castillas Figueroa
Escrutador



Laura Esther Oropeza Guerrero
Escrutadora



Wences De la Rosa

R. Ramos

Alonso
Luis

Handwritten signatures and initials on the left margin, including "CYK" and "CYK."

Handwritten signatures and initials at the bottom of the page.

9.4. Validación por la Coordinación General de Formación Profesional



"2026, Año de la Educación para la Construcción de la Paz"

COORDINACIÓN GENERAL DE FORMACIÓN PROFESIONAL

Oficio núm. 127/2026-1

Asunto: Validación de Programa Educativo

DR. ALBERTO LEOPOLDO MORÁN Y SOLARES

Director de la Facultad de Ciencias

Ensenada, B.C.

Presente. -

En atención a la solicitud para la creación del Programa Educativo de **Licenciatura en Ciencias Computacionales**, y derivado de la revisión técnica realizada por esta Coordinación General a mi cargo, me permito comunicarle lo siguiente:

Una vez analizada la propuesta y habiéndose verificado el cumplimiento de los requisitos académicos y curriculares, así como la pertinencia del plan de estudios propuesto, esta **Coordinación General de Formación Profesional otorga su VALIDACIÓN** al citado programa.

Lo anterior, en cumplimiento de lo instruido por la Secretaría General mediante el oficio circular **OF.CIR. 692-2026**, relativo a la integración del orden del día de la segunda sesión ordinaria del H. Consejo Universitario del presente año.

Se le recuerda que, para la integración del expediente definitivo que deberá remitirse a la Secretaría General a más tardar el **24 de abril**, es necesario adjuntar a este oficio la totalidad de la documentación probatoria, incluyendo el Acta de Consejo Técnico de su unidad académica y la propuesta curricular completa.

Sin más por el momento, agradezco su compromiso con la excelencia académica de la Facultad a su digno cargo y aprovecho la ocasión para enviarle un cordial saludo.

ATENTAMENTE

Mexicali, Baja California, a 24 de abril de 2026

"POR LA REALIZACIÓN PLENA DEL SER"

COORDINADORA GENERAL

**Universidad Autónoma
de Baja California**

24 de abril de 2026

**Coordinación General
de Formación Profesional**

DESPACHADO

DRA. YESSICA ESPINOSA DÍAZ



**Universidad Autónoma
de Baja California**

**RECTORÍA
COORDINACIÓN GENERAL
DE FORMACIÓN PROFESIONAL**

C.c.p. Dr. Luis Enrique Palafox Maestre, Rector de la UABC.
C.c.p. M.J. Edith Montiel Ayala, Secretaria General.
C.c.p. Minutario
YED/Ililiana

*Recibido
24-04-2026*

10. Referencias

- Asociación Nacional de Universidades e Instituciones de Educación Superior (ANUIES). (2023). *Anuario estadístico de educación superior*. <https://www.anuies.mx>
- Brovelli, M. (2001). Asesoramiento en educación: el asesoramiento curricular. *Fundamentos en Humanidades*, 1(1), 56-79. <https://www.redalyc.org/pdf/184/18400105.pdf>
- Centro de Investigación y de Estudios Avanzados (CINVESTAV). (s.f.). *Historia de la computación en México*. <https://www.cinvestav.mx>
- Centro Nacional de Evaluación para la Educación Superior. (2021, 15 de octubre). Sobre Ceneval. https://ceneval.edu.mx/sobre_el_ceneval-perfil_institucional/
- Consejo Nacional de Acreditación en Informática y Computación (CONAIC). (2022). *Marco de referencia para la acreditación de programas de informática y computación*.
- Consejo Nacional de Evaluación de la Política de Desarrollo Social (CONEVAL). (2020). *Medición de la pobreza en México*. <https://www.coneval.org.mx>
- Consejo Nacional de Población (CONAPO). (2019). *Proyecciones de la población de México y de las entidades federativas, 2016–2050*. <https://www.gob.mx/conapo>
- Dell Technologies. (2018). *Digital transformation index*. <https://www.delltechnologies.com>
- Deloitte. (2022). *Global 2022 Gen Z and Millennial Survey*. <https://www2.deloitte.com>

- Forbes. (2020, 4 de junio). Tres tendencias de talento que definirán el futuro laboral. <https://www.forbes.com.mx/tres-tendencias-de-talento-que-definiran-el-futuro-laboral/>
- Gobierno de México. (2023). Tendencias del empleo profesional: primer trimestre 2023. Observatorio laboral-Servicio Nacional del Empleo-Gobierno de México. https://www.observatoriolaboral.gob.mx/static/estudios-publicaciones/Tendencias_empleo.html
- Hays. (2022). Análisis de tendencias y salarios 2022: América Latina. https://www.hays.com.br/documents/63221/0/MX_Analisis_Tendencias_Salarios_2022_Spanish_compressed.pdf
- Hernández-Laos, E., Solís, R., & Stefanovich, A. (2012). *Educación superior y mercado laboral en México*.
- Huerta, L. (2023, junio 26). La importancia de las MIPYME en la economía de México. UNAM Revista Global. https://unamglobal.unam.mx/global_revista/la-importancia-de-las-mipyme-en-la-economia-de-mexico/
- Institute of Electrical and Electronics Engineers (IEEE). (2022). *Future directions in computing technologies*. <https://www.ieee.org>
- Technology predictions 2026 report,” The IEEE Comput. Soc., Washington, DC, USA, 2026. <https://ieeecs-media.computer.org/media/tech-news/>
- Instituto Nacional de Estadística y Geografía (INEGI). (2019). *Clasificación de ocupaciones*. <https://www.inegi.org.mx>
- Instituto Nacional de Estadística y Geografía (INEGI). (2020). *Censo de población y vivienda 2020*. <https://www.inegi.org.mx>

- Moncada, G., Plazas, L., (2015). Conceptos y fundamentos de simulación digital. Ecoeediciones.mx.<https://www.ecoeediciones.mx/wp-content/uploads/2015/08/Conceptos-y-fundamentos-de-simulacion.pdf>
- Organisation for Economic Co-operation and Development (OECD). (2022). *OECD digital economy outlook 2022*. <https://www.oecd.org>
- Organización de las Naciones Unidas (ONU). (2015). *Objetivos de Desarrollo Sostenible*. <https://www.un.org>
- Organización de las Naciones Unidas (ONU). (2019). *Informe sobre los Objetivos de Desarrollo Sostenible*. <https://www.un.org>
- Serna, A. y Castro, A. (2018). Metodología de los estudios de fundamentación para la creación, modificación y actualización de programas educativos de licenciatura. UABC.
http://web.uabc.mx/formacionbasica/documentos/metodologia_con_ficha.pdf
- Universidad Autónoma de Baja California (UABC). (1982). *Reglamento General de Exámenes Profesionales*. Autor.
- Universidad Autónoma de Baja California (UABC). (2004). *Reglamento de Prácticas Profesionales*. Autor.
- Universidad Autónoma de Baja California (UABC). (2007). *Reglamento de Servicio Social*. Autor.
- Universidad Autónoma de Baja California (UABC). (2010). *Guía Metodológica para la creación, modificación y actualización de los programas educativos de la Universidad Autónoma de Baja California*.
<http://www.uabc.mx/formacionbasica/documentos/guiametodol%F3gica.pdf>
- Universidad Autónoma de Baja California (UABC). (2012). *Manual de Tutorías*. Autor.

Universidad Autónoma de Baja California (UABC). (2017a). *Código de Ética de la Universidad Autónoma de Baja California*. Autor.

Universidad Autónoma de Baja California (UABC). (2018). *Modelo educativo de la UABC*. Autor.

Universidad Autónoma de Baja California (UABC). (2021). *Estatuto Escolar*. Autor.

Universidad Autónoma de Baja California (UABC). (2023). *Plan de Desarrollo Institucional 2023-2027*. Autor.

Universidad Autónoma de Baja California (UABC). (2024). *Evaluación interna del programa educativo de la Licenciatura en Ciencias Computacionales*. Documento institucional.

Universidad Autónoma de Baja California. (2020a). Documento ejecutivo para la elaboración de la Evaluación Externa e Interna para la modificación o actualización de Planes de Estudio de Licenciatura de la UABC (Propuesta de instrumentos para estudios).
http://web.uabc.mx/formacionbasica/documentos/documento_con_instrumentos.pdf

Universidad Autónoma de Baja California. (2020b). Estudio de preferencias vocacionales y demanda de carreras profesionales de los estudiantes de Educación Media Superior de Baja California. Coordinación General de Formación Profesional.

Universidad Autónoma de Baja California. (2023). Plan de Desarrollo Institucional 2023 - 2027. http://planeacion.uabc.mx/pdi2023/docs/UABC_PDI_2023-2027_Ejecutivo.pdf

Universidad Autónoma de Baja California. (2023c, 02 de agosto). Indicadores de trayectoria escolar. <https://sindi.uabc.mx/app?opcion=trayectoria-escolar>

Universidad Autónoma de Baja California. (2023d, 02 de agosto). Indicadores de seguimiento e-trayecto.
<https://sindi.uabc.mx/etrayecto-web/app?opcion=seguimiento-etrayecto>

Universidad Autónoma de Baja California. (2025). Misión. Facultad de Ciencias.
<https://ciencias.ens.uabc.mx/mision/>

World Economic Forum (WEF). (2020). *The future of jobs report 2020*.
<https://www.weforum.org>

World Economic Forum (WEF). (2023). *The future of jobs report 2023*.
<https://www.weforum.org>

11. Anexos

11.1. Anexo 1. Formatos metodológicos

FORMATO I.- Problemáticas y competencias profesionales

PROBLEMÁTICAS	COMPETENCIA PROFESIONAL	ÁMBITOS
1. Demanda de soluciones tecnológicas inteligentes capaces de abstraer, modelar y simular fenómenos complejos sociales, científicos y tecnológicos, lo cual limita la toma de decisiones informadas, la optimización de recursos y el desarrollo sostenible.	1. Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.	Local, regional, nacional e internacional.
2. Necesidad de diseñar e integrar infraestructuras tecnológicas resilientes, escalables y seguras que permitan gestionar datos de forma eficiente y garantizar el correcto funcionamiento, conectividad y protección de la información en entornos organizacionales complejos y en constante evolución	2. Desarrollar software de base eficiente y robusto mediante la integración de conocimientos en arquitectura y redes de computadoras y los fundamentos de compiladores y sistemas operativos, para optimizar el rendimiento, la gestión de recursos y la comunicación entre sistemas, asegurando aplicaciones funcionales, seguras, escalables y sostenibles, con una actitud ética, responsable y visionaria.	Organizaciones a nivel nacional e internacional
3. Existe una necesidad urgente de diseñar y desarrollar soluciones de software sustentables, accesibles, inclusivas y centradas en el usuario, capaces de operar en diversas plataformas tecnológicas y responder a las demandas sociales actuales.	3. Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social.	Organizaciones a nivel nacional e internacional
4. Demanda de soluciones tecnológicas inteligentes capaces de abstraer, modelar y simular fenómenos complejos sociales, científicos y tecnológicos, lo cual limita la toma de decisiones informadas, la optimización de recursos y el desarrollo sostenible.	4. Diseñar y evaluar soluciones de cómputo inteligente aplicando principios de inteligencia artificial, minería de datos, e integración de herramientas computacionales, con el propósito de apoyar la toma de decisiones en situaciones complejas propias de contextos científicos, sociales, industriales o tecnológicos, actuando con responsabilidad ética, pensamiento crítico, y colaboración interdisciplinaria	Organizaciones a nivel nacional e internacional

FORMATO 2.- Identificación de las competencias específicas que integran cada competencia profesional

COMPETENCIA PROFESIONAL	COMPETENCIAS ESPECÍFICAS
1. Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.	1.1. Diseñar soluciones algorítmicas eficientes y escalables, aplicando los fundamentos de matemáticas computacionales, y los diversos paradigmas de diseño de algoritmos, para la resolución de problemas en contextos reales y simulados, promoviendo el aprendizaje continuo, el trabajo en equipo y la ética profesional en las decisiones de diseño. 1.2. Analizar y evaluar algoritmos computacionales mediante la aplicación de técnicas formales de complejidad algorítmica, para seleccionar o proponer soluciones a problemas complejos, considerando criterios de corrección, eficiencia, escalabilidad y recursos computacionales disponibles, actuando con responsabilidad ética, pensamiento crítico y compromiso con la mejora continua. 1.3. Desarrollar soluciones que optimicen procesos y recursos, mediante la aplicación de modelado matemático y simulación de problemas complejos, para mejorar la eficiencia en la resolución de problemas e impulsar la innovación tecnológica y científica, con creatividad y perseverancia y manteniendo un enfoque ético en la resolución de problemas.
2. Desarrollar software de base eficiente y robusto mediante la integración de conocimientos en arquitectura y redes de computadoras y los fundamentos de compiladores y sistemas operativos, para optimizar el rendimiento, la gestión de recursos y la comunicación entre sistemas, asegurando aplicaciones funcionales, seguras, escalables y sostenibles, con una actitud ética, responsable y visionaria.	2.1. Diseñar, implementar y evaluar soluciones informáticas que utilicen arquitecturas de sistemas distribuidos, aplicando estrategias de gestión de datos, paradigmas de programación, <i>frameworks</i> y tecnologías de almacenamiento y comunicación en red, para garantizar el acceso eficiente, y el procesamiento y análisis de información en tiempo real, dentro de entornos colaborativos y descentralizados, actuando con responsabilidad, y compromiso ético en el manejo de los datos. 2.2. Gestionar infraestructuras tecnológicas seguras, resilientes y escalables mediante la aplicación de mecanismos de ciberseguridad, análisis de riesgos, normas internacionales y herramientas especializadas, para asegurar la disponibilidad, seguridad y continuidad operativa de los sistemas de información, con responsabilidad ética, compromiso social y enfoque preventivo. 2.3. Diseñar soluciones de software de base considerando el funcionamiento de las computadoras, los fundamentos de lenguajes

	formales, compiladores y sistemas operativos, aplicando los principios de arquitectura de computadoras, teoría de autómatas y programación de sistemas, con el fin de optimizar el rendimiento de los sistemas y uso eficiente de los recursos, actuando con pensamiento crítico y responsabilidad técnica.
3. Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social	3.1 Desarrollar soluciones de software innovadoras y eficientes, aplicando metodologías de desarrollo de software, centradas en el usuario, alineadas con normas de calidad nacionales e internacionales e integrando tecnologías emergentes para su implementación, para planificar, diseñar, implementar y evaluar proyectos de manera sistemática y eficiente, con empatía, sensibilidad social y trabajo en equipo. 3.2 Diseñar sistemas de gestión de bases de datos seguros y eficientes, integrando prácticas de protección de la información y estándares de ciberseguridad, para garantizar la integridad, disponibilidad y confidencialidad de los datos en soluciones informáticas, con responsabilidad técnica y social. 3.3 Gestionar proyectos de desarrollo de software sustentables, empleando metodologías de gestión de proyectos, conocimientos de ingeniería de software, así como herramientas colaborativas y de control de calidad, para asegurar el cumplimiento de objetivos técnicos, económicos y sociales en contextos organizacionales diversos, con liderazgo colaborativo, pensamiento estratégico y visión empresarial.
4. Diseñar y evaluar soluciones de cómputo inteligente aplicando principios de inteligencia artificial, minería de datos, e integración de herramientas computacionales, con el propósito de apoyar la toma de decisiones en situaciones complejas propias de contextos científicos, sociales, industriales o tecnológicos, actuando con responsabilidad ética, pensamiento crítico, y colaboración interdisciplinaria.	4.1 Aplicar fundamentos teóricos y técnicas de la inteligencia artificial, como la representación del conocimiento, los métodos de búsqueda, el aprendizaje automático y los modelos generativos, mediante el análisis de casos, la resolución de problemas y el estudio de referentes teóricos, para abordar situaciones complejas del mundo real, demostrando pensamiento crítico, curiosidad intelectual, apertura al aprendizaje y rigurosidad académica. 4.2 Implementar técnicas de minería de datos a través del procesamiento de conjuntos de datos reales y simulados, el uso de algoritmos de aprendizaje automático y la evaluación de resultados obtenidos, para identificar patrones relevantes que apoyen la toma de decisiones, actuando con responsabilidad ética, compromiso

	con la veracidad de la información, y sensibilidad ante el uso de datos personales. 4.3 Diseñar soluciones de cómputo inteligente que articulen los distintos enfoques estudiados, mediante el desarrollo de proyectos integradores e interdisciplinarios para la resolución de problemáticas del mundo real en contextos científicos, sociales, industriales o tecnológicos, evidenciando sentido ético, creatividad, autonomía, y orientación hacia la mejora social y tecnológica.

FORMATO 3.- Establecimiento de las evidencias de desempeño

Competencia profesional 1. Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

COMPETENCIAS ESPECÍFICAS	EVIDENCIA DE DESEMPEÑO
1.1. Diseñar soluciones algorítmicas eficientes y escalables, aplicando los fundamentos de matemáticas computacionales, y los diversos paradigmas de diseño de algoritmos, para la resolución de problemas en contextos reales y simulados, promoviendo el aprendizaje continuo, el trabajo en equipo y la ética profesional en las decisiones de diseño.	1.1.1. Propuesta técnica de solución algorítmica a un problema real o simulado, que incluya: planteamiento del problema, análisis de requerimientos, elección del paradigma de diseño algorítmico, justificación del enfoque adoptado, y discusión sobre eficiencia y escalabilidad. 1.1.2. Implementación funcional de un algoritmo propuesto en un lenguaje de programación, con código comentado, documentación técnica, validación mediante casos de prueba, y comparativa con al menos una solución alternativa. 1.1.3. Informe técnico que describa el proceso de diseño y desarrollo de una solución, incluyendo decisiones clave, dificultades superadas, colaboración en equipo, y reflexión sobre dilemas éticos en el diseño algorítmico. 1.1.4. Presentación multimedia o exposición oral de la solución algorítmica, que muestre la lógica del diseño de un algoritmo, la justificación del paradigma utilizado, los resultados obtenidos y el impacto potencial en el contexto del problema.
1.2. Analizar y evaluar algoritmos computacionales mediante la aplicación de técnicas formales de complejidad algorítmica, para seleccionar o proponer	1.2.1. Análisis comparativo de dos o más algoritmos para la resolución de un mismo

soluciones a problemas complejos, considerando criterios de corrección, eficiencia, escalabilidad y recursos computacionales disponibles, actuando con responsabilidad ética, pensamiento crítico y compromiso con la mejora continua.	problema, considerando: complejidad temporal y espacial, uso de recursos computacionales, comportamiento ante distintos tamaños de entrada y recomendación argumentada. 1.2.2. Serie de ejercicios resueltos de evaluación algorítmica, con cálculos de eficiencia (notación O), análisis de casos límite, demostraciones de corrección y análisis de desempeño bajo distintos escenarios. 1.2.3. Ensayo técnico o documento reflexivo sobre la importancia de la eficiencia algorítmica en contextos reales, con discusión sobre criterios responsables de selección de algoritmos y consecuencias técnicas, sociales o éticas de decisiones inadecuadas. 1.2.4. Portafolio técnico con evidencias de análisis y evaluación de algoritmos, incluyendo prácticas, gráficas de rendimiento, simulaciones comparativas y reflexiones sobre el proceso de aprendizaje.
1.3. Desarrollar soluciones que optimicen procesos y recursos,	1.3.1. Modelo matemático computacional que represente un problema complejo (como optimización de rutas, asignación de tareas, recursos o procesos), con definición formal de variables, restricciones, función objetivo y justificación del modelo elegido. 1.3.2. Implementación computacional de un modelo propuesto, en un lenguaje de programación o plataforma apropiada, con capacidad de simular distintos escenarios, mostrar resultados en tiempo real y modificar parámetros clave para evaluar el comportamiento del sistema en distintos escenarios. 1.3.3. Informe técnico de la solución implementada, con explicación detallada del modelo, presentación de resultados, análisis de eficiencia, discusión de mejoras posibles y reflexiones sobre el impacto tecnológico o social de su aplicación. 1.3.4. Presentación gráfica o audiovisual (póster académico, tablero interactivo o video explicativo) que comunique el proceso de modelado y simulación, beneficios de la solución, y consideraciones éticas o prácticas de su uso en la vida real.

Competencia profesional 2. Desarrollar software de base eficiente y robusto mediante la integración de conocimientos en arquitectura y redes de computadoras y los fundamentos de compiladores y sistemas operativos, para optimizar el rendimiento, la gestión de recursos y la comunicación entre sistemas, asegurando aplicaciones funcionales, seguras, escalables y sostenibles, con una actitud ética, responsable y visionaria.

COMPETENCIAS ESPECÍFICAS	EVIDENCIA DE DESEMPEÑO
2.1 Diseñar, implementar y evaluar soluciones informáticas que utilicen arquitecturas de sistemas distribuidos, aplicando estrategias de gestión de datos, paradigmas de programación, <i>frameworks</i> y tecnologías de almacenamiento y comunicación en red, para garantizar el acceso eficiente, y el procesamiento y análisis de información en tiempo real, dentro de entornos colaborativos y descentralizados, actuando con responsabilidad, y compromiso ético en el manejo de los datos.	2.1.1. El estudiante desarrollará un proyecto integrador de software distribuido, que implique el diseño, implementación y desarrollo de una solución informática que utilice arquitecturas distribuidas para procesar datos en tiempo real. Este proyecto debe incluir documentación técnica y evidencia funcional que demuestren el correcto funcionamiento y eficiencia del sistema. 2.1.2. El estudiante elaborará un diseño técnico detallado que incluya diagramas de arquitectura, descripción de componentes, flujo de datos y justificación de las tecnologías seleccionadas para el desarrollo de la solución distribuida. 2.1.3. El estudiante recopila en un portafolio de actividades las prácticas realizadas a lo largo del curso, como ejercicios de programación distribuida, manejo de bases de datos distribuidas, uso de frameworks y herramientas de comunicación en red. Este portafolio mostrará el progreso y dominio de los conceptos y habilidades técnicas. 2.1.4. El estudiante redactará un informe sobre aspectos éticos y de seguridad en el manejo de datos en sistemas distribuidos, donde analice riesgos, buenas prácticas y normativas aplicables, demostrando un compromiso con el manejo responsable de la información. 2.1.5. El estudiante presentará y defenderá ante sus compañeros o docentes el funcionamiento de su solución distribuida mediante una presentación/exposición, explicando la arquitectura, decisiones tecnológicas y mecanismos implementados para garantizar la eficiencia y seguridad del sistema.
2.2 Gestionar infraestructuras tecnológicas seguras, resilientes y escalables mediante la aplicación de mecanismos de ciberseguridad, análisis de riesgos, normas internacionales y herramientas especializadas, para asegurar la	2.2.1. El estudiante llevará a cabo prácticas de laboratorio orientadas al diseño, configuración e implementación de redes seguras y resilientes, empleando de manera fundamentada los mecanismos, herramientas y políticas de seguridad abordados en la asignatura.

disponibilidad, seguridad y continuidad operativa de los sistemas de información, con responsabilidad ética, compromiso social y enfoque preventivo.	2.2.2. El estudiante presentará exámenes teórico-prácticos diseñados para reforzar su comprensión de los fundamentos de la ciberseguridad, los marcos y normas internacionales, así como su capacidad de análisis crítico a través de estudios de caso. Estas evaluaciones incluirán preguntas abiertas y de opción múltiple que permitirán valorar tanto el dominio conceptual como la aplicación contextual del conocimiento adquirido. 2.2.3. El estudiante elaborará y expondrá un proyecto, real o simulado, orientado al diseño de la infraestructura tecnológica de una organización. Este proyecto deberá presentarse en formato escrito y mediante una exposición oral, e incluirá como mínimo: el diagrama de red, las políticas de seguridad, el plan de continuidad del negocio, la evaluación de riesgos y el análisis de cumplimiento normativo aplicable. La evaluación considerará la coherencia técnica del diseño, la adecuada aplicación de herramientas y marcos normativos, así como el enfoque ético, el compromiso social y la calidad del trabajo colaborativo.

2.3 Diseñar soluciones de software de base considerando el funcionamiento de las computadoras, los fundamentos de lenguajes formales, compiladores y sistemas operativos, aplicando los principios de arquitectura de computadoras, teoría de autómatas y programación de sistemas, con el fin de optimizar el rendimiento de los sistemas y uso eficiente de los recursos, actuando con pensamiento crítico y responsabilidad técnica.	2.3.1. El estudiante desarrollará un proyecto de software de base, este proyecto debe demostrar la comprensión del funcionamiento interno de la computadora y de los componentes fundamentales del software de base. 2.3.2. El estudiante resolverá una serie de ejercicios prácticos del curso, Los ejercicios pueden realizarse en papel o mediante herramientas de simulación digital, y permitirán aplicar los fundamentos del software de base.. 2.3.3. El estudiante elaborará un informe técnico comparativo de dos o más soluciones de software considerando su eficiencia en el uso de recursos del sistema, como memoria y tiempo de procesamiento. 2.3.4. El estudiante conformará un portafolio técnico, que recopile las prácticas realizadas a lo largo del curso, el portafolio servirá como evidencia del aprendizaje progresivo y del dominio técnico alcanzado.

	2.3.5. El estudiante presentará su proyecto final de software de base mediante una exposición oral o en video. En ella deberá explicar el funcionamiento de su solución, su relación con los conceptos teóricos abordados en clase.

Competencia profesional 3. Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social.

COMPETENCIAS ESPECÍFICAS	EVIDENCIA DE DESEMPEÑO
3.1 Desarrollar soluciones de software innovadoras y eficientes, aplicando metodologías de desarrollo de software, centradas en el usuario, alineadas con normas de calidad nacionales e internacionales e integrando tecnologías emergentes para su implementación, para planificar, diseñar, implementar y evaluar proyectos de manera sistemática y eficiente, con empatía, sensibilidad social y trabajo en equipo.	3.1.1 Documento de planeación del proyecto de software, que incluya: análisis de requerimientos centrado en el usuario, justificación del uso de metodologías (ágil o tradicional), plan de trabajo con cronograma, roles y entregables. 3.1.2 Prototipo funcional o versión inicial del software que: resuelva una problemática social, educativa, empresarial o comunitaria, integre al menos una tecnología emergente (IA, ciencia de datos, computación en la nube, IoT, etc.), y que cuente con interfaz accesible e inclusiva y guía del usuario. 3.1.3 Informe técnico de calidad y cumplimiento de estándares, con: evidencias del uso de buenas prácticas de codificación, aplicación de pruebas de funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad haciendo referencia a normas o estándares (ISO, IEEE, etc.).

	3.1.4 Presentación grupal del proyecto ante usuarios o evaluadores, que: explique cómo se integró la retroalimentación del usuario, destaque el trabajo colaborativo y el proceso de toma de decisiones éticas y muestre evidencias de empatía y sensibilidad hacia las personas usuarias finales. 3.1.5 Bitácora de trabajo o portafolio individual, donde el estudiante: reflexione sobre su rol en el equipo y los aprendizajes adquiridos, identifique desafíos éticos y sociales enfrentados durante el desarrollo y proporcione evidencia de autogestión y autoaprendizaje.

3.2 Diseñar sistemas de gestión de bases de datos seguros y eficientes, integrando prácticas de protección de la información y estándares de ciberseguridad, para garantizar la integridad, disponibilidad y confidencialidad de los datos en soluciones informáticas, con responsabilidad técnica y social.	3.2.1 Modelo lógico y físico de una base de datos, con: normalización adecuada, diseño de índices y claves primarias/foráneas y justificación de la estructura para garantizar eficiencia. 3.2.2 Implementación funcional de un sistema de gestión de bases de datos (SGBD) que: incluya procedimientos almacenados, vistas y triggers, use mecanismos de respaldo y recuperación, y aplique restricciones de integridad. 3.2.3 Informe de prácticas de seguridad implementadas, que evidencie: control de accesos (roles, privilegios, autenticación), cifrado de datos sensibles y comunicación segura, y aplicación de estándares de ciberseguridad (por ejemplo, OWASP, ISO/IEC 27001). 3.2.4 Simulación o evaluación de vulnerabilidades del sistema, con propuesta de medidas correctivas y/o preventivas documentadas. 3.2.5 Presentación técnica del sistema desarrollado, con: enfoque en la protección de datos, justificación técnica del diseño seguro y

	eficiente, y reflexión sobre las implicaciones éticas y sociales del tratamiento de información.

3.3 Gestionar proyectos de desarrollo de software sustentables, empleando metodologías de gestión de proyectos, conocimientos de ingeniería de software, así como herramientas colaborativas y de control de calidad, para asegurar el cumplimiento de objetivos técnicos, económicos y sociales en contextos organizacionales diversos, con liderazgo colaborativo, pensamiento estratégico y visión empresarial.	3.3.1 Plan de proyecto de software, que incluya: objetivos claros (técnicos, económicos y sociales, dentro del marco de Desarrollo Sostenible), alcance, cronograma, presupuesto y matriz de riesgos y justificación del enfoque metodológico (ágil, tradicional o híbrido). 3.3.2 Bitácora o tablero digital de seguimiento (Scrum board, Gantt, Kanban, etc.) que evidencie: asignación de tareas, seguimiento y control de avances, uso de herramientas colaborativas (Trello, Jira, GitHub Projects, Notion, etc.) y toma de decisiones estratégicas documentadas. 3.3.3 Informe de evaluación de calidad del proyecto, que contemple: indicadores de calidad del producto y del proceso, reporte de verificación y pruebas, lecciones aprendidas y mejora continua, validación y evaluación del cumplimiento de objetivos. 3.3.4 Presentación ejecutiva del proyecto ante un comité evaluador o stakeholders, que: destaque los resultados obtenidos y el impacto del proyecto, demuestre el enfoque sustentable y la contribución organizacional y fundamente la toma de decisiones desde una visión estratégica y empresarial. 3.3.5 Reflexión individual o grupal, sobre: el papel desempeñado como líder o colaborador del proyecto, los aprendizajes vinculados al trabajo en equipo, la ética y la gestión sostenible.

Competencia profesional 4. Diseñar y evaluar soluciones de cómputo inteligente aplicando principios de inteligencia artificial, minería de datos, e integración de herramientas computacionales, con el propósito de apoyar la toma de decisiones en situaciones complejas propias de contextos científicos, sociales, industriales o tecnológicos, actuando con responsabilidad ética, pensamiento crítico, y colaboración interdisciplinaria.

COMPETENCIAS ESPECÍFICAS	EVIDENCIA DE DESEMPEÑO
4.1 Aplicar fundamentos teóricos y técnicas de la inteligencia artificial, como la representación del conocimiento, los métodos de búsqueda, el aprendizaje automático y los modelos generativos, mediante el análisis de casos, la resolución de problemas y el estudio de referentes teóricos, para abordar situaciones complejas del mundo real, demostrando pensamiento crítico, curiosidad intelectual, apertura al aprendizaje y rigurosidad académica.	4.1.1 Estudio de un artículo académico sobre inteligencia artificial, y discusión sobre las metodologías y resultados obtenidos, aplicados a un problema concreto. Esto demuestra la capacidad de conectar teoría con práctica al realizar un análisis profundo de casos reales de IA y de la literatura académica, con un enfoque crítico sobre su aplicabilidad y limitaciones. 4.1.2 Realización de un proceso de tuning de un modelo de aprendizaje automático y análisis de su rendimiento utilizando métricas adecuadas. Esto incluye el ajuste de modelos de IA (como redes neuronales, algoritmos de búsqueda, etc.) mediante validación cruzada, ajuste de hiperparámetros y análisis de sesgo, asegurando su efectividad y precisión. 4.1.3 Modelo de clasificación o sistema de recomendación para una problemática del mundo real. Esto incluye la aplicación práctica de teorías y técnicas, como el aprendizaje supervisado, no supervisado o modelos generativos. 4.1.4 Ensayo sobre las implicaciones éticas de los algoritmos de IA o propuesta de un proyecto personal innovador basado en técnicas emergentes. Esto incluye la reflexión sobre los aspectos éticos, sociales y técnicos de la IA, así como la demostración de pensamiento crítico y curiosidad intelectual mediante la propuesta de mejoras o alternativas. 4.1.5 Presentación formal de los resultados de un proyecto de IA, con una justificación teórica y empírica adecuada. Esto incluye la documentación técnica detallada de los proyectos y resultados obtenidos, con rigor académico, en informes técnicos, presentaciones o investigaciones, mostrando la capacidad para comunicar de manera clara y precisa los hallazgos.
4.2 Implementar técnicas de minería de datos a través del procesamiento de conjuntos de datos reales y simulados, el uso de algoritmos de	4.2.1 Proyecto en el que se procesan datos desorganizados de un ejemplo de la vida real para su análisis posterior. Esto incluye la capacidad para

aprendizaje automático y la evaluación de resultados obtenidos, para identificar patrones relevantes que apoyen la toma de decisiones, actuando con responsabilidad ética, compromiso con la veracidad de la información, y sensibilidad ante el uso de datos personales.	limpiar, transformar y procesar conjuntos de datos reales o simulados, aplicando técnicas de preprocesamiento como normalización, tratamiento de valores faltantes y detección de outliers. 4.2.2 Análisis de patrones de los datos de un problema de la vida real en apoyo a la toma de decisiones. Esto incluye la identificación de patrones clave en los datos que puedan apoyar la toma de decisiones estratégicas o comerciales, utilizando técnicas de minería de datos como reglas de asociación o análisis de correlación. 4.2.3 Modelo de clasificación para predecir el comportamiento de un conjunto de personas en función de su comportamiento histórico. Esto incluye la implementación de modelos de aprendizaje supervisado y no supervisado (como clasificación, regresión o clustering) para identificar patrones y relaciones significativas en los datos. 4.2.4 Evaluación y validación de modelos de regresión, aplicados a problemas de la vida real. Esto incluye la evaluación y validación rigurosa de los modelos mediante técnicas como validación cruzada, ajuste de hiperparámetros y análisis de desempeño, garantizando que los modelos sean aplicables en el contexto real. 4.2.5 Informe detallado sobre cómo se garantiza la privacidad de los datos personales en un proyecto de minería de datos, cumpliendo alguna de las normativas vigentes. Esto incluye la aplicación de principios éticos en el uso de datos, especialmente en lo relacionado con la privacidad, seguridad y veracidad de la información, y la interpretación responsable de los resultados obtenidos.
4.3 Diseñar soluciones de cómputo inteligente que articulen los distintos enfoques estudiados, mediante el desarrollo de proyectos integradores e interdisciplinarios para la resolución de problemáticas del mundo real en contextos científicos, sociales, industriales o tecnológicos, evidenciando sentido ético, creatividad, autonomía, y orientación hacia la mejora social y tecnológica.	4.3.1 Diseño de un algoritmo para un caso práctico que elimine o minimice el impacto de sesgos inherentes al problema. Esto incluye la consideración de principios éticos en el diseño de las soluciones, asegurando que se respeten la privacidad, la equidad y los impactos sociales y ambientales de las tecnologías implementadas. 4.3.2 Sistema inteligente para resolver un problema de la vida real, combinando análisis de datos, modelado predictivo y tecnologías de optimización. Esto incluye el diseño y desarrollo de soluciones que integran enfoques de cómputo inteligente (IA, aprendizaje automático, etc.) para abordar problemas complejos en contextos científicos, sociales, industriales o tecnológicos.

	4.3.3 Aplicación inteligente innovadora que utilice elementos tecnológicos y análisis de datos para la resolución de un problema práctico. Esto incluye la aplicación creativa de tecnologías emergentes para desarrollar soluciones innovadoras a problemas del mundo real, demostrando pensamiento original y habilidades para generar nuevas ideas. 4.3.4 Sistema computacional inteligente para la optimización de los recursos en un caso de la vida real a través del uso de elementos tecnológicos de vanguardia. Esto incluye la capacidad para abordar y resolver problemas complejos de forma autónoma, utilizando herramientas y enfoques adecuados de cómputo inteligente para generar soluciones efectivas.

FORMATO 4.- Análisis de competencias específicas en conocimientos, habilidades, destrezas, actitudes y valores

Competencia Profesional 1: Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

COMPETENCIAS ESPECÍFICAS	CONOCIMIENTOS (Saber)	HABILIDADES Y DESTREZAS (Hacer)	ACTITUDES Y VALORES (Ser)
1.1. Diseñar soluciones algorítmicas eficientes y escalables, aplicando los fundamentos de matemáticas computacionales, y los diversos paradigmas de diseño de algoritmos, para la resolución de problemas en contextos reales y simulados, promoviendo el aprendizaje continuo, el trabajo en equipo y la ética profesional en las decisiones de diseño.	Paradigmas de diseño de algoritmos (divide y vencerás, voraces, programación dinámica, backtracking, etc.) Fundamentos de matemáticas computacionales aplicadas a algoritmos (teoría de grafos, álgebra lineal, combinatoria) Estructuras de datos (pilas, colas, listas, árboles, grafos) Tipos de complejidad computacional y análisis asintótico	Diseñar soluciones algorítmicas considerando criterios de eficiencia, escalabilidad y claridad estructural Implementar algoritmos funcionales en diferentes lenguajes de programación Analizar problemas reales y traducirlos en modelos computacionales resolubles Evaluar y comparar alternativas algorítmicas para una misma problemática Comunicar de manera clara y efectiva decisiones de diseño y resultados obtenidos	Compromiso con la calidad en el diseño algorítmico Ética profesional en la solución de problemas computacionales Perseverancia ante desafíos algorítmicos complejos Disposición al aprendizaje continuo y la mejora técnica Colaboración y responsabilidad compartida

COMPETENCIAS ESPECÍFICAS	CONOCIMIENTOS (Saber)	HABILIDADES Y DESTREZAS (Hacer)	ACTITUDES Y VALORES (Ser)
1.2. Analizar y evaluar algoritmos computacionales mediante la aplicación de técnicas formales de complejidad algorítmica, para seleccionar o proponer soluciones a problemas complejos, considerando criterios de corrección, eficiencia, escalabilidad y recursos computacionales disponibles, actuando con responsabilidad ética, pensamiento crítico y compromiso con la mejora continua.	Análisis de complejidad algorítmica (mejor, peor y caso promedio) Fundamentos de teoría de la computación (modelos de cómputo, funciones computables, clases P y NP) Herramientas y entornos para medición empírica del rendimiento algorítmico Técnicas de demostración de corrección algorítmica	Aplicar técnicas formales para determinar la eficiencia de un algoritmo Identificar cuellos de botella y puntos críticos de rendimiento Comparar resultados teóricos y experimentales del comportamiento de algoritmos Diagnosticar errores o ineficiencias en el código fuente de algoritmos Diseñar e implementar experimentos computacionales para validar hipótesis	Rigor analítico en la evaluación de soluciones Pensamiento crítico para la toma de decisiones técnicas Responsabilidad técnica en la validación de resultados Honestidad y transparencia en la presentación de resultados Curiosidad intelectual y disposición al análisis profundo

COMPETENCIAS ESPECÍFICAS	CONOCIMIENTOS (Saber)	HABILIDADES Y DESTREZAS (Hacer)	ACTITUDES Y VALORES (Ser)
1.3. Desarrollar soluciones que optimicen procesos y recursos, mediante la aplicación de modelado matemático y simulación de problemas complejos, para mejorar la eficiencia en la resolución de problemas e impulsar la innovación tecnológica y científica, con creatividad y perseverancia y manteniendo un enfoque ético en la resolución de problemas.	Técnicas de modelado matemático (programación lineal, no lineal, redes de flujo, problemas de transporte, etc.) Métodos numéricos y simulación computacional Fundamentos de optimización computacional Simulación de eventos discretos y continuos Técnicas de análisis de sensibilidad y validación de modelos	Formular modelos matemáticos de problemas complejos para su solución computacional Implementar soluciones optimizadas a partir de modelos matemáticos en plataformas de programación o software especializado Evaluar la eficiencia de procesos mediante indicadores cuantitativos Adaptar modelos a diferentes escenarios y restricciones del entorno Integrar herramientas computacionales para explorar escenarios y generar reportes	Creatividad e innovación en la formulación de modelos Responsabilidad en el uso ético de soluciones automatizadas Compromiso con la eficiencia y mejora continua Visión sistémica para abordar problemas complejos Autonomía intelectual y disciplina en el trabajo académico

Competencia Profesional 2: Desarrollar software de base eficiente y robusto mediante la integración de conocimientos en arquitectura y redes de computadoras y los fundamentos de compiladores y sistemas operativos, para optimizar el rendimiento, la gestión de recursos y la comunicación entre sistemas, asegurando aplicaciones funcionales, seguras, escalables y sostenibles, con una actitud ética, responsable y visionaria.

COMPETENCIAS ESPECÍFICAS	CONOCIMIENTOS (Saber)	HABILIDADES Y DESTREZAS (Hacer)	ACTITUDES Y VALORES (Ser)
2.1 Diseñar, implementar y evaluar soluciones informáticas que utilicen arquitecturas de sistemas distribuidos, aplicando estrategias de gestión de datos, paradigmas de programación, <i>frameworks</i> y tecnologías de almacenamiento y comunicación en red, para garantizar el acceso eficiente, y el procesamiento y análisis de información en tiempo real, dentro de entornos colaborativos y descentralizados, actuando con responsabilidad, y compromiso ético en el manejo de los datos.	Paradigmas de programación (funcional, orientada a objetos, concurrente,) Frameworks y herramientas (Spring Boot, Node.js, Docker, AWS, ...) Sistemas y Arquitecturas de sistemas distribuidos (cliente-servidor, peer-to-peer, cloud computing, contenedores, ...) Tecnologías y protocolos de comunicación en red (TCP/IP, APIs, WebSockets, ...) Redes de datos Seguridad informática y ética profesional	Analizar problemas Pensamiento algorítmico Implementar algoritmos Diagnosticar y resolver problemas Promover buenas prácticas en el desarrollo de software seguro Utilizar técnicas de comunicación eficientes Aplicar principios de seguridad y mecanismos de protección Colaboración en equipos multidisciplinarios Identificar vulnerabilidades y riesgos Interpretar y aplicar marcos normativos y legales en materia de seguridad y privacidad	Pensamiento crítico y abstracto Ética en el desarrollo de software Responsabilidad técnica y profesional Compromiso con la calidad y la confiabilidad ética en el manejo de los datos Colaboración y trabajo en equipo Compromiso ético Conciencia legal y social Disciplina y rigor Disposición al aprendizaje continuo Colaboración y responsabilidad compartida

COMPETENCIAS ESPECÍFICAS	CONOCIMIENTOS (Saber)	HABILIDADES Y DESTREZAS (Hacer)	ACTITUDES Y VALORES (Ser)
2.2 Gestionar infraestructuras tecnológicas seguras, resilientes y escalables mediante la aplicación de mecanismos de ciberseguridad, análisis de riesgos, normas internacionales y herramientas especializadas, para asegurar la disponibilidad, seguridad y continuidad operativa de los sistemas de información, con responsabilidad ética, compromiso social y enfoque preventivo.	Fundamentos de seguridad Gestión de infraestructuras tecnológicas (arquitectura de redes, servidores, servicios en la nube, almacenamiento, virtualización, ...) Normas y estándares internacionales de seguridad Herramientas especializadas de seguridad (Firewalls, Nessus, Metasploit, Nmap, Wireshark, ...) Aspectos Legales, Sociales y Éticos de las Tecnologías de la Información y la Comunicación.	Aplicar principios de seguridad y mecanismos de protección Colaboración en equipos multidisciplinarios Identificar vulnerabilidades y riesgos Interpretar y aplicar marcos normativos y legales en materia de seguridad y privacidad Evaluar el impacto ético y social	Pensamiento crítico y abstracto Ética en el desarrollo de software Responsabilidad técnica y profesional Compromiso con la calidad y la confiabilidad ética en el manejo de los datos Colaboración y trabajo en equipo Compromiso ético Conciencia legal y social Disposición al aprendizaje continuo Colaboración y responsabilidad compartida Transparencia y rendición de cuentas

COMPETENCIAS ESPECÍFICAS	CONOCIMIENTOS (Saber)	HABILIDADES Y DESTREZAS (Hacer)	ACTITUDES Y VALORES (Ser)
2.3 Diseñar soluciones de software de base considerando el funcionamiento de las computadoras, los fundamentos de lenguajes formales, compiladores y sistemas operativos, aplicando los principios de arquitectura de computadoras, teoría de autómatas y programación de sistemas, con el fin de optimizar el rendimiento de los sistemas y uso eficiente de los recursos, actuando con pensamiento crítico y responsabilidad técnica.	Arquitectura de computadoras y funcionamiento interno de las computadoras Sistemas Operativos (Unix, Linux) Programación de sistemas (bajo y alto nivel, llamadas al sistema, lenguaje ensamblador, C, C++ ...,) Teoría de autómatas Compiladores	Electrónica digital y lógica computacional Gestión de procesos e hilos, sistemas de archivos, programación de sistemas Manejo de la consola en Unix Lenguajes formales y gramáticas Análisis léxico, sintáctico, semántico Optimización de algoritmos Manejo de estructuras de datos complejos	Pensamiento crítico y abstracto Ética en el desarrollo de software Responsabilidad técnica y profesional Colaboración y trabajo en equipo Compromiso ético Conciencia legal y social Disciplina y rigor Disposición al aprendizaje continuo Colaboración y responsabilidad compartida Transparencia y rendición de cuentas

Competencia Profesional 3: Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social.

COMPETENCIAS ESPECÍFICAS	CONOCIMIENTOS (Saber)	HABILIDADES Y DESTREZAS (Hacer)	ACTITUDES Y VALORES (Ser)
3.1 Desarrollar soluciones de software innovadoras y eficientes, aplicando metodologías de desarrollo de software, centradas en el usuario, alineadas con normas de calidad nacionales e internacionales e integrando tecnologías emergentes para su implementación, para planificar, diseñar, implementar y evaluar proyectos de manera sistemática y eficiente, con empatía, sensibilidad social y trabajo en equipo.	Fundamentos de ingeniería de software y ciclo de vida del desarrollo. Metodologías de desarrollo ágil (Scrum, Kanban, XP) y tradicionales (cascada, RUP). Normas y estándares de calidad en software (ISO/IEC 25010, IEEE 830, etc.). Principios del diseño centrado en el usuario (DCU, UX, accesibilidad e inclusión). Tecnologías emergentes aplicadas al software (IA, ciencia de datos, computación en la nube, IoT, blockchain, etc.). Principios de pruebas y aseguramiento de calidad de software. Fundamentos de bases de datos y arquitecturas de software. Gestión de proyectos (planificación, seguimiento, control y evaluación).	Analizar necesidades del usuario y traducirlas en requerimientos funcionales y no funcionales. Seleccionar e implementar metodologías adecuadas según el contexto del proyecto. Diseñar arquitecturas de software modulares, escalables y mantenibles. Desarrollar soluciones de software que integren tecnologías emergentes. Aplicar pruebas unitarias, de integración, de aceptación y de usabilidad. Documentar de forma clara y estructurada el proceso y producto del desarrollo. Utilizar herramientas de	Empatía y sensibilidad social para diseñar soluciones centradas en las personas y responder a contextos diversos con compromiso humano. Ética profesional en la toma de decisiones técnicas y organizativas, asegurando integridad, honestidad y responsabilidad en el desarrollo del software. Proactividad y apertura al aprendizaje continuo para adaptarse a los cambios tecnológicos y buscar soluciones innovadoras con pensamiento crítico. Compromiso con la calidad, la sostenibilidad y la mejora continua, valorando el impacto social y ambiental de las soluciones propuestas. Colaboración, respeto y comunicación efectiva, trabajando

		control de versiones (como Git), gestión de proyectos y colaboración en línea. Evaluar el impacto social, ético y técnico de la solución desarrollada.	en equipo con actitud inclusiva, tolerante y constructiva ante la diversidad de ideas y personas.

COMPETENCIAS ESPECÍFICAS	CONOCIMIENTOS (Saber)	HABILIDADES Y DESTREZAS (Hacer)	ACTITUDES Y VALORES (Ser)
3.2 Diseñar sistemas de gestión de bases de datos seguros y eficientes, integrando prácticas de protección de la información y estándares de ciberseguridad, para garantizar la integridad, disponibilidad y confidencialidad de los datos en soluciones informáticas, con responsabilidad técnica y social.	Fundamentos de bases de datos relacionales y no relacionales. Modelado entidad-relación y diseño lógico/físico de BD. Lenguaje SQL (DDL, DML, DCL, TCL). Arquitectura y funcionamiento de los SGBD (MySQL, PostgreSQL, MongoDB, etc.). Políticas de seguridad de la información: confidencialidad, integridad, disponibilidad (CIA). Ciberseguridad aplicada a bases de datos (autenticación, control de acceso, cifrado, auditoría). Normativas y estándares de seguridad (ISO/IEC 27001, GDPR, Leyes	Diseñar bases de datos eficientes que respondan a requerimientos funcionales y de seguridad. Implementar controles de acceso, cifrado y auditoría en bases de datos. Detectar y corregir vulnerabilidades comunes en la gestión de datos. Aplicar mecanismos de respaldo, recuperación y monitoreo de datos. Documentar adecuadamente las decisiones de diseño y configuración del sistema. Utilizar herramientas de análisis de	Responsabilidad técnica y social en el manejo de datos sensibles y personales. Ética profesional en la protección de la información y respeto a la privacidad de los usuarios. Compromiso con la calidad y la seguridad, asegurando soluciones confiables y sostenibles. Proactividad y pensamiento crítico para identificar riesgos y aplicar medidas preventivas. Colaboración y comunicación efectiva, trabajando en equipo para el diseño e implementación de soluciones seguras.

	de protección de datos personales). Técnicas de respaldo, recuperación ante fallos y alta disponibilidad.	seguridad y monitoreo de bases de datos.	

COMPETENCIAS ESPECÍFICAS	CONOCIMIENTOS (Saber)	HABILIDADES Y DESTREZAS (Hacer)	ACTITUDES Y VALORES (Ser)
3.3 Gestionar proyectos de desarrollo de software sustentables, empleando metodologías de gestión de proyectos, conocimientos de ingeniería de software, así como herramientas colaborativas y de control de calidad, para asegurar el cumplimiento de objetivos técnicos, económicos y sociales en contextos organizacionales diversos, con liderazgo colaborativo, pensamiento estratégico y visión empresarial.	Ciclo de vida del desarrollo de software. Metodologías de gestión de proyectos: tradicional (PMBOK, cascada), ágil (Scrum, XP, Kanban) e híbridas. Fundamentos de ingeniería de software. Gestión de riesgos, calidad, costos, alcance y cronogramas. Herramientas colaborativas y de control de versiones. Estrategias de liderazgo, negociación y comunicación organizacional. Principios de sostenibilidad en el desarrollo tecnológico. Análisis de impacto técnico, económico y social de proyectos.	Formular y planificar proyectos de software con enfoque estratégico y sustentable. Seleccionar y aplicar metodologías de gestión adecuadas al contexto. Usar herramientas colaborativas para coordinar equipos y tareas. Gestionar cronogramas, recursos, riesgos y entregables con enfoque de calidad. Liderar equipos multidisciplinarios fomentando la corresponsabilidad. Comunicar avances y resultados a distintos públicos (técnicos, ejecutivos, sociales). Evaluar el impacto del proyecto desde una perspectiva técnica, económica y social.	Liderazgo colaborativo, promoviendo el diálogo, la corresponsabilidad y el empoderamiento del equipo. Pensamiento estratégico, anticipando escenarios y tomando decisiones informadas. Visión empresarial y sentido organizacional, orientando los proyectos hacia el valor social y económico. Compromiso con la calidad y la sustentabilidad, considerando el impacto a largo plazo. Ética profesional y responsabilidad social, priorizando decisiones justas, inclusivas y transparentes. Proactividad y apertura al cambio, adaptándose a los desafíos de la gestión real.

Competencia Profesional 4: Diseñar y evaluar soluciones de cómputo inteligente aplicando principios de inteligencia artificial, minería de datos, e integración de herramientas computacionales, con el propósito de apoyar la toma de decisiones en situaciones complejas propias de contextos científicos, sociales, industriales o tecnológicos, actuando con responsabilidad ética, pensamiento crítico, y colaboración interdisciplinaria.

COMPETENCIAS ESPECÍFICAS	CONOCIMIENTOS (Saber)	HABILIDADES Y DESTREZAS (Hacer)	ACTITUDES Y VALORES (Ser)
4.1 Aplicar fundamentos teóricos y técnicas de la inteligencia artificial, como la representación del conocimiento, los métodos de búsqueda, el aprendizaje automático y los modelos generativos, mediante el análisis de casos, la resolución de problemas y el estudio de referentes teóricos, para abordar situaciones complejas del mundo real, demostrando pensamiento crítico, curiosidad intelectual, apertura al aprendizaje y rigurosidad académica.	Fundamentos Teóricos de la Inteligencia Artificial Métodos de Búsqueda Representación y Procesamiento de Conocimiento Aprendizaje Automático Modelos Generativos Fundamentos de Lógica Difusa y Razonamiento con Incertidumbre Procesamiento de Lenguaje Natural Ética en la Inteligencia Artificial Técnicas de Optimización en IA Fundamentos de Hardware y Computación Paralela	Resolver Problemas Complejos Programación y Implementación de Algoritmos de IA Análisis y Preprocesamiento de Datos Modelado Predictivo y Generativo Razonamiento Lógico y Toma de Decisiones Capacidad de Pensamiento Crítico y Evaluación de Resultados Destrezas en la Representación del Conocimiento Comunicación Clara y Efectiva de Resultados Trabajo en Equipo Interdisciplinario Adaptabilidad y Aprendizaje Continuo Ética Profesional y Responsabilidad Social	Curiosidad Intelectual Compromiso con la Ética Profesional Responsabilidad y Transparencia Orientación a la Mejora Social y Humana Pensamiento Crítico Colaboración y Trabajo en Equipo Compromiso con la Excelencia Académica Autonomía y Responsabilidad Personal Integridad y Honestidad Sensibilidad y Respeto por la Diversidad Apertura al Aprendizaje y la Innovación

COMPETENCIAS ESPECÍFICAS	CONOCIMIENTOS (Saber)	HABILIDADES Y DESTREZAS (Hacer)	ACTITUDES Y VALORES (Ser)
4.2 Implementar técnicas de minería de datos a través del procesamiento de conjuntos de datos reales y simulados, el uso de algoritmos de aprendizaje automático y la evaluación de resultados obtenidos, para identificar patrones relevantes que apoyen la toma de decisiones, actuando con responsabilidad ética, compromiso con la veracidad de la información, y sensibilidad ante el uso de datos personales.	Fundamentos de minería de datos Tipos de datos y bases de datos Preprocesamiento de datos Algoritmos de aprendizaje automático Herramientas y lenguajes de programación Técnicas de evaluación de modelos Visualización de datos Estadística descriptiva e inferencial Fundamentos de probabilidad Álgebra lineal Cálculo Fundamentos de teoría de la información Ética en ciencia de datos Normativas de protección de datos personales Evaluación del impacto social de modelos y decisiones basadas en datos	Manipulación y transformación de datos Diseño e implementación de modelos de aprendizaje automático Codificación eficiente en lenguajes de alto nivel Evaluación y ajuste de modelos Automatización de procesos analíticos Interpretación crítica de datos y resultados Resolución de problemas complejos Evaluación de la calidad y confiabilidad de los datos Aprendizaje autónomo Identificación de riesgos éticos Aplicación de principios éticos Conciencia sobre sesgos algorítmicos Respeto por la privacidad y la	Responsabilidad Honestidad y veracidad Respeto por la privacidad Compromiso con la ética profesional Equidad Curiosidad intelectual y pensamiento crítico Rigor y precisión Compromiso con la mejora continua Colaboración y apertura al trabajo en equipo Sensibilidad social

	Interpretación de resultados Pensamiento crítico y analítico Enfoques para la toma de decisiones basada en evidencia	protección de los datos personales Comunicación efectiva de hallazgos técnicos Colaboración interdisciplinaria Pensamiento estratégico y visión sistémica Gestión del tiempo y priorización de tareas	

COMPETENCIAS ESPECÍFICAS	CONOCIMIENTOS (Saber)	HABILIDADES Y DESTREZAS (Hacer)	ACTITUDES Y VALORES (Ser)
4.3 Diseñar soluciones de cómputo inteligente que articulen los distintos enfoques estudiados, mediante el desarrollo de proyectos integradores e interdisciplinarios para la resolución de problemáticas del mundo real en contextos científicos, sociales, industriales o tecnológicos, evidenciando sentido ético, creatividad, autonomía, y orientación hacia la mejora social y tecnológica.	Fundamentos de inteligencia artificial Aprendizaje automático Redes neuronales profundas Procesamiento de lenguaje natural Visión por computadora Algoritmos bioinspirados Computación afectiva y cognitiva Fundamentos de lógica difusa,	Analizar problemas complejos del mundo real Aplicar enfoques interdisciplinarios Evaluar la pertinencia y viabilidad de distintas técnicas de cómputo inteligente Integrar conocimientos previos de IA Diseñar arquitecturas funcionales para soluciones inteligentes	Ética profesional Responsabilidad social Justicia y equidad Sostenibilidad Autonomía Actitud innovadora Actitud crítica y reflexiva Actitud colaborativa Actitud proactiva

	sistemas expertos y razonamiento automatizado Modelado y arquitectura de sistemas inteligentes Gestión de proyectos interdisciplinarios Integración de tecnologías emergentes Interoperabilidad de sistemas Pensamiento sistémico Análisis del impacto social y ético de la tecnología Evaluación de riesgos sociotécnicos Metodologías para resolución de problemas complejos Ética de la inteligencia artificial y tecnologías inteligentes Derechos digitales y protección de datos Autonomía y responsabilidad profesional Enfoques para la sostenibilidad tecnológica	Implementar algoritmos de inteligencia artificial y machine learning Optimizar modelos y sistemas Trabajar en equipos multidisciplinarios Combinar técnicas tradicionales y emergentes Explorar y experimentar con herramientas nuevas Reconocer los dilemas éticos asociados al uso de inteligencia artificial y datos Evaluar el impacto social y ambiental de las soluciones propuestas Incorporar principios de justicia, equidad y sostenibilidad en el diseño de sistemas inteligentes Actuar con autonomía y responsabilidad en el desarrollo tecnológico Mantener una actitud de mejora continua y aprendizaje autónomo	Actitud empática

FORMATO 5.- Identificación de unidades de aprendizaje y unidades de aprendizaje integradoras

Etapas de formación: básica, disciplinaria o terminal

Áreas de conocimiento: entorno social, matemáticas, arquitectura de computadoras, redes, software de base, programación e ingeniería de software, tratamiento de información, interacción hombre-máquina.

Competencia Profesional 1:

COMPETENCIA ESPECÍFICA	CONJUNTO DE UNIDADES DE APRENDIZAJE	UNIDAD DE APRENDIZAJE INTEGRADORA	ETAPA DE FORMACIÓN	ÁREA DE CONOCIMIENTO
1.1. Diseñar soluciones algorítmicas eficientes y escalables, aplicando los fundamentos de matemáticas para computación, y los diversos paradigmas de diseño de algoritmos, para la resolución de problemas en contextos reales y simulados, promoviendo el aprendizaje continuo, el trabajo en equipo y la ética profesional en las decisiones de diseño.	Introducción a las ciencias computacionales Algebra superior Algebra lineal Matemáticas discretas Diseño de algoritmos Estructuras de datos Introducción a la programación Programación orientada a objetos Paradigmas y lenguajes de programación Tecnología y Sociedad Metodología de la Programación	Metodología de la Programación	Disciplinaria	Matemáticas Programación e ingeniería de software

COMPETENCIA ESPECÍFICA	CONJUNTO DE UNIDADES DE APRENDIZAJE	UNIDAD DE APRENDIZAJE INTEGRADORA	ETAPA DE FORMACIÓN	ÁREA DE CONOCIMIENTO
1.2. Analizar y evaluar algoritmos computacionales mediante la aplicación de técnicas formales de complejidad algorítmica, para seleccionar o proponer soluciones a problemas complejos, considerando criterios de corrección, eficiencia, escalabilidad y recursos computacionales disponibles, actuando con responsabilidad ética, pensamiento crítico y compromiso con la mejora continua.	Introducción a las ciencias computacionales Algebra superior Cálculo diferencial Cálculo integral Algebra lineal Matemáticas discretas Probabilidad Diseño de algoritmos Introducción a la programación Estructuras de datos Programación orientada a objetos Análisis de algoritmos Tecnología y Sociedad	Análisis de algoritmos	Disciplinaria	Matemáticas Programación e ingeniería de software

COMPETENCIA ESPECÍFICA	CONJUNTO DE UNIDADES DE APRENDIZAJE	UNIDAD DE APRENDIZAJE INTEGRADORA	ETAPA DE FORMACIÓN	ÁREA DE CONOCIMIENTO
1.3. Desarrollar soluciones que optimicen procesos y recursos, mediante la aplicación de modelado matemático y simulación de problemas complejos, para mejorar la eficiencia en la resolución de problemas e impulsar la innovación tecnológica y científica, con creatividad y perseverancia y manteniendo un enfoque ético en la resolución de problemas.	Introducción a la programación Programación orientada a objetos Metodología de la programación Matemáticas discretas Cálculo vectorial Probabilidad Estadística Métodos Numéricos Ecuaciones diferenciales	Simulación	Terminal	Matemáticas Programación e ingeniería de software

	Investigación de operaciones Graficación Simulación Tecnología y sociedad			

Competencia Profesional 2:

COMPETENCIA ESPECÍFICA	CONJUNTO DE UNIDADES DE APRENDIZAJE	UNIDAD DE APRENDIZAJE INTEGRADORA	ETAPA DE FORMACIÓN	ÁREA DE CONOCIMIENTO
2.1 Diseñar, implementar y evaluar soluciones informáticas que utilicen arquitecturas de sistemas distribuidos, aplicando estrategias de gestión de datos, paradigmas de programación, frameworks y tecnologías de almacenamiento y comunicación en red, para garantizar el acceso eficiente, y el procesamiento y análisis de información en tiempo real, dentro de entornos colaborativos y descentralizados, actuando con responsabilidad, y compromiso ético en el manejo de los datos.	Programación orientada a objetos Estructuras de datos y algoritmos Organización y arquitectura de computadoras Bases de datos Sistemas distribuidos Redes de datos Sistemas operativos Sistemas Distribuidos Tecnología y sociedad Ciberseguridad	Sistemas distribuidos	Terminal	Redes Programación e ingeniería de software/redes

COMPETENCIA ESPECÍFICA	CONJUNTO DE UNIDADES DE APRENDIZAJE	UNIDAD DE APRENDIZAJE INTEGRADORA	ETAPA DE FORMACIÓN	ÁREA DE CONOCIMIENTO
2.2 Gestionar infraestructuras tecnológicas seguras, resilientes y escalables mediante la aplicación de mecanismos de ciberseguridad, análisis de riesgos, normas internacionales y herramientas especializadas, para asegurar la disponibilidad, seguridad y continuidad operativa de los sistemas de información, con responsabilidad ética, compromiso social y enfoque preventivo.	GNU Linux para las Ciencias Redes de datos Bases de datos Ciberseguridad Sistemas Distribuidos	Ciberseguridad	Terminal	Redes

COMPETENCIA ESPECÍFICA	CONJUNTO DE UNIDADES DE APRENDIZAJE	UNIDAD DE APRENDIZAJE INTEGRADORA	ETAPA DE FORMACIÓN	ÁREA DE CONOCIMIENTO
2.3 Diseñar soluciones de software de base considerando el funcionamiento de las computadoras, los fundamentos de lenguajes formales, compiladores y sistemas operativos, aplicando los principios de arquitectura de computadoras, teoría de autómatas y programación de sistemas, con el fin de optimizar el rendimiento de los sistemas y uso eficiente de los recursos, actuando con pensamiento crítico y responsabilidad técnica.	GNU Linux para las Ciencias Matemáticas Discretas Teoría de Autómatas Análisis de Algoritmos Organización y arquitectura de computadoras Sistemas operativos Compiladores	Compiladores	Disciplinaria	Software de base

Competencia Profesional 3:

COMPETENCIA ESPECÍFICA	CONJUNTO DE UNIDADES DE APRENDIZAJE	UNIDAD DE APRENDIZAJE INTEGRADORA	ETAPA DE FORMACIÓN	ÁREA DE CONOCIMIENTO
3.1 Desarrollar soluciones de software innovadoras y eficientes, aplicando metodologías de desarrollo de software, centradas en el usuario, alineadas con normas de calidad nacionales e internacionales e integrando tecnologías emergentes para su implementación, para planificar, diseñar, implementar y evaluar proyectos de manera sistemática y eficiente, con empatía, sensibilidad social y trabajo en equipo.	Ingeniería de software Introducción a la programación Programación orientada a objetos Reingeniería de Procesos Interacción humano computadora	Interacción humano computadora	Terminal	Programación e ingeniería de software, Interacción hombre-máquina

COMPETENCIA ESPECÍFICA	CONJUNTO DE UNIDADES DE APRENDIZAJE	UNIDAD DE APRENDIZAJE INTEGRADORA	ETAPA DE FORMACIÓN	ÁREA DE CONOCIMIENTO
3.2 Diseñar sistemas de gestión de bases de datos seguros y eficientes, integrando prácticas de protección de la información y estándares de ciberseguridad, para garantizar la integridad, disponibilidad y confidencialidad de los datos en soluciones informáticas, con responsabilidad técnica y social.	Base de datos Ciberseguridad Tecnología y sociedad Administración de bases de datos	Administración de bases de datos	Disciplinaria	Tratamiento de información

COMPETENCIA ESPECÍFICA	CONJUNTO DE UNIDADES DE APRENDIZAJE	UNIDAD DE APRENDIZAJE INTEGRADORA	ETAPA DE FORMACIÓN	ÁREA DE CONOCIMIENTO
3.3 Gestionar proyectos de desarrollo de software sustentables, empleando metodologías de gestión de proyectos, conocimientos de ingeniería de software, así como herramientas colaborativas y de control de calidad, para asegurar el cumplimiento de objetivos técnicos, económicos y sociales en contextos organizacionales diversos, con liderazgo colaborativo, pensamiento estratégico y visión empresarial.	Metodología de la Programación Tecnología y sociedad Ingeniería de software Administración de proyectos	Administración de proyectos	Terminal	Entorno social, Programación e ingeniería de software

Competencia Profesional 4:

COMPETENCIA ESPECÍFICA	CONJUNTO DE UNIDADES DE APRENDIZAJE	UNIDAD DE APRENDIZAJE INTEGRADORA	ETAPA DE FORMACIÓN	ÁREA DE CONOCIMIENTO
4.1 Aplicar fundamentos teóricos y técnicas de la inteligencia artificial, como la representación del conocimiento, los métodos de búsqueda, el aprendizaje automático y los modelos generativos, mediante el análisis de casos, la resolución de problemas y el estudio de referentes teóricos, para abordar situaciones complejas del mundo real, demostrando pensamiento crítico, curiosidad intelectual, apertura al aprendizaje y rigurosidad académica.	Fundamentos de Inteligencia Artificial Aprendizaje Automático Aprendizaje Profundo y Modelos Generativos Matemáticas discretas Metodología de la Investigación para las ciencias computacionales Tecnología y sociedad Proyecto Científico de Cómputo Inteligente	Proyecto Científico de Cómputo Inteligente	Terminal	Interacción Hombre Máquina

COMPETENCIA ESPECÍFICA	CONJUNTO DE UNIDADES DE APRENDIZAJE	UNIDAD DE APRENDIZAJE INTEGRADORA	ETAPA DE FORMACIÓN	ÁREA DE CONOCIMIENTO
4.2 Implementar técnicas de minería de datos a través del procesamiento de conjuntos de datos reales y simulados, el uso de algoritmos de aprendizaje automático y la evaluación de resultados obtenidos, para identificar patrones relevantes que apoyen la toma de decisiones, actuando con responsabilidad ética, compromiso con la veracidad de la información, y sensibilidad ante el uso de datos personales.	introducción a la Programación Matemáticas Discretas Estructuras de Datos y Algoritmos Bases de Datos Minería de Datos Aprendizaje Automático Aprendizaje Profundo y Modelos Generativos Tecnología y sociedad Sistemas Distribuidos Proyecto Científico de Cómputo Inteligente	Proyecto Científico de Cómputo Inteligente	Terminal	Interacción Hombre Máquina

COMPETENCIA ESPECÍFICA	CONJUNTO DE UNIDADES DE APRENDIZAJE	UNIDAD DE APRENDIZAJE INTEGRADORA	ETAPA DE FORMACIÓN	ÁREA DE CONOCIMIENTO
4.3 Diseñar soluciones de cómputo inteligente que articulen los distintos enfoques estudiados, mediante el desarrollo de proyectos integradores e interdisciplinarios para la resolución de problemáticas del mundo real en contextos científicos, sociales, industriales o tecnológicos, evidenciando sentido ético, creatividad, autonomía, y orientación hacia la mejora social y tecnológica.	Introducción a la Programación Estructuras de Datos y Algoritmos Matemáticas Discretas Álgebra Lineal Cálculos Probabilidad Fundamentos de Inteligencia Artificial Aprendizaje Automático Aprendizaje Profundo y	Proyecto Científico en Cómputo Inteligente	Terminal	Interacción Hombre Máquina

	Modelos Generativos Tecnología y sociedad Proyecto Científico en Cómputo Inteligente			

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Física, Licenciatura en Matemáticas Aplicadas, Licenciatura en Ciencias Computacionales y Licenciatura en Ciencia de Datos.
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Principios de la Física
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 03 Horas Laboratorio: 00 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Básica
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Jorge Alberto Villavicencio Aguilar Roberto Romo Martínez
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La finalidad de esta unidad de aprendizaje es proporcionar a las y los estudiantes de las licenciaturas en Física, Matemáticas Aplicadas, Ciencias Computacionales y Ciencia de Datos una formación introductoria en los principios fundamentales de la física, utilizando el lenguaje algebraico, la trigonometría y representaciones gráficas para analizar fenómenos naturales y tecnológicos. A través de prácticas de taller con simulaciones, programación en Python/Colab y herramientas de inteligencia artificial. Se pretende que desarrollen pensamiento físico, matemático y computacional, sentando bases sólidas para el estudio posterior de cálculo y de cursos avanzados de ciencias exactas. Se imparte en la etapa básica, es de carácter obligatorio y pertenece al área de conocimiento de Física.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Analizar los conceptos básicos de la física, medición, movimiento, fuerzas, energía, ondas, campos, termodinámica y percepción mediante el empleo del razonamiento algebraico, la representación gráfica y recursos computacionales para describir, explicar y comunicar fenómenos en contextos naturales y tecnológicos, con objetividad, creatividad y responsabilidad social en la reflexión sobre la transversalidad de la ciencia.

IV. EVIDENCIA DE APRENDIZAJE

Portafolio de evidencias que incluya:

- Ejercicios y problemas resueltos, individuales o en equipo, en clase y en plataforma donde se apliquen los principios de la física y se interpreten los resultados
- Productos gráficos, analíticos y computacionales (gráficas, simulaciones, modelos)
- Bitácoras de laboratorio y de simulación con reflexiones sobre el uso responsable de IA
- Informes breves con apoyo gráfico y análisis crítico

Elaboración y presentación en equipo de un proyecto integrador de datos o simulación de fenómenos físicos en contextos naturales y tecnológicos.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Esta asignatura promueve competencias genéricas de impacto en los diferentes perfiles de egreso de los programas educativos de la Facultad de Ciencias de la UABC.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Pensamiento Crítico.

VII. DESARROLLO POR UNIDADES

Unidad I. El lenguaje de la física en problemas básicos

Competencia

Aplicar las unidades básicas del Sistema Internacional y las normas internacionales de medición a través de la utilización de procedimientos de medición para expresar cuantitativamente magnitudes físicas y comunicar los resultados con objetividad, claridad y rigor científico.

Contenido

- 1.1. Encuadre
- 1.2. Principios, leyes, teorías y modelos en la física
 - 1.2.1. Diferencias conceptuales y ejemplos históricos
 - 1.2.2. Rol en la construcción del conocimiento científico
- 1.3. Magnitudes físicas y sistemas de unidades
 - 1.3.1. El Sistema Internacional de Unidades
 - 1.3.2. Patrones de Tiempo, Longitud y Masa
 - 1.3.3. Precisión y cifras significativas
 - 1.3.4. Notación científica y análisis dimensional introductorio

Duración

04 horas

Unidad II. Movimiento y las leyes de Newton

Competencia

Aplicar los conceptos fundamentales de cinemática y dinámica, posición, desplazamiento, velocidad, aceleración y leyes de Newton, para describir y analizar el movimiento de los cuerpos en una y dos dimensiones, mediante el empleo de representaciones gráficas, razonamiento algebraico y herramientas computacionales, con objetividad, claridad y honestidad en la interpretación de los resultados.

Contenido

- 2.1. Movimiento en una dimensión
 - 2.1.1. Concepto de posición y desplazamiento
 - 2.1.2. Velocidad media y velocidad instantánea
 - 2.1.3. Aceleración media e instantánea
 - 2.1.4. Representación gráfica: posición-tiempo y velocidad-tiempo
 - 2.1.5. Relaciones cualitativas entre gráficas y movimiento
- 2.2. Movimiento en dos dimensiones
 - 2.2.1. Introducción al concepto de vector
 - 2.2.2. Sistemas de coordenadas (cartesianas)

- 2.2.3. Composición de movimientos: lanzamiento horizontal y oblicuo
- 2.2.4. Alcance, tiempo de vuelo y altura máxima
- 2.2.5. Análisis gráfico y computacional de trayectorias
- 2.3. Fuerzas y leyes de Newton
 - 2.3.1 Diagramas de cuerpo libre: identificación de fuerzas
 - 2.3.2 Primera ley de Newton y el concepto de inercia
 - 2.3.3 Segunda ley de Newton: relación fuerza-aceleración
 - 2.3.4 Tercera ley de Newton: acción y reacción
 - 2.3.5 Aplicaciones sencillas: movimiento en superficies horizontales, inclinadas y cuerdas con tensión

Duración

06 horas

Unidad III. Conservación de la energía y el momento

Competencia

Aplicar los principios de conservación de la energía y del momento lineal para analizar situaciones mecánicas sencillas, mediante el empleo de los conceptos de trabajo, energía cinética y potencial, potencia e impulso, así como la interpretación gráfica y el razonamiento algebraico, con objetividad, responsabilidad y honestidad en la comunicación de resultados.

Contenido

- 3.1. Trabajo y energía mecánica
 - 3.1.1. Definición de trabajo mecánico y su relación con la fuerza y el desplazamiento
 - 3.1.2. Energía cinética: interpretación y cálculo
 - 3.1.3. Energía potencial gravitatoria y elástica
 - 3.1.4. Potencia mecánica como razón de cambio del trabajo
 - 3.1.5. Representación gráfica de trabajo y energía
- 3.2. Conservación de la energía y del momento
 - 3.2.1. Principio de conservación de la energía mecánica
 - 3.2.2. Trabajo de fuerzas no conservativas y disipación
 - 3.2.3. Impulso y cantidad de movimiento
 - 3.2.4. Conservación del momento lineal en colisiones
 - 3.2.5. Colisiones elásticas e inelásticas: análisis y ejemplos

Duración

04 horas

Unidad IV. Ondas, campos e información: principios universales de la física

Competencia

Analizar los fenómenos ondulatorios y los conceptos de campo eléctrico y magnético para explicar su papel transversal en la naturaleza y la tecnología, mediante la aplicación el principio de superposición, la interferencia y la resonancia, así como la representación gráfica y computacional de campos, con objetividad, rigor científico y responsabilidad en la comunicación de resultados y aplicaciones.

Contenido

- 4.1. Ondas: superposición, interferencia y resonancia
 - 4.1.1. Naturaleza de una onda: características y parámetros
 - 4.1.2. Principio de superposición
 - 4.1.3. Interferencia constructiva y destructiva
 - 4.1.4. Ondas estacionarias y resonancia en cuerdas y tubos sonoros
 - 4.1.5. Aplicaciones: acústica, instrumentos musicales y telecomunicaciones
- 4.2. Campos eléctricos y magnéticos
 - 4.2.1. Concepto de campo y su representación gráfica
 - 4.2.2. Campo eléctrico: fuentes, líneas de campo y ejemplos cotidianos
 - 4.2.3. Campo magnético: imanes, corrientes y fuerzas magnéticas
 - 4.2.4. Interacciones entre campos eléctricos y magnéticos
 - 4.2.5. Aplicaciones tecnológicas: motores eléctricos, almacenamiento magnético y resonancia magnética

Duración

04 horas

Unidad V. Termodinámica y el concepto de entropía en procesos físicos

Competencia

Aplicar los principios de la termodinámica para analizar la transferencia de energía térmica y la irreversibilidad de los procesos naturales, mediante el empleo de los conceptos de calor, temperatura, entropía y eficiencia de máquinas térmicas, con objetividad, responsabilidad y conciencia crítica sobre el impacto energético en la sociedad y el ambiente.

Contenido

- 5.1. Transferencia de energía térmica: calor y temperatura
 - 5.1.1. Temperatura y escalas térmicas
 - 5.1.2. Energía térmica y calor: definición y diferencias
 - 5.1.3. Métodos de transferencia de calor: conducción, convección y radiación

- 5.1.4. Ejemplos en sistemas naturales y tecnológicos
- 5.1.5. Representación gráfica de curvas de calentamiento y cambios de fase
- 5.2. Entropía e irreversibilidad: la segunda ley de la termodinámica
 - 5.2.1. Concepto de entropía como medida del desorden y la irreversibilidad
 - 5.2.2. La segunda ley de la termodinámica y sus enunciados
 - 5.2.3. Procesos reversibles e irreversibles
 - 5.2.4. Máquinas térmicas y eficiencia (motor de Carnot)
 - 5.2.5. Implicaciones de la entropía en fenómenos naturales y tecnológicos

Duración

04 horas

Unidad VI. Física de la percepción y tecnologías afines

Competencia

Aplicar los principios físicos que explican la audición y la visión humanas para analizar fenómenos de percepción y sus aplicaciones en tecnologías modernas de sonido e imagen, mediante el uso de representaciones gráficas y modelos computacionales, con objetividad, creatividad y responsabilidad en la comunicación de resultados y en la valoración del impacto tecnológico en la vida cotidiana.

Contenido

- 6.1. Sonido y audición: frecuencia, intensidad y decibeles
 - 6.1.1. Naturaleza de las ondas sonoras: frecuencia, longitud de onda y velocidad del sonido
 - 6.1.2. Intensidad sonora y escala logarítmica de decibeles
 - 6.1.3. Respuesta del oído humano: umbrales de audición y sensibilidad a distintas frecuencias
 - 6.1.4. Aplicaciones tecnológicas: acústica arquitectónica, dispositivos de grabación y reproducción de audio
- 6.2. Visión y color: percepción cromática, lentes y dispositivos ópticos
 - 6.2.1. Principios ópticos del ojo humano: formación de imágenes en la retina
 - 6.2.2. Visión del color: luz, espectro visible y percepción cromática
 - 6.2.3. Lentes: convergentes y divergentes, aumento y corrección visual
 - 6.2.4. Dispositivos ópticos: cámaras, microscopios, pantallas y fibra óptica
- 6.3. Taller integrador de percepción: del audio a la imagen digital
 - 6.3.1. Analogías entre percepción sonora y visual: señales, frecuencias y resolución
 - 6.3.2. Muestreo y digitalización de señales (audio e imagen)
 - 6.3.3. Tecnologías de captura, almacenamiento y procesamiento digital
 - 6.3.4. Reflexión sobre límites humanos y ampliación tecnológica de los sentidos

Duración

06 horas

Unidad VII. Física en acción: integración y proyectos finales

Competencia

Integrar los conceptos fundamentales de la física para diseñar, desarrollar y presentar proyectos aplicados a fenómenos naturales y tecnológicos, mediante el empleo de herramientas computacionales, simulaciones y recursos de inteligencia artificial, con objetividad, creatividad y responsabilidad social en la comunicación de resultados y en la reflexión sobre la transversalidad de la ciencia.

Contenido

- 7.1. Diseño y desarrollo del proyecto aplicado
 - 7.1.1. Selección de un fenómeno físico o tecnológico de interés
 - 7.1.2. Planteamiento de objetivos, preguntas y metodología
 - 7.1.3. Uso de simulaciones, Python/Colab o recursos de IA como apoyo en el análisis
- 7.2. Presentación y discusión de proyectos finales
 - 7.2.1. Exposición oral y escrita de resultados
 - 7.2.2. Discusión crítica con base en la física y el contexto tecnológico
 - 7.2.3. Evaluación de la claridad, objetividad y honestidad en la comunicación de resultados
- 7.3. Reflexión sobre la transversalidad de la física
 - 7.3.1. Identificación de conexiones entre cinemática, dinámica, energía, ondas, campos, termodinámica y percepción
 - 7.3.2. Reconocimiento del papel de la física en el desarrollo de la ciencia de datos, la computación y la ingeniería
 - 7.3.3. Reflexión sobre la relevancia social y ética del conocimiento científico

Duración

04 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

Unidad I. El lenguaje de la física en problemas básicos

Práctica 1: Modelar e interpretar relaciones físicas: gráficas, unidades y análisis dimensional

Duración

03 horas

Procedimiento

1. Utiliza la simulación [Equation Grapher de PhET](#) para explorar cómo diferentes expresiones matemáticas (lineales, cuadráticas y cúbicas) representan relaciones entre magnitudes físicas.
2. Colabora en equipos para plantear ejemplos físicos simples (como distancia vs. tiempo o energía vs. altura) y ajustar coeficientes para interpretar su significado en términos de unidades y magnitudes.
3. Complementa la actividad con ejercicios de notación científica y análisis dimensional en un cuaderno (notebook) de *Jupyter* en [Python/Colab](#), verificando la coherencia de expresiones físicas.
4. Elabora un reporte como evidencia de realización de la práctica.
5. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Simulación PhET: [Equation Grapher](#).
- *College Physics 2e*, secciones 1.1 ([Physics: An Introduction](#)) y 1.2 ([Physical Quantities and Units](#)).
- Cuaderno interactivo *Jupyter* en [Python/Colab](#) para cálculos y visualización.
- ChatGPT u otra IA para validar interpretaciones y generar ejemplos adicionales.

Práctica 2: Estimar órdenes de magnitud en problemas tipo Fermi

Duración

03 horas

Procedimiento

1. Resuelve problemas en equipo mediante estimaciones tipo Fermi. Desarrolla razonamientos de orden de magnitud y valida supuestos.
2. En equipo plantea estimaciones iniciales para responder a preguntas abiertas (por ejemplo: ¿cuántas veces late el corazón en un año?, ¿cuántas moléculas de aire hay en un salón de clases?, ¿qué potencia en watts genera una persona al subir una escalera?).
3. Presenta los resultados en notación científica cuidando las cifras significativas.

4. Utiliza un cuaderno de *Jupyter* en [Python/Colab](#) para realizar conversiones de unidades y automatiza los resultados.
5. Contrasta sus estimaciones con búsquedas en IA (ChatGPT u otra) para discutir coincidencias, discrepancias y la importancia de comunicar con claridad la incertidumbre de los resultados.
6. Elabora un reporte como evidencia de realización de la práctica.
7. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- *College Physics 2e* de OpenStax, sección 1.3 ([Accuracy, Precision, and Significant Figures](#)).
- Cuaderno (notebook) de *Jupyter* en [Python/Colab](#) para cálculos y órdenes de magnitud.
- Herramientas de inteligencia artificial (ChatGPT u otras) para validar y contrastar estimaciones.
- Pizarra o documento compartido para el registro colectivo de resultados y discusión.

Unidad II. Movimiento y las leyes de Newton

Práctica 3: Analizar el movimiento en una dimensión

Duración

03 horas

Procedimiento

1. Utiliza la simulación [PhET: Moving Man](#) para observar el movimiento de un cuerpo en una dimensión, modificando posición, velocidad y aceleración.
2. Colabora en equipo para registrar gráficas de posición–tiempo y velocidad–tiempo, interpretarlas y relacionar sus características con el movimiento observado.
3. Resuelve ejercicios en un cuaderno (notebook) de *Jupyter* en [Python/Colab](#) para calcular velocidad media, instantánea y aceleración a partir de datos simulados.
4. Elabora en equipo un breve reporte en el que comparen la información de las gráficas con sus cálculos numéricos y discutan posibles errores de interpretación.
5. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Simulación [PhET: The Moving Man](#).
- *College Physics 2e*, Capítulo 2 ([Kinematics](#)).
- Cuaderno (notebook) de *Jupyter* en [Python/Colab](#) para cálculos básicos y gráficas.
- ChatGPT u otra IA para discutir ejemplos adicionales de movimiento unidimensional.

Práctica 4: Predecir alcance, tiempo de vuelo y altura máxima de proyectiles

Duración

03 horas

Procedimiento

1. Utiliza la simulación [PhET: Projectile Motion](#) para explorar el movimiento de proyectiles. Varía el ángulo, velocidad inicial y condiciones (con o sin resistencia del aire) para observar alcance, tiempo de vuelo y altura máxima.
2. Registrar trayectorias y las compara con predicciones cualitativas.
3. Utiliza un cuaderno (notebook) de *Jupyter* en [Python/Colab](#) para simular trayectorias simples. Resuelve las ecuaciones de movimiento en 2D y grafica la posición en función del tiempo.
6. Cierra con una puesta en común y una reflexión: cómo los vectores y las coordenadas cartesianas describen trayectorias físicas.
7. Elabora un reporte como evidencia de realización de la práctica.
8. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Simulación [PhET: Projectile Motion](#).
- *College Physics 2e*, Capítulo 3 ([Two-Dimensional Kinematics](#)).
- Cuaderno (notebook) de *Jupyter* en [Python/Colab](#) para simulación numérica y gráficas de trayectorias.
- ChatGPT u otra IA como recurso de apoyo para generar variaciones de problemas de proyectiles.

Práctica 5: Analizar fuerzas y movimiento mediante diagramas de cuerpo libre

Duración

03 horas

Procedimiento

1. Utiliza la simulación [PhET: Forces and Motion: Basics](#) para explorar los efectos de distintas fuerzas sobre el movimiento de un objeto.
2. Identifica fuerzas de contacto y a distancia, y elabora diagramas de cuerpo libre para relacionarlos con la primera, segunda y tercera leyes de Newton.
3. Usa [PhET: Forces and Motion: Basic](#) para experimentar: ajusta masa, fricción y fuerza aplicada, y registra la aceleración resultante.
4. En un cuaderno (notebook) de *Jupyter* en [Python/Colab](#), resuelve ejercicios básicos con la segunda ley de Newton ($F=ma$). Luego, compara los resultados con los obtenidos en la simulación.

9. Elabora diagramas y conclusiones por equipo, que muestren la relación entre fuerza neta y movimiento.
10. Elabora un reporte como evidencia de realización de la práctica.
11. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Simulación [PhET: Forces and Motion: Basics](#).
- *College Physics 2e*, Capítulo 4 ([Dynamics: Newton's Laws of Motion](#)).
- Cuaderno (notebook) de *Jupyter* en [Python/Colab](#) para resolución de problemas y verificación numérica.
- ChatGPT u otra IA para generar ejemplos adicionales y verificar diagramas de cuerpo libre.

Unidad III: Conservación de la energía y el momento

Práctica 6: Trabajo, energía y potencia en el movimiento

Duración

03 horas

Procedimiento

1. Utiliza la simulación [PhET: Work, Energy, and Power](#) para explorar cómo se relacionan la fuerza aplicada, el desplazamiento y el trabajo realizado sobre un objeto.
2. Observa cómo varían la energía cinética y potencial en función de la posición y el tiempo, y analizar el papel de la potencia como razón de cambio del trabajo.
3. En equipo, registra datos de la simulación y representar gráficamente las variaciones de energía.
4. En un cuaderno (notebook) de *Jupyter* en [Python/Colab](#), realiza cálculos de trabajo y potencia usando datos simulados y comprueba la consistencia dimensional y numérica.
5. Por equipo, elabora un reporte breve que incluya gráficas, cálculos y conclusiones sobre la relación trabajo–energía–potencia.
6. Elabora un reporte como evidencia de realización de la práctica.
7. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Simulación [PhET: Work, Energy, and Power](#).
- College Physics 2e, Capítulo 7 ([Work, Energy, and Power](#)).
- Cuaderno (notebook) de *Jupyter* en [Python/Colab](#) para cálculos de trabajo y representaciones gráficas.
- ChatGPT u otra IA para generar ejemplos adicionales y validar resultados.

Práctica 7: Conservación de la energía y del momento en colisiones

Duración

03 horas

Procedimiento

1. Utiliza la simulación [PhET: Collision Lab](#) para analizar colisiones entre objetos en una dimensión y dos dimensiones.
2. Varía masas y velocidades iniciales para estudiar colisiones elásticas e inelásticas, observando cómo se conserva (o no) la energía cinética y cómo siempre se conserva el momento lineal.
3. En equipo, registra datos de la simulación, calcula el momento inicial y final y compara la energía cinética antes y después de cada choque.
4. Resuelve problemas de colisiones sencillas en un cuaderno (notebook) de *Jupyter* en [Python/Colab](#) aplicando las ecuaciones de conservación del momento y de la energía, y contrasta sus resultados con lo observado en la simulación.
5. Al cierre, analiza en equipo la importancia de los principios de conservación y sus aplicaciones en física e ingeniería.
6. Elabora un reporte como evidencia de realización de la práctica.
7. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo:

- Simulación [PhET: Collision Lab](#).
- College Physics 2e, Capítulo 7 ([Conservation of Energy](#)) y Capítulo 8 ([Linear Momentum and Collisions](#)).
- Cuaderno (notebook) de *Jupyter* en [Python/Colab](#) para resolución de problemas numéricos y gráficas de comparación.
- ChatGPT u otra IA para plantear variaciones de problemas y verificar razonamientos.

Unidad IV: Ondas, campos e información: principios universales de la física

Práctica 8: Estudiar la propagación de ondas y el principio de superposición

Duración

03 horas

Procedimiento

1. Emplea la simulación [PhET: Wave on a String](#) para explorar cómo se propagan las ondas en una cuerda bajo distintas condiciones de tensión, amplitud y frecuencia.
2. En equipo, analiza el principio de superposición observando interferencias constructivas y destructivas; ajusta parámetros para producir ondas estacionarias y condiciones de resonancia.
3. En equipo, relaciona estos fenómenos con casos cotidianos, como el sonido en instrumentos musicales o la transmisión de señales.
4. En un cuaderno (notebook) de *Jupyter* en [Python/Colab](#) representa funciones seno y coseno superpuestas para modelar interferencias y visualizar cómo la matemática describe los fenómenos observados en la simulación.
5. Elabora un reporte como evidencia de realización de la práctica.
6. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Simulación [PhET: Wave on a String](#).
- College Physics 2e, Capítulo 16 ([Oscillatory Motion and Waves](#)) y Capítulo 17 ([Sound](#)).
- Cuaderno (notebook) de *Jupyter* en [Python/Colab](#) para modelado y gráficas de funciones ondulatorias.
- ChatGPT u otra IA para generar analogías y ejemplos tecnológicos de resonancia e interferencia.

Práctica 9: Explorar y representar campos eléctricos y magnéticos

Duración

03 horas

Procedimiento

1. Utiliza la simulación [PhET: Electric Field Hockey](#) para introducir el concepto de campo eléctrico y visualizar cómo interactúan cargas positivas y negativas.
2. Emplea la simulación [PhET: Magnet and Compass](#) para explorar la representación de campos magnéticos, la interacción entre imanes y brújulas, y la relación entre corrientes y campos.
3. En equipo, representa líneas de campo y analiza el campo como modelo de interacción a distancia.
4. En un cuaderno (notebook) de *Jupyter* en [Python/Colab](#) representa campos sencillos (por ejemplo, un campo eléctrico radial) para reforzar la conexión entre modelos matemáticos y representaciones visuales.
5. Al cierre, analiza aplicaciones tecnológicas como motores eléctricos, resonancia magnética y almacenamiento magnético.
6. Elabora un reporte como evidencia de realización de la práctica.

7. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Simulación [PhET: Electric Field Hockey](#) .
- Simulación [PhET: Magnet and Compass](#).
- *College Physics 2e*, Capítulo 18 (*Electric Charge and Electric Field*) y Capítulo 22 (*Magnetism*).
- Cuaderno (notebook) de *Jupyter* en [Python/Colab](#) para graficar campos eléctricos y magnéticos simples.
- ChatGPT u otra IA para contextualizar aplicaciones tecnológicas.

Unidad V: Termodinámica y el concepto de entropía en procesos físicos

Práctica 10: Analizar la transferencia de calor

Duración

03 horas

Procedimiento

1. Utiliza la simulación [PhET: Heat Transfer](#) para observar los mecanismos de conducción, convección y radiación.
2. Varía materiales, áreas de contacto y gradientes de temperatura para comparar las diferentes formas de transferencia térmica.
3. En equipo, registra los datos de las simulaciones y representa gráficamente la variación de temperatura con el tiempo.
4. En un cuaderno (notebook) de *Jupyter* en [Python/Colab](#) resuelve ejercicios sencillos para estimar tasas de transferencia de calor y discute qué condiciones favorecen la conducción, la convección o la radiación.
5. Elabora un reporte breve que conecta los fenómenos analizados con ejemplos de la vida cotidiana y con aplicaciones tecnológicas.
6. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Simulación [PhET: Heat Transfer](#).
- *College Physics 2e*, Capítulo 13 ([Temperature and Kinetic Theory](#)) y Capítulo 14 ([Heat and Heat Transfer Methods](#)).
- Cuaderno (notebook) de *Jupyter* en [Python/Colab](#) para cálculos de transferencia de calor y representaciones gráficas.
- ChatGPT u otra IA para generar analogías con fenómenos naturales y tecnológicos.

Práctica 11: Entropía y la segunda ley de la termodinámica

Duración

03 horas

Procedimiento

1. Emplea la simulación [PhET: Reversible Reactions](#) (adaptada para ilustrar reversibilidad e irreversibilidad) y experimenta con modelos virtuales de máquinas térmicas para analizar la segunda ley de la termodinámica.
2. Observa cómo la entropía aumenta en procesos irreversibles y cómo se limita la eficiencia de una máquina térmica.
3. En un cuaderno (notebook) de *Jupyter* en [Python/Colab](#) simula un motor de Carnot sencillo, calculando su eficiencia ideal y comparándola con motores reales.
4. Al cierre, analiza en equipo ejemplos de irreversibilidad en la naturaleza y la tecnología, y reflexiona sobre cómo la entropía incide en la vida diaria y en los límites del uso de la energía.
5. Elabora un reporte como evidencia de realización de la práctica.
6. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Simulación [PhET: Reversible Reactions](#) y simulaciones de máquinas térmicas.
- *College Physics 2e*, Capítulo 15 ([Thermodynamics](#)).
- Cuaderno (notebook) de *Jupyter* en [Python/Colab](#) para cálculos de eficiencia de motores térmicos.
- ChatGPT u otra IA para explorar ejemplos de irreversibilidad en procesos naturales y tecnológicos.

Unidad VI: Física de la percepción y tecnologías afines

Práctica 12: Analizar el sonido y la audición

Duración

03 horas

Procedimiento

1. Utilizar la simulación [PhET: Sound](#) y explora cómo cambian la frecuencia, la amplitud y la intensidad de las ondas sonoras.
2. En equipo, analizar la relación entre estos parámetros y la percepción del oído humano (altura, volumen y timbre).
3. Elabora gráficas de intensidad sonora en escala de decibeles y compara con los umbrales de audición humana.
4. En un cuaderno (notebook) de *Jupyter* en [Python/Colab](#) representa ondas sonoras de distintas frecuencias e intensidades y simula mezclas de señales.
5. Al cierre, reflexiona sobre aplicaciones tecnológicas como auriculares, altavoces y la acústica de espacios.
6. Elabora un reporte como evidencia de realización de la práctica.

7. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Simulación [PhET: Sound](#).
- *College Physics 2e*, Capítulo 17 ([Sound](#)).
- Cuaderno (notebook) de *Jupyter* en [Python/Colab](#) para graficar ondas y analizar mezclas de frecuencias.
- ChatGPT u otra IA para generar ejemplos y analogías de fenómenos acústicos.

Práctica 13: Analizando la visión y el color

Duración

03 horas

Procedimiento

1. Emplea la simulación [PhET: Bending Light](#) para estudiar la refracción y el comportamiento de lentes convergentes y divergentes.
2. En equipo, explora cómo se forman imágenes y cómo varían con el índice de refracción y la curvatura de las lentes.
3. Utiliza la simulación [PhET: Color Vision](#) para comprender cómo percibe el ojo humano los colores a partir de la combinación de luces roja, verde y azul.
4. En un cuaderno (notebook) de *Jupyter* en [Python/Colab](#) representa mezclas de colores en el espacio RGB y analiza sus aplicaciones en pantallas digitales.
5. Al cierre, relaciona los fenómenos con cámaras, microscopios y otros dispositivos ópticos.
6. Elabora un reporte como evidencia de realización de la práctica.
7. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Simulación [PhET: Bending Light](#).
- Simulación [PhET: Color Vision](#).
- *College Physics 2e*, Capítulo 25 ([Geometric Optics](#)) y Capítulo 27 ([Wave Optics](#)).
- Cuaderno (notebook) de *Jupyter* en [Python/Colab](#) para representar combinaciones de colores y trayectorias ópticas simples.
- ChatGPT u otra IA para relacionar conceptos con tecnologías ópticas actuales.

Práctica 14: Taller de percepción digital: del audio a la imagen

Duración

03 horas

Procedimiento

1. En equipo, elabora un cuadro comparativo entre los parámetros del sonido (frecuencia ↔ tono, amplitud ↔ volumen) y de la visión (longitud de onda ↔ color, intensidad ↔ brillo), identificando además las limitaciones fisiológicas del oído y la vista como los umbrales de percepción y los rangos audible y visible.
2. Con [TwistedWave Online](#), sube un archivo de voz o música, y posteriormente lo exporta a 44 kHz, 22 kHz y 8 kHz, y analiza cómo cambia la calidad al reducir el muestreo
3. En [Photopea](#), abre una fotografía sencilla, y reduce su resolución y profundidad de color (8 bits, 4 bits), y registra a partir de qué punto la imagen pierde detalle o se vuelve irreconocible.
4. Concluye con una reflexión integradora que contrasta la pérdida de calidad en audio e imagen, valora la resolución temporal del sonido y la espacial de las imágenes, y describe cómo cámaras, pantallas, micrófonos y fibra óptica superan las limitaciones humanas.
5. Al cierre, elabora un reporte de una cuartilla que incluye: capturas de pantalla, una tabla comparativa audio–imagen y una reflexión final sobre la ampliación tecnológica de los sentidos.
6. Anexa el reporte al portafolio digital de evidencias.

Recursos de apoyo

- Software en línea:
 - [TwistedWave Online](#) – editor y exportador de audio.
 - [Photopea](#) – editor de imágenes en línea.
- Apuntes de clase sobre muestreo, resolución y percepción humana.
- *College Physics 2e*: Capítulo 17 ([Sound](#)), Capítulo 25 ([Geometric Optics](#)) y Capítulo 27 ([Wave Optics](#)).

Unidad VII: Física en acción: integración y proyectos

Práctica 15: Física en acción: presentación y reflexión de proyectos integradores

Duración

06 horas

Procedimiento

1. En equipo presenta su proyecto integrador, explicando un fenómeno físico o aplicación tecnológica a partir de los contenidos vistos en el curso (movimiento, energía, ondas, campos, termodinámica y percepción).
2. Las presentaciones deberán incluir: (i) Una explicación conceptual clara, (ii) un elemento computacional o de simulación (PhET, Cuaderno *Jupyter* en Python/Colab, IA u otro) y (iii) una conexión con aplicaciones tecnológicas o sociales. Al término de cada exposición, realiza una breve discusión crítica con preguntas de otros equipos y del docente.
3. Al cierre, realiza una reflexión grupal sobre cómo la física se articula con las matemáticas aplicadas, la ciencia de datos, la computación y la vida diaria.
4. Redacta una síntesis breve e individual: identifica tus aprendizajes clave y valora la objetividad, la creatividad y la ética en la práctica científica.

Recursos de apoyo

- Simulaciones [PhET](#) relevantes según el tema de cada equipo.
- Cuaderno (notebook) de *Jupyter* en [Python/Colab](#) para cálculos, visualización de datos y modelado.
- Herramientas de IA (ChatGPT u otras) como apoyo en la generación de ejemplos o validación de explicaciones.
- [College Physics 2e](#) como referencia conceptual transversal.
- Recursos audiovisuales (presentaciones, pósters digitales o videos breves).

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Aprendizaje basado en preguntas
- Técnica expositiva
- Debates
- Discusión guiada
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Mapas mentales
- Mapas conceptuales
- Cuadros comparativos
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales:	30 %
– Tareas núcleo:	25 %
– Reportes de prácticas de taller:	10 %
– Portafolio de evidencias	10 %
– Proyecto integrador:	25 %
<i>Total</i>	100 %

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Cutnell, J. D., Johnson, K. W., & Young, D. F. (2021). Physics (12th ed.). Wiley.

Giancoli, D. C. (2006). Física: Principios con aplicaciones (6.^a ed.). Pearson Educación. [clásica].

Giancoli, D. C. (2013). Physics: Principles with applications (7th ed.). Pearson.

Halliday, D., Resnick, R., & Walker, J. (2009). Fundamentos de física (J. H. Romo, trad.; 4.^a ed.; Vols. 1–2). Grupo Editorial Patria. [clásica].

Halliday, D., Resnick, R., & Walker, J. (2021). Fundamentals of physics (12th ed.). Wiley. [clásica]

Hewitt, P. G. (2016). Física conceptual (12.^a ed.). Pearson Educación. [clásica]

Hewitt, P. G. (2021). Conceptual physics (13th ed.). Pearson. [clásica]

Knight, R. D., Jones, B., & Field, S. (2019). College physics: A strategic approach (4th ed.). Pearson.

Knight, R. D., Jones, B., & Field, S. (2024). Física: Un enfoque estratégico (4.^a ed.). Pearson Educación.

Serway, R. A., & Vuille, C. (2019). Física (10.^a ed.). Cengage Learning. [clásica]

Serway, R. A., Vuille, C., & Hughes, J. (2020). College physics (12th ed.). Cengage. [clásica]

Tippens, P. E. (2012). Física: Conceptos y aplicaciones (7.^a ed.). McGraw-Hill. [clásica]

Young, H. D., & Freedman, R. A. (2018). Física universitaria con física moderna (14.^a ed., Vol. 1). Pearson. [clásica]

Recursos de acceso libre o complementarios externos

OpenStax. (2022). College Physics (2^a ed). OpenStax. (Inglés; gratuito. En español, OpenStax ofrece “Física universitaria”) <https://openstax.org/books/college-physics-2e>.

Recursos digitales: [GeoGebra](#), [Tracker](#), [Python/Matplotlib](#), [PhET Simulations](#).

XII. PERFIL DOCENTE

Licenciatura en Física, Ingeniería Física, preferentemente con posgrado en área afín; dominar los contenidos fundamentales de la mecánica, energía, ondas, campos, termodinámica y óptica, así como su vinculación con fenómenos naturales y tecnológicos; poseer experiencia docente en educación superior mínima de dos años, y manejo de estrategias didácticas activas; tener competencia en el uso de simuladores interactivos (PhET), herramientas computacionales (Python/Colab) y recursos de inteligencia artificial aplicados a la enseñanza; y demostrar compromiso con la formación integral del estudiantado, promoviendo la objetividad, el rigor científico, la creatividad y la responsabilidad social en el ejercicio de la física.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales, Licenciatura en Física, Licenciatura en Matemáticas Aplicadas, Licenciatura en Ciencia de Datos
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Cálculo Diferencial
Clave:	
Distribución horaria:	Horas Clase: 03 Horas Taller: 03 Horas Laboratorio: 00 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 03 Créditos:09
Etapas de formación:	Básica
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Julio Román Durán Camacho Eloísa del Carmen García Canseco Adina Jordan Arámburo Claudia Lara Silva Juan Crisóstomo Tapia Mercado Jorge Alberto Villavicencio Aguilar
Subdirección que aprueba el PUA:	
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

El propósito del Cálculo Diferencial es que las y los estudiantes del tronco común de Física, Matemáticas Aplicadas, Ciencias Computacionales y Ciencia de Datos desarrollen una comprensión inicial, intuitiva, lógica y analítica de las funciones de una variable real y de su variación, mediante el estudio de los conceptos de límite, continuidad y derivada. Para ello, se articula el trabajo con el lenguaje algebraico, las representaciones numéricas y gráficas, la resolución de problemas y el uso pertinente de recursos tecnológicos. Esta unidad de aprendizaje está ubicada en la etapa básica, como parte del Tronco Común de Ciencias Exactas, siendo de carácter obligatorio para todas las Licenciaturas que comparten el Tronco Común.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Analizar las relaciones de cambio y dependencia entre variables a partir del estudio intuitivo y formal de funciones, límites, continuidad y derivadas para modelar, interpretar y resolver problemas propios de las ciencias naturales y exactas, con un enfoque riguroso, responsabilidad y honestidad.

IV. EVIDENCIA DE APRENDIZAJE

- Resolución y documentación de situaciones problemáticas de cálculo diferencial en las que se integren los conceptos de función, límite, continuidad y derivada, mediante el uso articulado de representaciones algebraicas, numéricas y gráficas, así como de herramientas tecnológicas, para analizar el comportamiento de funciones, justificar procedimientos y formular conclusiones con rigor conceptual y claridad argumentativa.
- Elaboración de un portafolio de evidencias que incluya ejercicios resueltos, reportes de prácticas y una actividad integradora, en los que se evidencie la construcción progresiva del concepto de variación, desde el análisis de funciones hasta la aplicación de la derivada en problemas de modelación, optimización e interpretación.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Licenciatura en Física: Explorar fenómenos físicos mediante sus modelos matemáticos, para describir y predecir el comportamiento de la materia, la energía y sus interacciones, con disciplina y rigor científico.

Licenciatura en Ciencias Computacionales: Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

Licenciatura en Ciencia de Datos: Modelar y optimizar funciones matemáticas aplicadas a conjuntos de datos, utilizando los principios del cálculo diferencial para desarrollar modelos

predictivos y descriptivos que permitan la identificación de tendencias y la toma de decisiones basada en evidencia, contribuyendo a la generación de conocimiento y soluciones inteligentes ante problemas complejos, con rigor analítico, ética en el manejo de la información y compromiso social.

Licenciatura en Matemáticas Aplicadas: Evaluar los fundamentos teóricos de la geometría, el álgebra, el análisis y la lógica matemática mediante el planteamiento de definiciones y axiomas, y la construcción de propiedades para la formulación y resolución de problemas en contextos científicos y tecnológicos, de manera colaborativa y con actitud crítica.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia

VII. DESARROLLO POR UNIDADES

Unidad I. Relaciones y Funciones

Competencia

Analizar relaciones y funciones de una variable real, mediante la representación algebraica, tabulación y gráfica de sus correspondencias, con el propósito de reconocer sus propiedades, clasificaciones y operaciones, de manera reflexiva, ordenada y argumentada.

Contenido

- 1.1. Producto Cartesiano
 - 1.1.1. Caso: $\mathbb{R} \times \mathbb{R}$
- 1.2. Relación
 - 1.2.1. Definición
 - 1.2.2. Representación gráfica y ejemplos de relaciones
- 1.3. Función
 - 1.3.1. Pares ordenados
 - 1.3.2. Regla de correspondencia
 - 1.3.3. Representación gráfica
- 1.4. Operaciones con funciones
 - 1.4.1. Suma, resta, multiplicación y división
 - 1.4.2. Composición
- 1.5. Familias de funciones elementales
 - 1.5.1. Algebraicas
 - 1.5.2. Exponenciales
 - 1.5.3. Logarítmicas
 - 1.5.4. Trigonométricas
 - 1.5.4.1. Construcción a partir de la circunferencia unitaria
 - 1.5.4.2. Periodo

Duración

12 horas

Unidad II. Límites**Competencia**

Analizar el concepto de límite de una función a partir del estudio de su comportamiento local y global, mediante aproximaciones numéricas, representaciones gráficas y procedimientos algebraicos, para construir una comprensión formal que permita calcular límites y formular conclusiones justificadas sobre el comportamiento de las funciones, con rigor lógico, actitud crítica y argumentación matemática.

Contenido

- 2.1. Noción intuitiva
 - 2.1.1. Tablas de valores
 - 2.1.2. Límites laterales
- 2.2. Análisis geométrico
 - 2.2.1. Definición formal $\varepsilon - \delta$: Caso lineal
- 2.3. Propiedades de los límites
- 2.4. Límites especiales
 - 2.4.1. Límites en el infinito y comportamiento asintótico básico
 - 2.4.2. Límites infinitos
 - 2.4.3. Límites trigonométricos

Duración

10 horas

Unidad III. Continuidad**Competencia**

Determinar la continuidad de funciones reales mediante la comprensión intuitiva y formal del concepto, el uso de definiciones, así como el razonamiento algebraico y el análisis de representaciones gráficas, para describir e interpretar el comportamiento de funciones y modelos en puntos e intervalos de interés, identificar y clasificar sus discontinuidades con precisión, reflexión y rigor lógico.

Contenido

- 3.1. Noción de continuidad
 - 3.1.1. Continuidad en un punto y en un intervalo
 - 3.1.2. Relación entre límite, valor de la función y continuidad
- 3.2. Tipos de discontinuidad
 - 3.2.1. Discontinuidad removible, de salto e infinita
 - 3.2.2. Continuidad en funciones definidas por partes
- 3.3. Interpretación en contextos elementales

3.3.1. Interpretación del teorema del valor intermedio en contextos elementales

Duración

06 horas

Unidad IV. Derivadas

Competencia

Construir la derivada de una función a partir de su definición como límite, mediante el análisis de cambios, el uso de definiciones formales y el empleo de representaciones gráficas, numéricas y algebraicas, para describir la variación de una función en un punto y su comportamiento local, interpretándose como razón de cambio instantánea y como la pendiente de la recta tangente, con disciplina, claridad y pensamiento crítico.

Contenido

- 4.1. Definición de derivada como límite
- 4.2. Interpretación
 - 4.2.1. La derivada como razón de cambio instantánea
 - 4.2.2. La derivada como pendiente de la recta tangente
- 4.3. Reglas de derivación
 - 4.3.1. Potencias
 - 4.3.2. Sumas y restas
 - 4.3.3. Multiplicación
 - 4.3.4. División
 - 4.3.5. Composición
 - 4.3.5.1. Regla de la cadena
- 4.4. Técnicas de derivación
 - 4.4.1. Derivación implícita
 - 4.4.1.1. Derivada de una función inversa
 - 4.4.2. Derivación de orden superior
- 4.5. Regla de L'Hopital
- 4.6. Diferencial
 - 4.6.1. Interpretación geométrica
 - 4.6.2. Aproximación lineal

Duración

10 horas

Unidad V. Interpretación de la derivada

Competencia

Aplicar la derivada al análisis del comportamiento de funciones y a la resolución de problemas introductorios sobre tasas de cambio y optimización, mediante argumentos algebraicos, gráficos y tecnológicos, cuando sean pertinentes, para interpretar resultados y tomar decisiones

justificadas en contextos elementales de ciencias y modelación, con objetividad, creatividad y responsabilidad.

Contenido

- 5.1. Valores mínimos y máximos de funciones
 - 5.1.1. Definición de máximo y mínimo absoluto y relativo
 - 5.1.2. Interpretación geométrica y analítica
 - 5.1.3. Puntos críticos y su clasificación
 - 5.1.4. Existencia de extremos (noción en intervalos cerrados)
- 5.2. Teorema de Rolle y del Valor Medio
 - 5.2.1. Hipótesis y conclusiones del Teorema de Rolle
 - 5.2.2. Interpretación geométrica
 - 5.2.3. Teorema del Valor Medio (Lagrange)
 - 5.2.5. Interpretación y aplicaciones
- 5.3. Criterios de la primera y segunda derivada para el análisis de funciones
 - 5.3.1. Intervalos de crecimiento y decrecimiento
 - 5.3.2. Relación entre el signo de la primera derivada y el comportamiento de la función
 - 5.3.3. Identificación de extremos relativos mediante la primera derivada
 - 5.3.4. Concavidad y su relación con la segunda derivada
 - 5.3.5. Determinación de puntos de inflexión
 - 5.3.6. Uso de la segunda derivada para la clasificación de extremos relativos
- 5.4. Trazo de gráficas de funciones
 - 5.4.1. Análisis del dominio y comportamiento general
 - 5.4.2. Intersecciones con los ejes
 - 5.4.3. Asíntotas (verticales, horizontales y oblicuas)
 - 5.4.4. Intervalos de crecimiento y decrecimiento
 - 5.4.5. Concavidad y puntos de inflexión
 - 5.4.6. Integración de la información para el esbozo de la gráfica
 - 5.4.7. Uso de herramientas tecnológicas para verificación
- 5.5. Problemas de optimización
 - 5.6.1. Planteamiento del problema (modelación)
 - 5.6.2. Identificación de variables y restricciones
 - 5.6.3. Construcción de la función objetivo
 - 5.6.4. Determinación de extremos mediante derivadas
 - 5.6.5. Interpretación de resultados en el contexto

Duración

10 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

Unidad I. Relaciones y funciones

Práctica 1: Distinguir relaciones y funciones en distintas representaciones

Duración

04 horas

Procedimiento

1. Analiza, con base en la explicación docente, la diferencia entre relación y función, considerando definiciones, ejemplos y contraejemplos.
2. Analiza distintos casos de relaciones dadas en forma de tablas, expresiones algebraicas y gráficas, identificando cuáles corresponden a funciones y justificando su respuesta.
3. Resuelve de manera individual una serie de ejercicios en los que determines si una relación es función, utilizando criterios como la unicidad de la imagen y la prueba de la recta vertical.
4. Con apoyo de herramientas tecnológicas para la enseñanza de las matemáticas (Desmos, Geogebra, Colab, etc.), genera uno o dos ejemplos adicionales de relaciones (en forma algebraica o gráfica) y determina si son funciones, explicando su razonamiento.
5. Participa en la discusión grupal sobre las estrategias utilizadas para reconocer funciones, comparando procedimientos y criterios de validación.
6. Elabora una reflexión personal en la que describa lo aprendido durante la actividad, destacando cómo la comprensión de los conceptos de relación y función contribuye al análisis de problemas matemáticos.
7. Integra los ejercicios resueltos, los ejemplos generados y la reflexión en su portafolio de evidencias.

Recursos de apoyo

- Lápiz o bolígrafo
- Papel
- Acceso a herramientas tecnológicas

Práctica 2: Analizar dominio, rango y comportamiento de funciones elementales

Duración

04 horas

Procedimiento

1. Revisa en sus notas las características generales de funciones lineales, cuadráticas, racionales sencillas, radicales, exponenciales, logarítmicas y trigonométricas básicas.
2. Resuelve actividades en las que identifica dominio, rango, interceptos, simetrías y comportamiento global de funciones elementales.

3. Relaciona la expresión algebraica con su tabla de valores y con una representación gráfica aproximada.
4. Compara en equipo distintas familias de funciones y explica qué rasgos permiten diferenciarlas.
5. Elabora un reporte breve con los resultados y las conclusiones de la práctica.
6. Entrega el reporte para retroalimentación y anexa al portafolio de evidencias.

Recursos de apoyo

- Lápiz o bolígrafo
- Papel
- Notas de clase
- Calculadora básica
- <https://openstax.org>

Práctica 3: Construir funciones mediante operaciones, composición e inversa

Duración

04 horas

Procedimiento

1. Atiende las instrucciones de su docente.
2. Revisa en tus notas las operaciones con funciones, la composición y la noción operativa de función inversa.
3. Resuelve ejercicios en los que formes funciones nuevas a partir de otras (suma, producto, cociente y composición), y analiza cómo cambian el dominio y el comportamiento de la salida.
4. Interpreta composiciones sencillas a partir de situaciones descritas verbalmente o mediante tablas, explicando el significado del proceso compuesto.
5. Examina casos elementales de función inversa y justifica, con argumentos algebraicos o gráficos, cuándo una función puede invertirse en un intervalo dado.
6. Con acceso a una herramienta de inteligencia artificial generativa, solicita la generación de uno o dos ejercicios adicionales sobre operaciones o composición de funciones, y resuélvelos críticamente, verificando la validez de los resultados.
7. Elabora un reporte breve que incluya los ejemplos trabajados (propios y generados con IA), así como comentarios sobre los cambios que producen las operaciones entre funciones.
8. Entrega el reporte para retroalimentación y lo anexa a su portafolio de evidencias.

Recursos de apoyo

- Lápiz o bolígrafo
- Papel
- OpenStax o material equivalente
- Acceso a herramienta de IA generativa

Unidad II. Límites

Práctica 4: Estimar límites mediante tablas y gráficas

Duración

05 horas

Procedimiento

1. Revisa en sus notas la noción intuitiva de límite y la diferencia entre valor de la función y comportamiento cercano a un punto.
2. Construye tablas de valores y aproxima comportamientos laterales en funciones seleccionadas por la persona docente.
3. Construye e Interpreta gráficas para estimar límites, reconocer cuándo no existe el límite y explicar qué evidencia respalda la conclusión propuesta.
4. Compara resultados obtenidos por aproximación numérica y por inspección gráfica.
5. Utiliza Desmos, GeoGebra o *PhET Calculus Grapher* para explorar las funciones estudiadas.
6. Contrasta la evidencia tecnológica con los resultados obtenidos previamente en el taller.
7. Elabora un reporte breve con tablas, gráficas, interpretaciones, capturas, comparaciones y las conclusiones obtenidas.
8. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Desmos, GeoGebra o PhET Calculus Grapher
- Guía de ejercicios
- Calculadora básica
- Papel milimétrico o cuaderno cuadriculado
- OpenStax o material equivalente

Práctica 5: Resolver límites con apoyo de procedimientos algebraicos elementales

Duración

05 horas

Procedimiento

1. Revisa en sus notas las propiedades operativas de los límites y los procedimientos básicos de sustitución, factorización y racionalización.
2. Resuelve límites seleccionando el procedimiento más pertinente en cada caso.
3. Justifica por qué la estrategia utilizada permite obtener una conclusión válida sobre el comportamiento de la función.
4. Utiliza Desmos, GeoGebra o *PhET Calculus Grapher* para explorar las funciones estudiadas en cada límite.
5. Contrasta la evidencia tecnológica con la explicación algebraica elaborada previamente en el taller.
6. Identifica en equipo errores frecuentes y discute cómo evitarlos.

7. Elabora un reporte breve con el desarrollo de los ejercicios, capturas, comparaciones y las conclusiones obtenidas.
8. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Desmos, GeoGebra o PhET Calculus Grapher
- Guía de ejercicios
- Cuaderno y calculadora básica
- Material de apoyo docente
- OpenStax o material equivalente para refuerzo de técnicas de límite
-

Unidad III: Continuidad

Práctica 6: Determinar continuidad y clasificar discontinuidades

Duración

03 horas

Procedimiento

1. Revisa en sus notas la relación entre límite, valor de la función y continuidad.
2. Analiza funciones en puntos particulares e intervalos para decidir si son continuas o discontinuas, apoyándote en evidencia algebraica y gráfica.
3. Clasifica las discontinuidades (removible, de salto e infinita) y justifica tu clasificación.
4. Utiliza GeoGebra o PhET Calculus Grapher para representar funciones definidas por partes y analizar su comportamiento local.
5. Modifica parámetros sencillos en las funciones y observa cómo cambian las condiciones de continuidad en puntos específicos.
6. Contrasta las observaciones gráficas con justificaciones algebraicas breves, identificando la relación entre el comportamiento visual y el valor del límite.
7. Discute en equipo la diferencia entre una falla en el valor de la función y una falla en el comportamiento del límite.
8. Registra evidencias de tu trabajo: capturas de pantalla, observaciones y conclusiones sobre continuidad y tipos de discontinuidad
9. Elabora un reporte breve que integre los casos analizados, las justificaciones y las conclusiones obtenidas.
10. Entrega el reporte para retroalimentación y lo anexa al portafolio digital de evidencias.

Recursos de Apoyo

- Guía de ejercicios
- Cuaderno y calculadora básica
- Material de apoyo docente
- GeoGebra o PhET Calculus Grapher
- OpenStax o material equivalente

Práctica 7: Verificar formalmente la continuidad de funciones

Duración

03 horas

Procedimiento

1. Revisa en sus notas la definición formal de continuidad en un punto, identificando sus tres condiciones.
2. Analiza una serie de funciones (polinomiales, racionales y definidas por partes) en puntos específicos, y verifica cada una de las condiciones de continuidad, determina si la función es continua en ese punto.
3. Aplica el análisis de límites laterales en funciones definidas por partes para decidir la existencia del límite.
4. Justifica formalmente sus conclusiones, indicando explícitamente qué condición se cumple o falla en cada caso.
5. Contrasta tus resultados con representaciones gráficas (a mano o con apoyo tecnológico), identificando la correspondencia entre el análisis formal y el comportamiento visual.
6. Discute en equipo:
 - a. diferencias entre verificación intuitiva y formal,
 - b. ventajas de usar la definición formal en casos no evidentes.
7. Elabora un reporte breve que incluya: desarrollo de los ejercicios, verificación paso a paso, y conclusiones sobre el papel de la definición formal.
8. Entrega el reporte para retroalimentación y lo anexa al portafolio digital de evidencias.

Recursos de apoyo

- Lápiz o bolígrafo
- Papel
- Notas de clase
- Calculadora
- Software de graficación o herramienta de IA

Unidad IV: Derivada

Práctica 8: Interpretar la derivada como pendiente, razón de cambio y aproximación local

Duración

03 horas

Procedimiento

1. Revisa en sus notas la diferencia entre razón de cambio promedio y razón de cambio instantánea.
2. Analiza tablas, gráficas y situaciones contextualizadas para interpretar el cambio de una magnitud respecto de otra.

3. Describe el tránsito de una recta secante a una recta tangente y explica su significado geométrico.
4. Resuelve ejercicios introductorios en los que interpreta la derivada a partir de información gráfica y numérica, y utiliza esa interpretación para estimar cambios locales sencillos.
5. Elabora un reporte breve con el procedimiento seguido y las conclusiones de la práctica.
6. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Guía de actividades
- Cuaderno, lápiz y calculadora básica
- Papel milimétrico o cuaderno cuadriculado
- Material de apoyo docente con ejemplos de tangente y aproximación local

Práctica 9: Relacionar la derivación simbólica con la interpretación geométrica

Duración

03 horas

Procedimiento

1. Revisa en sus notas las reglas básicas de derivación y su aplicación a funciones elementales.
2. Resuelve derivadas de funciones seleccionadas y analiza el significado del signo de la derivada en el comportamiento de la gráfica.
3. Utiliza *Desmos*, *GeoGebra* o *PhET Calculus Grapher* para contrastar la función original, la recta tangente y la gráfica asociada a la derivada.
4. Compara el resultado algebraico con la evidencia gráfica y explica coincidencias o discrepancias.
5. Elabora un reporte breve con ejercicios, capturas y conclusiones como evidencia de realización de la práctica.
6. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Desmos, GeoGebra o PhET Calculus Grapher.
- Guía de ejercicios.
- Calculadora básica.

Práctica 10: Resolver derivadas de distintas funciones con reglas y métodos específicos

Duración

04 horas

Procedimiento

1. Revisa en sus notas fórmulas de derivación y reglas para la suma, el producto, el cociente, la regla de la cadena, la derivación implícita, la derivada de orden superior y la regla de L'Hôpital.

2. Resuelve derivadas de funciones como: algebraicas, racionales, radicales, trascendentes (exponenciales, logaritmos, etc.), trigonométricas (seno, coseno, tangente), entre otras.
3. Utiliza herramientas tecnológicas con sistemas de álgebra computacional (CAS), como GeoGebra y Wolfram Alpha, así como inteligencia artificial generativa para algunos ejercicios específicos propuestos por el docente.
4. Elabora un reporte breve con ejercicios resueltos, capturas de pantalla y conclusiones como evidencia de la realización de la práctica.
5. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- GeoGebra, Wolfram Alpha
- Guía de ejercicios
- Formulario de derivadas

Unidad V: Aplicaciones de la Derivada

Práctica 11: Analizar crecimiento, concavidad y extremos de funciones

Duración

03 horas

Procedimiento

1. Revisa en sus notas el uso de la derivada para analizar el crecimiento, el decrecimiento, la concavidad y los extremos relativos.
2. Resuelve ejercicios de análisis de curvas a partir de funciones elementales y de sus derivadas.
3. Identifica puntos críticos, intervalos de crecimiento y de decrecimiento, cambios de concavidad, así como máximos y mínimos en contextos sencillos.
4. Elabora tablas de signos y bosquejos cualitativos de las curvas para sintetizar el comportamiento global de cada función.
5. Interpreta los resultados obtenidos y los comunica con un lenguaje matemático claro.
6. Elabora un breve reporte sobre el desarrollo de los problemas y las conclusiones de la práctica.
7. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Guía de problemas aplicados
- Cuaderno y calculadora básica
- Material de apoyo docente sobre el análisis de curvas
- OpenStax, Khan Academy o material equivalente como apoyo para la consolidación

Práctica 13: Aplicación del Teorema del Valor Medio en cinemática: El "Dictamen Pericial"

Duración

04 horas

Procedimiento

1. Revisa en sus notas las hipótesis y conclusiones del Teorema de Rolle y del Teorema del Valor Medio (Lagrange), asegurando la comprensión de las condiciones de continuidad y derivabilidad.
2. Analiza un conjunto de datos de telemetría (tiempo y posición) de un vehículo que transita entre dos puntos de control de una autopista con un límite de velocidad establecido. Los datos de telemetría serán proporcionados por el docente.
3. Determina la velocidad promedio del trayecto utilizando la ecuación de la pendiente de la recta secante.
4. Aplica el Teorema del Valor Medio para demostrar que existió al menos un instante c en el intervalo (a, b) en el que la velocidad instantánea $f'(c)$ fue igual a la velocidad media calculada.
5. Utiliza herramientas tecnológicas, como GeoGebra, para graficar la función de posición, trazar la recta secante entre los puntos de control y visualizar el punto en el que la tangente es paralela a dicha secante.
6. Participa en una discusión grupal para determinar si la evidencia matemática es suficiente para emitir una sanción legal, considerando conceptos como el margen de error y la "certeza absoluta" frente a la "evidencia indirecta".
7. Elabora un "dictamen pericial" formal que incluya: un glosario técnico (continuidad, derivabilidad, velocidad media vs. velocidad instantánea), la demostración formal del teorema aplicada al caso de estudio, la conclusión sobre la infracción y una reflexión personal sobre la validez legal de las pruebas matemáticas.
8. Entrega el dictamen para su retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Guía de datos de telemetría y de límites de velocidad
- GeoGebra o calculadora gráfica
- Plantilla de "Dictamen Pericial" (formato institucional)
- Fragmentos de reglamentos de tránsito reales para el debate

Práctica 14: Modelar y resolver problemas introductorios de optimización con apoyo de herramientas tecnológicas para el aprendizaje de las matemáticas.

Duración

03 horas

Procedimiento

1. Revisa en sus notas la estrategia general para plantear y analizar problemas introductorios de optimización.
2. Modela una situación aplicada propuesta por la persona docente, identifica la variable de interés y construye la función que se analizará.
3. Utiliza herramientas tecnológicas como Desmos, GeoGebra, Excel o Google Colab para visualizar el comportamiento de la función y contrastar la interpretación algebraica con la representación gráfica.

4. Determina el valor óptimo y explica su significado en el contexto del problema.
5. Opcionalmente, contrasta la redacción de su explicación con una herramienta de inteligencia artificial autorizada, sólo para revisar la claridad y detectar omisiones en la argumentación.
6. Elabora un reporte breve que incluya el planteamiento, el análisis, la verificación tecnológica y las conclusiones, como evidencia de la realización de la práctica.
7. Entrega el reporte para retroalimentación y anexa al portafolio digital de evidencias.

Recursos de apoyo

- Desmos, GeoGebra, Excel, Google Sheets o Google Colab.
- Computadora o tableta.
- Opcionalmente, herramienta de inteligencia artificial autorizada institucionalmente.
- Guía de modelación.

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer la forma de trabajo, los criterios de evaluación, la calidad de los trabajos académicos y los derechos y obligaciones entre docente y estudiante.

Estrategia de enseñanza

- Aprendizaje basado en problemas
- Aprendizaje basado en preguntas
- Técnica expositiva
- Discusión guiada
- Instrucción guiada

Estrategia de aprendizaje

- Resolución de problemas
- Aprendizaje basado en preguntas
- Trabajo colaborativo
- Discusión guiada
- Argumentación de procedimientos y resultados
- Investigación documental.

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	40%
– Evaluación sumativa	30%
– Portafolio de ejercicios resueltos, reportes de prácticas y una actividad integradora	20%
– Participación en clase	10%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Apostol, T. M. (1972). *Calculus (Vol. 1)*. Reverté.

Ayres, F., Jr., & Mendelson, E. (2000). *Cálculo diferencial e integral*. McGraw-Hill.

Burden, R. L., Burden Annette M., Faires, J. D.(2017). *Análisis numérico*. Cengage Learning. <https://libcon.rec.uabc.mx:2258/?il=4202>

Larson, R., Edwards, B.(2023). *Cálculo de varias variables*. Cengage. <https://libcon.rec.uabc.mx:2258/?il=36331>

Leithold, L. (2001). *El cálculo con geometría analítica*. Harla.

Purcell, E. J., Varberg, D., & Rigdon, S. E. (2007). *Cálculo diferencial e integral*. Pearson Educación.

Stewart, J., Clegg, D., Watson, S.(2021). *Cálculo de una variable. Trascendentes tempranas*. Cengage. <https://libcon.rec.uabc.mx:2258/?il=21473>

Swokowski, E. W. (1989). *Cálculo con geometría analítica*. Grupo Editorial Iberoamérica.

Thomas, G. B., Weir, M. D., & Hass, J. (2010). *Cálculo*. Pearson.

Zill, D. G., & Dewar, J. M. (2012). *Precálculo con avances de cálculo*. McGraw-Hill Interamericana.

Vera Lázaro, A. (2013). *Cálculo diferencial con MATLAB*. Macro.

Recursos de acceso libre o complementarios externos

Boelkins, M. (s. f.). *Active Calculus*. <https://activecalculus.org/>

Desmos Studio. (s. f.). *Desmos*. <https://www.desmos.com/>

Renault, M. S. (s. f.). *GeoGebra Calculus Applets*. <https://www.nku.edu/~longa/classes/mat128/days/webpace.ship.edu/msrenault/GeoGebraCalculus/GeoGebraCalculusApplets.html>

Short, T. (s. f.). *Active Calculus Preview Activities on Desmos*. <https://sites.google.com/view/taylorshort/preview-activities>

Wolfram Research. (s. f.). *Wolfram Alpha*. <https://www.wolframalpha.com/>

XII. PERFIL DOCENTE

Licenciatura en Matemáticas, Matemáticas aplicadas o área afín; preferentemente con estudios de posgrado y dos años de experiencia de práctica docente, con amplio dominio de los contenidos temáticos y con habilidad de explicar mediante comunicación efectiva conceptos matemáticos, mostrando una actitud responsable y de ética profesional.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales, Licenciatura en Física, Licenciatura en Matemáticas Aplicadas, Licenciatura en Ciencia de Datos
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Matemáticas Superiores
Clave:	
Distribución horaria:	Horas Clase: 03 Horas Taller: 03 Horas Laboratorio: 00 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 03 Créditos:09
Etapas de formación:	Básica
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Brenda Leticia De La Rosa Navarro Carlos Yee Romero Jonathan Verdugo Olachea José Ariel Camacho Gutiérrez
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La unidad de aprendizaje de Matemáticas Superiores tiene como propósito introducir al estudiantado al pensamiento formal lógico-algebraico y brindarle las competencias necesarias para manipular las propiedades fundamentales de la lógica proposicional, los conjuntos, las relaciones, las funciones y las estructuras algebraicas de los números reales, números complejos y polinomios. Se pretende así que el estudiantado cuente con bases sólidas que le permitan aplicar el pensamiento algebraico de forma rigurosa y coherente en la resolución de problemas de sus programas educativos.

Esta unidad de aprendizaje está ubicada en la etapa básica, como parte del Tronco Común de Ciencias Exactas, siendo de carácter obligatorio para todas las Licenciaturas que comparten el Tronco Común.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Analizar los fundamentos del razonamiento lógico y de las estructuras algebraicas, mediante el uso del lenguaje formal, las propiedades y los axiomas que rigen dichas estructuras, con el propósito de resolver problemas, establecer conexiones entre distintos sistemas matemáticos y justificar resultados con coherencia, precisión y rigor.

IV. EVIDENCIA DE APRENDIZAJE

Portafolio de evidencias que contenga las prácticas de talleres, los exámenes parciales y las tareas resueltas con su respectiva retroalimentación.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Matemáticas: Evaluar los fundamentos teóricos de la geometría, el álgebra, el análisis y la lógica matemática mediante el planteamiento de definiciones y axiomas, y la construcción de propiedades para la formulación y resolución de problemas en contextos científicos y tecnológicos, de manera colaborativa y con actitud crítica.

Física: Explorar fenómenos físicos mediante sus modelos matemáticos, para describir y predecir el comportamiento de la materia, la energía y sus interacciones, con disciplina y rigor científico.

Ciencias Computacionales: Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

Ciencias de Datos: Manejar los conocimientos básicos de las matemáticas a través de la geometría, el álgebra, el análisis y la lógica matemática, para aportar al desarrollo de la ciencia y la tecnología de manera colaborativa y con actitud crítica.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia

VII. DESARROLLO POR UNIDADES

Unidad I. Lógica proposicional

Competencia

Identificar las proposiciones y operadores lógicos, mediante el uso del razonamiento formal y la interpretación de argumentos lógicos, para reconocer la estructura y validez de los razonamientos matemáticos, con actitud de rigor y pensamiento crítico.

Contenido

- 1.1.Proposiciones y operadores lógicos: OR, AND, NOT, THEN
- 1.2.Tablas de verdad y equivalencias lógicas
- 1.3.Métodos de demostración. Demostraciones directas, por contradicción y por contrapositiva
 - 1.3.1. Argumentos lógicos, inferencias y deducciones

Duración

09 horas

Unidad II. Conjuntos

Competencia

Aplicar los fundamentos de la teoría de conjuntos mediante el uso de notaciones formales, operaciones y demostraciones de propiedades, para resolver problemas y clasificar estructuras conjuntistas en distintos contextos matemáticos, con actitud de rigor lógico y pensamiento crítico.

Contenido

- 2.1.Definición de conjunto y notaciones formales
- 2.2.Pertenencia, conjunto vacío, subconjuntos y conjunto potencia
- 2.3.Operaciones con conjuntos: unión, intersección, complemento, producto directo.
- 2.4.Diagramas de Venn
- 2.5.Propiedades de las operaciones de Conjuntos
 - 2.5.1. Conmutatividad, asociatividad, distributividad
 - 2.5.2. Leyes de DeMorgan
- 2.6.Cardinalidad de conjuntos finitos

2.6.1. Principio de inclusión-exclusión

Duración

09 horas

Unidad III. Relaciones y funciones

Competencia

Examinar los tipos de relaciones y el concepto de función, mediante el uso de propiedades de conjuntos, para reconocer y clasificar funciones en diversos contextos matemáticos, con actitud de rigor lógico y pensamiento crítico.

Contenido

- 3.1. Definición de relación
- 3.2. Propiedades y tipos elementales de relaciones
- 3.3. Definición de función
- 3.4. Dominio, codominio, imagen y preimagen
- 3.5. Operaciones de conjuntos bajo funciones
- 3.6. Funciones sobreyectivas, inyectivas y biyectivas
- 3.7. Composición de funciones

Duración

12 horas

Unidad IV. Estructura algebraica de los números reales y complejos

Competencia

Examinar la estructura algebraica de los números reales y complejos, mediante la aplicación de los axiomas de campo, para establecer conexiones en los sistemas numéricos de los reales y los complejos, con pensamiento crítico y responsabilidad.

Contenido

- 4.1. Axiomas de campo.
- 4.2. Ley de cancelación de la suma y ley de cancelación del producto.
- 4.3. Leyes de los exponentes y radicales.
- 4.4. Propiedades elementales de los números reales.
 - 4.4.1. Propiedades elementales de la suma
 - 4.4.2. Propiedades elementales de la multiplicación
- 4.5. Relación de orden en los números reales.
 - 4.5.1. Axiomas de orden.
 - 4.5.2. Definición de las notaciones $<$, $>$, $\geq$, $\leq$.
 - 4.5.3. Propiedades elementales de orden de los números reales.
- 4.6. Inducción matemática.
- 4.7. Algoritmo de la división para enteros.

- 4.7.1. Máximo común divisor.
- 4.8. Teorema Fundamental de la Aritmética.
 - 4.8.1. Mínimo común múltiplo.
- 4.9. Números Complejos
 - 4.9.1. Los números complejos y su representación en el plano.
 - 4.9.2. Operaciones básicas con los números complejos.
 - 4.9.3. Módulo y conjugado.

Duración

12 horas

Unidad V. Polinomios

Competencia

Aplicar las operaciones de suma y multiplicación de polinomios mediante el análisis de sus componentes y propiedades, para determinar las raíces de un polinomio utilizando el algoritmo de la división y el Teorema Fundamental del Álgebra, con pensamiento crítico y responsabilidad.

Contenido

- 5.1. Polinomios
- 5.2. Definición de un polinomio
- 5.3. Identificación de coeficientes de un polinomio
- 5.4. Definición del grado de un polinomio
 - 5.4.1. Operaciones de polinomios
 - 5.4.2. Suma
 - 5.4.3. Multiplicación
 - 5.4.4. División: Cociente y residuo
 - 5.4.4.1. Algoritmo de la división para polinomios
- 5.5. Raíz de un polinomio
 - 5.5.1. Multiplicidad de una raíz
 - 5.5.2. Teorema Fundamental del Álgebra

Duración

06 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

Unidad I. Lógica proposicional

Práctica 1: Propositiones y operadores lógicos

Duración

05 horas

Procedimiento

1. Atiende las orientaciones docentes.
2. Identifica las proposiciones en una lista de enunciados.
3. Traduce a lenguaje simbólico una lista de proposiciones compuestas utilizando operadores lógicos.
4. Construye tablas de verdad de proposiciones compuestas.
5. Realiza consultas a la persona docente para obtener retroalimentación de sus avances.
6. Participa con compartir resultados y razonamientos a otro estudiantado.
7. Resguarda el trabajo realizado en su portafolio de evidencias.

Recursos de apoyo

- Hojas y bolígrafo
- Notas de la unidad de aprendizaje

Práctica 2: Demostraciones matemáticas

Duración

04 horas

Procedimiento

1. Atiende las orientaciones docentes.
2. Escribe una demostración directa de alguna proposición brindada, cuidando claridad y solidez en la argumentación.
3. Escribe una demostración por contradicción de alguna proposición brindada, cuidando claridad y solidez en la argumentación.
4. Escribe una demostración por contrapositiva de alguna proposición brindada, cuidando claridad y solidez en la argumentación.
5. Realiza consultas a la persona docente para obtener retroalimentación de sus avances.
6. Participa con compartir resultados y razonamientos a otro estudiantado.
7. Resguarda el trabajo realizado en su portafolio de evidencias.

Recursos de apoyo

- Hojas y bolígrafo
- Notas de la unidad de aprendizaje

Unidad II. Conjuntos

Práctica 3: Operaciones con conjuntos

Duración:

04 horas

Procedimiento:

1. Atiende las indicaciones de su docente para la realización de la práctica.
2. Revisa definiciones de operaciones entre conjuntos.
3. Resuelve ejercicios correspondientes a las operaciones de conjuntos.
4. Expresa la solución mediante notación de conjuntos y mediante diagramas de Venn.
5. Solicita revisión y retroalimentación de sus avances al profesor.
6. Comparte sus resultados y razonamientos con el resto del grupo, participando de manera activa.
7. Incorpora los ejercicios resueltos al portafolio de evidencias, considerando las observaciones realizadas durante la retroalimentación.

Recursos de apoyo

- Notas de la unidad de aprendizaje
- Recursos bibliográficos
- Hojas y bolígrafo
- Computadora
- Internet

Práctica 4: Demostración de propiedades de las operaciones con conjuntos

Duración

05 horas

Procedimiento

1. Atiende las indicaciones para la realización de la práctica.
2. Revisa definiciones y propiedades fundamentales de operaciones entre conjuntos.
3. Desarrolla demostraciones formales de las propiedades entre conjuntos indicadas por el docente, utilizando razonamiento lógico y notación de conjuntos, justificando cada paso del procedimiento.
4. Solicita revisión y retroalimentación sobre las demostraciones realizadas.
5. Comparte sus resultados y razonamientos con el resto del grupo, participando de manera activa.
6. Integra las demostraciones desarrolladas al portafolio de evidencias, considerando las observaciones realizadas durante la retroalimentación.

Recursos de apoyo

- Notas de la unidad de aprendizaje
- Recursos bibliográficos

- Hojas y bolígrafo
- Computadora
- Internet

Unidad III. Relaciones y funciones

Práctica 5: Relaciones en conjuntos

Duración

06 horas

Procedimiento

1. Atiende las orientaciones para dar solución a problemas sobre relaciones en conjuntos.
2. Reconoce las propiedades de diferentes tipos de relaciones.
3. Utiliza una IA para generar ejemplos de tipos de relaciones en un conjunto.
4. Identifica relaciones elementales en conjuntos.
5. Participa en un debate entre pares y el docente de cómo construir relaciones en conjuntos.
6. Solicita revisión y retroalimentación de los ejercicios a su docente.
7. Escribe el desarrollo de los ejercicios resueltos en un documento de texto y lo entrega a su docente para su revisión y retroalimentación.

Recursos de apoyo

- Notas de la unidad de aprendizaje
- Recursos bibliográficos
- Pizarrón
- Cuaderno y lápiz
- Computadora
- Internet
- Software de IA

Práctica 6: Funciones

Duración

06 horas

Procedimiento

1. Atiende las orientaciones para dar solución a problemas sobre dominio, contradominio, imagen y preimagen de funciones.
2. Reconoce las propiedades de diferentes tipos de funciones.
3. Identifica operaciones de conjuntos bajo funciones.
4. Identifica cuando una función es inyectiva, sobreyectiva o biyectiva.
5. Construye nuevas funciones a partir de otras mediante la operación composición de funciones.
6. Solicita revisión y retroalimentación de los ejercicios a su docente.

7. Escribe el desarrollo de los ejercicios resueltos en un documento de texto y lo entrega a su docente para su revisión y retroalimentación.

Recursos de apoyo

- Notas de la unidad de aprendizaje
- Recursos bibliográficos
- Pizarrón
- Computadora
- Cuaderno y lápiz

Unidad IV. Estructura algebraica de los números reales y complejos

Práctica 7: Números reales

Duración

06 horas

Procedimiento

1. Atiende las orientaciones para demostrar algunas propiedades de los números reales.
2. Identifica los axiomas de campo y los axiomas de orden de los números reales.
3. Participa en un debate entre pares y su docente de cómo aplicar los axiomas de campo y los axiomas de orden de los números reales.
4. Redacta las demostraciones de las propiedades básicas de los números reales.
5. Entrega las demostraciones para su revisión y retroalimentación docente.

Recursos de apoyo

- Notas de la unidad de aprendizaje
- Recursos bibliográficos
- Pizarrón
- Computadora
- Cuaderno y lápiz

Práctica 8: Divisibilidad de los enteros y números complejos

Duración

06 horas

Procedimiento

1. Atiende las orientaciones para aplicar las propiedades de divisibilidad en los enteros y definir las operaciones de números complejos.
2. Identifica el algoritmo de la división y el teorema fundamental de la aritmética.
3. Aplica el algoritmo de Euclides y el teorema fundamental de la aritmética para determinar el máximo común divisor y el mínimo común múltiplo.
4. Reconoce las operaciones de los números complejos.

5. Da solución a problemas de divisibilidad de los enteros y de operaciones de números complejos.
6. Entrega la solución de los problemas para su revisión y retroalimentación docente.

Recursos de apoyo

- Notas de la unidad de aprendizaje.
- Recursos bibliográficos.
- Pizarrón
- Computadora.
- Cuaderno y lápiz

Unidad V. Polinomios

Práctica 9: Operaciones con polinomios

Duración

03 horas

Procedimiento

1. Atiende las orientaciones para resolver problemas que involucren operaciones con polinomios.
2. Identifica las principales propiedades de las operaciones con polinomios (conmutativa, asociativa y distributiva) mediante ejemplos.
3. Resuelve ejercicios de suma, resta, multiplicación y división de polinomios aplicando correctamente la reducción de términos semejantes.
4. Participa en la revisión entre pares y con su docente para analizar procedimientos y corregir errores.
5. Integra las correcciones derivadas de la retroalimentación y mejora los ejercicios realizados.
6. Organiza y agrega los ejercicios resueltos, junto con las observaciones atendidas, al portafolio de evidencias.

Recursos de apoyo

- Notas de la unidad de aprendizaje
- Recursos bibliográficos
- Pizarrón
- Computadora
- Cuaderno y lápiz

Práctica 10: Cálculo de raíces de polinomios

Duración

03 horas

Procedimiento

1. Atiende las orientaciones para dar solución a problemas sobre cálculo de raíces de polinomios.
2. Reconoce las propiedades de las raíces de polinomios.
3. Identifica la multiplicidad de una raíz de un polinomio.
4. Identifica la relación entre el grado del polinomio y el total de raíces contadas con multiplicidad.
5. Solicita revisión y retroalimentación de los ejercicios a su docente.
6. Escribe el desarrollo de los ejercicios resueltos en un documento de texto y lo entrega a su docente para su revisión y retroalimentación.

Recursos de apoyo

- Notas de la unidad de aprendizaje
- Recursos bibliográficos
- Pizarrón
- Computadora
- Cuaderno y lápiz

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Aprendizaje basado en preguntas
- Técnica expositiva
- Discusión guiada
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Aprendizaje basado en preguntas

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	50%
– Tareas	20%
– Prácticas de taller	20%
– Portafolio de evidencias	10%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Angel, A. R., Runde, D. C., & Campos Olguín, V. (2019). *Álgebra intermedia* (9a ed.). Pearson Educación. [Clásica]
- Aufmann, R. N. ; S. Lockwood, J. (2013). *Álgebra intermedia* (8a. ed.). Cengage Learning. Biblioteca electrónica. [Clásica]
- Epp, S. S. (2012). *Matemáticas discretas con aplicaciones* (4a. ed.). Cengage Learning. Biblioteca electrónica. [Clásica]
- Madroñero Pabón, J. (2016). *Guía de matemáticas elementales*. Programa Editorial Universidad del Valle. Biblioteca electrónica (eLibro). [Clásica]
- Kaufmann, J. E., & Schwitters, K. L. (2010). *Álgebra* (8a. ed.). Cengage Learning Biblioteca electrónica. [Clásica]
- Rosen, K. H., & Morales, J. M. P. (2004). *Matemática discreta y sus aplicaciones* (5a ed.). McGraw-Hill. [Clásica]
- Spiegel, M. R., & Moyer, R. E. (2007). *Álgebra Superior* (3a ed.). McGraw-Hill. [Clásica]
- Swokowski, E. W., & Cole, J. A. (2018). *Precálculo: Álgebra y trigonometría con geometría analítica* (12a ed.). Cengage Learning. Biblioteca electrónica. [Clásica]
- Stewart, J., Redlin, L., & Watson, S. (2017). *Precálculo: matemáticas para el cálculo* (7a. ed.). Cengage Learning. [Clásica]

Zalamea Traba, F. (2000). *Fundamentos de matemáticas*. Editorial Universidad Nacional de Colombia. Biblioteca electrónica (eLibro). [Clásica]

Recursos de acceso libre o complementarios externos

Angoa Amador, J.J., Contreras Carreto, A., Ibarra Contreras, M., Linares Gracia, R., & Martínez García, A. (2004). *Matemáticas elementales*. Benemérita Universidad Autónoma de Puebla. [Clásica]

Bravo Mojica, A., Rincón Mejía, H., & Rincón Orta, C. (2013). *Álgebra Superior*. Universidad Nacional Autónoma de México. Acceso libre. [Clásica]

Bulajich Manfrino, R., Hinojosa Palafox, G., & Valdez Delgado, R. (2013). *Introducción al álgebra*. Universidad Autónoma del Estado de Morelos, 2013. Acceso libre. [Clásica]

Gómez Laveaga, C. (2014). *Álgebra Superior. Curso Completo*. Universidad Nacional Autónoma de México. [Clásica]

Villareal Rodríguez, C. E., & Rodríguez Sánchez, S.V. (2018). *Fundamentos de Matemáticas para Ciencias e Ingeniería*. Publicaciones Electrónicas, Sociedad Matemática Mexicana. Acceso libre.

Zakon, E. (2001). *Basic Concepts of Mathematics*. The Trillia Group, 2001.
<http://www.trillia.com/zakon1.html> Acceso libre. [Clásica]

Zubieta Russi, G. (2002). *Lógica Deductiva*. Publicaciones Electrónicas, Sociedad Matemática Mexicana, Serie: Textos, Vol. 1. Acceso libre. [Clásica]

XII. PERFIL DOCENTE

Licenciatura en Matemáticas, Matemáticas Aplicadas o área afín, preferentemente con estudios de posgrado y dos años de experiencia docente. Ser una persona proactiva, analítica y que fomente el trabajo en equipo.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales Licenciatura en Física
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Tecnología y Sociedad
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 02 Horas Laboratorio: 00 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:06
Etapas de formación:	Básica
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Evelio Martínez Martínez Eloisa del Carmen García Canseco Everardo Gutierrez López.
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La unidad de aprendizaje orientada al uso responsable de las Tecnologías de la Información y las Comunicaciones (TIC) es fundamental en la formación profesional del estudiantado, ya que contribuye al desarrollo de criterios éticos y legales necesarios para su desempeño en entornos digitales cada vez más complejos. A través de esta unidad, el estudiantado adquiere conocimientos sobre marcos normativos y legales, desarrolla habilidades para el análisis de dilemas éticos y fortalece actitudes de responsabilidad, integridad y compromiso social, lo que le permite tomar decisiones informadas en el ejercicio profesional relacionado con las TIC.

Para la Licenciatura de Física, se imparte en la etapa básica, es de carácter obligatoria y pertenece al área de conocimiento Computacional.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Analizar los aspectos legales, sociales y éticos relacionados con el uso y desarrollo de las tecnologías de la información y la comunicación, mediante el estudio de marcos normativos, teorías éticas y casos aplicados, con el propósito de tomar decisiones profesionales responsables, actuando con integridad, compromiso social y respeto a la legalidad.

IV. EVIDENCIA DE APRENDIZAJE

Exponer de forma oral y escrita un tema relevante de la asignatura que muestre sus implicaciones sociales, legales y éticas. La exposición deberá incluir una investigación bibliográfica, una estructura clara (introducción, desarrollo y conclusión), uso adecuado de normativa o ejemplos reales, argumentos bien fundamentados, y responder preguntas del auditorio para demostrar comprensión profunda.

Portafolio de trabajos de investigación realizados durante el taller impartido en clases, debidamente organizado, con referencias bibliográficas completas y una conclusión reflexiva que muestre su evolución, aprendizajes y aportaciones personales del proceso.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Esta unidad de aprendizaje promueve competencias genéricas de impacto en los diferentes perfiles de egreso de los programas educativos de la Facultad de Ciencias.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia; Vanguardia; e Innovación Social

VII. DESARROLLO POR UNIDADES

Unidad I. Tecnología y su rol en la sociedad

Competencia

Analizar los conceptos de tecnología, TIC, tecnologías disruptivas, obsolescencia programada y ciclo de vida de la tecnología mediante el estudio de teorías, modelos y casos prácticos con el propósito de comprender su evolución, impacto social y factores de éxito o fracaso, con pensamiento crítico, responsabilidad, ética profesional y apertura a la innovación.

Contenido

- 1.1. Qué es tecnología
 - 1.1.1. Definiendo el término tecnología
 - 1.1.2. Qué es Ciencia y Tecnología
 - 1.1.3. Qué son las TIC
 - 1.1.4. Tecnología y Sociedad
 - 1.1.5. La misión de la tecnología
 - 1.1.6. Beneficios o méritos de las TIC
 - 1.1.7. Desventajas y riesgos de la tecnología
- 1.2. Tecnologías disruptivas
 - 1.2.1. Ejemplos de tecnologías disruptivas
 - 1.2.2. Innovación disruptiva
 - 1.2.3. Caso de estudio: *Kodak*
- 1.3. Obsolescencia tecnológica
 - 1.3.1. Obsolescencia programada
 - 1.3.2. Tipos de obsolescencia programada
 - 1.3.3. El origen de la obsolescencia programada
- 1.4. Ciclo de vida de la tecnología
 - 1.4.1. Ciclo de vida de adopción de tecnologías e innovaciones
 - 1.4.2. El ciclo de *sobreexpectación* de *Gartner*
 - 1.4.3. Factores de éxito o fracaso de una tecnología, producto, proceso o servicio
 - 1.4.4. Tecnologías que han fracasado
 - 1.4.5. Razones por las que las empresas emergentes (*startups*) fracasan

Duración:

04 horas

Unidad II. Aspectos sociales

Competencia

Analizar los aspectos sociales de las TIC mediante el estudio de teorías, modelos, casos prácticos y herramientas tecnológicas con el propósito de comprender su influencia social, económica y ambiental, actuando con pensamiento crítico, responsabilidad social, ética profesional y compromiso con la sostenibilidad.

Contenido

- 2.1. La brecha digital
 - 2.1.1. Qué es la brecha digital
 - 2.2.2. ¿Cómo se puede medir?
 - 2.2.3. ¿Cómo se puede abatir o reducir la brecha digital?
- 2.2. Economía digital
 - 2.2.1. Introducción
 - 2.2.2. El Internet comercial
 - 2.2.3. El auge y la burbuja de las empresas punto-com
 - 2.2.4. ¿Qué es economía digital?
 - 2.2.5. Economía digital y comercio electrónico
 - 2.2.6. Modelos del comercio electrónico
 - 2.2.7. Disrupciones que ha generado la economía digital
 - 2.2.8. *Crowdfunding*
 - 2.2.9. *Crowdsourcing*
 - 2.2.10. Economía compartida
 - 2.2.11. Internet como agente democratizador
 - 2.2.12. Que se necesita para ingresar a la economía digital
- 2.3. Blockchain & criptomonedas
 - 2.3.1. Breve historia del dinero
 - 2.3.2. Las Criptomonedas
 - 2.3.3. Qué es *Blockchain*
 - 2.3.4. Aplicaciones de *Blockchain*
 - 2.3.5. Retos y futuro de *Blockchain*
- 2.4. Redes sociales
 - 2.4.1. Qué son las redes sociales
 - 2.4.2. Historia de las redes sociales
 - 2.4.3. Beneficio de las redes sociales
 - 2.4.4. Desventajas y peligro de las redes sociales
 - 2.4.5. El poder de las redes sociales
- 2.5. TIC y el medio ambiente
 - 2.5.1. Introducción
 - 2.5.2. Qué es *Green IT*
 - 2.5.3. Iniciativas verdes
 - 2.5.4. Propuestas a favor del medio ambiente
 - 2.5.5. Energías alternativas
 - 2.5.6. *Green IT* y las criptomonedas
 - 2.5.7. Economía Circular

Duración:

08 horas

Unidad III. Privacidad y seguridad de la información**Competencia**

Analizar los principios de privacidad, protección de datos personales, ciberseguridad y delitos informáticos mediante el estudio de marcos normativos, casos reales y estándares internacionales con el propósito de proponer estrategias para la protección de la información y la prevención de riesgos digitales, actuando con responsabilidad, ética profesional, respeto a la legalidad y compromiso con la seguridad.

Contenido

- 3.1. Protección de datos personales
 - 3.1.1. Datos personales
 - 3.1.2. Datos sensibles o confidenciales
 - 3.1.3. Datos personales vs. Datos sensibles
 - 3.1.4. Qué es privacidad
 - 3.1.5. Niveles de privacidad
 - 3.1.6. Privacidad vs Seguridad
 - 3.1.7. Caso de estudio: *iPhone - San Bernardino, California*
 - 3.1.8. Reglamentación de la protección de datos personales en el mundo
 - 3.1.9. Caso de estudio: *Cambridge Analytica*
- 3.2. Intercepción de comunicaciones
 - 3.2.1. Caso de estudio: Telegrama *Zimmermann*
 - 3.2.2. La intercepción de las comunicaciones en México
 - 3.2.3. La información es poder
 - 3.2.4. Herramientas de vigilancia
 - 3.2.5. Héroes o Villanos: *Edward Snowden, Bradley Manning, Julian Assange, Aaron Swartz, Brittany Kaiser*
- 3.3. Seguridad de la información
 - 3.3.1. Qué es un delito informático
 - 3.3.2. Tipos de delitos informáticos
 - 3.3.3. Estándares en ciberseguridad
 - 3.3.4. Otros recursos sobre ciberseguridad
 - 3.3.5. Guerra cibernética (*Cyber warfare*)
- 3.4. Privacidad, Seguridad e Inteligencia Artificial
 - 3.4.1. Protección de datos en entrenamiento de modelos de IA
 - 3.4.2. Derechos ARCO y procesamiento automatizado
 - 3.4.3. Inferencia de información sensible
 - 3.4.4. Seguridad de modelos de IA y brechas de datos
 - 3.4.5. Privacidad diferencial, cifrado y privacidad por diseño
 - 3.4.6. Shadow AI

Duración:

06 horas

Unidad IV. Propiedad Intelectual**Competencia**

Analizar los fundamentos, normativas y figuras de protección de la propiedad intelectual, incluyendo derechos de autor, propiedad industrial, licencias, *DRM* y *NFT*, mediante el estudio de leyes, tratados internacionales y casos prácticos con el propósito de aplicar correctamente la legislación en el desarrollo y protección de productos tecnológicos, actuando con responsabilidad, ética profesional, respeto a la legalidad y honestidad.

Contenido

- 4.1. Qué es la propiedad intelectual
 - 4.1.1. Historia y origen de la Propiedad Intelectual
 - 4.1.2. El Estatuto de Venecia, Italia (1474)
 - 4.1.3. El Estatuto de los Monopolios (1624)
 - 4.1.4. El Estatuto de Anne (1710)
 - 4.1.5. Convenio de París (1883)
 - 4.1.6. Convenio de Berna (1886)
 - 4.1.7. Convenio de Roma (1961)
 - 4.1.8. Fundación de la WIPO (1967)
 - 4.1.9. TRIPS Agreement (1995)
 - 4.1.10. Impacto y beneficios de la PI
- 4.2. La propiedad intelectual en México
 - 4.2.1. Derechos de autor; propiedad industrial; derechos de obtentor
- 4.3. Derechos de autor
 - 4.3.1. Tipos de obras protegidas por la LFDA
 - 4.3.2. Tipos de Derechos que adquieren los autores
 - 4.3.3. Programas de cómputo (*software*)
 - 4.3.4. Derechos conexos
 - 4.3.5. Limitaciones al derecho de autor y los derechos conexos por causa de utilidad pública
 - 4.3.6. Dominio público
 - 4.3.7. Símbolos patrios
 - 4.3.8. Reservas de Derechos al uso exclusivo
- 4.4. Propiedad industrial
 - 4.4.1. ¿Qué es propiedad industrial?
 - 4.4.2. Figuras de protección del IMPI
 - 4.4.3. Patentes
 - 4.4.4. Modelo de Utilidad
 - 4.4.5. Diseños industriales
 - 4.4.6. Tabla Comparativa entre Patentes, modelo de utilidad y diseños industriales
 - 4.4.7. Trazado de Circuitos

- 4.4.8. Secretos industriales
- 4.4.9. Signos distintivos
- 4.4.10. Marcas
- 4.4.11. Marcas no tradicionales
- 4.4.12. Resumen de las marcas tradicionales y no tradicionales
- 4.4.13. Avisos comerciales
- 4.4.14. Nombres comerciales
- 4.4.15. Denominaciones de origen
- 4.4.16. Indicaciones geográficas
- 4.4.17. Búsquedas
- 4.5. Licencias y transmisión de derechos de marcas
 - 4.5.1. Franquicias
- 4.6. Piratería
 - 4.6.1. Piratería y la propiedad intelectual
- 4.7. DRM
 - 4.7.1. Qué es DRM
 - 4.7.2. Ventajas y desventajas del DRM
- 4.8. Los NFT y la propiedad intelectual
 - 4.8.1. Origen de los NFT
 - 4.8.2. ¿Qué son los NFT?
 - 4.8.3. Aplicaciones
 - 4.8.4. Los NFT y la propiedad intelectual
- 4.9. Derechos de propiedad Intelectual en IA Generativa
 - 4.9.1. Autoría y Titularidad (*Ownership & Authorship*)
 - 4.9.2. Uso de datos protegidos para entrenamiento
 - 4.9.3. Producción de obras derivadas o imitaciones demasiado similares
 - 4.9.4. Originalidad, creatividad y novedad
 - 4.9.5. Licencias y remuneración
 - 4.9.6. Protección técnica e identificación de origen
 - 4.9.7. Derechos morales y éticos del creador original
 - 4.9.8. Desafíos legales y vacíos normativos
 - 4.9.10. Demandas legales recientes y precedentes
 - 4.9.11. Implicaciones para políticas, éticas y acceso al conocimiento

Duración:

08 horas

Unidad V. Aspectos éticos

Competencia

Analizar los principios, teorías y códigos éticos, así como los dilemas y procesos de toma de decisiones en el ámbito computacional mediante el estudio de fundamentos teóricos, casos prácticos y marcos normativos con el propósito de desarrollar criterios éticos para la práctica profesional, actuando con responsabilidad, integridad, pensamiento crítico y respeto a la legalidad.

Contenido

- 5.1. Introducción a la ética
 - 5.1.1. ¿Qué es la Ética?
 - 5.1.2. Ética vs moral
 - 5.1.3. la ética, la moral y la ley
 - 5.1.4. Relación entre la ética y la ley
 - 5.1.5. La ética vs. La ley
- 5.2. Teorías éticas
 - 5.2.1. La ética de la virtud (*Virtue Ethics*)
 - 5.2.2. La ética Deontológica
 - 5.2.3. La ética consecuencialista
- 5.3. Ética computacional
 - 5.3.1. Origen de la ética computacional
 - 5.3.2. ¿Qué es ética computacional?
- 5.4. Códigos de la ética
 - 5.4.1. ¿Qué es un código de ética?
 - 5.4.2. Objetivos de los códigos de ética
 - 5.4.3. Tipos de códigos de ética
- 5.5. Dilemas éticos
 - 5.5.1. ¿Qué es un dilema ético?
 - 5.5.2. El dilema del tranvía
 - 5.5.3. El dilema del prisionero
 - 5.5.4. El dilema del prisionero aplicado en los deportes
 - 5.5.5. Dilema ético de *Harry S. Truman*
- 5.6. Ética en la toma de decisiones
- 5.7. Proceso para la toma de decisiones éticas de Kallman & Grillo
 - 5.7.1. Enfoque deontológico
 - 5.7.2. Enfoque consecuencialismo
 - 5.7.3. Otros principios normativos
 - 5.7.4. Aspectos Legales
 - 5.7.5. Proceso de cuatro pasos para el análisis ético y toma de decisiones de Kallman & Grillo
- 5.8. Caso de estudio TheranosS
- 5.9. Aspectos éticos de la inteligencia artificial generativa
 - 5.9.1. Sesgo y equidad
 - 5.9.2. Transparencia y explicabilidad (XAI)
 - 5.9.3. Privacidad de datos y consentimiento informado
 - 5.9.4. Responsabilidad y rendición de cuentas
 - 5.9.5. Autonomía humana y control humano
 - 5.9.6. Impacto social y laboral
 - 5.9.7. Uso ético en contextos especiales
 - 5.9.8. Seguridad, robustez y mitigación de riesgos

Duración:

04 horas

Unidad VI. Aspectos legales**Competencia**

Analizar el marco jurídico y regulatorio de las Tecnologías de la Información y Comunicación (TIC) mediante el estudio de la legislación mexicana, la jerarquía normativa, los organismos reguladores de Internet y los principios de neutralidad de la red para fundamentar el desarrollo y implementación de soluciones tecnológicas que cumplan con las disposiciones legales vigentes y promuevan el acceso equitativo a la información con responsabilidad social, ética profesional y compromiso ciudadano.

Contenido

- 6.1. Introducción a la legislación en TIC
 - 6.1.1. Tópicos por legislar en TIC
 - 6.1.2. Ámbito material de validez de las normas jurídicas
- 6.2. La jerarquía de las Leyes
 - 6.2.1. La Pirámide de Kelsen
 - 6.2.2. El sistema jurídico mexicano
- 6.3. Quién regula internet
 - 6.3.1. Qué es Internet
 - 6.3.2. ¿Entonces quién regula Internet?
 - 6.3.3. Actores clave en el funcionamiento de Internet
- 6.4. La neutralidad de la red
 - 6.4.1. Qué es Neutralidad de la Red
 - 6.4.2. ¿Por qué es importante la Neutralidad?
 - 6.4.3. La censura en Internet
 - 6.4.4. Malas prácticas que atentan contra la neutralidad
- 6.5. La neutralidad de la red en México
- 6.6. Aspectos legales y la inteligencia artificial
 - 6.6.1. Regulación legal específica de IA en México
 - 6.6.2. Licencia y uso de datos para entrenamiento
 - 6.6.3. Responsabilidad y cumplimiento legal
 - 6.6.4. Privacidad, protección de datos y transparencia en IA legalmente regulada
 - 6.6.5. Neutralidad legal del autor/inventor de IA
 - 6.6.6. Comparativa internacional y buenas prácticas

Duración:

2 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER**Unidad I. Introducción a las TIC****Práctica 1. Obsolescencia Programada****Duración:**

02 horas

Procedimiento:

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Elabora un resumen y una conclusión a partir de una investigación sobre el documental Obsolescencia programada: Comprar, tirar, comprar" del Director: Cosima Dannoritzer, producido por: Televisión de Galicia (TVG) / Televisión de Galicia (TVG) / Arte France / Article Z / Media 3.14 con extensión de 1 a 2 cuartillas.
3. Reflexiona y responde las siguientes preguntas de manera escrita:
 - a. ¿Qué piensas de la Obsolescencia Programada?
 - b. ¿Es realmente necesaria?
 - c. ¿Es ético fabricar productos que fallen?
 - d. ¿Comprarías un producto que sabes que va a fallar?
4. Entrega un documento en formato *.pdf*, con una extensión mínima de dos cuartillas con el resumen y conclusión del documental y las respuestas a las preguntas, utilizando formato APA de las fuentes consultadas.

Recursos de apoyo

- El documental se encuentra disponible en *Youtube.com*, en el siguiente URL: <https://www.youtube.com/watch?v=fzwT1GGY0Ds> [puede variar]

Unidad II. Aspectos Sociales**Práctica 2. El valor del dinero en la economía digital****Duración:**

02 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Elabora un resumen y una conclusión a partir de una investigación sobre el documental "Bitcoin: el fin del dinero tal como lo conocemos" del director: *Torsten Hoffmann y Michael Watchulonis* (co-director).
3. Reflexiona y responde las siguientes preguntas de manera escrita:

Lista las 5 criptomonedas con más capital invertido (*market capitalization*) en el mundo e indique la cantidad en *billones de dólares* invertidos de cada moneda.

¿Están en peligro los bancos tradicionales?

Qué es *Blockchain* y mencione 5 aplicaciones

Liste 5 razones que definen el valor de una criptomoneda

4. Entrega un documento en formato *.pdf*, con una extensión mínima de dos cuartillas con el resumen y conclusión del documental y las respuestas a las preguntas, utilizando formato APA de las fuentes consultadas.

Recursos de apoyo

- El documental se encuentra disponible en *Youtube.com*, en el siguiente URL:
<https://youtu.be/-TmpLBG1ZdQ?si=njRC9NqtF2FBDS9C>

Práctica 3. ¿Qué es el fracking?

Duración:

02 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Elabora un resumen y una conclusión a partir de una investigación sobre el documental “¿Qué es el fracking? Explicado en 5 minutos” de *GreenPeace México*.
3. Reflexiona y responde las siguientes preguntas de manera escrita:
 - ¿Qué es el fracking y en qué consiste el proceso técnico de extracción?
 - ¿Cuáles son las principales zonas de México donde se han identificado reservas de gas o petróleo que podrían explotarse con fracking?
 - ¿Qué impactos ambientales negativos se han documentado en relación con el fracking (agua, aire, suelo, biodiversidad)?
 - ¿Cómo afecta el uso intensivo de agua en el fracking a regiones de México con problemas de sequía o escasez hídrica?
 - ¿Qué leyes, normas o regulaciones mexicanas existen en torno al fracking y qué limitaciones tienen en cuanto a la protección ambiental?
 - Desde una perspectiva ética y de sustentabilidad, consideras viable el uso del fracking en México? ¿Por qué?
4. Entrega un documento en formato *.pdf*, con una extensión mínima de dos cuartillas con el resumen y conclusión del documental y las respuestas a las preguntas, utilizando formato APA de las fuentes consultadas.

Recursos de apoyo

- El documental se encuentra disponible en *Youtube.com*, en el siguiente URL:
<https://youtu.be/S9MmaTO7-vA?si=fA82BYU0783sIJfZ>

Práctica 4. El fin del petróleo

Duración:

02 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Elabora un resumen y una conclusión a partir de una investigación sobre el documental “*El fin del petróleo, no hay mañana*” de *Incubate Pictures*. Director: *Dermot O’Connor*.
3. Reflexiona y responde las siguientes preguntas de manera escrita:

¿Cuál es la hipótesis principal que plantea el documental respecto al fin del petróleo?

¿Qué evidencias científicas o estadísticas se presentan para mostrar que el petróleo podría estar en declive?

¿Qué implicaciones económicas tiene para los países productores si el petróleo deja de ser rentable o útil?

En el caso de México, ¿qué impactos sociales, económicos y ambientales podrían derivarse de una reducción significativa en la producción o uso del petróleo?

¿Qué papel juegan las políticas públicas nacionales e internacionales en la transición energética?

¿Qué dilemas éticos surgen al decidir entre conservar modelos económicos basados en petróleo o apostar por energías renovables que podrían ser más costosas o menos lucrativas inicialmente?

¿Qué opinas sobre la viabilidad del *fracking* como alternativa para extender la dependencia del petróleo en México?

¿Qué responsabilidades tienen los ciudadanos, empresas, gobiernos y organismos internacionales ante la posibilidad de que los recursos fósiles disminuyan?

¿Cuál será la energía del futuro?

4. Entrega un documento en formato *.pdf*, con una extensión mínima de dos cuartillas con el resumen y conclusión del documental y las respuestas a las preguntas, utilizando formato APA de las fuentes consultadas.

Recursos de apoyo

- El documental se encuentra disponible en *Youtube.com*, en el siguiente URL:
<https://youtu.be/8VRgo62Dw5Y>

Práctica 5. Basura electrónica

Duración:

02 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Elabora un resumen y una conclusión a partir de una investigación sobre el documental *"El vertedero electrónico"* de RT en español.
3. Reflexiona y responde las siguientes preguntas de manera escrita:
 - ¿Cuántas toneladas de basura electrónica genera México al año y cómo se compara con otros países?
 - ¿Cuáles son los principales países que vierten basura electrónica?
 - ¿Qué países reciben basura electrónica por toneladas?
 - ¿Es legal enviar basura electrónica a estos países?

Desde una perspectiva ética, ¿cuál sería la responsabilidad de los fabricantes, usuarios y gobierno para reducir el impacto de la basura electrónica?
4. Entrega un documento en formato *.pdf*, con una extensión mínima de dos cuartillas con el resumen y conclusión del documental y las respuestas a las preguntas, utilizando formato APA de las fuentes consultadas.

Recursos de apoyo

- El documental se encuentra disponible en *Youtube.com*, en el siguiente URL:
<https://www.youtube.com/watch?v=zfDWE0TyS-w&t=12s>

Unidad III. Privacidad y seguridad de la información

Práctica 6. Seguridad y Privacidad de la Información

Duración:

02 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Reflexiona y responde las siguientes preguntas de manera escrita:
 - Qué organismo en México promueve la *transparencia, acceso a la información y la protección de datos personales*.
 - Menciona las leyes en México sobre transparencia, acceso a la información y protección de datos personales

¿Qué normativas universitarias existen en este sentido?

¿Cuáles son los principios y obligaciones que los sujetos obligados tienen bajo la Ley General de Transparencia y Ley General de Protección de Datos Personales en México?

¿Qué derechos ARCO (Acceso, Rectificación, Cancelación y Oposición) tiene un titular de datos personales, y cómo puede ejercerlos?

En tu opinión, ¿qué desafíos hay en México para que la transparencia y protección de datos realmente funcionen (por ejemplo: institucionales, culturales, tecnológicos)?

¿Cómo se equilibra el derecho a la información pública con el derecho a la privacidad de las personas?

¿Qué responsabilidad tienen los ciudadanos para exigir transparencia y proteger sus propios datos personales (por ejemplo al usar redes sociales, servicios digitales)?

Podrías elaborar una minibiografía de los siguientes personajes: *Bradley Manning*, *Edward Snowden*, *Julian Assange*, *Aaron Swartz* y *Brittany Kaiser*.

¿Qué tienen en común estos personajes?

Desde tu punto de vista ¿Los considera héroes o villanos?

3. Entrega un documento en formato *.pdf*, con una extensión mínima de dos cuartillas con el resumen y conclusión del documental y las respuestas a las preguntas, utilizando formato APA de las fuentes consultadas.

Recursos de apoyo

- Ley General de Transparencia y Acceso a la Información Pública <https://www.diputados.gob.mx/LeyesBiblio/pdf/LGTAIP.pdf>
- Ley General de Protección de Datos Personales en Posesión de Sujetos Obligados <https://www.diputados.gob.mx/LeyesBiblio/pdf/LGPDPSO.pdf>

Práctica 7: Internet de las cosas y la privacidad

Duración:

02 horas

Procedimiento

Atiende las orientaciones docentes para el desarrollo de la práctica.

1. Elabora un resumen y una conclusión a partir de una investigación sobre el documental *"The Internet of Everything"* del Director *Brett Gaylor*.
2. Reflexiona y responde las siguientes preguntas de manera escrita:

¿Cual de los casos que se mencionan en el documental le pareció el más invasivo a la privacidad? ¿Y por qué?

- ¿Hasta qué punto estamos dispuestos a sacrificar nuestra privacidad por la comodidad que ofrecen los dispositivos conectados?
 - ¿Estamos creando un mundo donde perdemos la capacidad de funcionar sin tecnología?
 - ¿Qué sucede cuando los sistemas interconectados fallan o son comprometidos?
 - ¿El IoT está ampliando o reduciendo la brecha digital entre diferentes comunidades?
 - ¿Quién se beneficia realmente de esta hiperconectividad?
 - ¿Mantenemos el control sobre nuestras decisiones o los algoritmos están decidiendo por nosotros?
3. Entrega un documento en formato *.pdf*, con una extensión mínima de dos cuartillas con el resumen y conclusión del documental y las respuestas a las preguntas, utilizando formato APA de las fuentes consultadas.

Recursos de apoyo

- El documental se encuentra disponible en los siguientes URL:
https://youtu.be/PYEAVC5lfes?si=ssklfoPYM2V62_FS (versión en inglés)
<https://www.facebook.com/watch/?v=436353974840100> (versión en español)

Unidad IV. Propiedad intelectual

Práctica 8: Documental - ¡Copiad Malditos!

Duración:

02 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Elabora un resumen y una conclusión a partir de una investigación sobre el documental *¿Copiad Malditos! de Elegantmobfilms, RTVE (2011)*
3. Reflexiona y responde las siguientes preguntas de manera escrita:
 - ¿Es realmente justo que las ideas puedan "poseerse" de la misma manera que los objetos físicos?
 - ¿El sistema actual de derechos de autor fomenta o limita la creatividad?
 - ¿Debería la cultura ser un bien común accesible para todos?
 - ¿Cómo afectan las restricciones de copyright a la educación y la investigación?
 - ¿Pueden las licencias *Creative Commons* y otros modelos alternativos sostener económicamente a los creadores?
 - ¿Es viable un mundo sin *copyright* tradicional?

¿Qué significa que el propio documental use la licencia que defiende?

4. Entrega un documento en formato *.pdf*, con una extensión mínima de dos cuartillas con el resumen y conclusión del documental y las respuestas a las preguntas, utilizando formato APA de las fuentes consultadas.

Recursos de apoyo

- El documental se encuentra disponible en *Youtube.com*, en el siguiente URL:
<https://youtu.be/YY0i4xJss9c>

Práctica 9. Documental - La Guerra de las Patentes

Duración:

2 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Elabora un resumen y una conclusión a partir de una investigación sobre el documental *"La Guerra de las Patentes"* de *Hannah Leonie Prinzler (RTVE)*.
3. Reflexiona y responde las siguientes preguntas de manera escrita:
 - ¿Cómo hemos pasado de un sistema diseñado para "estimular a los inventores" a crear "oscuras estrategias" monopolísticas?
 - ¿Quién se beneficia más del sistema actual: los inventores originales o las grandes corporaciones?
 - ¿Deberían existir límites a qué se puede patentar (medicamentos, software, genes)?
 - ¿Qué habría pasado si inventos como "la bombilla, la máquina de vapor o el teléfono" hubieran estado sujetos a las estrategias monopolísticas actuales?
 - ¿Es sostenible un mundo donde el conocimiento esencial esté fragmentado por monopolios?
4. Entrega un documento en formato *.pdf*, con una extensión mínima de dos cuartillas con el resumen y conclusión del documental y las respuestas a las preguntas, utilizando formato APA de las fuentes consultadas.

Recursos de apoyo

- El documental se encuentra disponible en las siguientes URL:
<https://youtu.be/ZmJektY7ToM>

Práctica 10. Documental - Code Linux

Duración:

02 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Elabora un resumen y una conclusión a partir de una investigación sobre el documental *"The Code (código linux)"* del director *Hannu Puttonen*.
3. Reflexiona y responde las siguientes preguntas de manera escrita:

Haga una lista de 5 personajes más importantes que aparecen en el documental

¿Quién fue el personaje iniciador del movimiento Software Libre?

¿Quién fue el personaje que inició el desarrollo del Sistema Operativo GNU/Linux?

¿Quién tiene la propiedad intelectual del Sistema Operativo GNU/Linux?

¿Quién se encarga de desarrollar el núcleo (core) del S.O. GNU/Linux y administrar los cambios?

Cuáles son las 4 libertades del Software Libre

¿Qué pasaría si Linux perteneciera a alguna empresa?

¿Por qué miles de programadores contribuyen gratuitamente a un proyecto común?

¿Es sostenible un modelo donde el producto principal es gratuito pero se monetizan los servicios?

¿Cómo puede el software libre democratizar el acceso a la tecnología?

¿Qué otros ámbitos podrían beneficiarse de modelos similares (medicina abierta, educación libre, etc.)?

¿Cómo se relaciona el éxito de Linux con las críticas al sistema de patentes y copyright que vimos en los otros documentales?

¿Es el software libre una respuesta práctica a "la guerra de las patentes"?

4. Entrega un documento en formato *.pdf*, con una extensión mínima de dos cuartillas con el resumen y conclusión del documental y las respuestas a las preguntas, utilizando formato APA de las fuentes consultadas.

Recursos de apoyo

- El documental se encuentra disponible en las siguientes URL:
<https://youtu.be/HNho1tXWNHI>

Unidad V. Aspectos éticos

Práctica 11. Análisis y toma de decisiones

Duración:

06 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Elabora un documento basado en el “proceso de 4 pasos para el análisis y toma de decisiones” de los autores *Kallman & Grillo* de un caso de estudio en el ámbito de las TIC que considere relevante, visto en clase o de actualidad.
3. Entrega un documento en formato *.pdf*, con una extensión mínima de cinco cuartillas con el resumen y conclusión del documental y las respuestas a las preguntas, utilizando formato APA de las fuentes consultadas.

Casos reales sugeridos:

1. El escándalo del físico *Jan Hendrik Schön* (manipulación de datos)
2. El caso del físico *Rusi Taleyarkhan* (misconduct in fusion researcher case)
3. Caso del matemático *Dănuț Marcu* (acusaciones de plagio)
4. Caso *Workday* (discriminación en herramientas de contratación)
5. Caso *Amazon* y sesgos en algoritmos de selección de currículos
6. Caso [Builder.ai](#) (IA), *One Coin*, *FTX*, *Wirecard* (sector Fintech), *Nikola* (autos eléctricos), *Volkswagen* (Dieselgate), *Bernie Madoff* (sector financiero), *Lehman Brothers*, *Wells Fargo* (sector bancario), *Valeant* (farmacéutica), *Parmalat* (alimentos).

Recursos de apoyo:

- Martínez, Evelio (2024). *Tecnología y Sociedad: Perspectivas legales, sociales y éticas sobre las tecnologías de la información y la comunicación*. Editorial Convergente. México.
- Kallman, Ernest A., Grillo, John P. (1995). *Ethical Decision Making & Information Technology: An Introduction with Cases*. McGraw-Hill/Irwin.

Unidad VI. Aspectos legales

Práctica 6. Análisis del marco legal en TIC

Duración:

06 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el marco jurídico de las TIC en México.
2. Analiza un caso real relacionado con regulación de internet, neutralidad de la red o uso de datos personales.}
3. Identifica las leyes, normas y organismos involucrados.
4. Evalúa si el caso cumple o vulnera el marco legal vigente.
5. Elabora un documento donde exponga el análisis del caso, los fundamentos legales y una postura crítica.
6. Entrega el documento en el buzón de tareas y recibe retroalimentación docente.

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Estudio de caso
- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Aprendizaje basado en preguntas
- Técnica expositiva
- Debates
- Discusión guiada
- Instrucción guiada
- Foros

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Análisis de videos documentales
- Cuadros comparativos
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	30%
– Tareas	40%
– Presentación oral y escrita	30%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Base, Sara. (2013). *A Gift of Fire: Social, Legal, and Ethical Issues for Computing Technology*. Pearson. (4th ed.). [clásica].
- Cámara de Diputados del H. Congreso de la Unión. (2020). *Ley Federal del Derecho de Autor*. <https://www.diputados.gob.mx/LeyesBiblio/pdf/LFDA.pdf>
- Cámara de Diputados del H. Congreso de la Unión. (2020). *Ley Federal de Protección a la Propiedad Industrial*. <https://www.diputados.gob.mx/LeyesBiblio/pdf/LFPPI.pdf>
- Ceruzzi, Paul E. (2012). *Computing: A concise history*. The MIT press Essential Knowledge Series [clásica].
- Kallman, Ernest A., Grillo, John P. (1995). *Ethical Decision Making & Information Technology: An Introduction with Cases*. McGraw-Hill/Irwin. [clásica].
- Kizza, Joseph Migga. (2016). *Ethics in Computing*. Springer. Switzerland. [clásica].
- Martínez, Evelio (2024). *Tecnología y Sociedad: Perspectivas legales, sociales y éticas sobre las tecnologías de la información y la comunicación*. Editorial Convergente. México.
- Rogers, Everett M. (2003). *Diffusion of innovations* (5th ed.). Free Press [clásica].
- Tapscott, Don. (1995). *The Digital Economy: Promise and Peril in the Age of Networked Intelligence*. McGraw-Hill. [clásica].

Recursos de acceso libre o complementarios externos

- Gorniak-Kocikowska, K. (1996). The computer revolution and the problem of global ethics. *Sci Eng Ethics*, 2, 177–190. <https://doi.org/10.1007/BF02583552>
- Haber, Stuart, Stornetta, W. Scott. (1991). How to time-stamp a digital document. *Journal of Cryptology* 3, 99–111 (1991). <https://doi.org/10.1007/BF00196791>
- Isaak, J., & Hanna, M. J. (2018). User data privacy: Facebook, Cambridge Analytica, and privacy protection. *Computer*, 51(8), 56-59. <https://doi.org/10.1109/MC.2018.3191268>
- Long, Pamela O. (1991). Invention, Authorship, Intellectual Property, and the Origin of Patents: Notes toward a Conceptual History. *Technology and Culture*, 32(4), 846-884 <https://doi.org/10.2307/3106154>

Scott J. Shackelford. (2013). Managing Cyber Attacks in International Law, Business, and Relations. *Search of Cyber Peace*. Cambridge University Press. <https://doi.org/10.1017/CBO9781139021838>

Skórzewska-Amberg, M. (2017). Impact of ICT on the Law. In: Zacher. L. Technology. Society and Sustainability. Springer, Cham. https://doi.org/10.1007/978-3-319-47164-8_11

XII. PERFIL DOCENTE

Formación de posgrado (maestría o doctorado) en el área de TIC, con conocimientos en derecho informático, ética profesional y responsabilidad social, así como dominio actualizado de la legislación nacional e internacional aplicable al entorno digital. Debe tener experiencia en el análisis de dilemas éticos y legales en contextos tecnológicos, habilidades didácticas para promover el pensamiento crítico y la reflexión ética, y demostrar integridad, compromiso con la formación de personas ciudadanas responsables y respeto por la diversidad de opiniones.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada.
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Introducción a la Programación
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 03 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Básica
Carácter:	Obligatoria
Requisito:	Diseño de algoritmos
Modalidad de impartición:	Presencial
Equipo de diseño:	María Victoria Meza Kubo Adrián Enciso Almanza Raúl Casillas Figueroa
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

En esta unidad de aprendizaje se fortalece el aprendizaje en lógica programática mediante el uso de un lenguaje de programación. Su finalidad es capacitar al estudiantado en el uso de los fundamentos de programación que le permitan producir programas eficientes que cumplan estándares de calidad.

La unidad de aprendizaje de Introducción a la Programación es de carácter obligatorio dentro de la etapa básica. Su área de conocimiento es Programación e Ingeniería de software y tiene como requisito haber cursado y aprobado la unidad de aprendizaje de Diseño de Algoritmos.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Desarrollar programas de computadora estructurados, mediante la aplicación e integración de estándares de buenas prácticas y técnicas inherentes a los conceptos de programación, para la adquisición de experiencia en el desarrollo de software, con una actitud crítica, responsable y de compromiso para el trabajo colaborativo.

IV. EVIDENCIA DE APRENDIZAJE

Proyecto integrador de programación estructurada que demuestre la aplicación efectiva de estructuras de control, estructuras de datos y estándares de programación, conforme a las buenas prácticas del desarrollo de software. Este proyecto, realizado de manera individual o grupal, deberá evidenciar la capacidad para analizar, diseñar, codificar, depurar y documentar soluciones funcionales, aplicando criterios de eficiencia, legibilidad y cumplimiento de requerimientos. Se evaluará la correcta implementación de las estructuras programáticas, la adherencia a los estándares de codificación, la funcionalidad del programa, la claridad en la documentación y la entrega oportuna. Con esta evidencia se busca fortalecer habilidades de pensamiento lógico, resolución de problemas, comunicación técnica y trabajo colaborativo, esenciales para el ejercicio profesional en el ámbito de la programación.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia.

VII. DESARROLLO POR UNIDADES

Unidad I. Introducción a la programación

Competencia

Interpretar los conceptos que se involucran en un ambiente de programación de computadoras, mediante la revisión de los paradigmas y fundamentos de programación, para obtener las bases teóricas de la lógica programática, con una actitud crítica y responsable.

Contenido

- 1.1. Cronología de los paradigmas de programación
- 1.2. Programación estructurada
- 1.3. Estructura básica de un programa
- 1.4. Edición, compilación y depuración de un programa
- 1.5. Estándares de buenas prácticas de programación

Duración

02 horas

Unidad II. Constantes, variables, tipos, expresiones y asignaciones

Competencia

Aplicar los distintos tipos de datos y expresiones de programación de computadoras, mediante el uso de un lenguaje de programación estructurado, para interpretar la manera de definir y manipular datos dentro de un programa de cómputo, con una actitud crítica, creativa y responsable.

Contenido

- 2.1. Tipos de datos constantes y variables
 - 2.1.1. Definición de variables y constantes (zonas de memoria)
 - 2.1.2. Tipos de datos: numéricos, carácter, cadena, booleanos
 - 2.1.3. Operadores de relación, asignación, aritméticos y asignación implícita de tipos
 - 2.1.4. Jerarquía de operadores
 - 2.1.5. Operadores de manejo de cadenas
- 2.2. Expresiones y sentencias
 - 2.2.1. Expresiones aritméticas y lógicas
 - 2.2.2. Sentencias simples y compuestas
 - 2.2.3. Operadores de incremento y decremento
 - 2.2.4. Bloque de sentencias

Duración

02 horas

Unidad III. Estructuras de control y depuración de programas: sentencias, condicionales e iteraciones

Competencia

Aplicar las estructuras de control en la formulación de programas de computadora, mediante el uso de un lenguaje de programación, para automatizar actividades y procesos de propósito general, con una actitud lógica, analítica y perseverante.

Contenido

- 3.1. Fundamentos básicos de estructuras de control
 - 3.1.1. Operaciones booleanas
 - 3.1.2. Negación, conjunción, expresiones complejas
 - 3.1.3. Cálculo de predicados
- 3.2. Secuencia
 - 3.2.1. Análisis de problemas de programación
 - 3.2.2. Secuencia lógica
 - 3.2.3. Bloque de sentencias
 - 3.2.4. Implementación y depuración
- 3.3. Selección
 - 3.3.1. Definición y componentes de una condicional
 - 3.3.3. Condicionales sencillas
 - 3.3.4. Condicionales dobles
 - 3.3.5. Condicionales múltiples
 - 3.3.6. Condicionales anidadas
 - 3.3.7. Implementación y depuración
- 3.4. Iteración
 - 3.4.1. Definición y componentes de un ciclo
 - 3.4.2. Tipos de ciclos: por contador y por centinela
 - 3.4.3. Anidación de ciclos
 - 3.4.4. Implementación y depuración

Duración

08 horas

Unidad IV. Funciones

Competencia

Integrar el uso de funciones en los programas de computadora mediante la modularización y simplificación de la complejidad de la solución para optimizar la funcionalidad y el mantenimiento del código de programación, con una actitud coherente, lógica y paciente.

Contenido

- 4.1. Definición y componentes de una función
- 4.2. Cuerpo, llamado y prototipos de funciones

- 4.3. Funciones con parámetros por valor
- 4.4. Funciones con parámetros por referencia
- 4.5. Implementación y depuración

Duración

06 horas

Unidad V. Estructuras de datos

Competencia

Diseñar espacios de almacenamiento de datos compuestos, mediante el uso de las diferentes estructuras de datos disponibles en los lenguajes de programación, para optimizar el manejo de memoria y la manipulación de datos dentro de un programa de cómputo, con una actitud creativa, propositiva y trabajo colaborativo.

Contenido

- 5.1. Estructuras de datos del mismo tipo (arreglos)
 - 5.1.1. Arreglos unidimensionales
 - 5.1.2. Arreglos multidimensionales
- 5.2. Estructuras de datos multi tipo (registros)
- 5.3. Implementación y depuración

Duración

06 horas

Unidad VI. Tópicos avanzados de programación

Competencia

Desarrollar programas de cómputo robustos, a través del uso de funcionalidades avanzadas disponibles en lenguajes de programación, para optimizar el desempeño de los programas de cómputo realizados, con una actitud propositiva, creativa y trabajo colaborativo.

Contenido

- 6.1. Recursividad
- 6.2. Manejo dinámico de memoria
- 6.3. Manejo de archivos
- 6.4 Estructuras avanzadas

Duración

08 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Introducción a la Programación

Práctica 1: Mi primer programa

Duración:

03 horas

Procedimiento:

1. Atiende a la explicación de la persona docente sobre el uso de los IDEs.
2. Analiza los problemas proporcionados en la plataforma del curso.
3. Identifica las partes de la solución: entradas, procesamiento, salidas.
4. Genera el pseudocódigo o diagrama de flujo para dar una solución.
5. Desarrolla la solución en el lenguaje de programación seleccionado para el curso, utilizando el IDE de tu preferencia.
6. Realiza capturas de pantalla que muestran la correcta ejecución del programa.
7. Integra en un documento el pseudocódigo, código y capturas de pantalla de las soluciones propuestas.
8. Entregar en formato PDF en la plataforma del curso.
9. Recibir retroalimentación docente.

Recursos de apoyo:

- IDE seleccionado para desarrollo.
- Computadora.
- Acceso a internet.

Unidad II. Constantes, variables, tipos, expresiones y asignaciones

Práctica 2: Manejo de tipo de datos y operadores

Duración:

03 horas

Procedimiento:

1. Atiende a la explicación de la persona docente sobre el uso de tipos de variables.
2. Analiza los problemas proporcionados en la plataforma del curso para distinguir los tipos de variables.
3. Identifica en cada problema las partes de la solución: entradas, procesamiento, salidas.
4. Genera el pseudocódigo o diagrama de flujo para dar una solución.
5. Desarrolla la solución en el lenguaje de programación seleccionado para el curso, utilizando el IDE de tu preferencia

6. Realiza capturas de pantalla que muestran la correcta ejecución de cada uno de los programas.
7. Integra en un documento el pseudocódigo, código y capturas de pantalla de las soluciones propuestas.
8. Entregar en formato PDF en la plataforma del curso.
9. Recibir retroalimentación docente.

Recursos de apoyo:

- IDE seleccionado para desarrollo.
- Computadora.
- Acceso a internet.

Unidad III: Estructuras de control y depuración de programas: sentencias, condicionales e iteraciones

Práctica 3: Condicionales simples y dobles

Duración:

03 horas

Procedimiento:

1. Atiende a la explicación de la persona docente sobre el uso de estructuras condicionales simples y dobles.
2. Analiza los problemas proporcionados en la plataforma del curso para identificar en qué casos se requiere tomar una decisión entre dos opciones.
3. Identifica en cada problema las partes de la solución: entradas, procesamiento, salidas.
4. Genera el pseudocódigo o diagrama de flujo para dar una solución.
5. Desarrolla la solución en el lenguaje de programación seleccionado para el curso, utilizando el IDE de tu preferencia.
6. Realiza capturas de pantalla que muestran la correcta ejecución de cada uno de los programas.
7. Integra en un documento el pseudocódigo, código y capturas de pantalla de las soluciones propuestas.
8. Entrega en formato PDF en la plataforma del curso.
9. Recibir retroalimentación docente.

Recursos de apoyo:

- IDE seleccionado para desarrollo.
- Computadora.
- Acceso a internet.

Práctica 4: Condicionales múltiples y anidados

Duración:

03 horas

Procedimiento:

1. Atiende a las indicaciones de la persona docente sobre condicionales múltiples y anidados.
2. Analiza los problemas proporcionados en la plataforma del curso para distinguir en qué casos se requiere evaluar varias decisiones o decisiones anidadas.
3. Para cada problema, identifica las partes de la solución: entradas, procesamiento, salidas.
4. Genera el pseudocódigo o diagrama de flujo para dar una solución.
5. Desarrolla la solución en el lenguaje de programación seleccionado para el curso, utilizando el IDE de tu preferencia.
6. Realiza capturas de pantalla que muestran la correcta ejecución de cada uno de los programas.
7. Integra en un documento el pseudocódigo, código y capturas de pantalla de las soluciones propuestas.
8. Entrega en formato PDF en la plataforma del curso.
9. Recibir retroalimentación docente.

Recursos de apoyo:

- IDE de programación,
- Computadora.
- Acceso a internet.

Práctica 5: Ciclos controlados por contador y centinela

Duración:

03 horas

Procedimiento:

1. Atiende a la explicación de la persona docente sobre el uso de ciclos controlados por contador y por centinela.
2. Analiza los problemas proporcionados en la plataforma del curso para distinguir cuándo es más adecuado aplicar cada tipo de ciclo.
3. Identifica en cada problema las partes de la solución: entradas, procesamiento, salidas.
4. Genera el pseudocódigo o diagrama de flujo para dar una solución.
5. Desarrolla la solución en el lenguaje de programación seleccionado para el curso, utilizando el IDE de tu preferencia.
6. Realiza capturas de pantalla que muestran la correcta ejecución de cada uno de los programas.
7. Integra en un documento el pseudocódigo, código y capturas de pantalla de las soluciones propuestas.
8. Entrega en formato PDF en la plataforma del curso.
9. Recibir retroalimentación docente.

Recursos de apoyo:

- IDE seleccionado para desarrollo.
- Computadora.
- Acceso a internet.

Práctica 6: Ciclos anidados y depuración**Duración:**

03 horas

Procedimiento:

1. Atiende a la explicación del docente sobre el uso de ciclos anidados y la importancia de la depuración de programas.
2. Analiza los problemas proporcionados en la plataforma del curso para identificar los casos en que se requieren ciclos dentro de ciclos.
3. Identifica en cada problema las partes de la solución: entradas, procesamiento, salidas.
4. Genera el pseudocódigo o diagrama de flujo para dar una solución.
5. Desarrolla la solución en el lenguaje de programación seleccionado para el curso, utilizando el IDE de tu preferencia.
6. Realiza capturas de pantalla que muestran la correcta ejecución de cada uno de los programas.
7. Integra en un documento el pseudocódigo, código y capturas de pantalla de las soluciones propuestas.
8. Entrega en formato PDF en la plataforma del curso.
9. Recibir retroalimentación docente.

Recursos de apoyo:

- IDE seleccionado para desarrollo.
- Computadora.
- Acceso a internet.

Unidad IV: Funciones**Práctica 7: Definición y uso de funciones****Duración:**

03 horas

Procedimiento:

1. Atiende a la explicación del docente sobre la definición, declaración y llamado de funciones.
2. Analiza los problemas proporcionados en la plataforma del curso para identificar cómo dividir la solución en funciones independientes.
3. Identifica en cada problema las partes de la solución: entradas, procesamiento, salidas.
4. Genera el pseudocódigo o diagrama de flujo para dar una solución.

5. Desarrolla la solución en el lenguaje de programación seleccionado para el curso, utilizando el IDE de tu preferencia.
6. Realiza capturas de pantalla que muestran la correcta ejecución de cada uno de los programas.
7. Integra en un documento el pseudocódigo, código y capturas de pantalla de las soluciones propuestas.
8. Entregar en formato PDF en la plataforma del curso.
9. Recibir retroalimentación docente.

Recursos de apoyo:

- IDE seleccionado para desarrollo.
- Computadora.
- Acceso a internet.

Práctica 8: Funciones con parámetros por valor

Duración:

03 horas

Procedimiento:

1. Atiende a la explicación del docente sobre el uso de parámetros por valor en funciones.
2. Analiza los problemas proporcionados en la plataforma del curso para distinguir en qué casos es útil pasar valores como argumentos.
3. Identifica en cada problema las partes de la solución: entradas, procesamiento, salidas.
4. Genera el pseudocódigo o diagrama de flujo para dar una solución.
5. Desarrolla la solución en el lenguaje de programación seleccionado para el curso, utilizando el IDE de tu preferencia.
6. Realiza capturas de pantalla que muestran la correcta ejecución de cada uno de los programas.
7. Integra en un documento el pseudocódigo, código y capturas de pantalla de las soluciones propuestas.
8. Entregar en formato PDF en la plataforma del curso.
9. Recibir retroalimentación docente.

Recursos de apoyo:

- IDE seleccionado para desarrollo.
- Computadora.
- Acceso a internet.

Práctica 9: Funciones con parámetros por referencia

Duración:

03 horas

Procedimiento:

1. Atiende a la explicación de la persona docente sobre el uso de parámetros por referencia en funciones.
2. Analiza los problemas proporcionados en la plataforma del curso para identificar casos en los que se requiere modificar los valores de las variables originales.
3. Identifica las partes de la solución: entradas, procesamiento, salidas de cada problema.
4. Genera el pseudocódigo o diagrama de flujo para dar una solución.
5. Desarrolla la solución en el lenguaje de programación seleccionado para el curso, utilizando el IDE de tu preferencia.
6. Realiza capturas de pantalla que muestran la correcta ejecución de cada uno de los programas.
7. Integra en un documento el pseudocódigo, código y capturas de pantalla de las soluciones propuestas.
8. Entrega en formato PDF en la plataforma del curso y recibe retroalimentación docente.

Recursos de apoyo:

- IDE seleccionado para desarrollo.
- Computadora.
- Acceso a internet.

Unidad V. Estructuras de datos**Práctica 10: Arreglos unidimensionales: recorrido y operaciones básicas****Duración**

03 horas

Procedimiento

1. Atiende a la explicación del docente sobre el concepto de arreglos.
2. Analiza los problemas proporcionados en la plataforma del curso para identificar situaciones que requieran almacenar y procesar datos en arreglos unidimensionales.
3. Identifica entradas, procesamiento y salidas en cada problema.
4. Elabora el pseudocódigo o diagrama de flujo correspondiente.
5. Desarrolla la solución en el IDE seleccionado, implementando operaciones de recorrido, búsqueda y cálculo.
6. Realiza capturas de pantalla de la ejecución de los programas.
7. Integra en un documento el pseudocódigo, código y capturas de pantalla de las soluciones propuestas.
8. Entrega en formato PDF en la plataforma del curso.
9. Recibir retroalimentación docente.

Recursos de apoyo:

- IDE seleccionado para desarrollo.
- Computadora.
- Acceso a internet.

Práctica 11: Arreglos multidimensionales: tablas y matrices

Duración

03 horas

Procedimiento

1. Atiende a la explicación del docente sobre el uso de arreglos multidimensionales.
2. Analiza los problemas proporcionados por el o la docente en la plataforma del curso para identificar situaciones que requieran arreglos bidimensionales.
3. Define entradas, procesamiento y salidas para cada problema.
4. Elabora pseudocódigo o diagramas de flujo antes de programar.
5. Implementa la solución en el IDE, realizando operaciones como sumas por filas/columnas o cálculo de transpuestas.
6. Realiza capturas de pantalla de la ejecución de los programas.
7. Integra en un documento el pseudocódigo, código y capturas de pantalla de las soluciones propuestas.
8. Entrega en formato PDF en la plataforma del curso.
9. Recibir retroalimentación docente.

Recursos de apoyo:

- IDE seleccionado para desarrollo.
- Computadora.
- Acceso a internet.

Práctica 12: Registros: definición y manipulación de datos compuestos

Duración

03 horas

Procedimiento

1. Atiende a la explicación del docente sobre el uso de registros.
2. Analiza los problemas proporcionados por el o la docente en la plataforma del curso para identificar datos que requieran agruparse en registros.
3. Define las entradas, procesamiento y salidas de cada caso.
4. Elabora pseudocódigo o diagramas de flujo que integren registros.
5. Implementa la solución en el IDE, capturando y mostrando información de los registros.
6. Realiza capturas de pantalla de la ejecución correcta.
7. Integra en un documento el pseudocódigo, código y capturas de pantalla de las soluciones propuestas.
8. Entrega en formato PDF en la plataforma del curso.
9. Recibir retroalimentación docente.

Recursos de apoyo:

- IDE seleccionado para desarrollo.
- Computadora y acceso a internet.

Práctica 13: Proyecto integrador con arreglos y registros

Duración

03 horas

Procedimiento

1. Atiende a la explicación de la persona docente sobre el uso de arreglos y registros.
2. Analiza el problema propuesto por el o la docente en la plataforma del curso para diseñar un sistema que combine el uso de arreglos y registros.
3. Identifica las entradas, procesamiento y salidas principales.
4. Elabora pseudocódigo o diagramas de flujo que reflejan la solución.
5. Desarrolla e implementa el programa en el IDE, incluyendo funciones de alta, búsqueda y consulta.
6. Captura la ejecución del sistema y documenta los resultados.
7. Integra en un documento el pseudocódigo, código y capturas de pantalla de las soluciones propuestas.
8. Entrega en formato PDF en la plataforma del curso.
9. Recibir retroalimentación docente.

Recursos de apoyo:

- IDE seleccionado para desarrollo.
- Computadora.
- Acceso a internet

Unidad VI. Tópicos avanzados de programación

Práctica 14: Recursividad: problemas clásicos

Duración

03 horas

Procedimiento

1. Atiende a la explicación de la persona docente sobre funciones recursivas.
2. Analiza los problemas proporcionados por el o la docente en la plataforma del curso para identificar casos en los que se pueda aplicar recursividad.
3. Define entradas, procesamiento y salidas en cada caso.
4. Elabora pseudocódigo o diagramas de flujo de las soluciones recursivas.
5. Desarrolla funciones recursivas en el IDE (ej. factorial, Fibonacci, Torres de Hanói).
6. Realiza capturas de pantalla de la ejecución correcta.
7. Integra en un documento el pseudocódigo, código y capturas de pantalla de las soluciones propuestas.
8. Entrega en formato PDF en la plataforma del curso.
9. Recibir retroalimentación docente.

Recursos de apoyo:

- IDE seleccionado para desarrollo.
- Computadora.
- Acceso a internet

Práctica 15: Manejo de archivos: lectura y escritura de datos**Duración**

03 horas

Procedimiento

1. Atiende a la explicación de la persona docente sobre manejo de archivos.
2. Analiza los problemas proporcionados por el o la docente en la plataforma del curso para identificar casos que requieran persistencia de datos en archivos.
3. Establece entradas, procesamiento y salidas para cada caso.
4. Elabora pseudocódigo o diagramas de flujo que incluya operaciones de archivo.
5. Implementa programas en el IDE para escribir y leer archivos de texto.
6. Realiza capturas de pantalla que evidencien su ejecución.
7. Integra en un documento el pseudocódigo, código y capturas de pantalla de las soluciones propuestas.
8. Entrega en formato PDF en la plataforma del curso.
9. Recibir retroalimentación docente.

Recursos de apoyo:

- IDE seleccionado para desarrollo.
- Computadora.
- Acceso a internet

Práctica 16: Manejo de memoria dinámica**Duración**

03 horas

Procedimiento

1. Atiende a la explicación de la persona docente sobre el uso de memoria dinámica.
2. Analiza los problemas proporcionados por el o la docente en la plataforma del curso para identificar casos en los que se requiera el uso de memoria dinámica.
3. Identifica las partes de la solución: entradas, procesamiento y salidas de cada problema.
4. Genera el pseudocódigo o diagrama de flujo que muestre la gestión de variables dinámicas.
5. Desarrolla la solución en el lenguaje de programación seleccionado para el curso, utilizando el IDE de tu preferencia.
6. Realiza capturas de pantalla que evidencien la correcta ejecución de cada programa.
7. Integra en un documento el pseudocódigo, código y capturas de pantalla de las soluciones propuestas.

8. Entrega el documento en formato PDF en la plataforma del curso.
9. Recibir retroalimentación docente.

Recursos de apoyo:

- IDE seleccionado para desarrollo.
- Computadora.
- Acceso a internet.

X. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer de forma clara y formal:

- La metodología de trabajo.
- Criterios de evaluación y rúbricas.
- Reglas sobre la calidad y entrega de trabajos académicos.
- Derechos y obligaciones del docente y los estudiantes.

Esto debe quedar registrado y comunicado por medios institucionales (como Blackboard o correo institucional).

Estrategia de enseñanza

Son acciones didácticas planificadas por el docente para facilitar el logro de los aprendizajes esperados. Estas estrategias deben seleccionarse en función del tipo de competencia, el nivel cognitivo y la naturaleza del contenido.

Estos son ejemplos de estrategias, puede usar algunas de esta lista u otras que se consideren pertinentes:

- Aprendizaje basado en problemas
- Técnica expositiva
- Método de proyectos

Estrategia de aprendizaje

Son enfoques que el estudiante emplea para organizar, adquirir, comprender y aplicar el conocimiento. Estas estrategias deben promover el desarrollo de competencias, el pensamiento crítico y el aprendizaje significativo.

Estos son ejemplos de estrategias, puede usar algunas de esta lista u otras que se consideren pertinentes:

- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Ejercicios con pseudocódigos
- Diagramas de flujo
- Análisis de problemas científicos

XI. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario, cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	30%
– Tareas y prácticas de laboratorio	30%
– Proyecto integrador	40%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Deitel, P. J., & Deitel, H. (2021). Cómo programar en C++. Pearson Hispanoamérica.
<https://uabc.vitalsource.com/books/9786073255424>

Joyanes Aguilar, L. (2006). Programación en C++: Un enfoque práctico. McGraw-Hill España.
<https://libcon.rec.uabc.mx:2321/es/ereader/uabc/50086?page=10>

Joyanes Aguilar, L. (2015). Fundamentos de programación: Algoritmos, estructuras de datos y objetos (4.ª ed.). McGraw-Hill.

Rodríguez-Losada González, D., Muñoz Cano, J., & García Cena, C. (2022). Introducción a la programación en C (1.ª ed.). UPM Press.
<https://libcon.rec.uabc.mx:2321/es/ereader/uabc/256906?page=1>

Sznajdleder, P. (2017). Programación estructurada a fondo: Implementación de algoritmos en C. Alpha Editorial.

Recursos de acceso libre o complementarios externos

Benjumea, V., & Roldán, M. (2022). *Fundamentos de programación con el lenguaje de programación C+*. Universidad de Málaga.

http://www.lcc.uma.es/~vicente/docencia/cppdoc/programacion_cxx.pdf

Seacord, R. C. (2024). *Effective C: An introduction to professional C programming*. No Starch Press.

Stroustrup, B. (2024). *Programming: Principles and practice using C++* (2.^a ed.). Addison-Wesley Professional.

XII. PERFIL DOCENTE

La persona docente de esta asignatura deberá contar con formación profesional en el área de computación o en disciplinas afines. Asimismo, tendrá que demostrar experiencia en docencia, así como en lógica programática, análisis, diseño e implementación de programas de computadora, y dominio de lenguajes de programación basados en el paradigma estructurado. Se requiere que cuente con habilidades para el manejo de grupos y que sea una persona responsable, comprometida y persistente en el acompañamiento del aprendizaje del estudiantado.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada.
Programa educativo:	Licenciatura en Física Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Cálculo Integral
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 04 Horas Laboratorio: 00 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos: 08
Etapas de formación:	Básica
Carácter:	Obligatoria
Requisito:	Cálculo Diferencial
Modalidad de impartición:	Presencial
Equipo de diseño:	Juan Crisóstomo Tapia Mercado
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La unidad de aprendizaje de Cálculo Integral es fundamental en la etapa básica de los programas ciencias exactas porque representa uno de los conceptos más importantes del análisis matemático; en combinación con la derivada permite entender el Teorema Fundamental del Cálculo y resolver una amplia variedad de problemas que involucran cambios. Su comprensión y manejo constituyen un preámbulo en el aprendizaje del Cálculo de Varias Variables e incursionar en áreas más avanzadas de las matemáticas como son: Ecuaciones Diferenciales Ordinarias, Probabilidad y Métodos Matemáticos de la Física. La unidad de aprendizaje pertenece al plan de estudios de Licenciatura en Física, se ubica en la etapa básica, es de carácter obligatoria y pertenece al área de conocimiento de Matemáticas, es deseable haber cursado la unidad de aprendizaje de Cálculo Diferencial; además se comparte con el plan de estudios de Licenciatura en Ciencias Computacionales, se ubica en etapa básica con carácter obligatorio.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Analizar conceptos del Cálculo Integral y las propiedades de la integral definida e indefinida mediante técnicas analíticas y numéricas para resolver integrales de funciones de una variable real y problemas relacionados con la tecnología, ciencias naturales y exactas con disposición al trabajo colaborativo, respeto y responsabilidad.

IV. EVIDENCIA DE APRENDIZAJE

Portafolio integrador

El portafolio debe contener la resolución de problemas y ejercicios con el desglose detallado del proceso y resultados del manejo de los métodos de solución de los diferentes tipos de integrales, se debe considerar lo siguiente para la integración del portafolio:

- Los conceptos fundamentales involucrados en el análisis cuantitativo y cualitativo
- Aplicar los métodos de integración apropiados a situaciones reales, cómo el cálculo de áreas y volúmenes de sólidos de revolución,
- Calcular sumas infinitas examinando el comportamiento de los términos y aplicando los criterios de convergencia típicos, mientras colabora proactivamente con sus compañeros.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Licenciatura en Física

Explorar fenómenos físicos mediante sus modelos matemáticos, para describir y predecir el comportamiento de la materia, la energía y sus interacciones, con disciplina y rigor científico.

Licenciatura en Ciencias Computacionales

Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia e Inclusión.

VII. DESARROLLO POR UNIDADES

Unidad I. Integral de Riemann

Competencia

Manipular la integral de Riemann mediante el cálculo de integrales, para resolver problemas de diferentes áreas de ingeniería y ciencias, con actitud analítica, disposición al trabajo en equipo y responsabilidad.

Contenido

1.1. Integral de Riemann

- 1.1.1. Antiderivada
- 1.1.2. Algunas técnicas de antiderivación
- 1.1.3. Aplicaciones a ecuaciones diferenciales y movimiento rectilíneo
- 1.1.4. Definición axiomática de área
- 1.1.5. Sumas superiores e inferiores.
- 1.1.6. La integral definida
- 1.1.7. Teorema del valor medio para integrales
- 1.1.8. Teoremas fundamentales del cálculo
- 1.1.9. Integral indefinida
- 1.1.10. Integración de funciones elementales

Duración

10 horas

Unidad II. Métodos de integración

Competencia

Aplicar los diferentes métodos de integración a través de la identificación de los diferentes tipos de integrandos para resolver integrales de funciones elementales, con actitud ordenada, reflexiva y responsable.

Contenido

2.1. Métodos de integración

- 2.1.1. Cambio de variable, integración por sustitución
- 2.1.2. Integración por partes
- 2.1.3. Sustitución Trigonométrica
- 2.1.4. Función logaritmo natural y su inversa: integración logarítmica y exponencial
- 2.1.5. Integración de funciones racionales: fracciones parciales

Duración

08 horas

Unidad III. Aplicaciones de la integral

Competencia

Aplicar el concepto de integral y su significado geométrico mediante la determinación de secciones geométricas y su manipulación en el plano o el espacio para calcular áreas, perímetros de superficies irregulares, así como volúmenes de sólidos de revolución, con orden, compromiso y tenacidad.

Contenido

3.1. Aplicaciones de la integral

- 3.1.1. Cálculo del área de una región plana
- 3.1.2. Cálculo de volúmenes de sólidos mediante el método de rebanado
- 3.1.3. Cálculo de volúmenes de sólidos mediante el método de discos
- 3.1.4. Cálculo de volúmenes de sólidos mediante el método de arandelas
- 3.1.5. Cálculo de volúmenes de sólidos mediante el método de capas cilíndricas
- 3.1.6. Cálculo de longitudes de arco: teorema de Pappus
- 3.1.7. La integral impropia: definición, clasificación, convergencia
- 3.1.8. Problemas de aplicación a las ciencias exactas

Duración

08 horas

Unidad IV. Sucesiones y Series

Competencia

Emplear series numéricas y de potencias a través de examinar el comportamiento de los términos y la aplicación de los criterios de convergencia típicos para calcular sumas infinitas o descartar su existencia esto con una actitud participativa, perseverante, paciencia y tolerancia.

Contenido

4.1. Series de Taylor

- 4.1.1. Aproximaciones polinomiales mediante la Fórmula de Taylor
- 4.1.2. Sucesiones
- 4.1.3. Series infinitas de términos constantes, positivos y negativos

4.1.4. Criterios de convergencia y divergencia

4.1.5. Series de potencias

Duración

06 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

Unidad I: Integral de riemann

Práctica 1: Exploración gráfica y numérica del concepto de área y sumas de Riemann

Duración

04 horas

Procedimiento

1. Atiende la explicación guiada de su docente sobre la relación entre área, sumas y la integral.
2. Grafica en software (GeoGebra/Desmos) funciones sencillas como la función exponencial, la parábola y la función seno.
3. Construye sumas inferiores y superiores para intervalos de diferente partición.
4. Calcula numéricamente las sumas de Riemann izquierda, derecha y punto medio.
5. Analiza el comportamiento al refinar la partición.
6. Elabora una tabla comparativa entre los valores aproximados y el valor exacto cuando sea posible.
7. Redacta conclusiones sobre convergencia y significado geométrico.
8. Entrega el reporte con resultados de los ejercicios realizados y recibe retroalimentación docente.

Recursos de apoyo

- GeoGebra o Desmos.
- Calculadora científica.
- Presentación del docente.
- Apuntes de clase.

Práctica 2: Antiderivadas y técnicas básicas de integración

Duración

04 horas

Procedimiento

1. Atiende indicaciones de su docente sobre la práctica a realizar.

2. Revisa la definición de antiderivada y su relación con la integral indefinida.
3. Identifica familias de funciones que comparten derivadas.
4. Resuelve integrales básicas con reglas directas: potencias, trigonométricas, exponenciales.
5. Verifica resultados derivando tus antiderivadas.
6. Compara métodos cuando existan dos formas correctas.
7. Realiza una actividad colaborativa para clasificar integrales según la técnica apropiada.
8. Entrega tu reporte con ejercicios seleccionados y recibe retroalimentación docente.

Recursos de apoyo

- Tabla de derivadas e integrales.
- Calculadora simbólica (WolframAlpha opcional).
- Notas de clase.

Práctica 3: Integral definida y teorema fundamental del cálculo

Duración

04 horas

Procedimiento

1. Atiende indicaciones de su docente sobre la práctica a realizar.
2. Repasa el enunciado de los Teoremas Fundamentales del Cálculo vistas en clase.
3. Evalúa integrales definidas usando antiderivadas.
4. Construye gráficas de funciones para interpretar el signo de la integral.
5. Analiza integrales en intervalos invertidos y propiedades de simetría.
6. Evalúa integrales donde el integrando cambia de signo.
7. Escribe una reflexión sobre el significado físico de área neta y área total, y recibe retroalimentación docente.

Recursos de apoyo

- GeoGebra / Desmos.
- Tablas de integrales.
- Apuntes del TFC.

Práctica 4: Sumas superiores e inferiores y definición axiomática de área

Duración

04 horas

Procedimiento

1. Atiende indicaciones de su docente sobre la práctica a realizar.
2. Revisa las propiedades axiomáticas del área.
3. Plantea sumas superiores e inferiores para funciones no monótonas.
4. Genera particiones no uniformes en el intervalo asignado.
5. Determina cuándo ambas sumas convergen al mismo valor.
6. Contrasta el resultado con la integral definida.
7. Redacta un análisis final sobre cuándo una función es integrable.
8. Entrega tu reporte con ejercicios seleccionados y recibe retroalimentación docente.

Recursos de apoyo

- Cuaderno cuadriculado.
- GeoGebra.
- Ejemplos proporcionados.

Unidad II: Métodos de integración

Práctica 5: Integración por sustitución (cambio de variable)

Duración

04 horas

Procedimiento

1. Atiende indicaciones de su docente sobre la práctica a realizar.
2. Revisa sustituciones directas y trigonométricas básicas.
3. Identifica integrales donde la composición de funciones lo sugiera.
4. Resuelve integrales que requieran manipulación algebraica previa.
5. Verifica la corrección reconstruyendo el diferencial.
6. Resuelve ejercicios avanzados propuestos por el docente.
7. Entrega un reporte con ejercicios seleccionados y recibe retroalimentación docente.

Recursos de apoyo

- Notas de métodos de integración.
- Calculadora.
- Software simbólico opcional.

Práctica 6: Integración por partes

Duración

04 horas

Procedimiento

1. Atiende indicaciones de su docente sobre la práctica a realizar.
2. Revisa la deducción de la fórmula desde la regla del producto vista en clase.
3. Selecciona la función $u(x)$ y el diferencial dv .
4. Resuelve integrales algebraicas, trigonométricas y logarítmicas
5. Aplica integración por partes repetida.
6. Resuelve integrales que regresan a la original.
7. Documenta el procedimiento paso a paso.
8. Entrega un reporte con ejercicios seleccionados y recibe retroalimentación docente.

Recursos de apoyo

- Ejemplos resueltos.
- GeoGebra.

Práctica 7: Sustitución trigonométrica

Duración

04 horas

Procedimiento

1. Atiende indicaciones de su docente sobre la práctica a realizar.
2. Repasa identidades trigonométricas fundamentales vistas en clase.
3. Identifica las formas estándar que requieren sustituciones $x = a \sin \theta$, $x = a \tan \theta$, $x = a \sec \theta$.
4. Realiza el trazo geométrico que justifica cada sustitución.
5. Resuelve integrales con raíces cuadradas de expresiones cuadráticas.
6. Regresa a la variable original mediante triángulos de referencia.
7. Elabora un reporte con diagramas.
8. Entrega un reporte con ejercicios seleccionados y recibe retroalimentación docente.

Recursos de apoyo

- Tabla de identidades trigonométricas.
- Notas del docente.

Práctica 8: Integración racional y fracciones parciales

Duración

04 horas

Procedimiento

1. Atiende indicaciones de su docente sobre la práctica a realizar.
2. Revisa la clasificación de denominadores: lineales, cuadráticos e irreducibles vistas en clase.
3. Descompón integrales racionales en fracciones parciales.
4. Ajusta coeficientes mediante igualación o comparación.
5. Resuelve integrales resultantes usando técnicas previas.
6. Verifica la respuesta mediante derivación.
7. Entrega un reporte con ejercicios seleccionados y recibe retroalimentación docente.

Recursos de apoyo

- Tabla de fracciones parciales.
- Calculadora algebraica.
- Software CAS (opcional).

Unidad III: Aplicaciones de la integral

Práctica 9: Cálculo de áreas entre curvas

Duración

04 horas

Procedimiento

1. Atiende indicaciones de su docente sobre la práctica a realizar.
2. Representa gráficamente las curvas asignadas.
3. Identifica puntos de intersección.
4. Define correctamente la función superior e inferior en cada intervalo.
5. Plantea la integral del área y calcula el área.
6. Analiza simetrías para simplificar el cálculo.
7. Entrega un reporte con ejercicios seleccionados y recibe retroalimentación docente.

Recursos de apoyo

- GeoGebra.
- Apuntes del tema.

Práctica 10: Volúmenes por rebanado y por discos

Duración

04 horas

Procedimiento

1. Atiende indicaciones de su docente sobre la práctica a realizar.
2. Diferencia los métodos de rebanado y discos vistos en clase.
3. Modela la región base y el eje de revolución.
4. Plantea la integral del volumen en cada método.
5. Resuelve problemas con funciones polinómicas y trigonométricas.
6. Interpreta físicamente el significado del volumen de revolución.
7. Entrega un reporte con ejercicios seleccionados y recibe retroalimentación docente.

Recursos de apoyo

- GeoGebra 3D
- Tablas de integrales

Práctica 11: Método de arandelas y capas cilíndricas

Duración

04 horas

Procedimiento

1. Atiende indicaciones de su docente sobre la práctica a realizar.
2. Revisa los esquemas geométricos de cada método visto en clase.
3. Compara las ventajas de usar arandelas vs. capas.
4. Plantea integrales de volumen para ejes de revolución desplazados.
5. Realiza un ejercicio mixto donde se compare el resultado por ambos métodos.
6. Analiza errores típicos.
7. Entrega un reporte con ejercicios seleccionados y recibe retroalimentación docente.

Recursos de apoyo

- GeoGebra.
- Guía de ejercicios.

Práctica 12: Integrales impropias en aplicaciones físicas

Duración

04 horas

Procedimiento

1. Atiende indicaciones de su docente sobre la práctica a realizar.
2. Revisa la clasificación de integrales impropias: tipo I y II vistas en clase.
3. Aplica el límite para definir la integral.
4. Resuelve problemas de probabilidad, decaimiento, trabajo o distancia infinita.
5. Determina convergencia o divergencia.
6. Presenta un reporte con las conclusiones físicas finales en grupo para su revisión y retroalimentación docente.
7. Entrega reporte final.

Recursos de apoyo

- Apuntes del docente.
- Tabla de límites.

Unidad VI: Sucesiones y Series

Práctica 13: Sucesiones: comportamiento y límites

Duración

04 horas

Procedimiento

1. Atiende indicaciones de su docente sobre la práctica a realizar.
2. Define explícitamente cada sucesión asignada.
3. Evalúa términos iniciales y construye tablas.
4. Identifica si es monótona, acotada o divergente.
5. Determina el límite cuando exista.
6. Representa puntos en el plano (n, a_n) .
7. Concluye sobre la convergencia o divergencia y presenta los argumentos analíticos de manera individual o grupal y recibe retroalimentación docente.
8. Entrega reporte final.

Recursos de apoyo

- GeoGebra.
- Tablas de límites.

Práctica 14: Series numéricas: pruebas de convergencia

Duración

04 horas

Procedimiento

1. Atiende indicaciones de su docente sobre la práctica a realizar.
2. Revisa criterios: razón, raíz, comparación, integral vistos en clase.
3. Analiza la serie asignada y selecciona el criterio adecuado.
4. Realiza el cálculo justificando cada paso.
5. Clasifica la serie como convergente/divergente.
6. Analiza series alternantes y su convergencia condicional.
7. Presenta frente a grupo los resultados de los ejercicios realizados y recibe retroalimentación docente.
8. Entrega reporte final.

Recursos de apoyo

- GeoGebra CAS.
- Apuntes del docente.
- Tabla de derivadas/integrales.

Práctica 15: Series de potencias

Duración

04 horas

Procedimiento

1. Atiende indicaciones de su docente sobre la práctica a realizar.
2. Revisa la definición formal de una serie de potencias y su centro.
3. Calcula el radio de convergencia mediante pruebas estándar.
4. Analiza el intervalo de convergencia evaluando extremos.
5. Comprueba la convergencia mediante ejemplos gráficos.
6. Discute de manera grupal las aplicaciones físicas y recibe retroalimentación docente.
7. Entrega reporte final.

Recursos de apoyo

- GeoGebra CAS.
- Apuntes del docente.
- Tabla de derivadas/integrales.

Práctica 16: Series de Taylor y aproximaciones polinomiales

Duración

04 horas

Procedimiento

1. Atiende indicaciones de su docente sobre la práctica a realizar.
2. Analiza la fórmula general de Taylor vista en clase.
3. Calcula derivadas sucesivas de la función real que será aproximada por polinomios.
4. Obtén el polinomio de Taylor de orden n .
5. Compara gráficamente la aproximación con la función real.
6. Evalúa errores locales y globales.
7. Presenta frente a grupo las aplicaciones en física o ingeniería y recibe retroalimentación docente.
8. Entrega reporte final.

Recursos de apoyo

- GeoGebra / Desmos
- Apuntes de derivación
- Software simbólico opcional

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer de forma clara y formal:

- La metodología de trabajo.
- Criterios de evaluación y rúbricas.
- Reglas sobre la calidad y entrega de trabajos académicos.
- Derechos y obligaciones del docente y los estudiantes.

Esto debe quedar registrado y comunicado por medios institucionales (como Blackboard o correo institucional).

Estrategia de enseñanza

- Método de proyectos
- Aprendizaje basado en problemas
- Técnica expositiva
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

● Evaluaciones parciales	40%
● Participación y tareas	10%
● Portafolio integrador	50%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Apostol, T. (2005). *Calculus*, vol. 1. Ed. Reverté. [clásica]

Boyce, W.E. y DiPrima, R. C. (s.f.). *Cálculo*. Ed. CECSA.

Larson, R. E. y Edwards, B. (2023). *Cálculo*. CENGAGE.

Leithold, L. (1994). *El Cálculo*. Oxford. [clásica]

Stewart J., Clegg, D. Watson, S. (2021). *Cálculo de una variable: trascendentes tempranas*. CENGAGE.

Spivak, M. (1998). *Calculus*. Ed. Reverté. [clásica]

Thomas, G. (2010). *Cálculo*. Ed. Pearson. [clásica]

Recursos de acceso libre o complementarios externos

Aguilar, A. (2016). *Cálculo diferencial e integral*. Mc Graw Hill. [clásica]

Fuenlabrada, S. (2013). *Cálculo integral*. Pearson. [clásica]

Oteyza, E., Lam, E., Hernández, C y Carrillo, A. (2019). *Cálculo diferencial e integral*. Pearson.

René, M., Estrada. R. (2020). *Cálculo integral*. Pearson.

Swokowski, E. A. (1982). *Cálculo con Geometría analítica*. Grupo Editorial Iberoamérica. [clásica]

XII. PERFIL DOCENTE

La persona docente que imparta la unidad de aprendizaje de Cálculo Integral debe contar con título de Licenciatura en Física, Matemáticas o área afín, con conocimientos de los diferentes métodos de solución de integrales, cálculo de áreas y volúmenes de sólidos de revolución, así como series y su convergencia, utilizando procedimientos analíticos y numéricos, para aplicarlas en la solución de problemas de las ciencias naturales y exactas, preferentemente con estudios de posgrado y experiencia docente mínima de un año. Ser una persona analítica, respetuosa, responsable y proactiva, contar con facilidad de expresión y fomentar la participación individual y en equipo.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Física Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Álgebra Lineal
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 04 Horas Laboratorio: 00 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos: 08
Etapas de formación:	Básica
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Juan Crisóstomo Tapia Mercado Ramón Carrillo Bastos Jorge Alberto Villavicencio Aguilar
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

El propósito es que el estudiantado aplique los conceptos y propiedades básicas de espacios vectoriales, transformaciones lineales, matrices, sistemas de ecuaciones lineales, determinantes, valores y vectores propios, que constituyen la base del álgebra lineal. Estos contenidos sustentan el desarrollo de la propia disciplina y de otras áreas de las ciencias naturales, exactas, computacionales y de datos. Pertenece al plan de estudios de Licenciatura en Física, es de carácter obligatorio y se ubica en la etapa básica de la formación y pertenece al área de conocimiento de Matemáticas; además se comparte con la Licenciatura en Ciencias Computacionales se ubica en etapa básica con carácter obligatorio.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Aplicar los conceptos básicos del álgebra lineal, mediante el estudio y uso de espacios vectoriales, matrices, sistemas de ecuaciones lineales, transformaciones lineales, determinantes y valores y vectores propios, para modelar y resolver problemas sencillos de la propia disciplina y de otras áreas de las ciencias, con rigor, iniciativa y disposición al trabajo en equipo.

IV. EVIDENCIA DE APRENDIZAJE

Resolución de problemas teóricos y realización de prácticas de taller y computacionales, con desglose claro del procedimiento y de los resultados obtenidos.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Licenciatura en Física

Explorar fenómenos físicos mediante sus modelos matemáticos, para describir y predecir el comportamiento de la materia, la energía y sus interacciones, con disciplina y rigor científico.

Licenciatura en Ciencias Computacionales

Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Pensamiento Crítico.

VII. DESARROLLO POR UNIDADES

Unidad I. Espacios vectoriales

Competencia

Emplear el concepto de espacio vectorial sobre $\mathbb{R}$ y sus propiedades, mediante la descripción axiomática y resultados básicos, para modelar y resolver problemas de las ciencias, con iniciativa, rigor y pensamiento crítico.

Contenido

1.1. Introducción

1.1.1. Motivación y ejemplos en $\mathbb{R}^n$ (geometría en el plano y en el espacio)

1.2. Definición y propiedades de espacio vectorial sobre $\mathbb{R}$

1.3. Subespacios vectoriales: criterio, propiedades y operaciones (suma e intersección)

1.4. Combinaciones lineales y conjuntos generadores

1.5. Dependencia e independencia lineal

1.6. Bases y dimensión

1.7. Coordenadas de un vector respecto a una base

1.8. Cambio de base en dimensión baja como reinterpretación de coordenadas en distintas bases (sin lenguaje matricial formal)

Duración

06 horas

Unidad II. Matrices y sistemas de ecuaciones lineales

Competencia

Resolver sistemas de ecuaciones lineales mediante técnicas matriciales básicas que comprenden la expresión del sistema en la forma $Ax = b$, la reducción por renglones y el uso elemental de determinantes en casos de baja dimensión, para interpretar los resultados en términos de existencia y unicidad de soluciones en problemas sencillos de las ciencias, con actitud ordenada, colaborativa y responsable.

Contenido

2.1. Sistemas de ecuaciones lineales y forma matricial

2.1.1. Formulación de sistemas lineales en $\mathbb{R}^2$ y $\mathbb{R}^3$

2.1.2. Matriz de coeficientes, vector de incógnitas y vector de términos independientes

2.1.3. Escritura de un sistema en la forma $Ax = b$

2.2. Matriz ampliada y reducción por renglones

2.2.1. Matriz ampliada de un sistema lineal

2.2.2. Operaciones elementales por renglones

- 2.2.3. Forma escalonada (y, en casos sencillos, forma escalonada reducida)
- 2.2.4. Resolución de sistemas lineales mediante reducción por renglones (método de eliminación Gaussiana)
- 2.3. Rango y clasificación de sistemas
 - 2.3.1. Idea de rango de una matriz a partir de su forma escalonada
 - 2.3.2. Criterios de existencia y unicidad de solución según los rangos de A y $[A|b]$
 - 2.3.3. Descripción del conjunto solución (única solución, infinitas soluciones, sin solución)
- 2.4. Determinantes básicos
 - 2.4.1. Definición y cálculo del determinante de matrices 2×2
 - 2.4.2. Cálculo del determinante de matrices 3×3 en casos sencillos
 - 2.4.3. Criterio $\det(A) \neq 0 \Leftrightarrow A$ es invertible (enunciado y ejemplos simples)
 - 2.4.4. Regla de Cramer para sistemas de ecuaciones lineales 2×2 y 3×3 : enunciado, uso en ejemplos concretos y discusión de sus limitaciones frente a métodos más generales.
- 2.5. Contenidos opcionales (si el tiempo lo permite)
 - 2.5.1. Interpretación geométrica elemental del determinante como factor de escala de áreas en $\mathbb{R}^2$
 - 2.5.2. Ejemplos muy sencillos de modelación de problemas mediante sistemas lineales $Ax = b$

Duración

08 horas

Unidad III. Transformaciones lineales y representación matricial

Competencia

Emplear el concepto de transformación lineal entre espacios vectoriales reales de dimensión finita, mediante su representación matricial en la base estándar y el análisis de su núcleo e imagen, para plantear, analizar y resolver problemas sencillos de las ciencias, con actitud analítica, rigurosa y propositiva.

Contenido

- 3.1. Definición y ejemplos de transformaciones lineales
 - 3.1.1. Transformaciones lineales $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$ y $T: \mathbb{R}^3 \rightarrow \mathbb{R}^3$ (escalamientos, proyecciones y simetrías sencillas) como ejemplos geométricos
 - 3.1.2. Transformaciones que no son lineales (ejemplos y contraejemplos)
- 3.2. Núcleo e imagen de una transformación lineal
 - 3.2.1. Definición de $\ker(T)$ e $\text{im}(T)$ en espacios de dimensión finita
 - 3.2.2. Interpretación de $\ker(T)$ como conjunto de soluciones de $T(x) = 0$
 - 3.2.3. Interpretación de $\text{im}(T)$ como conjunto de todos los valores posibles $T(x)$
- 3.3. Teorema de la dimensión (versión básica)
 - 3.3.1. Relación $\dim(\text{dom}(T)) = \dim(\ker(T)) + \dim(\text{im}(T))$ en ejemplos sencillos de $\mathbb{R}^2$ y $\mathbb{R}^3$

- 3.3.2. Conexión con la clasificación de sistemas lineales estudiada en la Unidad II
- 3.4. Representación matricial en la base estándar
 - 3.4.1. Matriz asociada a una transformación lineal $T: \mathbb{R}^n \rightarrow \mathbb{R}^n$ respecto a la base estándar
 - 3.4.2. Cálculo de la matriz de T a partir de $T(e_1), T(e_2), \dots, T(e_n)$
 - 3.4.3. Uso de la matriz asociada para calcular $T(x)$ mediante multiplicación de matrices
- 3.5. Interpretación de espacios columna y nulo
 - 3.5.1. Espacio columna de una matriz como imagen de la transformación lineal asociada
 - 3.5.2. Espacio nulo de una matriz como núcleo de la transformación lineal asociada
- 3.6. Contenidos opcionales (si el tiempo lo permite)
 - 3.6.1. Inyectividad y sobreyectividad de transformaciones lineales en $\mathbb{R}^n$
 - 3.6.2. Isomorfismos lineales entre espacios de la misma dimensión (idea intuitiva)
 - 3.6.3. Composición de transformaciones lineales y su efecto en la matriz asociada

Duración

06 horas

Unidad IV. Productos escalares y ortogonalidad

Competencia

Emplear el concepto de producto escalar en $\mathbb{R}^n$ y sus propiedades básicas, mediante el cálculo de normas, distancias, ángulos, relaciones de ortogonalidad y la construcción de bases ortonormales sencillas, para resolver problemas elementales de geometría y proyecciones en espacios euclidianos, con rigor, claridad y disposición al trabajo colaborativo.

Contenido

- 4.1. Producto escalar en $\mathbb{R}^n$
 - 4.1.1. Definición de producto escalar usual en $\mathbb{R}^n$
 - 4.1.2. Propiedades básicas (bilinealidad, simetría, positividad)
 - 4.1.3. Norma de un vector y distancia entre vectores inducidas por el producto escalar
 - 4.1.4. Ángulo entre vectores en $\mathbb{R}^2$ y $\mathbb{R}^3$ (interpretación geométrica)
- 4.2. Ortogonalidad y bases ortonormales
 - 4.2.1. Vectores ortogonales y conjuntos ortogonales
 - 4.2.2. Bases ortonormales en $\mathbb{R}^2$ y $\mathbb{R}^3$
 - 4.2.3. Ventajas de trabajar con bases ortonormales (cálculo sencillo de coordenadas)
- 4.3. Proyecciones ortogonales sencillas
 - 4.3.1. Proyección ortogonal de un vector sobre una recta en $\mathbb{R}^2$ y $\mathbb{R}^3$
 - 4.3.2. Descomposición de un vector como suma de componente paralela y perpendicular a una recta
- 4.4. Ortonormalización de Gram–Schmidt
 - 4.4.1. Idea del procedimiento de Gram–Schmidt para construir una base ortonormal

- a partir de dos o tres vectores linealmente independientes en $\mathbb{R}^2$ y $\mathbb{R}^3$
- 4.4.2. Aplicaciones sencillas: obtención de una base ortonormal a partir de una base cualquiera en $\mathbb{R}^2$ o $\mathbb{R}^3$
- 4.5. Extensiones y aplicaciones adicionales (si el tiempo lo permite)
 - 4.5.1. Proyección ortogonal de un vector sobre un plano generado por dos vectores ortogonales
 - 4.5.2. Matrices ortogonales básicas en $\mathbb{R}^2$ (rotaciones y reflexiones) como ejemplos de transformaciones que preservan el producto escalar

Duración

06 horas

Unidad V. Conceptos fundamentales de valores y vectores propios

Competencia

Usar las propiedades básicas de los valores y vectores propios de matrices reales de tamaño pequeño, mediante su definición y resultados elementales, para analizar situaciones sencillas de la misma disciplina y de las ciencias exactas, con actitud propositiva, crítica y responsable.

Contenido

- 5.1. Definiciones básicas
 - 5.1.1. Autovalores (valores propios) y autovectores (vectores propios) de una matriz cuadrada
 - 5.1.2. Ejemplos en matrices 2×2 y 3×3 sencillas
- 5.2. Polinomio característico y ecuación característica
 - 5.2.1. Polinomio característico de matrices 2×2 y 3×3 en casos simples
 - 5.2.2. Ecuación característica y cálculo de valores propios en ejemplos concretos
- 5.3. Vectores propios, espacios propios y sus bases
 - 5.3.1. Cálculo de vectores propios asociados a un valor propio dado
 - 5.3.2. Espacio propio como subespacio asociado a un valor propio
 - 5.3.3. Bases de espacios propios en ejemplos sencillos
- 5.4. Diagonalización básica
 - 5.4.1. Criterio práctico de diagonalización en dimensión pequeña (existencia de n autovectores linealmente independientes)
 - 5.4.2. Ejemplos de matrices diagonalizables y no diagonalizables
 - 5.4.3. Interpretación de la diagonalización en términos de cambios de base sencillos
- 5.5. Contenidos opcionales (si el tiempo lo permite)
 - 5.5.1. Enunciado del teorema espectral para matrices simétricas reales (sin demostración)

5.5.2. Comentarios sobre aplicaciones a modos normales y sistemas acoplados simples

Duración

06 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

Unidad I. Espacios vectoriales

Práctica 1: Espacios vectoriales y subespacios en $\mathbb{R}^2$ y $\mathbb{R}^3$.

Duración

04 horas

Procedimiento

1. Atiende la orientación docente para el desarrollo de la práctica.
2. Revisa en sus notas la definición de espacio vectorial sobre $\mathbb{R}$ y el criterio de subespacio.
3. Analiza, en equipo, distintos conjuntos de vectores en $\mathbb{R}^2$ y $\mathbb{R}^3$ (por ejemplo, rectas y planos dados por ecuaciones simples) e identifica cuáles cumplen los axiomas de espacio vectorial con las operaciones usuales.
4. Aplica el criterio de subespacio para decidir si subconjuntos como $\{(x,y,z) \in \mathbb{R}^3 : x + y + z = 0\}$, $\{(x,y,z) \in \mathbb{R}^3 : z = 0\}$ y $\{(x,y,z) \in \mathbb{R}^3 : x = 1\}$ son o no subespacios.
5. Representa gráficamente (a mano o con software de geometría dinámica) ejemplos de subespacios típicos: la recta que pasa por el origen, el plano que pasa por el origen y una recta que no pasa por el origen, explicando por qué este último caso no es subespacio.
6. Elabora una tabla donde clasifica cada conjunto analizado como “espacio vectorial / no espacio vectorial” y “subespacio/ no subespacio”, indicando brevemente el argumento (qué propiedad o axioma falla cuando no lo es).
7. Escribe un informe breve con los resultados de la práctica y lo entrega a su docente para revisión.

Recursos de apoyo

- Notas de la unidad de aprendizaje.
- Libro de texto de Álgebra Lineal recomendado por el docente.
- Pizarrón, marcadores, cuaderno y lápiz.
- Computadora o tableta con acceso a software de álgebra lineal y visualización, por ejemplo:

- [Python en Google Colab](#) (o entorno equivalente) para realizar cálculos sencillos (opcional).
- [GeoGebra](#) o software similar de geometría dinámica para representar vectores y subespacios.
- Calculadora científica o software de cálculo matricial básico.
- Opcionalmente, asistente de inteligencia artificial autorizado institucionalmente
- (por ejemplo, herramientas de IA integradas en la plataforma institucional)
- para apoyo en la verificación de procedimientos y exploración de ejemplos adicionales.

Práctica 2: Combinaciones lineales, conjuntos generadores e independencia lineal

Duración

04 horas

Procedimiento

1. Atiende la orientación docente para el desarrollo de la práctica.
2. Revisa en sus notas las definiciones de combinación lineal, conjunto generador, dependencia e independencia lineal.
3. Trabaja en equipo con ejemplos de vectores en $\mathbb{R}^2$ y $\mathbb{R}^3$: calcula combinaciones lineales simples de dos o tres vectores y describe geoméricamente el conjunto de todas las combinaciones lineales (recta, plano o todo el espacio).
4. Analiza distintos conjuntos de vectores (por ejemplo S_1, S_2, S_3 en $\mathbb{R}^3$) y decide qué subespacio generan, identificando si hay vectores redundantes que no aportan nada nuevo al espacio generado.
5. Plantea y resuelve sistemas lineales sencillos del tipo $a_1v_1 + a_2v_2 + a_3v_3 = 0$ para determinar si los vectores son linealmente dependientes o independientes, relacionando el resultado con la interpretación geométrica (vectores colineales o coplanares).
6. Usa, de manera guiada, una herramienta computacional básica (por ejemplo Python en Colab, calculadora de la computadora o software de álgebra lineal) para verificar algunos resultados de independencia lineal.
7. Elabora un cuadro de síntesis donde, para cada conjunto de vectores estudiado, indica: subespacio generado, si el conjunto es minimal (posible base) y si es dependiente o independiente, justificando brevemente.
8. Escribe un informe breve con los resultados de la práctica y lo entrega a su docente para revisión.

Recursos de apoyo

- Notas de la unidad de aprendizaje.
- Libro de texto de Álgebra Lineal recomendado por el docente.
- Pizarrón, marcadores, cuaderno y lápiz.
- Computadora o tableta con acceso a software de álgebra lineal y visualización, por ejemplo:

- [Python en Google Colab](#) (o entorno equivalente) para realizar cálculos sencillos (opcional).
- [GeoGebra](#) o software similar de geometría dinámica para representar vectores y subespacios.
- Calculadora científica o software de cálculo matricial básico.
- Opcionalmente, asistente de inteligencia artificial autorizado institucionalmente (por ejemplo, herramientas de IA integradas en la plataforma institucional) para apoyo en la verificación de procedimientos y exploración de ejemplos adicionales.

Práctica 3: Bases, dimensión, coordenadas y cambio de base conceptual

Duración

04 horas

Procedimiento

1. Atiende la orientación docente para el desarrollo de la práctica.
2. Revisa en sus notas las definiciones de base, dimensión y coordenadas de un vector respecto a una base.
3. Analiza conjuntos generadores en $\mathbb{R}^3$: a partir de un conjunto dado de vectores, elimina aquellos que resulten redundantes (dependientes) hasta obtener una base de un subespacio y determina su dimensión.
4. Compara distintas bases del mismo subespacio en $\mathbb{R}^2$ o $\mathbb{R}^3$ (por ejemplo, la base estándar y una base oblicua) y verifica que cada conjunto cumple las condiciones de base (independencia y generación).
5. Calcula, resolviendo sistemas lineales sencillos, las coordenadas de varios vectores respecto a dos bases distintas (por ejemplo, la base estándar y una base oblicua en $\mathbb{R}^2$) e interpreta el significado de esas coordenadas.
6. Discute, en equipo, el concepto de cambio de base como cambio de sistema de coordenadas: reflexiona sobre por qué el mismo vector geométrico puede tener representaciones numéricas distintas según la base elegida.
7. Opcionalmente, utiliza una herramienta computacional sencilla para automatizar el cálculo de coordenadas de un vector respecto a una base dada y verifica los resultados obtenidos a mano.
8. Escribe un informe breve con los resultados de la práctica y lo entrega a su docente para revisión.

Recursos de apoyo

- Notas de la unidad de aprendizaje.
- Libro de texto de Álgebra Lineal recomendado por el docente.
- Pizarrón, marcadores, cuaderno y lápiz.
- Computadora o tableta con acceso a software de álgebra lineal y visualización, por ejemplo:

- [Python en Google Colab](#) (o entorno equivalente) para realizar cálculos sencillos (opcional).
- [GeoGebra](#) o software similar de geometría dinámica para representar vectores y subespacios.
- Calculadora científica o software de cálculo matricial básico.
- Opcionalmente, asistente de inteligencia artificial autorizado institucionalmente (por ejemplo, herramientas de IA integradas en la plataforma institucional) para apoyo en la verificación de procedimientos y exploración de ejemplos adicionales.

Unidad II. Matrices y sistemas de ecuaciones lineales

Práctica 4: Sistemas lineales y forma $Ax = b$

Duración

04 horas

Procedimiento

1. Atiende la orientación docente para el desarrollo de la práctica.
2. Revisa en sus notas la definición de sistema de ecuaciones lineales y la forma matricial $Ax = b$.
3. En equipo, analiza varios problemas sencillos propuestos por el docente (por ejemplo: balance de mezclas, puntos de equilibrio muy simples, repartos de cantidades) y formula los sistemas de ecuaciones lineales correspondientes en $\mathbb{R}^2$ o $\mathbb{R}^3$.
4. Escribe cada sistema primero en forma de ecuaciones y, posteriormente, en forma matricial, identificando con claridad:
 - a. La matriz de coeficientes A .
 - b. El vector de incógnitas x .
 - c. El vector de términos independientes b .
5. Discute en equipo qué información aporta cada una de estas partes (A , x , b) sobre el problema original.
6. Resuelve al menos dos sistemas de 2×2 y dos de 3×3 utilizando métodos elementales (sustitución o eliminación “a mano”) y verifica si las soluciones obtenidas tienen sentido en el contexto del problema.
7. Elabora una tabla de síntesis donde registra, para cada problema:
 - a. El enunciado del problema.
 - b. El sistema en forma de ecuaciones.
 - c. La forma $Ax = b$ con A , x y b claramente identificados.
 - d. La solución encontrada y su interpretación en el contexto.
8. Escribe un informe breve con los resultados de la práctica y lo entrega a su docente para revisión.
- 9.

Recursos de apoyo

- Notas de la unidad de aprendizaje.
- Libro de texto de Álgebra Lineal recomendado por el docente.

- Pizarrón, marcadores, cuaderno y lápiz.
- Computadora o tableta con acceso a software de álgebra lineal y visualización, por ejemplo:
 - [Python en Google Colab](#) (o entorno equivalente) para realizar cálculos sencillos (opcional).
 - [GeoGebra](#) o software similar de geometría dinámica para representar vectores y subespacios.
 - Calculadora científica o software de cálculo matricial básico.
 - Opcionalmente, asistente de inteligencia artificial autorizado institucionalmente (por ejemplo, herramientas de IA integradas en la plataforma institucional) para apoyo en la verificación de procedimientos y exploración de ejemplos adicionales.

Práctica 5: Eliminación Gaussiana, matriz ampliada y clasificación de sistemas

Duración

04 horas

Procedimiento

1. Atiende la orientación docente para el desarrollo de la práctica.
2. Revisa en sus notas la definición de matriz ampliada $[A|b]$, las operaciones elementales por renglones y el concepto de forma escalonada.
3. Retoma algunos sistemas de 2×2 y 3×3 trabajados en la práctica anterior y otros nuevos propuestos por el docente; para cada uno:
 - a. Escribe la matriz de coeficientes A y la matriz ampliada $[A|b]$.
 - b. Aplica, paso a paso, operaciones elementales por renglones para llevar la matriz ampliada a forma escalonada (y, en uno o dos ejemplos, a forma escalonada reducida).
 - c. Registra explícitamente cada operación realizada y la matriz resultante.
4. A partir de la forma escalonada de $[A|b]$, determina si el sistema tiene:
 - a. Una solución única.
 - b. Infinitas soluciones.
 - c. Ninguna solución.
5. Obtiene las soluciones cuando existan, escribiéndolas en forma explícita (caso de solución única) o paramétrica (caso de infinitas soluciones), y comenta brevemente la interpretación geométrica (intersección de rectas o planos) en al menos dos ejemplos.
6. Identifica, a partir de la forma escalonada, el número de renglones no nulos de la matriz de coeficientes A y discute la idea de rango de A y de $[A|b]$ en ejemplos concretos.
7. Elabora una tabla de síntesis donde registres, para cada sistema:
 - a. La matriz ampliada inicial $[A|b]$.
 - b. Una forma escalonada de $[A|b]$.
 - c. Clasificación del sistema (única solución, infinitas soluciones, sin solución).
 - d. Comentario geométrico, cuando sea posible.
8. Escribe un informe breve con los resultados de la práctica y lo entrega a su docente para revisión.

Recursos de apoyo

- Notas de la unidad de aprendizaje.
- Libro de texto de Álgebra Lineal recomendado por el docente.
- Pizarrón, marcadores, cuaderno y lápiz.
- Computadora o tableta con acceso a software de álgebra lineal y visualización, por ejemplo:
 - [Python en Google Colab](#) (o entorno equivalente) para realizar cálculos sencillos (opcional).
 - [GeoGebra](#) o software similar de geometría dinámica para representar vectores y subespacios.
 - Calculadora científica o software de cálculo matricial básico.
 - Opcionalmente, asistente de inteligencia artificial autorizado institucionalmente (por ejemplo, herramientas de IA integradas en la plataforma institucional) para apoyo en la verificación de procedimientos y exploración de ejemplos adicionales.

Práctica 6: Determinantes y regla de Cramer en sistemas pequeños

Duración

04 horas

Procedimiento

1. Atiende la orientación docente para el desarrollo de la práctica.
2. Revisa en sus notas la definición de determinante para matrices 2×2 y 3×3 y el criterio $\det(A) \neq 0 \Leftrightarrow A$ es invertible, en ejemplos sencillos.
3. Trabaja en equipo con una colección de matrices 2×2 y 3×3 asociadas a sistemas lineales $Ax = b$:
 - a. Calcula el determinante de cada matriz A usando las fórmulas vistas en clase.
 - b. Clasifica las matrices como “invertibles” ($\det(A) \neq 0$) o “no invertibles” ($\det(A) = 0$) y discute qué implica esto para las soluciones de $Ax = b$.
4. Aplica la regla de Cramer en sistemas 2×2 y, en uno o dos casos simples, en sistemas 3×3 :
 - a. Calcula los determinantes necesarios para obtener las soluciones mediante Cramer.
 - b. Compara las soluciones obtenidas con las halladas por eliminación Gaussiana en la práctica anterior.
5. Discute en equipo las ventajas y limitaciones de la regla de Cramer:
 - a. En qué casos es útil (sistemas pequeños, cuando $\det(A) \neq 0$).
 - b. Por qué no es práctica para sistemas grandes.
6. Opcionalmente, explora un ejemplo sencillo donde el valor absoluto del determinante de una matriz 2×2 se interprete como factor de escala de área (por ejemplo, imagen de un cuadrado bajo una transformación lineal dada por una matriz A).
7. Elabora un cuadro de síntesis donde, para cada sistema estudiado, registra:

- a. Matriz de coeficientes A y su determinante.
 - b. Clasificación de A (invertible / no invertible).
 - c. Solución mediante regla de Cramer (cuando aplique) y comparación con la solución obtenida por eliminación Gaussiana.
8. Escribe un informe breve con los resultados de la práctica y lo entrega a su docente para revisión.

Recursos de apoyo

- Notas de la unidad de aprendizaje.
- Libro de texto de Álgebra Lineal recomendado por el docente.
- Pizarrón, marcadores, cuaderno y lápiz.
- Computadora o tableta con acceso a software de álgebra lineal y visualización, por ejemplo:
 - [Python en Google Colab](#) (o entorno equivalente) para realizar cálculos sencillos (opcional).
 - [GeoGebra](#) o software similar de geometría dinámica para representar vectores y subespacios.
 - Calculadora científica o software de cálculo matricial básico.
 - Opcionalmente, asistente de inteligencia artificial autorizado institucionalmente (por ejemplo, herramientas de IA integradas en la plataforma institucional) para apoyo en la verificación de procedimientos y exploración de ejemplos adicionales.

Unidad III. Transformaciones Lineales

Práctica 7: Exploración de transformaciones lineales en $\mathbb{R}^2$

Duración

04 horas

Procedimiento

1. Atiende la orientación docente para el desarrollo de la práctica.
2. Revisa en tus notas la definición de transformación lineal $T: V \rightarrow W$ y sus propiedades básicas (preserva la suma de vectores y el producto por escalar).
3. En equipo, analiza una colección de aplicaciones $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$ propuestas por el docente (por ejemplo: escalamientos, proyecciones sobre un eje y simetrías respecto a un eje) e identifica cuáles son transformaciones lineales y cuáles no, justificando en cada caso.
4. Para cada aplicación que sí sea transformación lineal, verifica explícitamente en al menos dos ejemplos que $T(u + v) = T(u) + T(v)$ y $T(\alpha u) = \alpha T(u)$ para algunos vectores u , v y escalares α representativos.
5. Representa gráficamente (a mano o con software de geometría dinámica) algunos de los ejemplos más sencillos, mostrando cómo se transforman los vectores de la base estándar y uno o dos vectores adicionales.

6. Elabora una tabla donde, para cada aplicación estudiada, indica: si es o no lineal, una breve justificación y una descripción cualitativa de su efecto geométrico (escala, proyecta y refleja, etc.).
7. Escribe un informe breve con los resultados de la práctica y lo entrega a su docente para revisión.

Recursos de apoyo

- Notas de la unidad de aprendizaje.
- Libro de texto de Álgebra Lineal recomendado por el docente.
- Pizarrón, marcadores, cuaderno y lápiz.
- Computadora o tableta con acceso a software de álgebra lineal y visualización, por ejemplo:
 - [Python en Google Colab](#) (o entorno equivalente) para realizar cálculos sencillos.
 - [GeoGebra](#) o software similar de geometría dinámica para representar vectores y subespacios.
 - Calculadora científica o software de cálculo matricial básico.
 - Opcionalmente, asistente de inteligencia artificial autorizado institucionalmente (por ejemplo, herramientas de IA integradas en la plataforma institucional) para apoyo en la verificación de procedimientos y exploración de ejemplos adicionales.

Práctica 8: ¿Es o no es transformación lineal?

Duración

04 horas

Procedimiento

1. Atiende la orientación docente para el desarrollo de la práctica.
2. Revisa en sus notas la definición de transformación lineal $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$ (preserva suma y producto por escalar).
3. En equipo, analiza una lista breve de aplicaciones $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$ dadas por el docente, por ejemplo:
 - a. $T_1(x,y) = (2x, 3y)$;
 - b. $T_2(x,y) = (x + y, y)$;
 - c. $T_3(x,y) = (x + 1, y)$;
 - d. $T_4(x,y) = (0,0)$.
4. Para cada aplicación:
 - a. Verifica, con uno o dos pares de vectores u, v y un escalar α , si se cumple $T(u + v) = T(u) + T(v)$ y $T(\alpha u) = \alpha T(u)$.
 - b. Decide si es transformación lineal o no, y anota en una frase cuál propiedad falla cuando no lo es.
5. Completa una tabla con columnas: nombre de la aplicación, expresión de $T(x,y)$, “¿es lineal?” (sí/no) y breve justificación.
6. Escribe un informe breve con los resultados de la práctica y lo entrega a su docente para revisión.

Recursos de apoyo

- Notas de la unidad de aprendizaje.
- Libro de texto de Álgebra Lineal recomendado por el docente.
- Pizarrón, marcadores, cuaderno y lápiz.
- Computadora o tableta con acceso a software de álgebra lineal y visualización, por ejemplo:
 - [Python en Google Colab](#) (o entorno equivalente) para realizar cálculos sencillos.
 - [GeoGebra](#) software similar de geometría dinámica para representar vectores y subespacios.
 - Calculadora científica o software de cálculo matricial básico.
 - Opcionalmente, asistente de inteligencia artificial autorizado institucionalmente (por ejemplo, herramientas de IA integradas en la plataforma institucional) para apoyo en la verificación de procedimientos y exploración de ejemplos adicionales.

Práctica 9: Núcleo, imagen y matriz asociada en $\mathbb{R}^2$

Duración

04 horas

Procedimiento

1. Atiende la orientación docente para el desarrollo de la práctica.
2. Revisa en tus notas las definiciones de núcleo $\ker(T)$, imagen $\text{im}(T)$ y matriz asociada de una transformación lineal $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$ en la base estándar.
3. En equipo, trabaja con 2 o 3 transformaciones lineales sencillas dadas por el docente, por ejemplo:
 - a. $T_1(x,y) = (x + y, y)$
 - b. $T_2(x,y) = (x - y, x + y)$
 - c. $T_3(x,y) = (0, x + y)$
4. Para cada T :
 - a. Calcula su matriz asociada A en la base estándar usando $T(e_1)$ y $T(e_2)$.
 - b. Resuelve $T(x,y) = (0,0)$ para describir el núcleo $\ker(T)$ como recta o $\{0\}$.
 - c. Describe informalmente la imagen $\text{im}(T)$ (por ejemplo: todo $\mathbb{R}^2$, una recta que pasa por el origen, etc.) a partir de la matriz o de ejemplos de $T(x,y)$.
5. Completa una tabla con columnas: transformación T , matriz A , descripción de $\ker(T)$, descripción de $\text{im}(T)$ y comentario rápido sobre el “efecto geométrico” (por ejemplo: proyecta sobre una recta, mezcla las coordenadas, etc.).
6. Escribe un informe breve con los resultados de la práctica y lo entrega a su docente para revisión.

Recursos de apoyo

- Notas de la unidad de aprendizaje.
- Libro de texto de Álgebra Lineal recomendado por el docente.
- Pizarrón, marcadores, cuaderno y lápiz.

- Computadora o tableta con acceso a software de álgebra lineal y visualización, por ejemplo:
 - [Python en Google Colab](#) (o entorno equivalente) para realizar cálculos sencillos.
 - [GeoGebra](#) o software similar de geometría dinámica para representar vectores y subespacios.
 - Calculadora científica o software de cálculo matricial básico.
 - Opcionalmente, asistente de inteligencia artificial autorizado institucionalmente (por ejemplo, herramientas de IA integradas en la plataforma institucional) para apoyo en la verificación de procedimientos y exploración de ejemplos adicionales.

Unidad IV: Productos escalares y ortogonalidad

Práctica 10: Producto escalar, norma, distancia y ángulo

Duración

04 horas

Procedimiento

1. Atiende la orientación docente para el desarrollo de la práctica.
2. Revisa en sus notas la definición de producto escalar usual en $\mathbb{R}^n$ y las fórmulas básicas en $\mathbb{R}^2$ y $\mathbb{R}^3$.
3. En equipo, calcula el producto escalar de pares de vectores sencillos en $\mathbb{R}^2$ y $\mathbb{R}^3$ (dados por el docente o propuestos por el equipo) y verifica propiedades básicas como:
 - a. Simetría: $u \cdot v = v \cdot u$.
 - b. Linealidad en cada entrada (en ejemplos concretos).
 - c. Positividad: $u \cdot u \geq 0$ y $u \cdot u = 0$ solo si $u = 0$.
4. A partir del producto escalar, calcula la norma de varios vectores ($\|u\| = \sqrt{u \cdot u}$) y la distancia entre pares de puntos/ vectores en $\mathbb{R}^2$, interpretando geoméricamente estos resultados (longitud de flechas, distancia entre puntos).
5. Utiliza la fórmula del coseno para el ángulo entre vectores en $\mathbb{R}^2$ y $\mathbb{R}^3$ en ejemplos sencillos, y representa algunos de ellos en el plano o en el espacio para visualizar el ángulo calculado.
6. Clasifica pares de vectores como “obtusos”, “agudos” u “ortogonales” de acuerdo con el signo del producto escalar y el valor del ángulo.
7. Elabora una tabla de síntesis con varios ejemplos donde registra, para cada par de vectores:
 - a. u, v .
 - b. $u \cdot v, \|u\|, \|v\|$.
 - c. Ángulo aproximado entre u y v .
 - d. Comentario geométrico (por ejemplo: “casi perpendiculares”, “casi paralelos”, etc.).
8. Escribe un informe breve con los resultados de la práctica y lo entrega a su docente para revisión.

Recursos de apoyo

- Notas de la unidad de aprendizaje.
- Libro de texto de Álgebra Lineal recomendado por el docente.
- Pizarrón, marcadores, cuaderno y lápiz.
- Computadora o tableta con acceso a software de álgebra lineal y visualización, por ejemplo:
 - [Python en Google Colab](#) (o entorno equivalente) para realizar cálculos sencillos.
 - [GeoGebra](#) o software similar de geometría dinámica para representar vectores y subespacios.
 - Calculadora científica o software de cálculo matricial básico.
 - Opcionalmente, asistente de inteligencia artificial autorizado institucionalmente (por ejemplo, herramientas de IA integradas en la plataforma institucional) para apoyo en la verificación de procedimientos y exploración de ejemplos adicionales.

Práctica 11: Ortogonalidad, bases ortonormales y proyección sobre una recta

Duración

04 horas

Procedimiento

1. Atiende la orientación docente para el desarrollo de la práctica.
2. Revisa en tus notas las definiciones de ortogonalidad, conjunto ortonormal y base ortonormal en $\mathbb{R}^2$ y $\mathbb{R}^3$.
3. En equipo, analiza distintos conjuntos de vectores en $\mathbb{R}^2$ y $\mathbb{R}^3$ para determinar:
 - a. Si contienen pares de vectores ortogonales (usando el producto escalar).
 - b. Si forman conjuntos ortonormales (ortogonales y de norma 1).
4. Normaliza varios vectores no nulos (es decir, construye vectores de norma 1 en la misma dirección) y a partir de ellos forma pequeños conjuntos ortonormales en $\mathbb{R}^2$ y $\mathbb{R}^3$.
5. Discute, con ejemplos numéricos sencillos, las ventajas de trabajar con bases ortonormales (por ejemplo, la fórmula simple para las coordenadas de un vector respecto a una base ortonormal).
6. Aplica la fórmula de proyección ortogonal de un vector sobre una recta en $\mathbb{R}^2$:
 - a. Elige un vector no nulo u que genere la recta y un vector v cualquiera.
 - b. Calcula la proyección de v sobre la recta generada por u y la componente perpendicular correspondiente.
 - c. Representa gráficamente algunos ejemplos y comenta la interpretación geométrica de la proyección.
7. Elabora una tabla de síntesis donde registres:
 - a. Ejemplos de conjuntos ortogonales y ortonormales.
 - b. Casos de proyección de un vector sobre una recta, con los valores de la componente paralela y perpendicular.
8. Escribe un informe breve con los resultados de la práctica y lo entrega a su docente para revisión.

Recursos de apoyo

- Notas de la unidad de aprendizaje.
- Libro de texto de Álgebra Lineal recomendado por el docente.
- Pizarrón, marcadores, cuaderno y lápiz.
- Computadora o tableta con acceso a software de álgebra lineal y visualización, por ejemplo:
 - [Python en Google Colab](#) (o entorno equivalente) para realizar cálculos sencillos.
 - [GeoGebra](#) o software similar de geometría dinámica para representar vectores y subespacios.
 - Calculadora científica o software de cálculo matricial básico.
 - Opcionalmente, asistente de inteligencia artificial autorizado institucionalmente (por ejemplo, herramientas de IA integradas en la plataforma institucional) para apoyo en la verificación de procedimientos y exploración de ejemplos adicionales.

Práctica 12: Ortonormalización de Gram–Schmidt en $\mathbb{R}^2$ y $\mathbb{R}^3$

Duración

04 horas

Procedimiento

1. Atiende la orientación docente para el desarrollo de la práctica.
2. Revisa en sus notas la idea del procedimiento de Gram–Schmidt para construir una base ortonormal a partir de un conjunto de vectores linealmente independientes.
3. En equipo, trabaja con ejemplos concretos de familias de vectores linealmente independientes en $\mathbb{R}^2$ y $\mathbb{R}^3$ (dados por el docente):
 - a. Verifica, en cada ejemplo, que los vectores son linealmente independientes (usando criterios vistos en unidades anteriores, en casos muy sencillos).
 - b. Aplica paso a paso el procedimiento de Gram–Schmidt para obtener primero una base ortogonal y después una base ortonormal del subespacio generado.
4. Registra de forma ordenada cada paso del procedimiento:
 - a. Cálculo de productos escalares necesarios.
 - b. Cálculo de las componentes proyectadas.
 - c. Obtención de los nuevos vectores ortogonales y su normalización.
5. Comprueba, en cada ejemplo, que la base final es realmente ortonormal (verificando ortogonalidad y norma 1) y que genera el mismo subespacio que la familia inicial.
6. Aplica la base ortonormal obtenida para resolver uno o dos problemas sencillos de proyección:
 - a. Proyección de un vector sobre la recta o plano generado por parte de la base ortonormal.
 - b. Descomposición de un vector en componentes respecto a la base ortonormal.
7. Discute en equipo las ventajas de disponer de una base ortonormal al trabajar con productos escalares y proyecciones (simplicidad de las fórmulas, interpretación geométrica clara).
8. Opcionalmente, implementa el procedimiento de Gram–Schmidt para un ejemplo sencillo en una herramienta computacional (por ejemplo, una hoja de cálculo o un pequeño script en Python) y compara los resultados con los cálculos hechos a mano.
9. Escribe un informe breve con los resultados de la práctica y lo entrega a su docente para revisión.

Recursos de apoyo

- Notas de la unidad de aprendizaje.
- Libro de texto de Álgebra Lineal recomendado por el docente.
- Pizarrón, marcadores, cuaderno y lápiz.
- Computadora o tableta con acceso a software de álgebra lineal y visualización, por ejemplo:
 - [Python en Google Colab](#) (o entorno equivalente) para realizar cálculos sencillos.
 - [GeoGebra](#) o software similar de geometría dinámica para representar vectores y subespacios.
 - Calculadora científica o software de cálculo matricial básico.
 - Opcionalmente, asistente de inteligencia artificial autorizado institucionalmente (por ejemplo, herramientas de IA integradas en la plataforma institucional) para apoyo en la verificación de procedimientos y exploración de ejemplos adicionales.

Unidad V. Conceptos fundamentales de valores y vectores propios

Práctica 13: Valores y vectores propios en matrices 2×2

Duración

04 horas

Procedimiento

1. Atiende la orientación docente para el desarrollo de la práctica.
2. Revisa en sus notas la definición de autovalor (valor propio) y autovector (vector propio) de una matriz cuadrada.
3. En equipo, trabaja con una colección de matrices reales 2×2 sencillas (proporcionadas por el docente), por ejemplo:
 - a. Matrices diagonales.
 - b. Matrices triangulares.
 - c. Matrices de rotación-escala simples.
4. Para cada matriz:
 - a. Calcula, por ensayo sencillo o usando la ecuación $A v = \lambda v$, algunos pares (λ, v) candidatos.
 - b. Verifica si cumplen la relación $A v = \lambda v$ para decidir si son realmente valores y vectores propios.
5. Discute qué ocurre cuando el vector candidato v es el vector cero y por qué no se considera vector propio.
6. Clasifica los ejemplos de matrices 2×2 según el número de valores propios reales que presentan (uno, dos o ninguno).
7. Elabora una tabla de síntesis donde registres, para cada matriz estudiada:
 - a. A .
 - b. Valores propios reales encontrados.
 - c. Uno o dos vectores propios asociados a cada valor propio.
 - d. Comentario sobre el efecto geométrico aproximado (por ejemplo: estira en una dirección, contrae en otra, etc.).
8. Escribe un informe breve con el desarrollo y las conclusiones de la práctica y lo entrega a su docente para revisión y retroalimentación.
9. Escribe un informe breve con los resultados de la práctica y lo entrega a su docente para revisión.

Recursos de apoyo

- Notas de la unidad de aprendizaje.
- Libro de texto de Álgebra Lineal recomendado por el docente.
- Pizarrón, marcadores, cuaderno y lápiz.
- Computadora o tableta con acceso a software de álgebra lineal y visualización, por ejemplo:
 - [Python en Google Colab](#) (o entorno equivalente) para realizar cálculos sencillos.
 - [GeoGebra](#) o software similar de geometría dinámica para representar vectores y subespacios.

- Calculadora científica o software de cálculo matricial básico.
- Opcionalmente, asistente de inteligencia artificial autorizado institucionalmente (por ejemplo, herramientas de IA integradas en la plataforma institucional) para apoyo en la verificación de procedimientos y exploración de ejemplos adicionales

Práctica 14: Polinomio característico y cálculo de valores propios

Duración

04 horas

Procedimiento

1. Atiende la orientación docente para el desarrollo de la práctica.
2. Revisa en sus notas la definición de polinomio característico $p_A(\lambda) = \det(A - \lambda I)$ para matrices 2×2 y 3×3 en casos sencillos.
3. En equipo, trabaja con varias matrices reales 2×2 y algunas 3×3 de estructura simple (diagonales, triangulares o casi diagonales):
 - a. Calcula el polinomio característico $p_A(\lambda)$ de cada matriz usando determinantes.
 - b. Escribe la ecuación característica $p_A(\lambda) = 0$ y resuélvela para encontrar los valores propios reales.
4. Comprueba que, en el caso 2×2 , el polinomio característico se puede escribir como $\lambda^2 - (\text{tr } A)\lambda + \det(A)$, y lo usa para simplificar algunos cálculos.
5. Relaciona, para cada matriz, los valores propios obtenidos con ejemplos de la práctica anterior en los que se habían encontrado autovalores “a mano”.
6. Discute en equipo qué ocurre cuando el polinomio característico no tiene raíces reales (por ejemplo, en algunas matrices de rotación) y qué significa esto para los valores propios reales.
7. Elabora una tabla de síntesis donde registres, para cada matriz:
 - a. A .
 - b. Polinomio característico $p_A(\lambda)$.
 - c. Valores propios reales encontrados.
8. Escribe un informe breve con el desarrollo y las conclusiones de la práctica y lo entrega a su docente para revisión y retroalimentación.
9. Escribe un informe breve con los resultados de la práctica y lo entrega a su docente para revisión.

Recursos de apoyo

- Notas de la unidad de aprendizaje.
- Libro de texto de Álgebra Lineal recomendado por el docente.
- Pizarrón, marcadores, cuaderno y lápiz.
- Computadora o tableta con acceso a software de álgebra lineal y visualización, por ejemplo:
 - [Python en Google Colab](#) (o entorno equivalente) para realizar cálculos sencillos.

- [GeoGebra](#) o software similar de geometría dinámica para representar vectores y subespacios.
- Calculadora científica o software de cálculo matricial básico.
- Opcionalmente, asistente de inteligencia artificial autorizado institucionalmente (por ejemplo, herramientas de IA integradas en la plataforma institucional) para apoyo en la verificación de procedimientos y exploración de ejemplos adicionales.

Práctica 15: Espacios propios, bases y significado geométrico

Duración

04 horas

Procedimiento

1. Atiende la orientación docente para el desarrollo de la práctica.
2. Revisa en sus notas la definición de espacio propio asociado a un valor propio y la idea de que se trata de un subespacio de $\mathbb{R}^n$.
3. En equipo, selecciona algunas matrices 2×2 y 3×3 sencillas con valores propios reales conocidos (por ejemplo, obtenidos en las prácticas anteriores).
4. Para cada matriz y cada valor propio λ :
 - a. Plantea la ecuación $(A - \lambda I) v = 0$.
 - b. Resuelve el sistema lineal correspondiente para encontrar todos los vectores propios asociados a λ (excepto el vector cero).
 - c. Describe el conjunto solución como un subespacio (recta o plano que pasa por el origen).
5. Identifica una base para cada espacio propio encontrado y discute su dimensión.
6. En el caso de matrices 2×2 , representa en el plano (a mano o con software de geometría dinámica) la dirección de los vectores propios y discute el efecto de la transformación A sobre ellos (escalamiento sin cambio de dirección).
7. Elabora una tabla de síntesis donde registres, para cada matriz:
 - a. Valores propios.
 - b. Una base de cada espacio propio.
 - c. Dimensión de cada espacio propio (dimensión geométrica).
8. Escribe un informe breve con los resultados de la práctica y lo entrega a su docente para revisión.

Recursos de apoyo

- Notas de la unidad de aprendizaje.
- Libro de texto de Álgebra Lineal recomendado por el docente.
- Pizarrón, marcadores, cuaderno y lápiz.
- Computadora o tableta con acceso a software de álgebra lineal y visualización, por ejemplo:
 - [Python en Google Colab](#) (o entorno equivalente) para realizar cálculos sencillos.
 - [GeoGebra](#) o software similar de geometría dinámica para representar vectores y subespacios.

- Calculadora científica o software de cálculo matricial básico.
- Opcionalmente, asistente de inteligencia artificial autorizado institucionalmente (por ejemplo, herramientas de IA integradas en la plataforma institucional) para apoyo en la verificación de procedimientos y exploración de ejemplos adicionales.

Práctica 16: Diagonalización básica y aplicaciones sencillas

Duración

04 horas

Procedimiento

1. Atiende la orientación docente para el desarrollo de la práctica.
2. Revisa en sus notas la idea de diagonalización $A = PDP^{-1}$ en dimensión pequeña y el criterio práctico: una matriz $n \times n$ es diagonalizable si admite n autovectores linealmente independientes.
3. En equipo, trabaja con matrices 2×2 y 3×3 sencillas con valores propios reales:
 - a. Reúne, para cada matriz, un conjunto de vectores propios correspondiente a cada valor propio.
 - b. Decide si el número total de vectores propios linealmente independientes es igual a la dimensión del espacio (2 o 3). En ese caso, concluye que la matriz es diagonalizable.
4. Para cada matriz diagonalizable:
 - a. Forma la matriz P cuyas columnas son los vectores propios elegidos.
 - b. Forma la matriz diagonal D que contiene los valores propios en la diagonal principal, en el mismo orden que las columnas de P .
 - c. Verifica, en ejemplos concretos, la relación $A \approx PDP^{-1}$ (al menos numéricamente).
5. Discute la interpretación de la diagonalización como un cambio de base que vuelve especialmente simple la acción de A (en la base de autovectores, A actúa como un escalamiento independiente en cada dirección propia).
6. Opcionalmente, analiza un ejemplo sencillo de aplicación (por ejemplo, un sistema lineal discreto $x_{k+1} = A x_k$) donde la diagonalización permite entender fácilmente la evolución en el tiempo en términos de valores propios.
7. Escribe un informe breve con los resultados de la práctica y lo entrega a su docente para revisión.

Recursos de apoyo

- Notas de la unidad de aprendizaje.
- Libro de texto de Álgebra Lineal recomendado por el docente.
- Pizarrón, marcadores, cuaderno y lápiz.
- Computadora o tableta con acceso a software de álgebra lineal y visualización, por ejemplo:
 - [Python en Google Colab](#) (o entorno equivalente) para realizar cálculos sencillos.
 - [GeoGebra](#) o software similar de geometría dinámica para representar vectores y subespacios.

- Calculadora científica o software de cálculo matricial básico.
- Opcionalmente, asistente de inteligencia artificial autorizado institucionalmente (por ejemplo, herramientas de IA integradas en la plataforma institucional) para apoyo en la verificación de procedimientos y exploración de ejemplos adicionales.

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer de forma clara y formal:

- La metodología de trabajo.
- Criterios de evaluación y rúbricas.
- Reglas sobre la calidad y entrega de trabajos académicos.
- Derechos y obligaciones del o la docente y del estudiantado

Esto debe quedar registrado y comunicado por medios institucionales (como Blackboard o correo institucional).

Estrategia de enseñanza

- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en preguntas
- Técnica expositiva
- Discusión guiada

Estrategia de aprendizaje

- Investigación documental
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

- Evaluaciones parciales	40 %
- Prácticas de taller	30 %

- Participación en clase	10 %
- Tareas	20 %
Total	100 %

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Flores Godoy, J. J., & Grossman, S. I. (2019). *Álgebra lineal* (8.ªed.). McGraw-Hill.
- Lay, D. C., Lay, S. R., & McDonald, J. J. (2021). *Linear algebra and its applications* (6th ed.). Pearson.
- Poole, D. (2018). *Álgebra lineal: Una introducción moderna* (4.ªed.). Cengage Learning.
- Strang, G. (2023). *Introduction to linear algebra* (6th ed.). Wellesley-Cambridge Press.

Recursos de acceso libre o complementarios externos

- Arroyo Paniagua, M. J., & Bromberg Silverstein, S. T. (2021). *Álgebra lineal*. Editorial Trillas.
- Axler, S. (2024). *Linear algebra done right* (4.ª ed.). Springer.
- Axler, S. (2024). *Linear algebra done right* (4.ª ed.). <https://linear.axler.net/>
- Beezer, R. A. (2012). *A first course in linear algebra* (Version 2.90). <http://linear.ups.edu/>
- Friedberg, S. H., Insel, A. J., & Spence, L. E. (2019). *Linear algebra* (5th ed.). Pearson.
- Lang, S. (1987). *Linear algebra* (3rd ed.). Springer. [clásica]
- Lipschutz, S., & Lipson, M. (2012). *Schaum's outline of linear algebra* (5th ed.). McGraw-Hill.
- Strang, G. (2023). *MIT OpenCourseWare - Linear Algebra*. <https://ocw.mit.edu/courses/18-06-linear-algebra-spring-2010/>
- Strang, G. (s.f.). *Video lectures*. <https://www.youtube.com/playlist?list=PL49CF3715CB9EF31D>

XII. PERFIL DOCENTE

Licenciatura en Matemáticas, Física o área afín, preferentemente con posgrado. Se requiere experiencia en docencia universitaria de cursos introductorios de álgebra lineal o matemáticas superiores y dominio sólido de los fundamentos del álgebra lineal, incluyendo espacios vectoriales reales de dimensión finita, transformaciones lineales, matrices, sistemas de ecuaciones lineales, determinantes, ortogonalidad y diagonalización básica. Debe mostrar capacidad para relacionar conceptos abstractos con interpretaciones geométricas en $\mathbb{R}^2$ y $\mathbb{R}^3$, así como familiaridad con aplicaciones elementales en ciencias exactas y naturales. Se busca una persona analítica y responsable, que fomente el pensamiento matemático riguroso con

desarrollo gradual, el razonamiento crítico, el uso de herramientas computacionales básicas Python, MATLAB, GeoGebra o similares para visualización y cálculo y el trabajo colaborativo en sus estudiantes.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	GNU/Linux para las Ciencias
Clave:	
Distribución horaria:	Horas Clase: 01 Horas Taller: 01 Horas Laboratorio: 02 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 01 Créditos:05
Etapas de formación:	Básica
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Gerardo Tovar Ramos Adrián Enciso Almanza
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Proporcionar al estudiantado los conocimientos, habilidades y actitudes necesarias para comprender y aplicar los fundamentos operativos y científicos de GNU/Linux en la gestión de sistemas, el procesamiento de información y la reproducción de entornos de trabajo orientados a la investigación. Se permitirá el desarrollo de competencias prácticas en la administración de usuarios y procesos, el manejo estructurado de archivos, el tratamiento formal de datos en texto plano, la automatización de flujos de trabajo y la construcción de entornos reproducibles con servicios de red, fomentando el pensamiento crítico, la responsabilidad y la disciplina académica. Se imparte en la etapa básica, es de carácter obligatorio y pertenece al área de conocimiento de Software de Base.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Aplicar los fundamentos del sistema operativo GNU/Linux en flujos de trabajo científicos mediante procedimientos de gestión de sistemas, metodologías de procesamiento de información y técnicas de automatización y control de versiones, para garantizar eficiencia, trazabilidad y calidad en proyectos académicos y tecnológicos, con responsabilidad profesional, pensamiento crítico y disposición al trabajo colaborativo.

IV. EVIDENCIA DE APRENDIZAJE

Presentar un entorno reproducible para el trabajo científico en otra disciplina, integrando scripts de automatización, control de versiones, servicios de red y un componente de visualización accesible, con el fin de demostrar la aplicabilidad de GNU/Linux en diversas áreas científicas.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Desarrollar software de base eficiente y robusto mediante la integración de conocimientos en arquitectura y redes de computadoras y los fundamentos de compiladores y sistemas operativos, para optimizar el rendimiento, la gestión de recursos y la comunicación entre sistemas, asegurando aplicaciones funcionales, seguras, escalables y sostenibles, con una actitud ética, responsable y visionaria.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Responsabilidad Social; Innovación y Pensamiento Crítico.

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos y arquitectura: de la filosofía Unix a la infraestructura GNU/Linux

Competencia

Distinguir los componentes del sistema GNU/Linux mediante la revisión y comprensión de los principios filosóficos y arquitectónicos, para aplicar los principios de diseño en entornos científicos, con actitud crítica, curiosidad intelectual y compromiso con el aprendizaje autónomo.

Contenido

1.1. Principios de diseño de sistemas tipo Unix

1.1.1 Modularidad

1.1.2. Composicionalidad

1.1.3. Simplicidad

1.2. Historia y marco conceptual

1.2.1. Unix

1.2.2. Proyecto GNU

1.2.3. Kernel Linux

1.2.4. Distribuciones y comunidades

1.3. GNU/Linux en ciencia e industria

1.3.1. Cómputo en la nube

1.3.2. Supercómputo

1.3.3. Inteligencia Artificial

1.3.4. Sistemas embebidos e IoT

1.4. Anatomía lógica del sistema

1.4.1. Intérprete de órdenes

1.4.2. Programas

1.4.3. Árbol del sistema de archivos

1.4.4. Procesos y permisos

Duración

3 horas

Unidad II. Gestión del sistema: todo es un archivo, hasta que es un proceso

Competencia

Configurar entornos de trabajo en GNU/Linux mediante la gestión de archivos, procesos, usuarios, servicios y dependencias del sistema, para optimizar la administración de recursos y asegurar la continuidad operativa en proyectos académicos y tecnológicos, con autonomía, iniciativa y responsabilidad en el desempeño individual.

Contenido

- 2.1. Modelos de interacción con el sistema
 - 2.1.1. Sesiones de usuario y entornos
 - 2.1.2. Variables y configuración
- 2.2. Estructura de órdenes y convenciones de uso
 - 2.2.1. Entrada y salida
 - 2.2.2. Redirecciones
 - 2.2.3. Composición de tareas
- 2.3. Organización del sistema de archivos
 - 2.3.1. Jerarquía
 - 2.3.2. Rutas y convenciones
 - 2.3.3. Usuarios y grupos
 - 2.3.4. Privilegios y control de acceso
- 2.4. Administración del sistema
 - 2.4.1. Cuentas de usuario
 - 2.4.2. Servicios esenciales del sistema
 - 2.4.3. Monitoreo de recursos y estados
 - 2.4.3.1. Proceso
 - 2.4.3.2. Memoria
 - 2.4.3.3. Red
 - 2.4.3.4. Almacenamiento
- 2.5. Principios de gestión de software y dependencias
 - 2.5.1. Instalación y actualización
 - 2.5.2. Repositorios

Duración

4 horas

Unidad III. Texto: el lenguaje universal

Competencia

Aplicar técnicas formales de tratamiento de texto en GNU/Linux mediante flujos de información, gramáticas de patrones regulares, transformaciones textuales, procesamiento estructurado y automatización con scripts, para filtrar, transformar y sintetizar información en entornos de investigación, con pensamiento analítico, disciplina y compromiso con la calidad.

Contenido

- 3.1. Flujos de información y composición de tareas
 - 3.1.1. Canales de entrada y salida
 - 3.1.2. Encadenamiento de operaciones
- 3.2. Gramáticas de patrones regulares
 - 3.2.1. Sintaxis
 - 3.2.2. Cuantificadores y anclas
 - 3.2.3. Grupos
- 3.3. Transformaciones textuales
 - 3.3.1. Selección y sustitución
 - 3.3.2. Normalización
 - 3.3.3. Tokenización
- 3.4. Procesamiento estructurado
 - 3.4.1. Campos y columnas
 - 3.4.2. Reglas condicionales
 - 3.4.3. Agregación
- 3.5. Síntesis de resultados
 - 3.5.1. Ordenamiento
 - 3.5.2. Conteo
 - 3.5.3. Reportes y resúmenes
- 3.6. Introducción a la automatización con scripts
 - 3.6.1. Secuencias de comandos a partir de flujos de texto
 - 3.6.2. Reutilización y modularidad básica
 - 3.6.3. Ejecución automatizada de tareas rutinarias

Duración

4 horas

Unidad IV. Ciencia: la magia de la automatización y la reproducibilidad

Competencia

Reproducir entornos de trabajo especializados mediante automatización con scripts, control de cambios, servicios de red y contenedores, para integrarse de manera eficiente a flujos científicos, con rigor técnico, colaboración y compromiso con la calidad.

Contenido

- 4.1. Automatización y control de versiones
 - 4.1.1. Desarrollo de scripts de sistema
 - 4.1.1.1. Encadenamiento de comandos en archivos de secuencia
 - 4.1.1.2. Parametrización de entradas y salidas
 - 4.1.1.3. Validación de condiciones y manejo de errores
- 4.2. Integración con control de versiones

- 4.2.1. Principios de trazabilidad y colaboración
- 4.2.2. Versionado de scripts y flujos de trabajo
- 4.2.3. Gestión de ramas y fusiones
- 4.2.4. Historial y auditoría de cambios
- 4.3. Reproducibilidad computacional
 - 4.3.1. Especificación de entornos
 - 4.3.2. Gestión de dependencias
 - 4.3.3. Portabilidad entre sistemas
- 4.4. Redes y servicios en GNU/Linux
 - 4.4.1. Comunicación entre procesos
 - 4.4.2. Servicios de red básicos
 - 4.4.3. Sockets y monitoreo
 - 4.4.4. Virtualización de servicios en contenedores
- 4.5. Entornos aislados y empaquetado
 - 4.5.1. Contenerización básica
 - 4.5.2. Definición de entornos reproducibles
 - 4.5.3. Despliegue y orquestación inicial
- 4.6. Comunicación científica
 - 4.6.1. Documentación técnica y metodológica
 - 4.6.2. Bitácoras de experimentación
 - 4.6.3. Transferencia y compartición de proyectos

Duración

5 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

Unidad I. Fundamentos y arquitectura: de la filosofía Unix a la infraestructura GNU/Linux

Práctica 1: Hitos de Unix vigentes en GNU/Linux

Duración

2 horas

Procedimiento

1. Recibe instrucciones para el desarrollo de la práctica.
2. Accede al entorno de GNU/Linux asignado.
3. Verifica la versión del kernel y la distribución instalada.

4. Explora cómo se representa el hardware y los procesos como archivos dentro del sistema.
5. Comprueba el principio de modularidad al encadenar herramientas pequeñas en un flujo.
6. Identifica evidencias del principio “todo es un archivo” recorriendo el sistema de archivos.
7. Documenta ejemplos de la simplicidad en la estructura del árbol de directorios.
8. Ejecuta operaciones básicas de manipulación de archivos y procesos para confirmar la regla del silencio y la regla de composición.
9. Registra en la bitácora cada hito de Unix (modularidad, simplicidad, todo es un archivo, silencio, composición) con un caso concreto observado en GNU/Linux.
10. Recibe retroalimentación docente.

Recursos de apoyo

- Entorno GNU/Linux instalado (físico o virtual).
- Guía de práctica con lista de hitos de Unix.
- Bitácora de actividades.
- Editor de texto para elaborar tabla y reflexión.

Unidad II. Gestión del sistema: todo es un archivo, hasta que es un proceso

Práctica 2: Configuración de entornos de trabajo en GNU/Linux

Duración

4 horas

Procedimiento

1. Accede al entorno de GNU/Linux asignado.
2. Recibe instrucciones para el desarrollo de la práctica.
3. Configura variables de entorno y directorios de trabajo para un usuario específico.
4. Crea y administra cuentas de usuario y grupos con privilegios diferenciados.
5. Asigna permisos de lectura, escritura y ejecución en directorios compartidos.
6. Inicia, detiene y monitorea procesos en ejecución para observar el ciclo de vida del sistema.
7. Registra en la bitácora los pasos realizados, dificultades encontradas y la justificación de las decisiones de configuración.
8. Recibe retroalimentación docente.

Recursos de apoyo

- Entorno GNU/Linux con cuentas de prueba.
- Guía de práctica con instrucciones generales.
- Bitácora de actividades.
- Editor de texto para documentación.

Unidad III. Texto: el lenguaje universal

Práctica 3: El Poder de los Patrones: Búsqueda y Transformación de Texto

Duración

4 horas

Procedimiento

1. Atiende instrucciones para el desarrollo de la práctica.
2. Recibe un archivo de texto de ejemplo y una guía de referencia con la sintaxis de expresiones regulares.
3. Utiliza la herramienta de filtrado de texto para ejecutar búsquedas. Inicia con patrones literales y progresa hasta el uso de metacaracteres y cuantificadores para encontrar líneas que cumplan con criterios complejos definidos por regex.
4. Aplica el editor de flujo de texto para realizar sustituciones selectivas, utilizando grupos de captura de regex para reutilizar partes del texto encontrado en el reemplazo.
5. Emplea el procesador de texto estructurado para filtrar y mostrar únicamente las columnas de aquellas líneas que coincidan con un patrón regex específico en uno de sus campos.
6. Diseña y combina una secuencia de comandos para generar un resumen basado en el conteo de patrones regex específicos.
7. Registra en la bitácora cada patrón regex diseñado, explicando su lógica y el resultado de su aplicación en el flujo de comandos.
8. Recibe retroalimentación docente.

Recursos de apoyo

- Entorno GNU/Linux.
- Archivo de texto con datos de ejemplo.
- Guía de referencia con la sintaxis de expresiones regulares (regex).

Unidad IV. Ciencia: La Magia de la Automatización y la Reproducibilidad

Práctica 4: Integración de la automatización y el control de versiones en el flujo de trabajo

Duración

6 horas

Procedimiento

1. Recibe instrucciones para el desarrollo de la práctica.
2. Accede al entorno de GNU/Linux asignado con soporte para repositorios de control de versiones.
3. Desarrolla un script que automatice una tarea científica básica (procesamiento de texto o datos).
4. Parametriza el script para aceptar diferentes entradas y validar errores comunes.
5. Crea un repositorio local, agrega el script y documenta los cambios en el historial.

6. Gestiona ramas para probar variaciones del script y fusiona resultados en la rama principal.
7. Registra en la bitácora los comandos utilizados, la trazabilidad de versiones y una reflexión sobre la reproducibilidad obtenida.
8. Recibe retroalimentación docente.

Recursos de apoyo

- Entorno GNU/Linux con cuentas de prueba.
- Guía de práctica con ejercicios de automatización y control de versiones.
- Bitácora de actividades.
- Editor de texto para documentación.

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Fundamentos y arquitectura: de la filosofía unix a la infraestructura GNU/Linux

Práctica 1: Diagnóstico forense de un comportamiento anómalo del sistema

Duración

2 horas

Procedimiento

1. Atiende las orientaciones docentes y recibe el caso de un sistema con un comportamiento anómalo reportado.
2. Verifica el síntoma reportado (la incapacidad de actualizar el sistema) mediante la ejecución de comandos en el intérprete de comandos para reproducir el error.
3. Formula una hipótesis inicial sobre la causa raíz del problema basándose en los mensajes de error arrojados por el sistema.
4. Diseña una estrategia de investigación para confirmar o refutar su hipótesis, enfocándose en el análisis del árbol del sistema de archivos para localizar el agotamiento de recursos.
5. Ejecuta la estrategia diseñada para identificar la partición, el directorio y el archivo específico responsable del problema.
6. Investiga y correlaciona los procesos en ejecución con el archivo problemático para identificar el programa y el usuario causante del comportamiento.
7. Analiza los permisos del archivo y el contexto del proceso para determinar la naturaleza del problema (error, abuso, etc.).
8. Implementa un plan de remediación para restaurar la operatividad del sistema de forma segura.
9. Redacta un reporte técnico forense que documenta la metodología, la evidencia encontrada y el plan de acción implementado.
10. Entrega reporte técnico para su revisión.
11. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora con entorno GNU/Linux virtualizado y pre-configurado con la anomalía.
- Acceso a las páginas del manual del sistema (man).
- Editor de texto para la elaboración del reporte.

Unidad II. Gestión del sistema: todo es un archivo, hasta que es un proceso

Práctica 2: Diseño de un entorno colaborativo seguro

Duración

4 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el escenario de un equipo científico que requiere un entorno de trabajo multiusuario.
2. Diseña un esquema de usuarios, grupos y permisos que garantice la privacidad de los datos personales y la integridad de un directorio de proyecto compartido.
3. Implementa la estructura de usuarios y grupos en el sistema.
4. Aplica la configuración de permisos en la estructura de directorios utilizando los comandos de control de acceso.
5. Verifica la efectividad de la configuración intentando realizar acciones permitidas y prohibidas desde las diferentes cuentas de usuario.
6. Redacta un manual de usuario para el equipo científico que documenta la política de uso del entorno.
7. Entrega manual para su revisión.
8. Recibe retroalimentación docente.

Recursos de apoyo

- Entorno GNU/Linux con privilegios de administrador.
- Documento con el escenario y los requerimientos del equipo científico.
- Editor de texto para la elaboración del manual.

Práctica 3: Auditoría de recursos y gestión de software del sistema

Duración

4 horas

Procedimiento

1. Recibe el caso de un servidor que presenta un rendimiento degradado y una lista de nuevo software requerido para un proyecto.
2. Analiza el estado actual de los recursos del sistema (procesos, memoria, red y almacenamiento).
3. Identifica el proceso y el usuario que consumen la mayor cantidad de recursos y formula una hipótesis sobre la causa.
4. Investiga los requerimientos y dependencias del nuevo software a instalar.
5. Verifica la disponibilidad del software en los repositorios configurados y resuelve las dependencias.
6. Instala y configura el software solicitado, asegurando su correcto funcionamiento.
7. Elabora un reporte de auditoría que incluye el diagnóstico del problema de rendimiento y la bitácora de la gestión de software.
8. Entrega reporte de auditoría.
9. Recibe retroalimentación docente.

Recursos de apoyo

- Entorno GNU/Linux con un proceso simulando alto consumo de recursos.
- Listado de software a instalar con sus especificaciones.
- Editor de texto para la elaboración del reporte.

Unidad III. Texto: el lenguaje universal

Práctica 4: Filtrado y extracción de datos a partir de un genoma

Duración

4 horas

Procedimiento

1. Recibe un conjunto de datos genómicos en formato de texto plano (FASTA) y una serie de preguntas de investigación.
2. Inspecciona la estructura y el contenido del archivo de datos sin cargarlo completamente en memoria.
3. Diseña un flujo de trabajo basado en la composición de comandos y gramáticas de patrones regulares para responder a cada pregunta.
4. Ejecuta los flujos de trabajo para filtrar y extraer la información específica solicitada.
5. Sintetiza los resultados obtenidos en un reporte claro y conciso.
6. Entrega el reporte de resultados junto con la documentación de los flujos de comandos diseñados para obtenerlos.
7. Recibe retroalimentación docente.

Recursos de apoyo

- Entorno GNU/Linux.
- Archivo de datos en formato FASTA.
- Documento con las preguntas de investigación.

Práctica 5: Limpieza y conversión de bitácoras de autenticación a formato CSV

Duración

4 horas

Procedimiento

1. Recibe un archivo auth.log con registros de autenticación generados por el sistema.
2. Inspecciona la estructura de las entradas (fecha, hora, servicio, usuario, IP, mensaje).
3. Identifica inconsistencias o ruido en los registros (mensajes incompletos, saltos de línea, variación en campos).
4. Diseña una secuencia de transformaciones textuales (selección, sustitución, normalización) para limpiar los registros.
5. Extrae los campos relevantes (fecha, hora, usuario, origen, evento) y conviértelos en columnas delimitadas.

6. Genera un archivo de salida en formato CSV que pueda ser procesado por herramientas estadísticas.
7. Realiza una síntesis preliminar de resultados (ejemplo: intentos fallidos por usuario, accesos por dirección IP).
8. Entrega síntesis preliminar de resultados.
9. Recibe retroalimentación docente

Recursos de apoyo

- Entorno GNU/Linux con acceso a auth.log de ejemplo.
- Guía de referencia de expresiones regulares.
- Editor de texto o visor de archivos de log.

Unidad IV. Ciencia: la magia de la automatización y la reproducibilidad

Práctica 6: Automatización del flujo OSEMN sobre registros de autenticación

Duración

6 horas

Procedimiento

1. Retoma el archivo auth.log previamente transformado a formato CSV.
2. Recibe la descripción de un flujo de análisis de datos basado en las fases OSEMN (Obtain, Scrub, Explore, Model, iNterpret).
3. Diseña, en equipo, un script de sistema que automatice las cinco fases del flujo:
 - a. Obtain: lectura de uno o varios archivos CSV.
 - b. Scrub: limpieza adicional de campos y estandarización de datos.
 - c. Explore: generación de estadísticas descriptivas (intentos fallidos por usuario, IPs frecuentes, horarios críticos).
 - d. Model: aplicación de un modelo sencillo (por ejemplo, detección de outliers o agrupamiento básico) para identificar patrones relevantes.
 - e. iNterpret: síntesis de resultados y hallazgos expresados en un reporte legible.
4. Parametriza el script para aceptar diferentes archivos de entrada y rangos de fechas como argumentos.
5. Inicializa un repositorio Git y divide el trabajo colaborativo en ramas:
 - a. Una rama por cada fase del flujo.
 - b. Una rama principal para integrar y validar el resultado final.
6. Documenta cada cambio significativo con mensajes claros y consistentes en el historial del repositorio.
7. Fusiona las ramas en la rama principal y valida que el script ejecuta el flujo de análisis de manera reproducible.
8. Elabora un archivo README.md con:
 - a. Descripción del flujo OSEMN aplicado a auth.log.
 - b. Instrucciones de uso del script.
 - c. Ejemplos de entrada y salida.
 - d. Breve interpretación de resultados apoyada en el modelo implementado.

Recursos de apoyo

- Entorno GNU/Linux con Git instalado.
- Archivo auth.log convertido a CSV.
- Documento de referencia con las fases OSEMN.
- Plataforma de repositorios remotos (GitHub, GitLab, etc.).

Práctica 7: Encapsulación de un entorno científico con contenedores

Duración

4 horas

Procedimiento

1. Recibe un proyecto científico (script y datos) que depende de versiones específicas de software y librerías.
2. Analiza el proyecto para identificar todas sus dependencias.
3. Diseña la especificación de un entorno reproducible utilizando una tecnología de contenedores
4. Construye el contenedor y valida que el entorno resultante sea funcional.
5. Ejecuta el análisis dentro del contenedor y comprueba que produce los resultados esperados de forma idéntica.
6. Prepara un archivo de documentación que incluya las dependencias y el procedimiento de construcción y ejecución del contenedor.
7. Entrega archivo de documentación.
8. Recibe retroalimentación docente

Recursos de apoyo

- Entorno GNU/Linux con soporte para tecnologías de contenedores.
- Proyecto científico con dependencias específicas.
- Acceso a un registro de imágenes de contenedores.
- Guía de referencia con la sintaxis del archivo del contenedor.

Práctica 8: Despliegue y comunicación de un servicio científico contenerizado

Duración

4 horas

Procedimiento

1. Atiende las orientaciones docentes y retoma el proyecto contenedorizado y funcional de la práctica anterior.
2. Diseña una modificación al proyecto para que exponga sus resultados a través de un servicio de red básico (ej. un micro-servidor web).
3. Ajusta la configuración del contenedor (Dockerfile) para instalar las dependencias de red y abrir el puerto necesario.

4. Reconstruye y ejecuta el contenedor, mapeando el puerto del contenedor al sistema anfitrión.
5. Verifica que el servicio es accesible desde el exterior del contenedor (ej. a través de un navegador).
6. Actualiza la documentación para incluir la arquitectura del servicio de red y las instrucciones de despliegue.
7. Entrega la documentación actualizada.
8. Recibe retroalimentación docente.

Recursos de apoyo

- Entorno GNU/Linux con soporte para tecnologías de contenedores.
- Proyecto y contenedor funcional de la práctica anterior.
- Guía de referencia sobre servicios de red básicos y la sintaxis del archivo del contenedor.

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer de forma clara y formal:

- La metodología de trabajo.
- Criterios de evaluación y rúbricas.
- Reglas sobre la calidad y entrega de trabajos académicos.
- Derechos y obligaciones del o la docente y del estudiantado

Esto debe quedar registrado y comunicado por medios institucionales (como Blackboard o correo institucional).

Estrategia de enseñanza

- Método de proyectos
- Aprendizaje basado en problemas
- Técnica expositiva
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	20%
– Prácticas taller	15%
– Prácticas de laboratorio	25 %
– Presentaciones	10%
– Entorno reproducible	30%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Blum, R., & Bresnahan, C. (2022). *Comptia: Linux+ Study Guide, exam XK0-005* (5th ed.). Sybex.
- Blum, R., & Bresnahan, C. (2021). *Linux Command Line and Shell Scripting Bible* (4th ed.). Wiley.
- Dakic, V., & Redzepagic, J. (2022). *Linux command line and shell scripting techniques: Master practical aspects of the Linux command line and then use it as a part of the shell scripting process*. Packt Publishing. eBook Academic Collection (EBSCOhost).
- Gale, J. (2023). *Linux: The complete manual*. Gale. eBook Academic Collection (EBSCOhost).
- Nemeth, E., Snyder, G., Hein, T. R., Whaley, B., & Mackin, D. (2018). *UNIX and Linux System Administration Handbook* (5th ed.). Addison-Wesley. [clásica].
- Tevault, D. A. (2022). *Linux service management made easy with systemd: Advanced techniques to effectively manage, control, and monitor Linux systems and services*. Packt Publishing. eBook Academic Collection (EBSCOhost).

Recursos de acceso libre o complementarios externos

- Hausenblas, M. (2022). *Learning Modern Linux*. O'Reilly.
- Janssens, J. (2021). *Data Science at the Command Line* (2nd ed.). O'Reilly.

Kerr, D. (2024). *Effective Shell*. No Starch Press.

Raymond, E. S. (2003). *The Art of Unix Programming*. Addison-Wesley.

Shotts, W. (2019). *The Linux Command Line: A Complete Introduction* (2nd ed.). No Starch Press. Recuperado de <https://linuxcommand.org/tlcl.php>. [clásica].

Ward, B. (2021). *How Linux Works: What Every Superuser Should Know* (3rd ed.). No Starch Press.

XII. PERFIL DOCENTE

Licenciatura en Ciencias Computacionales, Ingeniería en Computación, o áreas afines, preferentemente con estudios de posgrado en un área de alta demanda computacional. Debe poseer experiencia práctica demostrable en la administración de sistemas GNU/Linux y en el desarrollo de flujos de trabajo científicos. Con experiencia docente de mínimo dos años frente a grupo, además de poseer competencias pedagógicas que favorezcan el aprendizaje activo, la resolución de problemas y el trabajo colaborativo en contextos de desarrollo tecnológico.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Introducción a las Ciencias Computacionales
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 02 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:06
Etapas de formación:	Básica
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Adrián Enciso Almanza José Ángel González Fraga Everardo Gutiérrez López
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Tiene como finalidad proporcionar a las y los estudiantes una visión integral de las Ciencias Computacionales, desde su evolución histórica hasta las tendencias emergentes, dotándolo de un lenguaje técnico y crítico que le permita comprender los fundamentos de los sistemas computacionales y su impacto en la sociedad, incentivando el interés por la investigación y la innovación tecnológica. Esta asignatura es de carácter obligatorio, se ubica en la etapa básica y pertenece al área de conocimiento de Entorno Social.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Analizar los fundamentos, áreas y tendencias de las Ciencias Computacionales, a través del estudio de teorías, arquitecturas y tecnologías emergentes, para comprender su evolución, funcionamiento y proyección futura, desarrollando un pensamiento crítico, responsable y con ética profesional frente al uso de la tecnología.

IV. EVIDENCIA DE APRENDIZAJE

Proyecto final de investigación y exposición sobre un tema actual de las Ciencias Computacionales, que integre antecedentes, análisis, aplicación y conclusiones, demostrando dominio conceptual, claridad en la comunicación y el uso de recursos digitales, con actitud crítica, ética y responsabilidad en el trabajo académico.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Compromiso con la responsabilidad social, Excelencia, Vanguardia e Innovación Social.

VII. DESARROLLO POR UNIDADES

Unidad I. Evolución y fundamentos de la computación

Comprender la evolución de la computación desde sus orígenes hasta los paradigmas modernos, mediante la revisión de antecedentes históricos, líneas del tiempo, modelos y análisis comparativos, para comprender cómo han cambiado las arquitecturas y enfoques del cómputo, con una actitud innovadora y respeto al legado científico.

Contenido

- 1.1. Computación primitiva y dispositivos mecánicos: ábacos, máquinas mecánicas y primeras calculadoras
- 1.2. Generaciones de las computadoras y principales innovaciones tecnológicas
- 1.3. El modelo de Von Neumann y su influencia en la arquitectura moderna
- 1.4. Paradigmas modernos de cómputo (paralelo, distribuido, cuántico, en la nube, entre otros)

Duración

4 horas

Unidad II. Representación y funcionamiento de los sistemas computacionales

Competencia

Analizar la representación de la información y el funcionamiento de los principales componentes de un sistema de cómputo, por medio de ejercicios de sistemas de numeración y de configuración de hardware y software, para comprender la lógica básica de los sistemas modernos, con responsabilidad y apertura al trabajo colaborativo.

Contenido

- 2.1. Representación de la información: codificación de datos, caracteres e instrucciones en formato binario: unidades de medida de la información
- 2.2. Arquitectura básica de un sistema de cómputo: componentes principales (CPU, memoria principal, dispositivos de entrada/salida) y su interacción en el procesamiento de datos
- 2.3. Sistemas de numeración: conversión y operaciones fundamentales en bases binaria, octal, decimal y hexadecimal
- 2.4. Software de sistema: funciones del sistema operativo y de los programas de sistema en la gestión de recursos y comunicación con el hardware

Duración

8 horas

Unidad III. Lenguajes, programación y autómatas

Competencia

Explicar el papel de los lenguajes de programación y los modelos teóricos como la máquina de Turing, mediante simulaciones y revisión de ejemplos de compiladores e intérpretes, para comprender la importancia del software en la resolución de problemas computacionales, con respeto al trabajo académico y compromiso con el aprendizaje.

Contenido

- 3.1. Fundamentos teóricos de la computación: conceptos introductorios de autómatas finitos y de la máquina de Turing como modelos de procesamiento de información
- 3.2. Traducción y ejecución de programas: fases del proceso de programación, funcionamiento general de compiladores e intérpretes
- 3.3. Jerarquía y paradigmas de lenguajes de programación: niveles de abstracción, tipos de lenguaje y su relación con el hardware y el software
- 3.4 Tendencias y aplicaciones actuales: lenguajes y entornos emergentes en inteligencia artificial, ciencia de datos y desarrollo web

Duración

8 horas

Unidad IV. Comunicación y redes de computadoras

Competencia

Revisar los fundamentos de la comunicación de datos y el funcionamiento general de las redes de computadoras, mediante la presentación de ejemplos y esquemas de los elementos que intervienen en su conexión, para ilustrar cómo se establecen las conexiones y se transmite la información entre los sistemas, con responsabilidad y disposición al trabajo colaborativo.

Contenido

- 4.1. Fundamentos de la comunicación de datos: concepto de comunicación digital, elementos del proceso comunicativo y características básicas de la transmisión de información
- 4.2. Principios de organización de redes de computadoras: definición y propósito de las redes: noción de arquitectura, topología y tipos de red (LAN, WAN, Internet)
- 4.3. Componentes y medios de comunicación: dispositivos, canales físicos y lógicos que permiten la conexión e intercambio de datos entre sistemas
- 4.4. Protocolos y servicios de comunicación: concepto de protocolo; ejemplos esenciales de intercambio de información y aplicaciones representativas
- 4.5. Seguridad y ética en entornos conectados: conceptos básicos de seguridad informática, privacidad, uso responsable y ciudadanía digital.

Duración

6 horas

Unidad V. Tendencias y aplicaciones de las Ciencias Computacionales***Competencia***

Explorar las principales tendencias y aplicaciones modernas de las Ciencias Computacionales, mediante investigación documental, análisis de casos y la experimentación con herramientas de IA, para comprender su utilidad, relevancia y su impacto en la sociedad, con ética profesional y pensamiento crítico y creativo.

Contenido

- 5.1. Inteligencia Artificial Generativa y Agentes Autónomos: principios básicos, aplicaciones cotidianas y desafíos éticos asociados al uso de modelos generativos y agentes inteligentes
- 5.2. Realidad Extendida (XR): Realidad Aumentada y Virtual (fundamentos y usos actuales de la interacción inmersiva en educación, ciencia y entretenimiento)
- 5.3. Ciberseguridad y Ética Digital: conceptos esenciales de seguridad informática, privacidad de datos, responsabilidad en línea y ciudadanía digital
- 5.4. Computación cuántica y sus aplicaciones: conceptos fundamentales de la computación cuántica y ejemplos de su potencial en criptografía, optimización y simulación

Duración

6 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Evolución y fundamentos de la computación

Práctica 1: Línea de tiempo de la evolución de la computación

Duración:

2 horas

Procedimiento:

1. Atiende orientaciones para el desarrollo de práctica.
2. Investiga los hitos históricos de la computación.
3. Elabora una línea de tiempo digital que muestre desde los dispositivos mecánicos hasta los paradigmas modernos.
4. Presenta la línea de tiempo al grupo y explica los hitos más importantes.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Acceso a Internet.
- Software para presentaciones.
- Software crear líneas de tiempo (Canva, Timeline JS).

Práctica 2: Clasificación de generaciones de computadoras

Duración:

2 horas

Procedimiento:

1. Atiende orientaciones para el desarrollo de la práctica.
2. Investiga las características de las cinco generaciones de computadoras.
3. Elabora una tabla comparativa con componentes, tamaño, lenguaje, velocidad y ejemplos.
4. Socializa resultados en equipos pequeños.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Acceso a internet.
- Procesador de texto, hoja de cálculo o software gratuito para construcción de cuadros sinópticos.

Práctica 3: Modelo de Von Neumann

Duración:

2 horas

Procedimiento:

1. Atiende orientaciones para el desarrollo de la práctica.
2. Investiga los elementos del modelo de Von Neumann.
3. Realiza un diagrama explicativo con etiquetas y funciones.
4. Presenta el diagrama al grupo.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Acceso a internet.
- Software de diagramas (Lucidchart, Draw.io).

Práctica 4: Análisis de paradigmas modernos de computación

Duración:

2 horas

Procedimiento:

1. Atiende orientaciones para el desarrollo de la práctica.
2. Investiga los paradigmas modernos: paralelo, distribuido y cuántico.
3. Elabora una tabla comparativa con ventajas, desventajas y aplicaciones.
4. Comparte conclusiones con el grupo.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Acceso a internet.
- Procesador de texto.

Unidad II. Representación y funcionamiento de sistemas computacionales

Práctica 5: Conversión entre sistemas de numeración

Duración:

1 hora

Procedimiento:

1. Atiende orientaciones para el desarrollo de la práctica.

2. Resuelve ejercicios de conversión entre binario, decimal, octal y hexadecimal.
3. Explica los procedimientos usados.
4. Compara resultados con compañeros.
5. Recibir retroalimentación docente.

Recursos de apoyo:

- Calculadora.
- Acceso a internet.
- Hoja de cálculo.

Práctica 6: Componentes de un sistema de cómputo

Duración:

2 horas

Procedimiento:

1. Atiende orientaciones para el desarrollo de la práctica.
2. Identifica CPU, memoria, dispositivos de entrada/salida en su computadora.
3. Elabora un esquema con las funciones de cada componente.
4. Presenta al grupo el esquema y explica el flujo de datos.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Acceso a internet.
- Software de diagramas

Práctica 7: Operaciones básicas en binario

Duración:

1 hora

Procedimiento:

1. Atiende orientaciones para el desarrollo de la práctica.
2. Resuelve ejercicios de suma, resta, multiplicación y división en binario.
3. Documenta los pasos de resolución.
4. Comparte resultados y resolver dudas en grupo.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Calculadora binaria.
- Acceso a internet.

Unidad III. Lenguajes, programación y autómatas

Práctica 8: Simulación de máquina de Turing

Duración:

1 hora

Procedimiento:

1. Atiende orientaciones para el desarrollo de la práctica.
2. Explora un simulador de máquina de Turing en línea.
3. Programa una máquina que incremente un número binario.
4. Presenta y explica el funcionamiento.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Simulador de máquina de Turing.
- Acceso a internet.

Práctica 9: Construcción de un autómata finito

Duración:

1 hora

Procedimiento:

1. Atiende orientaciones para el desarrollo de la práctica.
2. Diseña un autómata que reconozca cadenas con cierta regla.
3. Dibuja el diagrama de estados y probarlo.
4. Explica el resultado en grupo.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Herramientas de diagramas.
- Simulador de autómatas en línea.

Práctica 10: Exploración de un compilador

Duración:

2 horas

Procedimiento:

1. Atiende orientaciones para el desarrollo de la práctica.
2. Analiza el proceso de compilación de un programa sencillo.
3. Identifica las fases: análisis léxico, sintáctico, semántico, generación de código.
4. Documenta el proceso.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Entorno de desarrollo (IDE).
- Acceso a internet.

Práctica 11: Exploración de un intérprete**Duración:**

2 horas

Procedimiento:

1. Atiende orientaciones para el desarrollo de la práctica.
2. Ejecuta un script en un lenguaje interpretado (Python).
3. Observa ejecución línea por línea y errores comunes.
4. Comparte hallazgos con el grupo.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Acceso a internet.
- Intérprete de Python.

Práctica 12: Jerarquía de lenguajes de programación**Duración:**

2 horas

Procedimiento:

1. Atiende orientaciones para el desarrollo de la práctica.
2. Investiga los niveles de lenguajes (máquina, bajo, medio, alto nivel).
3. Elabora un mapa jerárquico con ejemplos.
4. Presenta el mapa al grupo.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Acceso a internet.
- Software de mapas conceptuales.

Práctica 13: Tendencias actuales en lenguajes**Duración:**

2 horas

Procedimiento:

1. Atiende orientaciones para el desarrollo de la práctica.
2. Investiga lenguajes populares en IA, ciencia de datos y desarrollo web.
3. Elabora una tabla con sus características y usos.
4. Socializa conclusiones en grupo.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Acceso a internet.
- Procesador de texto.

Práctica 14: Mini proyecto de programación**Duración:**

2 horas

Procedimiento:

1. Atendiendo orientaciones para el desarrollo de la práctica.
2. Diseña un programa sencillo (calculadora, conversor, etc.).
3. Codifica, ejecuta y corrige errores.
4. Presenta el proyecto y su código.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Entorno de desarrollo.
- Acceso a internet

Unidad IV. Comunicación y redes de computadoras

Práctica 15: Simulación de transmisión de datos

Duración:

2 horas

Procedimiento:

1. Atiende orientaciones para el desarrollo de la práctica.
2. Investiga conceptos de señal, canal y transmisión.
3. Simula el envío de datos con diferentes medios (cables, ondas).
4. Explica el resultado.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Acceso a internet.
- Simulador de redes.

Práctica 16: Tipos de redes de computadoras

Duración:

1 hora

Procedimiento:

1. Atiende orientaciones para el desarrollo de la práctica.
2. Investiga LAN, WAN, MAN e Internet.
3. Elabora un cuadro comparativo de alcance, velocidad y costo.
4. Presenta hallazgos al grupo.
5. Recibir retroalimentación docente.

Recursos de apoyo:

- Acceso a internet.
- Procesador de texto.

Práctica 17: Elementos de una red

Duración:

1 hora

Procedimiento:

1. Atiende orientaciones para el desarrollo de la práctica.
2. Identifica dispositivos y medios de transmisión en un esquema de red.
3. Elabora un diagrama etiquetado con funciones.
4. Comparte el diagrama y explica las conexiones.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Acceso a internet.
- Software de diagramas.

Unidad V. Tendencias y aplicaciones de las Ciencias Computacionales**Práctica 18: Exploración de herramientas de IA generativa****Duración:**

2 horas

Procedimiento:

1. Atender orientaciones para el desarrollo de la práctica.
2. Prueba una herramienta de IA generativa (texto, imagen, código).
3. Analiza ventajas, limitaciones y posibles usos educativos.
4. Comparte resultados en grupo.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Acceso a internet.
- Plataforma de IA generativa.

Práctica 19: Tendencias emergentes en Ciencias Computacionales**Duración:**

2 horas

Procedimiento:

1. Atiende orientaciones para el desarrollo de la práctica.
2. Investiga brevemente computación cuántica, XR y ciberseguridad.
3. Elabora una infografía sobre su impacto social.
4. Presenta al grupo.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Acceso a internet.
- Software de diseño (Canva, Genially).

IX. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer de forma clara y formal:

- La metodología de trabajo.
- Criterios de evaluación y rúbricas.
- Reglas sobre la calidad y entrega de trabajos académicos.
- Derechos y obligaciones del o la docente y del estudiantado.

Esto debe quedar registrado y comunicado por medios institucionales (como Blackboard o correo institucional).

Estrategia de enseñanza

- Técnica expositiva.
- Discusión guiada.
- Estudio de caso.
- Aprendizaje basado en problemas (ABP).
- Instrucción guiada.

Estrategia de aprendizaje

- Investigación documental.
- Exposición de resoluciones de casos prácticos.
- Trabajo colaborativo.
- Técnica expositiva.
- Mapas mentales y conceptuales.
- Infografías.

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

- Evaluaciones parciales	50%
- Prácticas de laboratorio	30%
- Proyecto final	20%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Aguilar, L. J. (2021). *Internet de las cosas : un futuro hiperconectado : 5G, Inteligencia artificial, Big Data, Cloud, Blockchain, Ciberseguridad*. Alfaomega
- Brookshear, J. G. & Brylow, D. (2020). *Computer Science: An Overview*. Pearson.
- Tanenbaum, A. S. & Bos, H. (2015). *Modern Operating Systems*. Pearson. [clásica]
- Patterson, D. A. & Hennessy, J. L. (2017). *Computer Organization and Design: The Hardware/Software Interface*. Morgan Kaufmann. [clásica]
- Stallings, W. (2017). *Computer Organization and Architecture*. Pearson. [clásica]
- Ullman, J. D. & Hopcroft, J. E. (2007). *Introduction to Automata Theory, Languages, and Computation*. Pearson. [clásica]

Recursos de acceso libre o complementarios externos

- Association for Computing Machinery (ACM). (s. f.). *ACM Digital Library*. <https://dl.acm.org> [ACM Digital Library+2libraries.acm.org+2](https://dl.acm.org)
- Computer History Museum. (s. f.). *Computer History Museum*. <https://computerhistory.org> [CHM+1](https://computerhistory.org)
- Coursera. (s. f.). *Introducción a Ciencias de la Computación (cursos introductorios gratuitos)*. <https://www.coursera.org/browse/computer-science> [Coursera+1](https://www.coursera.org/browse/computer-science)
- CS Unplugged Project. (s. f.). *CS Unplugged – Recursos didácticos para enseñar fundamentos de la computación*. <https://csunplugged.org> [csunplugged.org+1](https://csunplugged.org)
- Institute of Electrical and Electronics Engineers (IEEE). (s. f.). *IEEE Xplore Digital Library*. <https://ieeexplore.ieee.org> [theiet.org+1](https://ieeexplore.ieee.org)
- Khan Academy. (s. f.). *Khan Academy – Computer Science*. <https://www.khanacademy.org/computing> [Khan Academy+1](https://www.khanacademy.org/computing)

National Institute of Standards and Technology (NIST). (s. f.). *NIST Cybersecurity Framework – Material introductorio a ciberseguridad*. <https://www.nist.gov/cyberframework>

XII. PERFIL DOCENTE

Licenciatura en Ciencias Computacionales o áreas afines, con conocimientos en fundamentos de la computación, programación, redes y tendencias tecnológicas emergentes; capaz de aplicar estrategias didácticas innovadoras, fomentar el pensamiento crítico y motivar a las y los estudiantes con ética profesional, respeto e inclusión hacia el estudio de las Ciencias Computacionales.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Estructura de Datos y Algoritmos
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 03 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos: 07
Etapas de formación:	Básica
Carácter:	Obligatoria
Requisito:	Introducción a la Programación
Modalidad de impartición:	Presencial
Equipo de diseño:	Alma Rocio Cabazos Marin, Everado Gutiérrez López, Evelio Martínez Martínez
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

El propósito de la unidad de aprendizaje Estructura de Datos y Algoritmos es desarrollar en el estudiantado la capacidad de diseñar, analizar e implementar soluciones computacionales eficientes mediante el uso adecuado de estructuras de datos y técnicas algorítmicas, fomentando el pensamiento lógico, la optimización de recursos y la resolución de problemas complejos, constituyendo así una base fundamental para asignaturas avanzadas y para el desempeño profesional en el ámbito de la ingeniería y el desarrollo de software. Se imparte en la etapa básica, es de carácter obligatorio y pertenece al área de conocimiento de Programación e Ingeniería de Software.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Desarrollar soluciones computacionales eficientes, mediante la selección e implementación de estructuras de datos y algoritmos adecuados, métodos de fundamentos científicos, modelos formales de cómputo, técnicas de programación y buenas prácticas de diseño, con el fin de resolver problemas de manera óptima y escalable, con responsabilidad, pensamiento crítico y espíritu de mejora continua.

IV. EVIDENCIA DE APRENDIZAJE

La evidencia de aprendizaje consistirá en:

- Portafolio de prácticas de laboratorio que apliquen correctamente diversas estructuras de datos y algoritmos, acompañados de un reporte técnico que incluya la descripción de la solución propuesta.
- Proyecto final desarrollado en un lenguaje de programación de medio o alto nivel para resolver problemas selectos, incluyendo el análisis de los resultados obtenidos así como la justificación del uso de las estructuras y técnicas algorítmicas empleadas en la resolución de las problemáticas planteadas.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia, Vanguardia e Innovación Social

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos de las Estructuras de Datos y Algoritmos

Competencia

Distinguir las estructuras de datos básicas mediante su representación formal, la utilización de sus operaciones fundamentales y selección de sus contextos de uso, para resolver problemas computacionales simples y fomentar habilidades de abstracción, con una actitud creativa, de pensamiento crítico y lógico.

Contenido

- 1.1. Conceptos fundamentales de representación y abstracción de datos
 - 1.1.1. Datos estructurados, semiestructurados y no estructurados
 - 1.1.2. Tipos Abstractos de Datos (TAD)
 - 1.1.3. Diferencias entre TAD, estructura de datos e implementación
 - 1.1.4. Selección de representaciones según el dominio
- 1.2. Modelos de cómputo y costo computacional
 - 1.2.1. Modelo RAM y su relevancia
 - 1.2.2. M caché, jerarquías y su impacto práctico en el rendimiento
 - 1.2.3. Operaciones elementales y costo real vs. costo teórico
 - 1.2.4. Análisis de rendimiento en arquitecturas modernas (CPU multinúcleo y compiladores modernos)
- 1.3. Aplicación integradora de la unidad
 - 1.3.1. Diseño experimental: hipótesis, variables, métricas y repetibilidad
 - 1.3.2. Instrumentación, benchmarking y profiling con herramientas contemporáneas
 - 1.3.3. Adquisición y análisis de datos experimentales: tiempo real vs. tiempo de CPU
 - 1.3.4. Comparación de dos algoritmos simples (p. ej., búsqueda lineal vs. binaria) mediante teoría + datos reales
 - 1.3.5. Implementación en un lenguaje de programación

Duración

06 horas

Unidad II. Estructuras de Datos Lineales

Competencia

Implementar estructuras de datos lineales considerando patrones de acceso, complejidad y manejo eficiente de memoria, para resolver problemas que requieren secuencias organizadas de información bajo criterios de eficiencia y claridad conceptual, con pensamiento crítico, creatividad y responsabilidad.

- 2.1. Arreglos estáticos y estructuras dinámicas
 - 2.1.1. Representación en memoria
 - 2.1.2. Redimensionamiento
 - 2.1.3. Registros y tuplas
 - 2.1.4. Conjuntos
- 2.2. Listas enlazadas
 - 2.2.1. Listas simples, dobles y circulares
 - 2.2.2. Invariantes y operaciones básicas
- 2.3. Pilas
 - 2.3.1. Implementaciones y usos
 - 2.3.2. Aplicaciones prácticas (p. ej. en compiladores y recorridos)
- 2.4. Colas
 - 2.4.1. Colas lineales y circulares
 - 2.4.2. Colas de prioridad simples

Duración

08 horas

Unidad III. Estructuras jerárquicas y de búsqueda

Competencia

Comparar estructuras jerárquicas y mecanismos de búsqueda eficientes, mediante el análisis de sus propiedades estructurales y su impacto computacional, con el fin de resolver problemas que demandan organización avanzada y acceso optimizado a datos, con responsabilidad y trabajo colaborativo.

Contenido

- 3.1. Árboles generales y árboles binarios
 - 3.1.1. Representación y recorrido (preorden, inorden, postorden, BFS)
 - 3.1.2. Árboles binarios completos y casi completos
- 3.2. Árboles balanceados
 - 3.2.1. Árboles AVL: rotaciones, inserción y eliminación
 - 3.2.2. Árboles rojo-negro: propiedades y operaciones
 - 3.2.3. Análisis de altura y garantías de rendimiento
- 3.3. Estructuras de montículo (heaps)
 - 3.3.1. Heap binario y heap d-ario
 - 3.3.2. Inserción, extracción y *heapify*
 - 3.3.3. Aplicaciones: colas de prioridad, *heapsort*
- 3.4. Tabla hash y hashing avanzado
 - 3.4.1. Funciones hash, colisiones y técnicas de resolución (encadenamiento, direccionamiento abierto)

3.4.2. Análisis de carga y complejidad promedio

3.4.3. Hashing universal y aplicaciones prácticas

Duración

08 horas

Unidad IV. Algoritmos Esenciales - Fundamentales

Competencia

Evaluar algoritmos esenciales de ordenación, búsqueda y grafos, integrando razonamiento analítico y experimentación sobre estructuras de datos para seleccionar y aplicar estrategias algorítmicas adecuadas a la resolución de problemas selectos, con una actitud analítica, propositiva y colaborativa.

Contenido

- 4.1. Algoritmos de ordenación
 - 4.1.1. Quicksort, mergesort, heapsort
 - 4.1.2. Estabilidad y análisis empírico
- 4.2. Algoritmos de búsqueda
 - 4.2.1. Búsqueda secuencial y binaria
 - 4.2.2. Búsquedas basadas en hash
- 4.3. Algoritmos en grafos
 - 4.3.1. BFS y DFS
 - 4.3.2. Dijkstra y A*
 - 4.3.3. Prim y Kruskal
- 4.4. Aplicaciones
 - 4.4.1. Compresión con árboles de Huffman
 - 4.4.2. Aplicaciones de grafos y árboles

Duración

10 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Fundamentos de las Estructuras de Datos y Algoritmos

Práctica 1: Análisis Empírico de Operaciones Básicas

Duración:

04 horas

Procedimiento:

1. Atiende orientaciones docentes para el desarrollo de la práctica.
2. Implementa operaciones básicas (acceso, inserción, eliminación) en un arreglo y una lista enlazada simple.
3. Instrumenta el código para medir tiempos de ejecución en diferentes tamaños de entrada.
4. Ejecuta cada operación múltiples veces para obtener promedios.
5. Gráfica resultados comparando operación vs. tamaño de entrada.
6. Analiza discrepancias entre el comportamiento teórico y el empírico.

Recursos de apoyo:

- Editor/IDE.
- Bibliotecas para medición y graficado.
- Plantillas de código base proporcionadas por el docente.

Práctica 2: Modelos de Cómputo y Jerarquías de Memoria

Duración:

04 horas

Procedimiento:

1. Atiende orientaciones docentes para el desarrollo de la práctica.
2. Desarrolla un programa que recorra un arreglo en diferentes patrones (secuencial, saltos fijos, bidireccional).
3. Observa variaciones en el tiempo debido a la localidad espacial y temporal.
4. Evalúa cómo el patrón de acceso afecta la cantidad de fallos de caché (usando herramientas del sistema o un profiler).
5. Redacta un breve reporte conectando los resultados con el modelo RAM y las limitaciones de la noción de “operaciones de costo constante”.
6. Recibe retroalimentación docente.

Recursos de apoyo:

- Editor/IDE.
- Bibliotecas para medición y graficado.
- Plantillas de código base proporcionadas por el docente.
- Conjuntos de prueba de diferentes tamaños.

Unidad II. Estructuras de Datos Lineales

Práctica 3: Implementación y Evaluación de Listas Enlazadas

Duración:

06 horas

Procedimiento:

1. Atiende orientaciones docentes para el desarrollo de la práctica.
2. Implementa una lista enlazada simple con métodos de inserción, eliminación y búsqueda.
3. Compara su rendimiento frente a un arreglo dinámico para operaciones en inicio, medio y final.
4. Elabora una tabla de tiempos para cada operación bajo distintos tamaños de entrada.
5. Evalúa impacto en localidad de referencia mediante patrones de acceso.
6. Recibe retroalimentación docente.

Recursos de apoyo:

- Editor/IDE.
- Bibliotecas para medición y graficado.
- Plantillas de código base proporcionadas por el docente.
- Scripts de benchmarking automatizados.
- Diagrama conceptual de punteros e invariantes.

Práctica 4: Pilas y Colas con Aplicaciones

Duración:

06 horas

Procedimiento:

1. Atiende orientaciones docentes para el desarrollo de la práctica.
2. Implementa una pila y una cola usando tanto arreglos como listas enlazadas.
3. Aplica la pila para evaluar una expresión postfija simple.
4. Aplica la cola para simular un sistema de atención (llegadas, salidas y tiempos).
5. Compara la eficiencia entre implementaciones y justificar la selección en cada caso de uso.
6. Recibe retroalimentación docente.

Recursos de apoyo:

- Editor/IDE.

- Bibliotecas para medición y graficado, Plantillas de código base proporcionadas por el docente.

Unidad III: Estructuras de datos no lineales

Práctica 5: Árboles Binarios: Construcción y Recorridos

Duración:

06 horas

Procedimiento:

1. Atiende orientaciones docentes para el desarrollo de la práctica.
2. Implementa un árbol binario con inserción simple.
3. Programa funciones para recorridos: preorden, inorden, postorden y BFS.
4. Genera árboles de distintos tamaños y compara el costo de cada recorrido.
5. Analiza cómo la forma del árbol afecta el rendimiento.
6. Recibe retroalimentación docente.

Recursos de apoyo:

- Editor/IDE.
- Bibliotecas para medición y graficado.
- Plantillas de código base proporcionadas por el docente.
- Diagrama guía de nodos y punteros.
- Datos sintéticos para generar árboles balanceados y degenerados.

Práctica 6: Tablas Hash y Resolución de Colisiones

Duración:

06 horas

Procedimiento:

1. Atiende orientaciones docentes para el desarrollo de la práctica.
2. Implementa una tabla hash con encadenamiento separado.
3. Implementa una tabla hash con direccionamiento abierto (lineal y cuadrático).
4. Inserta grandes volúmenes de claves y mide el tiempo de inserción/búsqueda.
5. Evalúa el impacto del factor de carga y compara técnicas de colisión.
6. Recibe retroalimentación docente.

Recursos de apoyo:

- Editor/IDE.
- Bibliotecas para medición y graficado
- Plantillas de código base proporcionadas por el docente.
- Conjunto de claves aleatorias y pseudoaleatorias.
- Documento de apoyo sobre funciones hash y propiedades.

Unidad IV: Algoritmos Fundamentales

Práctica 7: Comparación Experimental de Algoritmos de Ordenación

Duración:

08 horas

Procedimiento:

1. Atiende orientaciones docentes para el desarrollo de la práctica.
2. Implementar quicksort, mergesort y heapsort.
3. Diseñar un conjunto de pruebas con datos aleatorios, ordenados y casi ordenados.
4. Mide el tiempo de ejecución y número estimado de comparaciones (instrumentadas).
5. Grafica resultados y analizar condiciones donde cada algoritmo es más eficiente.
6. Recibe retroalimentación docente.

Recursos de apoyo:

- Editor/IDE.
- Bibliotecas para medición y graficado.
- Plantillas de código base proporcionadas por el docente.

Práctica 8: Algoritmos en Grafos: Caminos Mínimos

Duración:

08 horas

Procedimiento:

1. Atiende orientaciones docentes para el desarrollo de la práctica.
2. Representa un grafo usando listas de adyacencia.
3. Implementa BFS para grafos no ponderados.
4. Implementa Dijkstra para grafos ponderados positivos.
5. Aplica ambos algoritmos a un conjunto de grafos de ejemplo y compara tiempos y rutas encontradas.
6. Presenta un breve análisis justificando cuándo es pertinente cada algoritmo.
7. Recibe retroalimentación docente.

Recursos de apoyo:

- Editor/IDE.
- Bibliotecas para medición y graficado.
- Plantillas de código base proporcionadas por el docente.
- Instancias de grafos (matrices o archivos .txt), Documento guía sobre BFS y Dijkstra.

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer de forma clara y formal:

- La metodología de trabajo.
- Criterios de evaluación y rúbricas.
- Reglas sobre la calidad y entrega de trabajos académicos.
- Derechos y obligaciones del docente y los estudiantes.

Esto debe quedar registrado y comunicado por medios institucionales (como Blackboard o correo institucional).

Estrategia de enseñanza

- Aprendizaje basado en problemas
- Técnica expositiva
- Discusión guiada
- Instrucción guiada

Estrategia de aprendizaje

- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Mapas mentales
- Mapas conceptuales
- Cuadros comparativos
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	30%
– Portafolio de prácticas de laboratorio	40%
– Proyecto final	30%
Total	10

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). *Introduction to algorithms* (4th ed.). MIT Press.
- Lafore, R. (2018). *Data structures and algorithms in Java* (2nd ed.). Sams Publishing.[clásica]
- Raza Khan, N. (2023). *Data Structure and Algorithm* (E-book edition). Arcler Press. [eBook]
- Sedgewick, R., & Wayne, K. (2011). *Algorithms* (4th ed.). Addison-Wesley. [clásica]
- Weiss, M. A. (2019). *Data Structures and Algorithm Analysis in Java* (3ª ed.). Pearson.

Recursos de acceso libre o complementarios externos

- ACM Special Interest Group on Algorithms & Computation Theory. (s.f.)-
<https://www.acm.org/special-interest-groups/sigs/sigact>
- Aho, A. V., Hopcroft, J. E., & Ullman, J. D. (1983). *Data structures and algorithms*. Addison-Wesley. [clásica]
- Akhilesh Kumar Srivastava. (2020). *A Practical Approach to Data Structure and Algorithm with Programming in C*. Arcler Press. [eBook]
- Fritelli, V. (2020). *Algoritmos y estructuras de datos* (2ª ed.). Jorge Sarmiento Editor – Universitas.
- Karumanchi, N. (2018). *Data Structures and Algorithms Made Easy: Data Structures and Algorithmic Puzzles* (6ª ed.). CareerMonk Publications. [clásica]
- Knuth, D. E. (1997–2011). *The art of computer programming* (Vols. 1–4). Addison-Wesley.[clásica]
- Skiena, S. S. (2020). *The algorithm design manual* (3rd ed.). Springer.
- R. Vaidyanathan. (2023). *Data Structure and Algorithm*. Laxmi Publications Pvt Ltd. [eBook]

XII. PERFIL DOCENTE

Licenciatura en Estructura de Datos y Algoritmos , Ciencias Computacionales o áreas afines, con grado mínimo de maestría y experiencia en programación avanzada, diseño de algoritmos y desarrollo de software. Debe dominar lenguajes de programación de alto nivel (como C, C++, Java o Python) y contar con conocimientos sólidos en análisis de complejidad, estructuras de datos fundamentales (listas, pilas, colas, árboles, grafos, tablas hash) y técnicas de diseño algorítmico. Debe poseer competencias didácticas para facilitar el aprendizaje basado en problemas, promover el pensamiento lógico y analítico, y evaluar procesos y productos de programación. Finalmente, debe mostrar responsabilidad, ética profesional, comunicación efectiva y una actitud de acompañamiento y apertura al aprendizaje continuo para guiar al estudiante en el desarrollo de habilidades computacionales avanzadas.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Programación Orientada a Objetos
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 01 Horas Laboratorio: 02 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Básica
Carácter:	Obligatoria
Requisito:	Introducción a la Programación
Modalidad de impartición:	Presencial
Equipo de diseño:	Alma Rocío Cabazos Marín Raúl Casillas Figueroa
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vazquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

El paradigma de la programación orientada a objetos representa una evolución frente a la programación estructurada, al proponer un enfoque más natural y modular para el diseño, desarrollo y mantenimiento del software. Este modelo facilita la portabilidad, la reutilización y el mantenimiento del código, contribuyendo a reducir los ciclos de desarrollo y a eliminar prácticas de codificación poco intuitivas.

El propósito general de la asignatura es que el estudiantado sea capaz de diseñar e implementar aplicaciones orientadas a objetos, trasladando entidades del mundo real a abstracciones de código, estableciendo entre ellas relaciones como asociaciones, dependencias y generalizaciones. Asimismo, se busca que el estudiantado adquiera la capacidad de evaluar y validar dichas aplicaciones mediante el uso de un lenguaje de programación orientado a objetos.

La asignatura se ubica en la etapa básica, tiene carácter obligatorio para la Licenciatura en Ciencias Computacionales y optativo para la Licenciatura en Matemáticas Aplicadas.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Desarrollar aplicaciones computacionales por medio del paradigma de programación orientada a objetos con criterio analítico, rigor metodológico y buenas prácticas de desarrollo de software, para automatizar procesos y modelar problemas reales o hipotéticos, de forma creativa y objetiva.

IV. EVIDENCIA DE APRENDIZAJE

Desarrollo de un proyecto final en un lenguaje de programación orientado a objetos, acompañadas de la documentación formal que describa el diseño, la implementación y la justificación de las soluciones propuestas.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia y vanguardia.

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos y evolución de la Programación Orientada a Objetos.

Competencia

Comparar el paradigma orientado a objetos con otras técnicas de programación, por medio del análisis de sus características y sus aplicaciones típicas para identificar las ventajas de este nuevo modelo de programación con una actitud crítica, propositiva y trabajo colaborativo.

Contenido

- 1.1. Historia y evolución de la programación orientada a objetos
- 1.2. Principios fundamentales de la orientación a objetos
- 1.3. Lenguajes y tecnologías orientadas a objetos
- 1.4. Análisis de problemas y modelado básico de soluciones

Duración

2 horas

Unidad II. Conceptos orientados a objetos.

Competencia

Analizar la importancia de la Programación Orientada a Objetos, a través de la definición de sus principales elementos y características que le permitan identificar las ventajas de este paradigma de programación con una actitud crítica, propositiva y trabajo colaborativo.

Contenido

- 2.1. Definición y creación de clases
- 2.2. Objetos: Instanciación y ciclo de vida
- 2.3. Mensajes
- 2.4 Constructores y destructores
- 2.5. Encapsulamiento: atributos y métodos
- 2.6. Modificadores de acceso

Duración

4 horas

Unidad III. Modelado Orientado a Objetos.

Competencia

Modelar visualmente soluciones a problemas hipotéticos, con la aplicación de los elementos básicos del Lenguaje de Modelado Unificado (UML) que le permitan generar escenarios gráficos de sus soluciones con una actitud analítica y creativa.

Contenido

- 3.1. Elementos básicos de Lenguaje Unificado de Modelado (UML)
- 3.2. Diseño Estático: diagrama de casos de uso y diagramas de clases
- 3.3. Principios SOLID
- 3.4. Introducción a patrones de diseño (Singleton, Factory, Observer, Strategy)

Duración

6 horas

Unidad IV. Pilares de la Programación Orientada a Objetos.

Competencia

Diseñar soluciones a problemas reales o hipotéticos a través de la aplicación de los principios de la programación orientada a objetos —abstracción, encapsulamiento, herencia y polimorfismo— para explorar y modelar programas con este paradigma de manera analítica y creativa.

Contenido

- 4.1. Abstracción
- 4.2. Encapsulamiento
- 4.3. Herencia
 - 4.3.1. Concepto y tipos de herencia
 - 4.3.2. Sobrecarga y sobrescritura de métodos
- 4.4. Polimorfismo
 - 4.4.1. Concepto de polimorfismo
 - 4.4.2. Aplicación del polimorfismo
 - 4.4.3. Interfaces y clases abstractas
- 4.5. Relaciones entre clases
 - 4.5.1. Asociación
 - 4.5.2. Agregación
 - 4.5.3. Composición
 - 4.5.4. Dependencia

Duración

10 horas

Unidad V. Desarrollo de Aplicaciones Orientada o Objetos

Competencia

Aplicar los conceptos avanzados de la programación orientada a objetos, mediante la implementación de un sistema de cómputo que integre sus elementos y técnicas para la solución de un problema de la vida real con una actitud propositiva y creativa.

Contenido

- 5.1. Colecciones de objetos
- 5.2. Mecanismos de captura y propagación (flujos)
- 5.3. Excepciones
 - 5.3.1 Concepto de error y excepción
 - 5.3.2 Jerarquía de excepciones

Duración

10 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Fundamentos y evolución de la Programación Orientada a Objetos

Práctica 1: Entendiendo el paradigma Orientado a Objetos

Duración

02 horas

Procedimiento

1. Atiende las orientaciones del docente.
2. Analiza las técnicas de Programación Orientada a Objetos con base en las aplicaciones típicas en las que es utilizada esta técnica.
3. Identifica las ventajas y desventajas del paradigma.
4. Conoce las aplicaciones típicas en las que es utilizada esta técnica.
5. Discute en clase los hallazgos.

Recursos de apoyo

- Computadora.
- Internet.
- Material bibliográfico.

Unidad II. Conceptos orientados a objetos.

Práctica 2: Implementación básica orientada a objetos

Duración

04 horas

Procedimiento

1. Atiende las orientaciones del docente.
2. Analiza los elementos básicos de programación del paradigma Orientado a Objetos.
3. Realiza una serie de ejercicios donde desarrolla pequeñas aplicaciones, mediante algún lenguaje de desarrollo de vanguardia.
4. Desarrolla una solución a un problema real mediante el análisis y la argumentación, para identificar los componentes necesarios en la solución.
5. Modela mediante un lenguaje gráfico, para visualizar dicha solución, y exponer los modelos ante el grupo en la clase.
6. Presenta un reporte de los ejercicios realizados.

Recursos de apoyo

- Computadora.
- Internet.
- Material bibliográfico.
- Lenguaje de desarrollo de vanguardia.

- Acceso a compilador que soporte el paradigma Orientado a Objetos.
- Acceso al ambiente de desarrollo(IDE) de preferencia.

Unidad III. Modelado Orientado a Objetos.

Práctica 3: Conceptos fundamentales de Modelado Orientada a Objetos

Duración

06 horas

Procedimiento

1. Atiende las orientaciones del docente.
2. Analiza los elementos de programación del paradigma relacionados con objetos compuestos y con herencia simple.
3. Realiza una serie de ejercicios donde desarrolla pequeñas aplicaciones, mediante algún lenguaje de desarrollo de vanguardia.
4. Desarrolla una solución a un problema real de una organización mediante la implementación de los componentes necesarios en la solución.
5. Modela mediante un lenguaje gráfico la solución, y expone los modelos en la clase para su discusión.
6. Presenta un reporte de los ejercicios realizados.

Recursos de apoyo

- Computadora.
- Internet.
- Material bibliográfico.
- Lenguaje de desarrollo de vanguardia.
- Acceso a compilador que soporte el paradigma Orientado a objetos.

Unidad IV: Pilares de la Programación Orientada a Objetos.

Práctica 4: Conceptos avanzados de la Programación Orientada a Objetos

Duración

10 horas

Procedimiento

1. Atiende las orientaciones del docente.
2. Analiza los elementos de programación del paradigma Orientado a Objetos relacionados con herencia múltiple y polimorfismo, así como colecciones de objetos mediante memoria dinámica.
3. Realiza una serie de ejercicios donde desarrolle pequeñas aplicaciones, mediante algún lenguaje de desarrollo de vanguardia.
4. Desarrolla una solución a un problema real destacando los componentes necesarios en la solución.

5. Genera el modelo para visualizar dicha solución, y exponer los modelos ante la clase para su discusión.
6. Presente un reporte de los ejercicios realizados.

Recursos de apoyo

- Computadora.
- Internet.
- Material bibliográfico.
- Lenguaje de desarrollo de vanguardia.
- Acceso a compilador que soporte el paradigma Orientado a objetos.

Unidad V. Desarrollo de Aplicaciones Orientada o Objetos

Práctica 5: Aplicación de usuario final

Duración

10 horas

Procedimiento

1. Atiende las orientaciones del docente.
2. Selecciona un entorno de desarrollo de su preferencia.
3. De forma individual o grupal se define un proyecto, incluyendo elementos avanzados del paradigma.
4. Genera la propuesta de proyecto.
5. Implementa el proyecto.
6. Documenta resultados del proyecto.
7. Presenta la funcionalidad del proyecto y documentación generada ante la clase y sube al portafolio el material.

Recursos de apoyo

- Computadora.
- Internet.
- Material bibliográfico.
- Lenguaje de desarrollo de vanguardia.
- Acceso a compilador que soporte el paradigma Orientado a objetos.

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Técnica expositiva
- Discusión guiada
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Mapas mentales
- Mapas conceptuales
- Cuadros comparativos
- Infografías
- Método de proyectos

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	25%
– Prácticas laboratorio	25%
– Tareas	10%
– Proyecto final	40%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Blasco, F. (2019). Programación orientada a objetos en Java: (1 ed.). RA-MA Editorial. [Clásica]

Castro, C. C., de León Amador, P. J. P., & Boró, E. S. (2006). Introducción a la programación orientada a objetos. Publicaciones de la Universidad de Alicante. [Clásica]

Guardati, S. (2007). Estructura de datos orientada a objetos. Algoritmos con C++. Pearson. [Clásica]

Jiménez de Parga, C. (2021). UML: arquitectura de aplicaciones en Java, C++ y Python. Ra-Ma.

Recursos de acceso libre o complementarios externos

Cervantes Ojeda, J., Gómez Fuentes, M. C., González Pérez, P. P. y García Nájera, a. (2016). Introducción a la programación orientada a objetos. UAM, Unidad Cuajimalpa.
http://www.cua.uam.mx/pdfs/revistas_electronicas/libros-electronicos/2016/2intro-poo/programacion_web.pdf [Clásica]

XII. PERFIL DOCENTE

Licenciatura en Ciencias Computacionales o área afín; preferentemente con estudios de posgrado, con experiencia en lógica programática, análisis, diseño e implementación de programas de computadora y conocimiento de lenguajes de programación basados en el paradigma orientado objetos. Ser una persona creativa, que emplee y fomente el trabajo colaborativo, desarrolle el pensamiento crítico en sus estudiantes y promueva la ética profesional.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Física
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Cálculo Vectorial
Clave:	
Distribución horaria:	Horas Clase: 03 Horas Taller: 02 Horas Laboratorio: 00 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 03 Créditos:08
Etapas de formación:	Básica
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Roberto Romo Martínez Elmer Cruz Mendoza Duilio Valdespino Padilla
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La finalidad de esta unidad de aprendizaje es proporcionar a las y los estudiantes una formación esencial en la construcción y aplicación rigurosa de los conceptos y métodos matemáticos que permiten analizar funciones de varias variables y campos vectoriales, herramientas indispensables en el estudio posterior de sistemas físicos reales. A través de las actividades, el estudiantado desarrolla habilidades para calcular derivadas parciales, gradientes y derivadas direccionales, aplicar el teorema de Taylor en varias variables, determinar máximos, mínimos y puntos de silla, así como resolver integrales dobles y triples en distintas coordenadas, fomentando una actitud analítica, ordenada y reflexiva. Se imparte en la etapa básica con carácter obligatorio, de naturaleza teórico-práctica, dentro del área de Matemáticas aplicadas a la física, y se recomienda tener conocimientos de cálculo diferencial e integral en una variable, sirviendo de base para cursos posteriores de física matemática y electromagnetismo.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Establecer los conceptos de límite, derivada e integral en funciones de varias variables, mediante la aplicación de los fundamentos del cálculo diferencial e integral de una variable para resolver e interpretar problemas físicos y geométricos, con formalidad, rigor conceptual y actitud reflexiva.

IV. EVIDENCIA DE APRENDIZAJE

Exposición integradora final:

Exposición colaborativa en equipo donde las y los estudiantes presentan, analizan e interpretan los conceptos y métodos fundamentales del cálculo vectorial trabajados a lo largo del curso. Esta actividad integra los resultados teóricos y prácticos de las prácticas realizadas, demuestra rigor conceptual y claridad en la aplicación de las técnicas de cálculo con funciones de varias variables, utiliza software de graficación para ilustrar los resultados y su interpretación geométrica y física, y muestra una participación proactiva, crítica y reflexiva en el trabajo en equipo.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Explorar fenómenos físicos mediante sus modelos matemáticos, para describir y predecir el comportamiento de la materia, la energía y sus interacciones, con disciplina y rigor científico.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia y vanguardia

VII. DESARROLLO POR UNIDADES

Unidad I. Diferenciación en funciones de varias variables

Competencia

Analizar los conceptos de diferencial y derivada en funciones reales de variable vectorial, a través del concepto de límite, para resolver problemas que involucren razones de cambio y aproximaciones lineales en funciones de varias variables, con formalidad y actitud reflexiva.

Contenido

- 1.1. Gráficas de funciones con valores reales
- 1.2. Límites y continuidad
- 1.3. Diferenciación
- 1.4. Trayectorias y velocidad
- 1.5. Propiedades de la derivada
- 1.6. Gradientes y derivadas direccionales.

Duración

18 horas

Unidad II. Teorema de Taylor, Máximos y mínimos

Competencia

Calcular máximos, mínimos y puntos silla de funciones de dos variables, por medio del criterio del Hessiano y el método de Multiplicadores de Lagrange, para resolver problemas en las ciencias aplicadas, con objetividad y de manera ordenada.

Contenido

- 2.1. Derivadas parciales iteradas
- 2.2. Teorema de Taylor
- 2.3. Extremos de funciones con valores reales
- 2.4. Extremos restringidos y multiplicadores de Lagrange
- 2.5. Aplicaciones.

Duración

06 horas

Unidad III. Funciones con valores vectoriales

Competencia

Analizar el comportamiento de funciones vectoriales, a través de los conceptos de divergencia y rotacional, para resolver problemas de aplicaciones en distintas áreas de la física, con responsabilidad y actitud reflexiva.

Contenido

- 3.1 Longitud de arco
- 3.2. Campos vectoriales
- 3.3. Divergencia
- 3.4. Rotacional
- 3.5. Identidades vectoriales

Duración

09 horas

Unidad IV. Integrales dobles y triples

Competencia

Analizar el concepto de integral en funciones de dos y tres variables, por medio del uso de los fundamentos del cálculo integral de una variable, para resolver problemas de aplicaciones geométricas y físicas, con formalidad y actitud reflexiva.

Contenido

- 4.1. Integral doble sobre un rectángulo
- 4.2. Integral doble sobre regiones más generales
- 4.3. Cambio de orden de integración
- 4.4. Interpretación de la integral doble como un volumen
- 4.5. Aplicaciones geométricas y físicas de las integrales dobles
- 4.6. Integrales triples
- 4.7. Cambio de variables en integrales dobles y triples
- 4.8. Integrales impropias

Duración

15 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

Unidad I. Diferenciación en funciones de varias variables

Práctica 1: Gráficas de funciones de varias variables

Duración:

02 horas

Procedimiento:

1. Atiende las orientaciones docentes para el desarrollo de la práctica
2. Aplica los métodos de curvas de nivel y secciones para graficar funciones de dos variables
3. Aplica los métodos de conjuntos y superficies de nivel y secciones para graficar funciones de variable múltiple
4. Grafica funciones sencillas con software (plano, esfera, paraboloides, conos, sillas de montar, cilindros)
5. Identifica simetrías y cortes con planos coordenados
6. Compara los bosquejos de tus gráficas elaboradas a mano con las generadas por el software utilizado
7. Elabora y entrega el reporte de la práctica en formato digital como evidencia de su realización

Recursos de apoyo:

- Pizarrón
- Marcadores
- Cuaderno de trabajo
- Software de cálculo y graficación (Python, Gnuplot, Geogebra, Matlab, o software en línea)
- Bibliografía de funciones reales de varias variables

Práctica 2: Límites en dos variables

Duración:

02 horas

Procedimiento:

1. Atiende las orientaciones docentes sobre la definición de límite en varias variables
2. Evalúa límites de funciones bidimensionales por distintas trayectorias
3. Aplica la definición de límite para demostrar la existencia de límites de funciones

4. Aplica la definición de continuidad para determinar la continuidad de funciones de dos variables en puntos específicos
5. Elabora y entrega el reporte de la práctica en formato digital como evidencia de su realización

Recursos de apoyo:

- Pizarrón
- Marcadores
- Cuaderno de trabajo
- Software de cálculo y graficación (Geogebra o software en línea)
- Bibliografía de funciones reales de varias variables

Práctica 3: Derivadas parciales y plano tangente

Duración:

02 horas

Procedimiento:

1. Atiende las indicaciones de su docente para realizar la práctica
2. Calcula derivadas parciales de funciones de dos variables
3. Determina el vector normal y plano tangente en un punto dado
4. Representa gráficamente la superficie y el plano tangente
5. Redacta una explicación de los cálculos y gráficas obtenidos
6. Elabora y entrega el reporte de la práctica en formato digital como evidencia de su realización

Recursos de apoyo:

- Pizarrón
- Marcadores
- Cuaderno de trabajo
- Software de cálculo y graficación (Geogebra o software en línea)
- Bibliografía de funciones reales de varias variables

Práctica 4: Trayectorias en dos y tres dimensiones

Duración:

02 horas

Procedimiento:

1. Atiende las indicaciones de su docente para realizar la práctica
2. Define trayectorias paramétricas en el espacio

3. Grafica curvas paramétricas en R^2 y R^3
4. Calcular vectores tangentes a la trayectoria
5. Define velocidad y aceleración en puntos de una trayectoria
6. Redacta una explicación de los cálculos y gráficas obtenidos
7. Elabora y entrega el reporte de la práctica en formato digital como evidencia de su realización

Recursos de apoyo:

- Pizarrón
- Marcadores
- Cuaderno de trabajo
- Software de cálculo y graficación (Geogebra o software en línea)
- Bibliografía de funciones vectoriales de variable real

Práctica 5: Propiedades de la derivada

Duración:

02 horas

Procedimiento:

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Calcula matrices de derivadas parciales
3. Resuelve ejemplos que aplican la regla de la cadena
4. Ver casos especiales de la regla de la cadena
5. Redacta una explicación de los ejemplos resueltos donde comentes los detalles de los diferentes casos particulares
6. Elabora y entrega el reporte de la práctica en formato digital como evidencia de su realización

Recursos de apoyo:

- Pizarrón
- Marcadores
- Cuaderno de trabajo
- Bibliografía de funciones de varias variables

Práctica 6: Gradientes y derivadas direccionales

Duración:

02 horas

Procedimiento:

1. Atiende las indicaciones de su docente para realizar la práctica
2. Calcula el gradiente de una función $f(x,y)$
3. Aplicar el gradiente al cálculo de la dirección de más rápido crecimiento de una función
4. Evalúa derivadas direccionales en distintas direcciones unitarias
5. Grafica el campo gradiente y analiza su interpretación física
6. Redacta una explicación de los cálculos y gráficas obtenidos así como de su interpretación
7. Elabora y entrega el reporte de la práctica en formato digital como evidencia de su realización

Recursos de apoyo:

- Pizarrón
- Marcadores
- Cuaderno de trabajo
- Software de cálculo y graficación (Geogebra o software en línea)
- Bibliografía de funciones de varias variables

Unidad II. Teorema de Taylor, Máximos y mínimos

Práctica 7: Máximos, mínimos y puntos silla

Duración:

02 horas

Procedimiento:

1. Atiende las indicaciones de su docente para realizar la práctica
2. Desarrolla el polinomio de Taylor de segundo orden para funciones de dos variables en un punto dado
3. Verifica aproximaciones gráficamente en torno a un punto crítico
4. Calcula el Hessiano para funciones en dos variables
5. Clasifica puntos críticos en máximos, mínimos o puntos de silla
6. Representa gráficamente cada tipo de punto crítico, discutiendo la interpretación geométrica

7. Elabora y entrega el reporte de la práctica en formato digital como evidencia de su realización

Recursos de apoyo:

- Pizarrón
- Marcadores
- Cuaderno de trabajo
- Software de cálculo y graficación (Geogebra o software en línea)
- Bibliografía sobre teorema de Taylor en funciones de varias variables

Práctica 8: Multiplicadores de Lagrange

Duración:

02 horas

Procedimiento:

1. Plantea problemas de optimización de funciones con una o más restricciones
2. Aplica el método de los multiplicadores de Lagrange paso a paso
3. Resuelve ejemplos representativos y discute la interpretación geométrica de la condición de Lagrange
4. Analiza aplicaciones físicas y geométricas simples
5. Discute las diferencias con problemas de máximos y mínimos sin restricciones, así como las ventajas y limitaciones del método
6. Elabora y entrega el reporte de la práctica en formato digital como evidencia de su realización

Recursos de apoyo:

- Pizarrón
- Marcadores
- Cuaderno de trabajo
- Software de cálculo y graficación (Geogebra o software en línea)
- Bibliografía sobre el método de multiplicadores de Lagrange

Unidad III. Funciones con valores vectoriales

Práctica 9: Trayectorias en $\mathbb{R}^3$, longitud de arco y aplicaciones físicas

Duración:

02 horas

Procedimiento:

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Calcula la longitud de arco en un intervalo y discute su interpretación geométrica.
3. Obtén la velocidad y aceleración a partir de la trayectoria y analiza su relación con la segunda ley de Newton en un ejemplo simple.
4. Representa la trayectoria y las magnitudes calculadas usando software 3D
5. Elabora y entrega el reporte de la práctica en formato digital como evidencia de su realización

Recursos de apoyo:

- Pizarrón
- Marcadores
- Cuaderno de trabajo
- Software de graficación 3D (Matlab, Python)

Práctica 10: Divergencia, rotacional e interpretación física de campos vectoriales

Duración:

02 horas

Procedimiento:

1. Atiende las orientaciones docentes para el desarrollo de la práctica
2. Evalúa la divergencia y el rotacional de distintos campos vectoriales en R^3 .
3. Interpreta físicamente los resultados en términos de flujos que se expanden, se contraen o se conservan, y de campos que generan circulación o rotación.
4. Representa ejemplos de campos vectoriales con software de graficación y discute su interpretación geométrica.
5. Analiza casos representativos de la física (ejemplo: flujo de un fluido incompresible, campo eléctrico de una carga puntual, campo magnético en torno a un conductor)
6. Redacta una explicación de tu interpretación física y geométrica de la divergencia y el rotacional de los campos vectoriales considerados
7. Elabora y entrega el reporte de la práctica en formato digital como evidencia de su realización

Recursos de apoyo:

- Pizarrón
- Marcadores
- Cuaderno de trabajo
- Software de graficación 3D (Matlab, Python)

Práctica 11: Identidades vectoriales y aplicaciones en física

Duración:

02 horas

Procedimiento:

1. Atiende las orientaciones docentes para el desarrollo de la práctica
2. Verifica algebraicamente identidades vectoriales básicas
3. Aplica al menos una de estas identidades en un problema físico
4. Discute colectivamente el significado físico de las identidades trabajadas
5. Elabora y entrega el reporte de la práctica en formato digital como evidencia de su realización

Recursos de apoyo:

- Pizarrón
- Marcadores
- Cuaderno de trabajo
- Bibliografía de funciones de varias variables

Unidad IV. Integrales dobles y triples

Práctica 12: Integral doble sobre un rectángulo y regiones generales

Duración:

02 horas

Procedimiento:

1. Atiende la explicación docente sobre integrales dobles en rectángulos y regiones generales
2. Plantea integrales dobles de funciones polinómicas sobre rectángulos
3. Extiende los cálculos a regiones delimitadas por curvas sencillas
4. Representa gráficamente las regiones de integración
5. Redacta una explicación de la elección de los límites de integración en regiones tipo-1, tipo-2 y tipo-3.
6. Elabora y entrega el reporte con cálculos y gráficas de la práctica en formato digital como evidencia de su realización

Recursos de apoyo:

- Pizarrón
- Marcadores

- Cuaderno de trabajo
- Software de graficación 3D (Matlab, Python)

Práctica 13: Cambio de orden de integración

Duración:

02 horas

Procedimiento:

1. Atiende la explicación de su docente sobre las reglas que rigen el cambio de orden de integración en integrales dobles
2. Resuelve una integral doble planteada con un orden de integración específico
3. Reescribe la región de integración para invertir el orden de integración
4. Verifica que ambos resultados coinciden
5. Discute las ventajas de un orden de integración sobre otro en casos concretos
6. Elabora y entrega el reporte de la práctica en formato digital como evidencia de su realización

Recursos de apoyo:

- Pizarrón
- Marcadores
- Cuaderno de trabajo
- Software de graficación de regiones 2D (Matlab, Python)

Práctica 14: Interpretación geométrica y física de la integral doble

Duración:

02 horas

Procedimiento:

1. Atiende las indicaciones de su docente para realizar la práctica
2. Resuelve integrales dobles interpretándolas como volúmenes bajo superficies
3. Grafica superficies $z=f(x,y)$ y las regiones de integración
4. Calcula volúmenes de cuerpos sencillos (cilindros, paraboloides truncados)
5. Interpreta los resultados en términos geométricos y físicos y redacta una breve explicación
6. Elabora y entrega el reporte de la práctica en formato digital como evidencia de su realización

Recursos de apoyo:

- Pizarrón
- Marcadores
- Cuaderno de trabajo
- Software de graficación en 3D (Geogebra, Matlab)

Práctica 15: Integrales triples y cambio de variables**Duración:**

02 horas

Procedimiento:

1. Atiende la explicación docente sobre el teorema que permite relacionar una integral triple con integrales unidimensionales iteradas
2. Atiende la explicación docente sobre el teorema de cambio de variables en integrales triples
3. Resuelve integrales triples en coordenadas cartesianas para calcular volúmenes de sólidos
4. Reescribe el problema en coordenadas cilíndricas o esféricas
5. Compara los resultados obtenidos en los diferentes sistemas de coordenadas
6. Discute la conveniencia de elegir un sistema de coordenadas adecuado
7. Elabora y entrega el reporte de la práctica en formato digital como evidencia de su realización

Recursos de apoyo:

- Pizarrón
- Marcadores
- Cuaderno de trabajo
- Software de cálculo simbólico (Mathematica, Maple, Python)

Práctica 16: Aplicaciones geométricas y físicas de integrales múltiples**Duración:**

02 horas

Procedimiento:

1. Atiende las indicaciones de su docente para realizar la práctica
2. Plantea integrales dobles y triples para calcular masa de láminas o sólidos con densidad variable
3. Calcula el centro de masa de una región plana con densidad dada

4. Resuelve un ejemplo de momento de inercia en un sólido sencillo
5. Interpreta los resultados en términos de aplicaciones físicas
6. Elabora y entrega el reporte de la práctica en formato digital como evidencia de su realización

Recursos de apoyo:

- Pizarrón
- Marcadores
- Cuaderno de trabajo
- Software de cálculo simbólico (Mathematica, Maple, Python)
- Bibliografía de física aplicada

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Aprendizaje basado en preguntas
- Técnica expositiva
- Discusión guiada
- Instrucción guiada

Estrategia de aprendizaje

- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	40%
– Tareas	20%
– Prácticas taller	20%
– Exposición integradora final	20%
– Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Hughes-Hallet, D. (1992). *Calculus* (3rd edition). Wiley. ISBN: 0-471-408-27-1.

Marsden, J. E., & Tromba, A. (2018). *Cálculo vectorial* (6ta ed.). Pearson Educación. ISBN: 978-84-2056-866-9 <https://uabc.vitalsource.com/reader/books/9788420568669>

Stewart, J., Clegg, D., Watson, S. (2021). *Cálculo de varias variables (7ma ed.). Trascendentes tempranas*. Cengage. ISBN: 978-0-357-11350-9. <https://libcon.rec.uabc.mx:2383/?il=21475>

Thomas Jr., G. B., Hass, J., Heil, C., & Przemyslaw, B. (2024). *Cálculo: Varias variables* (14.^a ed.) [ePub]. Pearson Educación. ISBN: 978-607-32-6032-9. <https://uabc.vitalsource.com/reader/books/9786073260336>

Tom M. Apostol (2002). *Calculus* (2da ed.). Reverté Ediciones, S.A de C.V. ISBN: 84-291-5003-X

Williamson R. E., Cromwell R. H. y Trotter Hale F (1972). *Calculus of Vector Functions* (Third edition). Prentice Hall Inc. ISBN: 0-13-112367-X.

Recursos de acceso libre o complementarios externos

Khan Academy. (s.f.). *Cálculo multivariable*. Khan Academy. <https://es.khanacademy.org/math/multivariable-calculus>

XII. PERFIL DOCENTE

Licenciatura en Física, Matemáticas o área afín, con conocimientos sólidos en cálculo diferencial e integral de varias variables, gradiente, divergencia, rotacional, así como en integrales dobles y triples y su aplicación a problemas geométricos y físicos. Se sugiere contar con estudios de posgrado y con una experiencia docente mínima de dos años. Deberá mostrar

una actitud analítica, responsable, proactiva y respetuosa, contar con habilidades de comunicación clara y fomentar la participación individual y colaborativa del estudiantado.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Física Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Probabilidad
Clave:	
Distribución horaria:	Horas Clase: 03 Horas Taller: 02 Horas Laboratorio: 00 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 03 Créditos:08
Etapas de formación:	Básica
Carácter:	Obligatoria
Requisito:	Ninguno.
Modalidad de impartición:	Presencial
Equipo de diseño:	Juan Crisóstomo Tapia Mercado Adina Jordán Arámburo
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

El propósito de esta unidad de aprendizaje es que los y las estudiantes adquieran los fundamentos teóricos y metodológicos de la probabilidad necesarios para la formulación, modelación y análisis de fenómenos aleatorios en los ámbitos de la física, las matemáticas aplicadas y las ciencias computacionales, lo anterior mediante el estudio del análisis combinatorio, los espacios de probabilidad, la esperanza matemática y las distribuciones y densidades especiales, los y las estudiantes desarrollarán competencias para identificar, representar y resolver problemas relacionados con procesos aleatorios, así como para interpretar y aplicar los resultados en contextos científicos y tecnológicos. Perteneciente al plan de estudios de Física, se ubica en etapa básica con carácter obligatorio, en el área de conocimiento de matemáticas; además se comparte con la Licenciatura en Ciencias Computacionales en etapa básica con carácter obligatorio.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Analizar los fundamentos teóricos y metodológicos de la probabilidad mediante el análisis de los modelos matemáticos de poblaciones, para modelar, interpretar y resolver problemas asociados a fenómenos aleatorios en diferentes campos, con una actitud crítica y responsable, fomentando el trabajo autónomo y colaborativo.

IV. EVIDENCIA DE APRENDIZAJE

Exposición del proyecto final en la que presenten una resolución de un problema o situación real, aplicando de manera sistemática los métodos analíticos y/o numéricos adecuados para obtener una solución precisa y sustentada. Se deberá explicar claramente cada etapa del proceso, desde la identificación del problema, el planteamiento de hipótesis, la elección del método, el análisis de resultados y la validación de la solución obtenida. Asimismo, se valorará su participación proactiva, colaborativa y responsable dentro del equipo de trabajo, evidenciando habilidades de comunicación, pensamiento crítico y aplicación práctica de los conocimientos adquiridos.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Licenciatura en Física

Explorar fenómenos físicos mediante sus modelos matemáticos, para describir y predecir el comportamiento de la materia, la energía y sus interacciones, con disciplina y rigor científico.

Licenciatura en Ciencias Computacionales

Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Inclusión; Excelencia y Equidad social y de Género.

VII. DESARROLLO POR UNIDADES

Unidad I. Análisis combinatorio

Competencia

Analizar los principios del análisis combinatorio mediante la resolución de problemas de conteo y la aplicación de estrategias sistemáticas para organizar y cuantificar diferentes arreglos y selecciones posibles, con el fin de fundamentar el cálculo de probabilidades en diversos contextos, fomentando el trabajo colaborativo, el pensamiento crítico y la toma de decisiones fundamentadas.

Contenido

- 1.1. Principio básico de conteo aditivo
- 1.2. Principio básico de conteo multiplicativo: simple (PBCS) y generalizado (PBCG)
- 1.3. Permutaciones
 - 1.3.1. Permutaciones sin repetición
 - 1.3.1.1. Construcción a partir del PBCG
 - 1.3.1.1.1. Permutaciones: n de n elementos
 - 1.3.1.1.2. Permutaciones: r de n elementos
 - 1.3.2. Permutaciones con repetición
 - 1.3.3. Permutaciones circulares
- 1.4. Combinaciones
 - 1.4.1. Combinaciones sin repetición
 - 1.4.1.1. Construcción a partir de permutaciones
 - 1.4.2. Combinaciones con repetición

Duración

08 horas

Unidad II. Teoría de la probabilidad

Competencia

Utilizar los conceptos básicos de teoría de la probabilidad, mediante el uso de métodos y técnicas de esta área, para desarrollar modelos matemáticos de las poblaciones, con actitud crítica, trabajo en equipo y responsable.

Contenido

- 2.1 Concepto de azar y fenómenos aleatorios.

- 2.1.1. Diferencia entre fenómenos deterministas y estocásticos.
- 2.1.2. Ejemplos básicos: lanzamiento de dados, monedas, urnas.
- 2.1.3. Discusión sobre la incertidumbre en fenómenos reales.
- 2.2. Espacios de resultados y sucesos.
 - 2.2.1. Definiciones
 - 2.2.2. Espacio muestral
 - 2.2.3. Evento
 - 2.2.3.1. Evento elemental o simple
 - 2.2.3.2. Evento compuesto
 - 2.2.4. Operaciones con eventos: unión, intersección, complemento, diferencia.
 - 2.2.5. Álgebra de eventos
 - 2.2.6. Modelos de probabilidad
 - 2.2.6.1. Enfoques históricos
 - 2.2.6.1.1. Clásico
 - 2.2.6.1.2. Frecuentista
 - 2.2.6.1.3. Subjetivo
 - 2.2.6.1.4. Axiomático
 - 2.2.6.2. Enfoque axiomático de Kolmogórov
 - 2.2.6.2.1. Definición formal de probabilidad
 - 2.2.6.3.2. Axiomas de Kolmogórov
 - 2.2.6.3.2.1. No negatividad
 - 2.2.6.3.2.2. Normalización
 - 2.2.6.3.2.3. Aditividad numerable
 - 2.2.6.3. Demostraciones básicas
 - 2.2.6.3.1. Monotonía de la probabilidad
 - 2.2.6.3.2. Probabilidad del complemento
 - 2.2.6.3.3. Desigualdad de Boole
 - 2.2.6.3.4. Subaditividad y aditividad finita
- 2.3. Probabilidad condicional
 - 2.3.1. Definición
 - 2.3.2. Interpretación
 - 2.3.3. Ejemplos
- 2.4. Independencia de eventos
 - 2.4.1. Definición formal
 - 2.4.2. Interpretación
 - 2.4.3. Propiedades
 - 2.4.1.1. Simetría
 - 2.4.1.2. Complementos
 - 2.4.1.3. Múltiples eventos

- 2.4.4. Independencia de más de dos eventos
- 2.5. Teorema de la probabilidad total
 - 2.5.1. Demostración a partir de probabilidades condicionales.
- 2.6. Teorema de Bayes
 - 2.6.1. Definición formal
 - 2.6.2. Interpretación
 - 2.6.2. Aplicación práctica y ejemplos con eventos condicionales.

Duración

10 horas

Unidad III. Distribuciones y densidades de probabilidad

Competencia

Emplear los conceptos asociados a las funciones de distribución de probabilidad, mediante el desarrollo de ejercicios típicos para aplicarlos a problemas de la misma área de conocimiento, fomentando el pensamiento abstracto, con actitud crítica, de trabajo en equipo y responsable.

Contenido

- 3.1. Variables aleatorias discretas
- 3.2. Distribuciones de probabilidad
- 3.3. Variables aleatorias continuas
- 3.4. Funciones de densidad de probabilidad

Duración

08 horas

Unidad IV. Esperanza Matemática

Competencia

Diferenciar las propiedades de la media y la varianza, a través de la manipulación de los datos de una muestra, para describir y caracterizar la distribución de dicha muestra, con actitud crítica, propositiva y responsable

Contenido

- 4.1. Valor esperado de una variable aleatoria
- 4.2. Momentos
- 4.3. Teorema de Chebyshev
- 4.4. Función generadora de momentos

Duración

08 horas

Unidad V. Distribuciones discretas especiales

Competencia

Manipular las distribuciones de probabilidad de variable discreta, mediante la descripción y desarrollo de ejercicios típicos para aplicarlos a problemas del área de la estadística, con actitud crítica, de trabajo en equipo y responsable

Contenido

- 5.1. Distribución Discreta uniforme
- 5.2. Distribución Bernoulli
- 5.3. Distribución Binomial
- 5.4. Distribución Multinomial
- 5.5. Distribución Hipergeométrica
- 5.6. Distribución Geométrica
- 5.7. Distribución Binomial Negativa
- 5.8. Distribución Poisson

Duración

10 horas

Unidad VI. Distribuciones continuas especiales

Competencia

Manipular las distribuciones de probabilidad de variable continua, mediante la descripción y desarrollo de ejercicios típicos para aplicarlos a problemas del área de la estadística, con actitud crítica, de trabajo en equipo y responsable.

Contenido

- 6.1. Distribución Uniforme Continua
- 6.2. Distribución Gamma
- 6.3. Distribución Exponencial
- 6.4. Distribución *Chi*-Cuadrada
- 6.5. Distribución Normal
- 6.6. Distribución *t* Student
- 6.7. Distribución Beta
- 6.8. Distribución Weibull
- 6.9. Distribución *F*

Duración

04 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

Unidad I. Análisis combinatorio

Práctica 1: La regla de multiplicación

Duración

02 horas

Procedimiento

1. Atiende la introducción del docente sobre la regla de multiplicación.
2. Calcula todas las alternativas posibles en una situación dada o por lo menos determina cuántas posibilidades posibles existen utilizando la regla de multiplicación.
3. Resuelve problemas utilizando la regla de multiplicación o diagramas de árbol.
4. Utiliza los ejemplos proporcionados por el profesor como retroalimentación.
5. Consulta la bibliografía del curso.
6. Entrega el archivo o documento con los ejercicios resueltos como evidencia para revisión y retroalimentación docente.

Recursos de apoyo:

- Pizarrón.
- Cuaderno.
- Bibliografía de técnicas de conteo.
- Documentos PDF de la Plataforma Blackboard.
- Videos de problemas resueltos de la Plataforma Blackboard.

Práctica 2: Permutaciones

Duración

04 horas

Procedimiento

1. Atiende las orientaciones docentes para comprender las situaciones donde los productos o resultados son los órdenes o arreglos diferentes de un grupo de objetos.
2. Revisa la definición de permutación.
3. Aplica la definición de permutación en cálculos numéricos sencillos para conocer cuántos arreglos diferentes son posibles tomando todos los objetos a la vez.
4. Realizar cálculos para conocer el número de permutaciones de n objetos diferentes tomados r a la vez.
5. Utiliza los ejemplos proporcionados por el profesor como retroalimentación.
6. Consulta la bibliografía del curso.

7. Entrega un reporte con ejercicios seleccionados y recibe retroalimentación docente.

Recursos de apoyo:

- Pizarrón.
- Cuaderno.
- Bibliografía de permutaciones.
- Documentos PDF de la Plataforma Blackboard.
- Videos de problemas resueltos de la Plataforma Blackboard.

Práctica 3: Combinaciones

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes para comprender las situaciones donde nos interesa el número de maneras de seleccionar r objeto de entre n sin considerar el orden en el cual se seleccionan.
2. Revisa la diferencia entre combinación y permutación.
3. Aplica la definición de combinación en cálculos numéricos sencillos.
4. Utiliza los ejemplos proporcionados por el profesor como retroalimentación.
5. Consulta la bibliografía del curso.
6. Entrega un reporte con ejercicios seleccionados y recibe retroalimentación docente.

Recursos de apoyo:

- Pizarrón.
- Cuaderno.
- Bibliografía de combinaciones.
- Documentos PDF de la Plataforma Blackboard.
- Videos de problemas resueltos de la Plataforma Blackboard.

Práctica 4: Teoremas de coeficientes binomiales

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre la práctica a realizar.
2. Utiliza el concepto de combinaciones para relacionarlo con los coeficientes de un binomio.
3. Revisa los diferentes teoremas de coeficientes binomiales para simplificar el cálculo de coeficientes binomiales.
4. Utiliza los ejemplos proporcionados por el profesor como retroalimentación.

5. Consulta la bibliografía del curso.
6. Entrega un reporte con ejercicios seleccionados y recibe retroalimentación docente.

Recursos de apoyo:

- Pizarrón.
- Cuaderno.
- Bibliografía de análisis combinatorio.
- Documentos PDF de la Plataforma Blackboard.
- Videos de problemas resueltos de la Plataforma Blackboard.

Unidad II. Teoría de la probabilidad

Práctica 5: Probabilidad

Duración

06 horas

Procedimiento

1. Atiende las orientaciones docentes.
2. Realiza ejercicios que permitan identificar los conceptos básicos de teoría de la probabilidad.
3. Utiliza los métodos y técnicas de la teoría de probabilidad para desarrollar modelos matemáticos de las poblaciones.
4. Resuelve problemas utilizando los conceptos de eventos independientes.
5. Realiza ejercicios que permitan utilizar el teorema de Bayes.
6. Utiliza los ejemplos proporcionados por el profesor como retroalimentación.
7. Consulta la bibliografía del curso.
8. Entrega un reporte con ejercicios seleccionados y recibe retroalimentación docente.

Recursos de apoyo:

- Pizarrón.
- Cuaderno.
- Bibliografía de probabilidad.
- Documentos PDF de la Plataforma Blackboard.
- Videos de problemas resueltos de la Plataforma Blackboard.

Unidad III. Distribuciones y densidades de probabilidad

Práctica 6: Probabilidad

Duración

04 horas

Procedimiento

1. Atiende las orientaciones docentes.
2. Realiza ejercicios que permitan identificar los conceptos básicos de teoría de la probabilidad.
3. Utiliza los métodos y técnicas de la teoría de probabilidad para desarrollar modelos matemáticos de las poblaciones.
4. Resuelve problemas utilizando los conceptos de eventos independientes.
5. Realiza ejercicios que permitan utilizar el teorema de Bayes.
6. Utiliza los ejemplos proporcionados por el profesor como retroalimentación.
7. Consulta la bibliografía del curso.
8. Entrega un reporte con ejercicios seleccionados y recibe retroalimentación docente.

Recursos de apoyo:

- Pizarrón.
- Cuaderno.
- Bibliografía de probabilidad.
- Documentos PDF de la Plataforma Blackboard.
- Videos de problemas resueltos de la Plataforma Blackboard.

Unidad IV. Esperanza Matemática

Práctica 7: Parámetros más destacados en distribuciones de Probabilidad

Duración

04 horas

Procedimiento

1. Atiende las orientaciones docentes.
2. Realiza los ejercicios proporcionados por el docente para practicar la manipulación de las propiedades de la Esperanza matemática y en particular la media y la varianza.
3. Reconoce las propiedades de los momentos y de la función generadora, a través de ejercicios, que sirvan para describir y caracterizar la distribución.
4. Resuelve problemas utilizando el teorema de Chebyshev.
5. Utiliza los ejemplos proporcionados por el profesor como retroalimentación.
6. Consulta la bibliografía del curso.
7. Entrega un reporte con ejercicios seleccionados y recibe retroalimentación docente.

Recursos de apoyo:

- Pizarrón.
- Cuaderno.
- Bibliografía de probabilidad.
- Documentos PDF de la Plataforma Blackboard.

- Videos de problemas resueltos de la Plataforma Blackboard.

Unidad V. Distribuciones discretas especiales

Práctica 8: Ensayos repetidos e independientes

Duración

04 horas

Procedimiento

1. Atiende las orientaciones docentes.
2. Utiliza las características particulares de las funciones de distribución para distinguir las diferentes distribuciones discretas de probabilidad, distribución binomial, binomial negativa, geométrica, hipergeométrica y Poisson.
3. Realiza ejercicios que permitan discutir el comportamiento y significado de las funciones de probabilidad, así como sus propiedades.
4. Resuelve problemas utilizando las diferentes aproximaciones y parámetros para simplificar el cálculo de las probabilidades de un suceso.
5. Utiliza los ejemplos proporcionados por el profesor como retroalimentación.
6. Consulta la bibliografía del curso.
7. Entrega un reporte con ejercicios seleccionados y recibe retroalimentación docente.

Recursos de apoyo:

- Pizarrón.
- Cuaderno.
- Bibliografía de probabilidad.
- Documentos PDF de la Plataforma Blackboard.
- Videos de problemas resueltos de la Plataforma Blackboard.

Unidad VI. Distribuciones continuas especiales

Práctica 9: La distribución normal

Duración

04 horas

Procedimiento

1. Atiende las orientaciones docentes.
2. Documenta las diferentes tablas de la distribución Normal estándar para calcular la probabilidad de un evento.
3. Realiza ejercicios que permitan caracterizar distribuciones de probabilidad de variable continua, poniendo énfasis en la distribución Normal.

4. Resuelve problemas de variables aleatorias discretas utilizando aproximación a la distribución normal estándar.
5. Utiliza los ejemplos proporcionados por el profesor como retroalimentación.
6. Consulta la bibliografía del curso.
7. Entrega un reporte con ejercicios seleccionados y recibe retroalimentación docente.

Recursos de apoyo:

- Pizarrón.
- Cuaderno.
- Bibliografía de técnicas de conteo.
- Documentos PDF de la Plataforma Blackboard.
- Videos de problemas resueltos de la Plataforma Blackboard.

MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer de forma clara y formal:

- La metodología de trabajo.
- Criterios de evaluación y rúbricas.
- Reglas sobre la calidad y entrega de trabajos académicos.
- Derechos y obligaciones del o la docente y del estudiantado.

Esto debe quedar registrado y comunicado por medios institucionales (como Blackboard o correo institucional).

Estrategia de enseñanza

- Estudio de caso
- Método de proyectos
- Técnica expositiva
- Discusión guiada

Estrategia de aprendizaje

- Investigación documental
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

- Evaluaciones parciales	20%
- Prácticas de taller	40%
- Tareas	20%
- Exposición de la resolución de un caso	20%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Devore, J. L. (2019). *Introducción a la probabilidad y estadística para ingeniería y ciencias*. Cengage Learning. [clásica]
- Freund, J. E., Miller, I., & Miller, M. (2000). *Estadística matemática con aplicaciones*. Prentice Hall. [clásica]
- Freund, J. E., & Walpole, R. E. (1990). *Estadística matemática con aplicaciones*. Prentice Hall. [clásica]
- Mendenhall, W., Beaver, B. M., & Beaver, R. J. (2023). *Introducción a la probabilidad y estadística*. Cengage Learning.
- Walpole, R. E. (2012). *Probabilidad y estadística para ingeniería y ciencias* (9.^a ed.). Pearson Hispanoamérica. [clásica]
- Walpole, R. E., Myers, R. H., Myers, S. L., & Ye, K. (2022). *Probabilidad y estadística para ingeniería y ciencias*. Pearson.

Recursos de acceso libre o complementarios externos

- Pozo, F., Parés, N., Vidal, Y., & Agis, D. (2025). *Probabilidad y estadística matemática*. García Maroto Editores.
- Sanchez, E., Cacerez, I. (2021). *Probabilidad y estadística*. Patria.

Tyurin, A. V., & Akhmerov, A. Yu. (2020). *Teoría de la probabilidad y estadísticas matemáticas*. Scienca Scripts.

Velasco, G. (2015). *Probabilidad: Fundamentos y aplicaciones*. Trillas. [clásica]

Warpenter, A. (2025). *Esenciales de probabilidad y estadística*.

XII. PERFIL DOCENTE

La persona docente que imparta la unidad de aprendizaje de Probabilidad deberá contar con título de Licenciatura en Física, Matemáticas o áreas afines, así como con conocimientos sólidos en teoría de probabilidades y en el uso de procedimientos analíticos y numéricos para la resolución de problemas con variables aleatorias en diversos contextos de la vida real. Preferentemente, contar con estudios de posgrado y con al menos dos años de experiencia en docencia universitaria. Asimismo, se requiere que quien imparta la unidad sea una persona analítica, respetuosa, responsable y proactiva; que posea facilidad de expresión y que promueva la participación individual y colaborativa del estudiantado.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada.
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Matemáticas Discretas
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 03 Horas Laboratorio: 00 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Básica
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Eloísa del Carmen García Canseco José Ariel Camacho Gutiérrez Jonathan Verdugo Olachea
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

El curso de Matemáticas Discretas es fundamental para la formación en Ciencias Computacionales, pues provee el andamiaje matemático para el diseño, análisis y comprensión de los sistemas digitales. El propósito de esta unidad de aprendizaje es que el estudiantado no sólo domine los conceptos teóricos, sino que desarrolle la habilidad de aplicarlos para resolver problemas computacionales concretos.

Mediante un enfoque práctico y moderno, se utilizará el lenguaje de programación Python y herramientas interactivas como Google Colab para modelar, simular y analizar los conceptos. El curso conecta la teoría con áreas de vanguardia como la ciberseguridad, el análisis de redes y la lógica computacional, preparando al estudiante para los desafíos tecnológicos actuales.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Demostrar el dominio de los conceptos fundamentales de las matemáticas discretas, utilizando herramientas de programación para la implementación, el modelado y el análisis de algoritmos, para resolver problemas del ámbito de las ciencias computacionales, con un enfoque estructurado, lógico y creativo.

IV. EVIDENCIA DE APRENDIZAJE

Portafolio digital

Deberá contener ejercicios teóricos y prácticos, organizados por unidades temáticas, será realizado de manera individual en el que se presente: una selección curada de problemas teóricos resueltos a mano por el estudiante, demostrando el dominio de los procedimientos formales (ej. una simplificación algebraica, una tabla de verdad, un cálculo de combinatoria, etc.) así como su implementación práctica en una libreta de Google Colab para verificar el resultado o analizar el problema con datos más grandes.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia y Vanguardia.

VII. DESARROLLO POR UNIDADES

Unidad I. Aritmética computacional

Competencia

Analizar la representación de los datos numéricos, utilizando conversiones entre sistemas de numeración y operaciones a nivel de bits para la correcta selección y manipulación de tipos de datos en la programación, con pensamiento lógico y analítico, con una actitud de responsabilidad y ética.

Contenido

- 1.1. Las matemáticas discretas como lenguaje de las ciencias computacionales.
- 1.2. Aplicaciones prácticas de las matemáticas discretas.
- 1.3. Notación matemática y conceptos fundamentales.
- 1.4. Sistemas de numeración y representación de números enteros en diferentes bases.
- 1.5. Operaciones aritméticas en diferentes bases.
- 1.6. Aritmética computacional: adición, suma, multiplicación, división, complemento a 1, complemento a 2, representación de números negativos.
- 1.7. Números binarios en punto fijo y punto flotante.

Duración

06 horas

Unidad II. Álgebra booleana

Competencia

Encontrar funciones booleanas y circuitos lógicos combinacionales, utilizando métodos de simplificación algebraica y representación gráfica para la implementación eficiente de soluciones a problemas de lógica digital, con creatividad, compromiso y congruencia.

Contenido

- 2.1. Variables booleanas.
- 2.2. Funciones booleanas básicas.
- 2.3. Funciones booleanas compuestas.
- 2.4. Puertas lógicas básicas (AND, OR, NOT) y derivadas (NAND, NOR, XOR, XNOR); y su simbología estándar.
- 2.5. Representación tabular de funciones booleanas.
- 2.6. Teoremas y postulados del álgebra booleana.
- 2.7. Simplificación algebraica de funciones booleanas.
- 2.8. Dualidad y formas canónicas.
- 2.9. Analogía del álgebra booleana con el álgebra de conjuntos.
- 2.10. Mapas de Karnaugh.

- 2.11. Diseño de circuitos digitales.
- 2.12. Puertas universales (NAND y NOR).
- 2.11. Funciones booleanas y aritmética computacional.

Duración

06 horas

Unidad III. Numeración y conteo

Competencia

Examinar técnicas de conteo y probabilidad discreta, utilizando permutaciones, combinaciones y el principio de inclusión-exclusión, para cuantificar el número de resultados posibles de un sistema, con precisión, pensamiento abstracto con iniciativa, entusiasmo y compromiso.

Contenido

- 4.1 Principios fundamentales de conteo.
- 4.2 Principio de inclusión y exclusión.
- 4.3 Combinaciones y permutaciones, con y sin repetición.
- 4.4 Teorema del binomio y coeficientes binomiales.
- 4.5 Principio de Dirichlet (Principio del Palomar).
- 4.6 Relaciones de recurrencia.
- 4.7 Funciones generadoras.
- 4.8 Demostraciones combinatorias.

Duración

06 horas

Unidad IV. Teoría de grafos y árboles

Competencia

Establecer relaciones y sistemas jerárquicos, utilizando las definiciones y propiedades de grafos y árboles para clasificar sus características estructurales y determinar su aplicabilidad en la representación de problemas del mundo real, con abstracción, pensamiento lógico, sentido crítico y responsabilidad.

Contenido

- 5.1 Grafos y dígrafos.
 - 5.1.1 Definiciones básicas y terminología.
 - 5.1.2 Tipos fundamentales de grafos.
 - 5.1.3 Caminos, ciclos y conectividad en grafos y dígrafos.
 - 5.1.2 Recorrido y circuito euleriano.
 - 5.1.3 Grafos eulerianos y sus propiedades.

- 5.1.4 Ciclo hamiltoniano y el problema del agente de ventas viajero.
- 5.1.5 Representación y propiedades de los grafos.
- 5.1.5 Isomorfismos.
- 5.1.6 Grafos planos y la fórmula de Euler.
- 5.1.7 Matriz de Adyacencia.
- 5.1.8 Lista de Adyacencia
- 5.1.9 Coloración.
- 5.2 Árboles.
- 5.2.1 Definición, terminología y propiedades de los árboles.
- 5.2.2 Árboles con raíz.
- 5.2.2 Árboles de expansión.
- 5.2.3 Árboles binarios.
- 5.2.4 Aplicaciones conceptuales de los árboles.

Duración

08 horas

Unidad V. Fundamentos de criptografía aplicada

Competencia

Analizar el funcionamiento de los criptosistemas de clave pública, por medio de los fundamentos de la aritmética modular, para comprender cómo se establece la comunicación segura en entornos digitales, con una aproximación conceptual y metódica, promoviendo la ética, la responsabilidad y la conciencia en seguridad digital.

Contenido

- 6.1 El problema de la comunicación segura
- 6.2 Criptografía simétrica y asimétrica
- 6.3 Herramientas matemáticas esenciales para criptografía
- 6.4 Aritmética modular e inverso multiplicativo
- 6.5 Aplicaciones: algoritmo RSA, intercambio de claves Diffie-Hellman

Duración

06 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

Unidad I. Aritmética Computacional

Práctica 1: Explorando sistemas de numeración y operaciones a nivel de bit

Duración

04 horas

Procedimiento

1. Atiende instrucciones de su docente.
2. Resuelve manualmente ejercicios de conversión entre sistemas numéricos (decimal, binario, hexadecimal).
3. Aplica manualmente los operadores a nivel de bits (AND, OR, XOR) sobre números binarios.
4. Verifica los resultados de las conversiones en Google Colab utilizando las funciones bin(), hex() e int() de Python.
5. Comprueba los resultados de las operaciones a nivel de bits usando los operadores &, | y ^ en Python.
6. Documenta las comparaciones entre los resultados manuales y los computacionales en un reporte de práctica.
7. Discute de manera grupal los resultados y recibe retroalimentación docente.
8. Entrega reporte de práctica.

Recursos de apoyo

- Computadora con acceso a internet
- Google Colab
- Documentación oficial de Python sobre operadores bitwise.
- Tutorial de Google Colab para principiantes.

Práctica 2: Representación de números reales en binario (punto fijo y punto flotante)

04 horas

Procedimiento

1. Atiende instrucciones de su docente.
2. Convierte manualmente números decimales con parte fraccionaria a su representación en punto fijo binario.
3. Representa un número decimal en un formato simplificado de punto flotante (signo, exponente, mantisa).
4. Implementa un script en Python para automatizar la conversión de la parte fraccionaria de un número decimal a binario.
5. Analiza los errores de precisión de punto flotante en Python (p. ej., $0.1 + 0.2$).
6. Documenta el proceso de conversión y las conclusiones sobre los errores de precisión.
7. Discute de manera grupal los resultados y recibe retroalimentación docente.

8. Entrega reporte de resultados y conclusiones.

Recursos de apoyo

- Computadora con acceso a internet
- Google Colab
- Python 3
- Explicación del estándar de punto flotante IEEE 754
- Imagen con la estructura de un número en punto flotante IEEE 754

Unidad II. Álgebra Booleana

Práctica 3: Simplificación de funciones booleanas

Duración

04 horas

Procedimiento

1. Atiende instrucciones de su docente.
2. Simplifica las funciones booleanas utilizando los teoremas y postulados del Álgebra de Boole.
3. Diseña un circuito lógico simple (p. ej., decodificador) utilizando mapas de Karnaugh.
4. Define las funciones booleanas (original y simplificada) en Python con la librería SymPy.
5. Verifica la equivalencia lógica de las funciones generando sus tablas de verdad.
6. Documenta el proceso de diseño, de simplificación y los resultados de la verificación.
7. Discute de manera grupal los resultados y recibe retroalimentación docente.
8. Entrega las evidencias de los documentos y los resultados.

Recursos de apoyo

- Computadora con acceso a internet
- Google Colab
- Tutorial de la biblioteca SymPy para lógica booleana.
- Imagen de un mapa de Karnaugh de cuatro variables

Práctica 4: Diseño de circuitos digitales

Duración

04 horas

Procedimiento

1. Atiende instrucciones de su docente.
2. Obtiene las funciones booleanas para la suma y el acarreo de un circuito medio sumador.
3. Simplifica manualmente las funciones booleanas obtenidas.
4. Implementa las funciones de suma y acarreo en un script de Python.
5. Simula el comportamiento del medio sumador para todas las combinaciones de entrada posibles.

6. Documenta el diseño del circuito y los resultados de la simulación.
7. Discute de manera grupal los resultados y recibe retroalimentación docente.
8. Entrega el diseño y los resultados.

Recursos de apoyo

- Computadora con acceso a internet
- Google Colab
- Diagrama lógico de un circuito medio sumador

Unidad III: Numeración y conteo

Práctica 5: Problemas de conteo

Duración

06 horas

Procedimiento

1. Atiende instrucciones de su docente.
2. Resuelve manualmente problemas de conteo utilizando permutaciones y combinaciones.
3. Utiliza el módulo math de Python (perm, comb) para verificar los resultados.
4. Emplea Python para resolver problemas de conteo con números grandes.
5. Documenta la estrategia de resolución de cada problema y la comparación de resultados.
6. Discute de manera grupal los resultados y recibe retroalimentación docente.
7. Entrega la resolución de problemas y la comparación de resultados.

Recursos de apoyo

- Computadora con acceso a internet.
- Google Colab.
- Documentación del módulo “math” de Python.
- Documentación del módulo “itertools” de Python.

Práctica 6: Principio de inclusión-exclusión

Duración

04 horas

Procedimiento

1. Atiende instrucciones de su docente.
2. Resuelve manualmente un problema de conteo aplicando la fórmula de inclusión-exclusión.
3. Implementa un script en Python que resuelva el mismo problema por fuerza bruta (iteración).
4. Compara el resultado teórico (manual) con el resultado computacional (script).
5. Analiza la eficiencia y aplicabilidad de ambos métodos.
6. Documenta el análisis y las conclusiones.
7. Discute de manera grupal los resultados y recibe retroalimentación docente.
8. Entrega el análisis y conclusiones.

Recursos de apoyo

- Computadora con acceso a internet
- Google Colab
- Notas de clase y diagrama de Venn ilustrando el principio de inclusión-exclusión.

Unidad IV: Teoría de grafos y árboles

Práctica 7: Modelado y visualización de grafos

Duración

08 horas

Procedimiento

1. Atiende instrucciones de su docente.
2. Modela un problema simple (p. ej., red de amigos) dibujando un grafo en papel.
3. Determina manualmente las propiedades del grafo (grado de vértices, ciclos, matriz de adyacencia).
4. Utiliza la biblioteca NetworkX de Python para crear una representación computacional del grafo.
5. Verifica las propiedades del grafo utilizando las funciones de NetworkX.
6. Genera una visualización del grafo con Matplotlib.
7. Documenta el modelo del grafo, sus propiedades y la visualización generada.
8. Discute de manera grupal los resultados y recibe retroalimentación docente.
9. Entrega modelo final y conclusiones.

Recursos de apoyo

- Computadora con acceso a internet.
- Google Colab.
- Tutorial básico de NetworkX.
- Galería de ejemplos de Matplotlib.

Práctica 8: Estructuras de árboles

Duración

08 horas

Procedimiento

1. Atiende instrucciones de su docente.
2. Dibuja un árbol con raíz que represente una estructura jerárquica.
3. Identifica manualmente los componentes del árbol (raíz, hojas, altura, relaciones).
4. Crea una representación del árbol en Python utilizando la biblioteca NetworkX.
5. Utiliza las funciones de la biblioteca para verificar las propiedades del árbol.
6. Documenta la estructura del árbol y los resultados del análisis computacional.
7. Discute de manera grupal los resultados y recibe retroalimentación docente.
8. Entrega la estructura final del árbol y los resultados obtenidos del análisis computacional.

Recursos de apoyo

- Computadora con acceso a internet.
- Google Colab.
- Ejemplos de uso de NetworkX.
- Notas de clase.

Unidad IV: Fundamentos de criptografía aplicada

Práctica 9: Simulación del algoritmo de RSA y Diffie-Hellman

Duración

06 horas

Procedimiento

1. Atiende instrucciones de su docente.
2. Realiza manualmente operaciones de aritmética modular con números pequeños.
3. Sigue en papel el flujo del algoritmo RSA para cifrar y descifrar un mensaje corto.
4. Utiliza la función `pow(base, exp, mod)` de Python para verificar los cálculos de exponenciación modular.
5. Implementa una simulación conceptual del proceso RSA en un script de Python.
6. Analiza el flujo del protocolo Diffie-Hellman con la analogía de la mezcla de pintura.
7. Documenta los resultados de los cálculos y las simulaciones, y presenta conclusiones.
8. Discute de manera grupal los resultados y recibe retroalimentación docente.
9. Entrega el documento con los resultados, simulaciones y conclusiones.

Recursos de apoyo

- Computadora con acceso a internet.
- Google Colab.
- Artículo o video que explique la analogía de la mezcla de pintura Diffie-Hellman.

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer de forma clara y formal:

- La metodología de trabajo.
- Criterios de evaluación y rúbricas.
- Reglas sobre la calidad y entrega de trabajos académicos.
- Derechos y obligaciones del docente y los estudiantes.

Esto debe quedar registrado y comunicado por medios institucionales (como Blackboard o correo institucional).

Estrategia de enseñanza

Son acciones didácticas planificadas por el docente para facilitar el logro de los aprendizajes esperados. Estas estrategias deben seleccionarse en función del tipo de competencia, el nivel cognitivo y la naturaleza del contenido.

Estos son ejemplos de estrategias, puede usar algunas de esta lista u otras que se consideren pertinentes:

- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Aprendizaje basado en preguntas
- Aprendizaje colaborativo
- Técnica expositiva

Estrategia de aprendizaje

Son enfoques que el estudiante emplea para organizar, adquirir, comprender y aplicar el conocimiento. Estas estrategias deben promover el desarrollo de competencias, el pensamiento crítico y el aprendizaje significativo.

Estos son ejemplos de estrategias, puede usar algunas de esta lista u otras que se consideren pertinentes:

- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Cuadros comparativos
- Infografías
- Aprendizaje mediado por TIC

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho al examen ordinario y extraordinario, cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	60%
– Tareas	15%
– Portafolio de ejercicios	25%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Ensley, D. E., & Crawley, J. W. (2021). *Discrete mathematics: Mathematical reasoning and proof with puzzles, patterns, and games* (2nd ed.). John Wiley & Sons.
- Jenkyns, T., & Stephenson, B. (2018). *Fundamentals of discrete math for computer science: A problem-solving primer* (2nd ed.). Springer. Recuperado el 11 de septiembre de 2025, de <https://doi.org/10.1007/978-3-319-70151-6>
- Jiménez Murillo, J. A. (2015). *Matemáticas para la computación* (3a. ed.). Alfaomega Grupo Editor.
- Johnsonbaugh, R. (2019). *Matemáticas Discretas* (8a ed.). Pearson Education.
- Kurgalin, S., & Borzunov, S. (2018). *The discrete math workbook: A companion manual for practical study* (1a ed.). Springer. Recuperado el 11 de septiembre de 2025, de <https://doi.org/10.1007/978-3-319-92645-2>
- Rosen, K. H. (2024). *Discrete mathematics and its applications* (8a ed.). McGraw-Hill.

Recursos de acceso libre o complementarios externos

- Desarrolladores de NetworkX. (s.f.). *Documentación de NetworkX*. Recuperado el 10 de septiembre de 2025, de <https://networkx.org/documentation/stable/>
- Equipo de desarrollo de SymPy. (s.f.). *Documentación de SymPy*. Recuperado el 10 de septiembre de 2025, de <https://docs.sympy.org/>
- Google. (s.f.). *Google Colaboratory*. Recuperado el 10 de septiembre de 2025, de <https://colab.research.google.com/>
- Levin, O. (2024). *Discrete Mathematics: An Open Introduction* (4ª ed.). Recuperado el 11 de septiembre de 2025, de <https://discrete.openmathbooks.org/dmoi4/frontmatter.html>
- Python Software Foundation. (s.f.). *Documentación de Python 3*. Recuperado el 10 de septiembre de 2025, de <https://docs.python.org/3/>
- Sweigart, A. (2025). *Automate the boring stuff with Python*. No Starch Press.

XII. PERFIL DOCENTE

La persona docente de esta asignatura deberá ser profesionalista con formación en el área de matemáticas o ciencias computacionales, con sólida experiencia en herramientas didácticas interactivas como Jupyter Notebooks o Google Colab. Debe ser capaz de guiar un proceso de enseñanza-aprendizaje activo y basado en proyectos, fomentando en el alumno la conexión entre la teoría matemática y la aplicación computacional.

LEARNING MODULE

I. GENERAL INFORMATION

School:	Faculty of Sciences, Ensenada
Major:	Bachelor of Science in Computer Science
Study Program:	2027-1
Learning Module Name:	Research Methods in Computer Science
Number:	
Duration distribution:	Theory/lecture Hours: 02 Workshop Hours: 02 Laboratory Hours: 00 Field Practice Hours: 00 Clinical Hours:00 Credits: 06
Stage:	Basic
Module Type:	Obligatory
Requirements:	None
Mode of delivery:	In-person
Design Team:	Eloísa del Carmen García Canseco
Approved by Assistant Dean (s):	Priscilla Elizabeth Iglesias Vázquez
Date:	May 21th 2026
Approval and registration:	

II. PURPOSE OF LEARNING MODULE

To introduce students to the fundamental principles and practices of conducting research in computer science. Through guided reading of primary literature, structured exercises in experimental design and data handling, small empirical or simulation projects, and oral and written reporting, students will develop the ability to pose simple research questions, choose basic methods to investigate them, and communicate findings clearly. The module fosters curiosity, critical thinking, collaboration, and methodological awareness while linking early research skills to core disciplinary knowledge.

It is located in the basic stage, it is compulsory, and belongs to the Social area of knowledge.

III. COMPETENCE OF THE LEARNING MODULE

To identify and formulate basic research questions in computer science to investigate computing problems through guided empirical or simulation studies, using introductory methods and reproducible workflows, with ethical citation and data practices, and to demonstrate clear technical writing, accurate use of terminology, appropriate basic analysis, and effective collaborative communication in individual or small-team projects.

IV. EVIDENCES OF LEARNING/ACHIEVEMENT

As evidence of learning, the students will elaborate on the following:

- Literature brief: Identify research question, summarize methods and results, note limitations, and state one small follow-up question. The vocabulary list must show contextual paraphrases for each term.
- Lab notebook: Time-stamped entries documenting setup, parameter choices, data collection steps, and anomalies. Emphasize reproducible notes rather than polished prose.
- Analysis notebook: Clean, commented code or notebook that loads data, computes descriptive statistics, produces at least one clear visualization with a caption, and contains a short interpretation cell. Notebooks must run end-to-end.
- Reproducible repository: Include README.md with run instructions, environment notes, data provenance, and license/attribution. Use Git commits to show progress.
- IMRaD report: Concise introduction, methods, results (with figure captions), brief discussion, references in ACM/IEEE style, and a similarity report.
- Presentation & peer review: Slide deck focused on problem, method, key result, and takeaway. Presenters answer questions; peers complete structured rubrics assessing clarity, methods, and scientific reasoning.

V. PROFESSIONAL COMPETENCY OF THE GRADUATE PROFILE TO WHICH THE LEARNING UNIT CONTRIBUTES

Design and evaluate intelligent computing solutions by applying principles of artificial intelligence, data mining, and integration of computational tools, with the purpose of supporting decision-making in complex situations typical of scientific, social, industrial, or technological contexts, acting with ethical responsibility, critical thinking, and interdisciplinary collaboration.

VI. CROSS-CUTTING THEME TO WHICH IT CONTRIBUTES

Excellence, Vanguard, Inclusion, and Interculturality.

VII. CONTENT

Unit I. Foundations of Research in Computer Science

Competency

Identify and formulate basic research questions in computer science by analyzing scholarly sources and distinguishing research objectives, methods, and limitations, using AI tools responsibly to support literature exploration and conceptual clarification.

Content

- 1.1 Nature and purpose of research in computing
- 1.2 Types of research (experimental, empirical, simulation-based, analytical)
- 1.3 Research questions, objectives, and problem formulation
- 1.4 Structure of a scientific paper (IMRaD)
- 1.5 Literature search strategies (ACM DL, IEEE Xplore, Google Scholar)
- 1.6. Use of artificial language tools for literature exploration
 - 1.6.1 Query refinement
 - 1.6.2 Summarizing papers responsibly
 - 1.6.3 Identifying key concepts and terminology
 - 1.6.4 Avoiding hallucinations and verifying claims
- 1.7 Reading and synthesizing scientific articles
- 1.8 Ethical principles: plagiarism, citation, and responsible use of artificial intelligence tools

Time Allotted

10 hours

Unit II. Research Design, Methods, Data Practices, and AI-Assisted Workflows

Competency

Apply introductory research methods and reproducible workflows to design guided empirical or simulation studies, selecting appropriate variables, metrics, and data-collection strategies, and using artificial intelligence tools to support coding, documentation, and exploratory analysis, all within ethical and methodological constraints.

Content

2.1 Research designs in computing (experiments, simulations, case studies, observational studies)

2.2 Variables, metrics, benchmarks, and evaluation criteria

2.3 Sampling, datasets, and basic data collection techniques

2.4 Introduction to data management and data provenance

2.5 Artificial intelligence tools for research design support

2.5.1 Brainstorming methodological options

2.5.2 Generating alternative metrics or evaluation strategies

2.5.3 Drafting protocols with human verification

2.6 Reproducible workflows: version control (Git), notebooks (Jupyter), documentation

2.7 Introductory data analysis: descriptive statistics, simple hypothesis testing

2.8 Artificial intelligence-assisted coding and analysis

2.8.1 Code generation with verification

2.8.2 Debugging assistance

2.8.3 Producing initial data visualizations

2.9 Visualization principles for computing research

2.10 Threats to validity (internal, external, construct, statistical)

2.11 Responsible use of artificial intelligence in research: transparency, documentation, and limits of automation

Time Allotted

14 hours

Unit III. Communication of Research Results

Competency

Communicate research findings through clear technical writing, ethical citation, and collaborative oral presentation, using artificial intelligence tools to support drafting, editing, and visualization while maintaining academic integrity and audience-appropriate design.

Content

3.1 Writing structured abstracts (IMRaD cues)

3.2 Short research reports in IMRaD format

3.3 Figures, tables, captions, and visual clarity

3.4 Citation styles in computing (ACM, IEEE) and reference managers

3.5 Artificial intelligence tools for scientific writing

- 3.5.1 Draft refinement and clarity improvement
- 3.5.2 Grammar and structure assistance
- 3.5.3 Generating alternative phrasings
- 3.5.4 Ensuring proper attribution and avoiding plagiarism
- 3.6 Preparing slide decks for research presentations
- 3.7 Oral communication techniques for scientific audiences
- 3.8 Simulated mini-conference: presentation, Q&A, peer review, professional conduct
- 3.9 Presentation design (outline generation, slide suggestions, speaker notes) with artificial intelligence tools
- 3.10 Reflection on teamwork, human-AI collaboration, and research process

Time Allotted

08 hours

VIII. STRUCTURE OF WORKSHOP PRACTICES

Unit I. Foundations of Research in Computer Science

Practice 1: Critical reading and AI-supported analysis of computing research papers

Duration

08 hours

Procedure

1. Follow the teacher's guidelines for developing the practice.
2. Select a peer-reviewed computing article from an indexed scientific journal.
3. Read the article to identify the research question, objectives, methods, results, and limitations.
4. Use an AI/LLM tool to:
 - a. Generate a preliminary summary (to be verified manually).
 - b. Extract key terminology and definitions.
 - c. Formulate clarifying questions about methods or concepts.
5. Verify all AI-generated content against the original paper.
6. Identify biases, methodological limitations, and ethical considerations.
7. Highlight and define 15–20 technical terms with contextual paraphrases.
8. Write a 1–2 page brief summarizing the main findings.
9. Receive teacher feedback.
10. Submit the final version with revisions addressing the feedback.

Support resources

- Computer
- Internet
- Scientific databases
- AI/LLM tool (responsible use)
- Reference manager

Unit II. Research Design, Methods, Data Practices, and AI-Assisted Workflows

Practice 2: Designing a mini-research protocol with AI-assisted brainstorming

Duration

08 hours

Procedure

1. Follow the teacher's guidelines for developing the practice.
2. Select a simple computing problem suitable for a small experiment or simulation.
3. Formulate a research question and objectives.
4. Use an AI/LLM tool to:
 - a. Brainstorm possible methods, variables, and metrics.
 - b. Generate alternative experimental designs.
 - c. Suggest potential datasets or simulation approaches.
5. Evaluate AI suggestions for feasibility, correctness, and ethical implications.
6. Define variables, metrics, and evaluation criteria.
7. Draft a 1–2 page research protocol including:
 - d. Research question
 - e. Method
 - f. Data collection plan
 - g. Expected outputs
 - h. Ethical and reproducibility checklist
8. Receive teacher feedback and revise the protocol.
9. Submit the final protocol.

Support resources

- Computer
- Internet
- Git/GitHub or a version control platform
- AI/LLM tool (responsible use)
- Datasets or simulated tools

Practice 3: Data Collection, Analysis, and Reproducible Workflows

Duration

08 hours

Procedure

1. Follow the teacher's guidelines for the development of the practice.
2. Set up a small experiment or simulation based on the approved protocol.
3. Collect or generate a small dataset.
4. Document all steps in a lab notebook (digital or physical).
5. Create a Jupyter notebook (or equivalent) including:
 - a. Data loading and cleaning
 - b. Descriptive statistics
 - c. At least one visualization with caption
6. Use an AI/LLM tool to:
 - a. Suggest code snippets (verified manually)
 - b. Propose alternative visualizations
 - c. Explain statistical concepts or errors
7. Ensure all AI-generated code is tested and corrected as needed.
8. Create a reproducible project folder or Git repository with:
 - a. Notebook
 - b. Data sample
 - c. README with run instructions
 - d. Data provenance notes
9. Submit the notebook and repository link.
10. Receive teacher feedback

Support resources

- Computer
- Internet
- Jupyter notebook or equivalent
- Git/GitHub or a version control platform
- AI/LLM tool (responsible use)
- Datasets or simulation environment

Unit III. Communication of Research Results

Practice 4: Scientific Writing and Conference-Style Presentation with AI-Supported Editing

Duration

08 hours

Procedure

1. Follow the teacher's guidelines for developing the practice.
2. Write a structured abstract (≤ 200 words) using IMRaD cues.
3. Write a short IMRaD report (600–900 words) including at least one figure.
4. Use an AI/LLM tool to:
 - a. Improve clarity and structure
 - b. Suggest alternative phrasings

- c. Check grammar and coherence
5. Verify all AI-assisted edits and ensure originality.
6. Generate a similarity report ($\leq 10\%$).
7. Prepare a 6–8 slide deck summarizing the mini-project.
8. Use AI tools to refine slide design or speaker notes (with verification).
9. Deliver an 8–10 minute conference-style presentation with 3–5 minutes of Q&A.
10. Complete two peer-review forms (one abstract, one presentation).
11. Submit the final report, slides, and peer reviews.
12. Receive teacher feedback

Support resources

- Computer
- Internet
- Presentation software
- AI/LLM tool (responsible use)
- Reference manager
- Similarity checker

IX. METHODOLOGY AND STRATEGIES

Course framework

On the first day of class, each instructor must establish the working methods, evaluation criteria, quality of academic work, and teacher-student rights and responsibilities.

Teaching strategies

- Project-based method
- Question-based learning
- Expository technique
- Debates
- Guided discussion
- Guided instruction
- Forums

Learning strategies

- Documentary research
- Presentation of case resolution
- Collaborative work
- Expository technique
- Comparative charts

X. EVALUATION CRITERIA

Accreditation criteria

To be eligible for ordinary and extraordinary exams, each student must meet the attendance percentages established by the current School Statute. Grades are on a scale from 0 to 100, with a minimum passing grade of 60.

Assessment criteria

Critical Reading	10 %
Mini research protocol	20 %
Lab Notebooks	30 %
Reproducible repository	10 %
IMRaD Report	20%
Oral presentation and peer-review	10%
Total	100 %

XI. BIBLIOGRAPHY

Resources of the UABC physical and digital library

Berndtsson, M. (Ed.). (2008). *Thesis projects: A guide for students in computer science and information systems* (2nd ed.). Springer. [classic]

Databases available at the UABC Library. <https://bibliotecas.uabc.mx/>Marder, M. P. (2010). *Research methods for science*. Cambridge University Press. [classic]

Silyn-Roberts, H. (2013). *Writing for Science and Engineering: papers, presentations, and reports* (2nd ed.) [Electronic resource]. Elsevier. [classic]

Wallwork, A. (2011). *English for writing research papers* [Electronic resource]. Springer. [classic]

Wilson, G., Bryan, J., Cranston, K., Kitzes, J., Nederbragt, L., & Teal, T. K. (2017). *Good enough practices in scientific computing*. PLoS Computational Biology, 13(6), 1–20. <https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1005510>

Freely accessible or external complementary resources

European Association of Science Editors. (2021). *Writing science in English: A guide for authors and editors*. ScienceScape Editing. <https://ease.org.uk/wp-content/uploads/2021/10/Writing-Science-in-English-2021.pdf>

Gopen, G. D., & Swan, J. A. (1990). *The science of scientific writing*. American Scientist, 78(6), 550–558. https://www.usenix.org/sites/default/files/gopen_and_swan_science_of_scientific_writing.pdf

Iowa State University Library. (s. f.). *Free scientific research resources*. <https://instr.iastate.libguides.com/c.php?g=49318&p=318011>

Miao, F., & Holmes, W. (2023). *Guidance for generative AI in education and research*. UNESCO. <https://unesdoc.unesco.org/ark:/48223/pf0000386693>

XII. TEACHER PROFILE

A bachelor's degree in Computer Science or a related field, preferably with postgraduate studies and extensive experience in scientific research and English language proficiency, minimum B2 (upper intermediate level). Experience in teaching and pedagogical skills. A respectful, inclusive, honest, responsible, proactive, and analytical individual who encourages both individual and teamwork.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Análisis de Algoritmos
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 03 Horas Laboratorio: 00 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Disciplinaria
Carácter:	Obligatoria
Requisito:	Estructura de Datos y Algoritmos
Modalidad de impartición:	Presencial
Equipo de diseño:	Everardo Gutiérrez López José Ángel González Fraga
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

El estudio del análisis de los algoritmos representa en el perfil de los egresados de ciencias computacionales una primera etapa de abstracción y razonamiento lógico matemático para identificar las limitaciones de los modelos actuales de computación. Esto se enmarca en el área de la teoría de la computación, la cual se encarga de establecer los fundamentos de las ciencias computacionales de manera formal. Esta asignatura tiene como finalidad que el alumnado sea capaz de realizar un análisis lógico-matemático de las soluciones algorítmicas propuestas para diferentes problemáticas, así como de proponer soluciones alternativas utilizando técnicas avanzadas de diseño de algoritmos. Se encuentra en la etapa disciplinaria, es de carácter obligatoria y pertenece al área de conocimiento de Programación e Ingeniería de Software. Se recomienda el cursarla después de haber acreditado Estructura de Datos y Algoritmos.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Evaluar algoritmos computacionales, por medio del uso de las metodologías de análisis de algoritmos, para determinar su corrección y eficiencia asintótica en relación con la entrada de datos de un problema computacional, con una actitud analítica y autocrítica siguiendo un razonamiento lógico matemático.

IV. EVIDENCIA DE APRENDIZAJE

Elabora un portafolio el cual contenga los ejercicios de análisis con las demostraciones lógico-matemáticas donde se evalúe los algoritmos computacionales identificando su eficiencia y corrección. El portafolio deberá incluir como secciones mínimas: portada, introducción, ejercicios de análisis y conclusiones.

V. COMPETENCIAS PROFESIONALES DEL PERFIL DE EGRESO A LAS QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

Desarrollar software de base eficiente y robusto mediante la integración de conocimientos en arquitectura y redes de computadoras y los fundamentos de compiladores y sistemas operativos, para optimizar el rendimiento, la gestión de recursos y la comunicación entre sistemas, asegurando aplicaciones funcionales, seguras, escalables y sostenibles, con una actitud ética, responsable y visionaria.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Pensamiento Crítico.

VII. DESARROLLO POR UNIDADES

Unidad I. 1. Herramientas de análisis de algoritmos

Competencia

Distinguir las herramientas de análisis de algoritmos para contrastar su desempeño en tiempo y espacio mediante la caracterización de sus propiedades de eficiencia y corrección, de manera organizada, crítica y responsable.

Contenido

- 1.1. Definiciones e importancia de los algoritmos
- 1.2. Corrección de un algoritmo
- 1.3. Eficiencia en tiempo y espacio
- 1.4. Análisis asintótico
- 1.5. Solución de recurrencias
- 1.6. Análisis amortizado

Duración

10 horas

Unidad II. Analizando algoritmos de diferentes paradigmas de diseño

Competencia

Discriminar los principales paradigmas para el diseño de algoritmos computacionales mediante el análisis comparativo de sus características para decidir la pertinencia de su aplicación a casos de estudio prácticos, con una actitud crítica, analítica y propositiva.

Contenido

- 2.1. Paradigma divide y vencerás
- 2.2. Programación dinámica
- 2.3. Estrategias voraces
- 2.4. Métodos probabilísticos
- 2.5. Algoritmos heurísticos

Duración

6 horas

Unidad III. Algoritmos sobre estructuras no lineales

Competencia

Distinguir los principales algoritmos sobre estructuras no lineales, para el manejo eficiente de información en grafos, mediante el análisis de las problemáticas que pueden representarse en esas estructuras, con una actitud crítica, organizada y propositiva.

Contenido

- 3.1. Algoritmos sobre grafos
- 3.2. Algoritmos elementales para grafos
- 3.3. Árboles de expansión mínima
- 3.4. Camino más corto de fuente única y múltiple
- 3.5. Redes de flujo

Duración

6 horas

Unidad IV. Complejidad computacional

Competencia

Detectar las características y limitaciones de los modelos computacionales realizando una abstracción de sus componentes para determinar su aplicabilidad, con una actitud crítica, analítica y propositiva.

Contenido

- 4.1. Introducción a la teoría de la computación
- 4.2. Lenguajes formales y teoría de la computabilidad
- 4.3. Clases de complejidad P, NP, CoNP, NP-Hard y NP-Complete
- 4.4. Ejemplos de problemas NP-Completo

Duración

6 horas

Unidad V. Tópicos selectos de aplicación

Competencia

Evaluar las herramientas de análisis y diseño para determinar el desempeño de los algoritmos computacionales mediante la determinación de su desempeño en la aplicación a problemáticas teórico-prácticas, con una actitud organizada, crítica, propositiva.

Contenido

- 5.1. Polinomios y la FFT

- 5.2. Teoría de números y sistemas de encriptamiento
- 5.3. Manipulación de cadenas
- 5.4. Algoritmos multihilo
- 5.5. Algoritmos de aproximación

Duración

4 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

Unidad I. Herramientas de análisis de algoritmos

Práctica 1: Análisis de la corrección de algoritmos computacionales

Duración:

6 horas

Procedimiento:

1. Atiende indicaciones para el taller.
2. Aplica el método de invariante de lazo en algoritmos de ejemplo.
3. Elabora el reporte de la práctica.
4. Entrega el reporte al docente.
5. Recibe retroalimentación docente.

Recursos de apoyo:

- Bibliografía.
- Lápiz.
- Papel.
- Recursos digitales.

Práctica 2: Análisis de la eficiencia de algoritmos computacionales

Duración:

6 horas

Procedimiento:

1. Atiende las indicaciones del docente.
2. Analiza el desempeño en tiempo y espacio de algoritmos de ejemplo.
3. Utiliza las técnicas de análisis asintótico y de resolución de recurrencias.
4. Elabora el reporte final de la práctica.
5. Recibe retroalimentación del docente.

Recursos de apoyo:

- Bibliografía.
- Lápiz.
- Papel.
- Recursos digitales.

Unidad II. Analizando algoritmos de diferentes paradigmas de diseño

Práctica 3: Análisis aplicado a algoritmos de diferentes paradigmas de diseño

Duración:

8 horas

Procedimiento:

1. Atiende las indicaciones del docente.
2. Identifica los componentes principales de los paradigmas de diseño de algoritmos en la generación de soluciones a problemas prácticos.
3. Propone la utilización de las herramientas de análisis de algoritmos sobre casos de ejemplo de los paradigmas de diseño.
4. Elabora el reporte final de la práctica.
5. Recibe retroalimentación del docente.

Recursos de apoyo:

- Bibliografía.
- Lápiz.
- Papel.
- Recursos digitales.

Unidad III: Algoritmos sobre estructuras no lineales

Práctica 4: Algoritmos sobre estructuras no lineales

Duración:

8 horas

Procedimiento:

1. Atiende las indicaciones del docente.
2. Analiza las estructuras de almacenamiento no lineal.
3. Compara algoritmos para almacenamiento y manipulación de información en las estructuras de datos no lineales.
4. Elabora el reporte final de la práctica.
5. Recibe retroalimentación del docente.

Recursos de apoyo:

- Bibliografía.
- Lápiz.
- Papel.
- Recursos digitales.

Unidad IV: Complejidad computacional

Práctica 5: Análisis comparativo de algoritmos

Duración:

8 horas

Procedimiento:

1. Atiende las indicaciones del docente.
2. Selecciona algoritmos de ejemplo para ilustrar los temas de clase.
3. Utiliza herramientas de análisis de algoritmos para preparar una exposición explicando los algoritmos de ejemplo seleccionados.
4. Expone ante el grupo el análisis de los algoritmos seleccionados.
5. Elabora y entrega una presentación en diapositivas de la práctica.
6. Recibe retroalimentación del docente.

Recursos de apoyo:

- Bibliografía.
- Pintarrón.
- Plumones.
- Equipo de cómputo.
- Recursos digitales.

Unidad V: Tópicos selectos de aplicación

Práctica 6: Análisis integral de algoritmos

Duración:

12 horas

Procedimiento:

1. Atiende las indicaciones del docente.
2. Evalúa soluciones computacionales a problemas teórico-prácticos.
3. Analiza los algoritmos utilizados.
4. Elabora el reporte final de la práctica.
5. Recibe retroalimentación del docente.

Recursos de apoyo:

- Bibliografía.
- Lápiz.
- Papel.

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer de forma clara y formal:

- La metodología de trabajo.
- Criterios de evaluación y rúbricas.
- Reglas sobre la calidad y entrega de trabajos académicos.
- Derechos y obligaciones del o la docente y del estudiantado

Esto debe quedar registrado y comunicado por medios institucionales (como Blackboard o correo institucional).

Estrategia de enseñanza

- Estudio de caso
- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Aprendizaje basado en preguntas
- Técnica expositiva
- Debates
- Discusión guiada
- Instrucción guiada
- Foros
- Discusiones críticas

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Visitas a campo
- Mapas mentales
- Mapas conceptuales
- Cuadros comparativos
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	60%
– Prácticas de taller	10%
– Ejercicios de análisis (portafolio)	20%

– Presentaciones orales	10%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Cormen, T.H. (2013). *Algorithms Unlocked*. The MIT Press. [clásica]
- Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C. (2022). *Introduction to algorithms*. The MIT press.
- Dasgupta, S., Papadimitriou, C., Vazirani U. (2006). *Algorithms*. McGraw-Hill. [clásica]
- Garey, M.R., Johnson, D.S., Freeman, W.H. (1979). *Computers and Intractability: A Guide to the Theory of NP.Completeness*. [clásica]
- Motwani, R., Raghavan, P. (1995). *Randomized Algorithms*. Cambridge University Press. [clásica]
- Sedgewick, R. & Flajolet, P. (2013). *Introduction to the Analysis of Algorithms*. Addison-Wesley Professional. [Clásica]
- Sedgewick, R. & Wayne, K. (2011). *Algorithms* (4th Edition). Addison-Wesley Professional. [Clásica]
- Sipser, M. (2012). *Introduction to the Theory of Computation*. Cengage Learning. [Clásica]
- Vazirani, V.V. (2004). *Approximation Algorithms*. Springer. [Clásica]

Recursos de acceso libre o complementarios externos

- ACM SIGACT. (2025). *ACM Special Interest Group on Algorithms and Computation Theory*. <http://www.sigact.org/>
- IEEE Computer Society. (2025). *IEE Computer Society*. <http://www.computer.org/>
- Roughgarden, T. (2021). *Beyond the Worst-Case Analysis of Algorithms*. Cambridge University Press
- Skiena, S. (2010). *The Algorithm Design Manual*. Springer. [Clásica]

XII. PERFIL DOCENTE

Licenciatura en Ciencias computacionales, o área afín; preferentemente con estudios de posgrado y dos años de experiencia de práctica docente, además deberá tener conocimiento en las técnicas

de análisis de algoritmos, en técnicas de diseño de algoritmos y en las bases de la teoría de la computación que permitan analizar los modelos computacionales históricos y vigentes, debe tener capacidad de abstracción además de una actitud analítica, crítica y de razonamiento lógico matemático.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Metodología de la Programación
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 03 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Disciplinaria
Carácter:	Obligatoria
Requisito:	Programación Orientada a Objetos
Modalidad de impartición:	Presencial
Equipo de diseño:	Alma Rocío Cavazos Marín Alberto Leopoldo Morán y Solares
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vazquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La unidad de aprendizaje Metodología de la Programación se ubica en la etapa disciplinaria, con carácter obligatorio, dentro del área de conocimiento de Programación e Ingeniería de Software. Su finalidad es que el estudiantado comprenda y aplique métodos, técnicas y herramientas propias de las metodologías de desarrollo y producción de software de calidad, mediante la elaboración de soluciones de pequeña o mediana escala orientadas a atender las necesidades de un cliente real. Asimismo, tiene como propósito fortalecer el pensamiento lógico, la capacidad de análisis y la toma de decisiones, promoviendo el trabajo colaborativo, la responsabilidad profesional y la aplicación de buenas prácticas en el desarrollo de software.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Desarrollar un proyecto de software de pequeña o mediana escala mediante la aplicación de una metodología de programación orientada al proceso de desarrollo de software, considerando sus distintas etapas y el uso de las herramientas especializadas, para atender las necesidades de un cliente real, demostrando una actitud objetiva, crítica, con disciplina y ética profesional.

IV. EVIDENCIA DE APRENDIZAJE

- Entrega de un portafolio de evidencias que integra minutas de reuniones de avance periódicas y documentación técnica de cada etapa del proceso de desarrollo.
- Un proyecto de software funcional acorde a los requerimientos y la documentación final técnica del proyecto.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia y Vanguardia.

VII. DESARROLLO POR UNIDADES

1. Unidad I. Fundamentos de la Metodología de la Programación

Examinar conceptos, ámbito, terminología y modelos del desarrollo de software, así como aspectos del control del mismo proceso mediante la revisión de referencias especializadas y diversas metodologías de desarrollo, para contextualizar la importancia de esta actividad en las organizaciones, con disciplina, ética profesional, actitud objetiva y crítica.

Contenido

- 1.1. Principios del desarrollo de software
- 1.2. El proceso y el producto
- 1.3. Proceso de desarrollo de software
 - 1.3.1. Desarrollo (Análisis, diseño, implementación, pruebas, mantenimiento)
 - 1.3.2. Control (Documentación, calidad, validación y verificación, configuración)
- 1.4. Modelos de desarrollo de software

Duración

04 horas

Unidad II. Métricas y técnicas del desarrollo de software

Competencia

Analizar métricas y técnicas que se utilizan para el desarrollo de software, a través de una revisión de referencias especializadas en metodologías de desarrollo, gestión de calidad y control de configuraciones para su aplicación en el desarrollo de software, con disciplina y actitud objetiva.

Contenido

- 2. Métricas y técnicas del desarrollo de software
 - 2.1. Principios del desarrollo de software
 - 2.2. Calidad del proceso y del producto
 - 2.3. Documentación
 - 2.3.1. Estándares
 - 2.4. Administración de configuración y control de cambios
 - 2.4.1. Administración de versiones
 - 2.5. Validación y verificación
 - 2.5.1. Revisiones técnicas

Duración

04 horas

Unidad III. Fase de análisis

Competencia

Diseñar modelos de análisis de un problema propuesto mediante el uso de las herramientas y técnicas orientadas a objetos para proponer una aplicación de software que resuelva la problemática de forma específica y congruente con los procesos y los requerimientos del cliente, con actitud creativa, crítica y colaborativa.

Contenido

- 3.1. Aspectos de análisis
- 3.2. Análisis del dominio
- 3.3. El proceso de análisis
 - 3.3.1. Modelado funcional (Casos de uso)
 - 3.3.2. Modelado estático (Modelo de clases)
 - 3.3.3. Modelado dinámico (Diagrama de interacción)
 - 3.3.4. Diagramas de estado
- 3.4. Documentación de análisis

Duración

08 horas

Unidad IV. Fase de diseño

Competencia

Diseñar modelos de alto y bajo nivel de posibles soluciones de software mediante el uso de las herramientas y técnicas orientadas a objetos para proponer de manera exitosa un proyecto de software que resuelva la problemática especificada por el cliente de forma profesional y congruente con los procesos de la entidad que demanda los servicios, con una actitud colaborativa, disciplinada y creativa.

Contenido

- 4.1. Aspectos de diseño.
- 4.2. Proceso de Diseño del sistema
 - 4.2.1. Particionar el sistema en subsistemas
 - 4.2.2. Concurrencia entre subsistemas
 - 4.2.3. Administración de subsistemas
 - 4.2.4. Administración de datos
 - 4.2.5. Administración de recursos
 - 4.2.6. Comunicación entre subsistemas
- 4.3. Proceso de Diseño de Objetos
 - 4.3.1. Descripción de objetos
- 4.4. Patrones de diseño
- 4.5. Revisiones de diseño

Duración

08 horas

Unidad V. Fase de implementación y pruebas**Competencia**

Desarrollar aplicaciones de software de escala pequeña o mediana a través del uso de las herramientas y técnicas de construcción y prueba del software para resolver un problema propuesto por el cliente de manera satisfactoria con actitud responsable, creativa y colaborativa.

Contenido

- 5.1. Implementación
 - 5.1.1. Evaluación de lenguajes
 - 5.1.2. El modelo de clases
 - 5.1.3. Características de orientación a objetos
 - 5.1.4. Estrategias de implementación
- 5.2. Fundamentos de Pruebas
 - 5.2.1. Estrategias de pruebas
 - 5.2.2. Procedimientos de pruebas
 - 5.2.3. Diseño de casos de pruebas
 - 5.2.4. Resultados de las pruebas
- 5.3. Despliegue del proyecto

Duración

08 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I: Desarrollo de Software

Práctica 1: Comunicación

Duración

03 horas

Procedimiento

1. Atiende instrucciones docentes de la práctica a realizar.
2. Integra equipos de trabajo.
3. Define un tema para tu proyecto en función de un problema real.
4. Elabora el artefacto "Cuestionario" que permita conocer los requerimientos del cliente para el desarrollo del proyecto.
5. Aplica el cuestionario en una entrevista con el cliente, si es más de un cliente (comunidad) enviarlo por medios digitales (ej. Google Forms).
6. Elabora el artefacto "Resumen de Requerimientos"
7. Resume las respuestas del cliente y pon en contexto cada uno de los requerimientos en una redacción breve.
8. Elabora el artefacto "Lista de requerimientos", es una lista numerada de cada uno de los requerimientos identificados.
9. Enumera con una clave que denote que cada elemento pertenece a la lista de requerimientos (ej. R1, R2, ...Rn), esto facilitará hacer referencia a los mismos en las próximas etapas de desarrollo.
10. Entrega reporte de las actividades realizadas durante la práctica.
11. Recibe retroalimentación y evaluación docente.

Recursos de apoyo

- Computadora
- Sistema generador de formularios
- Editor de textos

Práctica 2: Planificación

Duración

03 horas

Procedimiento

1. Atiende instrucciones docentes de la práctica a realizar.
2. Integra equipos de trabajo.
3. Define el nombre de su empresa ficticia de desarrollo de software y logotipo.
4. Declara los roles de los integrantes del equipo:
 - a. PRODUCT OWNER (Dueño del producto).
 - b. SCRUM MASTER (Responsable del equipo).

- c. DEVELOPMENT TEAM (Equipo de desarrollo).
5. Designa las prioridades de lista de requerimientos-Backlog.
6. Genera un cronograma con base en las fases del Marco Teórico Genérico del Desarrollo de Software.
7. Elabora una cotización de acuerdo al presupuesto y requerimientos del cliente.
8. Entrega reporte de las actividades realizadas durante la práctica, para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Editor de textos

Práctica 3: Planteamiento de escenarios

Duración

03 horas

Procedimiento

1. Atiende instrucciones docentes de la práctica a realizar.
2. Comenta con tu equipo los escenarios identificados en la entrevista o conversaciones con tu cliente, durante la comunicación. Los escenarios describen los procesos, situaciones o problemáticas cotidianas o excepcionales a los que se enfrenta el negocio para el cual se desarrolla el sistema.
3. Plantea al menos tres escenarios de todos los posibles: lo esperado, problemas frecuentes, las excepciones y lo ideal.
4. Realiza una representación de los escenarios planteados por tu equipo, haciendo uso de una de las siguientes opciones: *Storyboard*, diagrama, video, animación, dibujos entre otros.
5. Entrega reporte de las actividades realizadas durante la práctica.
6. Recibe retroalimentación y evaluación docente.

Recursos de apoyo

- Computadora
- Aplicaciones para edición de storyboards, diagramas, videos, entre otros.
- Editor de textos.

Práctica 4: Modelado de casos de uso

Duración

03 horas

Procedimiento

1. Atiende instrucciones docentes de la práctica a realizar.
2. Diseña los casos de uso utilizando la simbología UML, cada diseño debe tener un pie de figura, describiendo el caso de uso en base a la lista de requerimientos.

3. Utiliza nombres cortos y significativos para los elementos del diseño (*estudiante, registro, autenticar, desplegar_informacion, etc*), recuerda que estos se convertirán en nombres de clases, atributos, métodos o componentes del sistema.
4. Entrega reporte de las actividades realizadas durante la práctica.
5. Recibe retroalimentación y evaluación docente.

Recursos de apoyo

- Computadora
- Aplicación de diagramación UML
- Editor de textos.

Unidad II: Métricas y técnicas del desarrollo de software

Práctica 5: Modelado del diagrama de clases

Duración

03 horas

Procedimiento

1. Atiende instrucciones docentes de la práctica a realizar.
2. Diseña las clases empleando el lenguaje de diagramación UML, con base en los los Casos de Uso obtenidos en la práctica anterior.
3. Diseña el diagrama de clases que relaciona las clases con herencia, asociación, dependencia, agregación, composición e implementación.
4. Entrega reporte de las actividades realizadas durante la práctica.
5. Recibe retroalimentación y evaluación docente.

Recursos de apoyo

- Computadora
- Aplicación de diagramación UML
- Editor de textos.

Práctica 6: Construcción

Duración

03 horas

Procedimiento

1. Atiende instrucciones docentes de la práctica a realizar.
2. Implementa en un lenguaje de programación Orientado a Objetos, dos elementos de mayor prioridad en tu lista de requerimientos, *Backlog*,
3. Desarrolla cada módulo basándose en el diseño de tu diagrama de clases.
4. Sigue el marco de trabajo SCRUM para el seguimiento de los *Sprint* de cada componente y toma el rol que te fue designado para la elaboración de tu reporte.
5. Entrega reporte de las actividades realizadas durante la práctica.

6. Recibe retroalimentación y evaluación docente.

Recursos de apoyo

- Computadora
- IDE para programación Orientada a Objetos
- Editor de textos.

Práctica 7: Diseño estático de la interfaz

Duración

03 horas

Procedimiento

1. Atiende instrucciones docentes de la práctica a realizar.
2. Elabore el diseño de la interfaz, el cual deberá ser descrito mediante el diseño de:
 - a. *Diseña el Prototipo estático de la interfaz*, consiste en el diseño de las pantallas o ventanas, donde se presentan los menús, contenidos, gráficos, panel de control, formularios, entre otros elementos que intervienen en la experiencia y satisfacción del usuario haciéndole sentir en control del sistema.
 - b. *Diseña el Mapa de Navegación*, consiste del diagrama que muestra la conexión entre pantallas o ventanas, mensajes de error, advertencias o instrucciones, de manera que permita visualizar de manera general cómo será la navegación en el sistema.
 - c. *Diseña el esquema de menús*, muestra la relación y profundidad a la que se encuentran los menús del sistema. Todo elemento del sistema debe ser accesible por medio de un menú principal por el cual se tiene acceso a sub-menús. Se recomienda que la jerarquía de menús no sea muy profunda, es recomendable un máximo de profundidad de 3 niveles, a menos que se justifique la necesidad de niveles a mayor profundidad.
3. Entrega reporte de las actividades realizadas durante la práctica.
4. Recibe retroalimentación y evaluación docente.

Recursos de apoyo

- Computadora
- Aplicación de diagramación UML
- Editor de textos.

Práctica 8: Diseño dinámico de la interfaz

Duración

03 horas

Procedimiento

1. Atiende instrucciones docentes de la práctica a realizar.
2. Elabore el diseño de la interfaz del sistema:

- a. *Implementa el prototipo dinámico de la interfaz*, consiste en el diseño de las pantallas o ventanas, donde se presentan los menús, contenidos, gráficos, panel de control, formularios, entre otros elementos que intervienen en la experiencia y satisfacción del usuario haciéndole sentir en control del sistema.
 - b. Comprueba que la ejecución de la implementación cumple con el seguimiento del diseño del Mapa de Navegación y el esquema de menús, propuesto.
3. Entrega reporte de las actividades realizadas durante la práctica.
 4. Recibe retroalimentación y evaluación docente.

Recursos de apoyo

- Computadora
- IDE para programación Orientada a Objetos
- Editor de textos.

Unidad III: Fase de análisis

Práctica 9: Verificación y pruebas

Duración

04 horas

Procedimiento

1. Atiende instrucciones docentes de la práctica a realizar.
2. Elabora el plan de pruebas, basándose en los casos de uso y requerimientos del proyecto.
3. Aplica el plan de pruebas de acuerdo a lo descrito en las 5 etapas de las pruebas, haciendo regresión hasta no tener ningún error o cambio.
4. Sube aquí tu plan y el reporte de pruebas.
5. Entrega reporte de las actividades realizadas durante la práctica.
6. Recibe retroalimentación y evaluación docente.

Recursos de apoyo

Práctica 10: Verificación de la Calidad

Duración

03 horas

Procedimiento

1. Atiende instrucciones docentes de la práctica a realizar.
2. Evalúa tu prototipo de proyecto con la tabla de verificación de calidad con base a los Factores y Métricas de Calidad de McCall.
3. En la tabla marca con una cruz los criterios que cumplen y relacionan cada factor con cada métrica.
4. Justifica tu evaluación, considerando los caso de uso y los requerimientos del cliente para contextualizar tu evaluación de calidad.

5. Entrega reporte de las actividades realizadas durante la práctica.
6. Recibe retroalimentación y evaluación docente.

Recursos de apoyo

- Computadora
- IDE para programación Orientada a Objetos
- Editor de textos.

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Técnica expositiva
- Discusión guiada
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Mapas mentales
- Mapas conceptuales
- Cuadros comparativos
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	30%
– Portafolio de evidencias	30%
– Proyecto de software	40%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Booch, G. (2006). *El lenguaje unificado de modelado: guía del usuario* (2a ed.). Pearson, Addison Wesley Longman. [clásico]

Pressman, R. (2020). *Software Engineering A practitioner's approach* (9th ed.). MacGraw Hill.

Recursos de acceso libre o complementarios externos

Biblioteca Digital ACM, artículos varios en Ingeniería de Software y temas relacionados de Ciencias Computacionales (Disponible en <http://dl.acm.org> a través de la biblioteca UABC, y en la Cimarred de los campus) – en Inglés.

Biblioteca Digital IEEE Xplore, artículos varios en Ingeniería de Software y temas relacionados de Ciencias Computacionales (Disponible en <http://ieeexplore.ieee.org/Xplore/home.jsp> a través de biblioteca UABC, y la Cimarred de los campus) – en Inglés.

XII. PERFIL DOCENTE

Profesionista con licenciatura en Ciencias Computacionales o área afín, preferentemente con estudios de posgrado, con experiencia comprobable en la aplicación de la ingeniería de software. Deberá contar con conocimientos en lógica de programación, análisis, diseño e implementación de programas de computadora, así como en metodologías de programación y desarrollo de software. Asimismo, se espera que sea una persona creativa, que fomente el trabajo colaborativo, el pensamiento crítico en el estudiantado y fortalezca la ética profesional.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Organización y Arquitectura de Computadoras
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 02 Horas Laboratorio: 00 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:06
Etapas de formación:	Disciplinaria
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Evelio Martínez Martínez Eloisa del Carmen García Canseco Gustavo Enrique Ramos Alcaraz
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Brindar conocimientos fundamentales sobre la estructura, los componentes y el funcionamiento de los sistemas de cómputo, así como la interacción entre hardware y software para el procesamiento eficiente de la información.

El estudiantado desarrolla habilidades para analizar, evaluar y optimizar arquitecturas, comprender jerarquías de memoria, técnicas de entrada y salida, manejo de instrucciones y procesamiento paralelo. La asignatura fomenta actitudes de pensamiento lógico, precisión, responsabilidad en el uso de la tecnología, trabajo colaborativo y disposición para la actualización continua. Tiene carácter obligatorio, pertenece a la disciplinaria y se sugiere contar con conocimientos previos sobre sistemas numéricos, fundamentos de lógica y electrónica digital

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Analizar la estructura, funcionamiento e interrelación de los componentes de un sistema de cómputo, mediante el uso de modelos, diagramas, herramientas de simulación y normas técnicas, con el propósito de diseñar y optimizar arquitecturas eficientes, con pensamiento crítico, responsabilidad, e innovación.

IV. EVIDENCIA DE APRENDIZAJE

Diseñar, implementar y probar en un simulador de circuitos (e.g. *LogicSim*) un circuito digital que integre compuertas lógicas, flip-flops y bloques combinacionales y secuenciales (como sumadores, multiplexores o contadores), documentando el proceso de construcción, los diagramas de diseño y las tablas de verdad correspondientes, para finalmente validar el correcto funcionamiento del sistema a través de simulaciones prácticas, demostrando con ello su capacidad de aplicar los principios del álgebra booleana y del diseño digital en la resolución de problemas, así como su responsabilidad y rigor técnico en la elaboración del proyecto.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Desarrollar software de base eficiente y robusto mediante la integración de conocimientos en arquitectura y redes de computadoras y los fundamentos de compiladores y sistemas operativos, para optimizar el rendimiento, la gestión de recursos y la comunicación entre sistemas, asegurando aplicaciones funcionales, seguras, escalables y sostenibles, con una actitud ética, responsable y visionaria.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia, inclusión, vanguardia, innovación social.

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos de la evolución y estructura de las computadoras

Competencia

Analizar los conceptos fundamentales sobre la naturaleza, evolución y estructura de las computadoras, mediante el estudio del modelo de *John Von Neumann*, los paradigmas de computación y programación, así como los factores de rendimiento, con el propósito de comprender la organización básica del sistema y su impacto en la tecnología actual, demostrando curiosidad científica, responsabilidad y pensamiento crítico.

Contenido

- 1.1. Qué es una computadora
- 1.2. El modelo de *John Von Neumann*
- 1.3. Evolución histórica de las computadoras
- 1.4. Personajes más influyentes en la computación
- 1.5. Paradigmas de la computación
- 1.6. Paradigmas de la programación
- 1.7. Estructura y función de una computadora
- 1.8. Rendimiento de una computadora
- 1.9. Ley de *Moore* y tendencias históricas del *hardware*
- 1.10. Ley de *Amdahl* y métrica *Speedup*
- 1.11. Factores modernos de rendimiento: pipelining, predicción de saltos, ejecución especulativa, superscalar y análisis de flujo de datos

Duración:

6 horas

Unidad II. Componentes de una computadora

Competencia

Analizar la organización interna de una computadora, incluyendo memoria, dispositivos de entrada/salida, interconexión, sistema operativo y conjunto de instrucciones, mediante el estudio de sus características, funciones y métodos de direccionamiento, con el propósito de comprender su funcionamiento integral y la interacción entre *hardware* y *software*, actuando con responsabilidad, pensamiento crítico y disposición al aprendizaje continuo.

Contenido

- 2.1. Componentes de una computadora
- 2.2. Unidad Central de Procesamiento (CPU)
- 2.2. Memoria (registros, caché, RAM, memoria secundaria)
- 2.3. Dispositivos de entrada y salida
- 2.4. Interconexión interna (buses, puertos y controladores)
- 2.5. El sistema operativo
- 2.6. Conjunto de instrucciones y direccionamiento

- 2.7. Registros del procesador y organización interna
- 2.8. Buses de datos, líneas de dirección y control
- 2.9. Unidades de tasa de transferencia (bit/ byte) y cálculo de ancho de banda
- 2.10. Reloj y tiempos de CPU

Duración:

6 horas

Unidad III. Aritmética computacional

Competencia

Aplicar los fundamentos de la aritmética computacional y los sistemas numéricos, mediante la representación de números, signos y operaciones aritméticas y lógicas en diferentes bases, utilizando métodos de conversión, complementos y detección de errores, con el propósito de comprender el procesamiento de datos en el nivel máquina, actuando con precisión, responsabilidad, pensamiento crítico y disposición al aprendizaje continuo.

Contenido

- 3.1. Fundamentos de aritmética computacional
- 3.2. Sistemas numéricos
 - 3.2.1. Decimal
 - 3.2.2. Binario
 - 3.2.3. Octal
 - 3.2.4. Hexadecimal
 - 3.2.5. Conversión de sistemas numéricos
- 3.3. Representación de números y signos
 - 3.3.1. Signo y magnitud
 - 3.3.2. Complemento a uno
 - 3.3.3. Complemento a dos
 - 3.3.4. Intervalos y rangos de representación
 - 3.3.5. Detección de desbordamiento (overflow)
- 3.4. Operaciones aritméticas en binario
 - 3.4.1. Suma binaria
 - 3.4.2. Resta binaria
 - 3.4.3. Multiplicación binaria
 - 3.4.4. División binaria
- 3.5. Operaciones lógicas y desplazamientos
- 3.6. Casos prácticos avanzados
 - 3.6.1. Técnica de numerado negativo en multiplicación con signo y verificación de overflow

Duración:

6 horas

Unidad IV. Lenguaje ensamblador y código máquina**Competencia**

Aplicar los fundamentos del lenguaje ensamblador MIPS utilizando el simulador en línea *WebSIM*, un entorno innovador para el aprendizaje, que permita comprender el código máquina subyacente, dominar la conversión bidireccional entre lenguaje ensamblador y código máquina, analizar el conjunto de instrucciones y las estructuras de control, así como desarrollar programas prácticos mediante un enfoque crítico y propositivo.

Contenido

- 4.1. Definición y diferencias entre lenguaje de alto nivel, lenguaje ensamblador y código máquina
- 4.2. Introducción al lenguaje ensamblador
 - 4.2.1. Conjunto de instrucciones, sintaxis y estructura básica
 - 4.2.2. Estructuras de control
 - 4.2.3. Operaciones para transferencias de datos
 - 4.2.4. Operaciones aritméticas y lógicas
 - 4.2.5. Manejo de cadenas
 - 4.2.6. Uso de la pila y llamadas a subrutinas
- 4.3. Introducción al código máquina
 - 4.3.1. Conversión de lenguaje ensamblador a código máquina y viceversa
- 4.4. Simulador *WebSIM*
 - 4.4.1. Microcódigo
 - 4.4.2. Depuración paso a paso
 - 4.4.3. CPU stats
- 4.5. Aplicaciones en *WebSIM*
 - 4.5.1. Tabla de instrucciones MIPS
 - 4.5.2. Registro de ticks
 - 4.5.3. Ejercicios
- 4.6. Arquitecturas y familias de CPU
 - 4.6.1. CISC, RISC, ARM, RISC-V, MIPS

Duración:

8 horas

Unidad V. Circuitos lógicos**Competencia**

Diseñar circuitos lógicos combinacionales y secuenciales, mediante la aplicación del álgebra de Boole, compuertas lógicas, flip-flops y dispositivos como multiplexores, decodificadores, contadores y memorias, utilizando métodos de simplificación, tablas de verdad y diagramas de temporización, con el propósito de comprender y construir la base lógica del hardware

computacional, actuando con precisión, responsabilidad, pensamiento crítico y disposición al trabajo colaborativo.

Contenido

- 5.1. Álgebra de *Boole*
 - 5.1.1. Operadores básicos y derivados
 - 5.1.2. Propiedades y leyes del álgebra booleana
 - 5.1.3. Simplificación de funciones lógicas
- 5.2. Compuertas lógicas
 - 5.1.1. Definición y representación simbólica
 - 5.1.2. Compuertas básicas y universales
- 5.3. Herramientas de simulación de circuitos lógicos
- 5.4. Latch & Flip-Flops
 - 5.4.1. Latch SR
 - 5.4.2. Tipos de Flips-Flops
 - 5.4.3. Tablas de verdad, ecuaciones y diagramas de temporización
 - 5.4.4. Aplicaciones típicas en registros y contadores
- 5.5. Circuitos lógicos y secuenciales
 - 5.5.1. Combinacionales
 - 5.5.1.1. Decodificadores y codificadores
 - 5.5.1.2. Multiplexores y demultiplexores
 - 5.5.1.3. Comparadores
 - 5.5.1.4. Sumadores y restadores binarios
 - 5.5.2. Secuenciales
 - 5.5.2.1. Registros síncronos y asíncronos
 - 5.5.2.2. Contadores síncronos y asíncronos
 - 5.5.2.3. Máquinas de estados finitos (FSM)
 - 5.5.3. Memoria
 - 5.5.3.1. Memorias estáticas y dinámicas (SRAM y DRAM)
 - 5.5.3.2. Concepto de celda básica de memoria

Duración:

6 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO**Unidad I: Introducción****Práctica 1.1: Los componentes de una computadora*****Duración:***

4 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Elabora un mapa mental que integre los principales componentes de la computadora: CPU (unidad de control, ALU, registros y caché), memoria (jerarquía y tipos), almacenamiento (HDD, SSD, NVMe), buses e interconexión (datos, direcciones y control) y dispositivos de entrada/salida (teclado, pantalla, red).
3. Incluye en cada bloque una descripción breve de su función y, cuando sea pertinente, ejemplos de métricas técnicas como frecuencia, latencia o ancho de banda.
4. Señala las dependencias y relaciones entre los bloques, destacando cómo influyen en el rendimiento global del sistema (por ejemplo, la limitación que impone el bus sobre la memoria secundaria).
5. Realiza el arte final en una plataforma de diseño en línea y complementa el trabajo con un árbol descriptivo que muestre las definiciones y características esenciales de cada componente.
6. Entrega un documento en formato *.pdf*, con una extensión mínima de dos cuartillas con el resumen y conclusión del documental y las respuestas a las preguntas, utilizando formato APA de las fuentes consultadas.

Recursos de apoyo

- Software para mapas mentales (Mindmeister, Canva, etc.)

Práctica 1.2: Rendimiento de una computadora

Duración:

4 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Selecciona cuatro equipos con diferentes configuraciones (por ejemplo: teléfono celular, laptop, PC de escritorio y dispositivo embebido) y documenta sus especificaciones técnicas principales (CPU, memoria, GPU, sistema operativo, etc.).
3. Ejecuta una aplicación de benchmark sintético (p. ej., *Geekbench 6*, *Basemark Web 3.0*) en cada equipo y registra los resultados de Single-Core y Multi-Core.
4. Corre Basemark Web 3.0 en un navegador compatible, anota el puntaje global y describe brevemente qué mide y qué componentes acelera (CPU y/o GPU).
5. Presenta los resultados en gráficas comparativas de barras que muestren el rendimiento *Single-Core vs. Multi-Core* y los puntajes Web; interpreta las diferencias observadas e identifica posibles cuellos de botella.
6. Redacta conclusiones técnicas, destacando fortalezas, debilidades y aprendizajes sobre el rendimiento relativo de los sistemas evaluados.
7. Entrega un documento en formato *.pdf*, con una extensión mínima de dos cuartillas, utilizando formato APA de las fuentes consultadas.

Recursos de apoyo

- Software para análisis de rendimiento de una computadora (Geekbench, Basemark, etc.)

Unidad II: Componentes de una computadora

Práctica 2.1: nombre de la práctica

Duración:

6 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Selecciona cinco buses o interfaces de comunicación (por ejemplo: *PCIe*, *USB*, *SPI*, *I²C*, *SATA*).

3. Documenta para cada uno sus principales características: tipo de conector, capa/protocolo de comunicación, versiones existentes y pinout, incluyendo al menos una imagen de referencia.
4. Calcula la tasa máxima de transferencia para cada bus, normalizando las unidades en bit/s y B/s; especifica los supuestos considerados (ancho de canal, codificación, *overhead*, entre otros).
5. Elabora una gráfica comparativa de las tasas de transferencia y analiza las implicaciones prácticas, señalando posibles cuellos de botella en aplicaciones como almacenamiento masivo, interconexión de GPU o periféricos de alta velocidad.
6. Entrega un documento en formato *.pdf*, con una extensión mínima de dos cuartillas, utilizando formato APA de las fuentes consultadas.

Recursos de apoyo

- Documentación oficial de estándares como PCI-SIG (PCIe), USB-IF (USB) y SATA-IO.
- Videos explicativos en YouTube sobre PCIe, USB, SATA, I²C y SPI.
- Documentos técnicos de fabricantes como Intel, AMD, ARM o Texas Instruments que describen arquitecturas de buses.
- Tutoriales interactivos en sitios como GeeksforGeeks, Tutorialspoint o Khan Academy (Computing).

Unidad III: Aritmética computacional

Práctica 3.1: Notación posicional y conversión de bases

Duración:

6 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. En MATLAB implementa funciones para convertir entre bases 2, 8, 10 y 16.
3. Agrega soporte para números con signo en complemento a 2 (8 y 16 bits).
4. Verifica tu implementación con al menos 10 casos de prueba mixtos (positivos/negativos y fracciones si las consideras).
5. Redacta un informe: (a) explicación del sistema posicional; (b) fórmulas y ejemplos de conversión; (c) tabla con resultados de tus casos; (d) conclusiones y consideraciones sobre errores comunes.
6. Entrega un documento en formato *.pdf*, con una extensión mínima de dos cuartillas, utilizando formato APA de las fuentes consultadas.

Recursos de apoyo

- Software Matlab

Unidad IV: Lenguaje ensamblador y código máquina

Práctica 4.1: Serie de ejercicios de lenguaje ensamblador MIPS y código máquina

Duración:

6 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Elaborará una serie de prácticas graduales comenzando con conceptos básicos como la identificación de los 32 registros (\$0-\$31) y operaciones aritméticas simples (add, sub, mul), progresando luego a instrucciones de transferencia de datos (lw, sw) para manipular memoria, seguido por estructuras de control con saltos condicionales e incondicionales (beq, bne, j) para implementar bucles y decisiones. Los ejercicios iniciales incluirán traducciones directas de expresiones matemáticas simples a MIPS, después convertir pequeños fragmentos de código C al ensamblador, implementar algoritmos básicos como búsqueda y ordenamiento, y finalmente escribir programas completos con manejo de arreglos, cadenas y funciones. En paralelo, aprenderá código máquina analizando cómo cada instrucción MIPS se codifica en sus 32 bits correspondientes según los formatos *R-type*, *I-type* y *J-type*, realizando ejercicios de codificación manual de instrucciones y decodificación de valores hexadecimales para entender la representación binaria. La práctica se refuerza utilizando simuladores (e.g. *WepSIM*, *MARS*,...) que permiten escribir, ensamblar, ejecutar y depurar programas paso a paso, visualizando cómo cambian los registros y la memoria con cada instrucción, consolidando así la comprensión tanto del ensamblador como de su representación en código máquina.
3. Entregará un documento en .pdf que incluya el código en lenguaje ensamblador y las pantallas de captura de las corridas en el simulador.

Recursos de apoyo

- Introduction to MIPS Assembly Language Programming: <https://open.umn.edu/opentextbooks/textbooks/497>
- The Web Elemental Processor SIMulator <https://wepsim.github.io/>
- Libro completo sobre programación en MIPS disponible en formatos HTML, PDF y EPUB, con código fuente incluido https://github.com/rswinkle/mips_book

Unidad V: Circuitos lógicos

Práctica 5.1: Serie de ejercicios utilizando simuladores de circuitos lógicos

Duración:

6 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica
2. Elaborará una serie de prácticas graduales utilizando simuladores de circuitos (e.g. Logisim, Logisim evolution, NI Multisim, ...), comenzando con ejercicios graduales que inician con la construcción de compuertas básicas (AND, OR, NOT) y verificación de tablas de verdad, progresando a circuitos combinacionales simples como sumadores de 1 bit (half-adder y full-adder), luego expandiendo a sumadores de múltiples bits (4-bit, 8-bit) y restadores binarios usando complemento a dos. Continuará con

comparadores de magnitud de 4 y 8 bits, seguido por elementos de memoria empezando con latches SR básicos, flip-flops D y JK con análisis de diagramas de temporización, registros de desplazamiento de 4 bits, contadores síncronos y asíncronos (binarios, BCD, módulo-N), máquinas de estados finitos simples como detectores de secuencias (001, 101), memorias RAM básicas de 4x4 bits, y culminando con proyectos integradores como una ALU simple que realice suma, resta, AND, OR con selección de operación, y un procesador básico que ejecute instrucciones simples, permitiendo al estudiante visualizar en tiempo real el comportamiento de señales, cronogramas, y la evolución de estados mientras construye gradualmente desde componentes elementales hasta sistemas digitales complejos que forman la base del hardware computacional.

3. Entregará un documento en formato .pdf, los circuitos y las impresiones de pantalla de la salida del simulador de circuitos.

Recursos de apoyo

- Electrónica Digital: <https://angelmicelti.github.io/4ESO/EDI/simuladorio.html>
- Apuntes de clase
- Prácticas guiadas entregadas por su docente

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Estudio de caso
- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Aprendizaje basado en preguntas
- Técnica expositiva
- Debates
- Discusión guiada
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Mapas mentales

- Mapas conceptuales
- Cuadros comparativos
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

- Evaluaciones parciales	30%
- Prácticas de laboratorio	40%
- Proyecto final	30%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Abd-El-Barr, Mostafa; El-Rewini, Hesham. (2004). *Fundamentals of Computer Organization and Architecture*. Wiley-Interscience. [Clásica]

Null, Linda; Lobur, Julia. (2018). *Essentials of Computer Organization and Architecture*. Jones & Bartlett (5th ed.). [Clásica]

Stallings, William. (2005). *Organización y arquitectura de computadores*. Pearson Education. (7^a ed.). [Clásica]

Stallings, William. (2022). *Computer Organization & Architecture*. Pearson, (11th ed.).

Stallings, William. (2022). *Organización y arquitectura de computadores: diseñar para mejorar el rendimiento*. Pearson Hispanoamérica.
<https://uabc.vitalsource.com/reader/books/9786073260244>

Recursos de acceso libre o complementarios externos

Jim Ledin, & Dave Farley. (2022). *Modern Computer Architecture and Organization: Learn X86, ARM, and RISC-V Architectures and the Design of Smartphones, PCs, and Cloud Servers*. Packt Publishing. [eBook].

Patterson, D. A., Hennessy, J. L., & Ashenden, P. J. (2007). *Computer Organization and Design: The Hardware/Software Interface*. Morgan Kaufmann. [eBook]

Tanenbaum, Andrew S.; Austin, Todd. (2013). *Structured Computer Organization*. Prentice Hall.

XII. PERFIL DOCENTE

Sólida formación académica en áreas de Ingeniería en Computación, Ciencias Computacionales, Ingeniería en Electrónica o afines, preferentemente con grado de maestría o doctorado. Conocimientos sobre arquitectura de procesadores, sistemas de memoria, bases de datos, entrada/salida, sistemas embebidos y diseño digital, así como experiencia en el uso de emuladores, simuladores y herramientas de modelado de sistemas computacionales. Asimismo, es deseable que cuente con dominio de lenguajes de bajo nivel (ensamblador) y comprensión de la interacción *hardware–software*.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Física Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Ecuaciones Diferenciales Ordinarias
Clave:	
Distribución horaria:	Horas Clase: 03 Horas Taller: 03 Horas Laboratorio: 00 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 03 Créditos:09
Etapas de formación:	Disciplinaria
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Juan Crisóstomo Tapia Mercado José Ariel Camacho Gutiérrez Jonathan Verdugo Olachea
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Ecuaciones Diferenciales permite al estudiantado ser competentes en la clasificación, resolución y análisis de la validez y comportamiento de las soluciones que calculen, se pretende que las y los estudiantes utilicen los saberes y las habilidades logradas previamente en Cálculo Diferencial, Álgebra Lineal y Mecánica, entre otras. Pertenece al plan de estudios de Licenciatura en Física, se imparte en la etapa básica, es de carácter obligatoria y pertenece al área de conocimiento de Matemáticas; además se comparte con la Licenciatura en Ciencias Computacionales, se ubica en etapa disciplinaria con carácter obligatorio.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Valorar la elección de los métodos de solución de ecuaciones diferenciales ordinarias más adecuados, mediante el reconocimiento de las características de las ecuaciones y de los métodos de solución, para encontrar y analizar soluciones cuantitativa y cualitativas, así como establecer sus regiones de validez en problemas relacionados con las ciencias naturales y exactas, con disposición al trabajo en equipo, actitud analítica, crítica y responsable.

IV. EVIDENCIA DE APRENDIZAJE

Carpeta de evidencias que contenga: Resolución de problemas y ejercicios con el desglose detallado del proceso y los resultados del manejo de los métodos de solución, con sus objetivos y sus limitantes, de los diferentes tipos de ecuaciones diferenciales, así como los conceptos fundamentales involucrados en el análisis cuantitativo y cualitativo, mostrando la capacidad para aplicar los métodos matemáticos apropiados a situaciones reales, la habilidad para utilizar procedimientos deductivos cumpliendo con los teoremas matemáticos, y la obtención de la solución correcta del problema, mientras colabora proactivamente con sus compañeros.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Licenciatura en Física

Explorar fenómenos físicos mediante sus modelos matemáticos, para describir y predecir el comportamiento de la materia, la energía y sus interacciones, con disciplina y rigor científico.

Licenciatura en Ciencias Computacionales

Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que

contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia, Inclusión

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos de las ecuaciones diferenciales ordinarias

Competencia

Aplicar los conceptos fundamentales asociados a una ecuación diferencial ordinaria, mediante un análisis crítico de definiciones y de ejemplos, para identificar las características de distintos tipos de ecuaciones diferenciales ordinarias así como las características de soluciones, con empatía, persistencia y responsabilidad.

Contenido

- 1.1. Conceptos y definiciones básicas
 - 1.1.1. Ecuación diferencial (ED), ED ordinaria (EDO), ED parcial, orden de EDs
 - 1.1.2. Estructura de EDOs lineales (homogéneas y no homogéneas, coeficientes constantes y no constantes)
 - 1.1.3. Solución, intervalo de solución, solución general
 - 1.1.4. Problemas de valor inicial, solución particular
- 1.2. Modelos matemáticos y ecuaciones diferenciales
 - 1.2.1. Caída libre, segunda ley de Newton
 - 1.2.2. Otros modelos clásicos
- 1.3. Ecuaciones diferenciales de integración directa
 - 1.3.1. Soluciones como integrales indefinidas
 - 1.3.2. Soluciones como integrales definidas

Duración

09 horas

Unidad II. Ecuaciones de primer orden

Competencia

Aplicar los métodos de solución clásicos en distintos tipos de ecuaciones diferenciales ordinarias de primer orden, mediante el análisis de la estructura de las ecuaciones y de la construcción de los métodos de solución, para encontrar soluciones y contrastarlas con los resultados del análisis cualitativo, con actitud reflexiva y disposición al trabajo colaborativo.

Contenido

- 2.1. Conceptos y resultados
 - 2.1.1. Forma normal $dy/dx = f(x,y)$
 - 2.1.2. Soluciones constantes
 - 2.1.3. Teorema clásico de existencia y unicidad de soluciones para problemas con valores iniciales
- 2.2. EDOs separables de primer orden
 - 2.2.1. División entre cero y completación de familia de soluciones
 - 2.2.2. Soluciones explícitas y soluciones implícitas
- 2.3. EDOs lineales de primer orden
 - 2.3.1. Derivada del producto y factor de integración
 - 2.3.2. Método para hallar solución general
 - 2.3.3. Soluciones con integrales definidas, función erf
 - 2.3.4. Teorema de existencia y unicidad de soluciones para problemas con valores iniciales
- 2.4. Cambios de variable
 - 2.4.1. Idea general
 - 2.4.2. Cambios de variable lineales $Ax+By+C$
 - 2.4.3. Ecuaciones homogéneas $y' = f(y/x)$
 - 2.4.4. Ecuaciones de Bernoulli
- 2.5. Ecuaciones exactas
 - 2.5.1. Regla de la cadena para funciones $f(x, y(x))$
 - 2.5.2. Definición de ecuación exacta y función potencial
 - 2.5.3. Método de solución, familia de soluciones implícitas
- 2.6. Campos de direcciones
 - 2.6.1. Método gráfico, e isoclinas (construcción de campos de dirección)
 - 2.6.2. Comportamiento asintótico en el campo de direcciones
 - 2.6.3. Soluciones de equilibrio y estabilidad local
 - 2.6.4. Bifurcaciones: cambio de estabilidad debido a un parámetro
 - 2.6.5. Relación con el teorema clásico de existencia y unicidad $dy/dx=f(x,y)$
 - 2.6.6. Ecuaciones autónomas
 - 2.6.7. Familias de curvas ortogonales

Duración

12 horas

Unidad III. Ecuaciones de orden superior

Competencia

Aplicar los métodos de solución clásicos para ecuaciones diferenciales ordinarias lineales de segundo orden homogéneas y no homogéneas, mediante el análisis de las raíces del

polinomio característico, para encontrar soluciones generales y particulares, con actitud reflexiva y disposición al trabajo colaborativo.

Contenido

- 3.1. Introducción
 - 3.1.1. Problemas con valores iniciales
 - 3.1.1.1. Condiciones iniciales en el plano
 - 3.1.1.2. Teorema de existencia y unicidad de soluciones
 - 3.1.2. Problemas con valores en la frontera
- 3.2. Principio de superposición, caso homogéneo y no homogéneo
- 3.3. Ecuaciones homogéneas
 - 3.3.1. Soluciones linealmente independientes
 - 3.3.2. Wronskiano y sus propiedades
 - 3.3.3. Conjunto fundamental de soluciones y sus propiedades
 - 3.3.4. Método de reducción de orden
 - 3.3.5. Ecuaciones con coeficientes constantes
 - 3.3.5.1. Ansatz: solución exponencial
 - 3.3.5.2. Polinomio característico
 - 3.3.5.3. Análisis de raíces, solución general
- 3.4. Ecuaciones no homogéneas
 - 3.4.1. Solución complementaria y solución particular
 - 3.4.2. Método de coeficientes indeterminados
 - 3.4.3. Método de variación de parámetros
- 3.5. Sistema masa - resorte - amortiguador
 - 3.5.1. Planteamiento del modelo matemático
 - 3.5.2. Análisis de las soluciones
- 3.6. Extensión de la teoría de ecuaciones lineales de segundo orden a las de orden superior

Duración

09 horas

Unidad IV. Soluciones en series

Competencia

Aplicar el método de series de potencias y el método de Frobenius, mediante el uso adecuado de sus propiedades, para encontrar soluciones de ecuaciones diferenciales lineales de segundo orden en la vecindad de puntos regulares y puntos singulares removibles, con actitud reflexiva y disposición al trabajo colaborativo.

Contenido

- 4.1. Solución en series de Taylor, de ecuaciones lineales en torno a puntos ordinarios.

- 4.1.1. Aritmética de series de potencias
- 4.1.2. Radio de convergencia
- 4.1.3. Definición de puntos ordinarios
- 4.1.4. Teorema de existencia de soluciones en series de potencias de ecuaciones diferenciales lineales de segundo orden alrededor de puntos ordinarios
- 4.2. Soluciones de ecuaciones lineales en torno a puntos singulares removibles: Método de Frobenius
 - 4.2.1. Definición de puntos puntos singulares regulares.
 - 4.2.2. La función Gamma.
 - 4.2.3. Solución de ecuaciones diferenciales clásicas como la ecuación diferencial de Bessel y la Ecuación diferencial de Legendre.

Duración

09 horas

Unidad V. Uso de transformada de Laplace en la solución de ecuaciones lineales

Competencia

Aplicar la transformada de Laplace y su inversa, mediante el uso adecuado de sus propiedades, para encontrar soluciones de ecuaciones diferenciales lineales de coeficientes constantes, con actitud reflexiva y disposición al trabajo colaborativo.

Contenido

- 5.1. Definición de transformada de Laplace
- 5.2. Linealidad de la transformada de Laplace
- 5.3. Transformada de Laplace de distintas funciones
 - 5.3.1. Monomios
 - 5.3.2. Exponenciales
 - 5.3.3. Seno y coseno
- 5.4. Transformada inversa de Laplace
 - 5.4.1. Linealidad
 - 5.4.2. Fracciones parciales
- 5.5. Transformada de Laplace de derivadas
- 5.6. Solución de ecuaciones lineales mediante transformada de Laplace
- 5.7. Introducción a sistemas lineales de ecuaciones diferenciales
 - 5.7.1. Transformación de una EDO de orden n , a un sistema de primer orden
 - 5.7.2. Solución de sistemas mediante transformada de Laplace
 - 5.7.3. Plano fase

Duración

09 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

Unidad I. Introducción

Práctica 1: Reconocer y clasificar ecuaciones diferenciales

Duración

03 horas

Procedimiento

1. Atiende la orientación docente para la práctica.
2. Revisa en tus notas las definiciones de ED, EDO, EDP, orden, linealidad, homogeneidad.
3. Analiza una lista de ecuaciones dadas y clasifícalas según su tipo.
4. Determina orden, linealidad y homogeneidad.
5. Construye una tabla comparativa.
6. Discute ejemplos límite.
7. Elabora un informe final.

Recursos de apoyo

- Notas.
- Pizarrón.
- Texto de EDOs.
- Calculadora.
- Software simbólico opcional (Python/SymPy o WolframAlpha).

Práctica 2: Modelos clásicos: caída libre y ley de Newton

Duración

03 horas

Procedimiento

1. Revisa en tus notas la forma general de modelos físicos simples.
2. En equipo, formula la ecuación diferencial de caída libre con y sin resistencia proporcional a la velocidad.
3. Identifica las variables dependientes y parámetros físicos.
4. Determina las unidades de cada término para verificar consistencia dimensional.
5. Plantea el problema de valor inicial apropiado.
6. Escribe la solución formal como integral definida, aunque no la resuelvas todavía.
7. Dibuja el campo de direcciones para el modelo con resistencia y predice el comportamiento asintótico.

8. Escribe un informe con el modelo, su interpretación y la predicción cualitativa.

Recursos de apoyo

- Notas.
- Texto.
- Software de campos de direcciones (GeoGebra, Python).

Práctica 3: Soluciones como integrales definidas

Duración

03 horas

Procedimiento

1. Revisa la idea de "solución como integral indefinida" versus "solución como integral definida".
2. Analiza ecuaciones del tipo $y'(x)=f(x)$, $y(x_0)=y_0$.
3. Escribe la solución general como integral indefinida.
4. Escribe la solución particular como integral definida desde x_0 .
5. Resuelve 4 ejemplos proporcionados por el docente.
6. Compara ambas expresiones y discute ventajas de la forma definida.
7. En software (opcional), grafica la función integrando numéricamente.
8. Elabora un informe con ejemplos y reflexiones.

Recursos de apoyo

- Notas.
- Pizarrón.
- Software (Python o GeoGebra).

Unidad II. Ecuaciones de primer orden

Práctica 4: Ecuaciones separables

Duración

03 horas

Procedimiento

1. Revisa en tus notas la forma separable.
2. Analiza una lista de 10 ecuaciones y clasifica cuáles son separables.
3. Para cada ecuación separable:
 - Escribe explícitamente la separación.
 - Identifica posibles valores prohibidos (división entre cero).
 - Resuelve la ecuación explícitamente.
4. Construye el campo de direcciones de dos de ellas.
5. Usa el campo para identificar soluciones constantes, creciendo o decreciendo.
6. Elabora un informe con soluciones, restricciones y observaciones cualitativas.

Recursos de apoyo

- Notas.
- Software para campos de direcciones.

Práctica 5: Ecuaciones lineales de primer orden y factor integrante

Duración

03 horas

Procedimiento

1. Revisa la forma estándar $y' + P(x)y = Q(x)$.
2. Para 4 ecuaciones dadas:
 - Identifica $P(x)$ y $Q(x)$.
 - Calcula el factor integrante $\mu(x) = e^{\int P(x) dx}$.
 - Aplica el método para obtener la solución general.
3. Resuelve una ecuación donde la integral involucra erf.
4. Verifica una solución sustituyéndola de regreso en la ecuación.
5. Discute ventajas del método.
6. Escribe un informe con el procedimiento detallado.

Recursos de apoyo

- Notas.
- Texto.
- Pizarrón.
- Software simbólico opcional.

Práctica 6: Cambios de variable de primer orden

Duración

03 horas

Procedimiento

1. Revisa ejemplos de cambios lineales $u = Ax + By + C$.
2. Aplica un cambio de variable en:
 - una ecuación homogénea $y' = f(y/x)$
 - una ecuación de Bernoulli
 - una ecuación reducible a lineal
3. En cada caso, muestra la reducción a una forma conocida.
4. Resuelve la ecuación transformada.
5. Interpreta y dibuja la transformación geométrica del cambio lineal.
6. Elabora un informe con el procedimiento y comentarios.

Recursos de apoyo

- Notas.

- Software para gráficas.

Práctica 7: Ecuaciones exactas y función potencial

Duración

03 horas

Procedimiento

1. Revisa la forma $M(x,y) dx + N(x,y) dy = 0$.
2. Verifica la condición de exactitud $\partial M / \partial y = \partial N / \partial x$.
3. Para 3 ecuaciones exactas:
 - Calcula la función potencial $\psi(x,y)$.
 - Obtén la familia de soluciones $\psi = C$.
4. Para 2 ecuaciones no exactas: determina si existe un factor integrante sencillo.
5. Representa gráficamente las curvas de nivel de ψ .
6. Elabora un informe con los procedimientos.

Recursos de apoyo

- Notas.
- Software para curvas de nivel.

Práctica 8: Campos de direcciones, estabilidad y bifurcaciones

Duración

03 horas

Procedimiento

1. Revisa en tus notas el concepto de ecuación autónoma $y' = f(y)$.
2. Analiza 3 funciones $f(y)$ y determina sus puntos de equilibrio.
3. Determina estabilidad usando la derivada $f'(y^*)$.
4. Construye el diagrama de fases (lineal).
5. Para un parámetro r , estudia cómo cambia la estabilidad (bifurcación).
6. Simula numéricamente ejemplos si es posible.
7. Elabora un informe con diagramas, estabilidad y bifurcaciones observadas.

Recursos de apoyo

- Notas.
- Software (GeoGebra, Python).

Unidad III: Ecuaciones de orden superior, forma general, ecuación orden n

Práctica 9: Soluciones linealmente independientes y Wronskiano

Duración

03 horas

Procedimiento

1. Revisa definiciones de independencia y Wronskiano.
2. Analiza pares de funciones y_1, y_2 dadas por el docente.
3. Calcula el Wronskiano y determina si son LI.
4. Verifica la solución en la ecuación dada (opcional).
5. Discute ejemplos donde el Wronskiano es cero en un punto, pero no en todos.
6. Elabora un informe con el cálculo y conclusiones.

Recursos de apoyo

- Notas.
- Texto.
- Software simbólico.

Práctica 10: EDO lineales con coeficientes constantes

Duración

03 horas

Procedimiento

1. Revisa el método: ansatz exponencial y polinomio característico.
2. Resuelve ejemplos con:
 - raíces reales y distintas,
 - raíces repetidas,
 - raíces complejas.
3. Construye la solución general en cada caso.
4. Resuelve un problema de valores iniciales.
5. Representa gráficamente cada solución.
6. Elabora un informe.

Recursos de apoyo

- Notas.
- Software (Python/GeoGebra).

Práctica 11: Métodos para ecuaciones no homogéneas

Duración

03 horas

Procedimiento

1. Revisa métodos: coeficientes indeterminados y variación de parámetros.
2. Para 3 ecuaciones con forzamiento (polinomial, exponencial, trigonométrico):
 - Aplica coeficientes indeterminados.
 - Verifica si hay resonancia.
3. Usa variación de parámetros en un caso con forzamiento no estándar.
4. Presenta soluciones generales y particulares.
5. Escribe un informe comparativo entre métodos.

Recursos de apoyo

- Notas.
- Texto.

Práctica 12: Sistema masa–resorte–amortiguador

Duración

03 horas

Procedimiento

1. Revisa la ecuación estándar $my'' + cy' + ky = F(t)$
2. Identifica los parámetros físicos y sus unidades.
3. Analiza 3 casos:
 - Subamortiguado
 - Amortiguamiento crítico
 - Sobreamortiguado.
4. Caracteriza la frecuencia natural y la frecuencia amortiguada.
5. Construye esquemas cualitativos de la solución.
6. En software, simula al menos un caso.
7. Elabora un informe con los resultados.

Recursos de apoyo

- Notas.
- Software de simulación.

Unidad IV: Soluciones en series

Práctica 13: Aritmética de series de potencias y radio de convergencia

Duración

03 horas

Procedimiento

1. Revisa operaciones básicas con series: suma, producto, derivación.
2. Para 4 series dadas:
 - Calcula el radio de convergencia.

- Determina si representan funciones conocidas.
3. Representa las series hasta los primeros 5 términos.
 4. Elabora un informe.

Recursos de apoyo

- Notas.
- Hojas de cálculo o Python.

Práctica 14: Soluciones en una vecindad de un punto ordinario

Duración

03 horas

Procedimiento

1. Revisa el método de serie de Taylor.
2. Para una ecuación como $y''+xy'+y=0$, desarrolla la solución como serie alrededor de $x=0$.
3. Calcula al menos 5 coeficientes.
4. Discute la validez y el radio de convergencia.
5. Elabora un informe con desarrollo y conclusiones.

Recursos de apoyo

- Notas.
- Software para cálculos.

Práctica 15: Método de Frobenius en un punto singular regular

Duración

03 horas

Procedimiento

1. Revisa la forma del método de Frobenius.
2. Aplica el método a una ecuación del tipo Bessel o similar.
3. Calcula la ecuación indicial.
4. Obtén la primera solución.
5. Discute el caso de raíces separadas o iguales.
6. Elabora un informe.

Recursos de apoyo

- Notas.
- Texto.
- Software simbólico.

Unidad V: Uso de transformada de Laplace en la solución de ecuaciones lineales

Práctica 16: Uso de la transformada de Laplace para resolver EDO

Duración

03 horas

Procedimiento

1. Revisa definiciones y propiedades básicas.
2. Para 3 ecuaciones lineales con condiciones iniciales:
 - Toma transformada.
 - Resuelve para $Y(s)$.
 - Usa fracciones parciales si aplica.
 - Toma la transformada inversa.
3. Comprueba sustituyendo en la ecuación original.
4. En un caso, compara con la solución obtenida por métodos anteriores.
5. Elabora un informe.

Recursos de apoyo

- Notas.
- Pizarrón.
- Calculadora o software.

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Método de proyectos
- Aprendizaje basado en problemas
- Técnica expositiva
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

- Evaluaciones parciales	50%
- Prácticas semanales	20%
- Tareas	5%
- Participación en clase	5%
- Examen final	20%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Boyce, W. E., DiPrima, R. C., & Haines, C. W. (2001). *Elementary differential equations and boundary value problems* (Vol. 9). Wiley. [clásica]
- Cengel, Y. A., Palm, W. J. (2023). *Ecuaciones diferenciales para ingeniería y ciencias*. McGraw Hill.
- Nagle, R. K, Saff, E. B, Snider, A. D. (2023). *Ecuaciones diferenciales con problemas de valores en la frontera*. Pearson.
- Rainville, E. D. (1998). *Ecuaciones diferenciales*. Pearson Education. [clásica]

- Simmons, G. F. (1993). *Ecuaciones diferenciales con aplicaciones y notas históricas*. McGraw Hill. [clásica]
- Simmons, G. F. (2007). *Ecuaciones diferenciales : teoría, técnica y práctica*. McGraw Hill. [clásica]
- Trench, W.F. (2001). *Ecuaciones diferenciales con problemas de valores en la frontera*. International Thompson. [clásica]
- Zill, D. G. (2018). *Ecuaciones diferenciales con problemas de valores en la frontera*. Cengage. [clásica]
- Zill, D. G. (2021). *Ecuaciones diferenciales aplicadas*. Cengage.
- Zill, D. G., Warren, S. W. (2024). *Fundamentos de ecuaciones diferenciales*. Cengage.

Recursos de acceso libre o complementarios externos

- Amritasu, S. (2013). *Applied differential equations*. Alpha Science International.
- Doshi, J. B. (2010). *Differential equations for scientists and engineers*. Alpha Science International

XII. PERFIL DOCENTE

Licenciatura en Física, Matemáticas o área afín, con conocimientos de los métodos de solución de ecuaciones diferenciales ordinarias más adecuados, utilizando procedimientos analíticos y numéricos, para aplicarlas en la solución de problemas de las ciencias naturales y exactas, preferentemente con estudios de posgrado y experiencia docente mínima de un año. Debe ser una persona analítica, respetuosa, responsable y proactiva, debe contar con facilidad de expresión y debe fomentar la participación individual y en equipo.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Bases de Datos
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 03 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Disciplinaria
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	María Victoria Meza Kubo Roberto Adolfo Romero Hernández
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Proporcionar al estudiantado los conocimientos, técnicas y herramientas necesarias para analizar, modelar, diseñar, implementar y administrar sistemas de bases de datos, a través de modelos conceptuales y lógicos, lenguajes de consulta, manejadores modernos y arquitecturas de software.

Esta unidad de aprendizaje se ubica en la etapa disciplinaria, es de carácter obligatorio, forma parte del área de conocimiento de Tratamiento de Información y es la base para la construcción de aplicaciones que involucren el almacenamiento de los datos.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Desarrollar sistemas de base de datos mediante el uso de técnicas de modelado, normalización, lenguajes de manipulación de datos, administración de manejadores de bases de datos y su integración con aplicaciones, para satisfacer necesidades reales de almacenamiento y gestión de información de las organizaciones, con responsabilidad y pensamiento crítico.

IV. EVIDENCIA DE APRENDIZAJE

Desarrolla un sistema de base de datos, aplicando estándares de modelado y lenguajes de programación de actualidad, para dar solución a problemas reales de manejo de información, la cual deberá contener: a) un modelo conceptual y lógico, b) la implementación física, c) consultas SQL básicas y avanzadas, d) mecanismos de integridad, seguridad y administración, e) una aplicación que lo integre y f) un plan de pruebas con la documentación formal del proceso.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Inclusión y Excelencia.

VII. DESARROLLO POR UNIDADES

Unidad I. Introducción a las Bases de Datos

Competencia

Analizar los elementos que intervienen en el ambiente de bases de datos, a través de la definición de sus principales elementos y características para comprender los contextos donde se requieren soluciones de almacenamiento de información, con actitud crítica y analítica.

Contenido

- 1.1. Conceptos básicos de bases de datos
- 1.2. Datos, información y sistemas de información
- 1.3. Componentes de un sistema de base de datos
- 1.4 Tipos y formatos de datos
- 1.5. Bases de datos tradicionales vs. bases de datos modernas
- 1.6. Retos actuales: volumen, variedad, velocidad, veracidad (Big Data)
- 1.7. Sistemas manejadores de bases de datos (MySQL, PostgreSQL, SQL Server, Oracle)

Duración

04 horas

Unidad II. Modelos y paradigmas de base de datos

Competencia

Comparar distintos modelos de datos tradicionales y modernos, mediante el análisis de proyectos reales, con el fin de seleccionar el enfoque más adecuado para un problema real con actitud honesta, analítica y propositiva.

Contenido

- 2.1. Modelo Jerárquico
- 2.2. Modelo de redes
- 2.3. Modelo Relacional
- 2.4. Modelo entidad-relación ampliado
- 2.5. Modelos no relacionales (NoSQL)
- 2.6. Criterios de selección de modelos y manejadores de bases de datos

Duración

06 horas

Unidad III. Diseño y lenguajes de bases de datos relacionales

Competencia

Diseñar bases de datos relacionales mediante la utilización del modelado conceptual, normalización y manejo del lenguaje SQL, para garantizar consistencia, integridad y eficiencia en la manipulación de la información, con actitud analítica y responsable.

Contenido

- 3.1. Modelo conceptual y lógico
 - 3.1.1. Modelo relacional y extendido
 - 3.1.2. Transformación al modelo relacional
- 3.2. Diseño del modelo relacional
 - 3.2.1. Estructuras relacionales
 - 3.2.2. Claves primarias y foráneas
 - 3.2.3. Integridad referencial
- 3.3. Técnicas de normalización
 - 3.3.1. 1ª, 2ª y 3ª forma normal
 - 3.3.2. 4ª y 5ª forma normal
 - 3.3.3. Detección y corrección de anomalías
- 3.4. SQL moderno
 - 3.4.1. DDL: CREATE, ALTER, DROP
 - 3.4.2. DML: INSERT, UPDATE, DELETE, SELECT
 - 3.4.3. Consultas avanzadas
 - 3.4.4. Índices, restricciones, triggers y stored procedures
 - 3.4.5. Transacciones y ACID

Duración

10 horas

Unidad IV. Sistemas manejadores de bases de datos

Competencia

Administrar manejadores de bases de datos, mediante el análisis de sus características principales, para gestionar de forma eficiente el almacenamiento, seguridad, respaldo y recuperación de la información, con actitud analítica y responsable.

Contenido

- 4.1. Arquitectura general de un sistema manejadores de bases de datos

- 4.2. Tipos de Manejadores de Bases de Datos y criterios de selección
- 4.3. Administración básica de un Manejador de Bases de Datos (PostgreSQL o MySQL)
- 4.4. Optimización de índices y consultas

Duración

04 horas

Unidad V. Bases de Datos en Aplicaciones y Arquitecturas Modernas

Competencia

Implementar una base de datos dentro de un entorno de aplicación mediante lenguajes de programación y frameworks modernos que permitan la consulta, manipulación y exposición de datos, con actitud honesta, crítica y responsable.

Contenido

- 5.1. Lenguajes de programación y conectores (Python, JavaScript/Node.js)
- 5.2. ORM (Object-Relational Mapping): SQLAlchemy, Sequelize
- 5.3. APIs y servicios REST conectados a base de datos
- 5.4. Integración con aplicaciones web
- 5.5. Entornos modernos: contenedores con Docker
- 5.6. Introducción a bases de datos en la nube (RDS, Firestore, Mongo Atlas)

Duración

04 horas

Unidad VI. Implementación de un sistema de base de datos

Competencia

Desarrollar un sistema funcional de base de datos, integrando técnicas de modelado, SQL, y administración del Manejador, para aplicarlo a una solución real, con actitud analítica y creativa.

Contenido

- 6.1. Análisis y diseño de la aplicación
- 6.2. Implementación del diseño en un manejador específico
- 6.3. Desarrollo de consultas, vistas y procedimientos
- 6.4. Integración con una aplicación
- 6.5. Implementación de casos de prueba funcionales y de integridad
- 6.6. Documentación del sistema

Duración

04 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO**Unidad I. Introducción a las Bases de Datos****Práctica 1: Exploración del Entorno de Datos y Manejadores de Bases de Datos*****Duración:***

03 horas

Procedimiento:

1. Atiende a la explicación docente sobre sistemas de información y bases de datos.
2. Explora diferentes tipos de formatos de datos (CSV, JSON, TXT, XML).
3. Identifica sus características, ventajas y limitaciones.
4. Investiga qué sistemas manejadores de bases de datos existen y los clasifica.
5. Documenta hallazgos y conclusiones en un reporte.
6. Entrega en PDF en la plataforma
7. Recibe retroalimentación de su docente.

Recursos de apoyo:

- Computadora
- Internet
- Ejemplos de archivos
- Lecturas sugeridas

Práctica 2: Instalación y Configuración de un SGBD (MySQL o PostgreSQL)***Duración:***

03 horas

Procedimiento:

1. Atiende las indicaciones de su docente sobre el proceso de instalación.
2. Instala el Manejador de BD seleccionado (MySQL o PostgreSQL).
3. Configura el entorno inicial: usuario, contraseña y base de datos de prueba.
4. Verifica conexión mediante línea de comandos y herramienta gráfica (Workbench o PgAdmin).
5. Documenta el proceso con capturas de pantalla.
6. Entrega en PDF en la plataforma
7. Recibe retroalimentación de su docente.

Recursos de apoyo:

- Instaladores del Manejador
- Computadora
- Internet

Unidad II: Modelos y Paradigmas de Bases de Datos**Práctica 3: Consulta de Metadatos y Exploración del Manejador de BD****Duración:**

03 horas

Procedimiento:

1. Atiende las indicaciones de su docente.
2. Conecta al Manejador desde la herramienta gráfica.
3. Lista bases de datos, tablas y vistas existentes.
4. Examina las tablas internas de metadatos (information_schema).
5. Realiza consultas básicas a estas tablas (SHOW TABLES, DESCRIBE).
6. Documenta resultados en un breve informe.
7. Entrega en PDF en la plataforma
8. Recibe retroalimentación de su docente.

Recursos:

- Manejador instalado
- Herramienta gráfica
- Computadora

Práctica 4: Identificación de Requerimientos y Análisis de un Caso de Estudio**Duración:**

03 horas

Procedimiento:

1. Atiende la explicación docente sobre levantamiento de requerimientos.
2. Selecciona un caso real (pequeña empresa, biblioteca, ventas, escuela).
3. Identifica datos, actores y procesos relevantes.
4. Elabora una lista de requerimientos funcionales iniciales.
5. Entrega en PDF en la plataforma
6. Recibe retroalimentación de su docente.

Recursos:

- Plantillas de análisis
- Computadora

Unidad III: Diseño y lenguajes de Bases de Datos relacionales**Práctica 5: Diseño de un Modelo Entidad–Relación (ER)****Duración:**

03 horas

Procedimiento:

1. Atiende explicación de símbolos y notación ER.
2. Utiliza una herramienta de modelado (Draw.io, Diagrams.net, Lucidchart).
3. Modela el caso estudiado (Práctica 4).
4. Incluye entidades, atributos, relaciones y cardinalidades.
5. Exporta el diagrama a PDF.
6. Entrega en PDF en la plataforma
7. Recibe retroalimentación de su docente.

Recursos:

- Herramienta de diagramación
- Computadora
- Internet.

Práctica 6: Conversión del Modelo ER al Modelo Relacional**Duración:**

03 horas

Procedimiento:

1. Atiende a la explicación de su docente.
2. Aplica reglas de transformación a tablas.
3. Define claves primarias, foráneas y restricciones.
4. Realiza un diccionario de datos por tabla.
5. Entrega el reporte en PDF.
6. Recibe retroalimentación de su docente

Recursos de apoyo:

- Computadora
- Acceso a internet
- Manejador de BD

Práctica 7: Normalización**Duración:**

03 horas

Procedimiento:

1. Atiende a la explicación de su docente.
2. Parte del modelo relacional elaborado (práctica 6).
3. Identifica dependencias funcionales.
4. Aplica 1NFm 2NFm 3NF y BCNF.
5. Documenta el proceso y captura evidencias.
6. Entrega el reporte en PDF.
7. Recibe retroalimentación de su docente

Recursos de apoyo:

- Computadora
- Acceso a internet
- Plantillas de normalización

Práctica 8: Comparación entre modelos SQL y NoSQL**Duración:**

03 horas

Procedimiento:

1. Atiende a la explicación de su docente sobre paradigmas NoSQL.
2. Instala o accede a un servicio en línea (MongoDB, Atlas o similar).
3. Crea una colección de ejemplo.
4. Inserta y consulta documentos.
5. Documenta el proceso y captura evidencias.
6. Entrega el reporte comparativo entre SQL y NoSQL en PDF.
7. Recibe retroalimentación de su docente

Recursos de apoyo:

- Computadora
- Acceso a internet
- MongoDB, Atlas

Práctica 9: Creación del Esquema de la Base de Datos

Duración:

03 horas

Procedimiento:

1. Atiende a la explicación de su docente sobre sentencias DDL.
2. Crea la BD del proyecto.
3. Define tablas, claves primarias y foráneas.
4. Documenta las sentencias ejecutadas.
5. Documenta el proceso y captura evidencias.
6. Entrega el reporte en PDF.
7. Recibe retroalimentación de su docente

Recursos de apoyo:

- Computadora
- Acceso a internet
- Manejador de BD
- editor SQL

Práctica 10: DML – Manipulación de Datos

Duración:

03 horas

Procedimiento:

1. Atiende a la explicación de su docente.
2. Inserta datos de prueba en las tablas (INSERT).
3. Actualiza registros (UPDATE).
4. Elimina registros (DELETE).
5. Documenta el proceso y captura evidencias.
6. Entrega el reporte en PDF.
7. Recibe retroalimentación de su docente

Recursos de apoyo:

- Computadora
- Acceso a internet
- Manejador de BD
- Editor SQL

Unidad IV: Sistemas Manejadores de Base de Datos y Administraciones

Práctica 11: Administración de usuarios, permisos y seguridad

Duración:

03 horas

Procedimiento:

1. Atiende a la explicación de su docente.
2. Crea usuarios con distintos roles.
3. Asigna permisos a nivel base, tabla y vista.
4. Prueba restricciones y accesos.
5. Documenta el proceso y captura evidencias.
6. Entrega el reporte en PDF.
7. Recibe retroalimentación de su docente

Recursos de apoyo:

- Computadora
- Acceso a internet
- Manejador de BD

Práctica 12: Respaldos y restauración de Bases de Datos

Duración:

03 horas

Procedimiento:

1. Atiende a la explicación de su docente.
2. Genera diferentes tipos de respaldo.
3. Realiza restauraciones controladas.
4. Documenta el proceso y captura evidencias.
5. Entrega el reporte en PDF.
6. Recibe retroalimentación de su docente

Recursos de apoyo:

- Computadora
- Acceso a internet
- Manejador de BD

Unidad V: Bases de Datos en Aplicaciones y Arquitecturas Modernas

Práctica 13: Optimización y planes de ejecución

Duración:

03 horas

Procedimiento:

1. Atiende a la explicación de su docente.
2. Ejecuta consultas con EXPLAIN o ANALIZE.
3. Identifica cuellos de botella.
4. Aplica índices y compara tiempos.
5. Documenta el proceso antes y después y captura evidencias.
6. Entrega el reporte en PDF.
7. Recibe retroalimentación de su docente

Recursos de apoyo:

- Computadora
- Acceso a internet
- Manejador de BD

Práctica 14: Integración con Aplicación y Pruebas del Sistema Final

Duración:

03 horas

Procedimiento:

1. Atiende a la explicación de su docente.
2. Usa el lenguaje de programación (Python, Node.js, otro).
3. Conecta la aplicación a la base de datos.
4. Implementa consultas desde el código.
5. Documenta el proceso y captura evidencias.
6. Entrega el reporte en PDF.
7. Recibe retroalimentación de su docente

Recursos de apoyo:

- Computadora
- Acceso a internet
- Manejador de BD

Unidad VI: Implementación de un Sistema de Base de Datos

Práctica 15: Construcción del Modelo Conceptual, Lógico y Físico del Proyecto Final

Duración:

03 horas

Procedimiento:

1. Atiende a la explicación de su docente sobre los lineamientos del proyecto final.
2. Elabora el Modelo ER utilizando alguna herramienta de modelado (Draw.io, ERDPlus, otra).
3. Identifica entidades, atributos, claves, relaciones, cardinalidad y restricciones.
4. Transforma el modelo al modelo lógico relacional.
5. Genera el modelo físico en el Manejador seleccionado (MySQL o PostgreSQL).
6. Documenta el proceso y captura evidencias.
7. Entrega el reporte en PDF.
8. Recibe retroalimentación de su docente

Recursos de apoyo:

- Computadora
- Acceso a internet
- Manejador de BD
- Herramienta de modelado

Práctica 16: Implementación, Pruebas y Documentación del Sistema de Base de Datos Final

Duración:

03 horas

Procedimiento:

1. Atiende a la explicación de su docente.
2. Implementa físicamente la base de datos final usando scripts SQL.
3. Desarrolla consultas necesarias para la aplicación.
4. Implementa consultas desde el código.
5. Realiza un plan de pruebas.
6. Integra la base de datos a una aplicación cliente para verificar el flujo completo de datos.
7. Exporta la base de datos o genera respaldo según el Manejador.
8. Documenta el proceso y captura evidencias.
9. Entrega el reporte en PDF.

10. Recibe retroalimentación de su docente

Recursos de apoyo:

- Computadora
- Acceso a internet
- Manejador de BD
- Editor SQL
- Cliente HTTP (python, Node.js)

IX. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Estudio de caso
- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Técnica expositiva

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Visitas a campo
- Mapas conceptuales
- Cuadros comparativos

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	30%
– Prácticas laboratorio	30%
– Sistema de BD	40%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Caballero, R., & Martín, E. (2020). *Las bases de big data*. Los Libros de la Catarata.
- Elmasri, R., & Navathe, S. B. (2016). *Fundamentals of database systems* (7.^a ed.). Pearson.
- Kuzmiakova, A. (2023). *Creation and management of database systems*. eBook Academic Collection (EBSCOhost).
- Pulido Romero, E., Escobar Domínguez, Ó., & Núñez Pérez, J. A. (2019). *Base de datos*. Grupo Editorial Patria Educación.
- Sánchez Ahijado, A. (2023). *Administración de sistemas gestores de bases de datos*. Marcombo Formación
- Sarasa, A. (2016). *Introducción a las bases de datos NoSQL usando MongoDB*. Universitat Oberta de Catalunya.
- Socas, R., Maho, A., & Gómez, L. (2025). *Bases de datos: Teoría y práctica aplicada a ingeniería del software*. Marcombo.
- Wiese, L. (2015). *Advanced data management: For SQL, NoSQL, cloud and distributed databases*. Walter de Gruyter GmbH & Co. KG. eBook Academic Collection (EBSCOhost)

Recursos de acceso libre o complementarios externos

- Elmasri, R., & Navathe, S. B. (2016). *Fundamentals of database systems* (7th ed.). Pearson. [classic]
- Kleppmann, M. (2017). *Designing data-intensive applications: The big ideas behind reliable, scalable, and maintainable systems*. O'Reilly Media. [classic]
- Massachusetts Institute of Technology. (2010). *Database systems (6.830/6.814)*. MIT OpenCourseWare. <https://ocw.mit.edu/courses/6-830-database-systems-fall-2010/> [classic]
- MongoDB, Inc. (n.d.). MongoDB documentation. <https://www.mongodb.com/docs/>
- Object Management Group. (n.d.). *UML specifications*. <https://www.omg.org/spec/>

Olatunju, D. (2025). *javascript Using SQL in Node.js with Sequelize*. AppSignal. <https://blog.appsignal.com/2025/06/11/using-sql-in-nodejs-with-sequelize.html>

PostgreSQL Global Development Group. (n.d.). *PostgreSQL* documentation. <https://www.postgresql.org/docs/>

World Wide Web Consortium. (n.d.). *W3C standards*. <https://www.w3.org/>

XII. PERFIL DOCENTE

Formación en el área de computación o áreas afines; con experiencia en planeación, diseño e implementación de proyectos de bases de datos, configuración de ambientes de desarrollo para bases de datos y manejo de las siguientes tecnologías: arquitecturas de desarrollo (frameworks), sistemas de control de versiones, lenguajes para desarrollo de aplicaciones en entornos de Internet. Ser una persona creativa, tener un eficiente manejo grupal, fomentar el pensamiento crítico y creativo en sus estudiantes.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Investigación de Operaciones
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 02 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:06
Etapas de formación:	Disciplinaria
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Everardo Gutierrez López Jonathan Verdugo Olachea José Ángel González Fraga
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Proporcionar a cada estudiante de Ciencias Computacionales experiencias para aplicar sus conocimientos en programación y modelación. Introducir la perspectiva estocástica y determinística de la modelación, mostrando cómo la teoría de probabilidad y el álgebra lineal intervienen en la construcción de modelos de Investigación de Operaciones, destacando el papel de las Ciencias Computacionales en la resolución de problemas presentes en la industria, la economía y las ciencias computacionales.

Esta unidad de aprendizaje pertenece a la etapa disciplinaria, es de carácter obligatorio y pertenece al área de conocimiento de Matemáticas.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Desarrollar sistemas de optimización a través del modelado de fenómenos de espera y de sistemas lineales para resolver problemas con eficiencia, responsabilidad y creatividad.

IV. EVIDENCIA DE APRENDIZAJE

Integración de un portafolio que recopile las prácticas de laboratorio, ejercicios y tareas realizadas durante el curso.

Desarrollo de un modelo y herramienta de software que implemente un modelo de Investigación de Operaciones para resolver un problema real, acompañado de su reporte técnico correspondiente. El producto podrá abordar, entre otros, temas de simulación de líneas de espera, programación lineal, asignación, transporte o estrategias de optimización de recursos.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia y vanguardia

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos de la Investigación de Operaciones

Competencia

Identificar la importancia de la Investigación de Operaciones a través del análisis de los conceptos de modelos y sistemas que la integran, con el fin de apoyar la toma de decisiones

en distintos tipos de organizaciones, con responsabilidad y sentido crítico.

Contenido

- 1.1. Descripción del origen de la Investigación de Operaciones
- 1.2. Evolución e impacto de la Investigación de Operaciones en el ámbito científico y tecnológico
- 1.3. Conceptos de Modelos y Sistemas en Investigación de Operaciones

Duración

02 horas

Unidad II. Programación Lineal

Competencia

Aplicar modelos de programación lineal mediante los métodos gráfico, algebraico y Simplex para resolver problemas propios de la industria, las organizaciones y los entornos tecnológicos, con actitud crítica.

Contenido

- 2.1. Sistemas de desigualdades lineales en dos variables
- 2.2. Formulación de modelos con Programación Lineal (PL)
- 2.3. Método gráfico para la solución de modelos de PL con dos variables
- 2.4. Método Algebraico para la solución de modelos de PL
- 2.5. Método Simplex
- 2.6. Método de la Gran M
- 2.7. Dualidad en PL
- 2.8. Análisis de sensibilidad

Duración

10 horas

Unidad III. Modelos de Transporte y Asignación

Competencia

Aplicar modelos especializados de programación lineal, particularmente los de transporte y asignación, para resolver problemas estructurados en contextos industriales, empresariales y tecnológicos, con actitud crítica.

Contenido

- 3.1. Problemas de Transporte
 - 3.1.1. Formulación del modelo de transporte
 - 3.1.2. Métodos de solución del problema de transporte
 - 3.1.3. Aplicaciones del Método de Transporte en la industria y los servicios
- 3.2. Problemas de Asignación

- 3.2.1. Formulación del modelo de asignación
- 3.2.2. El método Húngaro para problemas de asignación
- 3.2.3. Aplicaciones del modelo de asignación en procesos organizacionales y tecnológicos

Duración

06 horas

Unidad IV. Procesos Estocásticos y Líneas de Espera

Competencia

Aplicar modelos de líneas de espera a través de la resolución de ejercicios de problemas de colas para entender los conceptos básicos sobre los fenómenos de espera con actitud crítica.

Contenido

- 4.1. Procesos Estocásticos
 - 4.1.1. Conceptos Básicos de los procesos estocásticos
 - 4.1.2. Cadenas de Markov en tiempo discreto
 - 4.1.3. Caminata Aleatoria
- 4.2. Fundamentos de modelos de Líneas de Espera
 - 4.2.1. Conceptos básicos de la teoría de Líneas de Espera.
 - 4.2.2. Modelos estocásticos de Líneas de Espera
 - 4.2.3. Distribución Exponencial, Proceso de Poisson y Proceso de Nacimiento y Muerte

Duración

08 horas

Unidad V. Modelo de Líneas de Espera

Competencia

Aplicar modelos de líneas de espera para analizar y resolver problemas en sistemas de servicio utilizados en entornos computacionales, industriales y organizacionales, interpretando correctamente su ámbito de aplicación con una actitud crítica.

Contenido

- 5.1. Modelos con población Infinita
 - 5.1.1. Modelo M/M/1
 - 5.1.2. Modelo M/M/s
- 5.2. Modelos con población finita
 - 5.2.1. Modelo M/M/1/K
 - 5.2.2. Modelo M/M/s/K

Duración

06 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I: Fundamentos de la Investigación de Operaciones

Práctica 1: Modelación de problemas mediante Programación Lineal

Duración

02 horas

Procedimiento

1. Atiende las indicaciones de su docente para el desarrollo de la práctica
2. Revisa el proceso para convertir una descripción verbal de un problema en un modelo matemático de programación lineal, identificando:
 - a. variables de decisión
 - b. función objetivo
 - c. restricciones
3. Formula el modelo matemático correspondiente al problema proporcionado por su docente.
4. Analiza las posibles soluciones del modelo y verifica la factibilidad de cada una.
5. Interpreta la solución obtenida en el contexto del problema original, destacando su significado práctico.
6. Elabora un breve reporte que incluya el planteamiento del modelo, la solución obtenida y una conclusión final del problema.

Recursos de apoyo

- Computadora
- Editor de texto
- Formato de práctica proporcionada por su docente

Unidad II: Programación Lineal

Práctica 2: Resolución de Programas Lineales con el Método Gráfico

Duración

02 horas

Procedimiento

1. Atiende las indicaciones de su docente.
2. Revisa las notas de solución e interpretación geométrica de programas lineales.
3. Analiza un problema de programación lineal brindado por su docente.
4. Resuelve geoméricamente el problema de programación lineal.
5. Verifica la solución utilizando un programa de graficación.
6. Presenta los resultados y conclusiones en un reporte.

Recursos de apoyo

- Computadora
- Programa o software de graficación
- Editor de texto
- Formato de práctica proporcionada por su docente

Práctica 3: Resolución de Programas Lineales con el Método Algebraico

Duración

02 horas

Procedimiento

1. Atiende las indicaciones de su docente.
2. Revisa las notas de solución algebraica de programas lineales.
3. Analiza un problema de programación lineal brindado por el docente.
4. Resuelve algebraicamente el problema de programación lineal.
5. Verifica la solución obtenida, comparando la solución del método gráfico utilizando un programa de graficación.
6. Presenta los resultados y conclusiones en un reporte.

Recursos de apoyo

- Computadora
- Programa o software de graficación
- Editor de texto
- Formato de práctica proporcionada por su docente

Práctica 4: Resolución de Programas Lineales con el Método Simplex

Duración

02 horas

Procedimiento

1. Atiende las indicaciones de su docente.
2. Revisa las notas de solución por el Método Simplex.
3. Analiza un problema de programación lineal brindado por su docente.
4. Resuelve el problema de programación lineal por el método Simplex.
5. Verifica la solución obtenida, comparando la solución utilizando un programa de software o lenguaje de programación que resuelva problemas de optimización.
6. Presenta los resultados y conclusiones en un reporte.

Recursos de apoyo

- Computadora

- Programa o software para resolver problemas de optimización
- Editor de texto
- Formato de práctica proporcionada por su docente

Práctica 5: Técnicas con Variables Artificiales en Programación Lineal

Duración

02 horas

Procedimiento

1. Atiende las indicaciones de su docente.
2. Revisa la necesidad de las variables artificiales en problemas con restricciones de tipo $\geq$ o $=$.
3. Analiza un problema de programación lineal brindado por el docente.
4. Resuelve el problema de programación lineal por el método de la gran M.
5. Verifica la solución obtenida, comparando la solución utilizando un programa de software o lenguaje de programación que resuelva problemas de optimización.
6. Presenta los resultados y conclusiones en un reporte.

Recursos de apoyo

- Computadora
- Programa o software para resolver problemas de optimización
- Editor de texto
- Formato de práctica proporcionada por su docente

Práctica 6: Programación Lineal y Dualidad

Duración

02 horas

Procedimiento

1. Atiende las indicaciones de su docente.
2. Revisa el concepto de dualidad.
3. Formula el problema dual asociado a un modelo dado.
4. Resuelve ambos: primal y dual, utilizando un software de optimización.
5. Analiza las ventajas de emplear la dualidad para interpretar soluciones.
6. Presenta los resultados y conclusiones en un reporte.

Recursos de apoyo

- Computadora
- Programa de software para resolver problemas de optimización
- Editor de texto
- Formato de práctica proporcionada por su docente

Unidad III: Modelos de Transporte y Asignación

Práctica 7: Problemas de Transporte

Duración

04 horas

Procedimiento

1. Atiende las indicaciones de su docente.
2. Revisa el modelo clásico de transporte: oferta, demanda y tabla de costos.
3. Aplica métodos como Esquina Noroeste, Costo Mínimo o Aproximación de Vogel para obtener una solución factible inicial.
4. Utiliza el método de pasos secuenciales o distribución modificada para obtener la solución óptima del problema.
5. Analiza los resultados y discute la eficiencia del método seleccionado.
6. Presenta los resultados y conclusiones en un reporte.

Recursos de apoyo

- Computadora
- Programa de software para resolver problemas de optimización
- Editor de texto
- Formato de práctica proporcionada por su docente

Práctica 8: Problemas de Asignación

Duración

02 horas

Procedimiento

1. Atiende las indicaciones de su docente.
2. Revisa los métodos para resolver problemas de asignación (Método Húngaro).
3. Resuelve a mano un problema de asignación real o simulado.
4. Compara tu solución con la solución obtenida por un software de optimización.
5. Interpreta los resultados y su impacto en la eficiencia del sistema.
6. Presenta un reporte con gráficas o tablas relevantes.

Recursos de apoyo

- Computadora
- Programa de software para resolver problemas de optimización
- Editor de texto
- Formato de práctica proporcionada por su docente

Unidad IV: Procesos Estocásticos y Líneas de Espera

Práctica 9: Cadenas de Markov

Duración

04 horas

Procedimiento

1. Atiende las indicaciones de su docente.
2. Revisa los conceptos básicos de procesos estocásticos y transiciones de estados.
3. Construye una matriz de transición a partir de un problema contextual dado por su docente.
4. Simula la evolución del sistema utilizando un software que permita el producto de matrices.
5. Analiza estados absorbentes, estacionarios y distribuciones a largo plazo.
6. Presenta un reporte con gráficas o tablas relevantes.

Recursos de apoyo

- Computadora
- Programa de software para trabajar con matrices
- Editor de texto
- Formato de práctica proporcionada por su docente

Práctica 10: Introducción a las líneas de espera

Duración

04 horas

Procedimiento

1. Atiende las indicaciones de su docente.
2. Revisa conceptos básicos de teoría de colas de Poisson y el modelo M/M/1.
3. Resuelve ejercicios básicos utilizando fórmulas analíticas y software que maneje distribuciones de probabilidad para simular tiempos de espera y atención.
4. Analiza tiempos promedio de espera, longitud de la cola y utilización del sistema.
5. Presenta un reporte con gráficas o tablas relevantes.

Recursos de apoyo

- Computadora
- Programa de software para trabajar con números pseudoaleatorios y distribuciones de probabilidad.
- Editor de texto
- Formato de práctica proporcionada por su docente

Unidad V: Modelo de Líneas de Espera

Práctica 11: Líneas de Espera

Duración

02 horas

Procedimiento

1. Atiende las indicaciones de su docente
2. Estudia el modelo M/M/s y sus parámetros principales.
3. Resuelve ejercicios usando herramientas de simulación.
4. Analiza cómo cambia el desempeño del sistema al aumentar el número de servidores.
5. Presenta un reporte con gráficas o tablas relevantes.

Recursos de apoyo

- Computadora
- Programa de software para trabajar con líneas de espera
- Editor de texto
- Formato de práctica proporcionada por su docente

Práctica 12: Líneas de Espera en Autoservicio y Sistemas con Máquinas

Duración

04 horas

Procedimiento

1. Atiende las indicaciones de su docente.
2. Revisa modelos de colas aplicados a sistemas automáticos y de autoservicio.
3. Simula escenarios utilizando un software para trabajar con líneas de espera.
4. Analiza métricas clave como tasa de servicio, tasa de llegada, bloqueo y utilización.
5. Elabora un reporte final con discusión de resultados y recomendaciones.

Recursos de apoyo

- Computadora
- Programa de software para trabajar con líneas de espera (Google Colab)
- Editor de texto
- Formato de práctica proporcionada por su docente

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Aprendizaje basado en preguntas
- Técnica expositiva
- Discusión guiada
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Cuadros comparativos

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	30%
– Prácticas	20%
– Tareas	20%
– Proyecto Final	30%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Devore, J. L. (2005). Probabilidad y estadística para ingeniería y ciencias. Thomson. ISBN: 9706864571. [clásica]
- Guerrero Salas, H. (2009). *Programación lineal aplicada*. Gutiérrez. [clásica]
- Hillier, F. S., & Lieberman, G. J. (2023). *Introducción a la investigación de operaciones* (11.^a ed.). McGraw-Hill Interamericana.
[https://uabc.vitalsource.com/reader/books/9786071520609/epubcfi/6/2\[%3Bvnd.vst.idref%3Ditem-0\]!/4](https://uabc.vitalsource.com/reader/books/9786071520609/epubcfi/6/2[%3Bvnd.vst.idref%3Ditem-0]!/4)
- Hillier, F.S. & G.J. Lieberman (2023). *Introducción a la investigación de Operaciones*, (7na ed.) [traducción, Murrieta, M. J. E.] Mac Graw-Hill.
- Karlin, S. (2014). *A first course in stochastic processes*. Academic press. [classic]
- Palacios Figueroa, R. (2017). Investigación de operaciones I: Programación lineal. [Recurso en línea]. <https://libcon.rec.uabc.mx:2461/read/9786076228937/index> [clásica]
- Taha, H.A. (2017). *Investigación de Operaciones*, (10a ed.) [traducción, Murrieta, M. J. E.] Alfa Omega México. [clásica]
- Winston, W. L., & Goldberg, J. B. (2004). *Operations research: applications and algorithms* (Vol. 3). Duxbury press [clasic]

Recursos de acceso libre o complementarios externos

- Google Colab. (2025). *Google Colaboratory* [Plataforma de software].
<https://colab.research.google.com/>
- Haeussler, E. F., & Paul, R. S. (2003). *Matemáticas para administración y economía*. Pearson educación. [clásica]

XII. PERFIL DOCENTE

Profesionista en Ciencias Computacionales o área afín, preferentemente con posgrado, tener experiencia en docencia y en modelación matemática, probabilidad y programación lineal. Ser una persona creativa, que emplee y fomente el trabajo colaborativo, desarrolle el pensamiento crítico en sus estudiantes y promueva la ética profesional.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Teoría de Autómatas
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 03 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Disciplinaria
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	María Victoria Meza Kubo Everardo Guitierrez López Pedro Perez Ruiz
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Esta unidad de aprendizaje muestra la construcción matemática de la teoría de la computación, mediante el diseño de autómatas y lenguajes abstractos como base para la construcción de traductores y compiladores.

Los fundamentos matemáticos de la teoría de autómatas serán de gran utilidad para profesionales en ciencias computacionales, ya que le permitirán sustentar de manera formal los problemas que se les presenten relacionados con los lenguajes y la abstracción formal de máquinas de información.

La unidad de aprendizaje es de carácter obligatorio, se encuentra en la etapa disciplinaria y es del área de conocimiento de matemáticas. Esta asignatura está seriada con la asignatura subsecuente de Compiladores.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Construir modelos formales de cómputo para reconocer y describir lenguajes formales, aplicando los conocimientos que sustentan el modelo teórico y conceptual de las computadoras y del quehacer computacional, como base para la construcción de traductores y compiladores, con actitud crítica, analítica y propositiva.

IV. EVIDENCIA DE APRENDIZAJE

Entrega compendio de prácticas de laboratorio con implementación de autómatas solicitadas durante el curso, cada una de ellas en código fuente y ejecutable, acompañadas de la documentación correspondiente.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Desarrollar software de base eficiente y robusto mediante la integración de conocimientos en arquitectura y redes de computadoras y los fundamentos de compiladores y sistemas operativos, para optimizar el rendimiento, la gestión de recursos y la comunicación entre sistemas, asegurando aplicaciones funcionales, seguras, escalables y sostenibles, con una actitud ética, responsable y visionaria.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Pensamiento Crítico

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos de teoría de autómatas

Competencia

Interpretar los fundamentos de la teoría de autómatas, mediante el estudio de máquinas y lenguajes formales, como base para su diseño y construcción, con una actitud crítica y curiosidad científica.

Contenido

- 1.1. Introducción a lenguajes formales y teoría de la computación
- 1.2 Máquinas abstractas y su papel en compiladores
- 1.3 Notación, alfabetos, cadenas y lenguajes

Duración

04 horas

Unidad II. Gramáticas formales

Competencia

Construir gramáticas regulares y libres de contexto, mediante la identificación de sus características, derivaciones y ambigüedad, para modelar lenguajes formales, con pensamiento crítico y creativo.

Contenido

- 2.1. Gramáticas regulares.
- 2.2 Gramáticas libres de contexto (CFG).
- 2.3 Árboles de derivación.
- 2.4 Ambigüedad en gramáticas formales.

Duración

06 horas

Unidad III. Autómatas finitos y lenguajes regulares

Competencia

Diseñar autómatas de estado finito y expresiones regulares, mediante la implementación de simuladores y ejercicios prácticos en software, para resolver problemas de validación y búsqueda de patrones de cadenas, con responsabilidad y pensamiento crítico.

Contenido

- 3.1. Introducción a los autómatas finitos
- 3.2. Autómata finito determinístico
- 3.3. Autómata finito no determinístico
- 3.4. Autómatas finitos con transiciones- λ
- 3.5 Expresiones regulares y equivalencia con autómatas
- 3.6 Aplicaciones en búsqueda y validación de patrones

Duración

08 horas

Unidad IV. Propiedades de los lenguajes regulares

Competencia

Aplicar técnicas de minimización y demostraciones de no regularidad en los lenguajes regulares a través de algoritmos y ejercicios prácticos, con el fin de identificar los límites de los lenguajes regulares y optimizar autómatas, con rigor lógico y perseverancia.

Contenido

- 4.1. Propiedades de clausura
- 4.2. Minimización de autómatas
- 4.3. Demostraciones de no regularidad (lema de bombeo)

Duración

06 horas

Unidad V. Aplicaciones en análisis sintáctico

Competencia

Diseñar analizadores sintácticos mediante la aplicación de los conceptos de gramáticas formales y autómatas en el diseño de analizadores sintácticos, para comprender su papel en compiladores y procesamiento de lenguajes, con actitud crítica y analítica.

Contenido

- 5.1. Revisión de gramáticas en compiladores.
- 5.2 Conceptos básicos de análisis sintáctico.
- 5.3 Construcción de analizadores sintácticos a partir de gramáticas libres de contexto.
- 5.5 Estrategias de análisis: descendente y ascendente (introducción).
- 5.6 Proyecto integrador: implementación de un analizador sintáctico sencillo.

Duración

08 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Fundamentos de teoría de autómatas

Práctica 1: Manipulación de alfabetos, cadenas y lenguajes

Duración:

6 horas

Procedimiento:

1. Atiende a la explicación sobre alfabetos, cadenas y lenguajes.
2. En equipos, selecciona un alfabeto (ejemplo $\{0,1\}$, $\{a,b\}$).
3. Genera subconjuntos de cadenas aplicando concatenación, unión y estrella de Kleene.
4. Implementa un programa en Python/C que genere todas las cadenas hasta una longitud dada.
5. Identifica patrones y propiedades en los lenguajes obtenidos.
6. Documenta los resultados con ejemplos de ejecución.

Recursos de apoyo:

- Computadoras, acceso a Internet.

Unidad II. Gramáticas formales

Práctica 2: Construcción de gramáticas regulares

Duración

6 horas

Procedimiento

1. Atiende la explicación sobre gramáticas regulares.
2. Construye una gramática regular que genere cadenas binarias con número par de 1s.
3. Diseña una gramática formal para expresiones aritméticas simples.
4. Verifica la generación de cadenas y construye árboles de derivación.
5. Identifica si alguna gramática presenta ambigüedad.
6. Elabora un informe con diagramas de derivación y ejemplos de gramáticas formales.

Recursos de apoyo

- Computadora, editores de texto.

Práctica 3: Gramáticas ambiguas y árboles de derivación

Duración

6 horas

Procedimiento

1. Atiende a la explicación sobre ambigüedad en gramáticas.
2. Diseña una CFG para expresiones con suma y multiplicación.
3. Construye dos árboles de derivación diferentes para una misma cadena.
4. Modifica la gramática para eliminar la ambigüedad aplicando reglas de precedencia y asociatividad.
5. Documenta el proceso comparando la gramática ambigua y la no ambigua.

Recursos de apoyo

- Computadora, papel y lápiz.

Unidad III. Autómatas finitos y lenguajes regulares

Práctica 4: Construcción y simulación de DFA y NFA

Duración

6 horas

Procedimiento

1. Atiende a la explicación sobre DFA y NFA.
2. Diseña un DFA que reconozca cadenas que terminan en "01".
3. Diseña un NFA equivalente y verifica su funcionamiento.
4. Simula ambos autómatas en JFLAP.
5. Implementa un simulador en código para probarlos.
6. Redacta un informe con capturas de la simulación y código.

Recursos de apoyo

- Computadora, acceso a Internet.

Práctica 5: Expresiones regulares y validación de patrones

Duración

6 horas

Procedimiento

1. Atiende a la explicación sobre expresiones regulares.
2. Construye regex que validen:
3. Correos electrónicos.
4. Códigos postales.
5. Palíndromos de longitud par.
6. Implementa en código un validador que reciba cadenas de entrada y verifique si cumplen con el patrón.
7. Prueba tu validador con diferentes casos de prueba.

8. Documenta los resultados con capturas de pantalla.

Recursos de apoyo

- Computadora y acceso a internet.

Unidad IV. Propiedades de los lenguajes regulares

Práctica 6: Minimización de autómatas

Duración

6 horas

Procedimiento

1. Atiende a la explicación sobre minimización de autómatas.
2. Diseña un DFA para el lenguaje de cadenas binarias con número par de ceros.
3. Aplica el algoritmo de minimización paso a paso (tabla de equivalencia).
4. Verifica el autómata reducido con JFLAP.
5. Documenta el proceso en un informe con capturas.

Recursos de apoyo

- Computadora, lápiz, papel.

Práctica 7: Demostraciones con el lema de bombeo

Duración

6 horas

Procedimiento

1. Atiende a la explicación sobre el lema de bombeo.
2. En equipos, seleccionen lenguajes candidatos (ejemplo: $\{a^n b^n\}$, $\{0^p \mid p \text{ es primo}\}$).
3. Apliquen el lema de bombeo para probar si son o no regulares.
4. Elaboren una justificación formal paso a paso.
5. Presenten sus conclusiones en una exposición corta.

Recursos de apoyo

- Computadora, acceso a Internet.

Unidad V. Aplicaciones en análisis sintáctico

Práctica 8: Implementación de un analizador sintáctico

Duración

6 horas

Procedimiento

1. Atiende a la explicación sobre análisis sintáctico descendente.
2. Diseña una CFG para un lenguaje de expresiones aritméticas.
3. Implementa un analizador sintáctico descendente recursivo en Python/Java.
4. Prueba el analizador con expresiones correctas e incorrectas.
5. Documenta el código, resultados y posibles limitaciones.

Recursos de apoyo

- Computadora, acceso a Internet

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Estudio de caso
- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Técnica expositiva

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	30%
– Prácticas laboratorio	30%
– Compendio de prácticas laboratorio	20%
– Tareas	20%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Alfonseca Cubero, E., Alfonseca Moreno, M., Moriyón Salomón, R., & Sánchez González, C. (2007). *Teoría de autómatas y lenguajes formales*. McGraw-Hill. [clásica]

Denning, P. J., Dennis, J. B., & Qualitz, J. E. (1978). *Machines, languages and computation*. Prentice Hall Professional Technical Reference. [clásica]

Hopcroft, J. E., Ullman, J. D., & Motwani, R. (2007). *Introducción a la teoría de autómatas, lenguajes y computación* (3.^a ed.). McGraw-Hill. [clásica]

Recursos de acceso libre o complementarios externos

Carrasco Jiménez, R. C., Carrasco, R. C., Calera Rubio, J., Forcada Zubizarreta, M. L. (2000). *Teoría de lenguajes, gramáticas y autómatas para informáticos*. Publicaciones de la Universidad de Alicante. [clásica]

Hopcroft, J. E., Ullman, J. D., & Motwani, R. (2007). *Introducción a la teoría de autómatas, lenguajes y computación* (3.^a ed.). McGraw-Hill. (clásico) disponible en <https://docencia.eafranco.com/materiales/teoriacomputacional/books/TeoriaDeAutomatas,lenguajesYComputacion-Hopcroft.pdf> [clásica]

XII. PERFIL DOCENTE

Licenciatura en Ciencias Computacionales o área afín. Preferentemente con posgrado del área de matemáticas o computación, deberá contar con conocimientos en el manejo de lenguajes y teoría de la computación y al menos dos años de experiencia docente. Ser una persona honesta, proactiva y responsable con su labor docente.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Ingeniería de Software
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 03 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Disciplinaria
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Alma Rocío Cabazos Marín Alberto Leopoldo Morán y Solares
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vazquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La unidad de aprendizaje de Ingeniería de Software se encuentra ubicada en la etapa disciplinaria con carácter obligatorio. Su área de conocimiento es Programación e Ingeniería de Software, es deseable haber acreditado la unidad de aprendizaje de Metodología de la Programación. La finalidad de la unidad de aprendizaje es que el estudiantado comprenda y aplique los métodos, técnicas y herramientas actuales de la ingeniería de software, así como las diferentes actividades que se realizan para la producción de software de calidad. Su importancia reside en que permite comprender y aplicar un conjunto de métodos, técnicas y herramientas actuales de la Ingeniería de Software, así como diferentes actividades que se realizan para la producción de software de calidad a una escala mediana y grande en el desarrollo de software para un cliente real. Asimismo, permite asociar aspectos de control de calidad, gestión de configuración, y gestión de proyecto en el desarrollo de software.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Desarrollar un proyecto de software de mediana-grande escala de manera colaborativa por medio de técnicas y metodologías de la ingeniería de software, para resolver un problema que satisfaga las necesidades de un cliente real, demostrando una actitud objetiva, crítica, con disciplina y ética profesional.

IV. EVIDENCIA DE APRENDIZAJE

Portafolio integrador que dé cuenta de los procesos de administración, desarrollo y control de calidad del proyecto de software, que integra:

- Minutas de reuniones de avance periódicas y la documentación técnica por fase y disciplina del proceso de desarrollo
- Un producto de software funcional acorde a los requerimientos, y su respectiva documentación final técnica y de usuario del proyecto.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia y Vanguardia.

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos de ingeniería de software

Competencia

Examinar los conceptos, ámbito y terminología de la ingeniería del software, así como sus dominios de aplicación mediante las referencias teóricas especializadas en ingeniería de software, para establecer su importancia en el desarrollo de software en las organizaciones, con disciplina, ética profesional, actitud objetiva y crítica.

Contenido

- 1.1. Introducción a la Ingeniería de Software (IS)
 - 1.1.1. ¿Qué es el software?
 - 1.1.2. Naturaleza y cualidades del software
 - 1.1.3. Software de Calidad
 - 1.1.4. ¿Qué es la Ingeniería de software?
 - 1.1.5. Historia de la IS
 - 1.1.6. Ámbitos de aplicación en las organizaciones

Duración

03 horas

Unidad II. Procesos formales de desarrollo de Software

Competencia

Analizar los procesos formales que se utilizan para el desarrollo de software, así como los principales tipos y modelos del proceso de desarrollo, a través del estudio del estado del arte de las metodologías para su aplicación en el desarrollo de software, con una actitud disciplinada, objetiva y ética profesional.

Contenido

- 2.1. Proceso de Software
- 2.2. Modelo de ciclo de vida del software
- 2.3. Metodologías predictivas
- 2.4. Metodologías ágiles
- 2.5. Estudio de casos
 - 2.5.1. Proceso Unificado de Desarrollo (RUP)
 - 2.5.1.1. Fases
 - 2.5.1.2. Disciplinas fundamentales y de apoyo
 - 2.5.1.3. Artefactos
 - 2.5.1.4. Roles
 - 2.5.2. Programación Extrema (XP)
 - 2.5.3. Scrum

Duración

04 horas

Unidad III. Fundamentos de gestión de proyectos de software**Competencia**

Aplicar los fundamentos de la gestión de proyectos, mediante el uso de métodos y técnicas, para especificar, planear y dar seguimiento a un proyecto de software, con precisión técnica, responsabilidad y actitud crítica.

Contenido

- 3.1. Visión general de la gestión de proyectos
- 3.2. Definición del proyecto
 - 3.2.1. Panorámica del proyecto y condiciones de satisfacción
 - 3.2.2. Enunciado de definición del proyecto
- 3.3. Planeación y seguimiento del proyecto
 - 3.3.1. Actividades y tareas
 - 3.3.1.1. Estructura de Descomposición de Trabajo (Work Breakdown Structure)
 - 3.3.2. Estimación de duración de tareas y esfuerzo de trabajo
 - 3.3.3. Estimación de recursos humanos asociados al proyecto
 - 3.3.4. Estimación de costos
 - 3.3.5. Calendarización del proyecto
 - 3.3.5.1. Diagramas de Gantt

Duración

05 horas

Unidad IV. Análisis de requerimientos y especificación**Competencia**

Analizar los requerimientos de un proyecto de software, a través de la aplicación de técnicas de obtención, análisis y documentación de requisitos, así como del uso de modelos y herramientas propias de la ingeniería de software, para establecer y priorizar las necesidades del cliente, con una actitud creativa, objetiva y colaborativa, demostrando responsabilidad, pensamiento crítico y compromiso ético.

Contenido

- 4.1. Requerimientos vs especificación
- 4.2. Conceptos, actividades, actores y características de los requerimientos
- 4.3. Tipos de requerimientos
- 4.4. Actividades para la obtención de requerimientos
- 4.5. Notación
 - 4.5.1. Casos de uso
 - 4.5.2. Modelos Entidad-Relación

4.5.3. Diagramas de clases y objetos

4.5.4. Diagramas CRC

Duración

04 horas

Unidad V. Análisis y Diseño Orientado a Objetos (ADOO)

Competencia

Diseñar los modelos estático, dinámico, funcional y de despliegue mediante el uso de las herramientas y técnicas para la especificación y concepción de un producto de software, para resolver una problemática real de forma profesional y congruente con los procesos de la organización cliente, con responsabilidad, ética, disciplina y creatividad.

Contenido

5.1. Conceptos fundamentales de análisis y diseño

5.2. Actividades de ADOO

5.3. Modelo de análisis y diseño orientado a objetos

5.4. Notación

5.4.1. Refinando los casos de uso y el diagrama de clases y objetos

5.4.2. Diagramas de secuencia, actividad y comunicación

5.4.3. Diagramas de máquinas de estado

5.4.4. Diagramas de componentes y de despliegue

5.5. Patrones de Diseño

Duración

07 horas

Unidad VI. Construcción y Pruebas Orientada a Objetos

Competencia

Desarrollar una aplicación de software mediante el uso de las herramientas y técnicas de construcción y validación del software para resolver un problema propuesto por un cliente conforme a los procesos de su organización, con responsabilidad, ética, disciplina y creatividad.

Contenido

6.1. Modelado e implementación

6.2. Reutilización de código

6.3. Principios fundamentales de la construcción de software

6.3.1. Minimizar complejidad

6.3.2. Anticipar los cambios

6.3.3. Construir para verificar

6.3.4. Utilización de estándares

6.4. Metodología y objetivos de las pruebas

6.5. Técnicas de Pruebas

- 6.5.1. Pruebas de caja blanca
- 6.5.2. Pruebas de caja negra
- 6.6. Niveles y proceso de prueba
- 6.7. Estudio de caso

Duración

05 horas

Unidad VII. Control de Calidad y Gestión de configuraciones

Competencia

Integrar aspectos fundamentales del control de calidad y de la gestión de configuraciones, a través del uso de técnicas, herramientas y estándares de control y gestión para incrementar la calidad del proceso y del producto en un proyecto de software, con responsabilidad, ética, disciplina y creatividad.

Contenido

- 7.1. Conceptos y objetivos de la calidad de Software
- 7.2. Calidad del producto
- 7.3. Calidad del proceso
- 7.4. Estándares y modelos de calidad de software
- 7.5. Conceptos y objetivos del control de configuración
- 7.6. Actividades de la configuración del software
- 7.7. Control de Versiones
- 7.8. Control de cambios
- 7.9. Técnicas y herramientas

Duración

04 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I: Fundamentos de ingeniería de software

Práctica 1: Línea del tiempo de la Historia de la Ingeniería de Software

Duración

03 horas

Procedimiento

1. Atiende a la explicación de su docente sobre el surgimiento de la Ingeniería de Software y el contexto histórico de la llamada crisis del software.
2. Integra equipos de trabajo y revisa el objetivo de la actividad.
3. Investiga de fuentes confiables los principales hitos históricos de la Ingeniería de Software, tales como:
 - a. Crisis del software
 - b. Aparición de metodologías de desarrollo
 - c. Evolución de modelos de proceso
 - d. Surgimiento de estándares y buenas prácticas
4. Selecciona los hitos más representativos y ordena cronológicamente los eventos identificados.
5. Elabora en equipo una línea del tiempo (gráfica o esquemática) que muestre de forma clara la evolución de la Ingeniería de Software.
6. Redacta una descripción breve para cada hito, explicando su relevancia e impacto en el desarrollo de software.
7. Presenta el trabajo al resto del grupo, explicando los hitos más relevantes identificados.
8. Entrega la línea del tiempo acompañada de una breve referencia de las fuentes consultadas.
9. Recibe retroalimentación docente y del resto del grupo.

Recursos de apoyo

- Libros y artículos especializados en Ingeniería de Software
- Apuntes de clase
- Computadora con acceso a Internet
- Herramientas digitales para diagramación (PowerPoint, Canva, Draw.io, línea del tiempo en Word/Excel, etc.)
- Hojas y marcadores (en caso de elaboración manual)

Unidad II: Procesos formales de desarrollo de Software

Práctica 2: Análisis comparativo de procesos y metodologías de desarrollo de software

Duración

06 horas

Procedimiento

1. Atiende a la explicación de su docente sobre el concepto de proceso de software y su importancia en el desarrollo disciplinado de sistemas.
2. Integra equipos de trabajo y revisa el caso de estudio asignado.
3. Identifica y describe las etapas generales del proceso de software aplicables al caso (análisis, diseño, construcción, pruebas, despliegue y mantenimiento).
4. Analiza los principales modelos de ciclo de vida del software (cascada, incremental, iterativo, evolutivo), identificando sus características generales.
5. Clasifica los modelos analizados en metodologías predictivas y metodologías ágiles.
6. Investiga y analiza el Proceso Unificado de Desarrollo (RUP), considerando:
 - a. Fases
 - b. Disciplinas fundamentales y de apoyo
 - c. Artefactos
 - d. Roles
7. Investiga y analiza las metodologías ágiles Programación Extrema (XP) y Scrum, identificando sus prácticas, roles, eventos y artefactos principales.
8. Elabora un cuadro comparativo entre RUP, XP y Scrum, considerando estructura del proceso, nivel de formalidad, roles, artefactos y contexto de aplicación.
9. Presenta los resultados del análisis con el resto del grupo.
10. Entrega la documentación completa del ejercicio y recibe retroalimentación docente y grupal.

Recursos de apoyo

- Libros y artículos especializados en Ingeniería de Software
- Documentación oficial de RUP, Scrum y XP
- Apuntes de clase
- Computadora con acceso a Internet
- Procesador de texto o herramienta colaborativa

Unidad III: Fundamentos de gestión de proyectos de software

Práctica 3: Visión general y definición del proyecto de software

Duración

02 horas

Procedimiento

1. Atiende a la explicación de su docente sobre la gestión de proyectos de software y su importancia en el desarrollo organizado de sistemas.
2. Integra equipos de trabajo y analiza el caso de estudio asignado.
3. Identifica el propósito general del proyecto y el problema que busca resolver.
4. Delimita el alcance del proyecto, identificando qué se incluye y qué se excluye.
5. Define las condiciones de satisfacción del proyecto considerando:
 - a. Entregables principales
 - b. Restricciones de tiempo

- c. Restricciones de costo
 - d. Nivel de calidad esperado
6. Discute en equipo la coherencia entre alcance, restricciones y condiciones de satisfacción.
 7. Documenta la panorámica del proyecto.
 8. Presenta los resultados al grupo.
 9. Entrega la documentación completa del ejercicio y recibe retroalimentación docente y grupal.

Recursos de apoyo

- Apuntes de clase
- Caso de estudio proporcionado por el docente
- Bibliografía de gestión de proyectos de software
- Computadora con acceso a Internet
- Procesador de texto o herramienta colaborativa

Práctica 4: Enunciado de definición del proyecto

Duración

02 horas

Procedimiento

1. Atiende a la explicación de su docente sobre el enunciado de definición del proyecto.
2. Revisa la panorámica del proyecto elaborada en la práctica anterior.
3. Redacta el nombre formal del proyecto.
4. Elabora la justificación del proyecto en función de la necesidad organizacional.
5. Define el objetivo general del proyecto.
6. Establece el alcance del proyecto, especificando inclusiones y exclusiones.
7. Identifica los principales entregables del proyecto.
8. Define los supuestos y restricciones relevantes.
9. Revisa la claridad, coherencia y redacción del documento.
10. Presenta los resultados al grupo.
11. Entrega la documentación completa del ejercicio y recibe retroalimentación docente y grupal.

Recursos de apoyo

- Apuntes de clase
- Bibliografía de gestión de proyectos de software
- Computadora con acceso a Internet
- Procesador de texto o herramienta colaborativa

Práctica 5: Identificación de actividades y Estructura de Descomposición de Trabajo (WBS)

Duración

02 horas

Procedimiento

1. Atiende a la explicación de su docente sobre actividades, tareas y WBS.
2. Identifica las actividades principales necesarias para desarrollar el proyecto.
3. Descompone cada actividad en tareas manejables y medibles.
4. Organiza las tareas de forma jerárquica.
5. Construye la Estructura de Descomposición de Trabajo (WBS).
6. Revisa que el nivel de detalle sea adecuado para la planeación.
7. Presenta los resultados al grupo.
8. Entrega la documentación completa del ejercicio y recibe retroalimentación docente y grupal.

Recursos de apoyo

- Apuntes de clase
- Bibliografía de gestión de proyectos de software
- Computadora con acceso a Internet
- Procesador de texto o herramienta colaborativa
- Herramientas de diagramación o procesador de texto

Práctica 6: Estimación de duración y esfuerzo de las tareas

Duración

02 horas

Procedimiento

1. Atiende a la explicación de su docente sobre técnicas básicas de estimación.
2. Revisa la WBS elaborada previamente.
3. Estima la duración de cada tarea.
4. Estima el esfuerzo de trabajo en horas/persona por tarea.
5. Justifica brevemente las estimaciones realizadas.
6. Verifica la coherencia entre duración, esfuerzo y alcance.
7. Registra la información en una tabla estructurada.
8. Presenta los resultados al grupo.
9. Entrega la documentación completa del ejercicio y recibe retroalimentación docente y grupal.

Recursos de apoyo

- Apuntes de clase
- Bibliografía de gestión de proyectos de software
- WBS del proyecto

- Computadora con acceso a Internet
- Hoja de cálculo o herramienta colaborativa

Práctica 7: Estimación de recursos humanos y costos del proyecto

Duración

02 horas

Procedimiento

1. Atiende a la explicación de su docente sobre recursos humanos y costos en proyectos de software.
2. Define los roles necesarios para el proyecto (líder, analista, desarrollador, tester, etc.).
3. Asigna roles a las tareas definidas en la WBS.
4. Establece un costo por hora para cada rol (supuesto).
5. Calcula el costo por tarea con base en el esfuerzo estimado.
6. Calcula el costo total del proyecto.
7. Revisa la consistencia de la estimación.
8. Presenta los resultados al grupo.
9. Entrega la documentación completa del ejercicio y recibe retroalimentación docente y grupal.

Recursos de apoyo

- Apuntes de clase
- Bibliografía de gestión de proyectos de software
- Tabla de estimaciones de esfuerzo
- Computadora con acceso a Internet
- Hoja de cálculo o herramienta colaborativa

Práctica 8: Calendarización y Diagrama de Gantt del proyecto

Duración

02 horas

Procedimiento

1. Atiende a la explicación de su docente sobre calendarización y dependencias entre tareas.
2. Revisa la WBS y las estimaciones de duración.
3. Identifica las dependencias entre tareas.
4. Ordena las tareas de acuerdo con su secuencia lógica.
5. Elabora el cronograma del proyecto.
6. Construye un Diagrama de Gantt.
7. Revisa la viabilidad del calendario propuesto.
8. Presenta el cronograma al grupo.
9. Presenta los resultados al grupo.
10. Entrega el Diagrama de Gantt final y recibe retroalimentación docente y grupal.

Recursos de apoyo

- Apuntes de clase
- Bibliografía de gestión de proyectos de software
- Tabla de estimaciones de esfuerzo
- Computadora con acceso a Internet
- Hoja de cálculo o herramienta colaborativa

Unidad IV: Análisis de requerimientos y especificación

Práctica 9: Identificación y análisis de requerimientos

Duración

02 horas

Procedimiento

1. Atiende a la explicación de su docente sobre la diferencia entre requerimientos y especificación.
2. Integra equipos de trabajo y analiza el caso de estudio asignado.
3. Identifica a los principales actores involucrados en el sistema.
4. Reconoce las necesidades del cliente y del usuario final.
5. Redacta requerimientos preliminares de forma clara y comprensible.
6. Clasifica los requerimientos identificados según su naturaleza.
7. Discute en equipo la importancia de documentar adecuadamente los requerimientos.
8. Entrega el listado inicial de requerimientos y recibe retroalimentación docente.

Recursos de apoyo

- Apuntes de clase
- Caso de estudio
- Computadora con procesador de texto

Práctica 10: Obtención de requerimientos y elaboración de casos de uso

Duración

02 horas

Procedimiento

1. Atiende a la explicación de su docente sobre técnicas de obtención de requerimientos (entrevistas, cuestionarios, observación).
2. Simula una sesión de obtención de requerimientos con roles asignados (cliente, analista).
3. Ajusta los requerimientos con base en la información obtenida.
4. Identifica los principales casos de uso del sistema.
5. Elabora el diagrama de casos de uso.

6. Revisa la coherencia entre requerimientos y casos de uso.
7. Entrega el diagrama elaborado y recibe retroalimentación docente.

Recursos de apoyo

- Apuntes de clase
- Caso de estudio
- Herramienta de diagramación (e.g. Draw.io, Lucidchart, StarUML)

Práctica 11: Modelado de datos y estructura del sistema

Duración

02 horas

Procedimiento

1. Atiende a la explicación de su docente sobre Modelos Entidad-Relación y diagramas de clases y objetos.
2. Identifica las entidades principales del sistema.
3. Define atributos y relaciones entre entidades.
4. Elabora el modelo Entidad-Relación.
5. Deriva el diagrama de clases a partir del modelo de datos.
6. Analiza la coherencia entre datos, clases y requerimientos.
7. Entrega los modelos elaborados y recibe retroalimentación docente.

Recursos de apoyo

- Apuntes de clase
- Casos de uso
- Herramientas de modelado UML

Práctica 12: Diagramas CRC y especificación de requerimientos

Duración

02 horas

Procedimiento

1. Atiende a la explicación de su docente sobre diagramas CRC.
2. Identifica las clases clave del sistema.
3. Define responsabilidades y colaboraciones.
4. Elabora los diagramas CRC.
5. Integra los artefactos generados en un documento de especificación.
6. Revisa la trazabilidad entre requerimientos, modelos y diagramas.
7. Entrega la documentación final y recibe retroalimentación docente.

Recursos de apoyo

- Apuntes de clase
- Diagramas previos

- Procesador de texto

Unidad V. Análisis y Diseño Orientado a Objetos (ADOO)

Práctica 13: Refinamiento de casos de uso y diagramas de clases

Duración

01 hora

Procedimiento

1. Atiende a la explicación de su docente sobre el refinamiento de casos de uso.
2. Revisa los casos de uso definidos en la unidad anterior.
3. Ajusta y detalla los casos de uso considerando flujos principales y alternos.
4. Refina el diagrama de clases y objetos a partir de los requerimientos.
5. Verifica la coherencia entre casos de uso y clases.
6. Entrega los diagramas refinados y recibe retroalimentación docente.

Recursos de apoyo

- Apuntes de clase
- Casos de uso previos
- Herramienta UML (e.g. Draw.io, Lucidchart, StarUML)

Práctica 14: Diagramas de secuencia, actividad y comunicación

Duración

02 horas

Procedimiento

1. Atiende a la explicación de su docente sobre diagramas dinámicos.
2. Selecciona casos de uso representativos.
3. Elabora diagramas de secuencia que representan la interacción entre objetos.
4. Construye diagramas de actividad para describir flujos de procesos.
5. Elabora diagramas de comunicación.
6. Revisa la consistencia entre diagramas dinámicos y el modelo estático.
7. Entrega los diagramas elaborados y recibe retroalimentación docente.

Recursos de apoyo

- Apuntes de clase
- Casos de uso refinados
- Herramienta UML (e.g. Draw.io, Lucidchart, StarUML)

Práctica 15: Diagramas de máquinas de estado

Duración

01 hora

Procedimiento

1. Atiende a la explicación de su docente sobre diagramas de máquinas de estado.
2. Identifica objetos del sistema cuyo comportamiento depende de estados.
3. Define estados, eventos y transiciones.
4. Elabora los diagramas de máquinas de estado.
5. Revisa la coherencia con los diagramas de secuencia y actividad.
6. Entrega los diagramas elaborados y recibe retroalimentación docente.

Recursos de apoyo

- Apuntes de clase
- Diagramas dinámicos previos
- Herramienta UML (e.g. Draw.io, Lucidchart, StarUML)

Práctica 16: Diagramas de componentes y de despliegue

Duración

01 hora

Procedimiento

1. Atiende a la explicación de su docente sobre diagramas de componentes y despliegue.
2. Identifica los componentes principales del sistema.
3. Define interfaces y dependencias entre componentes.
4. Elabora el diagrama de componentes.
5. Identifica los nodos físicos y su distribución.
6. Elabora el diagrama de despliegue.
7. Entrega los diagramas elaborados y recibe retroalimentación docente.

Recursos de apoyo

- Apuntes de clase
- Diagramas previos
- Herramienta UML (e.g. Draw.io, Lucidchart, StarUML)

Práctica 17: Aplicación de patrones de diseño

Duración

02 horas

Procedimiento

1. Atiende a la explicación de su docente sobre patrones de diseño.
2. Identifica problemáticas recurrentes en el diseño del sistema.
3. Selecciona uno o más patrones de diseño adecuados (creacionales, estructurales o de comportamiento).
4. Justifica la selección del patrón.
5. Integra el patrón al modelo de diseño.
6. Ajusta los diagramas correspondientes.

7. Entrega la documentación final y recibe retroalimentación docente.

Recursos de apoyo

- Apuntes de clase
- Bibliografía sobre patrones de diseño
- Diagramas previos
- Herramienta UML (e.g. Draw.io, Lucidchart, StarUML)

Unidad VI: Construcción y Pruebas Orientada a Objetos

Práctica 18: Del modelo al código – Implementación orientada a objetos

Duración

02 horas

Procedimiento

1. Atiende a la explicación de su docente sobre la relación entre modelos UML y código orientado a objetos.
2. Revisa los diagramas de clases y casos de uso desarrollados en la Unidad V.
3. Selecciona un lenguaje de programación orientado a objetos (e.g. Java, C# o Python).
4. Implementa las clases principales del sistema respetando atributos, métodos y relaciones.
5. Aplica principios de construcción de software:
 - a. Minimizar la complejidad.
 - b. Anticipar cambios mediante diseño flexible.
 - c. Utilizar convenciones de codificación.
6. Documenta brevemente las decisiones de implementación.
7. Presenta el avance al resto del grupo y recibe retroalimentación.

Recursos de apoyo

- Apuntes de clase
- Diagramas UML de la unidad anterior
- IDE de desarrollo
- Guías de estilo del lenguaje seleccionado

Práctica 19: Reutilización de código y principios de construcción

Duración

02 horas

Procedimiento

1. Atiende a la explicación del docente sobre reutilización de código.
2. Identifica componentes o funcionalidades susceptibles de reutilización.
3. Implementa mecanismos de reutilización (herencia, composición, interfaces o bibliotecas).

4. Refactoriza el código para reducir duplicidad y mejorar la mantenibilidad.
5. Aplica estándares básicos de codificación y documentación.
6. Analiza en equipo los beneficios obtenidos y recibe retroalimentación grupal y docente.

Recursos de apoyo

- Apuntes de clase
- Material sobre principios de reutilización
- Repositorio de código
- Documentación oficial del lenguaje seleccionado

Práctica 20: Diseño y ejecución de pruebas de software

Duración

02 horas

Procedimiento

1. Atiende a la explicación de su docente sobre metodología y objetivos de las pruebas.
2. Identifica los niveles de prueba aplicables (unitarias, integración, sistema).
3. Diseña casos de prueba de caja negra a partir de los casos de uso.
4. Diseña casos de prueba de caja blanca considerando la lógica interna del código.
5. Ejecuta las pruebas y registra resultados.
6. Identifica defectos y propone acciones correctivas.
7. Entrega evidencia de la actividad realizada y recibe retroalimentación docente.

Recursos de apoyo

- Apuntes de clase
- Código del sistema
- Plantillas de casos de prueba
- Frameworks básicos de pruebas (opcional)

Unidad VII: Control de Calidad y Gestión de Configuraciones

Práctica 21: Análisis de la calidad del producto y del proceso de software

Duración

02 horas

Procedimiento

1. Atiende a la explicación de su docente sobre los conceptos y objetivos de la calidad de software.
2. Analiza un caso de estudio breve proporcionado por el docente.
3. Identifica atributos de calidad del producto (funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad).
4. Identifica actividades relacionadas con la calidad del proceso (planeación, seguimiento, verificación y mejora).

5. Relaciona los atributos de calidad con posibles métricas e indicadores.
6. Discute en equipo la importancia de la calidad en proyectos reales.
7. Presenta conclusiones al grupo y recibe retroalimentación docente y grupal.

Recursos de apoyo

- Apuntes de clase
- Casos de estudio
- Normas y modelos de calidad (e.g. ISO/IEC, CMMI)

Práctica 22: Estándares y modelos de calidad de software

Duración

02 horas

Procedimiento

1. Atiende a la explicación de su docente sobre estándares y modelos de calidad.
2. Investiga en equipo un modelo o estándar asignado (ISO/IEC 25010, ISO/IEC 12207, CMMI, entre otros).
3. Identifica objetivos, estructura y beneficios del modelo.
4. Analiza un ejemplo de aplicación del estándar en un proyecto.
5. Elabora un resumen ejecutivo del análisis.
6. Entrega resumen ejecutivo y recibe retroalimentación docente.

Recursos de apoyo

- Apuntes de clase
- Documentación oficial de estándares
- Internet
- Procesador de texto

Práctica 23: Gestión de configuraciones y control de versiones

Duración

02 horas

Procedimiento

1. Atiende a la explicación de su docente sobre conceptos y objetivos del control de configuración.
2. Identifica los elementos de configuración del proyecto (código, documentación, modelos).
3. Inicializa un repositorio de control de versiones (por ejemplo, Git).
4. Realiza operaciones básicas: commit, branch, merge y manejo de conflictos.
5. Define una convención básica de mensajes de cambios.
6. Registra y documenta los cambios realizados.
7. Entrega documento de registro y recibe retroalimentación docente.

Recursos de apoyo

- Git y repositorio remoto (e.g. GitHub, GitLab o similar)
- Manuales de control de versiones
- Computadora con acceso a Internet

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente establece la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Estudio de caso
- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Técnica expositiva
- Discusión guiada
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Mapas mentales
- Mapas conceptuales
- Cuadros comparativos
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	20%
– Prácticas de laboratorio y tareas	30%
– Portafolio integrador	50%
– Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Ortega, J. M. (2020). *Desarrollo seguro en ingeniería del software*. Alfaomega Grupo Editor.
- Piattini, M. G., García, F. O., García, I., & Pino, F. (2020). *Calidad de sistemas de información* (5a ed.). Ra-Ma Editorial.
- Pressman, R. S., & Maxim, B. R. (2021). *Ingeniería de software: Un enfoque práctico* (9a ed.). McGraw-Hill Interamericana.
- Project Management Institute. (2021). *Guía de los fundamentos para la dirección de proyectos (Guía del PMBOK) y el estándar para la dirección de proyectos* (7.ª ed.). Project Management Institute.
- Thomas, D., & Hunt, A. (2022). *El programador pragmático: Viaje a la maestría (Edición 20 aniversario)*. Anaya Multimedia.

Recursos de acceso libre o complementarios externos

- Association for Computing Machinery. (s.f.). *ACM digital library*. <https://dl.acm.org>
- Elsevier. (s.f.). *Journal of systems and software*. ScienceDirect. <https://www.sciencedirect.com/journal/journal-of-systems-and-software>
- IEEE. (s.f.). *IEEE Xplore digital library*. <https://ieeexplore.ieee.org>
- IEEE Computer Society. (s.f.). *IEEE software*. <https://www.computer.org/csdl/magazine/so>

XII. PERFIL DOCENTE

Profesionista con licenciatura en Ciencias Computacionales o área afín, preferentemente con estudios de posgrado, con experiencia comprobable en la aplicación de la ingeniería de software. Deberá contar con conocimientos en lógica de programación, análisis, diseño e implementación de programas de computadora, así como en metodologías de programación y desarrollo de software. Asimismo, se espera que sea una persona creativa, que fomente el trabajo colaborativo, el pensamiento crítico en el estudiantado y fortalezca la ética profesional.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Sistemas Operativos
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 03 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Disciplinaria
Carácter:	Obligatoria
Requisito:	Haber cursado la unidad de aprendizaje Organización y Arquitectura de Computadoras.
Modalidad de impartición:	Presencial
Equipo de diseño:	Evelio Martínez Martínez Gerardo Tovar Ramos
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Establece la conexión entre hardware y software y proporciona al estudiantado una visión integral sobre la gestión, administración y optimización de los recursos de un sistema de cómputo. El estudiantado adquiere conocimientos sobre arquitectura de sistemas, administración de memoria, planificación de procesos, sistemas de archivos, concurrencia, sincronización y seguridad. Desarrolla habilidades para instalar, configurar, programar y optimizar sistemas, resolver problemas complejos y utilizar herramientas de administración. Al mismo tiempo, fortalece actitudes como pensamiento crítico, atención al detalle, responsabilidad en el uso de recursos y colaboración efectiva.

Esta unidad proporciona la base para desempeñarse en áreas como desarrollo de software, administración de sistemas, ciberseguridad y computación distribuida, capacitando para diseñar, implementar y mantener sistemas seguros y eficientes. Tiene carácter obligatorio, pertenece a la etapa disciplinaria y requiere saberes de Organización y Arquitectura de Computadoras, Estructura de Datos y Algoritmos, así como conocimientos básicos del entorno Unix/Linux, virtualización, comandos de consola y fundamentos de programación en C/C++.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Implementar soluciones basadas en los principios y mecanismos de los sistemas operativos de la gestión de recursos de hardware y software, mediante la integración de procesos, memoria, archivos, comunicación y seguridad en productos funcionales que serán diseñados y probados en laboratorio, con el propósito de garantizar sistemas de cómputo eficientes, confiables y seguros, actuando con responsabilidad, pensamiento crítico, atención al detalle y compromiso con la calidad.

IV. EVIDENCIA DE APRENDIZAJE

Diseñar e implementar un videojuego en lenguaje C sobre un entorno UNIX/Linux, integrando los mecanismos de Comunicación entre Procesos (IPC) como *pipes*, colas de mensajes, memoria compartida y *sockets*, con el fin de coordinar la interacción entre múltiples procesos que gestionen tareas como entrada de usuario, lógica de juego y visualización en consola; este producto deberá estar debidamente documentado y probado en laboratorio, demostrando no solo la comprensión de los conceptos de concurrencia y sincronización, sino también su capacidad de aplicarlos en un contexto práctico y creativo, fomentando la responsabilidad, el trabajo colaborativo y la innovación tecnológica.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Desarrollar software de base eficiente y robusto mediante la integración de conocimientos en arquitectura y redes de computadoras y los fundamentos de compiladores y sistemas operativos, para optimizar el rendimiento, la gestión de recursos y la comunicación entre

sistemas, asegurando aplicaciones funcionales, seguras, escalables y sostenibles, con una actitud ética, responsable y visionaria.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia, inclusión, vanguardia, innovación social.

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos de los sistemas operativos

Competencia

Analizar los conceptos fundamentales, funciones y evolución de los sistemas operativos mediante el estudio de sus características, componentes y tipos con apoyo de documentación técnica y ejemplos prácticos para comprender su importancia en la gestión de recursos y el funcionamiento de un sistema de cómputo, mostrando actitud crítica, responsabilidad, compromiso y ética profesional.

Contenido

- 1.1. Concepto de sistema operativo
- 1.2. Evolución histórica
- 1.3. Clasificación de los sistemas operativos
- 1.4. Componentes de un sistema operativo
- 1.5. Funciones de un sistema operativo
- 1.6. Tareas de un sistema operativo
- 1.7. Servicios que proporciona un sistema operativo
- 1.8. Estructura y arquitectura de un sistema operativo

Duración:

4 horas

Unidad II. Concurrencia, sincronización y comunicación entre procesos

Competencia

Desarrollar mecanismos de concurrencia, sincronización y comunicación entre procesos, utilizando algoritmos, primitivas del sistema operativo y herramientas de programación, con el propósito de garantizar la correcta coordinación y eficiencia en la ejecución de procesos, asumiendo actitudes de responsabilidad, colaboración y pensamiento crítico.

Contenido

- 2.1. Concepto de proceso e hilos
- 2.2. Estados de un proceso
- 2.3. Bloque de control de proceso (*PCB*)
- 2.4. Calendarización de procesos
 - 2.4.1. Algoritmos de calendarización de procesos

- 2.5. Comunicación entre procesos (IPC)
 - 2.5.1. Mecanismos de comunicación entre procesos
- 2.6. Procesos concurrentes
- 2.7. Exclusión mutua y sección crítica
- 2.8. Interbloqueo (Deadlocks)

Duración:

6 horas

Unidad III. Organización y administración de la memoria principal

Competencia

Implementar técnicas de organización y administración de la memoria principal, mediante el uso de modelos, algoritmos y estrategias del sistema operativo, con el propósito de optimizar el uso eficiente de los recursos de memoria, asumiendo actitudes de responsabilidad, orden y compromiso con la calidad.

Contenido

- 3.1. Conceptos básicos de memoria en sistemas operativos
- 3.2. Administración de la memoria
- 3.3. Requisitos de la gestión de la memoria
- 3.3. Asignación de memoria
 - 3.3.1. Asignación contigua
 - 3.3.1.1. Particiones fijas
 - 3.3.1.2. Particiones variables
 - 3.3.2. Asignación no contigua
- 3.4. Métodos de ajuste
- 3.5. Fragmentación de memoria
- 3.6. Swapping
- 3.7. Paging
- 3.8. Tendencias y tecnologías actuales

Duración:

4 horas

Unidad IV. Organización y administración de la memoria virtual

Competencia

Implementar mecanismos de organización y administración de la memoria virtual, mediante el uso de algoritmos de asignación, paginación, segmentación y técnicas de intercambio del sistema operativo, con el propósito de optimizar el aprovechamiento y la eficiencia en la gestión de memoria, demostrando responsabilidad, precisión y pensamiento crítico.

Contenido

- 4.1. Concepto de memoria virtual
- 4.2. Espacio de direcciones
- 4.3. Técnicas de organización de la memoria virtual
 - 4.3.1. Segmentación
 - 4.3.2. Paginación
- 4.4. Administración de tablas de páginas
- 4.5. Mecanismos de traducción eficiente
- 4.6. Fallos de página
- 4.7. Algoritmos de reemplazo de páginas
- 4.8. Tendencias actuales

Duración:

4 horas

Unidad V. Diseño e Interacción de los Sistemas de Archivos

Competencia

Analizar la arquitectura y funcionamiento de los sistemas de archivos mediante el estudio de sus estructuras internas, algoritmos de gestión y mecanismos de interacción con el sistema operativo, para comprender cómo se organiza, accede y asegura la información en dispositivos de almacenamiento, con pensamiento crítico, rigor técnico y compromiso con la calidad académica.

Contenido

- 5.1. Fundamentos de los sistemas de archivos
 - 5.1.1. Concepto y funciones principales
 - 5.1.2. Relación entre kernel, sistema de archivos y hardware
 - 5.1.3. Tipos de sistemas de archivos (jerárquicos, planos, distribuidos, con journaling)
- 5.2. Estructuras internas y organización de datos
 - 5.2.1. Bloques, sectores y clusters
 - 5.2.2. Inodos y tablas de asignación
 - 5.2.3. Directorios, enlaces y metadatos
 - 5.2.4. Estrategias de direccionamiento y acceso
- 5.3. Algoritmos y operaciones del sistema de archivos
 - 5.3.1. Asignación de espacio y gestión de bloques libres
 - 5.3.2. Caché de disco y buffering
 - 5.3.3. Lectura y escritura de archivos
 - 5.3.4. Manejo de concurrencia y consistencia
- 5.4. Interacción del sistema operativo con el almacenamiento
 - 5.4.1. Montaje y abstracción de dispositivos
 - 5.4.2. Controladores y subsistemas de E/S
 - 5.4.3. Virtual File System (VFS) como capa de abstracción
- 5.5. Seguridad y confiabilidad en los sistemas de archivos

- 5.5.1. Permisos y control de acceso
- 5.5.2. Journaling y tolerancia a fallos
- 5.5.3. Respaldo y recuperación

Duración:

6 horas

Unidad VI. Seguridad y protección en Sistemas Operativos

Competencia

Integrar mecanismos básicos de seguridad en sistemas operativos, aplicando controles de acceso, técnicas de aislamiento, protección de datos y respaldo, con el propósito de fortalecer la confiabilidad y disponibilidad de la información, con responsabilidad y pensamiento crítico en el uso de recursos.

Contenido

- 6.1. Seguridad en llamadas al sistema
 - 6.1.1. Control de acceso a archivos en C
 - 6.1.2. Manejo seguro de descriptores en fork/exec
 - 6.1.3. Creación segura de archivos temporales
- 6.2. Aislamiento y privilegios
 - 6.2.1. Uso de setuid y seteuid
 - 6.2.2. Capacidades POSIX en Linux
 - 6.2.3. Restricción de syscalls con seccomp
- 6.3. Seguridad en memoria y concurrencia
 - 6.3.1. Errores comunes en C (overflow, race conditions)
 - 6.3.2. Mitigaciones del kernel (ASLR, NX, canarios)
 - 6.3.3. Herramientas de análisis (Valgrind, Sanitizers)
- 6.4. Protección de datos
 - 6.4.1. Generación de llaves con /dev/random
 - 6.4.2. Cifrado de archivos con librerías en C
- 6.5. Monitoreo y auditoría
 - 6.5.1. Rastreo de syscalls con strace
 - 6.5.2. Registro de eventos con auditd
 - 6.5.3. Análisis básico de logs en Linux

Duración:

4 horas

Unidad VII. Tendencias y prácticas avanzadas en Sistemas Operativos

Competencia

Investigar las tendencias y tecnologías actuales en sistemas operativos, utilizando fuentes de información confiables, documentación técnica y herramientas de análisis comparativo, con el propósito de identificar innovaciones, su impacto en el rendimiento y la seguridad, y su

aplicabilidad en distintos entornos, demostrando curiosidad científica, pensamiento crítico y responsabilidad.

Contenido

- 7.1. Evolución y contexto actual
- 7.2. Sistemas operativos para móviles
- 7.3. Sistemas operativos en la nube
- 7.4. Virtualización y Contenedores
- 7.5. Sistemas operativos en entornos reales y embebidos
- 7.6. Tendencias futuras

Duración:

4 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE CAMPO

Unidad I: Introducción a los Sistemas Operativos

Práctica 1.1: Argumentos de entrada de la función *main()* [*argc* & *argv*]

Duración:

4 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Elaborará una serie de prácticas graduales en entornos UNIX/Linux, escribiendo programas en C que utilicen los parámetros *argc* y *argv* de la función *main()* para recibir y procesar argumentos desde la línea de comandos, con el compilador GCC y ejecutándolos como si fueran comandos del sistema. A través de ejercicios progresivos, desde mostrar mensajes personalizados hasta implementar utilidades sencillas como leer archivos o procesar opciones, comprenderá cómo los programas interpretan parámetros externos y cómo el compilador transforma el código en ejecutables. Este aprendizaje se reforzará con la experimentación en terminal, la depuración de errores y la comparación con comandos reales de UNIX, desarrollando así habilidades prácticas de programación a nivel de sistema, junto con actitudes de disciplina, curiosidad y autonomía en el manejo de herramientas de bajo nivel.
3. Entrega un documento en formato *.pdf* de la práctica guiada que le proporcionará el maestro, incluyendo las capturas de pantalla indicadas de las corridas en la consola.

Recursos de apoyo

- Distribución de Linux, compilador gcc, consola.
- C tutorial line arguments
https://www.tutorialspoint.com/cprogramming/c_command_line_arguments.htm
- Command line arguments in C
<https://www.javatpoint.com/command-line-arguments-in-c>

- Tutorial en español –argumentos `argc`, `argv`
<https://youtu.be/6kuhblkoDtE>

Unidad II: Concurrencia, sincronización y comunicación entre procesos

Práctica 2.1: Comandos de administración de procesos

Duración:

4 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Elaborará una práctica guiada progresiva: comenzará ejecutando comandos básicos como *ps* y *top* para observar procesos en ejecución, seguido de ejercicios estructurados donde identifique procesos específicos, analice su consumo de recursos y trace relaciones padre-hijo usando *ps**tree*. A través de laboratorios prácticos, cada estudiante creará sus propios procesos con programas simples, los monitoreará en tiempo real, experimentará con señales usando *kill*, y realizará análisis comparativos del comportamiento del sistema bajo diferentes cargas de trabajo. La metodología incluye ejercicios de resolución de problemas reales (como identificar procesos que consumen mucha CPU), documentación de observaciones, y reflexión sobre cómo lo observado se relaciona con los conceptos teóricos de scheduling, gestión de memoria y concurrencia vistos en clase.
3. Entregar un documento en formato .pdf de la práctica guiada que le proporcionará el maestro, incluyendo las capturas de pantalla indicadas.

Recursos de apoyo

- Distribución de Linux
- Compilador de lenguaje C GCC
- Consola

Práctica 2.2: Gestión de procesos con *fork()*

Duración:

4 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.

2. Elaborará ejercicios prácticos progresivos en lenguaje C: comenzará con un programa básico que use *fork()* para crear un proceso hijo, observando cómo ambos procesos imprimen mensajes diferentes según el valor de retorno de *fork()* (0 para hijo, PID positivo para padre). Progresará implementando ejemplos donde modifique variables en cada proceso para comprobar que tienen espacios de memoria separados, luego agregará la directiva *wait()* en el padre para sincronizar la ejecución y evitar procesos zombie. La metodología incluye compilar, ejecutar y depurar código real, usar herramientas como ps para observar los procesos creados, analizar la salida del programa para entender el flujo de ejecución concurrente, y resolver problemas comunes.
3. Entregar un documento en formato .pdf de la práctica guiada que le proporcionará el maestro, incluyendo las capturas de pantalla indicadas.

Recursos de apoyo

- Distribución de Linux
- Compilador gcc
- Consola

Práctica 2.3: Algoritmos de Calendarización de procesos

Duración:

4 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Elaborará ejercicios estructurados donde implementará cada algoritmo paso a paso con el mismo conjunto de procesos de entrada, calculando manualmente las métricas (tiempo de espera, retorno y respuesta) para cada caso, incluyendo la gráfica de *Gantt* correspondiente y comparando los resultados en tablas. Esta metodología práctica, combinando cálculo manual y análisis de resultados reales, le proporcionará comprensión profunda de las ventajas, desventajas y casos de uso apropiados para cada algoritmo de calendarización.
3. Entregar un documento en formato .pdf de la práctica guiada que le proporcionará el maestro, incluyendo las capturas de pantalla indicadas

Recursos de apoyo

- Práctica guiada de ejercicios sobre los algoritmos de calendarización de procesos

Práctica 2.4: Comando IPCS

Duración:

4 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Elaborará corridas en la consola Unix/Linux utilizando el comando `ipcs` observando en tiempo real cómo el sistema operativo gestiona recursos compartidos de mecanismos de IPC tales como *pipas*, *memoria compartida*, *colas de mensajes* y *sockets*. Desarrolla habilidades para identificar y diagnosticar problemas en aplicaciones concurrentes, comprender la persistencia de recursos IPC independiente de los procesos que los crean, gestionar recursos "huérfanos" que pueden afectar el rendimiento del sistema, y analizar permisos, propietarios y estados de recursos compartidos. Esta experiencia práctica le proporcionará comprensión profunda sobre debugging de aplicaciones multiproceso, administración de recursos del sistema, y las implicaciones de seguridad y rendimiento de los mecanismos IPC, conectando directamente la teoría de comunicación entre procesos con su implementación real en sistemas Unix/Linux.
3. Entregar un documento en formato .pdf de la práctica guiada que le proporcionará el maestro, incluyendo las capturas de pantalla indicadas.

Recursos de apoyo

- Distribución de Linux
- Compilador de lenguaje C `gcc`
- Consola

Práctica 2.5: Intercomunicación entre procesos con Pipas

Duración:

4 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Desarrolla programas en lenguaje C para comprender el mecanismo de intercomunicación de procesos (IPC) mediante pipes, analizando el funcionamiento de las llamadas al sistema fundamentales: `pipe()`, `fork()`, `dup2()` y `exec()`, así como las redirecciones de entrada y salida en terminales UNIX/Linux. Para reforzar este conocimiento, implementará ejercicios prácticos que incluyen: Desarrollo de pequeños programas en C que establecen comunicación entre procesos padre e hijo. Prácticas en shell para encadenar comandos utilizando el operador de tubería (`|`). A través de la

experimentación, resolución de problemas y comparación con otros mecanismos de IPC, cada estudiante desarrolla la capacidad de aplicar pipes en escenarios reales, comprendiendo sus ventajas, limitaciones y su papel fundamental en el diseño de sistemas concurrentes.

3. Entregar un documento en formato .pdf de la práctica guiada que le proporcionará el maestro, incluyendo las capturas de pantalla indicadas.

Recursos de apoyo

- Distribución de Linux
- Compilador de lenguaje C gcc
- Consola

Práctica 2.6: Intercomunicación entre procesos con memoria compartida

Duración:

3 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Desarrolla ejercicios prácticos en lenguaje C sobre sistemas UNIX/Linux, utilizando llamadas al sistema como `shmget`, `shmat`, `shmdt` y `shmctl` propios del mecanismo de intercomunicación de procesos (IPC), memoria compartida, para crear, adjuntar y liberar segmentos de memoria compartida, reforzando el aprendizaje con ejemplos que muestren el intercambio de datos entre procesos .
3. Entregar un documento en formato .pdf de la práctica guiada que le proporcionará el maestro, incluyendo las capturas de pantalla indicadas.

Recursos de apoyo

- Distribución de Linux
- Compilador de lenguaje C gcc
- Consola

Práctica 2.7: Intercomunicación entre procesos con cola de mensajes

Duración:

3 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Desarrolla código en lenguaje C de ejercicios guiados en los que utilizará las llamadas al sistema `msgget`, `msgsnd`, `msgrcv` y `msgctl`, propios del mecanismo de intercomunicación de procesos (IPC), cola de mensajes, para crear colas, enviar mensajes y recibirlos en procesos distintos. A través de ejemplos básicos, como un proceso productor que envía mensajes y un proceso consumidor que los recibe, hasta

aplicaciones más complejas con múltiples productores y consumidores, cada alumno desarrollar la capacidad de implementar soluciones que gestionen datos de manera ordenada y segura.

3. Entregar un documento en formato .pdf de la práctica guiada que le proporcionará el maestro, incluyendo las capturas de pantalla indicadas.

Recursos de apoyo

- Distribución de Linux
- Compilador de lenguaje C gcc
- Consola

Práctica 2.8: Intercomunicación entre procesos con Sockets

Duración: 3 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Desarrolla código en Lenguaje C utilizando llamadas al sistema como *socket*, *bind*, *listen*, *accept*, *connect*, *send* y *recv*, propias del mecanismo de intercomunicación de procesos (IPC), Sockets, comenzando con ejemplos simples de eco (un servidor que recibe y devuelve mensajes) hasta aplicaciones más elaboradas que gestionen múltiples clientes concurrentes. A través de estas prácticas, cada alumno experimentará cómo se establece, mantiene y cierra una conexión entre procesos locales o remotos, reforzando el aprendizaje mediante la depuración de errores y la comparación con otros mecanismos de IPC. De esta forma, comprenderá no sólo la teoría, sino también la aplicación real de los sockets como base de la comunicación en red y de servicios distribuidos.
3. Entregar un documento en formato .pdf de la práctica guiada que le proporcionará el maestro, incluyendo las capturas de pantalla indicadas.

Recursos de apoyo

- Distribución de Linux
- Compilador de lenguaje C gcc
- Consola

Práctica 2.8: Lectores y escritores

Duración: 3 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Desarrolla código en lenguaje C usando semáforos POSIX (*sem_init*, *sem_wait*, *sem_post*) para crear múltiples procesos lectores que accedan simultáneamente a un archivo compartido mientras un proceso escritor requiere acceso exclusivo, observando

cómo los semáforos coordinan el acceso. A través de ejercicios prácticos, modificará los tiempos de `sleep` en lectores y escritores para observar *starvation*, implementará logging para rastrear el orden de acceso, usará herramientas como `ipcs` para monitorear recursos IPC, y desarrollará versiones con diferentes prioridades para comparar comportamientos. Esta metodología práctica, compilando, ejecutando y depurando código real, le proporcionará comprensión profunda de cómo los mecanismos de sincronización resuelven problemas de concurrencia en aplicaciones del mundo real.

3. Entregar un documento en formato .pdf de la práctica guiada que le proporcionará el maestro, incluyendo las capturas de pantalla indicadas.

Recursos de apoyo

- Distribución de Linux
- Compilador de lenguaje C gcc
- Consola

Práctica 2.9: Los filósofos comelones

Duración: 3 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Desarrolla programas progresivos en C donde cada filósofo es un proceso o hilo creado con `fork()` o `pthread_create()`, usando semáforos POSIX (`sem_init`, `sem_wait`, `sem_post`) para representar cada tenedor como recurso limitado, y observando directamente cómo se produce el *deadlock* cuando todos los filósofos toman el tenedor izquierdo simultáneamente. Progresará implementando diferentes soluciones: ordenamiento de recursos (filósofos pares toman tenedor izquierdo primero, impares el derecho), uso de un semáforo adicional que limite a 4 filósofos comiendo simultáneamente, o implementación de un "camarero" que coordine el acceso mediante un proceso controlador. A través de ejercicios prácticos, modificará tiempos de espera y comida usando `sleep()`, agregará logging para rastrear el estado de cada filósofo y tenedor, experimentará provocando deliberadamente *deadlocks* para observar el comportamiento, y usará herramientas como `ps` e `ipcs` para monitorear procesos y recursos IPC. Esta metodología *hands-on*, compilando y ejecutando código que falla y luego aplicando soluciones, le proporcionará comprensión visceral de cómo surgen los *deadlocks* y las técnicas prácticas para prevenirlos en sistemas concurrentes reales.
3. Entregar un documento en formato .pdf de la práctica guiada que le proporcionará el maestro, incluyendo las capturas de pantalla indicadas.

Recursos de apoyo

- Distribución de Linux
- Compilador de lenguaje C gcc
- Consola

Unidad III: Organización y administración de la memoria principal

Práctica 3.1: Capas de la memoria

Duración: 3 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Desarrolla código en lenguaje C mediante ejemplos prácticos en C que muestren de forma explícita cómo se utilizan y organizan las capas de memoria: declarando variables globales y estáticas para observar su ubicación en las secciones de datos y BSS, empleando funciones con variables locales y parámetros para evidenciar el uso de la pila (stack), reservando y liberando memoria dinámica con malloc o calloc para experimentar con el heap, y analizando cómo el propio código se ubica en la sección de texto (text segment). A través de la compilación y ejecución de estos programas en sistemas UNIX/Linux, junto con el uso de herramientas como size, nm o gdb, cada alumno visualizará las direcciones de memoria de cada tipo de variable y comprenderá cómo el sistema operativo organiza, protege y administra estos espacios, reforzando el aprendizaje con la práctica, la depuración y la interpretación de resultados.
3. Entregar un documento en formato .pdf de la práctica guiada que le proporcionará el maestro, incluyendo las capturas de pantalla indicadas.

Recursos de apoyo

- Distribución de Linux
- Compilador de lenguaje C gcc
- Consola

Práctica 3.2: Valgrind: herramienta para gestión de memoria

Duración: 3 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Comprenderá la herramienta para gestión de memoria Valgrind ejecutando programas en C con errores de memoria intencionales, como variables no inicializadas, accesos a memoria ya liberada o fugas de memoria por no usar free(), los cuales serán compilados y ejecutados bajo Valgrind para observar en la práctica cómo la herramienta detecta y reporta estos problemas. A través de ejemplos progresivos, primero simples y luego más complejos, podrá interpretar los mensajes generados por módulos como Memcheck o Massif, identificando las líneas de código responsables del

mal uso de la memoria y corrigiéndolas paso a paso. Este enfoque práctico le permitirá visualizar y comprender la organización de la memoria en tiempo de ejecución, reforzando el aprendizaje mediante la depuración iterativa y la reflexión crítica sobre la importancia de escribir programas seguros y eficientes.

3. Entregar un documento en formato .pdf de la práctica guiada que le proporcionará el maestro, incluyendo las capturas de pantalla indicadas.

Recursos de apoyo

- Distribución de Linux
- Compilador de lenguaje C gcc
- Consola

Unidad V: Sistemas de archivos y particiones de disco

Práctica 5.1: Archivos especiales - soft link & hard link

Duración: 3 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Comprenderá mediante prácticas directas en sistemas UNIX/Linux, comenzando con el uso de comandos como `ln` para crear hard links y `ln -s` para soft links, observando las diferencias en atributos con `ls -l` y el comportamiento al eliminar o mover el archivo original. Después, reforzará el aprendizaje con ejercicios en C, utilizando llamadas al sistema como `link()`, `symlink()`, `unlink()` y `readlink()` para crear y manipular enlaces desde programas, comprobando cómo se gestionan a nivel del sistema de archivos y qué ocurre cuando cambian los inodos o las rutas. A través de estos ejemplos, cada alumno no solo identificará las diferencias prácticas y conceptuales entre enlaces duros y simbólicos, sino que también comprenderá su relevancia en la organización de directorios, optimización de almacenamiento y accesibilidad de archivos, desarrollando pensamiento crítico y habilidades de administración de sistemas.
3. Entregar un documento en formato .pdf de la práctica guiada que le proporcionará el maestro, incluyendo las capturas de pantalla indicadas.

Recursos de apoyo

- Distribución de Linux
- Compilador de lenguaje C gcc
- Consola

Práctica 5.2: Implementación del comando ls -l

Duración: 3 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Desarrolla código en lenguaje C mediante la implementación práctica de un programa que recorra directorios y obtenga información de los archivos utilizando funciones del sistema como *opendir()*, *readdir()* y llamadas a *stat()* para llenar la estructura *struct stat*. A partir de un ejemplo inicial que muestre solo los nombres de los archivos, progresivamente añadirá características para desplegar detalles como permisos, propietario, tamaño, tipo de archivo y fechas de modificación, reproduciendo paso a paso las variantes más comunes del comando *ls*. Durante el proceso, utilizará herramientas de depuración, pruebas con diferentes directorios y comparación con la salida del *ls* real, lo que le permitirá comprender a nivel interno cómo el sistema operativo gestiona los metadatos de los archivos. Este enfoque práctico, apoyado en código, análisis y experimentación, reforzará no solo el aprendizaje de conceptos teóricos sobre sistemas de archivos, sino también el desarrollo de habilidades de programación a bajo nivel y administración de sistemas UNIX/Linux.
3. Entregar un documento en formato .pdf de la práctica guiada que le proporcionará el maestro, incluyendo las capturas de pantalla indicadas.

Recursos de apoyo

- Distribución de Linux
- Compilador de lenguaje C gcc
- Consola

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Estudio de caso
- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos

- Aprendizaje basado en preguntas
- Técnica expositiva
- Debates
- Discusión guiada
- Instrucción guiada
- Foros

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Visitas a campo
- Mapas mentales
- Mapas conceptuales
- Cuadros comparativos
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	30%
– Prácticas de laboratorio	40%
– Proyecto final	30%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Bovet, D. P., & Cesati, M. (2006). *Understanding the Linux Kernel* (3rd ed.). Sebastopol, CA: O'Reilly Media. [Clásica].
- Gale, J. (2023). *Linux The Complete Manual*. Richdale Ventures. [eBook].
- Love, R. (2010). *Linux Kernel Development* (3rd ed.). Indianapolis, IN: Addison-Wesley. [Clásica].

- Padallan, J. (2023). *Introductory Guide to Operating Systems*. Arcler Press. [eBook].
- Silberschatz, A., Galvin, P. B., & Gagne, G. (2018). *Operating System Concepts* (10th ed.). John Wiley & Sons. [Clásica].
- Stallings, W. (2018). *Operating Systems: Internals and Design Principles* (9th ed.). Harlow, Pearson Education Limited. [Clásica]
- Tanenbaum, A. S., & Bos, H. (2015). *Modern Operating Systems* (4th ed.). Pearson. [Clásica].

Recursos de acceso libre o complementarios externos

- Gancarz, M. (2003). *Linux and the Unix Philosophy*. Digital Press. [eBook].
- La Red Martínez, D. L. (2004). *Sistemas operativos*.
<https://libcon.rec.uabc.mx:2321/es/lc/uabc/titulos/77467>
- Prof. Pankaj Sinha. (2023). *Operating System Concepts*. Laxmi Publications Pvt Ltd. [eBook].
- Sawant, Uday R.; Pelz, Oliver; Hobson, Leemans, Jonathan & William (2017). *Linux: Powerful Server Administration*. Packt Publishing. [eBook]
- Vedran Dakic, & Jasmin Redzepagic. (2022). *Linux Command Line and Shell Scripting Techniques : Master Practical Aspects of the Linux Command Line and Then Use It As a Part of the Shell Scripting Process*. Packt Publishing. [eBook].

XII. PERFIL DOCENTE

Contar con sólida formación en arquitectura de computadoras, programación a nivel de sistema (C/C++), sistemas operativos UNIX/Linux, administración de procesos, gestión de memoria, sistemas de archivos, concurrencia y seguridad informática. Asimismo, se requiere experiencia práctica en instalación, configuración, virtualización y optimización de distintos entornos operativos (Unix, Linux, Windows y plataformas en la nube), así como dominio de herramientas de monitoreo y resolución de problemas. La persona docente debe mantener actualización constante frente a los avances tecnológicos y poseer habilidades didácticas que le permitan explicar conceptos complejos de manera clara, fomentando en el estudiantado pensamiento crítico, proactividad, responsabilidad y trabajo colaborativo.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Física Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Análisis Numérico
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 02 Horas Laboratorio: 02 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:08
Etapas de formación:	Disciplinaria
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Selene Solorza Calderón José Ariel Camacho Gutiérrez
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

El área de análisis numérico se ocupa del estudio y la fundamentación teórica de los métodos numéricos utilizados para resolver problemas matemáticos cuya solución no puede obtenerse mediante procedimientos analíticos exactos. Esta unidad de aprendizaje tiene como objetivo introducir al estudiantado en el uso de métodos numéricos clásicos para determinar raíces de ecuaciones no lineales, resolver sistemas de ecuaciones lineales y estimar el comportamiento de conjuntos de datos mediante técnicas de interpolación polinómica. Este tipo de problemas aparece en diversos campos del conocimiento, tanto en las ciencias exactas como en las humanidades. La unidad pertenece al plan de estudios de Física con carácter obligatorio y pertenece a la etapa disciplinaria; además se comparte con la Licenciatura en Ciencias Computacionales en etapa disciplinaria con carácter obligatorio.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Evaluar las soluciones numéricas, al interpretar los resultados obtenidos mediante diferentes métodos numéricos clásicos, para seleccionar el modelo numérico más apropiado en la solución de problemas que se presentan en la misma disciplina y en contextos interdisciplinarios, de forma crítica, reflexiva, independiente, creativa, honesta y responsable.

IV. EVIDENCIA DE APRENDIZAJE

En equipo, elabora una infografía integradora de análisis numérico que incluya el planteamiento del problema, la solución del problema mediante una o varias de las técnicas utilizadas en esta unidad de aprendizaje, la visualización de los resultados usando tablas o gráficas, el análisis de resultados, las conclusiones y las referencias bibliográficas. Se entregará en formato electrónico pdf en tiempo y forma, utilizando un lenguaje formal, apropiado y claro, en donde se muestre el dominio de los temas.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Licenciatura en Física

Desarrollar y utilizar programas computacionales aplicando los principios fundamentales de la física para la simulación o modelado de procesos físicos, con responsabilidad y honestidad.

Licenciatura en Ciencias Computacionales

Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia, vanguardia, innovación social.

VII. DESARROLLO POR UNIDADES

Unidad I. Aritmética computacional

Competencia

Analizar la forma de operación y las limitaciones de las computadoras, mediante el uso de aritmética de punto flotante, para evitar interpretaciones erróneas al momento de resolver un problema planteado, con una actitud crítica, reflexiva, paciente y honesta.

Contenido

- 1.1. Definición de modelo matemático, exactitud, precisión, inexactitud o sesgo, imprecisión o incertidumbre
- 1.2. Definición de número de máquina y número flotante decimal normalizado
- 1.3. Definición de truncamiento y de redondeo
- 1.4. Definición de error absoluto, error relativo, error relativo porcentual, error relativo porcentual aproximado
- 1.5. Definición de cifras significativas usando el valor de referencia o valor verdadero
- 1.6. Definición de cifras significativas usando una sucesión de valores aproximados
- 1.7. Definición de la unidad de medida palabra
- 1.8. El método de representación de magnitud con signo
- 1.9. Estándar IEEE 754, para los números de punto flotante
- 1.10. Estándar IEEE 754, tipos de precisión y regla de redondeo IEEE
- 1.11. Definición de épsilon de máquina
- 1.12. Definición de desbordamiento (*overflow*) y subdesbordamiento (*underflow*)
- 1.13. Aritmética con números de punto flotante
- 1.14. Cancelación de cifras significativas y pérdida de significancia
- 1.15. Definición de algoritmo, algoritmo estable, algoritmo condicionalmente estable
- 1.16. Definición de crecimiento del error lineal y exponencial
- 1.17. Definición de rapidez de convergencia usando una sucesión que converge a cero

Duración

08 horas

Unidad II. Ceros de ecuaciones no lineales

Competencia

Comparar los resultados de los métodos numéricos clásicos para la solución de búsqueda de raíces, mediante el análisis de errores y de rapidez de convergencia, para seleccionar el más apropiado acorde a las características de los problemas reales que surgen de las distintas ciencias, con actitud crítica, reflexiva, honesta y respetuosa.

Contenido

- 2.1. La equivalencia entre el problema de búsqueda de raíces y el problema de la búsqueda de un punto fijo
- 2.2. Teorema de existencia y unicidad del punto fijo

- 2.3. Funciones de iteración de punto fijo
- 2.4. Teorema del punto fijo
- 2.5. Método de la bisección
- 2.6. Teorema de la convergencia a la raíz del método de la bisección
- 2.7. Método de Newton-Raphson geoméricamente
- 2.8. Método de Newton-Raphson analíticamente usando el teorema de Taylor
- 2.9. Teorema de la convergencia a la raíz del método de Newton-Raphson
- 2.10. Método de la secante analítica usando la definición de límite
- 2.11. Método de la secante geoméricamente
- 2.12. Método de la falsa posición o regla falsa
- 2.13. Método de Müller
- 2.14. Teorema de la rapidez de convergencia lineal para métodos de iteración de punto fijo
- 2.15. Teorema de la rapidez de convergencia cuadrática para métodos de iteración de punto fijo
- 2.16. Aceleración de la convergencia: método Δ^2 de Aitken, método Δ^2 modificado de Aitken, método de Steffensen
- 2.17. El método de Newton-Raphson para raíces de multiplicidad m

Duración

09 horas

Unidad III. Sistemas de ecuaciones lineales

Competencia

Comparar las soluciones de sistemas de ecuaciones lineales, mediante el análisis de los errores vectoriales, para seleccionar el método numérico más apropiado al momento de resolver problemas reales que se presentan en las ciencias exactas, naturales e ingeniería con actitud crítica, propositiva, reflexiva y responsable.

Contenido

- 3.1. Métodos directos
 - 3.1.1. Eliminación gaussiana con sustitución hacia atrás y estrategias de pivoteo: parcial, parcial escalado y total
 - 3.1.2. Matrices especiales
 - 3.1.3. Factorización LU
 - 3.1.4. Factorización LDL^t
 - 3.1.5. Factorización LL^t
 - 3.1.6. Matrices de banda
 - 3.1.7. Factorización de matrices tridiagonales
 - 3.1.8. Matrices de permutación para la solución de sistemas $Ax=b$ utilizando la factorización LU
- 3.2. Métodos iterativos
 - 3.2.1. Método de Jacobi
 - 3.2.2. Método de Gauss-Seidel
 - 3.2.3. Métodos de iteración general

3.2.4. El teorema de convergencia

Duración

07 horas

Unidad IV Interpolación polinomial

Competencia

Comparar las soluciones de interpolación polinomial, mediante el análisis de los errores, para predecir el comportamiento de un conjunto de datos que se presentan en las ciencias exactas, naturales e ingeniería con actitud crítica, propositiva y honesta.

Contenido

- 4.1. Polinomio interpolante de Lagrange
- 4.2. Polinomio de Taylor
- 4.3. Teorema de la representación de una función mediante un polinomio interpolante de Lagrange
- 4.4. Método de Neville
- 4.5. Diferencias divididas progresivas de Newton
- 4.6. Definición de polinomios osculantes
- 4.7. Polinomio de Hermite
- 4.8. Trazadores cúbicos
- 4.9. Curvas de paramétricas
 - 4.9.1. Curvas de Bézier

Duración

08 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

Unidad I. Aritmética computacional

Práctica 1: Definición de exactitud, precisión, inexactitud o sesgo, imprecisión o incertidumbre

Duración

01 hora

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Define exactitud, precisión, inexactitud o sesgo, imprecisión o incertidumbre.
3. En sus palabras define de exactitud, precisión, inexactitud o sesgo, imprecisión o incertidumbre.
4. Documenta las definiciones y las entrega en formato digital para su evaluación.
5. Recibe retroalimentación docente.

Recursos de apoyo

- Bibliografía.
- Papel.
- Lápiz o pluma.

Práctica 2: Tipos de errores

Duración

01 hora

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Resuelve el ejercicio usando la definición de error absoluto.
3. Resuelve el ejercicio usando la definición de error relativo.
4. Resuelve el ejercicio usando la definición de error relativo porcentual.
5. Analiza los resultados.
6. Documenta los resultados y los entrega en formato digital para su evaluación.
7. Recibe retroalimentación docente.

Recursos de apoyo

- Calculadora.
- Papel.
- Lápiz o pluma.

Práctica 3: Cifras significativas

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Resuelve el ejercicio usando la definición de cifras significativas usando el valor de referencia o verdadero.
3. Resuelve el ejercicio usando la definición de cifras significativas usando una sucesión de valores aproximados.
4. Analiza los resultados.
5. Documenta los resultados y los entrega en formato digital para su evaluación.
6. Recibe retroalimentación docente.

Recursos de apoyo

- Calculadora.
- Papel.
- Lápiz o pluma.

Práctica 4: Punto flotante decimal normalizado

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Resuelve el ejercicio de magnitud con signo.
3. Resuelve el ejercicio de la representación de punto flotante decimal normalizado en una máquina hipotética base 10.
4. Resuelve el ejercicio de la representación de punto flotante decimal normalizado en una máquina hipotética base 5.
5. Resuelve el ejercicio de la representación de punto flotante decimal normalizado en una máquina hipotética base 2.
6. Resuelve el ejercicio de la representación de punto flotante decimal normalizado en una máquina hipotética de 7 bits.
7. Analiza los resultados.
8. Documenta los resultados y los entrega en formato digital para su evaluación.
9. Recibe retroalimentación docente.

Recursos de apoyo

- Calculadora .
- Papel.
- Lápiz o pluma.

Práctica 5: Cancelación de cifras significativas

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Resuelve el ejercicio de cancelación de cifras significativas usando la fórmula para encontrar ceros de polinomios cuadráticos. Determina el número de cifras significativas.
3. Resuelve el mismo ejercicio de cancelación de cifras significativas pero racionalizando el numerador para eliminar las restas. Determina el número de cifras significativas.
4. Resuelve el ejercicio de cancelación de cifras significativas evaluando un polinomio.
5. Resuelve el mismo ejercicio de cancelación de cifras significativas pero anidando el polinomio. Determina el número de cifras significativas.
6. Analiza los resultados.
7. Documenta los resultados y los entrega en formato digital para su evaluación.
8. Recibe retroalimentación docente.

Recursos de apoyo

- Calculadora.
- Papel.
- Lápiz o pluma.

Práctica 6: Crecimiento del error

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Resuelve el ejercicio sobre crecimiento lineal del error.
3. Resuelve el ejercicio sobre crecimiento exponencial del error.
4. Analiza los resultados.
5. Documenta los resultados y los entrega en formato digital para su evaluación.
6. Recibe retroalimentación docente.

Recursos de apoyo

- Calculadora.
- Papel.
- Lápiz o pluma.

Práctica 7: Rapidez de convergencia

Duración

01 hora

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Resuelve los ejercicios sobre rapidez de convergencia.
3. Analiza los resultados.
4. Documenta los resultados y los entrega en formato digital para su evaluación.
5. Recibe retroalimentación docente.

Recursos de apoyo

- Calculadora
- Papel
- Lápiz o pluma

Práctica 8: Foros sobre ética en la ciencia

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre los foros.
2. Revisa los vídeos. Por ejemplo, la carrera por explotar el poder de la computación cuántica en https://youtu.be/1_gJp2uAjO0?si=B2jwhV6WEir0yMzj, y la destacable historia detrás del algoritmo más importante de todos los tiempos (la FFT) en español en <https://youtube.com/watch?v=2Xkv-W9tOXU&feature=shares> o en inglés en <https://youtube.com/watch?v=nmgFG7PUHfo&feature=shares>
3. Participa en los foros.
4. El docente evalúa la participación.

Recursos de apoyo

- Computadora.
- Internet.
- Plataforma.

Unidad II: Ceros de ecuaciones no lineales

Práctica 9: Punto fijo

Duración

01 hora

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Resuelve el ejercicio sobre la existencia y unicidad del punto fijo.
3. Analiza los resultados.
4. Documenta los resultados y los entrega en formato digital para su evaluación.
5. Recibe retroalimentación docente.

Recursos de apoyo

- Calculadora.
- Papel.
- Lápiz o pluma.

Práctica 10: Funciones de iteración de punto fijo

Duración

01 hora

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Determina si las funciones de ejemplo cumplen las hipótesis del teorema del punto fijo.
3. Analiza los resultados.
4. Documenta los resultados y los entrega en formato digital para su evaluación.
5. Recibe retroalimentación docente.

Recursos de apoyo

- Calculadora.
- Papel.
- Lápiz o pluma.

Práctica 11: Método de Müller

Duración

01 hora

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Realiza una revisión bibliográfica del método de Müller.
3. Realiza un resumen del método de Müller.
4. Documenta el resumen y lo entrega en formato digital para su evaluación.
5. Recibe retroalimentación docente.

Recursos de apoyo

- Bibliografía.
- Papel.
- Lápiz o pluma.

Práctica 12: Rapidez de convergencia del método de la secante

Duración

01 hora

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Realiza una revisión bibliográfica de la rapidez de convergencia del método de la secante.
3. Documenta la demostración de la rapidez de convergencia del método de la secante y la entrega en formato digital para su evaluación.
4. Recibe retroalimentación docente.

Recursos de apoyo

- Bibliografía.
- Papel.
- Lápiz o pluma.

Práctica 13: Foros sobre ética en la ciencia

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre los foros.
2. Revisa los vídeos. Por ejemplo, Robo de criptomonedas en https://youtu.be/YxFGlllxyKE?si=SQw1c_X-rU9EZyID y One of the Greatest Feuds in the History of Science en <https://youtu.be/b1wNPkHUR10?si=Qrp2elhlc1ji1luo>.
3. Participa en los foros.
4. El docente evalúa la participación.

Recursos de apoyo

- Computadora.
- Internet.
- Plataforma.

Unidad III: Sistemas de ecuaciones lineales

Práctica 14: Método de eliminación gaussiana con sustitución hacia atrás

Duración

01 hora

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Resuelve el ejercicio usando el método de eliminación gaussiana con sustitución hacia atrás.
3. Analiza los resultados.
4. Documenta los resultados y los entrega en formato digital para su evaluación.
5. Recibe retroalimentación docente.

Recursos de apoyo

- Calculadora.
- Papel.
- Lápiz o pluma.

Práctica 15: Método de eliminación gaussiana con sustitución hacia atrás y pivoteo parcial

Duración

01 hora

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Resuelve el ejercicio usando el método de eliminación gaussiana con sustitución hacia atrás y pivoteo parcial.
3. Analiza los resultados.
4. Documenta los resultados y los entrega en formato digital para su evaluación.
5. Recibe retroalimentación docente.

Recursos de apoyo

- Calculadora.

- Papel.
- Lápiz o pluma.

Práctica 16: Método de eliminación gaussiana con sustitución hacia atrás y pivoteo parcial escalado

Duración

01 hora

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Resuelve el ejercicio usando el método de eliminación gaussiana con sustitución hacia atrás y pivoteo parcial escalado.
3. Analiza los resultados.
4. Documenta los resultados y los entrega en formato digital para su evaluación.
5. Recibe retroalimentación docente.

Recursos de apoyo

- Calculadora.
- Papel.
- Lápiz o pluma.

Práctica 17: Factorización LU

Duración

01 hora

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Resuelve el ejercicio usando el método de factorización LU.
3. Analiza los resultados.
4. Documenta los resultados y los entrega en formato digital para su evaluación.
5. Recibe retroalimentación docente.

Recursos de apoyo

- Calculadora.
- Papel.
- Lápiz o pluma.

Práctica 18: Foros sobre avances en la ciencia

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre los foros.
2. Revisa los vídeos. Por ejemplo, ¿Qué es y cómo funciona la COMPUTACIÓN CUÁNTICA? en <https://youtu.be/YpYuBEzfRIM?si=EwkpoFd8HoSxYQD2> y Cómo se programa un ORDENADOR CUÁNTICO (bien explicado) en <https://youtu.be/eRIQdW1lgJE?si=FIVcXUq1hY4j5xz2>
3. Participa en los foros.
4. El docente evalúa la participación.

Recursos de apoyo

- Computadora.
- Internet.
- Plataforma.

Unidad IV: Interpolación polinomial

Práctica 19: El polinomio interpolante de Lagrange

Duración

01 hora

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Construye el polinomio interpolante de Lagrange solicitado.
3. Construye el polinomio interpolante de Taylor solicitado.
4. Analiza los resultados.
5. Documenta los resultados y los entrega en formato digital para su evaluación.
6. Recibe retroalimentación docente.

Recursos de apoyo

- Calculadora.
- Papel.
- Lápiz o pluma.

Práctica 20: Trazadores cúbicos

Duración

01 hora

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Realiza una revisión bibliográfica sobre trazadores cúbicos.
3. Define un trazador cúbico.
4. Documenta la definición y la entrega en formato digital para su evaluación.
5. Recibe retroalimentación docente.

Recursos de apoyo

- Calculadora.
- Papel.
- Lápiz o pluma.

Práctica 21: Curvas paramétricas

Duración

01 hora

Procedimiento

1. Atiende las orientaciones docentes sobre el taller, si es individual o en equipo y la duración.
2. Realiza una revisión bibliográfica sobre curvas paramétricas.
3. Define las curvas paramétricas cúbicas.
4. Documenta la definición y la entrega en formato digital para su evaluación.
5. Recibe retroalimentación docente.

Recursos de apoyo

- Calculadora.
- Papel.
- Lápiz o pluma.

Práctica 22: Foros sobre ética en la ciencia

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre los foros.
2. Revisa los vídeos. Por ejemplo, La MAYOR ESTAFA de la CIENCIA MÉDICA que ha EXISTIDO / CASO Elizabeth Holmes en https://youtu.be/9ZqggCR_smU?si=rzKXBOpWelvxfMh
3. Participa en los foros.
4. El docente evalúa la participación.

Recursos de apoyo

- Computadora.
- Internet.
- Plataforma.

Práctica 23: Infografía

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre los foros.
2. Forma el equipo de trabajo.
3. Realiza una revisión bibliográfica.
4. Plantea un problema a resolver usando alguno o algunos de los métodos numéricos vistos en esta unidad de aprendizaje y la entrega en formato digital para su evaluación.
5. Recibe retroalimentación docente.

Recursos de apoyo

- Bibliografía .
- Papel.
- Lápiz o pluma.

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I: Aritmética computacional

Práctica 1: Introducción al lenguaje de programación y los tipos de variables

Duración

01 hora

Procedimiento

1. Atiende las orientaciones docentes sobre introducción al lenguaje de programación y los tipos de variables.
2. Verifica el tipo de variable utilizada.
3. Operaciones con variables.
4. Documenta las operaciones resultantes y las entrega en formato digital para su evaluación.
5. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora.
- Software como MATLAB o Python con OpenCV.

Práctica 2: Gráficas 2D

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre cómo definir una función.
2. Atiende las orientaciones docentes sobre generar y editar una gráfica 2D, incluyendo la incorporación de símbolos griegos.
3. Documenta las ediciones realizadas a una gráfica 2D en coordenadas cartesianas y las entrega en formato digital para su evaluación.
4. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora.
- Software como MATLAB o Python con OpenCV.

Práctica 3: Tablas

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre generar tablas y darles un formato.
2. Documenta las tablas generadas en el formato solicitado y las entrega en formato digital para su evaluación.
3. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora.
- Software como MATLAB o Python con OpenCV.

Práctica 4: Aritmética con números de punto flotante en la computadora, error lineal y exponencial

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre las operaciones aritméticas que provocan pérdida de significancia a realizar en el software seleccionado.
2. Atiende las orientaciones docentes sobre identificar error lineal y exponencial con el software seleccionado.
3. Documenta el resultado de las operaciones aritméticas agregando explicaciones de la generación de la pérdida de significancia.

4. Documenta las tablas y las gráficas de las iteraciones para las situaciones hipotéticas de error lineal y error exponencial, apoyándose del uso de tablas trabajadas en la práctica previa.
5. Entrega lo planteado en los últimos tres incisos anteriores en formato digital para su revisión.
6. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora.
- Software como MATLAB o Python con OpenCV.

Unidad II: Ceros de ecuaciones no lineales

Práctica 5: Funciones de iteración de punto fijo

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre funciones de iteración de punto fijo.
2. Documenta el resultado generando una tabla de cinco funciones de iteración de punto fijo para el problema planteado y las entrega en formato digital para su evaluación.
3. Documenta la gráfica de la función del problema planteado y la entrega en formato digital para su evaluación.
4. Documenta la gráfica de las funciones de iteración de punto fijo y la entrega en formato digital para su evaluación.
5. Documenta la gráfica de la primera derivada de las funciones de iteración de punto fijo y la entrega en formato digital para su evaluación.
6. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora.
- Software como MATLAB o Python con OpenCV.

Práctica 6: Método de la Bisección

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre la implementación del método de la bisección reescribiendo las variables.
2. Atiende las orientaciones docentes sobre la implementación del método de la bisección guardando las variables en vectores.
3. Encuentra raíces de funciones no lineales con el método de la bisección.
4. Documenta el resultado generando una tabla y la entrega en formato digital para su evaluación.
5. Documenta la gráfica de la función del problema planteado y la entrega en formato digital para su evaluación.
6. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora.
- Software como MATLAB o Python con OpenCV.

Práctica 7: Método de Newton-Raphson y método de la secante

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre la implementación del método de Newton-Raphson.
2. Atiende las orientaciones docentes sobre la implementación del método de la secante.
3. Encuentra raíces de funciones no lineales con el método de Newton-Raphson.
4. Encuentra raíces de funciones no lineales con el método de la secante.
5. Documenta el resultado del método de Newton-Raphson generando una tabla y la entrega en formato digital para su evaluación.
6. Documenta el resultado del método de la secante generando una tabla y la entrega en formato digital para su evaluación.
7. Documenta la gráfica de la función del problema planteado y la entrega en formato digital para su evaluación.
8. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora
- Software como MATLAB o Python con OpenCV

Práctica 8: Método de la falsa posición y método Müller

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre la implementación del método de la falsa posición.
2. Atiende las orientaciones docentes sobre la implementación del método de Müller.
3. Encuentra raíces de funciones no lineales con el método de la falsa posición.
4. Encuentra raíces de funciones no lineales con el método de Müller.
5. Documenta el resultado del método de la falsa posición generando una tabla y la entrega en formato digital para su evaluación.
6. Documenta el resultado del método de Müller generando una tabla y la entrega en formato digital para su evaluación.
7. Documenta la gráfica de la función del problema planteado y la entrega en formato digital para su evaluación.
8. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora.
- Software como MATLAB o Python con OpenCV.

Práctica 9: Método Δ^2 modificado de Aitken y método de Steffensen

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre la implementación del método Δ^2 modificado de Aitken.
2. Atiende las orientaciones docentes sobre la implementación del método de Steffensen.
3. Encuentra raíces de funciones no lineales con el método Δ^2 modificado de Aitken.
4. Encuentra raíces de funciones no lineales con el método de Steffensen.
5. Documenta el resultado del método Δ^2 modificado de Aitken generando una tabla y la entrega en formato digital para su evaluación.
6. Documenta el resultado del método de Steffensen generando una tabla y la entrega en formato digital para su evaluación.
7. Documenta la gráfica de la función del problema planteado y la entrega en formato digital para su evaluación.
8. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora.

- Software como MATLAB o Python con OpenCV.

Unidad III: Sistemas de ecuaciones lineales

Práctica 10: Factorización LU

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el método de factorización LU.
2. Resuelve sistemas de ecuaciones lineales con el método de factorización LU.
3. Documenta el resultado y lo entrega en formato digital para su evaluación.
4. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora.
- Software como MATLAB o Python con OpenCV.

Práctica 11: Solución de sistemas de ecuaciones lineales con pivoteo

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el método de factorización LU con pivoteo.
2. Resuelve sistemas de ecuaciones lineales con el método de factorización LU y pivoteo
3. Documenta el resultado y lo entrega en formato digital para su evaluación.
4. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora
- Software como MATLAB o Python con OpenCV

Práctica 12: Solución de sistemas de ecuaciones lineales con métodos iterativos

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el método de Jacobi y el método de Gauss-Seidel.
2. Resuelve sistemas de ecuaciones lineales con el método de Jacobi y el método de Gauss-Seidel.
3. Documenta el resultado y lo entrega en formato digital para su evaluación.
4. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora.
- Software como MATLAB o Python con OpenCV.

Unidad IV: Interpolación polinomial

Práctica 13: El polinomio interpolante de Lagrange y el polinomio de Taylor

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el polinomio interpolante de Lagrange.
2. Atiende las orientaciones docentes sobre el polinomio interpolante de Taylor.
3. Interpola un conjunto de datos con el polinomio interpolante de Lagrange.
4. Documenta el resultado graficando los datos y el polinomio interpolante de Lagrange y lo entrega en formato digital para su evaluación.
5. Documenta el resultado graficando el dato y el polinomio interpolante de Taylor y los entrega en formato digital para su evaluación.
6. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora.
- Software como MATLAB o Python con OpenCV.

Práctica 14: El método de Neville

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el método de Neville.
2. Interpola un conjunto de datos usando el método de Neville.
3. Documenta el resultado graficando los datos y el resultado del método de Neville y los entrega en formato digital para su evaluación.
4. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora.
- Software como MATLAB o Python con OpenCV.

Práctica 15: Trazador cúbico

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre trazadores cúbicos.
2. Interpola un conjunto de datos usando un trazador cúbico.
3. Documenta el resultado graficando los datos y el resultado del trazador cúbico y los entrega en formato digital para su evaluación.
4. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora.
- Software como MATLAB o Python con OpenCV.

Práctica 16: Curvas de Bézier

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre las curvas de Bézier.
2. Manipula una sección de una gráfica usando curvas de Bézier.
3. Documenta el resultado graficando los datos y el resultado de las curvas de Bézier y los entrega en formato digital para su evaluación.
4. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora.
- Software como MATLAB o Python con OpenCV.

Práctica 17: Proyecto final**Duración**

01 hora

Procedimiento

1. Atiende las orientaciones docentes sobre el proyecto final.
2. Plantea un problema a resolver en equipos.
3. Resuelve el problema planteado usando alguna de las metodologías vistas en esta Unidad de Aprendizaje.
4. Documenta el resultado en una infografía y la entrega en formato digital para su evaluación.
5. Recibe retroalimentación docente.

Recursos de apoyo

- Computadora..
- Software como MATLAB o Python con OpenCV
- Software como canva.

XI. MÉTODO DE TRABAJO**Encuadre**

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Aprendizaje basado en preguntas
- Foros
- Método de proyectos

Estrategia de aprendizaje

- Investigación documental
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

- Evaluaciones parciales	30%
- Prácticas de laboratorio	30%
- Talleres	20%
- Foros	05%
- Infografía integradora de análisis numérico	15%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Burden, A. M. & Faires, D. J. (2011). *Análisis numérico* (9ª ed.). Cengage Learning. [clásica]
- Burden, A. M., Faires, D. J. & Burden, R. L. (2017). *Análisis numérico* (10ª ed.). Cengage Learning. [clásica]
- Chapra, S. C. & Canale, R. P. (2015). *Métodos numéricos para ingenieros* (7ª ed.), McGraw-Hill. En Biblioteca física. [clásica]
- Chapra, S. C. (2023). *Métodos numéricos aplicados con MATLAB® para ingenieros y científicos* (5ª ed.), McGraw-Hill Interamericana. Disponible en físico y digital.
- Cheney, W. & Kincaid, D. (2011). *Métodos numéricos y computación* (6ª ed.). Cengage Learning. [clásica]
- Gerald, C. F., Wheatley, P. O. & del Valle Sotelo, J. C., (2000). *Análisis numérico con aplicaciones*, Pearson Educación. [clásica]
- Gilat, A. & Subramaniam, V. (2014). *Numerical methods for engineers and scientists: an introduction with applications using MATLAB* (3ª ed.). Wiley. [clásica]

Heister, T. & Rebholz, L. G. (2015). *Scientific Computing: For Scientists and Engineers*. De Gruyter. [clásica]

Lindfield, G. & Penny, J. (2012). *Numerical Methods: Using MATLAB* (3ª ed.). Academic Press. [clásica]

Sauer, T. (2013). *Análisis numérico* (2ª ed.), Pearson. [clásica]

Sauer, T. (2024). *Análisis numérico* (3ª ed.), Pearson Educación de México.

Recursos de acceso libre o complementarios externos

Autarkaw. (s. f.). *Online Course in Numerical Methods*. <http://www.autarkaw.com/books/h>

Massachusetts Institute of Technology. (2012). *Introduction to Numerical Analysis* (Curso 18.330). MIT OpenCourseWare. <https://ocw.mit.edu/courses/18-330-introduction-to-numerical-analysis-spring-2012/>

MathWorks. (s. f.). MathWorks – Creador de MATLAB y Simulink. <https://la.mathworks.com/>

XII. PERFIL DOCENTE

Licenciatura en Matemáticas Aplicadas, Física o Ciencias de la Computación, preferentemente con posgrado, y experiencia en el desarrollo e implementación de métodos numéricos (búsqueda de raíces, sistemas de ecuaciones lineales e interpolación polinomial). Debe dominar lenguajes de programación orientados al cómputo científico, como MATLAB o Python, y contar con competencias pedagógicas para promover el aprendizaje activo, la resolución de problemas y el pensamiento crítico, impulsando en el estudiantado la creatividad, la responsabilidad y la honestidad académica.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Minería de Datos
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 03 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Disciplinaria
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Everardo Gutiérrez López Antonio de Jesús García Chavez José Ángel González Fraga
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

El propósito de esta asignatura es que el estudiantado comprenda, aplique y evalúe los principios, técnicas y metodologías fundamentales de la Minería de Datos, como área de estudio de las Ciencias Computacionales, para la solución de problemas reales mediante el análisis riguroso, modelado supervisado y no supervisado, experimentación científica y responsabilidad ética, desarrollando habilidades para procesar, analizar y transformar datos en conocimiento útil dentro de contextos profesionales y de investigación. Se imparte en la etapa disciplinaria, es de carácter obligatorio y pertenece al área de conocimiento de Tratamiento de Información.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Construir procesos completos de minería de datos, mediante la aplicación de fundamentos estadísticos, algoritmos avanzados y metodologías científicas para preparar, modelar, evaluar visualizar e interpretar datos provenientes de problemas prácticos, con rigor analítico, eficiencia computacional y responsabilidad ética en el manejo de la información.

IV. EVIDENCIA DE APRENDIZAJE

Construye un proceso completo de minería de datos como proyecto integrador final que utilice las técnicas fundamentales involucradas en el análisis computacional de datos para solucionar alguna de las problemáticas prácticas especificadas por el docente. Presenta y reporta los resultados obtenidos tanto de forma oral como escrita, discutiendo de manera crítica los resultados, limitaciones y riesgos éticos.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar y evaluar soluciones de cómputo inteligente aplicando principios de inteligencia artificial, minería de datos, e integración de herramientas computacionales, con el propósito de apoyar la toma de decisiones en situaciones complejas propias de contextos científicos, sociales, industriales o tecnológicos, actuando con responsabilidad ética, pensamiento crítico, y colaboración interdisciplinaria.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Compromiso con la Responsabilidad Social, Excelencia, Vanguardia e Innovación Social

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos científicos de la minería de datos

Competencia

Aplicar los principios matemáticos, estadísticos y metodológicos que sustentan el proceso científico de minería de datos, mediante la selección de las tareas de análisis apropiadas para la formulación correcta de problemas de aplicación, con rigor analítico, espíritu autocrítico y responsabilidad ética en el manejo de la información.

Contenido

- 1.1. Introducción y panorama general
 - 1.1.1. Conceptos básicos: datos, información, conocimiento, patrones.
 - 1.1.2. Relación con aprendizaje automático, estadística e inteligencia artificial.
 - 1.1.3. Tipos de tareas: supervisada, no supervisada, semi-supervisada y por refuerzo.
- 1.2. Matemáticas y estadística para minería de datos
 - 1.2.1. Probabilidad básica y variables aleatorias.
 - 1.2.2. Estadística descriptiva y exploratoria.
 - 1.2.3. Correlaciones, dependencias y medidas de asociación.
- 1.3. Proceso científico para minería de datos
 - 1.3.1. Formulación de problemas.
 - 1.3.2. Metodologías CRISP–DM, KDD y variantes.
 - 1.3.3. Validación, replicabilidad y experimentación científica.

Duración

6 horas

Unidad II. Preparación, transformación y análisis de datos

Competencia

Reorganizar conjuntos de datos mediante la aplicación de técnicas de limpieza, integración, escalamiento y selección de características, para garantizar su calidad, coherencia y adecuación en el modelado analítico de problemas y su eficiencia computacional, con creatividad en la formulación de soluciones y responsabilidad ética en el manejo de la información.

Contenido

- 2.1. Preprocesamiento
 - 2.1.1. Limpieza: manejo de valores faltantes, ruido y valores atípicos.
 - 2.1.2. Normalización, estandarización y escalamiento.
 - 2.1.3. Integración de múltiples fuentes y detección de inconsistencias.
- 2.2. Ingeniería de atributos
 - 2.2.1. Codificación de variables categóricas.
 - 2.2.2. Selección de características: filtros, wrappers y métodos embebidos.
 - 2.2.3. Extracción de características con métodos estadísticos como PCA y LDA.

- 2.3. Manejo de grandes volúmenes de datos
 - 2.3.1. Conceptos de escalabilidad y eficiencia.
 - 2.3.2. Técnicas básicas de muestreo y particionamiento.
 - 2.3.3. Introducción a entornos distribuidos.

Duración

8 horas

Unidad III. Algoritmos y técnicas fundamentales de minería de datos

Competencia

Analizar algoritmos supervisados, no supervisados y de asociación, mediante la evaluación de su comportamiento, pertinencia y desempeño para la selección del algoritmo adecuado a la problemática planteada acorde a criterios computacionales y estadísticos, con rigor analítico, espíritu autocrítico y responsabilidad ética en el manejo de la información.

Contenido

- 3.1. Minería supervisada
 - 3.1.1. Clasificación: k-NN, árboles de decisión, Naive Bayes, SVM.
 - 3.1.2. Métodos de ensamble: bagging, boosting, random forests.
 - 3.1.3. Evaluación de clasificadores: matrices de confusión, ROC, AUC, precisión-recall.
- 3.2. Minería no supervisada
 - 3.2.1. Conglomerados: k-means, k-medoides, DBSCAN, conglomerado jerárquico.
 - 3.2.2. Métricas de evaluación: silhouette, cohesión, separación.
 - 3.2.3. Reducción de dimensionalidad y proyecciones.
- 3.3. Reglas de asociación
 - 3.3.1. Soporte, confianza y lift.
 - 3.3.2. Algoritmos a priori y FP-Growth.
 - 3.3.3. Interpretación y filtrado de reglas.
- 3.4. Series de tiempo y datos secuenciales
 - 3.4.1. Patrones secuenciales.
 - 3.4.2. Modelos simples para predicción: ARIMA y variantes conceptuales.

Duración

10 horas

Unidad IV. Implementación, evaluación científica y aplicaciones

Competencia

Evaluar modelos de minería de datos mediante la aplicación de metodologías científicas de validación, para la interpretación de resultados de forma rigurosa en la aplicación a soluciones completas de problemas reales considerando implicaciones éticas, con un espíritu autocrítico y rigor analítico.

Contenido

- 4.1. Validación y experimentación
 - 4.1.1. Validación cruzada, bootstrapping y particionado estratificado.
 - 4.1.2. Evitar sobreajuste y subajuste.
 - 4.1.3. Comparación estadística de modelos: pruebas t, Wilcoxon, Friedman.
- 4.2. Interpretabilidad y explicabilidad
 - 4.2.1. Importancia de variables.
 - 4.2.2. Explicación local y global.
 - 4.2.3. Consideraciones sobre modelos caja negra.
- 4.3. Ética, sesgos y responsabilidad
 - 4.3.1. Sesgos en datos y modelos.
 - 4.3.2. Implicaciones sociales, científicas y legales.
 - 4.3.3. Transparencia, reproducibilidad y documentación.
- 4.4. Aplicaciones prácticas
 - 4.4.1. Análisis de comportamiento.
 - 4.4.2. Detección de anomalías.
 - 4.4.3. Segmentación, recomendación y predicción.

Duración

8 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Fundamentos científicos de la minería de datos

Práctica 1: Formulación y Análisis Científico de Problemas de Minería de Datos

Duración: 5 horas

Procedimiento:

1. Atiende las instrucciones de su docente.
2. Realiza previo a la práctica una lectura sobre el proceso científico en minería de datos y los principios matemáticos y estadísticos básicos (probabilidad, distribuciones, etc.).
3. Selecciona un conjunto de datos (como el conjunto de Iris o Titanic) y define un problema práctico de minería de datos (clasificación, agrupamiento, etc.).
4. Elige las técnicas más apropiadas para resolver el problema planteado, discutiendo el criterio de selección.
5. Utiliza Python, Weka, o equivalente, para aplicar las técnicas seleccionadas al conjunto de datos, y realiza un análisis exploratorio de los datos (EDA).
6. Documenta el proceso y los resultados, discutiendo las implicaciones científicas de las decisiones tomadas y los resultados obtenidos.
7. Entrega reporte final y recibe retroalimentación de su docente.

Recursos de apoyo:

- Jupyter Notebook, Weka o equivalente.

- Documentación sobre el proceso científico en minería de datos.
- Dataset de ejemplo (por ejemplo Iris, Titanic, etc.).

Práctica 2: Aplicación de Técnicas Matemáticas y Estadísticas en Minería de Datos

Duración: 5 horas

Procedimiento:

1. Atiende las instrucciones de su docente.
2. Previo a la práctica repasa temas clave como distribuciones de probabilidad, regresión, correlación y pruebas estadísticas.
3. Utiliza un conjunto de datos (como Iris o ventas) y realiza un análisis exploratorio utilizando Python (pandas, numpy) o equivalente.
4. Calcula la media, desviación estándar, correlación entre variables y realiza pruebas de hipótesis (t-test, ANOVA).
5. Aplica modelos estadísticos simples y evalúa su desempeño.
6. Interpreta los resultados y discute las implicaciones de los análisis estadísticos en el proceso de minería de datos.
7. Entrega reporte final y recibe retroalimentación de su docente.

Recursos de apoyo:

- Jupyter Notebook o equivalente.
- Documentación de pandas, scikit-learn.
- Dataset (por ejemplo Iris, Ventas, etc.).

Unidad II. Preparación, transformación y análisis de datos

Práctica 3: Preparación, transformación y gestión de datos

Duración: 6 horas

Procedimiento:

1. Atiende las instrucciones de su docente.
2. Utiliza librerías (por ejemplo pandas) para cargar un conjunto de datos y realizar un análisis exploratorio (EDA) básico.
3. Identifica y trata valores faltantes, atípicos o erróneos, aplicando métodos de imputación o eliminación de valores.
4. Normaliza y estandariza variables utilizando librerías como scikit-learn y compara los resultados antes y después de la transformación.
5. Integra datos provenientes de múltiples fuentes (por ejemplo, archivos CSV, bases de datos SQL) y resuelve inconsistencias entre ellos.
6. Documenta el proceso y justifica las decisiones tomadas en cada paso del preprocesamiento.
7. Entrega reporte final y recibe retroalimentación de su docente.

Recursos de apoyo:

- Jupyter Notebook o equivalente.
- Documentación de librerías pandas y scikit-learn, o equivalentes.
- Dataset abierto (por ejemplo Titanic, Ventas, etc.).

Práctica 4: Selección de Características y Reducción de Dimensionalidad

Duración: 8 horas

Procedimiento:

1. Atiende las instrucciones de su docente.
2. Aplica técnicas de filtrado, envoltorio y embebido para identificar las características más relevantes en un conjunto de datos (por ejemplo, utilizando sklearn).
3. Implementa técnicas como PCA (Análisis de Componentes Principales) y LDA (Análisis Discriminante Lineal) para reducir la dimensionalidad del conjunto de datos.
4. Evalúa el impacto de la reducción de dimensionalidad en el rendimiento del modelo utilizando métricas de rendimiento (precisión, recall, etc.).
5. Visualiza los datos antes y después de la reducción de dimensionalidad en gráficos 2D o 3D.
6. Escribe un informe sobre cómo la reducción de dimensionalidad afecta el rendimiento y la interpretabilidad del modelo.
7. Entrega reporte final y recibe retroalimentación de su docente.

Recursos de apoyo:

- Jupyter Notebook o equivalente.
- Documentación de librerías scikit-learn y matplotlib, o equivalente.
- Dataset para clasificación (por ejemplo Iris o Wine).

Unidad III. Algoritmos y técnicas fundamentales de minería de datos**Práctica 5: Clasificación con k-NN y Árboles de Decisión**

Duración: 8 horas

Procedimiento:

1. Atiende las instrucciones de su docente.
2. Utiliza un conjunto de datos (por ejemplo Iris o Titanic) y lo divide en conjunto de entrenamiento y conjunto de prueba.
3. Implementa k-NN utilizando scikit-learn, o equivalente, y evalúa el rendimiento utilizando métricas como precisión, recall, matriz de confusión.
4. Implementa un árbol de decisión utilizando el algoritmo ID3 o C4.5 y compara el rendimiento con k-NN.
5. Implementa un modelo de Random Forests y compara su rendimiento con los modelos anteriores.
6. Compara los resultados de los modelos, analizando sus ventajas y desventajas.

7. Entrega reporte final y recibe retroalimentación de su docente.

Recursos de apoyo:

- Jupyter Notebook o equivalente.
- Documentación de librería scikit-learn o equivalente.
- Dataset de clasificación (por ejemplo Iris, Titanic, etc.).

Práctica 6: Agrupamiento de Datos con k-Means y DBSCAN

Duración: 6 horas

Procedimiento:

1. Atiende las instrucciones de su docente.
2. Elige un conjunto de datos adecuado para agrupamiento (como Iris o un dataset de clientes).
3. Implementa el algoritmo K-Means utilizando la librería scikit-learn, o equivalente, y visualiza los clusters resultantes.
4. Utiliza métricas como la silueta para evaluar la calidad de los clusters obtenidos.
5. Implementa el algoritmo DBSCAN y compara los resultados con K-Means, especialmente en datos con ruido.
6. Evalúa cuál de los dos algoritmos es más adecuado para el conjunto de datos y justifica su elección.
7. Entrega reporte final y recibe retroalimentación de su docente.

Recursos de apoyo:

- Jupyter Notebook o equivalente.
- Documentación de librería scikit-learn, o equivalente.
- Dataset para agrupamiento (por ejemplo Iris, Clientes, etc.).

Unidad IV. Implementación, evaluación científica y aplicaciones

Práctica 7: Evaluación de modelos con validación cruzada y comparación estadística

Duración: 10 horas

Procedimiento:

1. Atiende las instrucciones de su docente.
2. Selecciona dos o más algoritmos de clasificación (por ejemplo, k-NN y Árboles de Decisión) y los entrena sobre el mismo conjunto de datos.
3. Implementa validación cruzada k-fold para cada modelo y calcula las métricas de rendimiento (precisión, recall, AUC, etc.).
4. Realiza una comparación estadística entre los modelos utilizando pruebas t, Wilcoxon o Friedman para evaluar si las diferencias en el rendimiento son significativas.
5. Interpreta los resultados de las pruebas estadísticas y discute cuál es el modelo más adecuado.

6. Elabora un informe detallado con las métricas de evaluación y comparaciones entre los modelos.
7. Entrega reporte final y recibe retroalimentación de su docente.

Recursos de apoyo:

- Jupyter Notebook o equivalente.
- Documentación de librerías scikit-learn y scipy (o equivalentes para pruebas estadísticas).
- Dataset de clasificación.

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Aprendizaje basado en casos
- Aprendizaje basado en preguntas
- Aprendizaje basado en problemas
- Aprendizaje basado en proyectos
- Aprendizaje colaborativo
- Discusión guiada
- Instrucción directa o guiada
- Proyectos integradores o interdisciplinarios
- Simulación de casos reales

Estrategia de aprendizaje

- Elaboración de organizadores gráficos (infografías, mapas mentales, mapas conceptuales)
- Investigación acción participativa
- Investigación documental
- Investigación empírica
- Resolución de problemas
- Toma de apuntes estratégicos
- Trabajo colaborativo

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	30%
– Prácticas laboratorio	30%
– Proyecto integrador final	30%
– Exposición oral	10%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Bhargava, N., Bhargava, R., Rathore, P. S., y Agrawal, R. (2021). Artificial Intelligence and Data Mining Approaches in Security Frameworks. John Wiley & Sons.

Géron, A. (2023). Hands-on Machine learning with Scikit-Learn, Keras, and TensorFlow : concepts, tools, and techniques to build intelligent systems (3rd ed.). O'Reilly Media.

Han, J., Pei, J., Tong, H. (2022). Data Mining: Concepts and Techniques (4th ed.). Morgan Kaufmann.

Leskovec, J., Rajaraman, A., & Ullman, J. D. (2020). Mining of massive datasets (3rd ed.). Cambridge University Press.

Tan, P. N., Steinbach, M., Karpatne, A., Kumar, V. (2018). Introduction to data mining. Pearson Education (2nd ed.). Pearson. [clásica]

Witten, I. H., Frank, E., Hall, M. A., Pal, C. J. (2016). Data Mining: Practical Machine Learning Tools and Techniques (4th ed.). Morgan Kaufmann. [clásica]

Recursos de acceso libre o complementarios externos

ACM Special Interest Group on Knowledge Discovery in Data.
<https://www.acm.org/special-interest-groups/sigs/sigkdd>

Aggarwal, C. C. (2015). Data Mining: The Textbook. Springer. [clásica]

Bishop, C. M. (2006). Pattern recognition and machine learning. Springer.
<https://www.microsoft.com/en-us/research/people/cmbishop/>

Chen, X., & Yang, T. (2024). *Advanced Data Mining and Applications: Techniques, Algorithms, and Challenges*. Springer.

Géron, A. (2022). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow* (3rd ed.). O'Reilly Media.

Google Colab. (n.d.). Google Colaboratory. Recuperado de <https://colab.research.google.com/>

Kaggle. (n.d.). Kaggle Datasets. Recuperado de <https://www.kaggle.com/datasets>

López, V., Sánchez, J. S. (2017). *Minería de datos: Conceptos, métodos y aplicaciones*. Editorial Reverté. [clásica]

Piatetsky-Shapiro, G. (2013). Knowledge Discovery from Data: An Overview. *ACM Computing Surveys*, 45(4), 1-49. [clásica]

Scikit-learn Documentation. (n.d.). Scikit-learn: Machine Learning in Python. Recuperado de <https://scikit-learn.org/stable/>

UCI Machine Learning Repository. (n.d.). UCI Repository of Machine Learning Databases. Recuperado de <https://archive.ics.uci.edu/ml/index.php>

Zhao, F., & Wei, J. (2023). *Data Mining for Knowledge Discovery: Principles, Algorithms, and Applications*. Elsevier.

XII. PERFIL DOCENTE

Licenciatura en Ciencias Computacionales o áreas afines, preferentemente con estudios de posgrado en el área de las Minería de Datos, Análisis de Datos, Ciencia de Datos, o áreas afines. Experiencia profesional en docencia universitaria de al menos 2 años. Se requiere experiencia en docencia universitaria, dominio de los fundamentos de la Minería de Datos, experiencia de programación, así como integrar de manera ética y responsable el uso de tecnologías emergentes para manejo de datos. Asimismo, deberá tener la capacidad de explicar conceptos con rigor y claridad, diseñar actividades teóricas y prácticas, y promover disciplina, honestidad académica, trabajo colaborativo y actualización continua.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Fundamentos de Inteligencia Artificial
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 03 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos: 07
Etapas de formación:	Disciplinaria
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	José Ángel González Fraga Everardo Gutiérrez López Antonio de Jesús García Chávez
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La unidad de aprendizaje Fundamentos de Inteligencia Artificial tiene como propósito que el estudiantado analice los principios teóricos, los modelos computacionales y los métodos clásicos que sustentan el funcionamiento de los sistemas inteligentes. Para ello, se abordan conceptos esenciales relacionados con agentes, técnicas de búsqueda, representación del conocimiento y razonamiento lógico, con el fin de que cada estudiante comprenda, interprete y modele soluciones a problemas computacionales de diversa naturaleza. Su utilidad radica en que desarrolla habilidades para formular problemas, seleccionar enfoques de solución, aplicar técnicas básicas de inteligencia artificial y evaluar sus alcances y limitaciones. Asimismo, promueve actitudes críticas, éticas y responsables que favorecen el uso adecuado de la inteligencia artificial y la comprensión de su impacto en entornos científicos, educativos e industriales. Esta unidad es de carácter obligatorio, forma parte de la etapa disciplinaria y se integra al área de Interacción Humano-Computadora del programa de Licenciatura en Ciencias Computacionales.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Analizar problemas computacionales susceptibles de resolverse mediante el uso de agentes inteligentes, técnicas de búsqueda, representación del conocimiento y razonamiento lógico, empleando métodos formales y herramientas básicas de la inteligencia artificial, con el propósito de comprender y modelar soluciones adecuadas, actuando con responsabilidad, ética y pensamiento crítico.

IV. EVIDENCIA DE APRENDIZAJE

Proyecto integrador

Desarrolla un proyecto integrador que modele y resuelva un problema computacional mediante técnicas básicas de inteligencia artificial.

El proyecto deberá incluir:

- Planteamiento formal del problema
- Caracterización del entorno
- Definición y justificación del agente inteligente
- Implementación de una o varias de las técnicas abordadas en la unidad de aprendizaje (búsqueda no informada o informada, búsqueda local, representación del conocimiento, razonamiento lógico)
- Análisis crítico de los resultados obtenidos

La evidencia deberá presentarse en un reporte técnico y una presentación o infografía elaborada en herramientas como PowerPoint, Canva o Genially, mostrando el funcionamiento de la solución, las decisiones metodológicas y sus implicaciones éticas, con un lenguaje formal, claro y sustentado en referencias bibliográficas.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar y evaluar soluciones de cómputo inteligente aplicando principios de inteligencia artificial, minería de datos, e integración de herramientas computacionales, con el propósito de apoyar la toma de decisiones en situaciones complejas propias de contextos científicos, sociales, industriales o tecnológicos, actuando con responsabilidad ética, pensamiento crítico, y colaboración interdisciplinaria.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Compromiso con la Responsabilidad Social, Excelencia, Vanguardia e Innovación Social

VII. DESARROLLO POR UNIDADES

Unidad I. Introducción a la inteligencia artificial

Competencia

Identificar los conceptos fundamentales, paradigmas y aplicaciones de la inteligencia artificial, por medio de la revisión histórica, estudio de los paradigmas y el análisis de casos de aplicación de la IA, para determinar su relevancia en la solución de problemas computacionales y reconocer su impacto en distintos ámbitos, con actitud crítica y responsable.

Contenido

- 1.1 Conceptos fundamentales
 - 1.1.1 Definiciones de inteligencia artificial: débil, fuerte y aplicada
 - 1.1.2 Historia, hitos y ciclos de la IA
 - 1.1.3 Tipos de problemas computacionales adecuados para IA
- 1.2 Paradigmas de la IA
 - 1.2.1 Enfoque simbólico
 - 1.2.2 Enfoque conexionista
 - 1.2.3 Enfoque estadístico–probabilístico
 - 1.2.4 Enfoques híbridos
- 1.3 Aplicaciones contemporáneas de la IA
 - 1.3.1 Sectores de aplicación: salud, industria, finanzas, educación
 - 1.3.2 Tendencias actuales: IA generativa, automatización, sistemas interactivos

Duración

04 horas

Unidad II. Agentes inteligentes y toma de decisiones

Competencia

Analizar los componentes, tipos de entornos y arquitecturas de agentes inteligentes, a través del estudio de ejemplos prácticos de interacción agente–entorno y la revisión de bibliografía especializada, para caracterizar adecuadamente problemas computacionales que puedan modelarse mediante agentes inteligentes, con actitud crítica, sistemática y responsable.

Contenido

- 2.1 Agentes y ambientes
 - 2.1.1 Definición de agente y agente inteligente
 - 2.1.2 Especificación de entornos
 - 2.1.3 Arquitecturas de agentes
- 2.2 Funciones de agente
 - 2.2.1 Percepción, acción y función objetivo
 - 2.2.2 Medidas de desempeño y racionalidad
- 2.3 Racionalidad y autonomía
 - 2.3.1 Comportamiento racional
 - 2.3.2 Adaptabilidad y aprendizaje básico en agentes

Duración

06 horas

Unidad III. Solución de problemas mediante búsqueda

Competencia

Aplicar técnicas de búsqueda no informada, informada, local y adversaria, mediante la implementación y comparación de algoritmos clásicos en espacios de estados, para evaluar su eficiencia y pertinencia en la solución de problemas computacionales, con precisión, actitud analítica, ética y responsable.

Contenido

- 3.1 Búsqueda no informada
 - 3.1.1 Búsqueda primero en anchura (BFS)
 - 3.1.2 Búsqueda primero en profundidad (DFS)
 - 3.1.3 Búsqueda con profundidad limitada (DLS)
 - 3.1.4 Complejidad y análisis de rendimiento
- 3.2 Búsqueda informada
 - 3.2.1 Concepto de heurística
 - 3.2.2 Algoritmo A*
 - 3.2.3 Variantes y propiedades
- 3.3 Búsqueda local
 - 3.3.1 Hill Climbing
 - 3.3.2 Simulated Annealing

3.3.3 Algoritmos genéticos: fundamentos

3.4 Juegos y adversarios

3.4.1 Algoritmo Minimax

3.4.2 Poda alfa–beta

Duración

10 horas

Unidad IV. Representación del conocimiento y razonamiento

Competencia

Interpretar modelos simbólicos de representación del conocimiento e implementar mecanismos básicos de inferencia lógica, a través de ejemplos, reglas y bases de conocimiento, para explicar cómo se construyen y utilizan estos sistemas en la solución de problemas computacionales, con actitud lógica y responsable.

Contenido

4.1 Representación simbólica

4.1.1 Lógica proposicional

4.1.2 Lógica de primer orden

4.2 Inferencia lógica

4.2.1 Encadenamiento hacia adelante

4.2.2 Encadenamiento hacia atrás

4.2.3 Método de resolución

4.3 Bases de conocimiento

4.3.1 Concepto y estructura

4.3.2 Reglas de producción

4.3.3 Marcos, ontologías e introducción a grafos semánticos

4.4 Lenguajes declarativos

4.4.1 Introducción conceptual a Prolog

4.4.2 Reglas, consultas y hechos

Duración

06 horas

Unidad V. Introducción al aprendizaje automático

Competencia

Identificar los principios esenciales del aprendizaje automático, mediante la revisión de tipos de datos, problemas básicos y métricas de evaluación, para distinguir los enfoques fundamentales de análisis y modelado en aprendizaje automático, con actitud crítica, reflexiva, ética y responsable.

Contenido

5.1 Conceptos fundamentales

5.1.1 Datos, etiquetas y funciones objetivo

5.1.2 Generalización y sobreajuste

5.2 Tipos básicos de problemas de aprendizaje automático

5.2.1 Clasificación

5.2.2 Regresión

5.2.3 Agrupamiento (clustering)

5.3 Evaluación de modelos

5.3.1 Exactitud (accuracy)

5.3.2 Precisión, sensibilidad (recall) y F1-score

Duración

03 horas

Unidad VI. Ética, sesgos y regulación de la IA

Competencia

Examinar los sesgos algorítmicos y principios de IA responsable, mediante el análisis de casos reales y marcos regulatorios internacionales y nacionales, para valorar el impacto social y ético del desarrollo y uso de sistemas inteligentes, con actitud ética, objetiva y comprometida con la responsabilidad social.

Contenido

6.1 Sesgos algorítmicos

6.1.1 Tipos de sesgos: de datos, diseño y despliegue

6.1.2 Consecuencias en la toma de decisiones automatizadas

6.2 Principios de IA responsable

6.2.1 Transparencia

6.2.2 Explicabilidad

6.2.3 Derecho a explicación

6.3 Marco regulatorio internacional y nacional

6.3.1 Ley de Inteligencia Artificial de la UE

6.3.2 UNESCO — Recomendaciones éticas

6.3.3 Normatividad mexicana emergente

Duración

03 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I: Introducción a la Inteligencia Artificial

Práctica 1: Línea de tiempo de la inteligencia artificial

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes sobre la evolución histórica de la IA.
2. Revisa fuentes confiables (libros, artículos, cronologías académicas) para identificar acontecimientos relevantes.
3. Selecciona con base a la revisión anterior, al menos 10 eventos significativos (investigadores, algoritmos, avances tecnológicos).
4. Organiza los eventos en orden cronológico, agrupándolos por etapas (simbólica, conexionista, estadística, moderna).
5. Elabora una línea de tiempo digital con diseño claro, referencias y una breve explicación.
6. Entrega la línea de tiempo acompañada de una reflexión sobre qué paradigmas han influido más en la disciplina para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Internet
- Editor gráfico o plataforma digital (Canva, PowerPoint, Genially)
- Material bibliográfico o documental proporcionado por el docente

Unidad II: Agentes inteligentes y toma de decisiones

Práctica 2: Simulación de un agente simple en un entorno controlado

Duración

04 horas

Procedimiento

1. Atiende las instrucciones del docente sobre modelos básicos de agentes y entornos.
2. Implementa un entorno simple (aspiradora, recolector, robot en línea, grid world).
3. Diseña un agente reactivo basado en reglas estímulo–respuesta.
4. Ejecuta varias simulaciones modificando condiciones del entorno (deterministas, estocásticas, observables, parcialmente observables, según corresponda).
5. Registra el desempeño del agente (acciones, éxito, eficiencia).
6. Analiza cómo las propiedades del entorno afectan el comportamiento del agente.
7. Entrega un reporte con código, capturas, reflexiones sobre el comportamiento observado y conclusiones para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Lenguaje como Python o MATLAB
- Plantilla base de simulación
- Editor de texto o notebook interactivo (jupyter, google colab)

Unidad III: Solución de problemas mediante búsqueda

Práctica 3: Implementación de algoritmos de búsqueda no informada

Duración

06 horas

Procedimiento

1. Atiende las orientaciones docentes sobre espacios de estados y algoritmos de búsqueda no informada.
2. Implementa BFS y DFS para un problema asignado (laberinto, grafo o rompecabezas simple).
3. Registra nodos expandidos, profundidad, memoria utilizada y tiempo aproximado.
4. Analiza y compara el rendimiento de ambos algoritmos en diferentes configuraciones.
5. Representa los resultados mediante tablas o gráficas.
6. Entrega un informe con código, análisis comparativo y conclusiones para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Bibliotecas especializadas de Python o MATLAB
- Material bibliográfico o documental proporcionado por el docente
- Editor de texto o notebook jupyter o google colab

Práctica 4: Implementación de algoritmos de búsqueda informada

Duración

06 horas

Procedimiento

1. Atiende la introducción docente sobre heurísticas admisibles y consistentes.
2. Implementa el algoritmo A*.
3. Diseña al menos dos heurísticas: una admisible y una no admisible.
4. Evalúa el rendimiento del algoritmo con cada heurística.
5. Analiza cómo influye la heurística en el número de nodos expandidos y la calidad de la solución.
6. Elabora un reporte comparativo con visualizaciones claras.
7. Entrega reporte para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Bibliotecas especializadas de Python o MATLAB
- Material bibliográfico o documental proporcionado por el docente
- Editor de texto o notebook jupyter o google colab

Práctica 5: Simulación de un agente simple en un entorno controlado

Duración

06 horas

Procedimiento

1. Atiende las orientaciones del docente sobre los fundamentos de la búsqueda local y su diferencia respecto a la búsqueda sistemática.
2. Implementa el algoritmo Hill Climbing en un problema continuo o discreto (por ejemplo, optimización de una función simple o resolución parcial de un rompecabezas).
3. Implementa el algoritmo Simulated Annealing, ajustando parámetros básicos como temperatura inicial, esquema de decremento y criterio de aceptación de soluciones.
4. Registra para ambos algoritmos métricas como: número de iteraciones, convergencia, calidad de la solución y comportamiento frente a óptimos locales.
5. Atiende la introducción docente al paradigma evolutivo y los principios básicos de los algoritmos genéticos (AG).
6. Implementa un algoritmo genético simple, definiendo:
 - a. Representación de individuos (codificación binaria o real).
 - b. Función de aptitud (fitness).
 - c. Operadores de selección (ruleta o torneo).
 - d. Operadores de cruce (one-point o uniform crossover).
 - e. Operador de mutación con una probabilidad baja.
7. Ejecuta el algoritmo genético por varias generaciones y registra: evolución del fitness, estabilidad, diversidad poblacional y solución final obtenida.
8. Compara el desempeño del algoritmo genético con Hill Climbing y Simulated Annealing para el mismo problema, señalando ventajas y limitaciones de cada enfoque.
9. Elabora un informe con código, gráficas opcionales del proceso evolutivo y análisis comparativo de los tres métodos.
10. Entrega el informe final para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Lenguaje de programación como Python o MATLAB
- Librerías de apoyo (NumPy, Matplotlib)
- Editor de texto o notebook interactivo
- Material de consulta proporcionado por el docente

Práctica 6: Implementación de minimax y poda alfa-beta

Duración

04 horas

Procedimiento

1. Atiende la explicación del docente sobre juegos adversarios y árboles de decisión.
2. Implementa Minimax para un juego sencillo (Tic-tac-toe o Connect-4 reducido).
3. Añade la poda alfa-beta y verifica que reduzca el número de nodos evaluados.
4. Ejecuta partidas de prueba para observar decisiones del agente.
5. Registra y compara el rendimiento con y sin poda.
6. Documenta resultados y reflexiona sobre la racionalidad del jugador artificial.
7. Entrega reporte con los resultados y las conclusiones de la actividad para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Lenguaje de programación como Python o MATLAB
- Juego base o plantilla proporcionada
- Editor de texto o notebook interactivo
- Material de consulta proporcionado por el docente

Unidad IV: Solución de problemas mediante búsqueda

Práctica 7: Implementación de un sistema experto

Duración

04 horas

Procedimiento

1. Atiende las orientaciones del docente sobre bases de conocimiento y reglas de producción.
2. Selecciona un dominio sencillo (diagnóstico simple, recomendación, categorización).
3. Define hechos y reglas usando Prolog.
4. Ejecuta consultas y revisa cómo el sistema infiere conclusiones.
5. Analiza limitaciones del enfoque simbólico.
6. Entrega un reporte con código, consultas y conclusiones para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Lenguaje de programación SWI-Prolog
- Editor de texto

Unidad V: Introducción al aprendizaje automático

Práctica 8: Análisis de distribuciones en conjuntos de datos y riesgos de sesgo

Duración

04 horas

Procedimiento

1. Atiende indicaciones docentes sobre la importancia de revisar datos antes de usarlos en IA.
2. Selecciona un conjunto de datos sencillo (adult-income, iris o el proporcionado por el docente).
3. Examina la distribución de clases, proporciones, valores faltantes y variabilidad.
4. Genera tablas o gráficas (histogramas, gráficas de barras) que representen la estructura del dataset .
5. Identifica patrones que podrían afectar el rendimiento o equidad de un modelo (desbalance, atributos sensibles, subrepresentación).
6. Elabora una reflexión sobre cómo estas características podrían generar resultados inadecuados.
7. Entrega un reporte con gráficas y conclusiones para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Lenguaje Python con librerías especializadas (pandas, NumPy, matplotlib)
- Base de datos
- Editor de texto

Unidad VI: Ética, sesgos y regulación de la IA

Práctica 9: Elaboración de una infografía sobre ética, sesgos y regulación de la IA

Duración

03 horas

Procedimiento

1. Atiende las orientaciones docentes sobre principios de IA responsable.
2. Revisa documentos base (UE AI Act, UNESCO, normatividad mexicana emergente).
3. Selecciona un enfoque (sesgos, transparencia, regulación, impacto social).
4. Organiza la información en una estructura clara para una infografía.
5. Diseña la infografía en Canva, PowerPoint o herramienta similar.
6. Entrega infografía acompañada de una reflexión sobre la importancia de la IA responsable, para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Internet
- Canva, PowerPoint o Genially
- Documentos de referencia

Práctica 10: Proyecto integrador de inteligencia artificial

Duración

09 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el planteamiento formal del problema y el alcance del proyecto.
2. Selecciona un problema computacional adecuado para modelarse mediante agentes, búsqueda o razonamiento simbólico.
3. Define estado inicial, espacio de estados, reglas, heurísticas o base de conocimiento según corresponda.
4. Implementa la técnica de IA elegida y genera un prototipo funcional.
5. Evalúa el rendimiento del sistema con métricas objetivas.
6. Analiza limitaciones y posibles implicaciones éticas relacionadas con la aplicación propuesta.
7. Elabora un reporte técnico con los siguientes apartados:
 - a. Planteamiento del problema
 - b. Metodología
 - c. Resultados
 - d. Discusión crítica
 - e. Conclusiones
8. Diseña una presentación digital del proyecto.
9. Expone el proyecto ante el grupo.
10. Entrega reporte técnico para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Lenguaje de programación como Python o MATLAB
- Herramientas de presentación
- Documentación técnica

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer de forma clara y formal:

- La metodología de trabajo.
- Criterios de evaluación y rúbricas.
- Reglas sobre la calidad y entrega de trabajos académicos.
- Derechos y obligaciones del docente y los estudiantes.

Esto debe quedar registrado y comunicado por medios institucionales (como Blackboard o correo institucional).

Estrategia de enseñanza

Son acciones didácticas planificadas por el docente para facilitar el logro de los aprendizajes esperados. Estas estrategias deben seleccionarse en función del tipo de competencia, el nivel cognitivo y la naturaleza del contenido.

Estos son ejemplos de estrategias, puede usar algunas de esta lista u otras que se consideren pertinentes:

- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Técnica expositiva
- Instrucción guiada

Estrategia de aprendizaje

Son enfoques que el estudiante emplea para organizar, adquirir, comprender y aplicar el conocimiento. Estas estrategias deben promover el desarrollo de competencias, el pensamiento crítico y el aprendizaje significativo.

Estos son ejemplos de estrategias, puede usar algunas de esta lista u otras que se consideren pertinentes:

- Investigación documental
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Cuadros comparativos
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	30%
– Prácticas laboratorio	40%
– Proyecto Integrador	30%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Bhargava, N., Bhargava, R., Rathore, P. S., y Agrawal, R. (2021). *Artificial Intelligence and Data Mining Approaches in Security Frameworks*. John Wiley & Sons.

Bhuvaneswari, B. (2021). *Artificial Intelligence*. Astral International Pvt Ltd.
<https://libcon.rec.uabc.mx:2938/linkprocessor/plink?id=52bc4529-dabb-3336-a44d-53a5dbd29a41>

Bratko, I. (2012). *Prolog Programming for Artificial Intelligence* (4a ed.). Pearson. [clásica]

Centro de Investigación para el Aprendizaje Digital. (2024). *Recomendaciones para la integración efectiva de la inteligencia artificial generativa en la práctica docente de educación superior*. Universidad Autónoma de Baja California.
https://ciad.mxl.uabc.mx/wp-content/uploads/2024/01/IA_Practica_Docente_2024-01-24.pdf

Floridi, L. (2023). *The ethics of artificial intelligence: Principles, challenges, and opportunities*. Oxford University Press.

Mehan, J. (2024). *Artificial Intelligence : Ethical, Social, and Security Impacts for the Present and the Future* (2a ed.). ITGP.
<https://libcon.rec.uabc.mx:2938/linkprocessor/plink?id=8c92f0c4-578f-3017-bbec-a49ae00f6a26>

Moroney, L. (2020). *AI and Machine Learning for Coders: A Programmer's Guide to Artificial Intelligence*. O'Reilly Media.

Neapolitan, R. E., & Jiang, X. (2018). *Artificial intelligence: With an introduction to machine learning*. CRC press.

Russell, S., & Norvig, P. (2021). *Artificial Intelligence: A Modern Approach* (4a ed.). Pearson.

Stobo, J. (1989). *Problem Solving With Prolog*. CRC Press. [clásica]
<https://libcon.rec.uabc.mx:2938/linkprocessor/plink?id=733affb9-cdda-3686-99f9-a27f057ed107>

Recursos de acceso libre o complementarios externos

Barocas, S., Hardt, M., y Narayanan, A. (2023). *Fairness and Machine Learning : Limitations and Opportunities*. MIT Press. <https://fairmlbook.org/>

European Union (2023). *EU Artificial Intelligence Act (AI Act)*. <https://artificialintelligenceact.eu/>

Murphy, K. P. (2012). *Machine learning: A probabilistic perspective*. MIT Press.
<https://www.cs.ubc.ca/~murphyk/>

Poole, D., y Mackworth, A. (2023). *Artificial Intelligence: Foundations of Computational Agents* (3a ed.). Cambridge University Press. <https://artint.info/3e/html/ArtInt3e.html>

Shalev-Shwartz, S., y Ben-David, S. (2014). *Understanding machine learning: From theory to algorithms*. Cambridge University Press.
<https://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/>

UNESCO (2022). *Recommendation on the Ethics of Artificial Intelligence*.
<https://unesdoc.unesco.org/ark:/48223/pf0000381137>

XII. PERFIL DOCENTE

La persona docente que imparta esta unidad de aprendizaje se espera que cuente con Licenciatura en Ciencias Computacionales, Ingeniería en Computación, Matemáticas, Física o áreas afines, preferentemente con estudios de posgrado en inteligencia artificial, ciencia de datos o campos relacionados. Se requiere experiencia en docencia universitaria y dominio de los fundamentos de la inteligencia artificial clásica, incluyendo agentes inteligentes, algoritmos de búsqueda, representación del conocimiento y razonamiento lógico.

Deberá manejar programación en Python, Matlab y Prolog, así como integrar de manera ética y responsable el uso de tecnologías emergentes para fortalecer el pensamiento computacional. Asimismo, deberá tener la capacidad de explicar conceptos con rigor y claridad, diseñar actividades teóricas y prácticas, y promover disciplina, honestidad académica, trabajo colaborativo y actualización continua.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Compiladores
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 03 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Disciplinaria
Carácter:	Obligatoria
Requisito:	Teoría de Autómatas
Modalidad de impartición:	Presencial
Equipo de diseño:	Everardo Gutiérrez López María Victoria Meza Kubo Antonio de Jesús García Chávez Pedro Pérez Ruíz
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

El estudio del proceso de compilación representa en el perfil de los egresados de ciencias computacionales la integración de sus conocimientos sobre lenguajes, programación, arquitectura de computadoras y teoría de autómatas. Esto se enmarca en el área de desarrollo de software de base, la cual se encarga de la generación de soluciones de software que permitan optimizar el rendimiento, la gestión de recursos y la comunicación entre sistemas. El alumnado hará un estudio comparativo de los principios, técnicas y herramientas para el desarrollo de cada una de las etapas del proceso de compilación. Este curso se encarga de proporcionar al alumnado una visión amplia sobre el uso y aplicación de los compiladores como solución a distintos problemas reales. Adicionalmente, se muestra la importancia del desarrollo de compiladores, como una solución a problemas o como una herramienta más en el caso de requerir una herramienta con características especiales para el desarrollo de proyectos futuros. Se encuentra en la etapa disciplinaria, es de carácter obligatorio y pertenece al área de conocimiento de Software de Bases. Será requisito haber acreditado el curso de Teoría de Autómatas.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Desarrollar los diferentes módulos de un compilador a través de las técnicas correspondientes a cada fase del proceso de traducción de un lenguaje de programación, para utilizarlos en la solución de aplicaciones reales, con una actitud creativa, propositiva y de responsabilidad.

IV. EVIDENCIA DE APRENDIZAJE

Desarrolla los módulos de un compilador y expone su proyecto explicando la implementación realizada, los resultados obtenidos y las dificultades presentadas.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Desarrollar software de base eficiente y robusto mediante la integración de conocimientos en arquitectura y redes de computadoras y los fundamentos de compiladores y sistemas operativos, para optimizar el rendimiento, la gestión de recursos y la comunicación entre sistemas, asegurando aplicaciones funcionales, seguras, escalables y sostenibles, con una actitud ética, responsable y visionaria.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Pensamiento Crítico.

VII. DESARROLLO POR UNIDADES

Unidad I. Introducción a los compiladores

Competencia

Analizar los conceptos principales de los compiladores, a través de una revisión histórica de su evolución y etapas, para otorgar una visión general sobre su importancia en la resolución de problemas, con una actitud sistemática, metódica y con pensamiento crítico.

Contenido

- 1.1. Conceptos e importancia de los compiladores
- 1.2. Historia de los compiladores
- 1.3. Fases de un compilador
- 1.4. Compiladores modernos

Duración

04 horas

Unidad II. Análisis léxico

Competencia

Implementar un analizador léxico en el contexto del proceso de compilación, mediante la utilización de técnicas estandarizadas que permitan realizar el análisis del programa de entrada para su preparación a las siguientes etapas, con actitud creativa y propositiva.

Contenido

- 2.1. Función del analizador léxico
- 2.2. Componentes léxicos y sus atributos
- 2.3. Lenguajes regulares y expresiones regulares
- 2.4. Autómatas finitos
- 2.5. Tablas de símbolos
- 2.6. Manejo de errores léxicos

Duración

04 horas

Unidad III. Análisis sintáctico

Competencia

Implementar un analizador sintáctico como parte del proceso de compilación, mediante la utilización de técnicas de análisis sintáctico que permitan realizar la estructuración de los componentes del programa de entrada para generar el árbol sintáctico que represente el programa de entrada para su utilización en las etapas posteriores de la compilación, con actitud creativa y propositiva.

Contenido

- 3.1. Propósito del analizador sintáctico
- 3.2. Gramáticas libres de contexto
- 3.3. Árboles sintácticos
- 3.4. Gramáticas ambiguas
- 3.5. Análisis sintáctico descendente
- 3.6. Análisis sintáctico ascendente
- 3.7. Análisis sintáctico predictivo
- 3.8. Manejo de errores

Duración

08 horas

Unidad IV. Análisis semántico

Competencia

Desarrollar un analizador semántico como componente de un compilador moderno, mediante la implementación de técnicas de análisis semántico que verifiquen las instrucciones a realizar a partir del programa de entrada para generar todos los elementos requeridos en la etapa de generación de código, con actitud creativa y propositiva.

Contenido

- 4.1. Funciones del analizador semántico
- 4.2. Ámbito y tabla de símbolos
- 4.3. Tipos y operaciones
- 4.4. Sistema de tipos
- 4.5. Reglas de inferencia de tipos
- 4.6. Subtipos

Duración

08 horas

Unidad V. Generación y optimización de código

Competencia

Desarrollar un generador de código para un compilador moderno, mediante la implementación de técnicas de generación y optimización de código para finalizar la traducción de instrucciones de alto nivel a bajo nivel, con actitud creativa y propositiva.

Contenido

- 5.1. Entornos de ejecución
- 5.2. Máquinas basadas en pilas
- 5.3. Lenguaje ensamblador
- 5.4. Generador de código simple
- 5.5. Enlace dinámico

5.6. Estrategias para la optimización de código

Duración

08 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I: Introducción a los compiladores

Práctica 1: Exploración de compiladores y lenguajes

Duración

03 horas

Procedimiento

1. Atiende las instrucciones de su docente.
2. Investiga diferentes tipos de lenguajes (compilados, interpretados e híbridos).
3. Analiza el flujo de compilación de al menos dos lenguajes (como C, Python, Java).
4. Identifica las etapas del compilador en diagramas de flujo.
5. Elabora un reporte comparativo resaltando similitudes y diferencias.
6. Entrega reporte y recibe retroalimentación de su docente.

Recursos de apoyo

- Libro con los principios, técnicas y herramientas sobre compiladores.
- Documentación oficial de lenguajes de compiladores para programación de alto nivel.
- Presentación introductoria del docente.

Práctica 2: Generación y optimización de código

Duración

03 horas

Procedimiento

1. Atiende las instrucciones de su docente.
2. Revisa el código fuente de un compilador o intérprete sencillo.
3. Identifica módulos relacionados con análisis léxico, sintáctico y semántico.
4. Ejecuta ejemplos y observa resultados intermedios.
5. Documenta la función de cada módulo.
6. Entrega reporte y recibe retroalimentación de su docente.

Recursos de apoyo

- Repositorio base proporcionado por el docente.
- IDE (VS Code, Eclipse o similar).
- Guía de lectura dirigida.

Unidad II: Análisis léxico

Práctica 3: Diseño de tokens y expresiones regulares

Duración

03 horas

Procedimiento

1. Atiende las instrucciones de su docente.
2. Define el conjunto de tokens para un lenguaje imperativo sencillo.
3. Especifica expresiones regulares para identificadores, números y operadores.
4. Prueba las expresiones con cadenas de entrada.
5. Documenta la tabla de tokens resultante.
6. Entrega reporte y recibe retroalimentación de su docente.

Recursos de apoyo

- Apuntes de expresiones regulares.
- Herramientas en línea para pruebas de expresiones regulares.
- Editor de texto o IDE.

Práctica 4: Implementación de un analizador léxico

Duración

03 horas

Procedimiento

1. Atiende las instrucciones de su docente.
2. Implementa un analizador léxico usando una herramienta (por ejemplo Lex/Flex) o código manual.
3. Lee un archivo fuente como entrada.
4. Genera y muestra la secuencia de tokens.
5. Maneja errores léxicos básicos.
6. Entrega reporte y recibe retroalimentación de su docente.

Recursos de apoyo

- Flex o biblioteca equivalente para análisis léxico.
- Manual de usuario de la herramienta.
- Ejemplos de código proporcionados.

Unidad III: Análisis sintáctico

Práctica 5: Definición de gramática y análisis sintáctico

Duración

06 horas

Procedimiento

1. Atiende las instrucciones de su docente.
2. Define una gramática libre de contexto para el lenguaje del curso.
3. Calcula conjuntos FIRST y FOLLOW.
4. Verifica ambigüedad y factorización.
5. Entrega reporte con la gramática corregida y recibe retroalimentación de su docente.

Recursos de apoyo

- Apuntes sobre gramáticas para lenguajes de programación.
- Ejercicios resueltos.
- Pizarra o editor colaborativo.

Práctica 6: Implementación de un parser

Duración

06 horas

Procedimiento

1. Atiende las instrucciones de su docente.
2. Implementa un analizador sintáctico (descendente o ascendente).
3. Integra el analizador sintáctico con el analizador léxico.
4. Construye el árbol sintáctico.
5. Prueba con programas válidos e inválidos.
6. Entrega reporte y recibe retroalimentación de su docente.

Recursos de apoyo

- Librería para parsing (por ejemplo Yacc/Bison) o implementación manual.
- Diagramas de árboles sintácticos.
- IDE y depurador.

Unidad IV: Análisis semántico

Práctica 7: Tabla de símbolos

Duración

06 horas

Procedimiento

1. Atiende las instrucciones de su docente.

2. Diseña la estructura de una tabla de símbolos.
3. Implementa inserción y búsqueda de identificadores.
4. Asocia los conceptos de tipos y ámbitos.
5. Prueba con declaraciones y bloques anidados.
6. Entrega reporte y recibe retroalimentación de su docente.

Recursos de apoyo

- Ejemplos de tablas de símbolos.
- Código base del parser.
- Diagramas o reglas de ámbito.

Práctica 8: Verificación semántica

Duración

06 horas

Procedimiento

1. Atiende las instrucciones de su docente.
2. Implementa verificaciones de tipos.
3. Detecta errores semánticos comunes.
4. Integra mensajes de error claros.
5. Prueba con casos predefinidos.
6. Entrega reporte y recibe retroalimentación de su docente.

Recursos de apoyo

- Apuntes de análisis semántico.
- Casos de prueba proporcionados.
- IDE con soporte de depuración.

Unidad V: Generación y optimización de código

Práctica 9: Generación de código intermedio

Duración

06 horas

Procedimiento

1. Atiende las instrucciones de su docente.
2. Define una representación de código intermedio (tres direcciones).
3. Genera código a partir del árbol sintáctico.
4. Muestra el resultado en consola o archivo.
5. Valida la corrección del código generado.
6. Entrega reporte y recibe retroalimentación de su docente.

Recursos de apoyo

- Ejemplos de código intermedio.
- Notas de generación de código.
- Código base del compilador.

Práctica 10: Optimización básica de código

Duración

06 horas

Procedimiento

1. Atiende las instrucciones de su docente.
2. Implementa optimizaciones simples (como eliminación de código muerto, propagación de constantes).
3. Compara código antes y después de optimizar.
4. Mide mejoras en tamaño o eficiencia.
5. Elabora un breve informe de resultados.
6. Entrega reporte y recibe retroalimentación de su docente.

Recursos de apoyo

- Apuntes de optimización.
- Casos de prueba.
- Herramientas de análisis de código.

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Estudio de caso
- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Aprendizaje basado en preguntas
- Técnica expositiva
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica

- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Mapas conceptuales
- Cuadros comparativos

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	40%
– Prácticas de laboratorio	20%
– Proyecto final (compilador)	40%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Aho, A.V., Lam, Monica S, Sethi, R., Ullman, Jeffrey D. (2013). *Compilers: Principles, Techniques, and Tools*. Pearson. [clásica]
- Appel, A. W., & Palsberg, J. (2004). *Modern Compiler Implementation in Java*. Cambridge University Press. [Clásica]
- Campbell, B., Iyer, S., & Akbal-Delibas, B. (2012). *Introduction to compiler construction in a Java world*. CRC Press. [Clásica].
- Cooper, K. D., & Torczon, L. (2022). *Engineering a compiler (3rd ed.)*. Morgan Kaufmann.
- Flex, Bison & MinGw (2016). *Construcción de compiladores básicos*. Editorial Académica Española. [Clásica]
- Grune, D., van Reeuwijk, K., Bal, H. E., Jacobs, C. J. H., & Langendoen, K. (2012). *Modern Compiler Design*. Springer. [Clásica]
- Louden, K.C. (2004). *Construcción de compiladores : principios y práctica*. Thompson. [Clásica]
- Srikant, Y. N., & Shankar, P. (2002). *The compiler design handbook: optimizations and machine code generation*. CRC Press. [Clásica]

Recursos de acceso libre o complementarios externos

ACM Special Interest Group on Programming Languages. <https://www.sigplan.org/>

- Brais, H. (2015). *Compiladores: Lo que todo programador debe saber sobre las optimizaciones del compilador*. Microsoft Ignite, 30(2). [Clásica] <https://msdn.microsoft.com/en-us/magazine/dn904673.aspx>
- Colombet, Q. (2025). LLVM code generation: A deep dive into compiler backend development. Packt Publishing.
- Hall, M., Padua, D., & Pingali, K. (2009). *Compiler research: the next 50 years*. Communications of the ACM, 52(2), 60-67. [Clásica]
- Hopcroft, J. E., Motwani, R., & Ullman, J. D. (2006). *Automata theory, languages, and computation*. International Edition, 24. [Clásica]
- IEEE Computer Society. <http://www.computer.org/>
- IngenieríaSimple.com. (2025). *Lecturas sobre compiladores y lenguajes formales*. <https://www.ingenieriasimple.com/compiladores/>
- Ma, H. (2023). *A Survey of Modern Compiler Fuzzing*. arXiv preprint. <https://arxiv.org/abs/2306.06884>
- Mössenböck, H. (2025). *Compiler Construction: Fundamentals and Applications*. Springer Cham. <https://ssw.jku.at/CompilerBook/>
- Ni, Y., & Li, S. (2025). *Interleaving Large Language Models for Compiler Testing*. arXiv preprint. <https://arxiv.org/abs/2508.18955>
- Nystrom, R. (2021). *Crafting interpreters*. Genever Benning.
- Sandler, N. (2024). *Writing a C compiler: Build a real programming language from scratch*. No Starch Press.
- Torczon, L., & Cooper, K. D. (2011). *Engineering a Compiler* (2ª ed.). Morgan Kaufmann. [Clásica]

XII. PERFIL DOCENTE

Licenciatura en Ciencias Computacionales, o área afín; preferentemente con estudios de posgrado y dos años de experiencia de práctica docente, además deberá tener conocimiento en el manejo de las temáticas relativas al área de compiladores y teoría de autómatas, debe tener capacidad de abstracción además de una actitud analítica, crítica y de razonamiento lógico matemático.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Reingeniería de Procesos
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 03 Horas Laboratorio: 00 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Disciplinaria
Carácter:	Obligatoria
Requisito:	Ingeniería de Software
Modalidad de impartición:	Presencial
Equipo de diseño:	Mary Sol Aguirre Arzate María Victoria Meza Kubo
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La adopción del enfoque de procesos y el uso de TIC representan un cambio revolucionario en la perspectiva de las organizaciones. Lo nuevo y distintivo de esta combinación es su enorme potencial para ayudar a la organización a lograr reducciones mayores en costo y tiempo, y/o mejoras en calidad, flexibilidad, niveles de servicio y objetivos. De donde surge la necesidad del estudio, análisis y rediseño de estos procesos, así como el diseño, implementación y/o uso de herramientas de soporte. En el aspecto concerniente al desarrollo de soporte a los procesos por medio de TI un aspecto importante es, precisamente, el poder establecer claramente qué tipo de sistemas se necesitan en una organización determinada, así como los requerimientos para el desarrollo de los mismos, y los aspectos sociales que influyen tanto durante el desarrollo y su implantación en la organización. De lo anterior podemos establecer claramente que el propósito de esta unidad de aprendizaje es que el alumnado pueda rediseñar los procesos de una organización considerando un análisis socio-técnico de las necesidades de la organización. Se ubica en la etapa Disciplinaria, es de carácter obligatoria y pertenece al área de conocimiento de Entorno Social y es requisito haber cursado y aprobado la unidad de aprendizaje de Ingeniería de Software.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Rediseñar los procesos de una organización, con la aplicación de una metodología de reingeniería, para mejorar la competitividad de la empresa y el ambiente laboral de los trabajadores, con actitud crítica, propositiva y responsabilidad social.

IV. EVIDENCIA DE APRENDIZAJE

Genera un proyecto de rediseño de los procesos de una organización que integre los resultados de la aplicación de una metodología de reingeniería de procesos que fundamenten la solución propuesta. Este proyecto debe incluir un prototipo funcional y la documentación correspondiente.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Inclusión y Pensamiento Crítico.

VII. DESARROLLO POR UNIDADES

Unidad I. Introducción de la Ingeniería de procesos

Competencia

Analizar el origen, evolución y propósito de la reingeniería de procesos en las organizaciones, con revisión y comprensión de literatura de procesos organizacionales que permita distinguir su impacto en la competitividad y en la transformación digital, con una actitud crítica y reflexiva.

Contenido

- 1.1. Concepto de proceso y organización orientada a procesos
- 1.2. Evolución de la reingeniería de procesos y su relación con la transformación digital
- 1.3. Factores de éxito y fracaso en proyectos de reingeniería
- 1.4. Impacto socio-técnico de las Tecnologías de la Información (TI)

Duración

04 horas

Unidad II. Modelado de procesos

Competencia

Capturar un proceso real, con el uso de estándares de modelado y herramientas digitales, con el fin de lograr una comprensión integral y comunicable de la operación empresarial con responsabilidad y pensamiento crítico.

Contenido

- 2.1. Objetivos y beneficios del modelado de procesos
- 2.2. Elementos clave: actores, roles, agentes, flujos de información
- 2.3. Perspectivas que debe cubrir un modelo completo de un proceso (funcional, organizacional, informacional y de comportamiento)
- 2.4. Estándares de modelado: BPMN 2.0, UML de actividades, IDEF, RADs
- 2.5. Herramientas digitales para el modelado de procesos (Bizagi, Camunda, Signavio, alternativas open-source)

Duración

06 horas

Unidad III. Diagnóstico y análisis de procesos

Competencia

Evaluar procesos organizacionales mediante el uso de técnicas de análisis y métricas de desempeño, para identificar oportunidades de mejora en los procesos, con una actitud crítica, propositiva y responsable.

Contenido

- 3.1. Diagnóstico de procesos: eficiencia, eficacia, flexibilidad y calidad
- 3.2. Indicadores de desempeño de procesos (KPIs, SLAs, OKRs)
- 3.3. Métodos de análisis: cuellos de botella, análisis de valor agregado, mapas de calor
- 3.4. Identificación de procesos críticos y procesos enfermos
- 3.5. Uso de software de simulación de procesos

Duración

06 horas

Unidad IV. Metodologías de rediseño de procesos

Competencia:

Proponer rediseños viables alineados con la estrategia organizacional y soportados en tecnologías, mediante las metodologías de reingeniería y mejora de procesos, para mejorar la eficiencia y calidad del servicio de la organización con responsabilidad, ética profesional, respeto a la sociedad y al medio ambiente.

Contenido

- 4.1. Metodologías de reingeniería de procesos (Hammer & Champy, Davenport, BPM Lifecycle)
- 4.2. Enfoques ágiles para el rediseño (Scrum, Kanban, Design Thinking)
- 4.3. Automatización y digitalización de procesos (RPA, IoT, IA, plataformas Low-Code/No-Code)
- 4.4. Consideraciones éticas, sociales y de gestión del cambio en la reingeniería

Duración

08 horas

Unidad V. Implementación de la solución

Competencia

Desarrollar un prototipo funcional con el uso de tecnologías innovadoras y planes de gestión de cambio, para brindar un soporte a la propuesta de rediseño, con actitud crítica, propositiva y responsable.

Contenido

- 5.1. Planeación de la implementación: plan de transición, gestión del cambio, comunicación organizacional
- 5.2. Desarrollo de un prototipo tecnológico de soporte al proceso
- 5.3. Manuales, procedimientos y planes de contingencia
- 5.4. Evaluación de resultados: métricas de impacto económico, social y tecnológico
- 5.5. Mejora continua y sostenibilidad del rediseño

Duración

08 horas

ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

Unidad I. Introducción de la Ingeniería de procesos

Práctica 1: Identificando procesos y su impacto en la organización

Duración

06 horas

Procedimiento

1. Atiende a las explicaciones del docente sobre la definición de proceso y ejemplos en diferentes organizaciones.
2. En equipo, selecciona una organización conocida (universidad, tienda, hospital, empresa de servicios).
3. Identifica al menos tres procesos clave de la organización.
4. Clasifica los procesos en estratégicos, operativos y de apoyo.
5. Discute en equipo cómo los cambios tecnológicos han transformado dichos procesos.
6. Redacta un breve informe donde se señale un proceso que podría ser objeto de reingeniería, justificando la elección.
7. Presenta los hallazgos al resto de la clase.
8. Recibe retroalimentación docente.

Recursos de apoyo

- Pizarrón
- Plumones
- Computadora
- Acceso a internet

Unidad II. Modelado de procesos

Práctica 2: Modelando un proceso con BPMN 2

Duración

06 horas

Procedimiento

1. Atiende a la explicación del docente sobre los elementos básicos de BPMN 2.0.
2. En equipo, selecciona un proceso sencillo (ej. inscripción a cursos, compra en línea, solicitud de vacaciones).
3. Documenta los actores, roles, actividades y flujos de información del proceso.
4. Representa gráficamente el proceso utilizando una herramienta digital (Bizagi Modeler, Draw.io o Lucidchart).
5. Revisa y valida el modelo con otros equipos, verificando que sea claro y completo.
6. Entrega un archivo digital y una breve explicación del modelo elaborado.

Recursos de apoyo

- Computadora
- Software de modelado (Bizagi/Draw.io/Lucidchart)
- Acceso a internet

Unidad III. Diagnóstico y análisis de procesos

Práctica 3: Detectando problemas en procesos reales

Duración

09 horas

Procedimiento

1. Atiende la explicación del docente sobre técnicas de diagnóstico (cuellos de botella, desperdicios, actividades sin valor agregado).
2. En equipo, analiza el modelo creado en la práctica anterior.
3. Identifica posibles problemas del proceso (ineficiencia, duplicidad, tiempos muertos, falta de control).
4. Define al menos tres indicadores de desempeño (KPIs) para medir el proceso.
5. Representa gráficamente los hallazgos mediante un mapa de calor o diagrama de causa-efecto.
6. Presenta tus resultados al grupo y recibe retroalimentación del docente.

Recursos de apoyo

- Software de modelado
- Hojas de trabajo
- Plantillas de indicadores
- Computadora

Unidad IV. Metodologías de rediseño de procesos

Práctica 4: Rediseñando procesos con enfoque ágil

Duración

12 horas

Procedimiento

1. Atiende a las explicaciones del docente sobre metodologías de rediseño (Hammer & Champy, BPM Lifecycle, enfoques ágiles).
2. En equipo, selecciona un proceso previamente diagnosticado.
3. Propone un rediseño del proceso considerando:
 - a. Eliminación de actividades sin valor agregado.
 - b. Inclusión de mejoras tecnológicas (RPA, IA, IoT, automatización).
 - c. Impacto en las personas que lo ejecutan.
4. Representa el nuevo proceso en un diagrama BPMN actualizado.
5. Simula un breve “sprint ágil” de mejora (definir backlog, roles y entregables en dos iteraciones).
6. Documenta las mejoras propuestas y prepara una presentación breve para el grupo.
7. Recibe retroalimentación del grupo y del docente e integra a tu propuesta.

Recursos de apoyo

- Computadora
- Software de modelado
- Hojas
- Acceso a internet
- Material de apoyo sobre metodologías ágiles

Unidad V. Implementación de la solución

Práctica 5: Prototipando la solución y evaluando su impacto

Duración

12 horas

Procedimiento

1. Atiende la explicación del docente sobre la importancia del plan de transición, manuales y métricas de evaluación.
2. En equipo, desarrolla un prototipo funcional mínimo (MVP) que apoye el proceso rediseñado (ej. formulario en línea, flujo automatizado con una herramienta low-code, simulación de workflow).
3. Define un plan de implementación con etapas, responsables y recursos.
4. Establece métricas de éxito (reducción de tiempo, costo, mejora en satisfacción del cliente, etc.).
5. Documenta el plan y presenta un reporte final con el prototipo y la propuesta de evaluación.
6. Presenta el resultado en la sesión final para recibir retroalimentación del grupo y del docente.

Recursos de apoyo

- Computadora

- Herramientas digitales (ej. Power Apps, Airtable, Google Forms, Trello, Camunda Modeler), plantillas de planes de transición

Práctica 6: Presentación final del proyecto

Duración

03 horas

Procedimiento

1. Atiende la explicación del docente sobre la importancia del plan de transición, manuales y métricas de evaluación.
2. En equipo, desarrolla un prototipo funcional mínimo (MVP) que apoye el proceso rediseñado (ej. formulario en línea, flujo automatizado con una herramienta low-code, simulación de workflow).
3. Define un plan de implementación con etapas, responsables y recursos.
4. Establece métricas de éxito (reducción de tiempo, costo, mejora en satisfacción del cliente, etc.).
5. Documenta el plan y presenta un reporte final con el prototipo y la propuesta de evaluación.
6. Presenta el resultado en la sesión final para recibir retroalimentación del grupo y del docente.

Recursos de apoyo

- Computadora
- Herramientas digitales (ej. Power Apps, Airtable, Google Forms, Trello, Camunda Modeler), plantillas de planes de transición

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer de forma clara y formal:

- La metodología de trabajo
- Criterios de evaluación y rúbricas
- Reglas sobre la calidad y entrega de trabajos académicos
- Derechos y obligaciones del docente y los estudiantes

Esto debe quedar registrado y comunicado por medios institucionales (como Blackboard o correo institucional).

Estrategia de enseñanza

- Estudio de caso
- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos

- Aprendizaje basado en preguntas
- Técnica expositiva
- Discusión guiada

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Visitas a campo
- Mapas mentales
- Mapas conceptuales
- Cuadros comparativos
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

El primer día de clase, cada docente debe establecer de forma clara y formal:

- La metodología de trabajo
- Criterios de evaluación y rúbricas
- Reglas sobre la calidad y entrega de trabajos académicos
- Derechos y obligaciones del docente y los estudiantes

Esto debe quedar registrado y comunicado por medios institucionales (como Blackboard o correo institucional).

Criterios de evaluación

– Evaluaciones parciales	20%
– Prácticas de taller	30%
– Tareas	10%
– Proyecto de rediseño	40%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Champy, J.; M. HAMMER (1994): *Reengineering the corporation* (p. 30). GW National Satellite Network. [clásica]
- Stephen A. White; Michael zur Muehlen; Nathaniel Palmer; Thom Allweyer; Denis Gagné, etc. (Segunda edición). *BPMN 2.0 Handbook: Methods, Concepts, Case Studies and Standards in Business Process Modeling Notation (BPMN)*.
- Freund, J., Rücker, B. (2019). *Real-life BPMN: Using BPMN and DMN to Analyze, Improve and Automate Processes in Your Company*. Camunda.
- Fuehrer, J., Almeida, W. (2022). *Learning BPMN 2.0: An Introduction of Engineering Practices for Software Delivery Teams*. (n.p.): Joshua Fuehrer.
- Rodenas Adam, M. (2004). *Reingeniería de procesos y transformación organizativa*. Alfaomega. [clásica]

Recursos de acceso libre o complementarios externos

- Wagner, G. (2021). *Business process modeling and simulation with DPMN* [E-book de acceso abierto]. Disponible en <https://sim4edu.com/reading/bpms-dpmn/>
- Open-BPMN. (s. f.). *Open-BPMN — free and open modelling platform for BPMN 2.0* [Sitio web]. Recuperado el 21 de octubre del 2025 de <https://www.open-bpmn.org/>
- White, S. A. y Miers, D. (2009). *Guía de referencia y modelado BPMN: comprendiendo y utilizando BPMN* (versión digital en español) [PDF]. Future Strategies Inc. Recuperado de https://users.dcc.uchile.cl/~nbaloian/DSS-DCC/Software/ModeladoBPMN.pdf?utm_source=chatgpt.com
- Bizagi Ltd. (s. f.). *Bizagi — plataforma de automatización de procesos empresariales* [Sitio web]. Recuperado el 21 de octubre del 2025 de <https://www.bizagi.com/es/plataforma/>
- ProcessMaker Inc. (s. f.). *ProcessMaker — herramienta de gestión de procesos de negocio y automatización de flujos de trabajo* [Sitio web]. Recuperado el 21 de octubre del 2025 de <https://www.processmaker.com/>
- Esmerald Publishing. (s. f.). *Business Process Management Journal*. <https://www.emerald.com/journal/bprmj>

XII. PERFIL DOCENTE

Licenciatura en Computación o áreas afines, preferentemente contar con experiencia en desarrollo en la industria. Tener al menos 2 años de experiencia en la docencia, con conocimientos en rediseño e ingeniería de procesos. Ser una persona creativa, tener un eficiente manejo grupal, fomentar el pensamiento crítico y reflexivo en sus estudiantes.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada.
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Redes de Datos
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 01 Horas Laboratorio: 02 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Disciplinaria
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Evelio Martínez Martínez Gerardo Tovar Ramos Adrián Enciso Almanza
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Proporcionar a cada estudiante los conocimientos esenciales sobre los principios, protocolos y estándares que sustentan las redes de datos, así como el manejo de herramientas de análisis y observación del tráfico, para aplicarlos en el diseño, implementación y diagnóstico de sistemas computacionales y comunicación entre servicios de manera eficiente y segura. Esta asignatura es obligatoria, forma parte de la etapa disciplinaria y pertenece al área de conocimiento de Redes.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Aplicar los principios fundamentales teóricos y prácticos de redes de datos para diseñar, configurar y diagnosticar infraestructuras de comunicación, utilizando estándares y metodologías vigentes, con el propósito de garantizar la eficiencia, seguridad y disponibilidad de la información, actuando con responsabilidad, ética profesional, pensamiento crítico, colaboración y compromiso con la calidad.

IV. EVIDENCIA DE APRENDIZAJE

Desarrollo de un sistema IoT (*InternetOfThings*) de monitoreo en red, en el cual el estudiante integre sensores conectados a *Raspberry Pi* para recolectar datos (ej. temperatura, luz, movimiento), transmita la información mediante protocolos de red (*MQTT*, *CoAP*, *REST/gRPC* o *Redis Streams*), y los almacene en un backend que permita su consulta. El sistema deberá incluir la instrumentación de métricas de desempeño, la captura y análisis de tráfico con *Wireshark*, y la construcción de un *dashboard* en *Grafana* con bitácora histórica de los eventos registrados.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Desarrollar software de base eficiente y robusto mediante la integración de conocimientos en arquitectura y redes de computadoras y los fundamentos de compiladores y sistemas operativos, para optimizar el rendimiento, la gestión de recursos y la comunicación entre sistemas, asegurando aplicaciones funcionales, seguras, escalables y sostenibles, con una actitud ética, responsable y visionaria.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Inclusión, Vanguardia e Innovación Social.

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos de redes desde el interior de los datos

Competencia

Analizar el flujo extremo a extremo de la información en redes de datos mediante la captura, interpretación y evaluación de paquetes de protocolos vigentes en las comunicaciones de red actuales, aplicando modelos *OSI* y *TCP/IP*, y utilizando herramientas de captura y análisis de tráfico para identificar problemas básicos de conectividad y desempeño, con responsabilidad ética y pensamiento crítico.

Contenido

- 1.1. Conceptos básicos de redes de datos
 - 1.1.1. Concepto de una red.
 - 1.1.2. Tipos de redes según la información que transmiten.
 - 1.1.3. Tipos de redes según su cobertura (*PAN*, *LAN*, *CAN*, *MAN* y *WAN*).
- 1.2. Concepto de comunicación de datos y flujo extremo a extremo
 - 1.2.1. Encapsulamiento de información: tramas, paquetes, segmentos y datagramas.
 - 1.2.2. Referencias conceptuales de la teoría de la información (*Nyquist*, *Shannon*).
- 1.3. Topologías y dispositivos de red.
 - 1.3.1. Topologías físicas y lógicas.
 - 1.3.2. Dispositivos de interconexión: *hub*, *switch*, *router*, *access point*, *TAP*.
- 1.4. Modelos de referencia
 - 1.4.1. Modelo OSI.
 - 1.4.2. Modelo TCP/IP.
- 1.5. Protocolos de nivel de enlace y red
 - 1.5.1. Ethernet II.
 - 1.5.2. VLAN (IEEE 802.1Q).
 - 1.5.3. IPv4, IPv6.
 - 1.5.4. ARP, *Neighbor Discovery*.
- 1.6. Protocolos de control y transporte
 - 1.6.1. ICMPv4, ICMPv6.
 - 1.6.2. TCP.
 - 1.6.3. UDP.
 - 1.6.4. QUIC.
- 1.7. Protocolos de aplicación
 - 1.7.1. DNS.
 - 1.7.2. DHCP.
 - 1.7.3. HTTP/1.1, HTTP/2, HTTP/3.
 - 1.7.4. TLS 1.3.
- 1.8. Observación y métricas de red
 - 1.8.1. Latencia y RTT.
 - 1.8.2. Throughput.
 - 1.8.3. Pérdida de paquetes, *jitter*, fragmentación/PMTU.
 - 1.8.4. Métodos de diagnóstico: *Wireshark*, *tcpdump*.

- 1.8.5. Interpretación de métricas con base en fundamentos teóricos.
- 1.9. Direccionamiento IP
 - 1.9.1. Definición Dirección IP.
 - 1.9.2. Direccionamiento IPv4.
 - 1.9.3. Direccionamiento IPv6.
 - 1.9.4. Ejercicios de cálculo de subredes.

Duración

08 horas

Unidad II. Comunicación en la red: APIs y servicios

Competencia

Aplicar modelos y protocolos fundamentales de comunicación en red, con enfoque en el diseño, implementación, evaluación de aplicaciones y servicios informáticos que gestionen la transmisión de datos de manera confiable y segura, mediante el uso de tecnologías y herramientas comunes en redes, actuando con responsabilidad, ética profesional y compromiso con la calidad.

Contenido

- 2.1. Modelos de comunicación en red
 - 2.1.1. Cliente/servidor.
 - 2.1.1.1. *Sockets* y FIFOs.
 - 2.1.2. *Peer-to-peer* (P2P): conceptos y aplicaciones actuales.
 - 2.1.3. Publicación/suscripción (pub/sub): mensajería y manejo de eventos.
 - 2.1.4. Comunicación asincrónica y arquitecturas *event-driven*.
 - 2.1.5. *Multicast* y *broadcast*: comunicación uno-a-muchos.
- 2.2. Servicios de aplicación en la red
 - 2.2.1. DNS.
 - 2.2.2. DHCP.
 - 2.2.3. HTTP/1.1, HTTP/2, HTTP/3.
 - 2.2.4. TLS 1.3: seguridad extremo a extremo en servicios.
- 2.3. APIs y servicios web
 - 2.3.1. REST: principios, idempotencia, cache, compresión.
 - 2.3.2. gRPC: serialización con *Protocol Buffers*.
 - 2.3.3. *GraphQL*: visión conceptual y casos de uso.
- 2.4. Flujos de datos y mensajería
 - 2.4.1. *Redis Streams*: práctica aplicada.
 - 2.4.2. *Apache Kafka* y *Apache Pulsar*: visión conceptual.
- 2.5. Almacenamiento en red
 - 2.5.1. *S3/MinIO*.
 - 2.5.2. Verificación de integridad de objetos.
- 2.6. Infraestructura de servicios
 - 2.6.1. *Reverse proxy* y balanceadores (*Nginx/Envoy*).
 - 2.6.2. Estrategias de *rate limiting*.

- 2.6.3.Cache en servicios web.
- 2.7. Monitoreo básico de servicios web

Duración

06 horas

Unidad III. Redes inalámbricas y dispositivos conectados

Competencia

Implementar mecanismos de comunicación en redes inalámbricas y entornos de dispositivos conectados, considerando los modelos de interacción entre nodos, *gateways* y servicios, así como su integración con arquitecturas en la nube y en el borde, con el propósito de diseñar y evaluar aplicaciones conectadas que sean funcionales, seguras y escalables, actuando con responsabilidad, ética profesional y compromiso con la calidad.

Contenido

- 3.1.Redes inalámbricas de uso general
 - 3.1.1.Wi-Fi (IEEE 802.11): fundamentos y seguridad.
 - 3.1.2.*Bluetooth* y *Bluetooth Low Energy* (BLE): aplicaciones en wearables.
 - 3.1.3.Redes personales (PAN) y entornos inteligentes.
- 3.2.Protocolos y servicios en IoT
 - 3.2.1.MQTT: QoS, tópicos, *bridges* HTTP.
 - 3.2.2.CoAP: protocolo ligero.
 - 3.2.3.Ejemplos de telemetría y control remoto.
- 3.3.Arquitecturas de comunicación
 - 3.3.1.Dispositivo → gateway → servicio → nube/edge.
 - 3.3.2.Arquitecturas híbridas.
- 3.4.Edge computing
 - 3.4.1.Mini-inferencia en contenedores ligeros.
 - 3.4.2.Envío de métricas hacia la nube.
 - 3.4.3.Dashboards y pipelines de datos.
- 3.5.Seguridad y privacidad en IoT
 - 3.5.1.Autenticación y gestión de credenciales.
 - 3.5.2.Cifrado de mensajes.
 - 3.5.3.Consideraciones éticas y riesgos en redes personales.
- 3.6.Panorama de protocolos industriales
 - 3.6.1.Modbus.
 - 3.6.2.OPC UA.
 - 3.6.3.EtherCAT.

Duración

06 horas

Unidad IV. Principios de comunicación distribuida y seguridad básica

Competencia

Implementar comunicación entre múltiples servicios aplicando principios básicos de coordinación y herramientas de monitoreo para desarrollar aplicaciones conectadas confiables, utilizando técnicas de observación para evaluar el desempeño de la comunicación de red, actuando con responsabilidad profesional y ética en el manejo de servicios distribuidos.

Contenido

- 4.1. Comunicación entre múltiples servicios
 - 4.1.1. Modelos de comunicación entre servicios (servicio-a-servicio).
 - 4.1.2. Coordinación básica con colas de mensajes (Redis Streams).
 - 4.1.3. Balanceo de carga simple entre instancias de servicio.
- 4.2. Implementación práctica con dispositivos de bajo costo
 - 4.2.1. Configuración de múltiples servicios con Raspberry Pi.
 - 4.2.2. Despliegue básico con contenedores (Docker).
 - 4.2.3. HTTP/HTTPS entre servicios distribuidos.
- 4.3.1. Principios de seguridad en comunicación de servicios
 - 4.3.2. Autenticación básica entre servicios (API keys, tokens).
 - 4.3.3. Implementación práctica de HTTPS/TLS entre servicios.
 - 4.3.4. Ética profesional en el monitoreo de comunicaciones.
 - 4.3.5. Observabilidad y métricas de comunicación
 - 4.3.6. Métricas básicas de red: latencia, throughput, disponibilidad.
 - 4.3.7. Implementación de logging centralizado para servicios.
 - 4.3.8. Monitoreo de comunicación HTTP con herramientas básicas.
- 4.4. Consideraciones prácticas y éticas
 - 4.4.1. Impacto de la latencia de red en aplicaciones multi-servicio.
 - 4.4.2. Manejo básico de interrupciones de comunicación.
 - 4.4.3. Consideraciones éticas en el monitoreo de servicios.

Duración:

08 horas

Unidad V. Tendencias actuales y futuras en redes

Competencia

Analizar las tendencias actuales y emergentes en redes, mediante el estudio de nuevas arquitecturas, tecnologías y metodologías, con el fin de adaptarse a entornos cambiantes y promover soluciones innovadoras, actuando con pensamiento crítico, adaptabilidad y compromiso con la calidad.

Contenido

- 5.1. Redes definidas por software (SDN).
- 5.2. Virtualización de funciones de red (NFV).
- 5.3. Internet de las Cosas (IoT networking).
- 5.5. Redes móviles 5G y más allá.
- 5.6. Edge Computing Networks .
- 5.5. Green Networking & Sustainability.

Duración

04 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

Unidad I. Fundamentos de redes desde el interior de los datos

Práctica 1: Explorando el flujo extremo a extremo con Wireshark

Duración:

06 horas

Procedimiento:

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Explica los objetivos de la práctica y repasar los modelos OSI y TCP/IP, relacionando cada capa con los protocolos que se observarán.
3. Verifica la conectividad del equipo (IP, gateway, DNS) y abrir Wireshark para seleccionar la interfaz de red adecuada.
4. Ejecuta comandos ping hacia distintos destinos (local, gateway, internet) y capturar el tráfico.
5. Observa cabeceras Ethernet, IPv4/IPv6 e ICMP y relacionar cada campo con la capa del modelo OSI correspondiente.
6. Accede a un sitio web no visitado previamente, observar el intercambio DNS en la captura e identifica direcciones IP de servidores resueltos.
7. Ejecuta una conexión TCP (ej. abrir un sitio en HTTP/1.1) y analizar el proceso de three-way handshake.
8. Compara con tráfico UDP (ej. consulta DNS) y discutir la ausencia de handshake.
9. Abre una página con HTTPS, identificar el inicio de la sesión TLS 1.3 y distinguir qué parte del intercambio está cifrada.
10. Compara con HTTP/2 o HTTP/3 si están disponibles en el sitio elegido.
11. Mide latencia y RTT a partir de las respuestas de ping.
12. Observa jitter y pérdida de paquetes en caso de aparecer durante la captura.
13. Relaciona los resultados con los fundamentos teóricos de *Nyquist* y *Shannon* sobre ancho de banda y ruido.
14. Conduce una reflexión final sobre cómo las observaciones se vinculan con la teoría y la importancia de interpretar métricas de red para diagnosticar. problemas.

15. Entrega reflexión para revisión y retroalimentación docente.

Recursos de apoyo:

- Wireshark/tcpdump
- Red local
- Acceso a internet controlado
- Guías de protocolos

Práctica 2: Construcción y diagnóstico de topologías de red

Duración

06 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Arma diferentes topologías (estrella con switch, bus con hub, y conexión con router) utilizando el material disponible en el laboratorio.
3. Asigna direccionamiento IPv4/IPv6 a los nodos y validar conectividad básica con ping y traceroute.
4. Captura el tráfico en cada escenario con *Wireshark*.
5. Identifica cabeceras ARP/ND, ICMP, direcciones MAC e IP involucradas.
6. Compara la circulación de paquetes en hub (broadcast) y en switch (unicast dirigido).
7. Detecta dispositivos de red intermedios (*gateway*, *DNS*, *router*) a partir de las tramas capturadas.
8. Infiere la topología en uso con base en las capturas (ejemplo: en estrella se observan solo origen/destino; en bus se observa difusión de tramas).
9. Mide métricas simples (tiempos de respuesta, RTT promedio) y discute cómo la topología y el dispositivo impactan en el desempeño.
10. Elabora un reporte con diagramas de topología, capturas anotadas y conclusiones sobre ventajas y desventajas de cada escenario.

Recursos de apoyo:

- Hubs
- Switches
- Router
- TAP casero
- Cables de red
- PCs/VMs
- Wireshark
- Simulador opcional (GNS3/Packet Tracer)

Práctica 3: Direccionamiento IPv4/IPv6

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Resolverá una serie de ejercicios de direccionamiento IPv4/IPv6, lo cual le permitirá comprender en la práctica los retos de migración y la importancia del direccionamiento para garantizar comunicación eficiente en redes actuales y futuras.
3. Entrega un documento en formato pdf de la práctica guiada que le proporcionará el maestro, incluyendo las capturas de pantalla indicadas.

Recursos de apoyo

- Ejercicios de subneteo de redes proporcionadas por el docente.
- Configure IP Addresses and Unique Subnets for New Users.
<https://www.cisco.com/c/en/us/support/docs/ip/routing-information-protocol-rip/13788-3.html>

Unidad II. Comunicación en la red: APIs y servicios

Práctica 4: Introducción a la comunicación cliente-servidor con sockets en Python

Duración

04 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Explica los conceptos básicos de cliente/servidor, TCP y UDP, y cómo se reflejan en el modelo OSI/TCP-IP.
3. Crea un servidor TCP simple en Python que escuche en un puerto definido.
4. Crea un cliente TCP que envíe y reciba mensajes desde el servidor y observe la conexión establecida (three-way handshake).
5. Repite el ejercicio con *sockets UDP*, resaltando la ausencia de conexión y la diferencia en confiabilidad.
6. Captura ambos casos con Wireshark para identificar cabeceras, puertos de origen/destino y diferencias de flujo.
7. Elabora una reflexión sobre las implicaciones de cada protocolo en la construcción de servicios.

Recursos de apoyo

- Python 3
- IDE (VS Code o similar)
- Wireshark
- Ejemplos de código provistos por el docente

- Red local

Unidad IV. Fundamentos de Redes de Datos para Sistemas Distribuidos Seguros

Práctica 5: Simulación de fallas de red con tc/netem

Duración

04 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Introduce el concepto de limitaciones de red (latencia, pérdida, jitter, ancho de banda) y su impacto en sistemas distribuidos.
3. Configura un servicio simple en el cluster (por ejemplo, servidor de claves/valores con *Redis*).
4. Aplica reglas de tc/netem en uno de los nodos para simular condiciones adversas: retrasos fijos, pérdida aleatoria de paquetes y limitación de ancho de banda.
5. Observa el comportamiento del cliente al acceder al servicio bajo dichas condiciones.
6. Captura métricas de tiempo de respuesta y tasa de error.
7. Relaciona los resultados con las falacias de la computación distribuida y el teorema CAP.
8. Elabora conclusiones sobre la importancia del diseño resiliente.

Recursos de apoyo

- Cluster de Raspberry Pi
- Linux tc/netem
- Redis/Memcached
- Python cliente
- Wireshark opcional

IX. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad II. Comunicación en la red: APIs y servicios

Práctica 1: Métricas de consumo en la captura de Pokémon en reposo y a distancia

Duración

05 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Configura el entorno de desarrollo instalando Python 3, un IDE de preferencia y las librerías necesarias (``requests``, ``grpcio``, ``grpcio-tools``).

3. Realiza peticiones GET a la PokéAPI para consultar información de un Pokémon específico, verificar el código de estado y explorar los datos devueltos (nombre, habilidades, peso, movimientos).
4. Prueba consultas adicionales modificando el nombre del Pokémon en la URL y observar los cambios en las respuestas.
5. Configura un endpoint local con Flask o FastAPI para simular un POST en JSON, enviar un Pokémon inventado y verificar la respuesta del servidor.
6. Compila un archivo `.proto` de ejemplo con `grpc_tools.protoc` y generar el cliente en Python.
7. Implementa a un cliente gRPC que invoque un método definido en el servicio y observar la respuesta.
8. Captura con Wireshark las solicitudes REST a la PokéAPI, identificando cabeceras HTTP y cuerpo en JSON.
9. Captura con Wireshark la comunicación gRPC y observar la transmisión en formato binario (Protobuf).
10. Compara legibilidad, eficiencia y estructura de los mensajes entre REST y gRPC considerando métricas de consumo y rendimiento.
11. Elabora un reporte que incluya ejemplos de peticiones y respuestas en REST y gRPC, capturas de Wireshark con anotaciones y conclusiones sobre ventajas, limitaciones y escenarios de uso de cada enfoque.

Recursos de apoyo

- Python 3
- Librerías `requests`
- `grpcio`
- `grpcio-tools`
- PokéAPI (<https://pokeapi.co>)
- Entorno local con Flask/FastAPI para POST
- Wireshark

Práctica 2: Batallas Pokémon en el gimnasio Redis Stream

Duración

03 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.

2. Instala Redis en Raspberry Pi o VM y verificar el servicio con redis-cli ping.
3. Crea y activa un entorno virtual en Python e instala las librerías necesarias (`redis`, `requests`).
4. Consulta la PokéAPI para obtener información de un Pokémon y seleccionar uno de sus movimientos disponibles.
5. Publica eventos en un stream de Redis llamado pokemon_battle, utilizando como campos el nombre del Pokémon, el movimiento seleccionado y la marca de tiempo de publicación.
6. Implementa un consumidor que lea eventos desde el stream y registre la latencia de entrega en función de la marca de tiempo.
7. Prueba el modelo pub/sub en Redis creando un canal denominado pokemon_channel, publicar mensajes de ejemplo y recibirlos en tiempo real con un cliente suscriptor.
8. Compara el comportamiento entre streams (históricos y ordenados) y pub/sub (efímeros y solo en vivo).
9. Elabora un reporte con capturas de pantalla y conclusiones que incluyan: eventos publicados, latencias promedio, diferencias entre ambos modelos y escenarios de aplicación.

Recursos de apoyo

- Redis en Raspberry Pi o VM
- Python 3 con librerías `redis` y `requests`
- Conexión a Internet para la PokéAPI
- Terminal
- Editor de texto o IDE

Unidad III. Redes inalámbricas y dispositivos conectados

Práctica 3: Comparativo de seguridad en protocolos Wi-Fi

Duración

04 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Configura un router inalámbrico en modo abierto y conecta un cliente para capturar tráfico con Wireshark, identificando información transmitida sin cifrado.
3. Configura el router en modo WEP, conectar un cliente y capturar tráfico con Wireshark; aplicar aircrack-ng en un entorno controlado para descifrar la clave y verificar el acceso a datos en claro.
4. Configura el router en modo WPA, captura tráfico y compara el nivel de dificultad de acceso a los datos en relación con WEP.
5. Configura finalmente el router en modo WPA2, captura tráfico y constatar que aun disponiendo de la clave el contenido de la comunicación permanece protegido.
6. Analiza y documenta la evolución de la seguridad en los protocolos Wi-Fi, destacando la importancia de adoptar WPA2 como estándar mínimo en redes inalámbricas.

7. Elabora un reporte con capturas de Wireshark, resultados de pruebas con aircrack-ng en WEP, y un cuadro comparativo de fortalezas y debilidades de cada protocolo.

Recursos de apoyo:

- Router inalámbrico doméstico configurable
- Raspberry Pi o computadoras cliente con tarjeta Wi-Fi
- Wireshark
- Suite aircrack-ng

Práctica 4: Definición en penaltis: MQTT vs CoAP

Duración

04 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Configura un broker MQTT en Raspberry Pi utilizando Mosquitto.
3. Monta un circuito en protoboard con LEDs para simular el semáforo de inicio de tiro, un servomotor para indicar dirección y un motor DC controlado por el integrado L293D para simular la fuerza del disparo.
4. Construye una maqueta simple de portería y pelota que permita visualizar los movimientos de los actuadores durante los intentos de penalti.
5. Programa un cliente publicador en Python que envíe eventos al broker registrando datos como inicio de jugada, dirección del tiro, fuerza aplicada y resultado (gol o fallo).
6. Configura un cliente suscriptor que reciba los eventos en tiempo real y almacena los datos en un archivo de bitácora para conformar un dataset con los intentos de penalti.
7. Implementa el mismo flujo de eventos utilizando CoAP con libcoap, enviando solicitudes POST para registrar cada tiro y GET para consultar el estado de la bitácora desde el servidor.
8. Captura el tráfico con Wireshark para observar el tamaño de los mensajes y el formato de transmisión en cada protocolo.
9. Mide métricas de desempeño, incluyendo latencia promedio en la entrega de eventos, overhead en bytes por mensaje y confiabilidad de transmisión.
10. Compara los resultados obtenidos en ambos protocolos para identificar cuál es más eficiente para transmisión ligera y cuál ofrece mejor soporte para consultas de estado.
11. Elabora un reporte que incluya capturas de ejecución en consola, ejemplos de mensajes transmitidos, métricas de latencia y tamaño, y conclusiones sobre los escenarios de uso adecuados para MQTT y CoAP en aplicaciones IoT.

Recursos de apoyo

- Raspberry Pi
- Broker Mosquitto
- Librería Paho MQTT para Python
- Librería libcoap
- LEDs

- Resistencias
- Servomotor
- Motor DC
- Integrado L293D
- Protoboard jumper cables
- Breadboard
- Maqueta de portería y pelota
- Fuente de alimentación
- Wireshark

Práctica 5: Análisis de la final de penaltis con grafana

Duración

04 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Instala y configura Prometheus y Grafana en Raspberry Pi o VM.
3. Instrumentar el suscriptor de la práctica de penaltis para exponer métricas en formato Prometheus utilizando la librería ``prometheus_client``, incluyendo contadores de goles y fallos, conteo por dirección del tiro, histograma de latencia de entrega y tasa de eventos por minuto.
4. Publica las métricas mediante un endpoint HTTP ``/metrics`` en el suscriptor y verificar su accesibilidad desde la red local.
5. Configura Prometheus para recolectar las métricas del suscriptor, agregando el ``job`` correspondiente en ``prometheus.yml`` y reiniciar el servicio.
6. Genera una tanda de penaltis con el sistema IoT de la práctica anterior para poblar las métricas con un número representativo de eventos.
7. Crea en Grafana un dashboard con paneles que emulen un noticiario deportivo: marcador en vivo (goles vs fallos), serie temporal de tasa de penaltis por minuto, histograma de latencia de entrega, distribución por dirección del tiro (izquierda, centro, derecha) y porcentaje de conversión con indicadores visuales.
8. Compara visualmente el desempeño al transmitir eventos mediante MQTT y al transmitirlos con CoAP, registrando diferencias en latencia, overhead y confiabilidades observadas en los paneles.
9. Elabora un reporte con capturas del dashboard, explicación de cada métrica, interpretación en estilo de análisis post-partido y conclusiones sobre el protocolo más apto según las necesidades observadas.
10. Instala y configura Prometheus y Grafana en Raspberry Pi o VM.

11. Instrumenta el suscriptor de la práctica de penaltis para exponer métricas en formato Prometheus utilizando la librería ``prometheus_client``, incluyendo contadores de goles y fallos, conteo por dirección del tiro, histograma de latencia de entrega y tasa de eventos por minuto.
12. Publica las métricas mediante un endpoint HTTP ``/metrics`` en el suscriptor y verificar su accesibilidad desde la red local.
13. Configura Prometheus para recolectar las métricas del suscriptor, agregando el ``job`` correspondiente en ``prometheus.yml`` y reiniciar el servicio.
14. Genera una tanda de penaltis con el sistema IoT de la práctica anterior para poblar las métricas con un número representativo de eventos.
15. Crea en Grafana un dashboard con paneles que emulen un noticiario deportivo: marcador en vivo (goles vs fallos), serie temporal de tasa de penaltis por minuto, histograma de latencia de entrega, distribución por dirección del tiro (izquierda, centro, derecha) y porcentaje de conversión con indicadores visuales.
16. Compara visualmente el desempeño al transmitir eventos mediante MQTT y al transmitirlos con CoAP, registrando diferencias en latencia, overhead y confiabilidades observadas en los paneles.
17. Elabora un reporte con capturas del dashboard, explicación de cada métrica, interpretación en estilo de análisis post-partido y conclusiones sobre el protocolo más apto según las necesidades observadas.
18. Entrega reporte para revisión y retroalimentación docente.

Recursos de apoyo

- Raspberry Pi o VM
- Prometheus
- Grafana
- Broker Mosquitto
- Librería ``prometheus_client`` para Python
- Sistema IoT de penaltis de la práctica anterior
- Conexión de red local

Unidad IV: Fundamentos de Redes de Datos para Sistemas Distribuidos Seguros

Práctica 6: Impacto de fallas de red en un servicio distribuido

Duración

02 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.

2. Despliega un coordinador y dos o más trabajadores en un cluster de Raspberry Pi, utilizando Redis Streams como cola de tareas y Redis como almacén de resultados.
3. Genera un conjunto de hashes MD5 de prueba a partir de cadenas sintéticas con reglas acotadas (ocho caracteres, primera mayúscula, vocales sustituidas por números, último carácter especial), documentando el espacio de búsqueda.
4. Publica en la cola los rangos de búsqueda que cada trabajador debe explorar y verificar desde el cliente que el sistema resuelve coincidencias sobre el dataset sintético.
5. Aplica reglas de tc/netem en uno de los nodos para introducir latencia fija y pérdida aleatoria de paquetes durante la ejecución, observando tiempos de respuesta, tareas fallidas o reintentadas y posibles inconsistencias en la agregación de resultados.
6. Compara el comportamiento del sistema antes y después de inyectar fallas de red, identificando el impacto sobre disponibilidad y tiempo total de cómputo.
7. Elabora un reporte breve que incluya comandos principales, capturas de consola, tiempo total de resolución y observaciones sobre los efectos de red adversa en el desempeño del sistema distribuido.

Recursos de apoyo

- Cluster de Raspberry Pi
- Redis/Redis Streams
- Python
- Herramienta tc/netem
- Red local

Práctica 7: Resiliencia y observabilidad en sistemas distribuidos

Duración

04 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Configura múltiples servicios HTTP en Raspberry Pi: un servicio coordinador, un servicio de procesamiento de logs y un servicio de almacenamiento de resultados.
3. Implementa un sistema de procesamiento de logs web utilizando Redis Streams para distribuir archivos de log entre servicios y Redis para almacenar estadísticas procesadas.
4. Genera archivos de log sintéticos simulando tráfico web (formato Apache/Nginx) con diferentes códigos de estado HTTP, tiempos de respuesta y rutas de acceso.
5. Publica en la cola de Redis Streams los archivos de log que cada servicio trabajador debe procesar, verificando que las estadísticas (conteos por código de estado, promedios de tiempo de respuesta) se almacenen correctamente.
6. Instrumenta los servicios para exponer métricas HTTP en Prometheus utilizando la librería `prometheus_client`, incluyendo: solicitudes procesadas, errores de comunicación entre servicios, tiempo promedio de procesamiento por archivo y disponibilidad de cada servicio.

7. Configura políticas básicas de resiliencia con timeouts HTTP y reintentos automáticos para continuar la comunicación cuando un servicio se retrase o no responda.
8. Ejecuta el procesamiento de logs en condiciones normales y luego aplica reglas de tc/netem en uno de los nodos para introducir latencia o pérdida de paquetes, observando los efectos en la comunicación HTTP entre servicios.
9. Visualiza en Grafana un dashboard con paneles de: tasa de procesamiento por servicio, latencia de comunicación HTTP entre servicios, códigos de error HTTP y tiempo promedio de respuesta, comparando condiciones normales y con fallas simuladas.
10. Analiza los resultados obtenidos, relacionando los efectos de las fallas de red con la comunicación HTTP entre servicios y la importancia de implementar timeouts y reintentos.
11. Elabora un reporte que incluya capturas del dashboard, ejemplos de logs procesados, series temporales de métricas de comunicación y conclusiones sobre resiliencia en comunicación entre servicios.

Recursos de apoyo

- Cluster de Raspberry Pi
- Redis/Redis Streams
- Python con librería prometheus_client
- Lask/FastAPI para servicios HTTP
- Prometheus
- Grafana
- Herramienta tc/netem
- Red local

X. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer de forma clara y formal:

- La metodología de trabajo
- Criterios de evaluación y rúbricas
- Reglas sobre la calidad y entrega de trabajos académicos
- Derechos y obligaciones del docente y los estudiantes

Esto debe quedar registrado y comunicado por medios institucionales (como Blackboard o correo institucional).

Estrategia de enseñanza

- Estudio de caso
- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos

- Aprendizaje basado en preguntas
- Debates
- Discusión guiada
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Mapas mentales
- Mapas conceptuales
- Cuadros comparativos
- Infografías
- Reportes de prácticas

XI. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

- Evaluaciones parciales	30%
- Prácticas laboratorio/taller	40%
- Proyecto final	30%
Total	100%

XII. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Kleppmann, M. (2017). *Designing data-intensive applications: The big ideas behind reliable, scalable, and maintainable systems*. O'Reilly Media.

Kurose, J. F., & Ross, K. W. (2022). *Computer networking: A top-down approach* (8th ed.). Pearson.

Martínez, Evelio; Serrano, Arturo. (2012). *Fundamentos de telecomunicaciones y redes. Convergente*. [Clásica]. Tanenbaum, A. S., Feamster, N., & Wetherall, D. (2023). *Redes de computadoras* (1a ed.; V. Portillo Campos & J. A. Jiménez Gallegos, Trads.; S. Fuenlabrada Velázquez & E. M. Miranda Chávez, Revs. tec.). Pearson.

Parab, J. S., Lanjewar, M. G., Sequeira, M. D., Naik, G., & Shaikh, A. Y. (s.f.). *Python Programming Recipes for IoT Applications* [recurso electrónico]. Colección de Libros Electrónicos.

Recursos de acceso libre o complementarios externos

Bell, C. (2018). *Hands-on MQTT programming with Python: Work with the lightweight IoT protocol in Python*. Packt Publishing.

Bock, L. (2022). *Learn Wireshark: A definitive guide to expertly analyzing protocols and troubleshooting networks using Wireshark* (2nd ed.). Packt Publishing.

Docker Inc. (2024). *Docker Documentation*. <https://docs.docker.com/>

Eclipse Foundation. (s. f.). *Eclipse Mosquitto: An open source MQTT broker*. <https://mosquitto.org/documentation/>

Gonzalez, L. (2021). *Performance analysis of MQTT and CoAP protocols under security attacks*. IOTMADLAB. https://iotmadlab.es/wpcontent/uploads/2023/10/Performance_CN_2021.pdf

Grafana Labs. (2024). *Grafana documentation*. <https://grafana.com/docs/grafana/latest/>

Greer, C. [Chris Greer]. (s. f.). *Home* [Canal de YouTube]. 2025, de <https://www.youtube.com/chrisgreer>

HiveMQ. (s. f.). *MQTT Essentials: A lightweight IoT protocol*. <https://www.hivemq.com/mqtt-essentials/>

Nygard, M. T. (2018). *Release it! Design and deploy production-ready software* (2nd ed.). Pragmatic Bookshelf.

Pajankar, A. (2020). *Practical Python programming for IoT: Build advanced IoT projects using a Raspberry Pi 4, MQTT, RESTful APIs, WebSockets, and Python 3*. Packt Publishing.

Prometheus Authors. (2024). *Prometheus documentation*. <https://prometheus.io/docs/introduction/overview/>

Python Software Foundation. (s. f.). *Socket Programming HOWTO*. <https://docs.python.org/3/howto/sockets.html>

Raspberry Pi Foundation. (2024). *Raspberry Pi Documentation*. <https://www.raspberrypi.org/documentation/>

Redis Ltd. (s. f.). *Redis Documentation*. <https://redis.io/docs/>

REST API Tutorial. (s. f.). *REST API Tutorial*. <https://restfulapi.net/>

Wireshark Foundation. (2025). *Wireshark User's Guide*. https://www.wireshark.org/docs/wsug_html_chunked/

Wireshark desde Cero - Curso 2025. (2025). *Wireshark desde Cero - Curso 2025* [Video]. YouTube. <https://www.youtube.com/watch?v=l7zIZ1bQuQQ>

XIII. PERFIL DOCENTE

La persona docente que imparta la asignatura de Redes de Datos debe contar con formación profesional en áreas afines a las Tecnologías de la Información y Comunicaciones, como Ciencias Computacionales, Ingeniería en Computación o Ingeniería Electrónica, preferentemente con estudios de posgrado. Debe poseer experiencia práctica en diseño, implementación y administración de redes LAN y Wi-Fi, dominio de protocolos de comunicación (TCP/IP, IPv4/IPv6), enrutamiento, switching y seguridad en redes, así como en el uso de herramientas de diagnóstico y simulación (*Wireshark*, *Packet Tracer*, *GNS3*).

Es esencial que tenga habilidades en sistemas operativos Linux para la configuración de servicios y la operación de laboratorios con *Raspberry Pi*, además de conocimientos en virtualización, IoT y entornos distribuidos. Asimismo, debe contar con competencias pedagógicas para fomentar el aprendizaje activo, la resolución de problemas y el trabajo colaborativo.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Graficación
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 03 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Disciplinaria
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Everardo Gutiérrez López Antonio de Jesús García Chavez Sergio Omar Infante Prieto
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

El propósito de la unidad de aprendizaje es que el estudiantado desarrolle modelos, algoritmos y visualizaciones gráficas en 2D y 3D, aplicando fundamentos matemáticos, técnicas de programación gráfica, metodologías de visualización científica y herramientas de GPU, para representar datos, fenómenos y estructuras computacionales con precisión, eficiencia y claridad, actuando con ética profesional, pensamiento crítico, creatividad y responsabilidad técnica. Se imparte en la etapa disciplinaria, es de carácter obligatorio y pertenece al área de conocimiento de Interacción Hombre Máquina.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Desarrollar soluciones de graficación computacional que representen, modelen y visualicen información y fenómenos tridimensionales, mediante la utilización de principios matemáticos, técnicas de programación gráfica y herramientas especializadas, para analizar, comunicar y soportar procesos de investigación o desarrollo tecnológico, con pensamiento crítico, responsabilidad, creatividad y apego a la ética profesional.

IV. EVIDENCIA DE APRENDIZAJE

1. Portafolio de prácticas de laboratorio, que incluya las prácticas de laboratorio realizadas a lo largo del curso. Las prácticas entregadas deberán tener un funcionamiento correcto y eficiente, una explicación clara del procedimiento, incluir análisis visual y técnico de los resultados, y reflejar actitudes profesionales como orden, ética y responsabilidad.
2. Proyecto integrador final donde se diseñe e implemente una solución de graficación o visualización, que integre: Modelado y representación geométrica, Iluminación y render, Visualización avanzada o GPU computing, Análisis fundamentado del problema.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Desarrollar soluciones que optimicen procesos y recursos, mediante la aplicación de modelado matemático y simulación de problemas complejos, para mejorar la eficiencia en la resolución de problemas e impulsar la innovación tecnológica y científica, con creatividad y perseverancia y manteniendo un enfoque ético en la resolución de problemas.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia, Vanguardia e Innovación Social

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos de la Graficación Computacional

Competencia

Aplicar los fundamentos históricos, conceptuales y matemáticos de la graficación computacional, mediante la utilización de transformaciones, sistemas de coordenadas y el pipeline gráfico, para interpretar, estructurar y generar escenas básicas en 2D y 3D, con curiosidad científica, rigor intelectual y disposición al aprendizaje autónomo.

Contenido

- 1.1. Perspectiva histórica de la graficación
 - 1.1.1. Primeras interfaces gráficas y Sketchpad
 - 1.1.2. Algoritmos pioneros: líneas, polígonos, rellenos
 - 1.1.3. Evolución del hardware gráfico: rasterización → GPUs → pipeline programable
 - 1.1.4. Influencia en investigación científica, simulación y análisis visual
- 1.2. Conceptos fundamentales
 - 1.2.1. Definición y dominios de la graficación
 - 1.2.2. Representaciones geométricas básicas en 2D y 3D
 - 1.2.3. Sistemas de coordenadas y transformaciones elementales
 - 1.2.4. Cámaras virtuales y visión computacional básica
- 1.3. Matemáticas esenciales
 - 1.3.1. Transformaciones lineales: vectores y matrices
 - 1.3.2. Rotaciones, traslaciones y escalamiento
 - 1.3.3. Proyecciones: ortográfica y perspectiva
 - 1.3.4. Interpolación, curvas y fundamentos paramétricos
- 1.4. Pipeline de graficación
 - 1.4.1. Pipeline fijo: etapas y flujo de datos
 - 1.4.2. Pipeline programable: vertex y fragment shaders
 - 1.4.3. Procesamiento en GPU: VAOs, VBOs y shaders
 - 1.4.4. Rasterización y conversión final a píxeles
- 1.5. Herramientas y entornos
 - 1.5.1. OpenGL moderno y principios de su API
 - 1.5.2. GLSL y compilación de shaders
 - 1.5.3. Bibliotecas de apoyo: GLFW, GLM, stb
 - 1.5.4. Herramientas de visualización científica: VTK, ParaView (panorama general)

Duración

08 horas

Unidad II. Modelado y Representación Geométrica

Competencia

Construir objetos, superficies y datos tridimensionales mediante mallas, estructuras geométricas, texturas y técnicas de optimización, empleando algoritmos y herramientas de

graficación, para crear representaciones coherentes de modelos computacionales, con precisión, orden, pensamiento analítico y responsabilidad técnica.

Contenido

- 2.1. Modelos y estructuras geométricas
 - 2.1.1. Mallas poligonales: vértices, aristas y caras
 - 2.1.2. Normales, tangentes y orientación de superficies
 - 2.1.3. Curvas paramétricas: Bézier, splines
 - 2.1.4. Superficies paramétricas y NURBS
 - 2.1.5. Optimización en el modelado
- 2.2. Representación de datos y estructuras 3D
 - 2.2.1. Mallas estructuradas y no estructuradas
 - 2.2.2. Representación volumétrica y voxels
 - 2.2.3. Campos escalares y vectoriales
 - 2.2.4. Modelado basado en datos científicos
- 2.3. Texturas y materiales
 - 2.3.1. Mapeo UV y coordenadas de textura
 - 2.3.2. Texturas 2D, 3D y procedurales
 - 2.3.3. Materiales básicos: Phong, Blinn–Phong
 - 2.3.4. Mapas de color aplicados a datos

Duración

06 horas

Unidad III. Iluminación, Sombreado y Renderizado

Competencia

Implementar técnicas de iluminación, sombreado y renderizado mediante la utilización de *shaders*, modelos matemáticos y algoritmos avanzados de graficación, para producir imágenes visualmente correctas y técnicamente robustas en tiempo real o por trazado, con apego a estándares de calidad, ética profesional, creatividad y compromiso con la excelencia.

Contenido

- 3.1. Iluminación
 - 3.1.1. Modelos de iluminación locales
 - 3.1.2. Sombreado plano, Gouraud y Phong
 - 3.1.3. Normales, gradientes y efectos de superficie
 - 3.1.4. Introducción a modelos físicamente basados (PBR)
- 3.2. Técnicas de renderizado
 - 3.2.1. Renderizado por rasterización: técnicas avanzadas
 - 3.2.2. Fundamentos de ray tracing
 - 3.2.3. Path tracing simplificado
 - 3.2.4. Transparencia, blending y efectos atmosféricos
- 3.3. Visualización en tiempo real
 - 3.3.1. Optimización de *shaders*

- 3.3.2. Control de cámara y navegación 3D
- 3.3.3. Interactividad en escenas complejas
- 3.3.4. Técnicas de aceleración en GPU
- 3.4. Percepción visual
 - 3.4.1. Espacios de color perceptuales
 - 3.4.2. Mapas de color y legibilidad visual
 - 3.4.3. Profundidad, contraste y claridad
 - 3.4.4. Consideraciones perceptuales en análisis visual

Duración

06 horas

Unidad IV. Visualización Científica y Manejo de Datos

Competencia

Analizar datos científicos mediante la utilización de técnicas de visualización volumétrica, pipelines de análisis visual y herramientas especializadas, para interpretar estructuras y patrones relevantes en información compleja, con objetividad, pensamiento crítico, responsabilidad y claridad.

Contenido

- 4.1. Conceptos de visualización científica
 - 4.1.1. Objetivos de la visualización en ciencia e ingeniería
 - 4.1.2. Visualización de datos multidimensionales
 - 4.1.3. Técnicas de análisis visual interactivo
 - 4.1.4. Ética y fidelidad visual en resultados científicos
- 4.2. Volúmenes y datos 3D
 - 4.2.1. Renderizado volumétrico
 - 4.2.2. Ray marching y funciones de transferencia
 - 4.2.3. Visualización de simulaciones
 - 4.2.4. Extracción de isosuperficies
- 4.3. Pipelines para visualización de grandes volúmenes de datos
 - 4.3.1. Filtrado y reducción de datos
 - 4.3.2. Preprocesamiento y normalización
 - 4.3.3. Estrategias para visualización eficiente
 - 4.3.4. Integración con GPU para acelerar análisis científico
- 4.4. Herramientas de visualización
 - 4.4.1. VTK: conceptos y componentes
 - 4.4.2. ParaView y flujos de visualización
 - 4.4.3. Bibliotecas 3D en Python
 - 4.4.4. Integración con lenguajes de análisis

Duración

07 horas

Unidad V. Aplicaciones Avanzadas y Proyecto Final

Competencia

Desarrollar soluciones avanzadas de graficación aplicadas a problemas reales, mediante la incorporación de computación con unidades gráficas de procesamiento, entornos inmersivos o visualización interactiva, para elaborar un proyecto integrador que permita resolver necesidades científicas o tecnológicas, con liderazgo, colaboración, creatividad y compromiso ético con el impacto de sus desarrollos.

Contenido

- 5.1. Graficación en ciencias computacionales
 - 5.1.1. Visualización en aprendizaje automático
 - 5.1.2. Graficación para simulación numérica
 - 5.1.3. Graficación para análisis de datos y sistemas complejos
 - 5.1.4. Tendencias e investigación actual
- 5.2. Computación en GPU
 - 5.2.1. Conceptos de GPGPU
 - 5.2.2. Introducción a CUDA / OpenCL
 - 5.2.3. Paralelización para visualización interactiva
 - 5.2.4. Uso de GPU en aplicaciones científicas
- 5.3. Entornos inmersivos
 - 5.3.1. Fundamentos de VR/AR
 - 5.3.2. Sistemas de coordenadas y proyecciones para entornos inmersivos
 - 5.3.3. Interfaces y dispositivos de interacción
 - 5.3.4. Visualización inmersiva aplicada a investigación
- 5.4. Proyecto integrador
 - 5.4.1. Definición del fenómeno o problema computacional a visualizar
 - 5.4.2. Diseño del pipeline gráfico
 - 5.4.3. Implementación con técnicas avanzadas
 - 5.4.4. Presentación y evaluación de resultados

Duración

05 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Fundamentos de la graficación computacional

Práctica 1. Visualización de transformaciones básicas en 2D y 3D

Duración:

04 horas

Procedimiento:

1. Atiende las instrucciones de su docente.
2. Configura un entorno mínimo con OpenGL, GLFW y GLAD/GLM.
3. Crea una ventana y un contexto gráfico básico.
4. Dibuja un conjunto simple de primitivas (triángulo, cuadrado).
5. Implementa transformaciones básicas: traslación, rotación y escalamiento.
6. Muestra el efecto de cada transformación mediante interacción por teclado.
7. Genera capturas de pantalla y documenta los resultados.
8. Entrega reporte final y recibe retroalimentación docente.

Recursos de apoyo:

- OpenGL 3.3 o superior
- Bibliotecas: GLFW, GLAD/GLM
- Tutoriales oficiales de OpenGL
- Documento guía del pipeline gráfico.

Práctica 2. Exploración del pipeline programable con shaders

Duración:

03 horas

Procedimiento:

1. Atiende las instrucciones de su docente.
2. Crea un proyecto con soporte para shaders.
3. Implementa un shader de vértices y uno de fragmentos mínimos.
4. Pasa atributos de posición y color desde la CPU a la GPU.
5. Visualiza cómo influyen los shaders en la salida final.
6. Modifica parámetros del shader en tiempo real.
7. Elabora un breve reporte explicando el flujo CPU → GPU.
8. Entrega reporte final y recibe retroalimentación docente.

Recursos de apoyo:

- GLSL
- Depurador de shaders (RenderDoc, apitrace)
- Código base de su docente

Unidad II. Modelado y representación geométrica

Práctica 3. Construcción y Visualización de una Malla Poligonal

Duración:

04 horas

Procedimiento:

1. Atiende las instrucciones de su docente.
2. Crea una estructura de datos tipo “Mesh” con vértices, aristas y caras.
3. Carga una malla simple (teapot, bunny, cube) desde un archivo OBJ.
4. Calcula normales por vértice y por cara.
5. Visualiza la malla con y sin normales.
6. Analiza la calidad de la malla (densidad, uniformidad, continuidad).
7. Entrega reporte final y recibe retroalimentación docente.

Recursos de apoyo:

- Archivos OBJ simples
- Biblioteca Assimp (opcional)
- Visualizadores de referencia (MeshLab)

Práctica 4. Mapeo UV y Aplicación de Texturas

Duración:

06 horas

Procedimiento:

1. Atiende las instrucciones de su docente.
2. Carga un modelo con coordenadas UV.
3. Carga una textura sencilla (imagen JPG o PNG).
4. Aplica la textura mediante un shader modificado.
5. Prueba texturas 2D, patrones repetitivos y texturas procedurales simples.
6. Compara los resultados según calidad de UV mapping.
7. Entrega reporte final y recibe retroalimentación docente.

Recursos de apoyo:

- GLM para matrices, Imágenes simples de prueba
- Ejemplos de texturas (checkerboard, ruido, mapas).

Unidad III: Iluminación, sombreado y renderizado

Práctica 5. Implementación de Modelos de Iluminación

Duración:

04 horas

Procedimiento:

1. Atiende las instrucciones de su docente.
2. Prepara una escena con uno o dos objetos 3D.
3. Implementa sombreado plano, Gouraud y Phong utilizando shaders.
4. Compara visualmente los resultados.
5. Modifica la posición de la luz y observa el efecto.
6. Documenta diferencias en rendimiento y calidad.
7. Entrega reporte final y recibe retroalimentación docente.

Recursos de apoyo:

- GLSL
- Material de referencia sobre modelo de iluminación estándar

Práctica 6. Introducción al Ray Tracing CPU/GPU**Duración:**

06 horas

Procedimiento:

1. Atiende las instrucciones de su docente.
2. Implementa un trazador de rayos minimalista en CPU o compute shader.
3. Renderiza esferas básicas con iluminación difusa.
4. Agrega sombras sencillas (shadow rays).
5. Mide el tiempo de ejecución y comparar escenarios.
6. Documenta la arquitectura del mini-ray tracer.
7. Entrega reporte final y recibe retroalimentación docente.

Recursos de apoyo:

- Documentación de compute shaders o C++
- Artículos introductorios al ray tracing
- Código base de referencia

Unidad IV. Visualización científica y manejo de datos**Práctica 7. Visualización Científica y Manejo de Datos****Duración:**

06 horas

Procedimiento:

1. Atiende las instrucciones de su docente.
2. Carga un volumen simple (CT/MRI o dataset sintético).
3. Implementa un shader para ray marching básico.
4. Diseña una función de transferencia para resaltar estructuras.
5. Controla la interacción de cámara en 3D.

6. Evalúa la calidad visual mediante comparación con visualizadores externos.
7. Entrega reporte final y recibe retroalimentación docente.

Recursos de apoyo:

- Dataset volumétrico (CT-head, nebulosa, simulación CFD)
- Shader para muestreo 3D
- Herramientas: ParaView/VTK.

Práctica 8: Visualización y Análisis de Datos Multidimensionales

Duración:

03 horas

Procedimiento:

1. Atiende las instrucciones de su docente.
2. Carga un dataset multidimensional (por ejemplo: Iris, clima, sensores).
3. Implementa visualizaciones básicas: scatterplots, parallel coordinates.
4. Aplica color mapping para resaltar variables.
5. Implementa interacción: selección y filtrado.
6. Elabora análisis visual de los patrones detectados.
7. Entrega reporte final y recibe retroalimentación docente.

Recursos de apoyo:

- Python + Matplotlib/VisPy + NumPy
- Datasets públicos (UCI Machine Learning Repository)

Unidad V. Aplicaciones avanzadas y proyecto final

Práctica 9. Programación en GPU con CUDA u OpenCL

Duración:

06 horas

Procedimiento:

1. Atiende las instrucciones de su docente.
2. Crea un kernel simple para operar con vectores o matrices.
3. Acelera una operación gráfica simple (filtro de imagen o blur).
4. Compara tiempos de CPU vs GPU.
5. Integra el resultado en un pipeline gráfico sencillo.
6. Documenta el impacto de la paralelización.
7. Entrega reporte final y recibe retroalimentación docente.

Recursos de apoyo:

- CUDA Toolkit u OpenCL SDK
- Tarjeta gráfica compatible

- Ejemplos de kernels básicos

Práctica 10. Prototipo de Entorno Inmersivo o Interactivo 3D

Duración:

06 horas

Procedimiento:

1. Atiende las instrucciones de su docente.
2. Configura un entorno de desarrollo VR/AR o 3D interactivo (Unity, Unreal o WebGL).
3. Importa un modelo o dataset previamente trabajado en el curso.
4. Implementa interacciones básicas: navegación, selección o manipulación.
5. Integra iluminación y texturas mínimas.
6. Presenta el prototipo y discute posibles aplicaciones científicas.
7. Entrega reporte final y recibe retroalimentación docente.

Recursos de apoyo:

- Unity/Unreal Engine o WebXR
- Casco VR (opcional) o Simulador en escritorio
- Modelos previamente generados en el curso

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Técnica expositiva
- Discusión guiada
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Mapas conceptuales
- Cuadros comparativos
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	20%
– Portafolio de prácticas de laboratorio	50%
– Proyecto final	30%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Akenine-Möller, T., Haines, E., Hoffman, N., Pesce, A., Iwanicki, M., & Barré-Brisebois, C. (2018). *Real-Time Rendering* (4th ed.). CRC Press. [clásica]
- Angel, E., & Shreiner, D. (2019). *Interactive Computer Graphics: A Top-Down Approach with WebGL* (7th ed.). Pearson.[clásica]
- Ferwerda, J. (2019). *Visualization Analysis and Design*. CRC Press. [clásica]
- Schroeder, W., Martin, K., & Lorensen, B. (2018). *The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics* (5th ed.). Kitware. [clásica]
- Shirley, P., Marschner, S., Ashikhmin, M., Gleicher, M., Hoffman, N., & Ionescu, N. (2021). *Fundamentals of Computer Graphics* (5th ed.). CRC Press.

Recursos de acceso libre o complementarios externos

- ACM Special Interest Group on Computer Graphics and Interactive Techniques.
<https://www.acm.org/special-interest-groups/sigs/siggraph>
- Haine, T., Akenine-Möller, T., & Hoffman, N. (2019). *The Graphics Codex* (2nd ed.). Graphics Codex Press. [clásica]
- LaValle, S. (2017). *Virtual Reality*. Cambridge University Press. [clásica]
- Möller, T., & Trumbore, B. (2021). *Computer Graphics: Principles and Practice* (4th ed.). Addison-Wesley.

Pharr, M., Jakob, W., & Humphreys, G. (2023). *Physically Based Rendering: From Theory to Implementation* (4th ed.). MIT Press.

Sanders, J., & Kandrot, E. (2010). *CUDA by Example: An Introduction to General-Purpose GPU Programming*. Addison-Wesley. [clásica]

Ware, C. (2021). *Information Visualization: Perception for Design* (4th ed.). Morgan Kaufmann.

XII. PERFIL DOCENTE

Licenciatura en Ciencias Computacionales o áreas afines, preferentemente posgrado en el área de las ciencias computacionales. Experiencia profesional en docencia universitaria de al menos 2 años. Asimismo, deberá tener conocimiento sobre el desarrollo de aplicaciones gráficas, visualización científica, simulación o modelado, y competencia en programación de GPU. Debe poseer la habilidad para guiar procesos de diseño visual y razonamiento matemático, a la vez que fomenta la creatividad y el análisis y resolución de problemas.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Aprendizaje Automático
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 03 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos: 07
Etapas de formación:	Disciplinaria
Carácter:	Obligatoria
Requisito:	Minería de Datos Fundamentos de Inteligencia Artificial
Modalidad de impartición:	Presencial
Equipo de diseño:	José Ángel González Fraga Everardo Gutiérrez López Antonio de Jesús García Chávez
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Esta unidad de aprendizaje tiene como propósito brindar a cada estudiante los conceptos fundamentales, técnicas y métodos que sustentan el campo del aprendizaje automático, desarrollando habilidades para formular problemas de aprendizaje supervisado y no supervisado, preparar y analizar conjuntos de datos, seleccionar modelos apropiados, entrenarlos, validarlos y evaluar su desempeño mediante métricas especializadas. Asimismo, se fomentan actitudes críticas, éticas y responsables que fortalezcan el uso adecuado de algoritmos y prácticas de aprendizaje automático en la solución de problemas de la computación y disciplinas afines, considerando la calidad de los datos, la interpretación de resultados y el impacto social de los modelos predictivos. La unidad de aprendizaje es de carácter obligatorio, se encuentra en la etapa disciplinaria, pertenece al área de conocimiento de Interacción Hombre Máquina, y tiene como requisito haber aprobado la unidad de aprendizaje de Fundamentos de Inteligencia Artificial.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Evaluar modelos de aprendizaje supervisado y no supervisado mediante su entrenamiento, validación y análisis estadístico, mediante el uso de técnicas matemáticas, computacionales y métricas especializadas, con el fin de determinar su pertinencia, desempeño y adecuación a problemas de distintos ámbitos, con rigor, ética y responsabilidad en el manejo e interpretación de datos.

IV. EVIDENCIA DE APRENDIZAJE

Desarrolla un proyecto integrador de aprendizaje automático que incluya el planteamiento del problema, la preparación y análisis del conjunto de datos, la selección justificada de modelos supervisados o no supervisados, su entrenamiento y validación mediante las métricas estudiadas en la unidad de aprendizaje, así como la comparación y evaluación crítica del desempeño de los modelos. El proyecto deberá entregarse en un reporte técnico y una presentación o infografía digital que muestre la metodología empleada, los resultados, sus limitaciones, el uso ético de los datos y el impacto de la solución en un contexto aplicado, con lenguaje formal, claro y respaldado por referencias bibliográficas.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar y evaluar soluciones de cómputo inteligente aplicando principios de inteligencia artificial, minería de datos, e integración de herramientas computacionales, con el propósito de apoyar la toma de decisiones en situaciones complejas propias de contextos científicos, sociales, industriales o tecnológicos, actuando con responsabilidad ética, pensamiento crítico, y colaboración interdisciplinaria.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Compromiso con la responsabilidad social; Excelencia; Vanguardia e Innovación Social

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos matemáticos y conceptuales del aprendizaje automático

Competencia

Analizar los fundamentos matemáticos, estadísticos y conceptuales del aprendizaje automático para formular correctamente problemas supervisados y no supervisados, mediante la preparación de conjuntos de datos de problemáticas aplicadas con rigor técnico, objetividad y sentido crítico.

Contenido

- 1.1. Conceptos fundamentales del ML
 - 1.1.1. Aprendizaje supervisado, no supervisado y semi-supervisado
 - 1.1.2. Conjuntos de entrenamiento, validación y prueba
 - 1.1.3. Funciones de pérdida, riesgo empírico y riesgo esperado
 - 1.1.4. Generalización y sobreajuste (visión conceptual)
- 1.2. Fundamentos matemáticos
 - 1.2.1. Álgebra lineal esencial: vectores, matrices, operaciones básicas
 - 1.2.2. Probabilidad y estadística: distribuciones, esperanza, varianza
 - 1.2.3. Introducción al gradiente y al descenso por gradiente (conceptual)
- 1.3. Calidad y preparación de los datos
 - 1.3.1. Valores faltantes, ruido y outliers
 - 1.3.2. Normalización y estandarización
 - 1.3.3. División estratificada de conjuntos de datos
 - 1.3.4. Introducción al análisis exploratorio de datos (EDA)

Duración

06 horas

Unidad II. Modelos supervisados básicos

Competencia

Implementar y evaluar modelos supervisados básicos de clasificación y regresión, mediante la interpretación de su comportamiento a través de métricas y análisis de desempeño para seleccionar modelos apropiados acordes a la naturaleza del problema, con rigor analítico, pensamiento crítico, y colaboración interdisciplinaria.

Contenido

- 2.1. Modelos lineales
 - 2.1.1. Regresión lineal: formulación, pérdida MSE, ajuste
 - 2.1.2. Regresión logística: función sigmoide, frontera de decisión

- 2.1.3. Regularización L1 (Lasso) y L2 (Ridge)
- 2.2. Árboles de decisión
 - 2.2.1. Entropía e Impureza Gini
 - 2.2.2. Ganancia de información
 - 2.2.3. Sobreajuste, profundidad, poda (conceptual)
- 2.3. Clasificadores probabilísticos
 - 2.3.1. Naïve Bayes: independencia condicional
 - 2.3.2. Estimación por máxima verosimilitud
 - 2.3.3. Comparación entre modelos lineales y probabilísticos
- 2.4. Métricas fundamentales
 - 2.4.1. Matriz de confusión
 - 2.4.2. Precision, recall, F1-score
 - 2.4.3. ROC-AUC y curvas ROC

Duración

06 horas

Unidad III. Aprendizaje supervisado avanzado

Competencia

Comparar algoritmos supervisados avanzados mediante el uso de técnicas de validación y optimización de hiperparámetros en el análisis de su rendimiento para determinar el modelo más adecuado para la problemática a tratar acorde a criterios computacionales y estadísticos, apegado a estándares científicos, con pensamiento crítico y responsabilidad ética.

Contenido

- 3.1. Máquinas de Soporte Vectorial (SVM)
 - 3.1.1. Clasificación lineal y margen máximo
 - 3.1.2. Kernel trick: RBF, polinomial (conceptual + práctico)
- 3.2. Métodos basados en vecinos
 - 3.2.1. k-Nearest Neighbors
 - 3.2.2. Métricas: Euclidiana, Manhattan, Minkowski
 - 3.2.3. Efecto de la dimensionalidad en k-NN
- 3.3. Ensembles
 - 3.3.1. Bagging
 - 3.3.2. Random Forest
 - 3.3.3. Gradient Boosting (Conceptual, XGBoost/LightGBM referencia)
- 3.4. Ajuste y selección de modelos
 - 3.4.1. Validación cruzada k-fold
 - 3.4.2. Grid search y random search
 - 3.4.3. Curvas de aprendizaje y curvas de validación

Duración

06 horas

Unidad IV. Aprendizaje No Supervisado

Competencia

Aplicar algoritmos de aprendizaje no supervisado para identificar patrones, estructuras y relaciones subyacentes en conjuntos de datos, mediante el uso de técnicas de agrupamiento, reducción de dimensionalidad y evaluación de la calidad de los resultados con métricas apropiadas, para apoyar el análisis exploratorio y la toma de decisiones, con pensamiento crítico, rigor científico y responsabilidad en el manejo de los datos.

Contenido

- 4.1. Clustering
 - 4.1.1. K-means
 - 4.1.2. Clustering jerárquico
 - 4.1.3. DBSCAN y detección de densidad
- 4.2. Evaluación de clusters
 - 4.2.1. Inercia
 - 4.2.2. Silhouette score
 - 4.2.3. Comparación de resultados entre algoritmos
- 4.3. Reducción de dimensionalidad
 - 4.3.1. PCA: autovectores y varianza explicada
 - 4.3.2. Proyecciones 2D/3D y visualización
 - 4.3.3. Relación entre clustering y PCA en pipelines reales

Duración

06 horas

Unidad V. Introducción a las redes neuronales y aprendizaje profundo

Competencia

Aplicar los principios básicos de las redes neuronales artificiales, mediante el análisis de su estructura y funcionamiento, el entrenamiento de un perceptrón multicapa sencillo y la interpretación de su comportamiento, desempeño y limitaciones, para contextualizar su uso dentro del panorama general del aprendizaje profundo, con pensamiento analítico, rigor técnico y disposición al aprendizaje continuo.

Contenido

- 5.1. Neuronas artificiales y modelos lineales
 - 5.1.1. Neurona artificial: pesos, bias y función de activación
 - 5.1.2. Perceptrón simple: definición, regla de decisión y limitaciones (separabilidad lineal)

5.2 .Perceptrón multicapa (MLP)

- 5.2.1. Arquitecturas básicas: capas, neuronas, pesos y bias
- 5.2.2. Funciones de activación no lineales: ReLU, sigmoid, tanh
- 5.2.3. Retropropagación (backpropagation)
- 5.2.4. Sobreajuste en redes neuronales y su impacto en el aprendizaje

5.3. Proceso de entrenamiento y aplicación

- 5.3.1. Batch, epoch, learning rate
- 5.3.2. Inicialización de pesos y su efecto en el desempeño
- 5.3.3. Regularización básica: dropout
- 5.3.4. Entrenamiento de un MLP (scikit-learn o Keras)
- 5.3.5. Comparación MLP vs. modelos clásicos (SVM, RF)

Duración

05 horas

Unidad VI. Evaluación, interpretación y ética

Competencia

Evaluar e interpretar modelos y resultados de aprendizaje automático, mediante el uso de criterios estadísticos, técnicas de interpretación y marcos éticos, que incluya la identificación de sesgos, limitaciones e impactos potenciales en contextos técnicos y sociales, para comunicar conclusiones científicamente fundamentadas y apoyar decisiones socialmente responsables, con pensamiento crítico, honestidad académica y responsabilidad social.

Contenido

6.1. Evaluación avanzada

- 6.1.1. ROC-AUC en profundidad
- 6.1.2. Curvas de aprendizaje
- 6.1.3. Análisis de error

6.2. Interpretabilidad

- 6.2.1. Importancia de características
- 6.2.2. Errores de interpretación comunes
- 6.2.3. Primeros pasos hacia XAI (SHAP/LIME conceptual)

6.3. Ética y responsabilidad

- 6.3.1. Sesgos algorítmicos y de datos
- 6.3.2. Justicia y equidad
- 6.3.3. Privacidad y uso responsable de modelos

Duración

03 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I: Fundamentos matemáticos y conceptuales del aprendizaje automático

Práctica 1: Preparación y análisis exploratorio de datos (EDA) para aprendizaje automático

Duración

05 horas

Procedimiento

1. Atiende las indicaciones docentes sobre la importancia de preparar datos antes de entrenar modelos.
2. Selecciona un conjunto de datos sencillo (Iris, Titanic o el proporcionado por el docente).
3. Carga el conjunto de datos utilizando Python y librerías especializadas.
4. Examina las variables disponibles e identifica su tipo (numéricas, categóricas, continuas).
5. Identifica valores faltantes, ruido y outliers mediante funciones estadísticas básicas.
6. Genera visualizaciones (histogramas, boxplots, pairplots) para describir la estructura del dataset.
7. Aplica normalización o estandarización cuando corresponda e interpreta sus efectos.
8. Documenta hallazgos preliminares sobre la calidad del conjunto de datos.
9. Elabora un reporte con gráficas, tablas y conclusiones para revisión docente.

Recursos de apoyo

- Computadora
- Python (pandas, numpy, matplotlib, seaborn)
- Conjunto de datos proporcionado
- Editor de texto

Unidad II: Modelos supervisados básicos

Práctica 2: Implementación de modelos lineales supervisados

Duración

05 horas

Procedimiento

1. Atiende las indicaciones docentes sobre el uso de modelos lineales en clasificación y regresión.
2. Carga un dataset adecuado (Iris, Boston, o similar).
3. Divide el conjunto en entrenamiento y prueba usando partición estratificada cuando corresponda.
4. Implementa un modelo de regresión lineal y ajusta sus parámetros.

5. Implementa un modelo de regresión logística para un problema de clasificación.
6. Calcula métricas de desempeño básicas (MSE, accuracy, precision, recall).
7. Genera gráficas de predicción o fronteras de decisión.
8. Escribe conclusiones comparando ambos modelos.

Recursos de apoyo

- Python (scikit-learn)
- Conjunto de datos
- Computadora

Práctica 3: Árboles de decisión y Naïve Bayes en clasificación supervisada

Duración

05 horas

Procedimiento

1. Atiende las indicaciones docentes sobre el objetivo de comparar modelos no lineales con modelos probabilísticos.
2. Carga el dataset asignado por el docente.
3. Implementa un árbol de decisión y ajusta sus hiperparámetros básicos (profundidad, criterio).
4. Entrena un modelo *Naïve Bayes* para el mismo problema.
5. Evalúa ambos modelos con matriz de confusión, F1-score y ROC-AUC.
6. Compara ventajas y limitaciones de cada enfoque.
7. Documenta resultados en un breve informe técnico.

Recursos de apoyo

- Python (scikit-learn)
- Computadora
- Dataset asignado

Unidad III: Aprendizaje supervisado avanzado

Práctica 4: SVM y k-NN: comparación de algoritmos supervisados avanzados

Duración

06 horas

Procedimiento

1. Atiende las orientaciones docentes sobre la importancia de comparar algoritmos basados en margen y en distancia.
2. Selecciona un dataset con atributos continuos.
3. Entrena un modelo SVM lineal y registra su desempeño.
4. Entrena un modelo SVM con kernel RBF y analiza el efecto del parámetro gamma.
5. Implementa k-NN con diferentes valores de k.
6. Evalúa modelos utilizando validación cruzada k-fold.

7. Calcula métricas (accuracy, F1, ROC-AUC).
8. Elabora gráficas de fronteras de decisión o curvas de validación.
9. Documenta resultados comparativos destacando fortalezas y debilidades.

Recursos de apoyo

- Python
- Scikit-learn
- Notebook interactivo

Práctica 5: Métodos de ensamble y ajuste de hiperparámetros

Duración

05 horas

Procedimiento

1. Atiende las indicaciones docentes sobre la utilidad de los métodos de ensamble en tareas reales.
2. Carga un dataset de complejidad moderada (por ejemplo, Wine Quality, o, Breast Cancer).
3. Entrena un modelo Random Forest ajustando el número de árboles y la profundidad.
4. Entrena un modelo Gradient Boosting utilizando valores por defecto.
5. Ejecuta Grid Search o Random Search sobre al menos dos hiperparámetros.
6. Evalúa los modelos con validación cruzada.
7. Genera gráficas de importancia de variables.
8. Compara resultados y argumenta qué modelo es más adecuado y por qué.

Recursos de apoyo

- Python
- Scikit-learn
- Dataset asignado

Unidad IV: Aprendizaje no supervisado

Práctica 6: Clustering y reducción de dimensionalidad con PCA

Duración

05 horas

Procedimiento

1. Atiende las indicaciones docentes sobre el uso de clustering para explorar estructuras internas.
2. Carga un dataset mixto o continuo.
3. Aplica K-means y registra los centroides obtenidos.
4. Implementa DBSCAN y analiza su sensibilidad a eps y min_samples.
5. Aplica PCA y genera proyecciones en 2D o 3D.

6. Evalúa los resultados con Silhouette Score y/o inercia.
7. Identifica el algoritmo más adecuado según la distribución de los datos.
8. Documenta resultados en un informe con visualizaciones.

Recursos de apoyo

- Python (scikit-learn)
- Biblioteca de visualización (matplotlib o seaborn)

Unidad V: Introducción a redes neuronales y aprendizaje profundo

Práctica 7: Entrenamiento y evaluación de un perceptrón multicapa (MLP)

Duración

05 horas

Procedimiento

1. Atiende las orientaciones docentes sobre la arquitectura básica de redes neuronales.
2. Carga un dataset simple (Iris, MNIST reducido o equivalente).
3. Implementa un MLP con scikit-learn o Keras.
4. Ajusta hiperparámetros (capas ocultas, activación, learning rate).
5. Evalúa el modelo con accuracy, F1-score y curvas de aprendizaje.
6. Compara el MLP con SVM o Random Forest entrenados previamente.
7. Elabora conclusiones sobre capacidades y limitaciones del MLP.

Recursos de apoyo

- Computadora
- Python (scikit-learn o Keras)
- Dataset proporcionado

Unidad VI: Evaluación, interpretación y ética

Práctica 8: Interpretabilidad, análisis de error y sesgos en modelos de ML

Duración

04 horas

Procedimiento

1. Atiende las orientaciones docentes sobre la importancia de la evaluación responsable en ML.
2. Selecciona uno de los modelos entrenados previamente.
3. Genera la matriz de confusión y calcula métricas detalladas de error.
4. Analiza importancia de características y discute posibles errores de interpretación.
5. Identifica atributos o patrones que pudieran generar sesgos en el modelo.
6. Elabora una reflexión corta sobre impactos éticos y consideraciones sociales.

Recursos de apoyo

- Python
- Visualizaciones
- Documentos de referencia sobre ética

Práctica 9: Proyecto integrador de aprendizaje automático

Duración

08 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el alcance, entregables y criterios de evaluación del proyecto.
2. Selecciona un problema realista modelable mediante aprendizaje automático.
3. Formula el problema, define objetivos y justifica la elección del dataset.
4. Prepara el conjunto de datos aplicando técnicas de limpieza, transformación y análisis exploratorio.
5. Selecciona modelos supervisados o no supervisados de acuerdo con el problema.
6. Entrena los modelos y registra sus parámetros y configuraciones.
7. Evalúa los modelos mediante validación cruzada y métricas adecuadas.
8. Compara críticamente el desempeño de los modelos.
9. Analiza limitaciones técnicas y éticas de la solución.
10. Elabora un reporte técnico con: problema, metodología, resultados, discusión y conclusiones.
11. Diseña una presentación digital (PowerPoint, Canva o Genially).
12. Expone el proyecto ante el grupo.
13. Entrega archivo final para retroalimentación docente.

Recursos de apoyo

- Computadora
- Python (scikit-learn, pandas)
- Editor de texto y herramientas de presentación
- Bases de datos seleccionadas

IX. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos

- Técnica expositiva
- Discusión guiada
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Cuadros comparativos
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	30%
– Prácticas laboratorio	35%
– Proyecto Integrador	35%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Chollet, F. (2021). *Deep learning with Python*. Manning.
- Ekman, M. (2021). *Learning deep learning: Theory and practice of neural networks, computer vision, natural language processing, and transformers using TensorFlow*. Addison-Wesley Professional.
- Geron, A. (2022). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (3rd ed.)*. O'Reilly Media.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press. <https://www.deeplearningbook.org>. [Clásica]
- Han, J., Pei, J., Tong, H. (2022). *Data Mining: Concepts and Techniques (4th ed.)*. Morgan Kaufmann.
- Moroney, L. (2020). *AI and Machine Learning for Coders: A Programmer's Guide to Artificial Intelligence*. O'Reilly Media.

Neapolitan, R. E., & Jiang, X. (2018). *Artificial intelligence: With an introduction to machine learning*. CRC press. [Clásica]

Witten, I. H., Frank, E., Hall, M. A., Pal, C. J. (2016). *Data Mining: Practical Machine Learning Tools and Techniques* (4th ed.). Morgan Kaufmann. [clásica]

Recursos de acceso libre o complementarios externos

ACM SIGAI. (s.f.). ACM SIGAI – Special Interest Group on Artificial Intelligence. <https://sigai.acm.org/main/>

ACM Special Interest Group on Knowledge Discovery in Data. <https://www.acm.org/special-interest-groups/sigs/sigkdd>

Aggarwal, C. C. (2015). *Data Mining: The Textbook*. Springer. [Clásica]

Aggarwal, C. C., & Reddy, C. K. (2014). *Data clustering. Algorithms and applications*. Chapman&Hall/CRC Data mining and Knowledge Discovery series, Londra. [Clásica]

Bishop, C. M., & Nasrabadi, N. M. (2006). *Pattern recognition and machine learning* (Vol. 4, No. 4, p. 738). Springer. [Clásica]

European Union (2023). *EU Artificial Intelligence Act (AI Act)*. <https://artificialintelligenceact.eu/>

Murphy, K. P. (2012). *Machine learning: A probabilistic perspective*. MIT Press. <https://www.cs.ubc.ca/~murphyk/>

Poole, D., y Mackworth, A. (2023). *Artificial Intelligence: Foundations of Computational Agents* (3a ed.). Cambridge University Press. <https://artint.info/3e/html/ArtInt3e.html>

Shalev-Shwartz, S., y Ben-David, S. (2014). *Understanding machine learning: From theory to algorithms*. Cambridge University Press. <https://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/> [Clásica]

Suh, N., & Cheng, G. (2025). A survey on statistical theory of deep learning: Approximation, training dynamics, and generative models. *Annual Review of Statistics and Its Application*, 12, 177–207. <https://doi.org/10.1146/annurev-statistics-040522-013920>

Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2023). *Dive into deep learning*. Cambridge University Press. <https://d2l.ai/index.html>

XII. PERFIL DOCENTE

Licenciatura en Ciencias Computacionales o áreas afines, preferentemente con estudios de posgrado en Aprendizaje Automático, Ciencia de Datos o Inteligencia Artificial. Experiencia en docencia universitaria y dominio de los fundamentos matemáticos, estadísticos y computacionales del aprendizaje automático. Conocimientos en programación, especialmente en Python y librerías para análisis y modelado de datos. Capacidad para integrar de manera ética y responsable el uso de tecnologías emergentes, explicar conceptos con claridad, y diseñar actividades teóricas y prácticas que promuevan el trabajo colaborativo, el pensamiento crítico y la actualización continua.

LEARNING MODULE

I. GENERAL INFORMATION

School:	Faculty of Sciences, Ensenada
Major:	Bachelor's degree in Physics Bachelor of Science in Computer Science
Study Program:	2027-1
Learning Module Name:	Academic English for Science
Number:	
Duration distribution:	Workshop Hours: 03 Laboratory Hours: 00 Field Practice Hours: 00 Clinical Hours:00 Credits: 03
Stage:	Terminal
Module Type:	Compulsory
Requirements:	None
Mode of delivery:	In-person
Design Team:	Priscilla Elizabeth Iglesias Vázquez Jorge Alberto Villavicencio Aguilar Ramón Carrillo Bastos Manuel Iván Ocegueda Miramontes
Approved of Assistant Dean (s):	Priscilla Elizabeth Iglesias Vázquez
Date:	May 21, 2026
Approval and registration:	

II. PURPOSE OF LEARNING MODULE

To enable students to use the English language in an academic context related to science. Through reading, writing, and oral presentations on scientific topics, students will strengthen their ability to interpret research papers and communicate scientific results in English. The learning module promotes clarity, analytical reasoning, curiosity, inclusion, and collaborative learning, linking linguistic competence with disciplinary understanding. It is located in the terminal stage, is compulsory, and belongs to the Physics area of knowledge.

III. COMPETENCE OF THE LEARNING MODULE

To use English to read, write, and present scientific research with accurate terminology, concise structure, and ethical citation, demonstrating audience-appropriate reasoning and teamwork.

IV. EVIDENCES OF LEARNING/ACHIEVEMENT

As evidence of learning, students will submit work in three domains: reading, writing, and scientific presentations, including participation in a simulated scientific conference.

- Reading: a 1–2 page brief on a peer-reviewed scientific article (research objectives, methods, results, limits) and a vocabulary long of 20–25 terms with context and paraphrase.
- Writing: a structured abstract (≤ 250 words) and a short report in IMRaD format (750–1000 words, ≥ 5 scholarly sources, figures with captions), plus a similarity report ($\leq 10\%$) and a properly formatted reference list.
- Oral: a 8–10 slide deck and a 10-minute conference-style talk with a 3–5 minute Q&A.
- Conference: active participation and professional conduct during the session, two peer-review forms (one abstract and one talk), and engagement in advanced Q&A. Assessment uses concise analytic rubrics, organization, scientific accuracy, language/style, audience design, and integrity citation.

V. PROFESSIONAL COMPETENCY OF THE GRADUATE PROFILE TO WHICH THE LEARNING UNIT CONTRIBUTES

Bachelor's degree in Physics

To propose models of physical systems based on the understanding of natural phenomena to solve physics problems with scientific rigor, responsibility and honesty.

Bachelor of Science in Computer Science

Design and evaluate intelligent computing solutions by applying principles of artificial intelligence, data mining, and integration of computational tools, with the purpose of supporting decision-making in complex situations typical of scientific, social, industrial or technological contexts, acting with ethical responsibility, critical thinking, and interdisciplinary collaboration.

VI. CROSS-CUTTING THEME TO WHICH IT CONTRIBUTES

Excellence, Vanguard, Inclusion, and Interculturality.

VII. CONTENT

1. Reading and understanding scientific texts in English.
 - 1.1. Reading strategies for scientific papers and reports.
 - 1.2. Scientific vocabulary.
 - 1.3. Analyzing abstract and scientific summaries.
 - 1.4. Identifying biases and stereotypes.
 - 1.5. Paraphrasing and summarizing results.
2. Writing scientific reports and abstracts in English.
 - 2.1. Structure of scientific reports following the IMRaD format.
 - 2.2. Writing abstracts and conclusions.
 - 2.3. Grammar style in scientific writing.
 - 2.4. Avoiding plagiarism and ensuring academic integrity and equity.
3. Oral communications and presentation in English on scientific topics.
 - 3.1. Pronunciation and scientific terminology.
 - 3.2. Designing effective slides and visual aids.
 - 3.3. Presenting results and defending ideas.
 - 3.4. Simulated conference participation.

VIII. STRUCTURE OF WORKSHOP PRACTICES

Practice 1: Critical reading of research papers in English

Duration

08 hours

Procedure

1. Follows the teacher's guidelines for the development of the practice.
2. Selects an article in English from an indexed scientific journal.
3. Identifies abstract methods, results, and conclusions.
4. Identifies biases and stereotypes.
5. Highlights and defines technical vocabulary.
6. Writes a text summarizing the main findings.
7. Receives teacher feedback.
8. Submits the final version with observations addressed in the feedback.

Support resources

- Computer
- Internet
- Scientific databases
- Dictionary

Practice 2: Development of scientific and technical vocabulary in English

Duration

08 hours

Procedure

1. Follows the teacher's guidelines for the development of the practice.
2. Creates a glossary of scientific and technical terms in English.
3. Writes example sentences using each term.
4. Discuss meanings and contexts in small groups.
5. Prepares and submit an infographic glossary with contextual explanations.
6. Receives teacher feedback.
7. Submits the final version with observations addressed in the feedback.

Support resources

- Computer
- Internet
- Designing platform
- Scientific databases
- Dictionary

Practice 3: Writing abstracts in English and peer review

Duration

08 hours

Procedure

1. Follows the teacher's guidelines for the development of the practice.
2. Writes an abstract in English on a selected scientific topic.
3. Exchanges abstracts with classmates for peer correction following the teacher's rubric.
4. Discuss feedback in pairs.
5. Receives teacher feedback.
6. Submits the final version with observations addressed in the feedback.

Support resources

- Computer
- Internet
- Scientific databases
- Dictionary
- Rubric to evaluate abstracts.

Practice 4: Writing a scientific report in English

Duration

08 hours

Procedure

1. Follows the teacher's guidelines for the development of the practice.
2. Writes a scientific report in English on a selected topic, adhering to the IMRaD structure.
3. Includes graphs and data interpretation.
4. Avoids plagiarism and ensures academic integrity and equity.
5. Receives teacher feedback.
6. Submits the final version with observations addressed in the feedback.

Support resources

- Computer
- Internet
- Scientific databases
- Dictionary

Practice 5: Scientific presentation in English**Duration**

08 hours

Procedure

1. Follows the teacher's guidelines for the development of the practice.
2. Prepares a presentation in English on a selected scientific topic.
3. Uses slides with graphs, figures and keywords.
4. Delivers the presentation and answer questions.
5. Receives teacher feedback.

Support resources

- Computer
- Software or internet platform to create presentations
- Projector
- Internet
- Scientific databases
- Dictionary

Practice 6: Scientific conference simulation in English**Duration**

08 hours

Procedure

1. Follows the teacher's guidelines for the development of the practice.
2. Organizes a simulated conference where students present in English.
3. Assigns roles: speaker, moderator and audience.
4. Conducts sessions with Q&A.
5. Reflects on communication strategies and improvement areas.

6. Receives teacher feedback.

Support resources

- Computer
- Projector
- Internet
- Scientific databases
- Dictionary

IX. METHODOLOGY AND STRATEGIES

Course framework

On the first day of class, each instructor must establish the working methods, evaluation criteria, quality of academic work, and teacher-student rights and responsibilities.

Teaching strategies

- Project-based method
- Question-based learning
- Expository technique
- Debates
- Guided discussion
- Guided instruction
- Forums

Learning strategies

- Documentary research
- Presentation of case resolution
- Collaborative work
- Expository technique
- Comparative charts

X. EVALUATION CRITERIA

Accreditation criteria

To be eligible for ordinary and extraordinary exams, each student must meet the attendance percentages established by the current School Statute. Grades are on a scale from 0 to 100, with a minimum passing grade of 60.

Assessment criteria

– Reading practice	30 %

– Writing practice	30 %
– Oral presentations	20 %
– Scientific conference	20 %
Total	100 %

XI. BIBLIOGRAPHY

Resources of the UABC physical and digital library

Barrett, G. (2016). *Perfect English grammar: the indispensable guide to excellent writing and speaking*. Zephyros Press. [classic]

Blackwell, J. & Martin, J. (2011). *A scientific approach to scientific writing* [Electronic resource]. Springer. [classic]

Berry, R. (2012). *English grammar: a resource book for students* [Electronic resource]. Routledge. [classic]

DiGiacomo, M. (2020). *The English grammar workbook for adults: a self-study guide to improve functional writing*. Rockridge Press.

Glasman-Deal, H. (2020). *Science research writing: for native and non-native speakers of English*. World Scientific.

Silyn-Roberts, H. (2013). *Writing for Science and Engineering: papers, presentations and reports* (2nd ed.) [Electronic resource]. Elsevier. [classic]

Wallwork, A. (2011). *English for writing research papers* [Electronic resource]. Springer. [classic]

Databases available at the UABC Library. <https://bibliotecas.uabc.mx/>

Freely accessible or external complementary resources

Centers for Disease Control and Prevention (CDC). (2019). *Scientific writing: Participant workbook*. Association of Public Health Laboratories (APHL). https://www.aphl.org/programs/infectious_disease/Documents/Scientific_Writing_Participant_Workbook.pdf

Cornell University Library. (s. f.). *arXiv: Open access to physics, mathematics, and computer science research*. <https://arxiv.org/>

European Association of Science Editors. (2021). *Writing science in English: A guide for authors and editors*. ScienceScape Editing. <https://ease.org.uk/wp-content/uploads/2021/10/Writing-Science-in-English-2021.pdf>

Gopen, G. D., & Swan, J. A. (1990). *The science of scientific writing*. American Scientist, 78(6), 550–558.

https://www.usenix.org/sites/default/files/gopen_and_swan_science_of_scientific_writing.pdf

Hewings, M. (2023). *Advanced grammar in use* (4th ed.). Cambridge University Press.

<https://dn721806.ca.archive.org/0/items/english-grammar-in-use-raymond-murphy/3-Advanced%20Grammar%20in%20Use%203rd%20Edition%20Martin%20Hewings.pdf>

Iowa State University Library. (s. f.). *Free scientific research resources*.

<https://instr.iastate.libguides.com/c.php?g=49318&p=318011>

Open access scientific databases.

XII. TEACHER PROFILE

A bachelor's degree in Physics or a related field, preferably with postgraduate studies and extensive experience in scientific research and English language proficiency, minimum B2 (upper intermediate level). Experience in teaching and pedagogical skills. A respectful, inclusive, honest, responsible, proactive, and analytical individual who encourages both individual and teamwork.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Administración de Proyectos
Clave:	
Distribución horaria:	Horas Clase: 01 Horas Taller: 00 Horas Laboratorio: 04 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 01 Créditos:06
Etapas de formación:	Terminal
Carácter:	Obligatoria
Requisito:	Ninguno.
Modalidad de impartición:	Presencial
Equipo de diseño:	María Victoria Meza Kubo Roberto Adolfo Romero Hernández
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Proporcionar al estudiantado los fundamentos, procesos y herramientas esenciales de la administración de proyectos de software, aplicando estos conocimientos para el análisis, planeación, ejecución, monitoreo y cierre de un proyecto de pequeña escala. Se busca que desarrolle la capacidad de organizar recursos, estimar esfuerzos, gestionar riesgos y comunicarse eficazmente con su equipo y con el cliente, promoviendo una actitud profesional, ética y colaborativa.

La unidad de aprendizaje es de carácter obligatorio dentro de la etapa disciplinaria. Su área de conocimiento es Programación e Ingeniería de Software y se recomienda haber cursado y aprobado la unidad de aprendizaje Ingeniería de Software.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Gestionar proyectos de desarrollo de software mediante el uso de metodologías y buenas prácticas de administración de proyectos, para definir, planear, ejecutar, monitorear y finalizar un proyecto con calidad, en tiempo y forma, comunicándose efectivamente con el equipo y el cliente, con responsabilidad, ética y trabajo colaborativo.

IV. EVIDENCIA DE APRENDIZAJE

Proyecto integrador de administración de proyectos, que incluya una propuesta inicial, planeación, bitácora y entregables finales.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Inclusión; Excelencia.

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos de la administración de proyectos

Competencia

Examinar las fases y enfoques contemporáneos de la administración de proyectos, mediante el análisis de experiencias propias y externas de problemáticas en el proceso de desarrollo de proyectos, para interpretar la importancia y beneficios del uso de estrategias en la gestión de proyectos, con actitud analítica y crítica

Contenido

- 1.1. Definición, antecedentes y objetivos de administración de proyectos.
- 1.2. Fases de la administración de proyectos.
- 1.3. Definición de proyectos.
- 1.4. Problemas más frecuentes en la administración de proyectos.
- 1.5. Importancia y beneficios de administración de proyectos.

Duración

02 horas

Unidad II. Inicio y definición del proyecto

Competencia

Elaborar una propuesta de proyecto, mediante la identificación de un problema real con el uso de técnicas de levantamiento de requerimientos y estimación de tiempos y costos, para definir y validar las necesidades de automatización de una organización, con una actitud honesta, analítica y propositiva.

Contenido

- 2.1. Identificación de problemas y oportunidades.
- 2.2. Levantamiento inicial de requerimientos.
- 2.3. Acta de constitución del proyecto.
- 2.4. Determinación de alcance y criterios de aceptación.
- 2.5. Identificación inicial de interesados.

Duración

03 horas

Unidad III. Planeación del proyecto

Competencia

Diseñar un plan de trabajo del proyecto, mediante herramientas de estimación, recursos, calendarización y análisis de riesgos, para generar un plan coherente, viable y alineado con los requerimientos del cliente, con una actitud crítica.

Contenido

- 3.1. Estructura de desglose del trabajo (EDT/WBS).
- 3.2. Estimación de tareas, esfuerzos y costos.
- 3.3. Roles del equipo y asignación de actividades.
- 3.4. Herramientas de calendarización (Gantt, ruta crítica).
- 3.5. Matriz de riesgos.
- 3.6. Plan de comunicación.
- 3.7. Construcción del plan de proyecto.

Duración

04 horas

Unidad IV. Organización y ejecución del proyecto

Competencia

Ejecutar la administración del proyecto, mediante el uso de herramientas de comunicación y control, para asegurar el avance ordenado y colaborativo del proyecto con actitud colaborativa y equitativa.

Contenido

- 4.1. Gestión del equipo y roles (liderazgo, motivación, acuerdos).
- 4.2. Herramientas colaborativas: Trello, Jira, GitHub Projects.
- 4.3. Reuniones efectivas (con cliente y equipo).
- 4.4. Control y seguimiento del avance.
- 4.5. Gestión de incidencias y control de cambios.

Duración

03 horas

Unidad V. Evaluación y pruebas del proyecto

Competencia

Evaluar el producto de software generado, mediante la elaboración de un plan de pruebas que incluya revisiones en el ambiente de desarrollo y el entorno real de operación de una manera controlada, para identificar elementos críticos que impidan el buen funcionamiento del sistema construido, con una actitud honesta, crítica y responsable

Contenido

- 5.1. Estrategias de pruebas de software.
- 5.2. Diseño de casos de prueba (funcional, integración, aceptación).
- 5.3. Pruebas en entornos controlados.
- 5.4. Registro, categorización y seguimiento de errores.
- 5.5. Evidencia de pruebas y métricas de calidad.

Duración

02 horas

Unidad VI. Cierre y liberación del proyecto***Competencia***

Liberar el producto terminado al cliente real, mediante la instalación, capacitación y presentación del sistema de software al grupo de usuarios finales, para poner en operación y validar el correcto funcionamiento del producto generado, con una actitud honesta y responsable.

Contenido

- 6.1. Actividades de cierre de proyecto.
- 6.2. Entrega e instalación del producto.
- 6.3. Manual de usuario técnico/funcional.
- 6.4. Capacitación a usuarios finales.
- 6.5. Carta de liberación y validación.
- 6.6. Lecciones aprendidas y evaluación final.

Duración

02 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Fundamentos de la administración de proyectos

Práctica 1. Análisis de proyectos exitosos y fallidos

Duración

04 horas

Procedimiento

1. Atiende la explicación sobre factores de éxito y fracaso en proyectos.
2. Analiza dos casos de estudio proporcionados por el docente.
3. Identifica problemas relacionados con alcance, tiempo, costo y comunicación.
4. Registra hallazgos en una tabla comparativa.
5. Elabora conclusiones sobre buenas prácticas.
6. Entrega el reporte en formato digital.
7. Recibe retroalimentación de su docente.

Recursos de apoyo

- Computadora
- Internet
- Documento de casos de estudio

Práctica 2. Definición del proyecto de software

Duración

04 horas

Procedimiento

1. Atiende la explicación sobre definición de proyectos.
2. Propone una problemática real a resolver.
3. Analiza la viabilidad del proyecto (técnica, económica y temporal).
4. Redacta la descripción del proyecto.
5. Ajusta el documento con base en observaciones.
6. Entrega el documento en la plataforma.
7. Recibe retroalimentación de su docente.

Recursos de apoyo

- Computadora
- Internet
- Procesador de texto

Unidad II. Inicio y definición del proyecto

Práctica 3. Identificación de interesados (stakeholders)

Duración

04 horas

Procedimiento

1. Atiende la explicación sobre stakeholders.
2. Identifica los actores involucrados en su proyecto.
3. Clasifica los interesados según nivel de influencia e interés.
4. Construye una matriz de stakeholders.
5. Entrega el documento en formato digital.
6. Recibe retroalimentación de su docente.

Recursos de apoyo

- Computadora
- Internet
- Hoja de cálculo o procesador de texto

Práctica 4: Levantamiento de requerimientos

Duración

04 horas

Procedimiento

1. Atiende la explicación sobre técnicas de levantamiento de requerimientos.
2. Simula una entrevista con el cliente.
3. Redacta requerimientos funcionales y no funcionales.
4. Organiza la información en un documento estructurado.
5. Entrega el documento en la plataforma.
6. Recibe retroalimentación de su docente.

Recursos de apoyo

- Computadora
- Internet
- Procesador de texto
- Guía de entrevista

Práctica 5: Elaboración del acta de constitución del proyecto

Duración

04 horas

Procedimiento

1. Atiende la explicación sobre el acta de constitución.
2. Define objetivos, alcance preliminar y restricciones.
3. Identifica entregables principales.
4. Redacta el acta de constitución del proyecto.
5. Exporta el documento a PDF.
6. Entrega en la plataforma.
7. Recibe retroalimentación de su docente.

Recursos de apoyo

- Computadora
- Internet
- Procesador de texto
- Plantilla de Project Charter

Unidad III. Planeación del proyecto

Práctica 6: Construcción de la EDT (WBS)

Duración

04 horas

Procedimiento

1. Atiende la explicación sobre la EDT.
2. Identifica entregables del proyecto.
3. Descompone el proyecto en tareas jerárquicas.
4. Utiliza una herramienta de diagramación.
5. Genera la estructura EDT.
6. Exporta el diagrama a PDF.
7. Entrega en la plataforma.
8. Recibe retroalimentación de su docente.

Recursos de apoyo

- Computadora
- Internet
- Herramienta de diagramación

Práctica 7: Estimación de tiempos y costos

Duración

04 horas

Procedimiento

1. Atiende la explicación sobre técnicas de estimación.
2. Asigna duración a cada tarea de la EDT.
3. Calcula costos aproximados del proyecto.
4. Registra la información en una tabla.
5. Entrega el documento en formato digital.
6. Recibe retroalimentación de su docente.

Recursos de apoyo

- Computadora
- Internet
- Hoja de cálculo

Práctica 8: Asignación de roles y responsabilidades

Duración

04 horas

Procedimiento

1. Atiende la explicación sobre roles en proyectos.
2. Define los integrantes del equipo.
3. Asigna responsabilidades por tarea.
4. Construye una matriz RACI.
5. Entrega el documento en la plataforma.
6. Recibe retroalimentación de su docente.

Recursos de apoyo

- Computadora
- Internet
- Hoja de cálculo

Práctica 9: Elaboración del cronograma (Gantt)

Duración

04 horas

Procedimiento

1. Atiende la explicación sobre diagramas de Gantt.
2. Registra tareas y duraciones.
3. Establece dependencias entre actividades.

4. Utiliza software de gestión de proyectos.
5. Genera el cronograma.
6. Exporta el archivo a PDF.
7. Entrega en la plataforma.
8. Recibe retroalimentación de su docente.

Recursos de apoyo

- Computadora
- Internet
- Software de gestión (Project, Excel, etc.)

Práctica 10: Análisis de riesgos y plan de comunicación

Duración

04 horas

Procedimiento

1. Atiende la explicación sobre gestión de riesgos.
2. Identifica riesgos del proyecto.
3. Evalúa probabilidad e impacto.
4. Construye una matriz de riesgos.
5. Define el plan de comunicación del proyecto.
6. Entrega ambos documentos en la plataforma.
7. Recibe retroalimentación de su docente.

Recursos de apoyo

- Computadora
- Internet
- Plantillas de gestión de riesgos

Unidad IV. Organización y ejecución del proyecto

Práctica 11: Configuración de herramientas colaborativas

Duración

04 horas

Procedimiento

1. Atiende la explicación sobre herramientas colaborativas.
2. Crea un tablero en Trello, Jira o GitHub Projects.
3. Define columnas y tareas del flujo de trabajo.
4. Asigna responsables a cada actividad.
5. Comparte el tablero con el equipo.
6. Entrega evidencia del tablero.
7. Recibe retroalimentación de su docente.

Recursos de apoyo

- Computadora
- Internet
- Herramienta colaborativa

Práctica 12: Simulación de reuniones de proyecto

Duración

04 horas

Procedimiento

1. Atiende la explicación sobre reuniones efectivas.
2. Participa en una reunión simulada con el equipo.
3. Registra acuerdos y compromisos.
4. Elabora minuta de reunión.
5. Entrega el documento en la plataforma.
6. Recibe retroalimentación de su docente.

Recursos de apoyo

- Computadora
- Internet
- Formato de minuta

Práctica 13: Seguimiento y control del proyecto

Duración

04 horas

Procedimiento

1. Atiende la explicación sobre control de avance.
2. Analiza el estado actual del proyecto.
3. Compara avance real contra planificado.
4. Identifica desviaciones.
5. Elabora reporte de seguimiento.
6. Entrega el documento en la plataforma.
7. Recibe retroalimentación de su docente.

Recursos de apoyo

- Computadora
- Internet
- Herramienta de gestión

Unidad V. Evaluación y pruebas del proyecto

Práctica 14: Diseño de casos de prueba

Duración

04 horas

Procedimiento

1. Atiende la explicación sobre pruebas de software.
2. Define escenarios de prueba.
3. Diseña casos de prueba funcionales y de integración.
4. Organiza los casos en un formato estructurado.
5. Entrega el documento en la plataforma.
6. Recibe retroalimentación de su docente.

Recursos de apoyo

- Computadora
- Internet
- Plantilla de casos de prueba

Práctica 15: Ejecución de pruebas y registro de errores***Duración***

04 horas

Procedimiento

1. Atiende la explicación sobre gestión de errores.
2. Ejecuta los casos de prueba definidos.
3. Identifica fallos en el sistema.
4. Registra errores en una bitácora.
5. Clasifica los errores por tipo y severidad.
6. Entrega la bitácora en la plataforma.
7. Recibe retroalimentación de su docente.

Recursos de apoyo

- Computadora
- Internet
- Sistema desarrollado
- Plantilla de bitácora

Unidad VI. Cierre y liberación del proyecto**Práctica 16: Cierre y entrega del proyecto*****Duración***

04 horas

Procedimiento

1. Atiende la explicación sobre cierre de proyectos.

2. Integra los entregables finales.
3. Elabora manual de usuario.
4. Prepara la presentación del proyecto.
5. Documenta lecciones aprendidas.
6. Entrega todos los productos en la plataforma.
7. Recibe retroalimentación de su docente.

Recursos de apoyo

- Computadora
- Internet
- Procesador de texto
- Software de presentación

MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Estudio de caso
- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Técnica expositiva

Estrategia de aprendizaje

- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Visitas a campo

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

- Evaluaciones parciales	30%
- Prácticas laboratorio	30%
- Proyecto integrador	40%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Canosa Ferreiro, A. J. (2024). *Scrum: Teoría e implementación práctica*. RA-MA Editorial y Publicaciones. <https://libcon.rec.uabc.mx:2186/read/9788410181380/index>

Hernández, Z. T., Martínez, H. T., Pantoja, J. V. R., Flores, J. A. R., Perego, N. R., Olvera, E. M., ... & Espinosa, E. O. C. (2014). *Administración de proyectos*. Grupo Editorial Patria. [clásica]

Kerzner, H. (2022). *Project management: A systems approach to planning, scheduling, and controlling* (13th ed.). Wiley. <https://libcon.rec.uabc.mx:2061/c/6wx7bd/search/details/lrwc47oiz?db=nlebk#:~:text=Acess%20options-,Download,-Details>

Nicholas, J. M., & Steyn, H. (2008). *Project management for business, engineering, and technology: Principles and practice*. Elsevier Butterworth-Heinemann. [clásica]

Project Management Institute. (2021). *A guide to the project management body of knowledge (PMBOK® guide)* (7th ed.). Project Management Institute.

Recursos de acceso libre o complementarios externos

Asana.ink.(2026). Colaboración entre humanos e IA que garantiza resultados. <https://asana.com/>

Nicholas, J. M., & Steyn, H. (2017). *Project management for engineering, business and technology*. Routledge. [clásica] [https://ftp.idu.ac.id/wp-content/uploads/ebook/ip/BUKU%20MANAJEMEN%20PROYEK/Project%20Management%20for%20Business,%20Engineering,%20and%20Technology,%20Third%20Edition%20\(%20PDFDrive%20\).pdf](https://ftp.idu.ac.id/wp-content/uploads/ebook/ip/BUKU%20MANAJEMEN%20PROYEK/Project%20Management%20for%20Business,%20Engineering,%20and%20Technology,%20Third%20Edition%20(%20PDFDrive%20).pdf)

University of Minnesota Libraries. (s.f.). *Project management: Navigating the complexity with a systematic approach*. Open Textbook Library. <https://open.umn.edu/opentextbooks>

XII. PERFIL DOCENTE

Licenciatura en Ciencias Computacionales, Ingeniería de software o áreas afines, preferentemente con posgrado en área afín. Experiencia docente de 2 años. Ampla experiencia en planeación, diseño e implementación de proyectos reales de desarrollo de software. Ser una persona responsable y proactiva que proporcione al estudiantado las herramientas que le permitan desarrollar las aptitudes y competencias descritas para esta unidad de aprendizaje.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Ciberseguridad
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 03 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Terminal
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Gerardo Tovar Ramos
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Proporcionar al estudiantado los conocimientos, habilidades y actitudes necesarias para aplicar principios y modelos de ciberseguridad en la protección de sistemas computacionales y su flujo de datos en sus tres estados. Se permitirá el desarrollo competencias prácticas en la gestión de riesgos, análisis de vulnerabilidades, fortalecimiento de sistemas, protección de la privacidad, monitoreo y auditoría de incidentes, así como en la trazabilidad y aseguramiento del ciclo de vida de la información, fomentando el pensamiento crítico, la ética profesional y el trabajo colaborativo.

Esta unidad de aprendizaje pertenece al área de conocimiento de redes y corresponde a la etapa terminal, tiene carácter obligatorio y se recomienda haber cursado Redes de Datos y GNU/Linux para las Ciencias.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Evaluar los principios, modelos y marcos normativos de ciberseguridad mediante la implementación de medidas de protección a lo largo del ciclo de vida del dato con herramientas técnicas y metodologías internacionales, para proteger los sistemas, servicios e información con ética profesional, responsabilidad social y trabajo colaborativo.

IV. EVIDENCIA DE APRENDIZAJE

Desarrollar un proyecto final (de forma individual o en equipo de máximo 3 integrantes) en el que se apliquen principios, modelos y marcos normativos de ciberseguridad para analizar un sistema, servicio o flujo de datos real o simulado. El proyecto deberá considerar el ciclo de vida de la información en sus tres estados y proponer un plan de medidas de protección que garantice la confidencialidad, integridad, disponibilidad, privacidad, autenticidad, trazabilidad y no repudio. El informe técnico deberá justificar las decisiones con base en datos obtenidos del análisis, incluir una propuesta de mejoras aplicables en el rediseño, implementación o reprogramación del entorno evaluado, y ser defendido de forma oral ante el grupo.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Compromiso con la responsabilidad social; Excelencia.

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos de la ciberseguridad

Competencia

Aplicar principios, modelos y marcos normativos fundamentales de la ciberseguridad, así como la legislación básica aplicable al ámbito tecnológico, mediante el análisis crítico de referentes técnicos y estándares internacionales, con el propósito de valorar la importancia de la protección integral de los sistemas y de la información a lo largo de su ciclo de vida, con honestidad y responsabilidad profesional.

Contenido

- 1.1. Principios y atributos de la ciberseguridad
 - 1.1.1. Tríada CIA: confidencialidad, integridad y disponibilidad
 - 1.1.2. Atributos complementarios: autenticidad, trazabilidad, no repudio, privacidad
 - 1.1.3. Principio de menor privilegio y seguridad por defecto
 - 1.1.4. Resiliencia y continuidad de negocio (enfoque técnico)
- 1.2. Modelos conceptuales
 - 1.2.1. Cubo de McCumber
 - 1.2.2. Zero Trust Architecture (nociones básicas)
 - 1.2.3. Defensa en profundidad
 - 1.2.4. Privacy by Design
- 1.3. Referentes técnicos y estándares
 - 1.3.1. NIST Cybersecurity Framework
 - 1.3.2. CIS Controls
 - 1.3.3. ISO/IEC 27001
 - 1.3.4. OWASP Top 10
- 1.4. Legislación aplicable al ciberespacio
 - 1.4.1. Nacional
 - 1.4.2. Internacional

Duración

6 horas

Unidad II. Amenazas, vulnerabilidades y riesgos asociados al ciberespacio

Competencia

Interpretar el nivel del riesgo asociado al ciberespacio en sistemas, servicios y flujos de datos, por medio de metodologías de gestión e inteligencia de riesgos, con el propósito de priorizar controles y proponer estrategias de protección que reduzcan la superficie de ataque, con pensamiento crítico, integridad y responsabilidad social.

Contenido

- 2.1. Conceptos fundamentales
 - 2.1.1. Amenaza, vulnerabilidad y riesgo
 - 2.1.2. Agentes de amenaza y vectores de ataque
 - 2.1.3. Relación con marcos internacionales
- 2.2. Clasificación de amenazas y vulnerabilidades
 - 2.2.1.1. Humanas
 - 2.2.1.2. Tecnológicas
 - 2.2.1.3. Emergentes
- 2.3. Fundamentos de la gestión de riesgos aplicada a la ciberseguridad
 - 2.3.1. Identificación y clasificación de activos y datos
 - 2.3.2. Estimación de impacto y probabilidad
 - 2.3.3. Priorización del riesgo
 - 2.3.4. Estrategias de tratamiento del riesgo

Duración

7 horas

Unidad III. Controles y mecanismos de protección del ciberespacio

Competencia

Elegir controles de protección en sistemas, servicios y flujos de datos, con base en marcos internacionales y buenas prácticas, con el propósito de reducir la superficie de ataque y garantizar la confidencialidad, integridad y disponibilidad de la información, con disciplina operativa, trabajo en equipo y responsabilidad.

Contenido

- 3.1. Principios de selección de controles
 - 3.1.1. Relación entre riesgos priorizados y medidas de protección
- 3.2. Gestión de identidad y acceso
 - 3.2.1. Autenticación y control de privilegios
 - 3.2.2. Modelos de control de acceso
- 3.3. Endurecimiento de sistemas
 - 3.3.1. Configuración segura y gestión de parches
 - 3.3.2. Validación con guías y benchmarks reconocidos
- 3.4. Protección de datos y aplicaciones
 - 3.4.1. Estados del dato y cifrado
 - 3.4.2. Clasificación de la información y controles básicos de protección
- 3.5. Seguridad de redes y servicios
 - 3.5.1. Principios de Zero Trust y segmentación
 - 3.5.2. Monitoreo, detección y respuesta en redes y servicios
- 3.6. Seguridad en la nube
 - 3.6.1. Responsabilidad compartida proveedor-usuario
 - 3.6.2. Estrategias de configuración y monitoreo seguro

- 3.7. Continuidad y recuperación
 - 3.7.1. Estrategias de respaldo y planes básicos de continuidad

Duración

7 horas

Unidad IV. Operaciones de ciberseguridad: Perspectivas Red, Blue y Purple Team

Competencia

Ejecutar auditorías técnicas de seguridad desde las perspectivas Red, Blue y Purple Team, con metodologías internacionales y marcos operativos, con el propósito de identificar vulnerabilidades, evaluar controles defensivos y proponer mejoras continuas en la protección de sistemas, servicios y datos, con disciplina operativa, trabajo en equipo y responsabilidad.

Contenido

- 4.1. Auditorías y análisis forense
 - 4.1.1. Metodologías de auditoría de seguridad (referentes internacionales)
 - 4.1.2. Preservación de evidencia y cadena de custodia
 - 4.1.3. Análisis forense inicial en sistemas y servicios
- 4.2. Perspectiva Red Team
 - 4.2.1. Fases de un ataque estructurado con Cyber Kill Chain)
 - 4.2.2. Técnicas ofensivas según MITRE ATT&CK
 - 4.2.3. Ejercicios de explotación controlada e ingeniería social
- 4.3. Perspectiva Blue Team
 - 4.3.1. Detección y respuesta ante incidentes
 - 4.3.2. Monitoreo y análisis de registros
 - 4.3.3. Threat hunting básico con indicadores de ataque y compromiso
- 4.4. Perspectiva Purple Team
 - 4.4.1. Integración de tácticas ofensivas y defensivas
 - 4.4.2. Medición de cobertura y métricas de defensa
 - 4.4.3. Elaboración de reportes técnicos y ejecutivos para mejora continua

Duración

6 horas

Unidad V. Tecnologías emergentes en el ciberespacio

Competencia

Evaluar el impacto técnico, operativo y ético de tecnologías emergentes en el ciberespacio, mediante el análisis de riesgos con base en marcos internacionales y criterios técnicos, con el propósito de anticipar riesgos y oportunidades y proponer estrategias de protección innovadoras que fortalezcan la resiliencia digital, actuando con pensamiento crítico, integridad profesional y apertura a la actualización continua.

Contenido

- 5.1. Inteligencia artificial y ciberseguridad
 - 5.1.1. Usos defensivos y ofensivos de la IA
 - 5.1.2. Riesgos de manipulación y confianza en modelos
- 5.2. Criptografía poscuántica y computación cuántica
 - 5.2.1. Algoritmos de nueva generación y escenarios de migración
 - 5.2.2. Impacto en la protección de datos y comunicaciones seguras
- 5.3. Ciberseguridad en IoT y sistemas ciberfísicos
 - 5.3.1. Riesgos en entornos domésticos, industriales y médicos
 - 5.3.2. Retos de edge computing y convergencia IT/OT
- 5.4. Web3 y ecosistemas descentralizados
 - 5.4.1. Seguridad en blockchain y contratos inteligentes
 - 5.4.2. Identidad digital, trazabilidad y privacidad
- 5.5. Escenarios tecnosociales y éticos
- 5.6. Retos de la vigilancia masiva
- 5.7. Interacción humana con tecnologías autónomas

Duración

6 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Fundamentos de la Ciberseguridad

Práctica 1: Mapeo inicial de vulnerabilidades en un entorno controlado

Duración

5 horas

Procedimiento

Observación inicial

Las prácticas de esta unidad y de las subsecuentes se desarrollan bajo la perspectiva de distintos roles de ciberseguridad (a los que provisionalmente llamaremos descubridor, analista y protector) de manera progresiva, simulando el flujo de trabajo real de los profesionales. El docente proporciona acceso a un servidor de virtualización con las máquinas virtuales necesarias para la práctica.

Acciones comunes

1. Accede al entorno de virtualización asignado.
2. Inicia las máquinas virtuales: un sistema atacante (Linux con herramientas de seguridad) y un sistema objetivo (VM vulnerable).

Rol protector (monitorización y evidencia)

3. Levanta un capturador de tráfico en la máquina víctima.
4. Verifica que se registren los paquetes de entrada durante toda la práctica.
5. Al finalizar, detiene la captura y guarda el archivo como evidencia.

Rol descubridor (exploración activa)

6. Descubre el segmento de red al que pertenece la máquina objetivo mediante análisis del tráfico local.
7. Identifica una dirección IP libre dentro del segmento y configura la tarjeta de red en la máquina atacante.
8. Ejecuta un escaneo inicial sobre el rango identificado.
9. Selecciona un objetivo con base en los resultados obtenidos.
10. Realiza un escaneo exhaustivo sobre el objetivo seleccionado.
11. Identifica el sistema operativo y la versión estimada, enumera todos los servicios detectados y aprovecha los scripts de detección automática.
12. Prioriza los servicios expuestos según su nivel de riesgo (alto, medio, bajo), considerando factores como exposición, criticidad del servicio y potencial de explotación.
13. Elabora un mapa de superficie de ataque inicial, identificando puntos de entrada potenciales (puertos abiertos, servicios expuestos, versiones obsoletas).

Rol analista (procesamiento y organización de resultados)

14. Convierte los resultados a un formato estructurado, resaltando campos relevantes:
 - a. IP
 - b. Puerto
 - c. Protocolo
 - d. Servicio
 - e. Versión
 - f. Sistema operativo
 - g. Nivel de riesgo (Bajo, Medio, Alto o Crítico)
15. Organiza un dataset estructurado con base en el reporte de descubrimiento.
16. Integra un reporte técnico preliminar de hallazgos que resuma:
 - a. Servicios expuestos.
 - b. Versiones detectadas.
 - c. Observaciones iniciales sobre riesgos potenciales.
17. Entrega el dataset y el reporte al docente, servirán de insumo de la siguiente práctica.

Recursos de apoyo

- Servidor de virtualización con VMs (atacante y objetivo)
- Herramientas de descubrimiento de protocolos de red
- Editor de texto y scripts para conversión de resultados
- Plantilla de reporte de hallazgos

Práctica 2: Análisis de hallazgos y correlación con principios de ciberseguridad

Duración

5 horas

Procedimiento

Observación inicial

Esta práctica utiliza como insumo los productos de la Práctica 1: dataset estructurado, reporte técnico preliminar y archivo de captura. Los estudiantes desempeñan roles diferenciados (descubridor, analista y protector), y en conjunto elaboran un informe integrador donde relacionan los hallazgos prácticos con los principios y atributos de la ciberseguridad revisados en clase.

Acciones comunes

1. Reúne los entregables de la práctica anterior: dataset estructurado, reporte de hallazgos y archivo evidencia.pcap.
2. Verifica la integridad de las evidencias y establece línea de tiempo inicial.

Rol descubridor (documentación de actividad ofensiva)

3. Revisa el escaneo realizado en la Práctica 1 y reconstruye la metodología usada.

4. Identifica si se aprovecharon técnicas automatizadas.
5. Evalúa si el ataque simulado fue eficiente (cobertura vs. ruido generado).
6. Documenta hallazgos con relación a tácticas MITRE ATT&CK (TA0043: Reconnaissance).

Rol analista (procesamiento de datos y evidencias)

7. Analiza el archivo .pcap con la herramienta disponible.
8. Extrae métricas de tráfico relevantes:
 - a. Ratio SYN/ACK anormal (indicador de escaneo).
 - b. Conexiones a puertos no estándar (ej. 3306, 5900).
 - c. Patrones de escaneo identificables (ráfagas, intervalos fijos).
9. Correlaciona eventos de red con logs del sistema (si están disponibles).
10. Mapea eventos críticos a la tríada CIA:
 - a. ¿Qué servicio comprometido afecta confidencialidad?
 - b. ¿Cuál pone en riesgo integridad o disponibilidad?
11. Construye una línea de tiempo visual (diagrama de secuencia) del ataque simulado.

Rol protector (detección y propuesta defensiva)

12. Evalúa qué controles podrían haber detectado el escaneo (firewall, IDS, logging).
13. Propone indicadores de anomalía (IoA) basados en las métricas obtenidas.
14. Consulta el checklist NIST CSF (Identify/Detect) y asocia controles aplicables.
15. Formula 3 recomendaciones técnicas para mejorar la detección temprana.

Trabajo integrador (equipo completo)

16. Elabora un informe conjunto que integre:
 - a. Narrativa del ataque (descubridor).
 - b. Dataset procesado, métricas y diagrama de eventos (analista).
 - c. Evaluación de controles y propuestas de mejora (protector).
17. Genera un resumen ejecutivo (máximo 1 página) con:
 - a. Hallazgos principales.
 - b. Riesgos asociados a la exposición de servicios.
 - c. Recomendaciones prioritarias.
18. Entrega el informe y el diagrama al docente para retroalimentación

Recursos de apoyo

- Dataset y reporte de la Práctica 1
- Archivo evidencia.pcap
- Sistema víctima con bitácoras accesibles
- Herramientas de análisis de tráfico
- Procesador de texto y software para diagramas
- Plantilla de informe integrador

Unidad II: Amenazas, vulnerabilidades y riesgos asociados al ciberespacio

Práctica 3: Identificación de amenazas, vulnerabilidades y riesgos en un entorno controlado

Duración

5 horas

Procedimiento

Observación inicial

Esta práctica enlaza los resultados de la Unidad I con los contenidos de la Unidad II. Los estudiantes utilizan la información y productos generados en las prácticas previas como insumo para identificar y clasificar amenazas, vulnerabilidades y riesgos. El énfasis está en aplicar los fundamentos metodológicos de la gestión de riesgos para identificar y clasificar amenazas, asociar vulnerabilidades a activos, y priorizar riesgos utilizando una matriz cualitativa, de manera que el ejercicio se mantenga vigente aun cuando cambien las técnicas o los datos disponibles en futuros escenarios.

Acciones comunes

1. Reúne los entregables de las prácticas anteriores: dataset estructurado, informe técnico y línea de tiempo del ataque simulado.
2. Establece el contexto del escenario: "Universidad ficticia con sistemas académicos, correo institucional y portal web".
3. Define activos críticos: servidor de datos estudiantiles, sistema de autenticación, sitio web público.
4. Establece un repositorio común de trabajo para integrar resultados por rol.

Rol descubridor (identificación de vulnerabilidades)

5. Describe cada activo seleccionado y justifica su criticidad (impacto en confidencialidad, integridad, disponibilidad).
6. Asocia a cada activo posibles vulnerabilidades genéricas:
 - a. Configuración insegura
 - b. Software obsoleto
 - c. Contraseña débil
 - d. Error humano
7. Documenta cada vulnerabilidad con una descripción abstracta (ejemplo: servicio expuesto sin cifrado → riesgo de divulgación no autorizada).
8. Asocia al menos una amenaza plausible por vulnerabilidad:
 - a. Humana (phishing, negligencia, ingeniería social)
 - b. Tecnológica (malware, botnet, ransomware)
 - c. Falla digital (servicios expuestos, software sin parche)

Rol analista (clasificación y métricas de riesgo)

9. Para cada combinación activo-vulnerabilidad-amenaza, estima:

- a. Probabilidad: Bajo, Medio, Alto, Crítico (basado en exposición, histórico de incidentes, controles existentes)
 - b. Impacto: Bajo, Medio, Alto, Crítico (basado en alcance, sensibilidad de datos, criticidad del servicio)
10. Usa una matriz de riesgo 4x4 proporcionada por el docente para asignar el nivel de riesgo global (Bajo, Medio, Alto, Crítico).
 11. Mapea las tácticas utilizadas en prácticas previas a MITRE ATT&CK.
 12. Calcula un índice de exposición relativo (puntaje simple: suma de riesgos altos/críticos).
 13. Prioriza las top 5 vulnerabilidades según riesgo global.

Rol protector (evaluación y estrategias de control)

14. Consulta el checklist NIST CSF (Identify / Protect) y asocia controles aplicables a cada riesgo prioritario.
15. Propone estrategias de tratamiento para cada uno:
 - a. Mitigar (aplicar parche, endurecer servicio)
 - b. Transferir (asegurar riesgo, si aplica)
 - c. Aceptar (solo si justificación técnica es válida)
 - d. Evitar (desactivar servicio no esencial)
16. Formula tres medidas concretas (técnica, organizacional, procedimental) basadas en buenas prácticas de ciberseguridad operativa y principios de endurecimiento, monitoreo y gestión de accesos.
17. Evalúa brevemente el ROI de seguridad (Return on Security Investment) para una medida crítica: ¿cuál es el costo-beneficio esperado?

Trabajo integrador (equipo completo)

18. Elabora un informe conjunto que incluya:
 - a. Inventario de activos y su criticidad (descubridor)
 - b. Matriz de riesgos completa y top 5 priorizados (analista)
 - c. Propuesta de tratamiento y controles recomendados (protector)
19. Genera un resumen ejecutivo (máximo 1 página) con:
 - a. Hallazgos principales
 - b. Riesgos críticos identificados
 - c. Recomendaciones prioritarias
20. Entrega el informe y el resumen al docente; servirá de insumo para la siguiente unidad.

Recursos de apoyo

- Dataset y reporte de las prácticas 1 y 2
- Plantilla de matriz de riesgo 4x4 (PDF/impresión/digital)
- Guía de clasificación de probabilidad e impacto (criterios definidos)
- Checklist NIST CSF (secciones Identify y Protect)
- Referencias a buenas prácticas de ciberseguridad (ej. CIS, CISA KEV, ANSSI, ENISA) y principios de hardening, gestión de parches y monitoreo de eventos.
- Plantilla de informe técnico integrador

Práctica 4: Simulación de ataque de Ingeniería Social avanzada desde la planeación hasta la post-explotación

Duración

5 horas

Procedimiento

Observación inicial

Esta práctica integra las dimensiones humana y técnica de un ataque, combinando tácticas de Ingeniería Social (IS) con abuso controlado de herramientas nativas del sistema y acciones de post-explotación simuladas, con el propósito de mostrar cómo las amenazas humanas y técnicas se combinan para incrementar el riesgo.

Acciones comunes

1. Accede al entorno preconfigurado con escenario fijo:
 - a. Objetivo: "Universidad ficticia con portal web y correo".
 - b. Máquinas desplegadas (sandbox):
 - i. Estación atacante
 - ii. Estación víctima
 - iii. Servidor web/sistema académico
 - iv. Servidor de correo/servicios.
2. Inicia la captura de telemetría y logging mediante la herramienta/script provisto por el docente.

Rol descubridor (planeación y ejecución del ataque)

3. Emplea la plantilla de campaña de phishing proporcionada por el docente.
4. Adapta el contenido al contexto de la organización simulada (logos, texto mínimo) sin incorporar material sensible.
5. Realiza la entrega simulada mediante el mecanismo local del laboratorio (no se usa infraestructura pública).
6. Activa el artefacto de persistencia preconfigurado en la VM objetivo preparado por el docente.
7. Documenta las acciones con evidencia de tiempo (screenshots, registros de envío) relacionándolas con la táctica ofensiva correspondiente del MITRE ATT&CK.

Rol analista (recolección y análisis)

8. Recolecta automáticamente las evidencias (telemetría, logs, pcap) exportadas por el laboratorio.
9. Carga las evidencias en la herramienta/notebook provista y estructura un dataset (CSV/JSON) con campos:
 - a. Táctica
 - b. Técnica ATT&CK

- c. Timestamp
 - d. Host
 - e. Descripción del evento
10. Visualiza la línea de tiempo de eventos, detecta anomalías relevantes y marca eventos clave.
 11. Mapea los eventos a MITRE ATT&CK y construye un diagrama de la secuencia (mini killchain).

Rol protector (detección y mitigación)

12. Correlaciona evidencias para evaluar si los controles del entorno simulado detectaron la campaña.
13. Estima probabilidad e impacto con la matriz de riesgo proporcionada.
14. Consulta checklist NIST CSF simplificado y determina controles relevantes.
15. Propone 3 medidas concretas (técnica / organizacional / procedimental) conforme a la guía de acciones de respuesta a incidentes.

Rol víctima (interacción simulada)

16. Interactúa en la estación víctima siguiendo un guión controlado.
17. Completa el cuestionario sobre percepción y factores cognitivos (formato corto).

Trabajo integrador (equipo completo)

18. Completa una tabla de análisis de la táctica con columnas:
 - a. Objetivo seleccionado.
 - b. Vulnerabilidad humana/técnica explotada.
 - c. Amenaza asociada.
 - d. Evidencia observada.
 - e. Riesgo (impacto/probabilidad).
 - f. Estrategia de tratamiento.
19. Elabora un informe conjunto que contenga:
 - a. Planificación del ataque (descubridor).
 - b. Dataset y evidencias procesadas (analista).
 - c. Evaluación de riesgos y controles (protector).
 - d. Relato de la experiencia de la víctima.
 - e. Reflexión sobre cómo la IA puede amplificar la efectividad de la Ingeniería Social.
20. Genera resumen ejecutivo (máximo 1 página) con 3 medidas prioritarias.
21. Presenta hallazgos ante el grupo y docente; recibe retroalimentación.

Recursos de apoyo

- Plantilla de campaña de phishing
- Scripts de LOLBIN funcionales
- Entorno de análisis de datos con herramientas abiertas
- Matriz de riesgo imprimible
- Checklist de NIST CSF simplificado

- Guía de acciones de respuesta a incidentes (detección, contención, recuperación)
- Cuestionario de percepción y factores cognitivos
- Plantilla de informe técnico integrador

Unidad III: Controles y mecanismos de protección del ciberespacio

Práctica 5: Gestión de identidad y acceso

Duración

4 horas

Procedimiento

Observación inicial

El control de identidad y acceso regula el uso de recursos y servicios críticos. Los modelos DAC, MAC y RBAC permiten distintos niveles de granularidad en la autorización, mientras que la autenticación multifactor (MFA) refuerza la confianza en la verificación de identidad. En esta práctica se exploran configuraciones y vulnerabilidades derivadas de una mala gestión de accesos en el sistema operativo, con el propósito de elegir controles de protección basados en marcos internacionales (NIST CSF, ISO/IEC 27001) que reduzcan la superficie de ataque y garanticen la confidencialidad, integridad y disponibilidad de la información.

Acciones comunes

1. Accede al entorno preconfigurado en Proxmox:
 - a. VM atacante (Linux con herramientas básicas).
 - b. VM servidor vulnerable (Ubuntu con configuraciones inseguras).
2. Verifica que los logs de autenticación (auth.log, journalctl) estén activos y sean accesibles.
3. Establece un repositorio común para documentar hallazgos por rol.

Rol descubridor

4. Identifica cuentas y servicios críticos mal configurados:
 - a. Usuario admin con contraseña débil (admin123).
 - b. Acceso SSH con PermitRootLogin yes.
 - c. Archivos sensibles legibles por todos (ej. /etc/shadow).
5. Intenta escalar privilegios o acceder a recursos restringidos usando credenciales conocidas o técnicas simples (sin explotación activa).
6. Documenta cada intento con evidencia temporal (screenshots, logs).

Rol analista

7. Analiza los logs generados durante las acciones del descubridor.
8. Cuantifica eventos críticos:
 - a. a. Número de accesos fallidos.
 - b. b. Tiempo entre intentos.
 - c. c. Recursos accedidos sin autorización.
9. Mapea hallazgos a controles específicos del NIST CSF y del ISO/IEC 27001.
10. Propone qué controles podrían haber prevenido cada hallazgo.

Rol protector

11. Aplica medidas de mitigación basadas en buenas prácticas:
 - a. Cambia contraseñas débiles por claves robustas.
 - b. Deshabilita acceso directo de root por SSH.
 - c. Ajusta permisos DAC en archivos sensibles (chmod 600 /etc/shadow).
 - d. Configura grupos y políticas RBAC para usuarios estándar.
 - e. Activa autenticación multifactor (MFA) en SSH usando un método basado en factores posescriptorio (por ejemplo, códigos temporales generados localmente, como TOTP u otros mecanismos equivalentes).
12. Valida la efectividad de los cambios realizando una prueba controlada.

Rol víctima

13. Describe cómo un acceso indebido afectaría la confidencialidad, integridad o disponibilidad de los datos.
14. Reflexiona brevemente sobre la experiencia de usuario en un sistema seguro vs. uno inseguro (balance entre usabilidad y seguridad).

Trabajo integrador

15. Elabora una tabla con columnas:
 - a. Vulnerabilidad identificada.
 - b. Modelo de acceso comprometido (DAC, RBAC, etc.).
 - c. Evidencia observada.
 - d. Control propuesto (según NIST CSF o ISO/IEC 27001).
 - e. Estado tras mitigación (resuelto/pendiente).
16. Genera un informe conjunto que incluya:
 - a. Hallazgos (descubridor).
 - b. Métricas y mapeo a marcos (analista).
 - c. Controles aplicados (protector).
 - d. Impacto percibido (víctima).

Recursos de apoyo

- Servidor de virtualización con VMs preconfiguradas
- Plantilla de tabla de controles
- Checklist NIST CSF
- Referencia de controles ISO/IEC 27001
- Guía básica de configuración de MFA (TOTP)

Práctica 6: Hardening y gestión de parches

Duración

4 horas

Procedimiento

Observación inicial

Esta práctica permite la aplicación de técnicas de endurecimiento (hardening) en un sistema operativo previamente identificado como vulnerable. Se aplicaran buenas prácticas y referencias reconocidas (CIS Controls, NIST CSF) para reducir la superficie de ataque mediante la aplicación de controles para la deshabilitación de servicios innecesarios, actualización de software, endurecimiento de servicios críticos y verificación de cambios.

Acciones comunes

1. Accede al entorno preconfigurado en Proxmox:
 - a. VM atacante (Linux con herramientas básicas).
 - b. VM servidor vulnerable (Ubuntu/Debian con servicios innecesarios activos).
2. Verifica que los logs del sistema estén activos (journalctl, syslog).
3. Establece un repositorio común para documentar hallazgos por rol.

Rol descubridor

3. Ejecuta un escaneo rápido del sistema objetivo:
 - a. Identifica servicios innecesarios.
 - b. Detecta paquetes desactualizados.
 - c. Busca reglas de firewall inadecuadas.
4. Documenta hallazgos con evidencia temporal.

Rol analista

6. Clasifica los hallazgos según impacto potencial:
 - a. Servicios expuestos sin autenticación.
 - b. Software sin parche con CVE conocido.
 - c. Configuraciones inseguras (SSH con contraseña, root login).
7. Asocia cada hallazgo a controles de marcos internacionales:
 - a. CIS Controls.
 - b. Guía de endurecimiento del NIST SP.
8. Propone prioridad de corrección: Crítico, Alto, Medio.

Rol protector

9. Aplica medidas de mitigación basadas en buenas prácticas:
 - a. Deshabilita servicios innecesarios.
 - b. Actualiza todos los paquetes.
 - c. Configura el firewall.
 - d. Endurece el servicio de conexión remota.
 - e. Reinicia servicios.

10. Valida los cambios con comandos de gestión básica.

Trabajo integrador (equipo completo)

11. Completa una tabla única con las siguientes columnas:

- a. Vulnerabilidad
- b. Riesgo (Crítico/Alto/Medio)
- c. Control aplicado
- d. Estado final (Mitigado/Pendiente)

12. Elabora un informe comparativo antes/después que incluya:

- a. Hallazgos iniciales (descubridor).
- b. Mapeo a marcos y priorización por riesgo (analista).
- c. Registro de controles aplicados y resultados de validación (protector).

13. Entrega el informe técnico con sus evidencias al docente.

Recursos de apoyo

- Servidor de virtualización con VMs preconfiguradas
- Plantilla de tabla de controles.
- Checklist simplificado de CIS Controls
- Guía de endurecimiento de referencia del NIST SP

Práctica 7: Protección de datos y aplicaciones

Duración

4 horas

Procedimiento

Observación inicial

La protección de la información debe considerar sus diferentes estados: en tránsito, en reposo y en uso. Esta práctica permite aplicar controles técnicos para proteger datos en tránsito mediante cifrado TLS y en reposo mediante cifrado de almacenamiento, con base en marcos internacionales como ISO/IEC 27002. También se explora cómo las vulnerabilidades comunes en aplicaciones web pueden exponer datos sensibles, reforzando la necesidad de defensa en profundidad.

Acciones comunes

1. Accede al entorno preconfigurado en Proxmox:
 - a. VM servidor web sin protección de cifrado de tráfico.
 - b. VM cliente atacante con herramientas de análisis de tráfico.
2. Habilita logging en el servidor web.
3. Establece un repositorio común para evidencias.

Rol descubridor

4. Captura tráfico HTTP en claro durante una transacción simulada.

5. Identifica campos sensibles visibles en el .pcap.
6. Intenta acceder a un archivo crítico sin cifrar (ej. /data/alumnos.csv).
7. Explora una vulnerabilidad básica de aplicación web (proporcionada):
 - a. Inyección de entrada en formulario de búsqueda.
 - b. Visualiza respuesta que revela estructura de base de datos.
8. Documenta hallazgos con screenshots y extractos del tráfico.

Rol analista

9. Compara tráfico cifrado vs. no cifrado usando una captura de referencia proporcionada.
10. Cuantifica riesgo:
 - a. Número de paquetes con datos sensibles expuestos.
 - b. Tiempo promedio hasta detección (simulado).
11. Mapea hallazgos a controles de ISO/IEC 27002:
 - a. Seguridad en comunicaciones
 - b. Almacenamiento seguro de datos
12. Relaciona la vulnerabilidad explorada con categorías generales de riesgos en aplicaciones web, según referencias como OWASP Top 10 (ej. inyección, manejo inseguro de datos, configuración incorrecta).

Rol protector

13. Configura el cifrado de tráfico TLS en el servidor web:
 - a. Genera un certificado autofirmado.
 - b. Instala en el servidor web.
 - c. Habilita solo cipher suites actuales.
14. Valida la configuración del cifrado.
15. Aplica cifrado en reposo a directorio crítico:
 - a. Crea volumen LUKS con clave de 256 bits.
 - b. Monta y mueve archivo sensible.
 - c. Realiza backup del header LUKS.
16. Verifica acceso solo con clave correcta.

Trabajo integrador

17. Completa una tabla con columnas:
 - a. Estado del dato (en tránsito / en reposo)
 - b. Vulnerabilidad identificada
 - c. Control aplicado
 - d. Evidencia de mitigación
18. Elabora un informe breve que incluya:
 - a. Hallazgos iniciales (descubridor)
 - b. Métricas y mapeo a ISO/IEC 27002 (analista)
 - c. Controles aplicados y resultados (protector)
19. Entrega el informe, el .pcap y las configuraciones al docente.

Recursos de apoyo

- Servidor de virtualización con VMs preconfiguradas

- Guía de configuración de TLS con certificados autofirmados
- Guía de cifrado de disco con LUKS (Linux Unified Key Setup)
- Dataset simulado de información académica (anónimo)
- Referencia conceptual del OWASP Top 10 (categorías de riesgo, no checklist técnico)
- Secciones pertinentes de la norma ISO/IEC 27002

Práctica 8: Redes y servicios seguros

Duración

5 horas

Procedimiento

Observación inicial

La seguridad tradicional basada en perímetro ha quedado obsoleta frente a amenazas internas, movilidad de usuarios, infraestructura en la nube y ataques sofisticados que evaden detección. El paradigma Zero Trust ("nunca confíes, siempre verifica") surge como respuesta a este escenario, eliminando la confianza implícita y exigiendo autenticación, autorización y validación continua para cada acceso, sin importar su origen. En esta práctica se simula un entorno segmentado donde se aplica este modelo mediante políticas de acceso estricto, monitoreo constante y defensa en profundidad, reforzando su relevancia operativa en contextos reales de ciberseguridad.

Acciones comunes

1. Accede al entorno preconfigurado en Proxmox:
 - a. Topología con VLANs básicas: Data (usuarios), Servers (servicios), Mgmt (administración).
 - b. VM atacante
 - c. VM servidor web víctima
 - i. Firewall
 - ii. Dispositivo de red con funciones de firewall e IDS/IPS (preconfigurado)
2. Verifica que el logging esté activo en la MV víctima.
3. Establece un repositorio común para documentar hallazgos por rol.

Rol descubridor

4. Ejecuta escaneos controlados sobre las zonas objetivo.
 - a. Identifica servicios accesibles desde la subred Data.
 - b. Intenta acceder a recursos en Mgmt o Servers sin autorización.
5. Documenta intentos con evidencia temporal (screenshots, salida de comandos).

Rol analista

6. Recolecta métricas de tráfico:
 - a. Paquetes bloqueados vs permitidos (firewall).
 - b. Alertas generadas por el sistema de detección de intrusos (IDS).

7. Correlaciona eventos con intentos de intrusión.
8. Mapea hallazgos a:
 - a. a. NIST CSF (Protect/Detect)
 - b. b. Principios de Zero Trust (NIST SP)
9. Identifica brechas: ¿qué tácticas no fueron detectadas?

Rol protector

10. Aplica política de default deny en el dispositivo de red:
 - a. Bloquea todo tráfico entrante, excepto conexiones establecidas y loopback.
 - b. Permite solo tráfico necesario:
 - i. SSH desde Mgmt
 - ii. HTTP/HTTPS desde Data
 - iii. DNS hacia servidores autorizados
11. Habilita reglas de detección para identificar actividades sospechosas:
 - a. Escaneo de puertos
 - b. Tráfico inusual entre zonas
 - c. Intentos de conexión a servicios bloqueados
12. Valida los cambios realizando un nuevo escaneo controlado.

Trabajo integrador

13. Completa una tabla con columnas:
 - a. Segmento de red
 - b. Amenaza explorada
 - c. Evidencia observada
 - d. Control aplicado
 - e. Resultado (Detectado/Bloqueado/Pendiente)
14. Elabora un informe breve que combine:
 - a. Configuraciones aplicadas (protector)
 - b. Métricas de detección (analista)
 - c. Reflexión sobre el impacto del modelo Zero Trust
15. Entrega el informe y las evidencias al docente.

Recursos de apoyo

- Servidor de virtualización con topología preconfigurada (Proxmox)
- Plantilla de tabla de controles de red
- Guía de comandos básicos de firewall
- Referencia de reglas IDS/IPS básicas (ET Open)
- Checklist NIST CSF (funciones Protect y Detect)
- Documento resumen de NIST SP (Zero Trust Architecture)

Unidad IV: Operaciones de ciberseguridad: Perspectivas Red, Blue y Purple Team

Práctica 9: Auditoría de un ciberataque: de la planeación a la sanción

Duración

7 horas

Procedimiento

Observación inicial

Se aplicarán las metodologías vigentes en ciberseguridad, junto con marcos de referencia internacionales y la legislación aplicable al contexto.

Acciones comunes

1. Accede al entorno preconfigurado en Proxmox:
 - a. VMs: servidor web, sistema académico, correo, estación víctima, atacante.
 - b. Topología segmentada (Data, Servers, Mgmt).
2. Activa captura de tráfico y logging en todas las VMs.
3. Establece un repositorio común para evidencias (logs, .pcap, screenshots, reportes).
4. Divide al grupo en equipos con roles: Red Team, Blue Team, Purple Team.

Rol Red Team (planeación y ataque)

5. Aplica PTES para estructurar la auditoría ofensiva.
6. Ejecuta fases de Cyber Kill Chain:
 - a. Reconocimiento: escaneo de red con nmap (servicios expuestos)
 - b. Intrusión: explotación controlada (payload preconfigurado, sin malware real)
 - c. Persistencia: creación de usuario temporal o backdoor benigno
 - d. Exfiltración: copia simulada de archivo académico (1 MB)
7. Documenta cada fase con screenshots y registros de comandos.
8. Mapea tácticas utilizadas a MITRE ATT&CK.

Rol Blue Team (detección, respuesta y análisis forense)

9. Monitorea alertas de firewall e IDS/IPS.
10. Correlaciona eventos con logs del sistema (auth.log, syslog).
11. Identifica indicadores de compromiso (IoCs):
 - a. Nuevos usuarios, conexiones inusuales, cambios en archivos críticos.
12. Realiza análisis forense básico:
 - a. Captura de memoria volátil (script proporcionado).
 - b. Extracción de hashes y metadatos.
13. Preserva evidencias con cadena de custodia digital (plantilla provista).
14. Elabora un reporte forense con:
 - a. Evidencias recolectadas
 - b. Procedimiento aplicado
 - c. Resultados técnicos

Rol Purple Team (integración y mejora continua)

15. Integra hallazgos de Red y Blue Team en un mapeo con ATT&CK Navigator.
16. Mide métricas de defensa:
 - a. a. MTTD (Mean Time to Detect)
 - b. b. MTTR (Mean Time to Respond)
17. Evalúa cobertura: tácticas detectadas vs. no detectadas.
18. Elabora un reporte ejecutivo con:
 - a. a. Narrativa del incidente
 - b. b. Defensas que funcionaron/fallaron
 - c. c. Recomendaciones de mejora

Trabajo integrador (equipo completo)

19. Elabora un expediente de caso con:
 - a. a. Evidencias preservadas (logs, .pcap, capturas)
 - b. b. Cadena de custodia
 - c. c. Mapas kill chain ofensiva/defensiva
 - d. d. Informe forense + reporte ejecutivo
 - e. e. Métricas de detección
20. Entrega el expediente digital.
21. Graba un video institucional (5–7 min) donde se presenten:
 - a. a. Fases del ataque (Red Team)
 - b. b. Respuesta defensiva (Blue Team)
 - c. c. Hallazgos clave (Purple Team)
 - d. d. Conclusión sobre impacto legal
22. Edita video fuera de clase; entrega final en próxima sesión.

Recursos de apoyo

- Servidor Proxmox con VMs preconfiguradas (snapshots limpios)
- Scripts de reset automático
- Plantillas: cadena de custodia, informe forense, reporte ejecutivo
- Herramientas disponibles para escaneo de red, captura y análisis de tráfico, monitoreo de sistemas y análisis forense
- MITRE ATT&CK Navigator (versión offline)
- Checklist NIST SP, PTES, ISO/IEC 27037
- Guion de video aprobado previamente (obligatorio antes de grabar)

Unidad V. Tecnologías emergentes en el ciberespacio

Práctica 10: Criptografía poscuántica

Duración

7 horas

Procedimiento

Observación inicial

Se aprovechará la disponibilidad en la Facultad de Ciencias de una computadora cuántica SPINQ de 2 qubits como recurso demostrativo y motivacional. Su función será ilustrar principios básicos de la computación cuántica —como el entrelazamiento y la distribución cuántica de claves (BB84)— exclusivamente con fines conceptuales, con el objetivo de contextualizar el impacto de la computación cuántica en los sistemas criptográficos actuales.

El énfasis de la práctica se centra en la evaluación de riesgos y la definición de estrategias de protección basadas en criptografía poscuántica (PQC), alineadas con el proceso de estandarización del NIST Post-Quantum Cryptography Project y con marcos de gestión de seguridad como ISO/IEC 27001. No se busca la programación avanzada ni el dominio de herramientas específicas, sino la toma de decisiones técnicas informadas y la planeación de migraciones viables.

Acciones comunes

1. Accede al entorno preconfigurado proporcionado por el docente:
 - a. Scripts de simulación (BB84, Grover).
 - b. Tablas comparativas: RSA vs. Kyber, ECC vs. Dilithium.
 - c. Plantillas de informe técnico y roadmap de migración.
2. Recibe el escenario simulado: "Universidad con datos sensibles que requieren confidencialidad a largo plazo (>10 años).

Rol Red Team (evaluación de riesgo cuántico)

3. Identifica activos críticos que requieran secreto prolongado:
 - a. Expedientes académicos.
 - b. Información médica.
 - c. Claves privadas de infraestructura.
4. Analiza vulnerabilidades de algoritmos actuales:
 - a. RSA y ECC → rotos por algoritmo de Shor.
 - b. AES-256 → debilitado por Grover, pero aún seguro.
5. Estima horizonte de riesgo:
 - a. ¿Cuándo podría una computadora cuántica romper RSA-2048?
 - b. ¿Qué datos estarían expuestos para entonces?

Rol Blue Team (análisis técnico y métricas)

6. Compara algoritmos clásicos vs. post cuánticos:

ALGORITMO	TIPO	SEGURO FRENTE A CUÁNTICOS	TAMAÑO CLAVE PÚBLICA	RENDIMIENTO
RSA-2048	Clásico	No	256 B	Lento
Kyber-512	PQC	Sí	~800 B	Rápido
Dilithium	Firma PQC	Sí	~2 KB	Medio

7. Calcula overhead de transición:

- Incremento de ancho de banda (certificados más grandes).
- Impacto en latencia de handshake TLS.

8. Mapea los hallazgos al NIST PQC Standardization Project y a marcos de gestión como ISO/IEC 27001.

Rol Purple Team – Arquitectura y transición (crypto-agility)

9. Diseña un plan de migración híbrida para la adopción de criptografía poscuántica (PQC):

- Fase 1: Inventario de activos criptográficos.
- Fase 2: Implementación híbrida (RSA/ECC + Kyber/Dilithium).
- Fase 3: Transición completa a PQC.
- Fase 4: Monitoreo continuo y revisión de estándares.

10. Analiza la viabilidad operativa del plan considerando compatibilidad, costos y continuidad del servicio.

Rol Blue Team – Demostración complementaria (QKD)

11. Simula el protocolo BB84 mediante un script proporcionado, con el objetivo de:

- ilustrar conceptualmente la generación de claves cuánticas, y
- la detección de espionaje (tasa de error > 25 %).

12. En caso de contar con acceso a SPINQ:

- Ejecuta una demostración controlada del circuito Bell State (entrelazamiento).
- Compara resultados hardware vs. simulador (Qiskit).
- Documenta diferencias atribuibles a ruido y tiempo de coherencia.

Nota: Esta demostración tiene fines exclusivamente ilustrativos y no constituye una estrategia de mitigación poscuántica en infraestructuras institucionales.

Trabajo integrador

13. Elabora un informe técnico que incluya:

- a. Activos críticos identificados y horizonte de riesgo (Red Team).
 - b. Comparativa de algoritmos y métricas técnicas (Blue Team).
 - c. Plan de migración híbrida y roadmap de transición (Purple Team).
 - d. Mapeo explícito de decisiones técnicas a:
 - i. lineamientos del NIST PQC, y
 - ii. controles relevantes de ISO/IEC 27001.
14. Reflexión ética vinculada a decisiones técnicas:
- a. priorización de activos,
 - b. impacto social de la obsolescencia criptográfica,
 - c. equidad en la protección de información a largo plazo.
15. El informe se entrega al docente y servirá como insumo para la evaluación final.

Recursos de apoyo

- Scripts de simulación BB84 y Grover.
- Tablas comparativas: RSA vs. Kyber, ECC vs. Dilithium.
- Plantilla de roadmap de migración a PQC.
- Guía resumida del NIST Post-Quantum Cryptography Project.
- Acceso remoto a SPINQ.
- Simuladores cuánticos.

IX. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Aprendizaje basado en preguntas
- Técnica expositiva
- Debates
- Discusión guiada

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Mapas mentales
- Debates

- Mapas conceptuales
- Cuadros comparativos
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	20%
– Prácticas laboratorio	40%
– Investigación	10%
– Proyecto final	30%
Total	100%

Los instrumentos de evaluación considerados para cada componente son los siguientes:

- Evaluaciones parciales (20%): exámenes escritos con reactivos de opción múltiple y casos prácticos. Instrumento: rúbrica analítica que valora comprensión conceptual, razonamiento técnico y aplicación de principios de ciberseguridad.
- Prácticas de laboratorio (40%): entregables técnicos (bitácoras, capturas, scripts, datasets, informes). Instrumentos: lista de cotejo para verificación de entregables y rúbrica para evaluar calidad del análisis, coherencia metodológica, correcta aplicación de marcos de referencia y trabajo colaborativo.
- Investigación (10%): producto de investigación (reporte y/o exposición). Instrumento: rúbrica para evaluar pertinencia y confiabilidad de las fuentes, profundidad del análisis y claridad de la presentación.
- Proyecto final (30%): proyecto integrador (individual o en equipo de hasta tres integrantes) con defensa oral. Instrumento: rúbrica integral para evaluar la aplicación de principios y modelos de ciberseguridad, la justificación técnica basada en evidencias, la viabilidad de las propuestas y el desempeño en la defensa oral.

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Alexandrou, A. (2022). Cybercrime and information technology: Theory and practice: the computer network infrastructure and computer security, cybersecurity laws, internet of things (IoT), and mobile devices. CRC Press, Taylor & Francis Group.
<https://doi.org/10.1201/9780429318726>

Cisco Networking Academy Program. (2018). CCNA cybersecurity operations: Lab manual. Cisco Press. [Clásica].

Sarker, I. H. (2024). AI-driven cybersecurity and threat intelligence: Cyber automation, intelligent decision-making and explainability. Springer Nature Switzerland.
<https://doi.org/10.1007/978-3-031-54497-2>

Ventre, D. (2020). Artificial intelligence, cybersecurity and cyber defense. ISTE Ltd; John Wiley & Sons. <https://doi.org/10.1002/9781119707257>

Recursos de acceso libre o complementarios externos

Center for Internet Security. (2024). CIS critical security controls v8.1. CIS.
<https://www.cisecurity.org/controls>

Howard, R. (2023). Cybersecurity first principles: A reboot of strategy and tactics. Packt Publishing.

OWASP Foundation. (2024). OWASP Secure Coding Practices Quick Reference Guide.
<https://owasp.org/www-project-secure-coding-practices-quick-reference-guide/>

National Institute of Standards and Technology. (2024). Marco de ciberseguridad del NIST (CSF) 2.0 [Traducción al español]. NIST.
<https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.29.spa.pdf>

National Institute of Standards and Technology. (2024). Post-quantum cryptography standardization: Status report (NISTIR 8413). U.S. Department of Commerce.
<https://doi.org/10.6028/NIST.IR.8413>

OWASP Foundation. (2021). OWASP Top Ten: 2021. OWASP. <https://owasp.org/Top10/>

XII. PERFIL DOCENTE

La persona docente que imparta la asignatura de Ciberseguridad debe contar con formación profesional en áreas afines a las Tecnologías de la Información, como Licenciado en Ciencias Computacionales, Ciencia de Datos, Ingeniería en Computación o Ingeniería en Computación y áreas afines, preferentemente con estudios de posgrado en Tecnologías de la Información y Ciberseguridad. Debe poseer experiencia práctica en la gestión de riesgos, análisis de amenazas y vulnerabilidades, implementación de controles de seguridad. Con experiencia docente de mínimo dos años frente a grupo, además de poseer competencias pedagógicas que favorezcan el aprendizaje activo, la resolución de problemas y el trabajo colaborativo en contextos de seguridad digital.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Aprendizaje Profundo y Modelos Generativos
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 03 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos: 07
Etapas de formación:	Terminal
Carácter:	Obligatoria
Requisito:	Aprendizaje Automático
Modalidad de impartición:	Presencial
Equipo de diseño:	José Ángel González Fraga, Everardo Gutiérrez López, Antonio de Jesús García Chávez
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Esta unidad de aprendizaje tiene como propósito brindar al estudiantado los conceptos avanzados, arquitecturas y técnicas que conforman el campo del aprendizaje profundo y los modelos generativos, desarrollando habilidades para diseñar, entrenar y evaluar redes neuronales profundas, aplicar arquitecturas especializadas como redes convolucionales, recurrentes y transformers, así como implementar modelos generativos contemporáneos en diferentes contextos computacionales. Asimismo, fomenta actitudes críticas, éticas y responsables que fortalezcan el uso adecuado de metodologías de aprendizaje profundo y tecnologías generativas frente a sus implicaciones sociales, riesgos y aplicaciones en la solución de problemas de la computación y disciplinas afines. Esta asignatura, se imparte en la etapa terminal, es de carácter obligatorio y pertenece al área de conocimiento de Interacción Hombre Máquina.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Evaluar modelos avanzados de aprendizaje profundo y modelos generativos mediante la implementación de arquitecturas neuronales especializadas, el análisis de su desempeño y la interpretación de sus resultados, para valorar su efectividad, limitaciones e impacto en la solución de problemas complejos, con sentido ético, responsabilidad y conciencia de los riesgos asociados a estas tecnologías.

IV. EVIDENCIA DE APRENDIZAJE

- Elabora un reporte técnico donde se realice el análisis, la comparación y la evaluación del desempeño de modelos de aprendizaje profundo y modelos generativos mediante la implementación, experimentación, interpretación de resultados y reflexión ética. El reporte debe incluir el planteamiento del problema, la justificación del modelo seleccionado, la preparación del conjunto de datos, la arquitectura y configuración del modelo, los resultados y métricas, un análisis crítico del desempeño, identificación de limitaciones, una reflexión ética y de riesgos, y conclusiones.
- Realiza una presentación oral donde se evidencien los resultados obtenidos de su análisis de modelos, justificando sus decisiones, interpretando métricas, explicando limitaciones y discutiendo riesgos asociados al uso de modelos avanzados de IA.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar y evaluar soluciones de cómputo inteligente aplicando principios de inteligencia artificial, minería de datos, e integración de herramientas computacionales, con el propósito de apoyar la toma de decisiones en situaciones complejas propias de contextos científicos, sociales, industriales o tecnológicos, actuando con responsabilidad ética, pensamiento crítico, y colaboración interdisciplinaria.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Compromiso con la responsabilidad social, Excelencia, Vanguardia e Innovación Social.

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos del aprendizaje profundo

Competencia

Evaluar los componentes esenciales de redes neuronales profundas mediante la implementación y análisis de arquitecturas basales, técnicas de optimización, normalización y regularización, con el propósito de identificar sus fortalezas y limitaciones en tareas de aprendizaje supervisado, actuando con rigor técnico y responsabilidad académica.

Contenido

- 1.1. Redes Neuronales Profundas
 - 1.1.1. Arquitectura general (capas, parámetros, activaciones)
 - 1.1.2. Funciones de activación avanzadas (ReLU, LeakyReLU, GELU, Softmax)
- 1.2. Entrenamiento de redes profundas
 - 1.2.1. Backpropagation revisitado
 - 1.2.2. Optimizadores modernos (SGD, Momentum, Adam, RMSProp)
 - 1.2.3. Inicialización y normalización (BatchNorm, LayerNorm)
- 1.3. Regularización en redes profundas
 - 1.3.1. Dropout
 - 1.3.2. Early stopping
 - 1.3.3. Data augmentation

Duración

06 horas

Unidad II. Redes convolucionales y visión por computadora

Competencia

Analizar y evaluar arquitecturas convolucionales clásicas y modernas mediante la experimentación práctica con técnicas de transfer learning, fine-tuning y aplicaciones de clasificación y detección, con el fin de determinar su desempeño y adecuación a distintos problemas de visión por computadora, demostrando pensamiento crítico y uso responsable de los recursos computacionales.

Contenido

- 2.1. Redes Convolucionales
 - 2.1.1. Convolución, pooling y activación
 - 2.1.2. Arquitecturas clásicas: LeNet, AlexNet, VGG
- 2.2. Redes modernas
 - 2.2.1. ResNet

- 2.2.2. Inception
 - 2.2.3. MobileNet y redes ligeras
- 2.3. Técnicas de entrenamiento
 - 2.3.1. Fine-tuning
 - 2.3.2. Transfer learning
- 2.4. Aplicaciones
 - 2.4.1. Clasificación de imágenes
 - 2.4.2. Detección de objetos
 - 2.4.3. Segmentación semántica"

Duración

08 horas

Unidad III. Transformers y modelos de lenguaje

Competencia

Evaluar modelos basados en la arquitectura Transformer mediante el uso de mecanismos de atención, tokenización moderna y aplicación de modelos preentrenados para tareas de procesamiento de lenguaje natural, para interpretar su comportamiento, limitaciones y riesgos, con ética, criterio técnico y conciencia de los sesgos inherentes a los modelos de lenguaje.

Contenido

- 3.1. Arquitectura Transformer
 - 3.1.1. Mecanismo de atención
 - 3.1.2. Multi-Head Attention
 - 3.1.3. Codificación posicional
 - 3.1.4. Variantes de arquitecturas
- 3.2. Modelos de lenguaje
 - 3.2.1. Embeddings y tokenización moderna
 - 3.2.2. Modelos autoregresivos (GPT-like)
 - 3.2.3. Modelos enmascarados (BERT-like)
- 3.3. Large Language Models (LLMs)
 - 3.3.1. Concepto y entrenamiento a gran escala (Pretraining, Quantization, SFT, RLHF)
 - 3.3.2. In-context learning
 - 3.3.3. Limitaciones y alucinaciones (Riesgos, Sesgos, Alineación y Robustez)
 - 3.3.4. Métodos de evaluación de LLMs
- 3.4. Prompt Engineering
 - 3.4.1. Zero-shot, few-shot
 - 3.4.2. Chain-of-thought
 - 3.4.3. Ejemplos básicos en tareas reales

Duración

12 horas

Unidad IV. Introducción a los modelos generativos modernos***Competencia***

Analizar y evaluar modelos generativos clásicos y modernos mediante la implementación guiada de autoencoders, demostraciones prácticas de GAN y el uso de modelos de difusión preentrenados, con el propósito de valorar su capacidad de síntesis y sus implicaciones técnicas y éticas, con responsabilidad en la manipulación de contenido generado artificialmente.

- 4.1. Modelos generativos clásicos
 - 4.1.1. Autoencoders
 - 4.1.2. Variational Autoencoders (VAE)
- 4.2. Redes Generativas Antagónicas (GAN)
 - 4.2.1. Generador y discriminador
 - 4.2.2. Función adversarial
 - 4.2.3. Problemas comunes: mode collapse
- 4.3. Modelos de Difusión (Diffusion Models)
 - 4.3.1. Difusión directa e inversión
 - 4.3.2. DDPM
 - 4.3.3. Aplicaciones: imágenes, audio, video
- 4.4. Aplicaciones de modelos generativos
 - 4.4.1. Síntesis de imágenes
 - 4.4.2. Restauración y super-resolución
 - 4.4.3. Generación de texto e imágenes multimodales"

Duración

06 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO**Unidad I. Fundamentos del aprendizaje profundo****Práctica 1: Implementación de un perceptrón multicapa profundo*****Duración***

04 horas

Procedimiento

1. Atiende las orientaciones docentes sobre la arquitectura básica de redes profundas.
2. Implementa un perceptrón multicapa con al menos tres capas ocultas.
3. Prueba diferentes funciones de activación (ReLU, LeakyReLU, GELU) y registra su efecto en la convergencia.
4. Entrena el modelo en un conjunto de datos sencillo (MNIST o equivalente).
5. Registra pérdidas, exactitud y estabilidad del entrenamiento.
6. Reflexiona sobre el impacto de la profundidad y el tipo de activación en el desempeño del modelo.
7. Entrega código, gráficas y conclusiones, para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Python con librerías (PyTorch o TensorFlow)
- Notebook interactivo (Jupyter o Google Colab)
- Material de referencia proporcionado por su docente

Práctica 2: Optimización y normalización en redes profundas

Duración

04 horas

Procedimiento

1. Atiende las indicaciones docentes sobre optimizadores modernos y técnicas de estabilización del entrenamiento.
2. Implementa entrenamiento usando SGD, Momentum, Adam y RMSProp.
3. Incorpora la inicialización Xavier o He y compara estabilidad.
4. Añade BatchNorm o LayerNorm en distintas capas.
5. Registra para cada experimento la velocidad de convergencia y la calidad del modelo.
6. Analiza qué combinaciones producen modelos más estables y generalizables.
7. Entrega código, gráficas y conclusiones para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Python con librerías (PyTorch/TensorFlow)
- Notebook interactivo
- Material técnico proporcionado por su docente

Práctica 3: Regularización en aprendizaje profundo

Duración

04 horas

Procedimiento

1. Atiende las orientaciones docentes sobre overfitting y técnicas de regularización.

2. Implementa Dropout en distintas proporciones y analiza su impacto.
3. Aplica Early Stopping durante el entrenamiento del modelo.
4. Integra Data Augmentation básico (rotaciones, traslaciones, inversión horizontal).
5. Registra métricas y compara el desempeño de los modelos con y sin regularización.
6. Reflexiona sobre qué técnicas resultan más efectivas según el problema.
7. Entrega código, visualizaciones y conclusiones, para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Python con librerías de visión (TorchVision/Keras)
- Notebook interactivo
- Material de apoyo proporcionado por su docente

Unidad II. Redes convolucionales y visión por computadora

Práctica 4: Convolución, pooling y visualización de mapas de características

Duración

04 horas

Procedimiento

1. Atiende las indicaciones docentes sobre operaciones convolucionales.
2. Implementa manualmente capas de convolución y pooling en una red simple.
3. Visualiza filtros y mapas de activación en distintas capas.
4. Analiza cómo cambia la representación conforme aumenta la profundidad.
5. Documenta los resultados y reflexiona sobre su relevancia en visión por computadora.
6. Entrega código, capturas y conclusiones, para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Python con PyTorch/TensorFlow
- Notebook interactivo
- Conjunto de datos proporcionado por su docente

Práctica 5: Transfer learning y fine-tuning con modelos preentrenados

Duración

04 horas

Procedimiento

1. Atiende las orientaciones docentes sobre modelos preentrenados.
2. Carga un modelo ResNet o MobileNet preentrenado en ImageNet.
3. Congela capas y entrena solo el clasificador final.
4. Después, realiza fine-tuning parcial y compara resultados.

5. Evalúa desempeño mediante métricas apropiadas (accuracy, pérdida).
6. Entrega análisis comparativo, gráficas y conclusiones, para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Python con PyTorch/TensorFlow
- Notebook interactivo
- Dataset de imágenes

Práctica 6: Aplicaciones de CNN: detección y segmentación mediante modelos preentrenados

Duración

04 horas

Procedimiento

1. Atiende la explicación docente sobre detección y segmentación.
2. Ejecuta inferencia con un modelo de detección preentrenado (YOLOv8, SSD o equivalente).
3. Ejecuta segmentación con un modelo preentrenado (Mask R-CNN o DeepLabv3).
4. Analiza precisión, errores y limitaciones del modelo en distintos escenarios.
5. Entrega ejemplos visuales, análisis y conclusiones, para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Python con librerías de visión
- Notebook interactivo
- Modelos preentrenados

Unidad III. Transformers y modelos de lenguaje

Práctica 7: Implementación del mecanismo de atención

Duración

04 horas

Procedimiento

1. Atiende la explicación docente sobre el mecanismo de atención.
2. Implementa atención escalada dot-product con tensores simples.
3. Visualiza mapas de atención en secuencias cortas.
4. Analiza cómo cambia la atención según la entrada.
5. Entrega código, visualizaciones y reflexiones, para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Python con PyTorch/TensorFlow
- Notebook interactivo

Práctica 8: Tokenización moderna y uso de modelos BERT/GPT pequeños

Duración

04 horas

Procedimiento

1. Atiende las indicaciones docentes sobre tokenización y embeddings.
2. Tokeniza texto con BPE o WordPiece.
3. Ejecuta tareas básicas con DistilBERT o GPT-2 pequeño (clasificación, completado).
4. Evalúa precisión y analiza las representaciones generadas.
5. Entrega código, experimentos y conclusiones, para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Python (HuggingFace Transformers)
- Notebook interactivo
- Conjunto de datos de texto

Práctica 9: Evaluación de modelos de lenguaje e identificación de sesgos y alucinaciones

Duración

05 horas

Procedimiento

1. Atiende la explicación docente sobre métricas de modelos de lenguaje y riesgos asociados.
2. Evalúa tareas de resumen, QA o clasificación usando un modelo preentrenado.
3. Registra métricas relevantes (accuracy, BLEU, ROUGE o perplexity).
4. Identifica ejemplos de sesgo, inconsistencias o alucinaciones.
5. Reflexiona sobre sus posibles implicaciones éticas.
6. Entrega un notebook con evidencias, análisis crítico y conclusiones, para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Python con Transformers
- Notebook interactivo
- Documentos base sobre ética en IA

Práctica 10: Diseño y evaluación de prompts (Prompt Engineering aplicado)

Duración

05 horas

Procedimiento

1. Atiende las orientaciones docentes sobre estrategias de prompting.
2. Diseña prompts zero-shot, few-shot y chain-of-thought para una tarea asignada.
3. Compara resultados con distintas estructuras de prompt.
4. Evalúa claridad, precisión y posibles fallas en las respuestas generadas.
5. Entrega un reporte con prompts utilizados, resultados y conclusiones críticas, para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Acceso a un modelo generativo (OpenAI, HuggingFace u otro)
- Notebook interactivo o plataforma web
- Material técnico proporcionado por su docente

Unidad IV: Modelos generativos modernos

Práctica 11: Implementación de un autoencoder y análisis del espacio latente

Duración

03 horas

Procedimiento

1. Atiende la explicación docente sobre autoencoders y su uso en compresión.
2. Implementa un autoencoder simple para imágenes.
3. Visualiza reconstrucciones y compara la calidad obtenida.
4. Analiza el espacio latente mediante proyecciones o interpolaciones.
5. Entrega código, resultados visuales y conclusiones, para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Python con PyTorch/TensorFlow
- Notebook interactivo
- Conjunto de datos visual simple

Práctica 12: Demostración aplicada de GAN y modelos de difusión preentrenados

Duración

03 horas

Procedimiento

1. Atiende las orientaciones docentes sobre GAN y modelos de difusión.
2. Ejecuta inferencia con una GAN preentrenada (DCGAN o equivalente).
3. Genera imágenes usando un modelo de difusión preentrenado (Stable Diffusion).
4. Analiza la calidad, variabilidad y limitaciones de las imágenes generadas.
5. Reflexiona sobre riesgos y responsabilidades al utilizar modelos generativos.
6. Entrega un reporte con ejemplos visuales y conclusiones, para revisión y retroalimentación docente.

Recursos de apoyo

- Computadora
- Python con librerías generativas
- Acceso a un modelo de difusión
- Notebook interactivo
- Documentos de referencia ética

X. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer de forma clara y formal:

- La metodología de trabajo.
- Criterios de evaluación y rúbricas.
- Reglas sobre la calidad y entrega de trabajos académicos.
- Derechos y obligaciones del o la docente y del estudiantado

Esto debe quedar registrado y comunicado por medios institucionales (como Blackboard o correo institucional).

Estrategia de enseñanza

- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Técnica expositiva
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Cuadros comparativos

- Infografías

XI. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

- Evaluaciones parciales	30%
- Prácticas laboratorio	30%
- Reporte técnico	30%
- Presentación del reporte técnico	10%
Total	100%

XII. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Bhuvaneswari, B. (2021). *Artificial Intelligence*. Astral International Pvt Ltd.

<https://libcon.rec.uabc.mx:2938/linkprocessor/plink?id=52bc4529-dabb-3336-a44d-53a5dbd29a41>

Centro de Investigación para el Aprendizaje Digital. (2024). *Recomendaciones para la integración efectiva de la inteligencia artificial generativa en la práctica docente de educación superior*. Universidad Autónoma de Baja California.

https://ciad.mxl.uabc.mx/wp-content/uploads/2024/01/IA_Practica_Docente_2024-01-24.pdf

Chollet, F. (2021). *Deep learning with Python*. Manning.

Ekman, M. (2021). *Learning deep learning: Theory and practice of neural networks, computer vision, natural language processing, and transformers using TensorFlow*. Addison-Wesley Professional.

Geron, A. (2022). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow* (3rd ed.). O'Reilly Media.

Goodfellow, I. (2016). *Deep learning-ian goodfellow, yoshua bengio, aaron courville-google books*. [Clásica]

Moroney, L. (2020). *AI and Machine Learning for Coders: A Programmer's Guide to Artificial Intelligence*. O'Reilly Media.

Neapolitan, R. E., & Jiang, X. (2018). *Artificial intelligence: With an introduction to machine learning*. CRC press. [Clásica]

Russell, S., & Norvig, P. (2021). *Artificial Intelligence: A Modern Approach* (4a ed.). Pearson.

Recursos de acceso libre o complementarios externos

ACM SIGAI. (s.f.). *ACM SIGAI – Special Interest Group on Artificial Intelligence*.

<https://sigai.acm.org/main/>

European Union (2023). *EU Artificial Intelligence Act (AI Act)*. <https://artificialintelligenceact.eu/>

Gemma Team, Kamath, A., Ferret, J., Pathak, S., Vieillard, N., Merhej, R., Perrin, S., Matejovicova, T., Ramé, A., Rivière, M., Rouillard, L., Mesnard, T., Cideron, G., Grill, J.-B., Ramos, S., Yvinec, E., Casbon, M., Pot, E., Penchev, I., Liu, G. & Hussenot, L. (2025). *Gemma 3 technical report*. arXiv. <https://arxiv.org/abs/2503.19786>

Goyal, M., & Mahmoud, Q. H. (2024). A systematic review of synthetic data generation techniques using generative AI. *Electronics*, 13(17), 3509.
<https://doi.org/10.3390/electronics13173509>

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., Yang, A., Fan, A., Goyal, A., Hartshorn, A., Yang, A., Mitra, A., Sravankumar, A., Korenev, A., Hinsvark, A., ... Ma, Z. (2024). *The Llama 3 herd of models*. arXiv. <https://arxiv.org/abs/2407.21783>

He, R., & al. (2023). A survey on GANs for computer vision: Recent research, analysis and taxonomy. *Computer Science Review*, 48, 100553.
<https://doi.org/10.1016/j.cosrev.2023.100553>

Ibrahim, M., Al Khalil, Y., Amirrajab, S., Sun, C., Breeuwer, M., Pluim, J., Elen, B., Ertaylan, G., & Dumontier, M. (2024). *Generative AI for synthetic data across multiple medical modalities: A systematic review of recent developments and challenges*. arXiv.
<https://arxiv.org/abs/2407.00116>

Koo, H., & Kim, T. E. (2023). *A comprehensive survey on generative diffusion models for structured data*. arXiv. <https://arxiv.org/abs/2306.04139>

Murphy, K. P. (2012). *Machine learning: A probabilistic perspective*. MIT Press.
<https://www.cs.ubc.ca/~murphyk/>

OpenAI, Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., Avila, R., Babuschkin, I., Balaji, S., Balcom, V., Baltescu, P., Bao, H., Bavarian, M., Belgum, J., Bello, I., ... Zoph, B. (2024). *GPT-4 technical report*. arXiv. <https://arxiv.org/abs/2303.08774>

Shalev-Shwartz, S., y Ben-David, S. (2014). *Understanding machine learning: From theory to algorithms*. Cambridge University Press. [clásica]
<https://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/>

- Suh, N., & Cheng, G. (2025). A survey on statistical theory of deep learning: Approximation, training dynamics, and generative models. *Annual Review of Statistics and Its Application*, 12, 177–207. <https://doi.org/10.1146/annurev-statistics-040522-013920>
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... y Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30. https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf
- Yazdani, S., Singh, A., & Saxena, N. (2025). Generative AI in depth: A survey of recent advances, model variants, and real-world applications. *Journal of Big Data*, 12, 230. <https://doi.org/10.1186/s40537-025-01247-x>
- Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2023). *Dive into deep learning*. Cambridge University Press. <https://d2l.ai/index.html>

XIII. PERFIL DOCENTE

Licenciatura en Ciencias Computacionales, Ingeniería en Computación, Matemáticas, Física o áreas afines, preferentemente con posgrado en inteligencia artificial, ciencia de datos o campos relacionados. Es indispensable la experiencia en docencia universitaria y el dominio de Python y bibliotecas como PyTorch, TensorFlow y HuggingFace Transformers para desarrollar y evaluar modelos avanzados. Debe integrar tecnologías emergentes de manera ética, explicar conceptos con precisión, diseñar actividades teóricas y prácticas, y promover la disciplina, la honestidad académica, el trabajo colaborativo y la actualización continua.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Interacción Humano Computadora
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 03 Horas Laboratorio: 00 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Terminal
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	María Victoria Meza Kubo Alberto Leopoldo Morán y Solares
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Introducir al estudiantado en el diseño, implementación y evaluación de sistemas de cómputo interactivos para ser utilizados por humanos, considerando las principales características tanto del ser humano como de la tecnología. Permite sustentar sus diseños y evaluarlos a lo largo de su desarrollo de tal forma que se ajuste de manera transparente a las capacidades de la población a la que está dirigida para que su uso sea fácil, útil y satisfactorio. La unidad de aprendizaje es de carácter obligatorio, se encuentra en la etapa terminal y pertenece al área de conocimiento de Interacción Hombre Máquina.

COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Desarrollar aplicaciones interactivas, por medio de las teorías y técnicas de la interacción humano-computadora, para generar aplicaciones con una alta usabilidad para el usuario objetivo, con responsabilidad y actitud creativa.

III. EVIDENCIA DE APRENDIZAJE

Desarrolla una aplicación interactiva funcional acompañada de manuales de usuario y del sistema, así como el análisis de los resultados de una evaluación de usabilidad realizada a la aplicación con usuarios reales.

IV. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social.

V. EJE TRANSVERSAL AL QUE CONTRIBUYE

Inclusión; Pensamiento Crítico.

VI. DESARROLLO POR UNIDADES

Unidad I. Fundamentos de interacción humano-computadora

Competencia

Analizar los fundamentos de la interacción humano-computadora, a través de la aplicación de modelos de percepción y memoria junto con los paradigmas de interacción, para comprender las bases del diseño de interfaces, con actitud crítica y compromiso con la calidad.

Contenido

- 1.1. Evolución histórica y paradigmas de interacción.
- 1.2. El ser humano: percepción, memoria, carga cognitiva.
- 1.3. La computadora: dispositivos de entrada/salida, arquitecturas de interacción.
- 1.4. Concepto de usabilidad y experiencia de usuario (UX).

Duración

04 horas

Unidad II. Procesos y metodologías de diseño centrado en el usuario

Competencia

Diseñar interfaces interactivas mediante el uso de metodologías de diseño centrado en el usuario, heurísticas de usabilidad y prototipado, para generar soluciones acordes a las necesidades de la población objetivo, con creatividad y responsabilidad.

Contenido

- 2.1. Diseño centrado en el usuario (UCD) y ciclo de vida de la interacción humano-computadora (IHC).
- 2.2. Principios y heurísticas de Nielsen y Shneiderman.
- 2.3. Modelos cognitivos aplicados al diseño.
- 2.4. Prototipado de interfaces (baja, media y alta fidelidad).

Duración

06 horas

Unidad III. Evaluación de interfaces y usabilidad

Competencia

Evaluar interfaces digitales, con la aplicación de técnicas de evaluación heurística, pruebas con usuarios y análisis de datos, para obtener retroalimentación válida que permita mejorar la experiencia de usuario, con objetividad, ética y sentido crítico.

Contenido

- 3.1. Métodos de evaluación: heurística, pruebas con usuarios, inspecciones.
- 3.2. Obtención y análisis de datos (cuantitativos y cualitativos).
- 3.3. Herramientas digitales de evaluación.
- 3.4. Interpretación y presentación de resultados.

Duración

06 horas

Unidad IV. Inclusión, accesibilidad y diseño universal

Competencia

Diseñar propuestas inclusivas de interacción que cumplan principios de diseño universal, lineamientos de accesibilidad y tecnologías multimodales, para atender a poblaciones diversas, con empatía, respeto y compromiso social.

Contenido

- 4.1. Principios de diseño universal.
- 4.2. Accesibilidad.
- 4.3. Interacción multimodal: voz, gestos, dispositivos hápticos.
- 4.4 Interfaces adaptadas a poblaciones especiales.

Duración

06 horas

Unidad V. Tendencias avanzadas en IHC

Competencia

Evaluar tecnologías emergentes en interacción humano-computadora, por medio del uso de interfaces naturales, realidad extendida y sistemas inteligentes, para generar aplicaciones innovadoras en distintos contextos, con iniciativa, creatividad y visión ética.

Contenido

- 5.1. Interfaces naturales (NUI): AR, VR y MR.
- 5.2. Interacción con inteligencia artificial y agentes conversacionales.
- 5.3. Interfaces tangibles, vestibles y ambientales.
- 5.3. Retos éticos en el diseño de interfaces.

Duración

06 horas

Unidad VI. Proyecto integrador

Competencia

Desarrollar un proyecto de aplicación interactiva que aplique técnicas de levantamiento de requerimientos, prototipado iterativo y evaluación con usuarios, para generar aplicaciones con una alta usabilidad, con colaboración, responsabilidad y compromiso con la calidad.

Contenido

- 6.1. Definición de necesidades del usuario.
- 6.2. Diseño y prototipado de la interfaz.
- 6.3. Evaluación con usuarios reales.
- 6.4. Presentación de resultados.

Duración

04 horas

VII. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

Unidad I. Fundamentos de interacción humano-computadora

Práctica 1: Reconociendo las capacidades humanas en la interacción

Duración

03 horas

Procedimiento

1. Atiende a las explicaciones de su docente sobre los límites de la memoria de corto plazo y la percepción visual.
2. Se reúne en equipo de trabajo.
3. En equipo, recibe una lista de palabras o imágenes y dispone de 30 segundos para memorizarlas.
4. Después de un intervalo de 1 minuto, escribe todo lo que recuerde.
5. Se repite la actividad con listas de distinta longitud.
6. En equipo, compara los resultados y discute la relación con el diseño de menús, botones o mensajes en interfaces.
7. En equipo, redacta una conclusión breve con ejemplos de buenas prácticas de diseño basadas en los hallazgos.
8. Presenta su conclusión al resto de la clase.
9. Recibe retroalimentación docente y grupal.

Recursos de apoyo

- Cronómetro
- Listas de palabras/imágenes
- Hojas
- Lápices

Práctica 2: Explorando paradigmas de interacción

Duración

03 horas

Procedimiento

1. Atiende a la presentación de los ejemplos de paradigmas (WIMP, NUI, AR/VR, dispositivos móviles).
2. Se reúne en equipo de trabajo.
3. En equipo, investiga un caso real de aplicación de un paradigma (ej. Photoshop con WIMP, Oculus en VR).
4. En equipo, elabora una tabla comparativa con ventajas, desventajas y contexto de uso.
5. Participa en una plenaria en la que cada equipo presenta su análisis en máximo 5 minutos.
6. De manera grupal, elabora de manera conjunta un mapa conceptual de paradigmas en el pizarrón.
7. Recibe retroalimentación de su docente.

Recursos de apoyo

- Computadoras
- Acceso a internet
- Plumones
- Pizarrón

Unidad II. Procesos y metodologías de diseño centrado en el usuario

Práctica 3: Análisis de necesidades del usuario

Duración

03 horas

Procedimiento

1. Atiende a la presentación sobre el concepto de perfiles de usuario y escenarios de uso.
2. Se reúne en equipo de trabajo.
3. En equipo, recibe un problema (ej. app de transporte, sistema de biblioteca, juego educativo).
4. En equipo, simula entrevistas con usuarios (compañeros del grupo).
5. Registra necesidades, frustraciones y expectativas.
6. Con esa información, en equipo, elabora un perfil de usuario (edad, motivaciones, habilidades digitales, etc.).
7. Comparte un perfil en clase y recibe retroalimentación grupal.

Recursos de apoyo

- Plantillas de entrevista
- Hojas
- Lápices
- Grabadora/celular

Práctica 4: Prototipado de baja fidelidad

Duración

03 horas

Procedimiento

1. Atiende a las explicaciones de su docente sobre los tipos de prototipos (baja, media y alta fidelidad).
2. Se reúne en equipo de trabajo.
3. En equipo, diseña en papel la interfaz de un sistema sencillo (ej. app para pedir comida).
4. Intercambia prototipos con otro equipo que actúa como usuario y da retroalimentación escrita.
5. Ajusta su prototipo según la retroalimentación recibida.
6. Presenta el prototipo en clase, explicando los cambios realizados.
7. Recibe retroalimentación docente y grupal.

Recursos de apoyo

- Papel
- Cartulinas
- Marcadores
- Tijeras
- Cinta adhesiva

Unidad III. Evaluación de interfaces y usabilidad

Práctica 5: Evaluación heurística

Duración

03 horas

Procedimiento

1. Atiende al repaso de su docente sobre las 10 heurísticas de Nielsen.
2. Se reúne en equipo de trabajo.
3. En equipo, selecciona un sitio web o aplicación.
4. Evalúa con su equipo, la interfaz usando una lista de verificación (checklist).
5. Documenta problemas encontrados y los clasifica en menores, mayores o críticos.
6. En equipo, elabora un informe corto con ejemplos visuales (capturas de pantalla).
7. Presenta los hallazgos al resto del grupo y recibe retroalimentación.

Recursos de apoyo

- Computadora con acceso a internet
- Plantillas de checklist

Práctica 6: Pruebas con usuarios

Duración

03 horas

Procedimiento

1. Atiende a la explicación de su docente sobre las fases de una prueba de usabilidad: planeación, ejecución y análisis.
2. Se reúne en equipo de trabajo.
3. En equipo, diseña 3 tareas sencillas que un usuario debe realizar en una aplicación real (ej. crear cuenta en Gmail).
4. Simula pruebas con integrantes de su equipo como usuario.
5. Registra el tiempo de ejecución, errores y comentarios verbales.
6. Analiza la información y elabora 3 recomendaciones de mejora.
7. En equipo presenta su experiencia en plenaria.
8. Recibe retroalimentación docente y grupal.

Recursos de apoyo

- Computadora
- Acceso a internet
- Cronómetro
- Hojas de registro

Unidad IV. Inclusión, accesibilidad y diseño universal

Práctica 7: Simulación de accesibilidad

Duración

03 horas

Procedimiento

1. Atiende la explicación de su docente sobre los principios básicos de accesibilidad y el estándar WCAG.
2. Se reúne en equipo de trabajo.
3. En equipo, selecciona un sitio web y lo revisa con herramientas de validación de accesibilidad (ej. WAVE).
4. Simula condiciones de discapacidad: navegar con los ojos vendados (uso de lector de pantalla), con guantes (discapacidad motriz), o con subtítulos desactivados (discapacidad auditiva).
5. Documenta las barreras encontradas.
6. Propone mejoras en un listado priorizado.
7. Presenta con su equipo de trabajo sus propuestas al resto del grupo.
8. Recibe retroalimentación grupal.

Recursos de apoyo

- Computadora
- Extensión WAVE (<https://wave.webaim.org>)
- Vendas
- Guantes

Práctica 8: Diseño inclusivo de prototipos

Duración

03 horas

Procedimiento

1. Atiende la presentación de su docente sobre ejemplos de diseño inclusivo (botones grandes, contraste, navegación por teclado).
2. Se reúne en equipo de trabajo.
3. En equipo, rediseña un prototipo creado en la práctica 4, ahora considerando accesibilidad.
4. Intercambia prototipos con otro equipo para obtener retroalimentación.
5. Ajusta el diseño final y lo presenta con su equipo en plenaria.
6. Recibe retroalimentación docente y grupal.

Recursos de apoyo

- Papel
- Figma u otra herramienta de prototipado

Unidad V. Tendencias avanzadas en IHC

Práctica 9: Experiencias con interfaces emergentes

Duración

03 horas

Procedimiento

1. Atiende a la introducción de su docente sobre ejemplos de interfaces emergentes: VR, AR, asistentes de voz.
2. Se reúne en equipo de trabajo.
3. Prueba una aplicación (ej. AR en un smartphone, chatbot de IA).
4. Documenta la experiencia en una tabla (interacciones fluidas, problemas, emociones percibidas).
5. En equipo, elabora un breve reporte sobre potenciales aplicaciones prácticas.
6. Entrega y presenta su reporte y recibe retroalimentación docente y grupal.

Recursos de apoyo

- Smartphone con AR
- Dispositivo VR

- Asistentes virtuales (Siri, Alexa, ChatGPT)

Práctica 10: Diseño de un escenario de interacción futura

Duración

03 horas

Procedimiento

1. Atiende al reto de su docente: diseñar un escenario de interacción para dentro de 10 años.
2. Se reúne en equipo de trabajo.
3. En equipo, crea un storyboard con personajes, contexto y uso de la tecnología.
4. Discute los beneficios, riesgos y dilemas éticos de su escenario.
5. Presenta el storyboard en plenaria y reciben retroalimentación.

Recursos de apoyo

- Papel
- Marcadores
- Software de storyboard

Unidad VI. Proyecto integrador

Práctica 11: Definición del proyecto y requerimientos

Duración

03 horas

Procedimiento

1. Atiende las explicaciones de su docente sobre los pasos para definir un proyecto de IHC.
2. Se reúne en equipo de trabajo.
3. En equipo, elige un problema real o simulado.
4. Elabora una lista de requerimientos con base en usuarios potenciales.
5. Presenta el proyecto en una ficha técnica breve.
6. Recibe retroalimentación docente y grupal.

Recursos de apoyo

- Plantillas de levantamiento de requerimientos
- Papel
- Lápiz

Práctica 12: Prototipo inicial y retroalimentación

Duración

06 horas

Procedimiento

1. Se reúne en equipo de trabajo.
2. En equipo, diseña un prototipo de baja o media fidelidad.
3. Intercambia prototipos con otro equipo para probarlos como usuarios.
4. Registra comentarios, errores y sugerencias.
5. Ajusta el prototipo con base en la retroalimentación.

Recursos de apoyo

- Figma
- Balsamiq
- Papel
- Cartulinas

Práctica 13: Prototipo avanzado y evaluación iterativa

Duración

06 horas

Procedimiento

1. Con su equipo, mejora el prototipo a un nivel avanzado (media o alta fidelidad).
2. Ejecuta pruebas con usuarios reales o simulados (mínimo 2).
3. Documenta resultados en una tabla (tiempos, errores, satisfacción).
4. Ajusta el prototipo final.

Recursos de apoyo

- Herramienta de prototipado
- Computadora
- Cámara/celular

Práctica 14: Presentación de resultados

Duración

03 horas

Procedimiento

1. Con su equipo, prepara una presentación final de su proyecto.
2. Muestra el prototipo y explica el proceso de diseño, evaluación y ajustes realizados.
3. El grupo y su docente hacen preguntas y comentarios.
4. Se genera un cierre colectivo destacando aprendizajes y retos.

Recursos de apoyo

- Proyector
- Computadora
- Presentación digital
- Prototipo final

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Aprendizaje basado en preguntas
- Técnica expositiva
- Discusión guiada

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Mapas conceptuales
- Cuadros comparativos
- Infografías
- Método de proyectos

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario, cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	30%
– Prácticas taller	30%
– Proyecto	40%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Albert, B., & Tullis, T. (2022). *Measuring the user experience: Collecting, analyzing, and presenting UX metrics*. Morgan Kaufmann.
- Dumas, J. S., & Loring, B. A. (2008). *Moderating usability tests: Principles and practices for interacting*. Elsevier. [classic]
- Kim, G. J. (2015). *Human-computer interaction: fundamentals and practice*. CRC press. [classic]
- Sharp, H., Rogers, Y., & Preece, J. (2023). *Interaction design: beyond human-computer interaction* (6th Ed.) Wiley.
- Lazar, J., Feng, J. H., & Hochheiser, H. (2017). *Research methods in human-computer interaction*. Morgan Kaufmann. [classic]
- Nielsen, J. (1994). *Usability engineering*. Morgan Kaufmann. [classic]
- Rubin, J. (1995). Handbook of usability testing. *Technical Communication*, 42(2), 361. [clasica].
- Shneiderman, B. (2016). *Designing the user interface: strategies for effective human-computer interaction* (6th ed.) Pearson Education India. [classic]

Recursos de acceso libre o complementarios externos

- Chowdhury, A. (2022). *A pocket guide to HCI and UX design*. Partridge Publishing.
- Benyon, D. (2019). *Designing user experience*. Pearson UK.
- Boy, G. A. (Ed.). (2017). *The handbook of human-machine interaction: a human-centered design approach*. CRC Press.
- Fu, Z., Barbara, A., & Scupelli, P. (Eds.). (2025). *Digital Futures in Human-computer Interaction: Design Thinking for Digital Transformation*. CRC Press.
- Norman Donald, A. (2013). *The design of everyday things*. MIT Press. [classic].
- Sarmiento, A. (Ed.). (2005). *Issues of human computer interaction*. IGI Global. [classic]
- Digital.gov. <https://digital.gov/topics/usability/>
- HCI Bibliography : Human-Computer Interaction Resources. <http://hcibib.org/>
- Interaction Design Foundation. <https://www.interaction-design.org>
- Nielsen Norman Group. <https://www.nngroup.com/>
- Smashing Magazine. <https://www.smashingmagazine.com/>
- Stanford HCI Group. <http://hci.stanford.edu/>
- W3C. Web Accesibility Initiative WAI. <https://www.w3.org/WAI/standards-guidelines/wcag/>
- WAVE. <https://wave.webaim.org/>
- WebAIM. <https://webaim.org/resources/>

XII. PERFIL DOCENTE

Profesionista en computación o áreas afines con al menos 2 años de experiencia en la docencia, tener conocimientos en interacción humano-computadora. Ser una persona creativa, tener un eficiente manejo grupal, fomentar el pensamiento crítico y creativo en sus estudiantes.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Simulación
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 03 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Terminal
Carácter:	Obligatoria
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Alma Rocío Cabazos Marín Jonathan Verdugo Olachea
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Brindar los fundamentos del modelo de simulación estocástica para usar y desarrollar paquetes de simulación que permitan modelar cursos de acción de los sistemas y permitan decidir cuál es el más adecuado.

La unidad de aprendizaje de simulación se encuentra ubicada en la etapa terminal del plan de estudios, es de carácter obligatorio y pertenece al área de conocimiento de Matemáticas. Es una de las unidades integradoras donde las y los estudiantes emplean los conocimientos adquiridos con anterioridad, principalmente aquellos relacionados al conocimiento científico y matemático.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Modelar sistemas de simulación computacional que representen el comportamiento de sistemas reales o hipotéticos, utilizando herramientas matemáticas, estadísticas y de programación, para analizar, predecir y optimizar procesos en diversos contextos científicos, tecnológicos y sociales; integrando análisis de datos y validación de resultados, con actitud crítica, ética y colaborativa.

IV. EVIDENCIA DE APRENDIZAJE

Desarrollo de un portafolio de prácticas de laboratorio y tareas.

Elaboración de un prototipo funcional de simulación de un sistema real o ficticio, acompañado de la documentación técnica correspondiente.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia; Vanguardia.

VII. DESARROLLO POR UNIDADES

Unidad I. Conceptos básicos de probabilidad aplicados a la simulación

Competencia

Analizar los principios teóricos de la simulación, mediante el uso de herramientas y conceptos básicos de probabilidad y generación de números pseudoaleatorios, para representar modelos simples, con una actitud creativa y de colaboración.

Contenido

- 1.1. Principios de Simulación
 - 1.1.1 Definición y propósito
 - 1.1.2 Historia de la simulación
 - 1.2.3 Aplicaciones de la simulación
- 1.2 Conceptos básicos de la teoría de probabilidad que se emplean en los modelos de simulación
- 1.3. Generación de números pseudoaleatorios con distribución homogénea
 - 1.3.1 Generadores Lineales Congruenciales
 - 1.3.2 Método de Cuadrados Medios
 - 1.3.3 Generadores de Tausworthe
 - 1.3.4 Generadores de Fibonacci generalizado
- 1.4. Generación de números pseudoaleatorios con distribución específica
 - 1.4.1 Método de la Transformada Inversa
 - 1.4.2 Método de Rechazo
- 1.5. Pruebas estadísticas sobre números aleatorios
 - 1.5.1 Criterio de autocorrelación
 - 1.5.2 Prueba de los promedios
 - 1.5.3 Prueba de la varianza
 - 1.5.4 Prueba de Series
 - 1.5.5 Prueba de Autocorrelación
 - 1.5.6 Prueba de Póker
 - 1.5.7 Prueba de bondad de ajuste Chi cuadrada
 - 1.5.8 Prueba de bondad de ajuste Kolmogorov-Smirnov)
- 1.6. Base de la Estrategia Montecarlo

Duración

10 horas

Unidad II. Componentes y organización de un modelo discreto

Competencia

Analizar eventos discretos, mediante pseudocódigos y diagramas de flujo para el análisis de sistemas de líneas de espera que permitan la simulación de fenómenos de espera y de propagación, con actitud propositiva y de colaboración.

Contenido

- 2.1. Fundamentos y conceptos básicos de simulación
 - 2.1.1 Sistemas y modelos
 - 2.1.2 tipos de simulación (discreta y continua)
 - 2.1.3 Componentes y etapas
- 2.2. Modelos y sistemas de simulación:
 - 2.2.1 colas, SIR, tráfico, producción
- 2.3. Modelado de llegadas y tiempos de servicio
- 2.4 Construcción y ejecución de simulaciones discretas:
 - 2.4.1 Pseudocódigos
 - 2.4.2 Manejo del reloj de simulación
 - 2.4.3 Listas de eventos
 - 2.4.4 Registro de métricas o datos
 - 2.4.5 Aplicaciones de simulación de fenómenos de espera y propagación

Duración

04 horas

Unidad III. Lenguaje de simulación

Competencia

Comparar modelos de simulación de fenómenos simples, mediante el uso de lenguajes específicos para simulación, con actitud creativa, razonamiento lógico y abstracción.

Contenido

- 3.1. Lenguajes de propósito general
- 3.2 Lenguajes de Simulación (SimPy-Python, GPSS, GASP, SIMSCRIPT, SLAM)
- 3.3. Plataformas de simulación (MATLAB & Simulink, AnyLogic, PhET)
- 3.4. Simulación y diseño basado en modelos

Duración

06 horas

Unidad IV. Simulación de un sistema

Competencia

Elaborar simulaciones computacionales de sistemas multivariantes, mediante la selección y optimización de parámetros, a través de la comparación de modelos, el análisis y la validación de resultados, con una actitud crítica, propositiva y de colaboración.

Contenido

- 4.1. Modelos avanzados: construcción de sistemas con múltiples variables y dependencias
- 4.2. Métodos de validación y verificación
- 4.3. Análisis de resultados mediante estadística descriptiva e inferencial.
- 4.4. Estrategias de mejora en la simulación (reducción de costos, eficiencia energética, flujo óptimo de recursos)
- 4.5. Documentación de resultados con evidencia numérica y visualizaciones (gráficas, tablas, animaciones)

Duración

04 horas

Unidad V. Proyectos

Competencia

Desarrollar un prototipo de simulación computacional para la solución de un problema real, mediante la integración de los conceptos vistos en el curso, con actitud ética, responsable y de trabajo colaborativo.

Contenido

- 5.1. Proyectos de simulación aplicables al mundo real
- 5.2. Propuesta de proyecto de simulación
- 5.3. Modelado
- 5.5. Recolección de datos
- 5.6. Codificación, integración y pruebas
- 5.7. Análisis y validación de resultados
- 5.7. Documentación y exposición del proyecto

Duración

08 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Conceptos básicos de probabilidad

Práctica 1. Conceptos básicos de probabilidad y aleatoriedad, 1

Duración

2 horas

Procedimiento

1. Atiende las indicaciones de su docente para la realización de la práctica
2. Revisa los conceptos de aleatoriedad y probabilidad, como base teórica para la simulación de eventos
3. Realiza un experimento físico con 30 lanzamientos de una moneda, y 60 lanzamientos de un dado, registrando los resultados obtenidos
4. Desarrolla un programa computacional que permita simular el lanzamiento de una moneda y el lanzamiento de un dado, mediante un generador de números pseudo aleatorio con distribución homogénea
5. Registra los resultados del número de veces que se obtuvo {águila, sello} en 30, 300, y 3000 lanzamientos de la moneda; y el número de veces que se obtuvo {1, 2, 3, 4, 5, 6} en 60, 600 y 6000 lanzamientos del dado
6. Compara los resultados de la simulación con los del experimento físico y analiza las diferencias observadas
7. Elabora y presenta los resultados en un reporte. El reporte debe incluir gráficas que muestran el porcentaje de ocurrencia de cada valor, para cada uno de los experimentos; y una reflexión sobre la convergencia de las frecuencias relativas a las probabilidades teóricas. Concluye destacando cómo la simulación computacional permite reproducir y analizar fenómenos aleatorios

Recursos de apoyo

- Computadora
- Internet
- Plataforma de Desarrollo Integrado, IDE
- Editor de texto
- Formato de práctica
- Moneda
- Dado

Práctica 2. Conceptos básicos de probabilidad y aleatoriedad, 2

Duración

02 horas

Procedimiento

1. Atiende las indicaciones de su docente para la realización de la práctica

2. Revisa los conceptos de aleatoriedad y probabilidad, como base teórica para la simulación de eventos
3. Desarrolla un programa computacional que permita simular el lanzamiento de dos dados, mediante un generador pseudo aleatorio con distribución uniforme, empleando un generador independiente para cada dado
4. Registra los resultados del número de veces que la suma de los dos dados obtenidos fue $\{2, 3, 4, 5, \dots, 12\}$, prueba para 6, 60, 600, 6000, ..., 12,000 lanzamientos
5. Elabora y presenta los resultados en un reporte. El reporte debe incluir gráficas que muestran el porcentaje de ocurrencia de cada valor, y una reflexión sobre la convergencia de las frecuencias relativas a las probabilidades teóricas al aumentar el número de lanzamientos. Concluye destacando cómo la simulación computacional permite reproducir y analizar fenómenos aleatorios

Recursos de apoyo

- Computadora
- Internet
- Plataforma de Desarrollo Integrado, IDE
- Editor de texto
- Formato de práctica

Práctica 3. Generadores de números pseudoaleatorios

Duración

02 horas

Procedimiento

1. Atiende las indicaciones de su docente para la realización de la práctica
2. Investiga los principios de funcionamiento de distintos algoritmos para generar números pseudoaleatorios (Generadores lineales congruenciales, Método de los cuadrados medios, Generadores de Tausworthe, Generadores de Fibonacci generalizado)
3. Selecciona e implementa un algoritmo generador de números pseudoaleatorios mediante un programa computacional que permita generar una secuencia de números pseudoaleatorios
4. Analiza los resultados obtenidos y verifica si existe algún patrón en el cual se repiten los valores pseudoaleatorios, identificando el período y grado de aleatoriedad
5. Elabora y presenta los resultados en un reporte, en una tabla o gráfica de resultados muestra el patrón generado y explica cómo se puede reducir o evitar los patrones en la generación pseudoaleatoria

Recursos de apoyo

- Computadora
- Internet
- Plataforma de Desarrollo Integrado, IDE
- Editor de texto
- Formato de práctica

Práctica 4. Pruebas estadísticas para los números pseudoaleatorios

Duración

02 horas

Procedimiento

1. Atiende las indicaciones de su docente para la realización de la práctica
2. Revisa los fundamentos de las pruebas estadísticas utilizadas para evaluar la calidad de los números pseudoaleatorios
3. Aplica pruebas estadísticas a una secuencia de números pseudoaleatorios generados
4. Analiza los resultados obtenidos y determina si la secuencia cumple con las propiedades esperadas de uniformidad e independencia de números pseudoaleatorios
5. Elabora y presenta un reporte que incluya los procedimientos aplicados, las gráficas de la secuencia de los números pseudoaleatorios, los resultados numéricos y una conclusión sobre la calidad del generador pseudoaleatorio evaluado

Recursos de apoyo

- Computadora
- Internet
- Plataforma de Desarrollo Integrado, IDE
- Editor de texto
- Formato de práctica

Práctica 5. Generación de variables aleatorias no uniformes

Duración

02 horas

Procedimiento

1. Atiende las indicaciones de su docente para la realización de la práctica
2. Revisa los fundamentos teóricos de la generación de variables aleatorias no uniformes y sus principales aplicaciones en simulación
3. Implementa distintos métodos de generación de variables aleatorias no uniformes, y genera muestras de distintas distribuciones no uniformes (por ejemplo, exponencial, normal o triangular) aplicando ambos métodos, y comparar los resultados obtenidos
4. Compara resultados obtenidos y analiza las diferencias entre los métodos aplicados
5. Elabora y presenta un reporte que incluya los procedimientos aplicados, las gráficas de la secuencia de los números pseudoaleatorios generados, los resultados numéricos y una conclusión sobre el comportamiento estadístico de la secuencia generada

Recursos de apoyo

- Computadora
- Internet

- Plataforma de Desarrollo Integrado, IDE
- Editor de texto
- Formato de práctica

Práctica 6. Pruebas estadísticas para determinar el tipo de distribución

Duración

02 horas

Procedimiento

1. Atiende las indicaciones de su docente para la realización de la práctica
2. Revisa los fundamentos teóricos de las pruebas de bondad de ajuste, empleadas para evaluar si una muestra de datos sigue una distribución teórica determinada
3. Aplica a un conjunto de datos generado o proporcionado las siguientes pruebas estadísticas: Prueba de bondad de ajuste Chi cuadrada y prueba de bondad de ajuste Kolmogorov–Smirnov
4. Analiza los resultados obtenidos y determina el tipo de distribución (por ejemplo, uniforme, normal o exponencial) que mejor se ajusta a los datos evaluados
5. Elabora y presenta un reporte que incluya los procedimientos aplicados, los resultados numéricos y gráficos, y una conclusión sobre la distribución identificada y la validez del ajuste estadístico

Recursos de apoyo

- Computadora
- Internet
- Plataforma de Desarrollo Integrado, IDE
- Editor de texto
- Formato de práctica

Práctica 7. Estrategia Montecarlo

Duración

03 horas

Procedimiento

1. Atiende las indicaciones de su docente para la realización de la práctica
2. Revisa los fundamentos teóricos del método de Montecarlo y su aplicación en la estimación del área bajo una curva o el cálculo de integrales definidas
3. Diseña e implementa un programa computacional que estime el área bajo una función $y=f(x)$ en un intervalo $[a,b]$, utilizando puntos pseudoaleatorios con distribución uniforme
4. Realiza simulaciones variando el número de puntos generados (por ejemplo, 100, 1000, 10000 y 100000) y registra los resultados obtenidos
5. Compara los valores estimados con el área calculada de forma analítica y analiza el error relativo obtenido en cada caso

6. Elabora y presenta un reporte que incluya los procedimientos aplicados, las gráficas de los puntos aleatorios generados y de la función, las tablas comparativas de los resultados, y una reflexión sobre la precisión y eficiencia del método de Montecarlo conforme aumenta el número de simulaciones

Recursos de apoyo

- Computadora
- Internet
- Plataforma de Desarrollo Integrado, IDE
- Editor de texto
- Formato de práctica

Unidad II. Componentes y organización de un modelo discreto

Práctica 8. Modelado del tiempo entre llegadas y tiempos de servicio

Duración

06 horas

Procedimiento

1. Atiende las indicaciones de su docente para la realización de la práctica
2. Revisa los fundamentos teóricos del modelado de llegadas y tiempos de servicio en sistemas de simulación de eventos discretos, identificando las distribuciones de probabilidad comúnmente utilizadas (exponencial, normal, uniforme, entre otras)
3. Diseña e implementa un programa computacional que genere muestras de tiempos entre llegadas y tiempos de servicio utilizando variables aleatorias no uniformes
4. Simula una secuencia de eventos de llegada y salida de clientes en un sistema con un solo servidor, registrando los tiempos generados
5. Analiza los valores estadísticos básicos de los tiempos obtenidos (media, varianza, desviación estándar) y representa los resultados mediante gráficas de distribución
6. Elabora y presenta un reporte que incluya los procedimientos realizados, las gráficas, el análisis estadístico y una reflexión sobre cómo los patrones de llegada y servicio influyen en el desempeño de los sistemas de espera

Recursos de apoyo

- Computadora
- Internet
- Plataforma de Desarrollo Integrado, IDE
- Editor de texto
- Formato de práctica

Unidad III. Lenguaje de simulación

Práctica 9. Explorar lenguajes de programación

Duración

09 horas

Procedimiento

1. Atiende las indicaciones de su docente para la realización de la práctica
2. Selecciona y explora un lenguaje de programación o entorno de simulación que permita modelar sistemas de eventos discretos (se recomienda el uso de Python con la librería SimPy, aunque pueden emplearse otros lenguajes equivalentes)
3. Investiga las estructuras básicas del lenguaje o entorno elegido (definición de procesos, manejo del tiempo, eventos, recursos y colas)
4. Desarrolla un programa introductorio que modele un proceso simple de llegada y atención de clientes en un sistema con un solo servidor
5. Ejecuta la simulación, registra los eventos generados y analiza cómo se representan las llegadas, los tiempos de espera y la atención de los clientes
6. Elabora y presenta un reporte que incluya una breve descripción del lenguaje o entorno utilizado, el pseudocódigo o diagrama de flujo del programa, las salidas principales obtenidas y una reflexión sobre las ventajas y limitaciones del lenguaje para el desarrollo de modelos de simulación

Recursos de apoyo

- Computadora
- Internet
- Plataforma de Desarrollo Integrado, IDE
- Editor de texto
- Formato de práctica

Unidad IV. Simulación de un sistema

Práctica 10. Simular eventos discretos

Duración

06 horas

Procedimiento

1. Atiende las indicaciones de su docente para la realización de la práctica
2. Revisa los fundamentos teóricos del modelado y simulación de sistemas de eventos discretos, identificando sus principales componentes: eventos, reloj de simulación, colas, recursos y variables de estado
3. Diseña e implementa un programa computacional que simule el comportamiento de un sistema de eventos discretos, definiendo claramente las condiciones iniciales, los tipos de eventos y las reglas de transición
4. Utiliza un lenguaje de programación o entorno de simulación adecuado (por ejemplo, Python con SimPy u otro equivalente)
5. Ejecuta la simulación bajo distintos escenarios o configuraciones y registra las métricas de desempeño más relevantes del sistema (por ejemplo, tiempo de espera, utilización de recursos o número de eventos procesados)

6. Elabora y presenta un reporte que incluya el modelo desarrollado, el pseudocódigo o diagrama de flujo, las gráficas y tablas de resultados, y una reflexión sobre la validez, eficiencia y posibles mejoras del sistema simulado

Recursos de apoyo

- Computadora
- Internet
- Plataforma de Desarrollo Integrado, IDE
- Editor de texto
- Formato de práctica

Unidad V. Proyectos

Práctica 11. Desarrollo de un proyecto

Duración

12 horas

Procedimiento

1. Atiende las indicaciones de su docente para la realización del proyecto
2. Diseña y desarrolla un proyecto de simulación que represente un fenómeno real o hipotético, aplicando los conocimientos adquiridos durante el curso
3. Documentar cada una de las fases del desarrollo del proyecto, desde la formulación del problema y diseño del modelo, hasta la implementación computacional y análisis de resultados
4. Elabora, en equipo, un programa de computadora que permita experimentar con el modelo implementado, demostrando iniciativa, creatividad y rigor técnico
5. Presenta un reporte final que incluya la descripción del modelo, los procedimientos aplicados, los resultados obtenidos y las conclusiones del proyecto

Recursos de apoyo

- Computadora
- Internet
- Plataforma de Desarrollo Integrado, IDE
- Editor de texto
- Formato de práctica

X. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Estudio de caso
- Método de proyectos
- Aprendizaje basado en problemas
- Técnica expositiva
- Debates
- Discusión guiada
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Mapas conceptuales
- Cuadros comparativos
- Elaboración de prototipos

XI. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	20%
– Tareas y actividades	25%
– Portafolio de prácticas laboratorio	25%
– Prototipo de simulación funcional	30%
Total	100%

XII. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Coss Bu, R. (2011). *Simulación un enfoque práctico* (2ª ed.). Limusa SA de CV. ISBN 978-968-18-1506-6. [Clásico]
- Esfandiari, M. R. A., & Dunna, E. G. (1996). *Simulación y análisis de modelos estocásticos*. MacGraw-Hill. ISBN 9701011732. [Clásico]
- Gould, Harvey (1938). *An introduction to computer simulation methods: applications to physical systems / Harvey Gould, Jan Tobochnik, Wolfgang Christian* (3rd ed.). Pearson Addison Wesley. ISBN 0-201-16503-1. [Clásico]
- Hillier, F. S., Lieberman, G. J & Murrieta, J. E. (2023). *Introducción a la investigación de operaciones* (7ª ed.). McGraw-Hill Interamericana Editores. ISBN: 9786071520296.
- Wainer, G.A. (2003). *Metodologías de modelización y simulación de eventos discretos* (1ª ed.). Nueva Librería. ISBN 9789871104017 9871104014. [Clásico]
- Woolfson, M. M., Pert, G.J. (1999). *An introduction to computer simulation*. Oxford University Press. ISBN 0198504233. [Clásico]

Recursos de acceso libre o complementarios externos

- Beaverstock, M., Greenwood, A., Lavery, E., & Nordgren, W. (2011). *Applied simulation. Modeling and Analysis using Flexsim*, Flexsim Software Products, Inc., Canyon Park Technology Center, Orem, USA. [Clásico]
- Beaverstock, M., Greenwood, A., Nordgren, W. (2017) *Applied Simulation: Modeling and Analysis using FlexSim* (5th ed.). FlexSim Software Products.
https://www.flexsim.com/wp-content/uploads/2025/07/Applied-Simulation-Modeling-and-Analysis-using-FlexSim_5th-ed.pdf
- Downey, A. B. (2023). *Modeling and Simulation in Python: An Introduction for Scientists and Engineers*. No Starch Press.
- Fishman, G. (2013). *Discrete-event simulation: modeling, programming, and analysis*. Springer Science & Business Media. [Clásico]
- Google. (2024). *Google Colab* [Computer software]. <https://colab.research.google.com>
- Gordon, G. (1969). *System simulation*. Prentice Hall. [Clásico]
- Kelton, W. D., & Law, A. M. (2000). *Simulation modeling and analysis*. McGraw-Hill. [Clásico]
- Michael Pidd (2006). *Computer Simulation in Management Science* (5th ed). [Clásico]
- University of Colorado Boulder. (2025). *PhET Interactive Simulations*. <https://phet.colorado.edu>

XIII. PERFIL DOCENTE

Profesionista en Ciencias Computacionales o área afín, preferentemente con posgrado, tener experiencia en docencia y en modelación matemática, probabilidad y simulación por computadora. Ser una persona creativa, que emplee y fomente el trabajo colaborativo, desarrolle el pensamiento crítico en sus estudiantes y promueva la ética profesional.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada.
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Sistemas Distribuidos
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 03 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Terminal
Carácter:	Obligatoria
Requisito:	Ninguno.
Modalidad de impartición:	Presencial
Equipo de diseño:	José E. Lozano Rizk Eloísa del Carmen García Canseco
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La importancia de esta unidad de aprendizaje radica en la omnipresencia de los sistemas distribuidos en la infraestructura tecnológica actual, desde las aplicaciones móviles hasta la computación en la nube. Las y los estudiantes adquirirán las habilidades para diseñar, implementar y gestionar sistemas de software que coordinan múltiples componentes en red, garantizando eficiencia, escalabilidad y tolerancia a fallos.

Se ubica en la etapa terminal, es de carácter obligatorio y pertenece al área de conocimiento de Programación e Ingeniería de Software.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Diseñar soluciones de software distribuido, mediante la integración e implementación de modelos arquitectónicos, protocolos de comunicación y mecanismos de sincronización, para resolver problemas de escalabilidad y disponibilidad de la información en organizaciones modernas, con una actitud crítica, responsable y orientada a estándares de la industria

IV. EVIDENCIA DE APRENDIZAJE

Portafolio de prácticas que documente la resolución de los laboratorios del curso. Este portafolio deberá evidenciar el proceso de desarrollo técnico mediante la inclusión del código fuente documentado de los servicios (backend/frontend), diagramas de arquitectura y secuencia, y registros de ejecución (logs estructurados, trazas o video) que demuestren la correcta comunicación entre nodos, la sincronización de procesos y la consistencia de los datos distribuidos.

Proyecto final de un prototipo funcional de un sistema distribuido escalable (basado en microservicios o clúster de procesamiento) que resuelva una problemática específica de alta disponibilidad o gestión de información masiva. Esta evidencia se materializa en dos entregables: el sistema de software desplegado y operando en un entorno orquestado (que debe integrar comunicación heterogénea RPC/REST, replicación de base de datos y mecanismos de recuperación automática ante fallos); y un reporte técnico formal. Dicho reporte debe detallar la justificación de los patrones de arquitectura seleccionados (análisis de compromisos del Teorema CAP), la metodología de despliegue, las pruebas de resiliencia (caos) y estrés realizadas, y el manual de instalación/operación.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Desarrollar software de base eficiente y robusto mediante la integración de conocimientos en arquitectura y redes de computadoras y los fundamentos de compiladores y sistemas operativos, para optimizar el rendimiento, la gestión de recursos y la comunicación entre sistemas, asegurando aplicaciones funcionales, seguras, escalables y sostenibles, con una actitud ética, responsable y visionaria.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia.

VII. DESARROLLO POR UNIDADES

Unidad I. Introducción a los sistemas distribuidos

Competencia

Analizar los fundamentos y conceptos generales de los sistemas distribuidos, mediante la revisión de su definición, objetivos y taxonomía para comprender el cambio de paradigma frente a los sistemas centralizados, con objetividad y disposición al aprendizaje.

Contenido

- 1.1. Definición de un sistema distribuido
- 1.2. Objetivos
- 1.3. Tipos de sistemas distribuidos

Duración

04 horas

Unidad II. Modelos y Arquitecturas

Competencia

Discriminar las arquitecturas de sistemas distribuidos por medio del estudio de los modelos fundamentales y capas de middleware, para identificar la estructura más adecuada según los requisitos del sistema, con actitud crítica y responsable.

Contenido

- 2.1. Modelos arquitectónicos
- 2.2. Arquitecturas de sistemas
- 2.3. Arquitecturas y middleware
- 2.4. Autoadministración en sistemas distribuidos

Duración

04 horas

Unidad III. Procesos y objetos distribuidos

Competencia

Discriminar los distintos tipos de procesos y objetos distribuidos a través del análisis de patrones de virtualización y concurrencia, para organizar eficientemente la ejecución en sistemas cliente-servidor, con objetividad y colaboración.

Contenido

- 3.1 Hilos
- 3.2 Virtualización y contenedores
- 3.3 Clientes
- 3.4 Servidores
- 3.5 Arquitectura de objetos distribuidos

Duración

04 horas

Unidad IV. Comunicación en sistemas distribuidos

Competencia

Analizar los conceptos fundamentales de comunicación mediante la revisión de protocolos de intercambio de información y mecanismos RPC/MOM, para implementar interacciones eficientes y seguras entre procesos, con actitud crítica y ética profesional.

Contenido

- 4.1. Protocolos de comunicación: TCP/IP, HTTP/REST, gRPC
- 4.2. Llamadas a procedimientos remotos (RPC, gRPC)
- 4.3. Comunicación orientada a mensajes (MOM)
- 4.4. Comunicación orientada a flujos
- 4.5. Seguridad en la comunicación (TLS y Autenticación)

Duración

04 horas

Unidad V. Nombrado

Competencia

Analizar los métodos de nombrado a través de las técnicas de identificación plana y estructurada, para resolver el descubrimiento y localización de recursos en la red, con objetividad y rigor técnico.

Contenido

- 5.1 Nombrado, identificadores y direcciones

- 5.2 Nombrado plano
- 5.3 Nombrado estructurado
- 5.4 Nombrado basados en atributos

Duración

04 horas

Unidad VI. Sincronización y consenso

Competencia

Organizar la coordinación de procesos a través del estudio de algoritmos de sincronización de relojes y elección, para resolver disputas de recursos en el tiempo y evitar condiciones de carrera, con actitud propositiva y respeto a estándares.

Contenido

- 6.1 Sincronización del reloj: Algoritmos de Cristian y Berkeley.
- 6.2 Relojes lógicos: Lamport y Vectores de Tiempo.
- 6.3 Exclusión mutua distribuida: Algoritmos centralizados y distribuidos
- 6.4 Algoritmos de elección

Duración

04 horas

Unidad VII. Consistencia y tolerancia a fallos

Competencia

Determinar los mecanismos de manipulación de datos por medio de técnicas de replicación, modelos de consistencia y protocolos de recuperación, para mejorar la confiabilidad y disponibilidad de la información ante fallos, con objetividad, actitud crítica y honestidad.

Contenido

- 7.1 Introducción
- 7.2 Modelos de consistencia y Teorema CAP
- 7.3 Estrategias de replicación de datos
- 7.4 Consenso distribuido
- 7.5 Introducción a la resiliencia y tolerancia a fallos

Duración

02 horas

Unidad VIII. Sistemas de archivo y almacenamiento distribuido

Competencia

Analizar los diferentes tipos de sistemas de archivo y almacenamiento a través de la comparación de técnicas de distribución de datos, para la creación de sistemas de persistencia robustos y transparentes, con objetividad y responsabilidad.

Contenido

- 8.1 Sistema de archivos de red.
- 8.2 Sistemas de archivo distribuido
- 8.3 Sistemas de archivos paralelo y distribuido
- 8.4 Bases de datos distribuidas

Duración

02 horas

Unidad IX. Tendencias tecnológicas en sistemas distribuidos

Competencia

Examinar los componentes de arquitecturas modernas, a través de los paradigmas de microservicios, nube y orquestación, para la implementación de sistemas distribuidos de última generación con actitud objetiva, propositiva e innovadora.

Contenido

- 9.1 Arquitectura de microservicios vs monolíticos
- 9.2 Contenedores y orquestación
- 9.3 Modelos nativos de cómputo en la nube
- 9.4 Sistema de computación en el borde
- 9.5 Blockchain como sistema distribuido

Duración

04 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad II. Modelos y arquitectura

Práctica 1: Creación de puertos de comunicación entre dos computadoras en red

Duración

4 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.

2. Analiza la documentación de las librerías de Sockets (POSIX, Java Net o Python Socket).
3. Diseña un algoritmo de comunicación síncrona entre un cliente y un servidor.
4. Codifica los scripts necesarios para abrir puertos de escucha y realizar conexiones de red.
5. Verifica la conectividad y el intercambio de mensajes de texto plano entre dos computadoras distintas.
6. Documenta el código fuente y las pruebas de conexión en el reporte de la práctica.
7. Entrega el reporte y recibe retroalimentación por parte del docente.

Recursos de apoyo

- 2 computadoras personales conectadas en red
- Software de desarrollo (IDE o editor de texto, intérprete/compilador).

Unidad III. Procesos y objetos distribuidos

Práctica 2: Transmisión de información en un sistema distribuido

Duración

06 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Selecciona un formato de serialización estándar (JSON, XML, Pickle o Protocol Buffers).
3. Diseña una clase u objeto con múltiples tipos de datos (cadenas, enteros, listas).
4. Implementa el mecanismo de marshalling (serialización) en el emisor y unmarshalling (deserialización) en el receptor.
5. Transmite el objeto a través de la red y valida la integridad de la información recibida.
6. Elabora un reporte técnico comparando el tamaño de los datos antes y después de la serialización.
7. Entrega el reporte y recibe retroalimentación por parte del docente.

Recursos de apoyo

- 2 computadoras personales conectadas en red
- Software de desarrollo

Práctica 3: Atención de peticiones de comunicación entre diferentes clientes y un servidor.

Duración

06 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Analiza el problema de bloqueo en servidores iterativos (single-threaded).

3. Diseña la arquitectura de un servidor concurrente que instancie un nuevo hilo por cada petición recibida.
4. Programa la gestión de hilos utilizando las librerías de concurrencia del lenguaje seleccionado.
5. Ejecuta pruebas de estrés conectando múltiples clientes simultáneos al servidor.
6. Registra los tiempos de respuesta y documenta la mejora en el rendimiento.
7. Entrega el reporte y recibe retroalimentación por parte del docente.

Recursos de apoyo

- 2 computadoras personales conectadas en red
- Software de desarrollo

Unidad IV. Comunicación en sistemas distribuidos

Práctica 4: Implementación de comunicación vía HTTP (API REST)

Duración

08 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Define los recursos y las rutas (endpoints) de la API a implementar.
3. Implementa el servidor utilizando un framework web ligero (ej. Flask, Express, Spring Boot).
4. Programa los métodos HTTP adecuados (GET, POST, PUT, DELETE) para las operaciones del sistema.
5. Desarrolla un cliente que consuma la API y procese las respuestas de estado HTTP.
6. Valida el funcionamiento de la API utilizando herramientas de prueba (ej. Postman o cURL) y documenta los resultados.
7. Entrega el reporte y recibe retroalimentación por parte del docente.

Recursos de apoyo

- 2 computadoras personales conectadas en red
- Software de desarrollo

Unidad VI. Sincronización y consenso

Práctica 5: Uso eficiente de recursos compartidos

Duración

08 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.

2. Identifica una sección crítica en un problema de acceso concurrente a un archivo o variable compartida.
3. Selecciona una técnica de exclusión mutua (semáforos distribuidos, algoritmo de anillo o centralizado).
4. Implementa la lógica de bloqueo y desbloqueo de recursos en el sistema distribuido.
5. Simula un escenario de competencia por el recurso entre múltiples nodos.
6. Comprueba y documenta que solo un proceso accede al recurso a la vez, garantizando la consistencia.
7. Entrega el reporte y recibe retroalimentación por parte del docente.

Recursos de apoyo

- 2 computadoras personales conectadas en red
- Software de desarrollo

Unidad VII. Consistencia y tolerancia a fallos

Práctica 6: Técnicas de consistencia y replicación de datos

Duración

08 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Configura un entorno con un nodo maestro y al menos un nodo esclavo (réplica).
3. Implementa un mecanismo de replicación (activa o pasiva) o configura una base de datos que soporte replicación nativa.
4. Realiza operaciones de escritura en el nodo principal y verifica la propagación de datos a las réplicas.
5. Simula la caída del nodo principal y verifica la disponibilidad de lectura en las réplicas.
6. Analiza y reporta el comportamiento del sistema en términos de consistencia eventual o fuerte.
7. Entrega el reporte y recibe retroalimentación por parte del docente.

Recursos de apoyo

- 2 computadoras personales conectadas en red
- Software de desarrollo y gestor de base de datos (ej. MongoDB, Redis, PostgreSQL).

Unidad IX. Tendencias tecnológicas

Práctica 7: Despliegue de sistemas modernos (Nube y Contenedores)

Duración

08 horas

Procedimiento

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Selecciona una aplicación distribuida desarrollada previamente.
3. Crea los archivos de configuración para la contenedorización (ej. Dockerfile) de cada servicio.
4. Construye las imágenes y configura la orquestación de los contenedores (ej. Docker Compose o Kubernetes).
5. Despliega el sistema completo en un entorno local o en un servicio de nube gratuito.
6. Verifica la interoperabilidad de los servicios contenerizados y documenta el proceso de despliegue en el reporte técnico.
7. Entrega el reporte y recibe retroalimentación por parte del docente.

Recursos de apoyo

- 2 computadoras personales con soporte para virtualización
- Software de contenedorización (Docker, Podman) y orquestación.

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer la forma de trabajo, los criterios de evaluación, la calidad de los trabajos académicos y los derechos y obligaciones docente-estudiante. Esto debe quedar registrado y comunicado por medios institucionales (como Blackboard o correo institucional).

Estrategia de enseñanza

- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Técnica expositiva
- Debates
- Discusión guiada
- Instrucción guiada
- Foros

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Mapas mentales
- Mapas conceptuales

- Cuadros comparativos
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a los exámenes ordinarios y extraordinarios, cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación está en una escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Portafolio de prácticas	30%
– Tareas	10%
– Exámenes	20%
– Proyecto final	40%
Total	100

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Burns, B. (2018). *Designing distributed systems: Patterns and paradigms for scalable, reliable services* (1st ed.). Sebastopol, CA: O'Reilly Media. [Básica]
- Coulouris, G., Dollimore, J., Kindberg, T., & Blair, G. (2012). *Distributed systems: Concepts and design* (5th ed.). Boston: Addison-Wesley. [Básica]
- Institute of Electrical and Electronics Engineers (IEEE). (1989–presente). *IEEE Transactions on Parallel and Distributed Systems*. IEEE. ISSN 1045-9219.
- Kleppmann, M. (2017). *Designing data-intensive applications: The big ideas behind reliable, scalable, and maintainable systems*. Sebastopol, CA: O'Reilly Media. [Básica]
- Newman, S. (2015). *Building microservices: Designing fine-grained systems* (1st ed.). O'Reilly Media. [Básica]
- Tanenbaum, A. S., & Van Steen, M. (2002). *Distributed systems: Principles and paradigms*. Upper Saddle River, NJ; London: Prentice Hall PTR. [Básica]

Recursos de acceso libre o complementarios externos

Beyer, B., Jones, C., Petoff, J., & Murphy, N. R. (2016). *Site reliability engineering: How Google runs production systems*. O'Reilly Media. Disponible gratuitamente en <https://sre.google/books/>

Martin, D. (s.f.). *The System Design Primer*. GitHub. <https://github.com/donnemartin/system-design-primer>

Microsoft Azure Architecture Center. (s.f.). *Patrones de diseño en la nube (Cloud Design Patterns)*. <https://learn.microsoft.com/es-es/azure/architecture/patterns/>

The Cloud Native Computing Foundation (CNCF). (s.f.). *Cloud Native landscape & documentation*. Recuperado de <https://www.cncf.io/>

Takada, M. (2013). *Distributed systems: For fun and profit*. <https://book.mixu.net/distsys/ebook.html>

XII. PERFIL DOCENTE

Se requiere que la persona docente cuente con Licenciatura en el área de ciencias computacionales y, preferentemente, con estudios de posgrado en un área afín. Deberá tener experiencia profesional y docente mínima de dos años, así como conocimientos en el desarrollo de sistemas de cómputo distribuido. Asimismo, se espera que sea una persona organizada, proactiva y responsable en el desempeño de sus funciones.

****PROGRAMA DE UNIDAD DE APRENDIZAJE**

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Proyecto Científico de Cómputo Inteligente
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 02 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos: 06
Etapas de formación:	Terminal
Carácter:	Obligatoria
Requisito:	Aprendizaje Profundo y Modelos Generativos
Modalidad de impartición:	Presencial
Equipo de diseño:	Everardo Gutiérrez López Alma Rocío Cavazos Marín Antonio de Jesús García Chávez
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Esta unidad de aprendizaje tiene como propósito que el estudiantado integre y aplique los enfoques del Cómputo Inteligente mediante el diseño, desarrollo y validación de un proyecto científico interdisciplinario, orientado a la resolución de una problemática real. Debe a su vez fomentar el pensamiento crítico, rigor científico, ética profesional, creatividad, autonomía, compromiso social y tecnológico, tal que fortalezca el uso adecuado del Cómputo Inteligente en la solución de problemas de la computación y disciplinas afines. Se imparte en la etapa terminal, es de carácter obligatorio y pertenece al área de conocimiento de Interacción Hombre Máquina, y tiene como requisito haber aprobado la unidad de aprendizaje de Aprendizaje Profundo y Modelos Generativos.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Crear soluciones de cómputo inteligente, a través de la integración de manera crítica y creativa de diversos enfoques de la Inteligencia Artificial, para resolver problemáticas reales en contextos científicos, sociales, industriales o tecnológicos, con ética, responsabilidad social y trabajo colaborativo.

IV. EVIDENCIA DE APRENDIZAJE

Crear un proyecto integrador mediante el diseño, desarrollo, implementación y evaluación de una solución basada en técnicas de Inteligencia Artificial y Cómputo Inteligente, orientado a la resolución de una problemática real en contextos científicos, sociales, industriales o tecnológicos. El estudiantado deberá integrar los distintos enfoques estudiados a lo largo de la licenciatura, aplicar metodología científica, y demostrar capacidad de análisis, creatividad, autonomía, trabajo colaborativo y compromiso ético, evidenciando el impacto social y tecnológico de la solución propuesta.

El proyecto deberá incluir:

- Planteamiento formal del problema.
- Estado del arte.
- Marco metodológico.
- Implementación del sistema.
- Evaluación y resultados.
- Impacto y reflexión ética.
- Conclusiones.

La evidencia del proyecto integrador deberá presentarse con los siguientes elementos:

- Informe escrito.
- Producto funcional.
- Presentación oral.
- Anexos (en caso de requerirlos).

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar y evaluar soluciones de cómputo inteligente aplicando principios de inteligencia artificial, minería de datos, e integración de herramientas computacionales, con el propósito de apoyar la toma de decisiones en situaciones complejas propias de contextos científicos, sociales, industriales o tecnológicos, actuando con responsabilidad ética, pensamiento crítico, y colaboración interdisciplinaria.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Compromiso con la responsabilidad social; Excelencia; Vanguardia e Innovación Social

VII. DESARROLLO POR UNIDADES

Unidad I. Planteamiento del problema y fundamentación científica

Competencia

Analizar los componentes de una problemática aplicada, mediante la identificación de variables, contexto y restricciones, con el apoyo de la revisión crítica de literatura científica, para formular un problema de investigación abordable con técnicas de cómputo inteligente, con pensamiento crítico, ética académica y responsabilidad social.

Contenido

- 1.1. Naturaleza del proyecto científico en Cómputo Inteligente
- 1.2. Tipos de problemas abordables con IA
- 1.3. Identificación de contextos: social, científico, industrial, tecnológico
- 1.4. Formulación del problema y objetivos
- 1.5. Preguntas de investigación e hipótesis
- 1.6. Revisión sistemática del estado del arte
- 1.7. Ética en la investigación y uso de datos

Duración

06 horas

Unidad II. Diseño metodológico de la solución inteligente

Competencia

Diseñar una solución de cómputo inteligente, mediante la selección y articulación de modelos, algoritmos y técnicas de IA, para plantear una metodología de cómputo inteligente viable y reproducible, apegado a una toma de decisiones fundamentada, con creatividad y trabajo colaborativo.

Contenido

- 2.1. Enfoques de Cómputo Inteligente

- 2.2. Selección de datasets
- 2.3. Diseño del flujo del sistema inteligente
- 2.4. Definición de métricas de evaluación
- 2.5. Consideraciones éticas y sesgos algorítmicos

Duración

08 horas

Unidad III. Implementación y experimentación

Competencia

Implementar y entrenar un sistema de cómputo inteligente, mediante el uso de herramientas computacionales y experimentación controlada, para validar el funcionamiento de la solución propuesta, mostrando perseverancia, rigor técnico y responsabilidad profesional.

Contenido

- 3.1. Implementación del modelo (en lenguaje Python o equivalente)
- 3.2. Uso de librerías de Inteligencia Artificial (por ejemplo scikit-learn, TensorFlow, PyTorch)
- 3.3. Preprocesamiento de datos
- 3.4. Entrenamiento y validación
- 3.5. Ajuste de hiperparámetros
- 3.6. Registro y análisis de resultados experimentales

Duración

10 horas

Unidad IV. Evaluación, comunicación científica y reflexión ética

Competencia

Evaluar y comunicar los resultados del proyecto científico, mediante el análisis crítico, la comparación con el estado del arte y la divulgación académica, para valorar el impacto social, científico y tecnológico de la solución, con honestidad intelectual, compromiso social y pensamiento reflexivo.

- 4.1. Evaluación cuantitativa y cualitativa del sistema
- 4.2. Análisis comparativo con trabajos relacionados
- 4.3. Impacto social y tecnológico
- 4.4. Limitaciones y trabajo futuro
- 4.5. Redacción de informe científico
- 4.6. Presentación y defensa del proyecto

Duración

08 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO**Unidad I. Planteamiento del problema y fundamentación científica****Práctica 1: Identificación y análisis de una problemática abordable con Cómputo Inteligente*****Duración***

02 horas

Procedimiento

1. Atiende las instrucciones de su docente.
2. Analiza casos reales donde se haya aplicado IA en contextos sociales, científicos o industriales.
3. Selecciona un dominio de interés (salud, educación, industria, medio ambiente, etc.).
4. Identifica una problemática concreta dentro del dominio elegido.
5. Describe el contexto, actores involucrados y posibles beneficios sociales o tecnológicos.
6. Determina si el problema es de clasificación, regresión, optimización, predicción o decisión.
7. Documenta riesgos éticos preliminares (sesgos, privacidad, impacto social).
8. Presenta el análisis y recibe retroalimentación.

Recursos de apoyo

- Artículos de IEEE Xplore / ACM Digital Library
- Presentaciones introductorias del docente
- Recursos para trabajo colaborativo
- Plantilla de análisis de problemas

Práctica 2: Revisión sistemática del estado del arte***Duración***

04 horas

Procedimiento

1. Atiende las indicaciones de su docente.
2. Define palabras clave y criterios de búsqueda.
3. Consulta bases de datos académicas (IEEE, Springer, Scopus, Google Scholar).
4. Selecciona al menos 8–10 artículos relevantes.
5. Clasifica los trabajos por enfoque, técnica y resultados.
6. Identifica tendencias, limitaciones y oportunidades de mejora.
7. Elabora una matriz comparativa de literatura.

8. Redacta un resumen crítico del estado del arte y recibe retroalimentación.

Recursos de apoyo

- Gestores bibliográficos (por ejemplo Zotero, Mendeley)
- Plantilla de revisión sistemática
- Normas APA / IEEE
- Tutorial de búsqueda académica

Unidad II. Diseño metodológico de la solución inteligente

Práctica 3: Selección y justificación del enfoque de Cómputo Inteligente

Duración

04 horas

Procedimiento

1. Atiende las indicaciones de su docente.
2. Revisa las características del problema definido.
3. Identifica posibles enfoques de Cómputo Inteligente aplicables.
4. Compara ventajas y desventajas de cada enfoque.
5. Selecciona el modelo o combinación de modelos más adecuado.
6. Justifica la selección con base en literatura científica.
7. Documenta supuestos, restricciones y riesgos.
8. Socializa la propuesta y recibe retroalimentación.

Recursos de apoyo

- Libros de IA y aprendizaje automático
- Diagramas comparativos de modelos
- Artículos de referencia
- Plantilla de justificación metodológica

Práctica 4: Diseño arquitectónico y métricas de evaluación

Duración

04 horas

Procedimiento

1. Atiende las indicaciones docentes.
2. Identifica entradas, salidas y procesos del sistema.
3. Diseña el flujo de datos y procesamiento.
4. Define arquitectura general del sistema.
5. Selecciona métricas de evaluación adecuadas al problema.
6. Analiza posibles sesgos y limitaciones del diseño.
7. Elabora un documento de diseño técnico.
8. Recibe retroalimentación docente.

Recursos de apoyo

- Herramientas de diagramación
- Guías de métricas de evaluación
- Casos de estudio
- Plantilla de diseño arquitectónico

Unidad III. Implementación y experimentación

Práctica 5: Implementación inicial del modelo inteligente

Duración

06 horas

Procedimiento

1. Atiende las orientaciones docentes.
2. Prepara el entorno de desarrollo (por ejemplo Python, librerías).
3. Carga y explora el conjunto de datos.
4. Realiza el preprocesamiento básico.
5. Implementa el modelo seleccionado.
6. Ejecuta el entrenamiento inicial.
7. Registra resultados preliminares.
8. Documenta el código y decisiones tomadas.
9. Recibe retroalimentación docente.

Recursos de apoyo

- Python, Jupyter Notebook
- Scikit-learn / TensorFlow / PyTorch
- Datasets abiertos
- Guías de implementación

Práctica 6: Ajuste del modelo y experimentación controlada

Duración

06 horas

Procedimiento

1. Atiende la explicación docente.
2. Analiza los resultados iniciales.
3. Ajusta los hiperparámetros del modelo.
4. Experimenta con diferentes configuraciones del modelo.
5. Valida los resultados con conjuntos de prueba.
6. Compara métricas de desempeño.
7. Registra resultados en tablas y gráficas.
8. Reflexiona sobre limitaciones técnicas.
9. Presenta informe preliminar y recibe retroalimentación.

Recursos de apoyo

- Librerías de optimización
- Herramientas de visualización (Matplotlib, Seaborn)
- Plantilla de bitácora experimental

Unidad IV. Evaluación, comunicación científica y reflexión ética

Práctica 7: Evaluación integral e impacto del sistema inteligente

Duración

04 horas

Procedimiento

1. Atiende la explicación docente.
2. Analiza métricas finales de desempeño.
3. Compara resultados con trabajos del estado del arte.
4. Evalúa el impacto social y tecnológico del sistema.
5. Identifica riesgos éticos y posibles mitigaciones.
6. Documenta limitaciones y trabajo futuro.
7. Elabora conclusiones fundamentadas.
8. Recibe retroalimentación docente.

Recursos de apoyo

- Artículos comparativos
- Guías de evaluación ética en IA
- Plantilla de análisis de impacto

Práctica 8: Redacción y presentación del proyecto científico

Duración

02 horas

Procedimiento

1. Atiende las indicaciones docentes.
2. Estructura el informe final del proyecto.
3. Redacta cada sección con rigor científico.
4. Aplica normas de citación adecuadas.
5. Prepara presentación oral del proyecto.
6. Ensayo defensa técnica y ética del trabajo.
7. Presenta el proyecto ante el grupo.
8. Recibe y documenta retroalimentación.

Recursos de apoyo

- Plantillas de artículos científicos
- Normas IEEE / APA

- Herramientas de presentación
- Rúbrica de evaluación

X. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Técnica expositiva
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Cuadros comparativos
- Infografías

XI. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	30%
– Prácticas laboratorio	30%
– Proyecto integrador	40%
Total	100%

XII. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Bhuvaneswari, B. (2021). *Artificial Intelligence*. Astral International Pvt Ltd. <https://libcon.rec.uabc.mx:2938/linkprocessor/plink?id=52bc4529-dabb-3336-a44d-53a5dbd29a41>
- Centro de Investigación para el Aprendizaje Digital. (2024). *Recomendaciones para la integración efectiva de la inteligencia artificial generativa en la práctica docente de educación superior*. Universidad Autónoma de Baja California. https://ciad.mxlib.uabc.mx/wp-content/uploads/2024/01/IA_Practica_Docente_2024-01-24.pdf
- Chollet, F. (2021). *Deep learning with Python*. Manning.
- Ekman, M. (2021). *Learning deep learning: Theory and practice of neural networks, computer vision, natural language processing, and transformers using TensorFlow*. Addison-Wesley Professional.
- Geron, A. (2022). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow* (3rd ed.). O'Reilly Media.
- Goodfellow, I. (2016). *Deep learning*. Ian Goodfellow, Yoshua Bengio, Aaron Courville-Google Books. [Clásica]
- Moroney, L. (2020). *AI and Machine Learning for Coders: A Programmer's Guide to Artificial Intelligence*. O'Reilly Media.
- Neapolitan, R. E., & Jiang, X. (2018). *Artificial intelligence: With an introduction to machine learning*. CRC press. [Clásica]
- Russell, S., & Norvig, P. (2021). *Artificial Intelligence: A Modern Approach* (4a ed.). Pearson.

Recursos de acceso libre o complementarios externos

- ACM SIGAI. (s.f.). *ACM SIGAI – Special Interest Group on Artificial Intelligence*. <https://sigai.acm.org/main/>
- European Union (2023). *EU Artificial Intelligence Act (AI Act)*. <https://artificialintelligenceact.eu/>
- Goyal, M., & Mahmoud, Q. H. (2024). A systematic review of synthetic data generation techniques using generative AI. *Electronics*, 13(17), 3509. <https://doi.org/10.3390/electronics13173509>
- He, R., & al. (2023). A survey on GANs for computer vision: Recent research, analysis and taxonomy. *Computer Science Review*, 48, 100553. <https://doi.org/10.1016/j.cosrev.2023.100553>
- Ibrahim, M., Al Khalil, Y., Amirrajab, S., Sun, C., Breeuwer, M., Pluim, J., Elen, B., Ertaylan, G., & Dumontier, M. (2024). *Generative AI for synthetic data across multiple medical*

modalities: A systematic review of recent developments and challenges. arXiv. <https://arxiv.org/abs/2407.00116>

Koo, H., & Kim, T. E. (2023). *A comprehensive survey on generative diffusion models for structured data.* arXiv. <https://arxiv.org/abs/2306.04139>

Murphy, K. P. (2012). *Machine learning: A probabilistic perspective.* MIT Press. <https://www.cs.ubc.ca/~murphyk/>

Shalev-Shwartz, S., y Ben-David, S. (2014). *Understanding machine learning: From theory to algorithms.* Cambridge University Press. [clásica] <https://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/>

Suh, N., & Cheng, G. (2025). A survey on statistical theory of deep learning: Approximation, training dynamics, and generative models. *Annual Review of Statistics and Its Application*, 12, 177–207. <https://doi.org/10.1146/annurev-statistics-040522-013920>

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... y Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30. https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

Yazdani, S., Singh, A., & Saxena, N. (2025). Generative AI in depth: A survey of recent advances, model variants, and real-world applications. *Journal of Big Data*, 12, 230. <https://doi.org/10.1186/s40537-025-01247-x>

Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2023). *Dive into deep learning.* Cambridge University Press. <https://d2l.ai/index.html>

XIII. PERFIL DOCENTE

Licenciatura en Ciencias Computacionales o áreas afines, preferentemente con posgrado en Ciencias Computacionales, Inteligencia Artificial o áreas afines. Debe tener un dominio sólido y actualizado de los fundamentos teóricos y prácticos del Cómputo Inteligente, el Aprendizaje Automático y la Metodología de la Investigación Científica aplicada a la computación. Se recomienda poseer experiencia en el diseño, desarrollo y evaluación de proyectos científicos, así como en el uso de herramientas computacionales modernas para la implementación de sistemas inteligentes, manteniendo una visión crítica sobre el impacto social, ético y tecnológico de la Inteligencia Artificial.

Adicionalmente, debe demostrar experiencia docente a nivel licenciatura, con capacidad para diseñar ambientes de aprendizaje activos, guiar proyectos integradores interdisciplinarios y evaluar por competencias. Se espera que promueva en sus estudiantes el pensamiento crítico, la creatividad, la autonomía y el trabajo colaborativo, actuando con ética profesional, honestidad académica y responsabilidad social, además de asumir un rol alienado con la mentoría académica comprometida con la formación integral y la mejora social y tecnológica a través del uso responsable del Cómputo Inteligente.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Física Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Pensamiento Crítico, Ciencia y Humanidades
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 02 Horas Laboratorio: 00 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:06
Etapas de formación:	Básica
Carácter:	Optativa
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Jorge Alberto Villavicencio Aguilar Roberto Romo Martínez Claudio Ismael Valencia
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La finalidad de esta unidad de aprendizaje es que el estudiantado de Física desarrolle una comprensión profunda y situada del origen, construcción y sentido del conocimiento científico, mediante una genealogía filosófica que inicia en el pensamiento presocrático y llega a la ciencia moderna y contemporánea. A través de experiencias de taller encarnadas orientadas a “ponerse en la piel” de quienes pensaron antes, a preguntar desde la curiosidad, la duda y la imaginación la unidad busca fortalecer el pensamiento crítico y creativo como habilidad fundamental para la formación científica, articulando rigor conceptual con sensibilidad humana. Asimismo, se pretende que el estudiantado reconozca la ciencia como práctica histórica y social, capaz de dialogar con otros saberes sin soberbia epistémica, e integre su aprendizaje a una reflexión ética y personal sobre su vocación y proyecto vital como futuros científicos o científicas. La unidad de aprendizaje pertenece al plan de estudios de Licenciatura en Física, se ubica en la etapa básica, es de carácter optativo y pertenece al área de conocimiento de Física; además se comparte con la Licenciatura en Ciencias Computacionales, se ubica en etapa disciplinaria con carácter optativo.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Examinar profundamente la construcción del conocimiento científico desde una perspectiva histórico-filosófica, mediante el estudio de sus bases desde el pensamiento presocrático hasta la ciencia moderna y contemporánea integrando la experiencia corporal, curiosidad, imaginación y argumentación rigurosa, para desarrollar un pensamiento crítico propio que permita analizar, cuestionar y comunicar ideas científicas de forma situada, plural y éticamente responsable, con creatividad, empatía, respeto a la diversidad y a la dimensión humana del conocer.

IV. EVIDENCIA DE APRENDIZAJE

Portafolio integrador de evidencias, elaborado a lo largo del semestre, que reúna productos escritos, reflexivos y vivenciales, donde el estudiantado muestra la construcción progresiva de su pensamiento crítico científico. El portafolio incluirá:

- **Bitácora de asombro y de preguntas “inocentes”**: registros breves donde el estudiantado documenta experiencias de curiosidad, extrañamiento y duda surgidas en prácticas de taller, transformándolas en preguntas y reflexiones filosófico-científicas.
- **Ensayos cortos / comentarios argumentativos** en los que articule conceptos presocráticos, históricos y epistemológicos con su comprensión del quehacer científico en Física, justificando sus ideas con rigor conceptual, claridad y referencias pertinentes.

- **Lecturas críticas guiadas** de textos filosóficos y científicos (o de divulgación), donde identifique supuestos, reconstruya argumentos, detecte límites, sesgos o falacias y contraste posiciones desde una perspectiva situada y pluralista.
- **Reportes reflexivos de prácticas encarnadas** (“ponerse en la piel de nuestros antecesores”): descripciones y análisis de actividades como medición sin instrumentos, traducción conceptual, experimentos situados o escenarios contrafácticos, explicando cómo la experiencia corporal y el diseño creativo de procedimientos fortalecen el pensamiento científico.
- **Síntesis final integrada (escrita o multimodal)** que exponga la postura epistemológica del estudiantado y su conexión con la ciencia como práctica histórica, humana y éticamente responsable; esta síntesis podrá tomar forma de ensayo, presentación oral, pieza audiovisual breve, mapa conceptual narrado o portafolio curado.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE:

Licenciatura en Física

Plantear modelos de sistemas físicos basados en la comprensión de los fenómenos de la naturaleza para resolver problemas de física con rigor científico, responsabilidad y honestidad.

Licenciatura en Ciencias Computacionales

Diseñar y evaluar soluciones de cómputo inteligente aplicando principios de inteligencia artificial, minería de datos, e integración de herramientas computacionales, con el propósito de apoyar la toma de decisiones en situaciones complejas propias de contextos científicos, sociales, industriales o tecnológicos, actuando con responsabilidad ética, pensamiento crítico, y colaboración interdisciplinaria.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Pensamiento Crítico. Compromiso con la responsabilidad social, Equidad social y de género, Inclusión, Excelencia, Innovación social e Interculturalidad.

VII. DESARROLLO POR UNIDADES

Unidad I. El asombro y los orígenes del conocer

Competencia

Examinar el surgimiento del pensamiento racional a partir de la experiencia humana del asombro, la curiosidad y la duda, en el tránsito del mito al logos y en los fundamentos del pensamiento presocrático, para comprender el origen histórico-filosófico de la ciencia y elaborar preguntas, conceptos y argumentaciones críticas iniciales, con una actitud de apertura al pensamiento libre y humildad epistémica.

Contenido

- 1.1. Experiencia, percepción y pregunta
 - 1.1.1. Conocer desde lo humano: cuerpo, sentidos, mundo vivido.
 - 1.1.2. Asombro, curiosidad e ignorancia fértil como motores del pensar.
 - 1.1.3. La pregunta como gesto originario del conocimiento.
- 1.2. Del mito al logos
 - 1.2.1. Explicaciones míticas y su función humana.
 - 1.2.2. Emergencia de la explicación racional del mundo (logos).
 - 1.2.3. La physis como objeto de interrogación.
- 1.3. Pensamiento presocrático (núcleo)
 - 1.3.1. Milesios y búsqueda del arché: Tales, Anaximandro, Anaxímenes.
 - 1.3.2. Devenir y permanencia: Heráclito y Parménides.
 - 1.3.3. Pluralistas y atomistas: Empédocles, Anaxágoras, Leucipo y Demócrito.
- 1.4. Actitud filosófica inicial y puente hacia la ciencia
 - 1.4.1. Duda, diálogo y pensamiento libre.
 - 1.4.2. Límites del saber y humildad epistémica.
 - 1.4.3. De la reflexión filosófica al nacimiento de lo científico.

Duración: 8 horas

Unidad II. De la filosofía natural a la ciencia moderna: cómo nace lo científico

Competencia

Analizar la transición histórica y conceptual de la filosofía natural a la ciencia moderna, mediante la comprensión de los principales quiebres epistemológicos y metodológicos que consolidan el conocimiento científico, para reconocer cómo se construyen leyes, modelos y procedimientos experimentales, desarrollando una mirada crítica sobre sus supuestos, alcances y límites.

Contenido

- 2.1. Filosofía natural y síntesis clásica (puente necesario)
 - 2.1.1. Aristóteles y la explicación de la naturaleza: causas, movimiento, cosmos.
 - 2.1.2. Límites de la filosofía natural para explicar ciertos fenómenos.
- 2.2. Ruptura renacentista y nueva mirada del universo
 - 2.2.1. Copérnico y el descentramiento cosmológico.
 - 2.2.2. Kepler y la matematización del movimiento celeste.
 - 2.2.3. Galileo: experimento, idealización y lenguaje matemático.
- 2.3. Nacimiento del método científico moderno
 - 2.3.1. Bacon: experiencia organizada e inducción.
 - 2.3.2. Descartes: duda metódica y racionalismo.
 - 2.3.3. Tensión empirismo–racionalismo como motor histórico de la ciencia.
- 2.4. Newton y la consolidación de la ciencia moderna

- 2.4.1. Ley científica, modelo e idealización.
- 2.4.2. Matematización de la naturaleza y alcance de la explicación newtoniana.
- 2.5. Ilustración y circulación del pensamiento científico
 - 2.5.1. La ciencia como empresa pública: difusión, traducción y enseñanza.
 - 2.5.2. Presencia femenina en la recepción newtoniana: Émilie du Châtelet y Maria Gaetana Agnesi como agentes de traducción, clarificación y enseñanza del marco moderno.
- 2.6. Alcances y límites del conocimiento moderno
 - 2.6.1. Qué permite el método científico y qué deja fuera.
 - 2.6.2. Condiciones históricas y humanas de la objetividad científica.

Duración: 8 horas

Unidad III. Herramientas fundamentales del pensamiento crítico científico

Aplicar herramientas esenciales del pensamiento crítico y creativo en contextos científicos y cotidianos, mediante la construcción, análisis y evaluación de argumentos, el reconocimiento de falacias y sesgos, y el uso de la imaginación científica en la exploración de escenarios posibles, con el fin de comprender, cuestionar y comunicar explicaciones físicas con rigor conceptual y sentido situado.

Contenido

- 3.1. Fundamentos operativos del pensamiento crítico en ciencia
 - 3.1.1. Qué significa pensar críticamente en Física: comprender, cuestionar, justificar.
 - 3.1.2. Duda, contraste y revisión como prácticas científicas.
- 3.2. Argumentación y lógica mínima
 - 3.2.1. Estructura de un argumento: premisas, conclusiones e inferencias.
 - 3.2.2. Validez, coherencia y fuerza explicativa.
 - 3.2.3. Diferencia entre explicar, describir y persuadir.
- 3.3. Falacias, sesgos y límites del razonamiento
 - 3.3.1. Falacias frecuentes en ciencia y vida cotidiana.
 - 3.3.2. Sesgos cognitivos y su impacto en la construcción de conocimiento.
 - 3.3.3. Criterios básicos para reconocer errores de razonamiento.
- 3.4. Imaginación científica y escenarios contrafácticos
 - 3.4.1. El “¿qué pasaría si...?” como herramienta de descubrimiento.
 - 3.4.2. Modelos posibles, mundos alternativos y consecuencias físicas.
 - 3.4.3. Creatividad rigurosa: imaginar sin perder consistencia lógica.
- 3.5. Alfabetización científica crítica

- 3.5.1. Lectura crítica de divulgación y medios científicos.
- 3.5.2. Identificación de supuestos, datos, interpretaciones y conclusiones.
- 3.5.3. Comunicación clara de argumentos y hallazgos.

Duración: 8 horas

Unidad IV. Ciencia, sentido y responsabilidad: una epistemología sensible

Competencia

Integrar una comprensión situada de la ciencia como práctica humana, histórica y social, mediante el reconocimiento de sus implicaciones éticas, sus límites y su diálogo posible con otros saberes, para construir una postura epistemológica personal y responsable que vincule la formación en Física con el sentido de vida, la vocación científica y el cuidado del mundo común.

Contenido

- 4.1. Ciencia como práctica humana situada
 - 4.1.1. La ciencia en contexto: historia, cultura y condiciones materiales.
 - 4.1.2. Objetividad, valores y límites del conocimiento científico.
 - 4.1.3. Responsabilidad social de la ciencia.
- 4.2. Pluralismo epistémico y humildad del saber
 - 4.2.1. Ciencia y otros modos de conocer: diálogo sin jerarquías.
 - 4.2.2. Riesgos de la soberbia epistémica y del cientificismo.
 - 4.2.3. Conocimiento, poder y cuidado de la vida.
- 4.3. Ciencia, literatura y arte como formas humanas de comprensión
 - 4.3.1. Imaginación, lenguaje y metáfora en la construcción de conocimiento.
 - 4.3.2. Extrañamiento e ignorancia fértil como motores de pensamiento.
 - 4.3.3. Cruces creativos para comprender lo real.
- 4.4. Vocación científica, proyecto vital y anagnórisis final
 - 4.4.1. Autenticidad, deseo y sentido en la formación científica.
 - 4.4.2. Autobiografía científica: ¿qué quiero hacer con lo que sé?
 - 4.4.3. Síntesis de postura epistemológica personal.

Duración: 8 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

Unidad I. El asombro y los orígenes del conocer

Práctica 1. Asombro y extrañamiento: ver como si fuera la primera vez

Duración:

2 horas

Procedimiento

1. Elige un objeto cotidiano de su entorno cercano (reloj, taza, puerta, escalera, cuaderno u otro).
2. Observa en silencio durante 3–5 minutos como si nunca lo hubiera visto.
3. Describe por escrito su forma, textura, peso, función y lo que percibe con sus sentidos.
4. Formula al menos cinco preguntas “inocentes” sobre el objeto (qué es, por qué es así, cómo funciona, qué pasaría si...).
5. Comparte con su equipo una descripción y una pregunta, y escucha las de lxs demás.
6. Distingue, en diálogo grupal, explicaciones míticas espontáneas y explicaciones racionales emergentes.
7. Elabora e integra el registro de observación, preguntas y reflexión breve en el portafolio de evidencias.

Recursos de apoyo

- Objeto cotidiano elegido.
- Bitácora o libreta de trabajo.
- Notas de clase sobre mito, logos y asombro.

Práctica 2. Ponerse en la piel de lxs primerxs pensadorxs: medir sin instrumentos

Duración:

2 horas

Procedimiento

1. Se organiza en equipo y acepta el reto de medir una distancia y un intervalo de tiempo sin regla ni cronómetro.
2. Diseña un procedimiento propio de medición usando cuerpo, entorno y creatividad.
3. Establece una unidad inventada y define cómo registrará los datos.
4. Realiza la medición en un espacio abierto o pasillo amplio.

5. Registra resultados y observaciones sobre dificultades, ajustes y supuestos del procedimiento.
6. Compara resultados con otros equipos e identifica coincidencias y diferencias.
7. Discute qué límites del saber aparecieron y qué idea de orden o principio de la naturaleza emergió.
8. Elabora e integra reporte breve del procedimiento, resultados y reflexión crítica en el portafolio de evidencias.

Recursos de apoyo

- Espacio abierto o pasillo amplio.
- Materiales simples opcionales (cuerda, tiza, piedras).
- Bitácora o libreta de trabajo.
- Notas de clase sobre presocráticos, arché y physis.

Práctica 3. Anagnórisis: reconocerse en el acto de conocer

Duración:

2 horas

Procedimiento

1. Lee un texto breve detonador sobre asombro y origen del pensamiento racional en los presocráticos.
2. Escribe en su bitácora una narración personal sobre su primera experiencia de hambre de conocer.
3. Identifica las preguntas que lo trajeron a la Física y las ordena para reconocer el hilo que alimenta su curiosidad.
4. Expresa qué parte esencial de su persona reconoce cuando indaga y cómo se siente en el cuerpo ese impulso.
5. Comparte de manera voluntaria una idea en círculo de diálogo y escucha sin juzgar a los demás.
6. Conecta su experiencia con un concepto presocrático (physis, arché, logos, devenir o permanencia).
7. Elabora e integra el texto “Mi anagnórisis científica” con su conexión filosófica en el portafolio de evidencias.

Recursos de apoyo

- Texto breve seleccionado por el docente.
- Bitácora o libreta de trabajo.
- Notas de clase sobre presocráticos y actitud filosófica inicial.

Práctica 4. Línea viva del pensar presocrático: leer, situar, reconocer

Duración:

2 horas

Procedimiento

1. Lee un fragmento breve seleccionado por el docente de un presocrático (o una reconstrucción comentada) relacionado con *arché*, *physis* o *logos*.
2. Subraya dos ideas que te llamen la atención y una pregunta que e despierte asombro.
3. Identifica a qué problema filosófico responde el texto (principio de lo real, cambio/permanencia, materia/vacío).
4. Ubica al autor o corriente en una línea de tiempo simple presocrática (milesios → Heráclito/Parménides → pluralistas/atomistas).
5. Explica en 5–6 líneas cómo esa idea podría ser un antecedente de una pregunta científica actual en Física.
6. Comparte su línea y su conexión con el equipo y escucha las de lxs demás.
7. Elabora e integra una síntesis breve (1 cuartilla) con su línea de tiempo, ideas clave y reflexión en el portafolio de evidencias.

Recursos de apoyo

- Fragmento breve presocrático seleccionado por el docente.
- Bitácora o libreta de trabajo.
- Notas de clase sobre presocráticos, *arché*, *physis*, *logos*.

Unidad II. De la filosofía natural a la ciencia moderna: cómo nace lo científico

Práctica 5. Ruptura renacentista: mirar el cielo con ojos nuevos

Duración:

2 horas

Procedimiento

1. Observa un fenómeno astronómico simple o una representación accesible (movimiento aparente del Sol, fases de la Luna, trayectoria de una estrella, simulación básica).
2. Registra en tu bitácora lo observado mediante un dibujo y una descripción breve cuidando detalles de forma, ritmo y posición.
3. Identifica qué explicación “natural” o intuitiva surge primero (tipo geocéntrica/mítica/cotidiana).

4. Replantea la observación desde una explicación racional alternativa inspirada en Copérnico y Kepler (descentramiento, órbitas, regularidades).
5. Formula dos preguntas que muestren el quiebre entre ambas miradas.
6. Comparte con tu equipo tu registro y discute cómo cambia el mundo cuando cambia el marco explicativo.
7. Elabora e integra una síntesis breve (1 cuartilla) sobre la ruptura renacentista en el portafolio de evidencias.

Recursos de apoyo

- Bitácora o libreta de trabajo.
- Notas de clase sobre Copérnico y Kepler.
- Material visual simple seleccionado por el docente (cielo observable o simulación).

Práctica 6. Galileo y el nacimiento del experimento: diseñar un método situado

Duración:

2 horas

Procedimiento

1. Identifica un fenómeno de movimiento cotidiano en el entorno (rodar una pelota, deslizar un objeto, caída controlada, péndulo simple).
2. Plantea una pregunta investigable inspirada en Galileo (por ejemplo: ¿de qué depende el tiempo de caída?, ¿cómo cambia la velocidad?).
3. Diseña un procedimiento experimental con recursos simples, especificando pasos, variables a controlar y forma de registro.
4. Realiza el experimento en aula o exterior según aplique.
5. Registra datos y observaciones cualitativas sobre idealizaciones o fuentes de error.
6. Interpreta sus resultados señalando qué parte es “hecho observado” y qué parte es “modelo/idealización”.
7. Compara con otro equipo y discute semejanzas y diferencias metodológicas.
8. Elabora e integra un reporte breve (procedimiento, datos, interpretación y reflexión) en el portafolio de evidencias.

Recursos de apoyo

- Espacio de aula o exterior.
- Materiales simples del entorno (pelota, rampa improvisada, cuerda, etc.).
- Bitácora o libreta de trabajo.
- Notas de clase sobre Galileo, experimento e idealización.

Práctica 7. Newton: leyes, modelos e idealización

Duración:

2 horas

Procedimiento

1. Elige una ley o principio físico básico trabajado en la carrera (por ejemplo, inercia, acción–reacción, gravitación, conservación).
2. Identifica el fenómeno real que busca describir y describe sus rasgos observables.
3. Reconoce las idealizaciones implicadas en la ley (p. ej., “sin fricción”, “masa puntual”, “campo uniforme”).
4. Construye un modelo simple del fenómeno señalando qué se incluye y qué se deja fuera.
5. Explica por escrito en qué sentido la ley “funciona” y en qué sentido tiene límites.
6. Identifica una posible dificultad conceptual en el enunciado o uso de la ley (inspirándote en la crítica de Mach), señalando qué supuesto queda oculto o qué concepto requiere revisión.
7. Reformula la ley o su interpretación en tus propias palabras para hacer explícitos esos supuestos o límites.
8. Discute en equipo la diferencia entre ley, modelo y realidad, y comparte tu dificultad/reformulación.
9. Elabora e integra una síntesis breve (1 cuartilla) sobre leyes, idealización y crítica conceptual en el portafolio de evidencias.

Recursos de apoyo.

- Notas de clase sobre Newton, ley científica y modelo.
- Bitácora o libreta de trabajo.
- Apuntes de asignaturas previas de Física.

Práctica 8. Ilustración y circulación del conocimiento: traducir para comprender

Duración:

2 horas

Procedimiento

1. Lee un fragmento breve seleccionado por el docente relacionado con Newton (texto original o explicación técnica).
2. Identifica los conceptos centrales y subraya los pasajes que resulten difíciles.
3. Reescribe el fragmento en lenguaje propio, buscando claridad sin perder rigor (traducción conceptual).

4. Crea un ejemplo, analogía o representación que haga comprensible la idea a alguien de primer semestre.
5. Reflexiona sobre qué se gana y qué se pierde al traducir conocimiento científico.
6. Relaciona tu experiencia con el papel de la Ilustración en la difusión científica y con figuras como Émilie du Châtelet y Maria Gaetana Agnesi.
7. Comparte tu traducción con el equipo y discute distintas formas de claridad científica.
8. Elabora e integra tu traducción, ejemplo y reflexión en el portafolio de evidencias.

Recursos de apoyo

- Fragmento breve seleccionado por el docente.
- Bitácora o libreta de trabajo.
- Notas de clase sobre Ilustración, traducción científica y recepción newtoniana.

Unidad III. Herramientas fundamentales del pensamiento crítico científico

Práctica 9. Clínica de argumentos: construir y evaluar explicaciones

Duración:

2 horas

Procedimiento

1. Elige un enunciado o explicación breve relacionada con Física (puede ser de clase, de divulgación o de conversación cotidiana).
2. Identifica la conclusión principal que sostiene el enunciado.
3. Extrae y escribe las premisas explícitas que apoyan esa conclusión.
4. Reconstruye las premisas implícitas necesarias para que el argumento tenga sentido.
5. Evalúa la validez del argumento revisando coherencia lógica e inferencias.
6. Detecta si falta evidencia o si hay saltos no justificados.
7. Reformula el argumento mejorándolo en claridad, rigor y justificación.
8. Comparte su versión con su equipo, contrasta mejoras y acuerdos.
9. Elabora e integra una síntesis breve (argumento original, análisis y reformulación) en el portafolio de evidencias.

Recursos de apoyo

- Notas de clase sobre argumento, premisa e inferencia.
- Texto breve o ejemplo seleccionado por el docente.
- Bitácora o libreta de trabajo.

Práctica 10. Falacias y sesgos: desmontar lo “irrefutable”

Duración:

2 horas

Procedimiento

1. Lee o escucha tres afirmaciones breves seleccionadas por el docente (científicas o cotidianas).
2. Identifica cuál le parece más convincente a primera vista y explica por qué.
3. Analiza cada afirmación buscando falacias comunes (autoridad, generalización apresurada, falsa causa, etc.).
4. Reconoce posibles sesgos cognitivos implicados (confirmación, disponibilidad, anclaje, etc.).
5. Señala qué partes del razonamiento son frágiles o no justificadas.
6. Reescribe cada afirmación de manera más cuidadosa y crítica, corrigiendo la falacia.
7. Comparte con su equipo un caso y escucha las correcciones de los demás.
8. Elabora e integra una tabla breve de falacia/sesgo detectado y corrección en el portafolio de evidencias.

Recursos de apoyo

- Guía breve de falacias y sesgos (proporcionada por el docente).
- Notas de clase sobre pensamiento crítico.
- Bitácora o libreta de trabajo.

Práctica 11. Escenarios contrafácticos: imaginar con rigor (“modo Asimov”)

Duración: 2 horas

Procedimiento

1. Elige un principio físico fundamental (gravitación, conservación de energía, segunda ley, etc.).
2. Modifica imaginativamente una condición del principio (por ejemplo: gravedad con otro valor, constante distinta, fricción nula universal).
3. Describe el escenario alternativo con claridad, indicando qué cambió y qué se mantiene.
4. Deduce al menos tres consecuencias físicas razonables del cambio propuesto.
5. Identifica una consecuencia inesperada o paradójica y la explora.
6. Explica qué enseña este mundo alternativo sobre el principio real.
7. Comparte su escenario con su equipo y contrasta consecuencias.
8. Elabora e integra un escrito breve con escenario, deducciones y reflexión crítica en el portafolio de evidencias.

Recursos de apoyo

- Notas de clase sobre el principio elegido y modelación.
- Bitácora o libreta de trabajo.
- Apuntes de Física.

Práctica 12. Alfabetización científica crítica: leer, distinguir, argumentar

Duración: 2 horas

Procedimiento

1. Lee un texto breve de divulgación científica o nota periodística seleccionada por el docente.
2. Identifica el tema central y la pregunta que el texto intenta responder.
3. Distingue datos/observaciones de interpretaciones/opiniones en el contenido.
4. Reconstruye el argumento principal del texto (premisas y conclusión).
5. Evalúa la fuerza del argumento señalando supuestos, alcances y límites.
6. Detecta posibles exageraciones, simplificaciones o sesgos en el lenguaje usado.
7. Redacta un comentario crítico breve que integre su evaluación y postura.
8. Comparte su comentario con su equipo y escucha retroalimentación.
9. Elabora e integra el texto analizado y su comentario crítico en el portafolio de evidencias.

Recursos de apoyo

- Texto breve seleccionado por el docente.
- Guía de lectura crítica proporcionada por el docente.
- Bitácora o libreta de trabajo.

Unidad IV. Ciencia, sentido y responsabilidad: una epistemología sensible

Práctica 13. Ciencia situada: teoría, contexto y consecuencias

Duración: 2 horas

Procedimiento

1. Lee un caso breve seleccionado por el docente sobre un desarrollo científico/tecnológico con impacto social (histórico o actual).
2. Identifica el problema científico central del caso y describe qué teoría o modelo lo sostiene.
3. Reconoce el contexto histórico, social o cultural en el que surge ese conocimiento.
4. Identifica decisiones metodológicas clave y explica por qué fueron tomadas.
5. Analiza dos consecuencias éticas o sociales derivadas del caso (positivas, negativas o ambiguas).

6. Discute en equipo cómo cambia la comprensión de la ciencia cuando se observa en su contexto.
7. Redacta una síntesis crítica breve integrando teoría, contexto y consecuencias.
8. Elabora e integra la síntesis en el portafolio de evidencias.

Recursos de apoyo

- Caso breve seleccionado por el docente.
- Notas de clase sobre ciencia situada y responsabilidad social.
- Bitácora o libreta de trabajo.

Práctica 14. Pluralismo epistémico: dialogar sin soberbia

Duración: 2 horas

Procedimiento

1. Lee dos textos breves seleccionados por el docente:
 - a) uno desde una perspectiva científica,
 - b) otro desde un saber no científico (comunitario, artístico, filosófico o histórico).
2. Identifica qué pregunta intenta responder cada texto y qué tipo de evidencia usa.
3. Reconoce los supuestos principales de cada perspectiva.
4. Señala en qué puntos se complementan y en qué puntos entran en tensión.
5. Redacta un argumento breve defendiendo un diálogo posible entre ambas perspectivas sin jerarquizarlas.
6. Comparte su argumento con su equipo y escucha retroalimentación.
7. Identifica qué aprendizajes ofrece el pluralismo para su formación como físicx.
8. Elabora e integra tu argumento y reflexión en el portafolio de evidencias.

Recursos de apoyo

- Dos textos breves seleccionados por el docente.
- Notas de clase sobre pluralismo de saberes y humildad epistémica.
- Bitácora o libreta de trabajo.

Práctica 15. Ciencia, literatura y arte: extrañar para comprender

Duración: 2 horas

Procedimiento

1. Lee un texto breve literario seleccionado por el docente (por ejemplo, un fragmento tipo “instrucciones” o extrañamiento cotidiano).

2. Identifica qué hace el texto para volver extraño lo familiar (lenguaje, ritmo, detalle, mirada).
3. Elige un fenómeno físico cotidiano (luz en una ventana, sonido en un pasillo, sombra, caída de un objeto).
4. Describe el fenómeno “como si lo vieras por primera vez”, usando recursos literarios simples (metáfora, comparación, detalle sensorial).
5. Conecta esa descripción con una idea científica: qué pregunta física despierta, qué modelo podría explicarla.
6. Comparte tu texto con tu equipo y discute cómo la imaginación abre conocimiento.
7. Redacta una reflexión breve sobre el valor del arte/literatura para comprender el mundo científicamente.
8. Elabora e integra descripción y reflexión en el portafolio de evidencias.

Recursos de apoyo

- Texto literario breve seleccionado por el docente.
- Notas de clase sobre imaginación, lenguaje y conocimiento científico.
- Bitácora o libreta de trabajo.

Práctica 16. Autobiografía científica y anagnórisis final

Duración: 2 horas

Procedimiento

1. Recupera sus bitácoras, textos y prácticas del semestre.
2. Identifica tres momentos del curso que hayan transformado su forma de pensar o sentir la ciencia.
3. Escribe una autobiografía científica breve que responda:
 - a) quién soy cuando conozco,
 - b) qué preguntas me habitan,
 - c) qué deseo hacer con lo que aprendo en Física.
4. Explicita su postura epistemológica personal conectándola con al menos dos conceptos del curso (presocráticos, ciencia moderna, pensamiento crítico, ciencia situada, pluralismo).
5. Define un compromiso ético o vital que desea sostener como futurx científicx.
6. Comparte voluntariamente una idea central en círculo de cierre y escucha a lxs demás.
7. Ajusta tu texto con la retroalimentación recibida.
8. Elabora e integra la autobiografía y postura final en el portafolio de evidencias como síntesis del curso.

Recursos de apoyo

- Portafolio/bitácora del semestre.
- Notas de clase generales del curso.
- Guía breve de autobiografía científica proporcionada por el docente.

IX. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Seminario–taller (alternancia entre análisis conceptual y prácticas encarnadas)
- Método de proyectos (portafolio integrador y síntesis final)
- Aprendizaje basado en problemas (planteamiento y resolución de preguntas científicas)
- Aprendizaje basado en casos (ciencia situada: teoría, contexto y consecuencias)
- Aprendizaje basado en preguntas (asombro, duda razonada y “preguntas inocentes”)
- Técnica expositiva dialogada (breve, con participación activa)
- Debates
- Discusión guiada y círculos de diálogo

Estrategia de aprendizaje

- Elaboración de portafolio/bitácora (asombro, prácticas, lecturas críticas, anagnórisis)
- Resolución y exposición de casos y problemas prácticos
- Trabajo colaborativo
- Aprendizaje experiencial/encarnado (observación, medición sin instrumentos, experimentos situados)
- Técnica expositiva del estudiantado (presentaciones breves, socialización de hallazgos)
- Técnica reflexiva de aprendizaje (escritura crítica y autobiografía científica)

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

- Evaluaciones parciales	30%
- Prácticas taller	20%
- Trabajo en el aula	15%
- Examen final	20%
- Portafolio de evidencia	15%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Aristóteles. (2015). *Metafísica* (Alberto Rodríguez López, Trad.). Editores Mexicanos Unidos. [clásica]

Descartes, R. (1974). *Discurso del método* (Rovira Armengol, Trad.). Losada. [clásica]

Kuhn, T. S. (2004). *The structure of scientific revolutions* (4.^a ed.). Fondo de Cultura Económica. [clásica]

Ziman, J. (2000). *Real science: What it is, and what it means*. Cambridge University Press. [clásica]

Gaarder, J. (1994/1995). *El mundo de Sofía*. Siruela (o edición equivalente en español). [clásica]

Julio Cortázar (2006). *Historias de cronopios y de famas*. Alfaguara. [clásica]

Recursos de acceso libre o complementarios externos

Bacon, F. (2000). *Novum organum* (J. Pérez, Trad.). Tecnos. [clásica]

Popper, K. (2002). *The logic of scientific discovery*. Routledge. [clásica]

Ziman, J. (2000). *Real science: What it is, and what it means*. Cambridge University Press. [clásica]

Asimov, I. (1941). *Anochecer (Nightfall)*. En *Nightfall and Other Stories / Anochecer y otros relatos* (edición en español equivalente). [clásica]

Feinmann, J. P. (s. f.). *Filosofía aquí y ahora* [Canal de YouTube]. YouTube. Recuperado el 8 de diciembre de 2025, de:

<https://www.youtube.com/channel/UCB725gZ4c8vs7JY7wAl5A7A>

Sagan, C. (1980). *Cosmos: A Personal Voyage* [Serie documental]. PBS / Canal Encuentro / disponible en YouTube (selección de episodios por el docente).

[https://www.youtube.com/results?search_query=Sagan%2C+C.+\(1980\).+Cosmos+%3A+A+Personal+Voyage](https://www.youtube.com/results?search_query=Sagan%2C+C.+(1980).+Cosmos+%3A+A+Personal+Voyage) [clásica]

UNED. (2021). *Historia y Filosofía de las Matemáticas* [Video]. Canal UNED. Recuperado el 8 de diciembre de 2025, de: <https://canal.uned.es/>

XII. PERFIL DOCENTE

Licenciatura en Ciencias o Filosofía, preferentemente con posgrado en ciencias, humanidades o filosofía, con experiencia en epistemología, historia de la ciencia, comunicación de la ciencia o análisis crítico, experiencia docente de dos años, ser una persona capaz de articular contenidos interdisciplinarios y promover reflexión ética y argumentativa en el estudiantado.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Comunicación para la Divulgación de las Ciencias
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 02 Horas Laboratorio: 00 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:06
Etapas de formación:	Básica
Carácter:	Optativa
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Alma Rocío Cabazos Marín, María Victoria Meza Kubo
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vazquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

En esta unidad de aprendizaje el estudiantado desarrollará habilidades verbales, escritas y visuales, identificando y aplicando destrezas necesarias para la redacción de ensayos y exposición de discursos de divulgación, que le permitan un mejor desempeño profesional.

La divulgación científica es un puente entre el conocimiento especializado y la sociedad. En este curso, el estudiantado desarrollará competencias comunicativas que les permitan transmitir conceptos científicos complejos de manera clara, precisa y atractiva, tanto para públicos especializados como no especializados.

La unidad de aprendizaje se encuentra ubicada en la etapa básica de la Licenciatura en Ciencias Computacionales. Aporta elementos requeridos para las unidades de aprendizaje del perfil profesional que requieran la elaboración de reportes de proyectos, prácticas de laboratorio, ensayos, entre otros, así como, la exposición oral de los mismos. La unidad de aprendizaje es de carácter optativo y pertenece al área de conocimiento de entorno social.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Desarrollar la capacidad de comunicar el conocimiento científico a diferentes sectores de la sociedad de forma efectiva, ética y creativa en distintos medios y formatos, por medio de la expresión oral, escrita, visual y gestual para divulgar y democratizar la ciencia en la sociedad con respeto, empatía y creatividad.

IV. EVIDENCIA DE APRENDIZAJE

- Portafolio de reportes de las prácticas del taller.
- Proyecto expositivo de divulgación científica en eventos dirigidos a la comunidad.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Compromiso con la responsabilidad social, equidad social y de género, inclusión, excelencia, vanguardia, innovación social, interculturalidad.

VII. DESARROLLO POR UNIDADES

Unidad I. Expresión lógica y clara

Competencia

Distinguir la comunicación técnica o periodística, para reconocer las características del lenguaje científico y adaptar su discurso según el público y el medio, mediante el análisis de ejemplos de divulgación científica, con pensamiento crítico y compromiso.

Contenido

- 1.1. Conceptos básicos: ciencia, comunicación y divulgación
- 1.2. Diferencias entre comunicación científica, técnica y periodística
- 1.3. Historia y evolución de la divulgación científica
- 1.4. La divulgación como parte de la cultura científica y la alfabetización científica
- 1.5. Ética, responsabilidad y veracidad en la comunicación de la ciencia

Duración

6 horas

Unidad II. Técnicas de expresión oral

Competencia

Aplicar recursos visuales, gestuales y narrativos para explicar conceptos científicos complejos mediante técnicas básicas de comunicación oral como exposiciones o cápsulas de divulgación hacia una audiencia, con creatividad y respeto.

Contenido

- 2.1 Habilidades del comunicador eficaz
 - 2.1.1 Canalizar el nerviosismo
 - 2.1.2 Credibilidad
 - 2.1.3 El arte de escuchar/actitud receptiva
 - 2.1.4 Tipos de actitudes
- 2.2 El discurso
 - 2.2.1 Elaboración de un discurso
 - 2.2.2 Introducción de un discurso
 - 2.2.3 Conclusión de un discurso

- 2.2.4 Análisis del público
- 2.3 Tipos de discurso
 - 2.3.1 Social
 - 2.3.2 Informativo
 - 2.3.3 Motivación
- 2.4 Técnicas efectivas de exposición
 - 2.4.1 Desarrolle y utilice medios visuales
 - 2.4.2 Lenguaje y medios para presentaciones

Duración

8 horas

Unidad III. Producción de contenidos de divulgación

Competencia

Desarrollar textos de divulgación sobre temas de su disciplina con claridad, estructura y pertinencia mediante técnicas básicas de comunicación oral y Técnicas de la Información y la Comunicación (TIC) para explicar conceptos científicos complejos, con respeto, creatividad y objetividad.

Contenido

- 3.1. Estructura y redacción de textos divulgativos
- 3.2. Técnicas de exposición oral y storytelling científico
- 3.3. Diseño de recursos gráficos: infografías, carteles, presentaciones visuales
- 3.4. Comunicación digital: blogs, videos, podcasts, redes sociales
- 3.5. Adaptación de contenido a distintos formatos y públicos

Duración

8 horas

Unidad IV. Producción de contenidos de divulgación

Competencia

Desarrollar un proyecto de divulgación científica orientado a un público específico como campaña, video, exposición interactiva, blog o podcast, mediante técnicas de comunicación oral y TIC, para comunicar y democratizar la ciencia, con actitud propositiva, creatividad y respeto.

Contenido

- 4.1. Elección del tema y definición del público objetivo
- 4.2. Planeación y guión del producto comunicativo
- 4.3. Producción, edición y difusión del producto
- 4.4. Evaluación del impacto y retroalimentación
- 4.5. Presentación del proyecto final

Duración

10 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

Unidad I. Expresión lógica y clara

Práctica 1: Análisis comparativo de géneros comunicativos

Procedimiento

1. Atiende a la explicación docente sobre las diferencias estructurales entre la comunicación científica (paper), técnica (manual) y periodística (noticia).
2. En equipo, analiza tres textos proporcionados por el docente que traten el mismo tema (ej. "Computación Cuántica") pero en formatos distintos: un resumen de IEEE Xplore, una nota de un diario nacional y un manual de usuario.
3. Completa una matriz comparativa donde identifiques para cada texto: público objetivo, nivel de tecnicismos, tono y objetivo de la comunicación.
4. Discute en clase: ¿Qué elementos del texto científico se perdieron en la nota periodística? ¿Se mantiene la veracidad?
5. Presenta una conclusión grupal sobre la importancia de la alfabetización científica.
6. Recibe retroalimentación docente y del grupo.

Duración

03 horas

Recursos de apoyo

- Set de lecturas (Científica, Técnica y Periodística).
- Cuadro comparativo impreso o digital.
- Pizarrón y plumones.

Práctica 2: Adaptación de discurso y ética en la información

Procedimiento

1. Escucha la exposición sobre la ética y responsabilidad en la divulgación: el peligro de las *fake news* científicas y el sensacionalismo.
2. Selecciona un descubrimiento reciente en Ciencias Computacionales (fuente técnica).

3. Realiza un ejercicio de "adaptación de discurso": redacta un hilo de redes sociales (X o Instagram) que explique el descubrimiento de forma atractiva pero ética, evitando el "clickbait" engañoso.
4. Intercambia tu redacción con otro equipo para una "auditoría ética": el otro equipo debe señalar si simplificaste demasiado el concepto hasta volverlo falso o si mantuviste la precisión.
5. Reflexiona por escrito sobre cómo la divulgación contribuye a la cultura científica de la sociedad actual.
6. Recibe retroalimentación del docente sobre el equilibrio entre claridad y veracidad.

Duración

03 horas

Recursos de apoyo

- Computadoras con acceso a internet.
- Ejemplos de noticias científicas sensacionalistas vs. veraces.
- Plataforma de trabajo colaborativo.

Unidad II. Técnicas de expresión oral

Práctica 3: El comunicador frente al espejo: Manejo de actitud y análisis del público

Duración

04 horas

Procedimiento

1. Atiende a la explicación del docente sobre las técnicas para canalizar el nerviosismo y la construcción de credibilidad.
2. En parejas, realiza un ejercicio de escucha receptiva donde uno explique un concepto de computación y el otro identifique barreras actitudinales en la comunicación.
3. En equipo, selecciona un tema de ciencia y analiza tres perfiles de público distintos (niños, especialistas y público general).
4. Adapta el mensaje central del tema elegido según las características y necesidades de cada perfil de audiencia identificado.
5. Realiza una breve simulación de "actitud frente a grupo" aplicando el lenguaje corporal adecuado para el perfil asignado.
6. Recibe retroalimentación del docente y del grupo sobre la coherencia entre el discurso y la actitud mostrada.

Recursos de apoyo

- Hojas
- Lápiz
- Plumones

Práctica 4: Estructura del discurso y medios visuales de apoyo

Duración

04 horas

Procedimiento

1. Escucha la exposición docente sobre la estructura de los discursos informativos y motivacionales, enfatizando en introducciones y conclusiones de impacto.
2. Elabora el guion de un discurso breve sobre una innovación tecnológica, siguiendo la estructura: introducción (gancho), desarrollo y conclusión.
3. Diseña un medio visual (presentación digital o infografía) que sirva de apoyo al discurso, cuidando el equilibrio entre imagen y texto.
4. Expone el discurso frente al grupo utilizando el medio visual diseñado y aplicando técnicas de exposición efectiva.
5. Graba un fragmento de la participación para analizar posteriormente la gesticulación y el uso de los apoyos visuales.
6. Participa en una sesión de preguntas y respuestas para demostrar dominio del tema y capacidad de respuesta.
7. Recibe retroalimentación docente sobre la calidad de los recursos visuales y la fluidez narrativa.

Recursos de apoyo

- Computadora.
- Proyector.
- Software de presentaciones (Canva/PowerPoint).
- Acceso a internet.

Unidad III. Producción de contenidos de divulgación

Práctica 5: Narrativa científica y diseño de recursos gráficos

Duración

04 horas

Procedimiento

1. Atiende a la explicación docente sobre la estructura de los textos divulgativos y la técnica de *storytelling* (narrativa) aplicada a la ciencia.
2. Selecciona un concepto complejo de las Ciencias Computacionales (ej. Redes Neuronales o Blockchain) para convertirlo en una historia breve o caso de uso.
3. Elabora un borrador de texto divulgativo aplicando principios de claridad, objetividad y una estructura que incluya: gancho narrativo, explicación del concepto y conclusión.
4. Diseña un recurso gráfico complementario (infografía o cartel digital) que sintetice visualmente los puntos clave de la historia creada.
5. Presenta el diseño gráfico al grupo para evaluar si la jerarquía visual y el lenguaje facilitan la comprensión del tema sin necesidad de explicaciones verbales extensas.

6. Recibe retroalimentación del docente y del grupo sobre la creatividad y pertinencia del contenido.

Recursos de apoyo

- Computadora con software de diseño (Canva, Piktochart).
- Acceso a bancos de imágenes libres de derechos.
- Ejemplos de infografías científicas.

Práctica 6: Adaptación de contenidos para medios digitales

Duración

04 horas

Procedimiento

1. Escucha la exposición sobre las características de los distintos formatos digitales: blogs (texto largo), redes sociales (microcontenidos), video y podcasts.
2. En equipo, selecciona el texto desarrollado en la práctica anterior para adaptarlo a dos formatos digitales distintos (ej. un guion para video de TikTok/Reel y una entrada para blog).
3. Realiza la adaptación del contenido cuidando el tono, el lenguaje y la duración específica requerida para cada medio seleccionado.
4. Graba un clip de video o audio (podcast) de un minuto donde se apliquen las técnicas de comunicación oral y narrativa científica revisadas en clase.
5. Comparte los productos digitales en una plataforma común de la clase para realizar una sesión de coevaluación basada en la objetividad y el respeto.
6. Analiza en grupo cómo cambia la estructura del mensaje al pasar de un formato visual estático a uno digital dinámico.
7. Recibe retroalimentación docente sobre la efectividad de la adaptación del contenido a los diferentes públicos digitales.

Recursos de apoyo

- Smartphones o cámaras para grabación.
- Micrófonos (opcional).
- Software de edición básico.
- Conexión a internet.

Unidad IV: Producción de contenidos de divulgación

Práctica 7: Planeación estratégica y guionismo técnico

Duración

05 horas

Procedimiento

1. Atiende a la explicación del docente sobre los pasos para la planeación de proyectos de comunicación y la estructura de escaletas para medios digitales.
2. Selecciona un tema final de investigación y define el perfil del público objetivo utilizando una ficha de segmentación (edad, intereses, nivel educativo).
3. Elabora una tabla de planeación que incluya el objetivo del mensaje, el formato elegido (video, podcast, blog o campaña) y los canales de difusión.
4. Redacta el guion técnico o escaleta del producto, detallando los tiempos, el lenguaje a utilizar y los recursos visuales o sonoros necesarios.
5. Presenta el guion en una sesión de "picheo" ante el grupo para validar si la narrativa es coherente con el público seleccionado.
6. Recibe retroalimentación docente sobre la viabilidad técnica y narrativa del guion antes de iniciar la producción.

Recursos de apoyo

- Formatos de guion y escaleta.
- Pizarrón y plumones.
- Computadora con acceso a internet.

Práctica 8: Producción, montaje y presentación del proyecto final

Duración

05 horas

Procedimiento

1. Ejecuta la fase de producción (grabación, redacción o diseño final) siguiendo estrictamente el guion aprobado en la práctica anterior.
2. Realiza la edición del material utilizando herramientas de TIC para asegurar una calidad profesional en el audio, video o diseño gráfico.
3. Implementa una estrategia de difusión simulada o real en redes sociales o plataformas institucionales para captar atención del público.
4. Expone el proyecto final ante la clase, explicando el proceso de creación y cómo se democratizó el conocimiento científico a través del producto.
5. Evalúa el impacto potencial del proyecto mediante una rúbrica de retroalimentación compartida con los compañeros.
6. Entrega el portafolio final que integre todos los reportes de taller realizados durante el semestre.
7. Recibe la evaluación final del docente basada en la creatividad, respeto y objetividad del contenido producido.

Recursos de apoyo

- Software de edición (CapCut, Audacity, Canva).
- Smartphones o equipo de grabación.
- Proyector y sistema de audio.

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Aprendizaje basado en proyectos
- Técnica expositiva
- Discusión guiada
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Trabajo colaborativo

- Técnica expositiva
- Cuadros comparativos
- Infografías
- Aprendizaje basado en proyectos

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	25%
– Portafolio de Reportes de taller	25%
– Tareas y Actividades	20%
– Proyecto expositivo de divulgación	30%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Alcaraz Flores, M. P., & Medina Zavala, C. (Coords.). (2019). *Ventanas a la divulgación científica: Miradas desde el diseño, la comunicación y las artes* [Libro electrónico]. Universidad Autónoma de Baja California. <https://research.ebsco.com/linkprocessor/v2-external?opid=vfl6sr&recordId=mrb725fhxb&url=https%3A%2F%2Fdrive.google.com%2Ffile%2F0B%2F1sIPpVUaOqV-cMbJ39uUgdqaoVCUO5zCA%2Fview%3Fusp%3Dsharing>

Fonseca Yerena, M. S. (2016). *Comunicación oral y escrita* (2.^a ed.). Pearson Hispanoamérica Contenido. <https://libcon.rec.uabc.mx:2185/9786073236881>

Escarpanter, J. (2010). *La comunicación oral*. Firms Press. <https://libcon.rec.uabc.mx:2246/es/ereader/uabc/36451>

Verderber, R. F., Verderber, K. S., Sellnow, D. D., Fernández Pereda, J. A., Lama Noriega, M. A. de la, & Soto, J. A. (2016). *Comunicación oral efectiva en la era digital* / (Decimosexta edición).

Cengage Learning. <https://libcon.rec.uabc.mx:2258/visorBook.aspx?i=1985&t=8FD3A771-E764-41B6-AE6C-21405CD70B1E&p=1>

Chávez Pérez, F. (2016). Redacción avanzada: Un enfoque lingüístico (5.ª ed.). Pearson Hispanoamérica Contenido. <https://libcon.rec.uabc.mx:2185/9786073238175>

Recursos de acceso libre o complementarios externos

Montes, S., & Navarro, F. (2019). Hablar, persuadir, aprender: manual para la comunicación oral en contextos académicos. Universidad de Chile.

<https://repositorio.uchile.cl/bitstream/handle/2250/175330/Hablar-persuadir-aprender.pdf?sequence=1&isAllowed=y>

Olson, R. (2015). Houston, we have a narrative: Why science needs story. In Houston, we have a narrative. University of Chicago Press.

AAAS - Communication Toolkit: <https://www.aaas.org/resources/communication-toolkit>. Guías gratuitas para científicos sobre cómo hablar con el público y medios.

Guerrero Jiménez, G. (2010). Expresión oral y escrita (1.ª ed.). Universidad Técnica Particular de Loja. <https://galoquerrerojimenez.com/wp-content/uploads/2020/06/Expresion-Oral-y-Escrita-Libro-Base.pdf>

XII. PERFIL DOCENTE

Licenciado en áreas de la ciencias como Ciencias Computacionales, Física, Matemáticas, Biología o área afín. Preferentemente con estudios de doctorado en Ciencias. Profesionista con conocimientos y habilidades para la comunicación oral y escrita de artículos de divulgación e investigación. Con al menos dos años de experiencia docente. Ser una persona creativa, propositiva, que fomente la participación y el trabajo en equipo, persona ética que sea sensible a los valores de inclusión y respeto.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	
Nombre de la unidad de aprendizaje:	Paradigmas y Lenguajes de Programación
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 02 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:06
Etapas de formación:	Básica
Carácter:	Optativa
Requisito:	Ninguno.
Modalidad de impartición:	Presencial
Equipo de diseño:	Adrián Enciso Almanza María Victoria Meza Kubo
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

El propósito de la asignatura de Lenguajes de programación es que el estudiantado comprenda y aplique los paradigmas de programación más importantes, integrando fundamentos avanzados de estructuras de datos, programación orientada a objetos, funcional y multiparadigma. A través de proyectos prácticos, interdisciplinarios y documentados profesionalmente, desarrollará pensamiento crítico, autonomía y capacidad de innovación, seleccionando el enfoque más adecuado según el problema a resolver en contextos científicos y tecnológicos.

La asignatura es optativa, ubicada en la etapa básica de la Licenciatura en Ciencias Computacionales, y puede ser cursada por estudiantes de áreas afines como biología, matemáticas, física y ciencia de datos. Pertenece al área de conocimiento de Programación e Ingeniería de Software.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Evaluar los diversos paradigmas y ambientes de programación a través del estudio y comparación de los lenguajes más actuales y utilizados en la programación de computadoras, para facilitar el proceso de desarrollo de un programa de acuerdo a su dominio de aplicación, con actitud crítica y propositiva.

IV. EVIDENCIA DE APRENDIZAJE

Trabajo final en el que evaluará al menos cuatro lenguajes de programación pertenecientes a diferentes paradigmas, mediante la solución de un problema común. El documento deberá incluir información sobre el origen, creador, genealogía, paradigma, propiedades básicas y características distintivas de cada lenguaje, así como una tabla comparativa de sus estructuras y diseño. Además, se integrará el código fuente desarrollado en cada lenguaje para resolver el problema, permitiendo analizar sus diferencias en la implementación.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Pensamiento crítico, ética digital y responsabilidad social, inclusión y colaboración.

VII. DESARROLLO POR UNIDADES

Unidad I. Estudio de los lenguajes de programación

Competencia

Analizar los atributos de un lenguaje de programación, mediante la definición y exposición de tópicos de interés y ejemplos, para clasificar a los lenguajes de acuerdo a su dominio de aplicación, con actitud crítica y reflexiva.

Contenido

- 1.1. La importancia de los lenguajes de programación
- 1.2. Breve historia de los lenguajes de programación
- 1.3. Atributos de un buen lenguaje
- 1.4. Dominios de aplicación
- 1.5. Estandarización de lenguajes
- 1.6. Entorno (ambiente) de programación

Duración

6 horas

Unidad II. Características de los lenguajes de programación

Competencia

Analizar las características de los lenguajes de programación, mediante el uso de materiales audiovisuales, para comprender la naturaleza de los lenguajes de programación y su estructura, con actitud positiva y responsabilidad.

Contenido

- 2.1. La estructura de una computadora
- 2.2. Traductores e intérpretes
- 2.3. Máquinas virtuales y tiempos de enlace
- 2.4. Cuestiones de sintaxis y semántica de los lenguajes
- 2.5. Los paradigmas de programación

Duración

08 horas

Unidad III. Implementación y diseño de los lenguajes de programación

Competencia

Explorar la implementación de los lenguajes de programación, mediante la definición de estructuras formales, tipos de datos, encapsulamiento y secuencia, para comprender la

relación entre un programa fuente y su comportamiento en la ejecución del programa, con actitud crítica y propositiva

Contenido

- 3.1. Sintaxis de los lenguajes de programación
- 3.2. Modelos formales
- 3.3. Tipos de datos
- 3.4. Encapsulamiento
- 3.5. Control de secuencia de subprogramas

Duración

08 horas

Unidad IV. Paradigmas y lenguajes

Competencia

Utilizar diversos lenguajes de programación a través de la aplicación de las estructuras y técnicas de programación estructuradas, orientada a objetos, lógicas y funciones para la definición y entendimientos de cada paradigma, con actitud crítica y responsable

Contenido

- 4.1. Lenguajes por procedimiento o estructurado
- 4.2. Lenguajes orientados a objetos
- 4.3. Lenguajes funcionales
- 4.4. Lenguajes de programación lógica
- 4.5. Lenguajes multiparadigmas

Duración

10 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Estudio de los lenguajes de programación

Práctica 1: Línea del tiempo de los lenguajes de programación

Duración: 02 horas

Procedimiento:

1. Atiende las orientaciones docentes para el desarrollo de la práctica.
2. Realiza una investigación en equipo sobre los principales lenguajes de programación y su año de aparición.
3. Identifica hitos relevantes en la evolución de los lenguajes.
4. Elabora una línea del tiempo (digital o física) que incluya al menos 10 lenguajes, indicando su paradigma y dominio de aplicación.
5. Presenta brevemente el trabajo ante el grupo.
6. Participa en la retroalimentación grupal y recibe comentarios de su docente.

Recursos de apoyo:

- Sitios web como GitHub History, Wikipedia, artículos especializados
- Canva, Miro, Genially o herramientas similares para crear la línea del tiempo
- Lectura: Historia de los lenguajes de programación

Práctica 2: Clasificación de lenguajes según su dominio de aplicación

Duración: 02 horas

Procedimiento:

1. Atiende las orientaciones docentes.
2. Realiza una investigación individual o en parejas sobre distintos dominios de aplicación (científico, web, comercial, sistemas embebidos, inteligencia artificial, entre otros).
3. Clasifica al menos cinco lenguajes de programación de acuerdo con su dominio.
4. Justifica la clasificación con base en sus características.
5. Elabora un cuadro comparativo con atributos clave de cada lenguaje.
6. Entrega el reporte y recibe retroalimentación.

Recursos de apoyo:

- Computadora con acceso a internet
- Plantilla de cuadro comparativo

- Documentación oficial y sitios de comunidades .

Unidad II. Características de los lenguajes de programación

Práctica 3: Simulación del proceso de traducción de código

Duración: 02 horas

Procedimiento:

1. Atiende las orientaciones docentes.
2. Analiza los conceptos de compilador, intérprete y máquina virtual.
3. Participa en la simulación grupal del proceso de traducción de código fuente a lenguaje máquina.
4. Representa el flujo mediante un diagrama o storyboard.
5. Expone el resultado y recibe retroalimentación.

Recursos de apoyo:

- Videos didácticos sobre compilación e interpretación
- PSeInt, Python Tutor u otros simuladores educativos
- Material de apoyo proporcionado por su docente

Práctica 4: Comparación de paradigmas mediante análisis de sintaxis

Duración: 04 horas

Procedimiento:

1. Atiende las orientaciones docentes.
2. Analiza ejemplos de código en distintos lenguajes (Python, C, Java, Haskell, Prolog).
3. Identifica elementos básicos de sintaxis y semántica.
4. Reconoce diferencias clave entre los lenguajes y paradigmas.
5. Elabora un reporte comparativo.
6. Entrega el reporte y recibe retroalimentación.

Recursos de apoyo:

- Fragmentos de código entregados por su docente
- Lecturas: Fundamentos de sintaxis y semántica
- Herramientas: Replit, Ideone, o entornos online simples

Unidad III. Implementación y diseño de los lenguajes de programación

Práctica 5: Construcción de gramáticas básicas

Duración: 04 horas

Procedimiento:

1. Atiende las orientaciones docentes.
2. Analiza los fundamentos de gramáticas formales y notación BNF.
3. Diseña una gramática sencilla (por ejemplo, para expresiones aritméticas).
4. Valida su funcionamiento mediante ejemplos.
5. Documenta la gramática y sus resultados.
6. Entrega el reporte y recibe retroalimentación.

Recursos de apoyo:

- Lectura: Introducción a gramáticas y sintaxis
- Herramienta: BNF Playground, Visual Grammar Tools

Práctica 6: Análisis de tipos de datos y estructuras de control en lenguajes comunes

Duración: 04 horas

Procedimiento:

1. Atiende las orientaciones docentes.
2. Compara la declaración y uso de tipos de datos en dos o más lenguajes.
3. Analiza estructuras de control (condicionales, ciclos y funciones).
4. Desarrolla fragmentos de código para ejemplificar diferencias.
5. Documenta resultados y conclusiones.
6. Entrega el reporte y recibe retroalimentación.

Recursos de apoyo:

- Documentación oficial de los lenguajes
- Entornos en línea como Replit, JDoodle

Unidad IV. Paradigmas y lenguajes

Práctica 7: Desarrollo de un programa simple en cada paradigma

Duración: 06 horas

Procedimiento:

1. Atiende las orientaciones docentes.
2. Resuelve un problema sencillo utilizando distintos paradigmas de programación.
3. Implementa soluciones en al menos dos enfoques (estructurado y orientado a objetos, por ejemplo).
4. Analiza ventajas y limitaciones de cada paradigma.
5. Elabora un informe comparativo con resultados.
6. Entrega el reporte y recibe retroalimentación.

Recursos de apoyo:

- IDEs o compiladores según el lenguaje
- Ejemplos de código base proporcionados por su docente

Práctica 8: Proyecto final – Lenguaje multiparadigma

Duración: 08 horas

Procedimiento:

1. Atiende las orientaciones docentes.
2. Selecciona un lenguaje multiparadigma (Python, JavaScript u otro).
3. Diseña un proyecto sencillo que integre al menos dos paradigmas.
4. Implementa la solución propuesta.
5. Presenta el proyecto ante el grupo explicando el uso de los paradigmas.
6. Entrega el reporte final y recibe retroalimentación.

Recursos de apoyo:

- Documentación oficial del lenguaje elegido
- Plantilla de presentación del proyecto
- Herramientas de desarrollo
- Computadora personal

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategias de Enseñanza

- **Estudio de caso:** Para analizar implementaciones reales o ejemplos de paradigmas y lenguajes.
- **Método de proyectos:** Aplicado en el desarrollo de proyectos en lenguajes multiparadigma o por paradigma.
- **Técnica expositiva:** Para introducir conceptos clave como historia, atributos, estructuras sintácticas, paradigmas, etc.
- **Discusión guiada:** Para reflexionar sobre paradigmas, ventajas de lenguajes, estándares, semántica.
- **Instrucción guiada:** Para acompañar prácticas como el diseño de gramáticas o codificación básica.
- **Foros:** Para debate asincrónico sobre comparaciones entre lenguajes o reflexiones de prácticas.
- **Exposición de resoluciones de casos prácticos:** Cuando los estudiantes presentan análisis de código, comparaciones o su proyecto final.

Estrategias de Aprendizaje

- **Investigación documental:** Historia de los lenguajes, características, paradigmas, entornos de programación.
- **Trabajo colaborativo:** En prácticas como la línea del tiempo, análisis comparativo o desarrollo de código en grupo.
- **Mapas conceptuales:** Para organizar temas como los tipos de paradigmas o la estructura de un lenguaje.
- **Cuadros comparativos:** Para comparar lenguajes en atributos, sintaxis, tipos de datos o aplicaciones.
- **Infografías:** Para representar visualmente un lenguaje, sus características o su evolución.

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	40%
– Prácticas de laboratorio	30%
– Trabajo final y exposición	30%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Allen Tucker, Robert Noonan (2006). *Programming: Principles and Paradigms Languages*. (2nd ed.), McGraw Hill. [clásica].

Pierce, B. C. (2002). *Types and programming languages*. MIT Press. [clásica].

Van Roy, P., & Haridi, S. (2004). *Concepts, techniques, and models of computer programming*. MIT Press. [clásica].

Harper, R. (2016). *Practical foundations for programming languages* (2nd ed.). Cambridge University Press. [clásica].

Vasconcelos Santillán, J. (2018). *Introducción a la computación*. Grupo Editorial Patria. Disponible en: <https://libcon.rec.uabc.mx:2246/es/ereader/uabc/98314> [clásica].

Recursos de acceso libre o complementarios externos

Krishnamurthi, S. (2017). *Programming languages: Application and interpretation* (2nd ed.). Brown University. [clásica].

Disponible en: <https://cs.brown.edu/courses/cs173/2012/book/>

Tutorial de codeblocks, [recurso electrónico], disponible en: <http://www.cplusplus.com/doc/tutorial/introduction/codeblocks/>

Tutorial de C, C++, Open GL, STL, [recurso electrónico], disponible en: <http://www.cprogramming.com/tutorial.html>

Tutorial de Racket (Lisp and Scheme), [recurso electrónico], disponible en: <http://docs.racket-lang.org/quick/>
<http://docs.scala-lang.org/es/tutorials/tour/tour-of-scala.html>

Tutorial de Python, [recurso electrónico], disponible en: <https://www.python.org/>

XII. PERFIL DOCENTE

Licenciatura en Ciencias Computacionales o área afín, y tener conocimientos sólidos en teoría, diseño e implementación de lenguajes de programación, incluyendo sus paradigmas, estructuras sintácticas y semánticas, así como experiencia práctica en el uso de lenguajes actuales. Debe ser capaz de guiar al estudiantado en el análisis crítico y comparativo de lenguajes, fomentar el pensamiento lógico y estructurado, y aplicar estrategias didácticas activas que promuevan el aprendizaje colaborativo, reflexivo y orientado a la resolución de problemas.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Desarrollo Web
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 02 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:06
Etapas de formación:	Básica
Carácter:	Optativa
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	María Victoria Meza Kubo Gabriel Aparicio Hernández
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La unidad de aprendizaje de desarrollo web permite al estudiantado adquirir la competencia para diseñar, desarrollar y desplegar aplicaciones web modernas, integrando tecnologías del front-end, servicios REST, control de versiones y buenas prácticas de programación. La unidad de aprendizaje de Desarrollo Web es de carácter optativo dentro de la etapa disciplinaria. Su área de conocimiento es Programación e Ingeniería de software y sería recomendable haber acreditado la unidad de aprendizaje de Introducción a la Programación.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Desarrollar aplicaciones web mediante la aplicación de distintos frameworks y lenguajes de programación web para compartir datos desde cualquier navegador, con responsabilidad y creatividad.

IV. EVIDENCIA DE APRENDIZAJE

Una aplicación web funcional construida con HTML5, CSS3, JavaScript y un framework web. La aplicación deberá acompañarse de su documentación técnica, reportes de laboratorio y presentación final.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar y analizar algoritmos utilizando fundamentos matemáticos, de modelado y lógicos, para generar soluciones optimizadas e innovadoras a problemas complejos, que contribuyan al avance tecnológico y científico, con pensamiento crítico, actitud ética y perseverancia.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia; Vanguardia.

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos del Desarrollo Web

Competencia

Analizar los conceptos fundamentales de la programación en Internet, identificando los componentes principales de una aplicación web, sus protocolos, arquitecturas y funcionamiento general, para conocer su aplicación en el contexto de las tecnologías del desarrollo web, con actitud crítica.

Contenido

1.1 Internet y el world wide web (WWW)

- 1.2 Arquitectura cliente–servidor
- 1.3 HTTP / HTTPS: métodos, estados y mensajes
- 1.4 Tipos de aplicaciones web (estáticas, dinámicas, SPA, híbridas)
- 1.5 Herramientas básicas del desarrollador web

Duración

06 horas

Unidad II. Herramientas y Entornos de Desarrollo

Competencia

Configurar herramientas y entornos de desarrollo web, mediante el uso de editores, IDEs, control de versiones y contenedores para favorecer el trabajo colaborativo, con responsabilidad y actitud crítica.

Contenido

- 2.1 Editores de código e IDEs
- 2.2 Control de versiones con Git
- 2.3 Repositorios remotos (GitHub)
- 2.4 Gestión de paquetes (npm)
- 2.5 Introducción a Docker y entornos portables
- 2.6 Flujo básico de trabajo colaborativo

Duración

6 horas

Unidad III. Desarrollo Front-End

Competencia

Desarrollar interfaces web dinámicas mediante el uso de HTML5, CSS3, JavaScript y frameworks front-end para construir aplicaciones responsivas y accesibles con responsabilidad y creatividad.

Contenido

- 3.1 HTML y estructura
 - 3.1.1 Estructura base del documento HTML5
 - 3.1.2 Etiquetas semánticas
 - 3.1.3 Formularios y validación
 - 3.1.4 Multimedia y Canvas
- 3.2 Diseño con CSS
 - 3.2.1 Selectores, box model, propiedades fundamentales
 - 3.2.2 Responsive Web Design
 - 3.2.3 Frameworks CSS (Bootstrap)
- 3.3 Programación con JavaScript
 - 3.3.1 Variables, funciones, DOM
 - 3.3.2 Eventos y manipulación dinámica de elementos

3.3.3 Fetch API y consumo básico de datos

3.4 Frameworks JavaScript

3.4.1 Introducción a React (o equivalente)

3.4.2 Componentes, props y estado

Duración

12 horas

Unidad IV. Servicios REST e Integración con Datos

Competencia

Desarrollar servicios REST mediante el empleo de tecnologías de desarrollo de vanguardia para permitir la consulta y almacenamiento de información desde distintos clientes, con responsabilidad y creatividad.

Contenido

4.1. Conceptos de API y REST

4.2 Métodos HTTP para CRUD

4.3 Express.js básico (o framework equivalente)

4.4 Exposición de datos y manejo de JSON

4.5 Integración front-end ↔ back-end

4.6 Pruebas con POSTMAN

4.7 Introducción al despliegue (servidor / nube)

Duración

08 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Fundamentos del Desarrollo Web

Práctica 1: Exploración del Entorno Web y HTTP

Duración:

04 horas

Procedimiento:

1. Atiende a la explicación de la persona docente sobre la arquitectura cliente–servidor e Internet.
2. Abre la herramienta de desarrolladores en distintos navegadores (Chrome, Firefox).
3. Visita tres sitios web y analiza: estructura HTML, tiempo de carga, recursos solicitados, códigos HTTP.
4. Compara cómo cada navegador procesa la misma página.

5. Realiza un diagrama del flujo de petición-respuesta HTTP.
6. Toma capturas de pantalla de las herramientas de inspección.
7. Integra un reporte técnico con análisis y evidencias.
8. Genera un PDF y súbelo a la plataforma del curso.
9. Recibe retroalimentación docente.

Recursos de apoyo:

- Navegadores web
- Computadora
- Acceso a internet

Práctica 2: Mapa de Conceptos del Desarrollo Web

Duración:

02 horas

Procedimiento:

1. Atiende a la explicación docente sobre conceptos fundamentales (HTML, CSS, JS, HTTP, servidor).
2. Usa una herramienta de diagramación para elaborar un mapa mental que conecte los conceptos analizados.
3. Incluye definiciones breves y relaciones entre elementos.
4. Exporta el diagrama a PDF.
5. Entregar en la plataforma.
6. Recibir retroalimentación docente.

Recursos de apoyo:

- Herramienta de mapas conceptuales (Draw.io, Miro, Lucidchart)
- Computadora
- Internet

Unidad II. Herramientas y Entornos de Desarrollo

Práctica 3: Instalación y Configuración del Entorno de Desarrollo

Duración:

04 horas

Procedimiento:

1. Atiende a la explicación docente sobre IDEs, extensiones y flujos de trabajo del desarrollo web.
2. Instala VSCode y configura extensiones recomendadas.

3. Realiza pruebas con archivos HTML, CSS y JS básicos.
4. Instala Git y verifica su funcionamiento desde la terminal.
5. Documenta cada paso con capturas de pantalla.
6. Genera un reporte técnico.
7. Entregar en PDF vía plataforma.
8. Recibir retroalimentación docente.

.Recursos de apoyo:

- VSCode
- Extensiones
- Git
- Computadora
- Internet

Práctica 4: Uso de Git y Repositorios Remotos

Duración:

2 horas

Procedimiento

1. Atiende a la explicación docente sobre control de versiones.
2. Crea un repositorio local y realiza commits estructurados.
3. Publica el repositorio en GitHub.
4. Realiza cambios desde la web y sincroniza con tu entorno local.
5. Documenta la secuencia de comandos utilizados.
6. Entregar reporte en PDF.
7. Recibir retroalimentación docente.

Recursos de apoyo:

- Git / GitHub
- Computadora
- Internet

Unidad III. Desarrollo Front-End

Práctica 5: Arreglos unidimensionales: recorrido y operaciones básicas

Duración

04 horas

Procedimiento

1. Atiende a la explicación docente sobre estructura HTML5 y estilizado con CSS3.

2. Crea una página web con encabezados, secciones semánticas, listas, imágenes y vínculos.
3. Desarrolla una hoja de estilos externa aplicando diseño responsivo básico.
4. Verifica la visualización en distintos navegadores y resoluciones.
5. Documenta el proceso con capturas y explicaciones breves.
6. Entregar en PDF en la plataforma.
7. Recibir retroalimentación docente.

Recursos de apoyo:

- Editor de código
- Navegadores web
- Computadora
- Internet

Práctica 6: Interactividad con JavaScript

Duración

04 horas

Procedimiento:

1. Atiende a la explicación docente sobre eventos y funciones en JavaScript.
2. Agrega interactividad a la página creada: validación de campos, botones dinámicos, cambios visuales con JS.
3. Toma capturas de pantalla del funcionamiento.
4. Integrar evidencia en PDF y cargar a la plataforma.
5. Recibir retroalimentación docente.

Recursos de apoyo:

- Editor de código
- Navegadores
- Computadora
- Internet

Práctica 7: Uso de Bootstrap y Consumo de Datos con Fetch API

Duración

04 horas

Procedimiento:

1. Atiende a la explicación docente sobre Bootstrap y Fetch API.
2. Integra Bootstrap a la página existente y mejora su diseño con componentes predefinidos.

3. Consume una API pública utilizando Fetch API (por ejemplo Pokémon, JSONPlaceholder, SWAPI).
4. Muestra información dinámica en la página.
5. Documenta el funcionamiento e incluye capturas.
6. Entregar en PDF.
7. Recibir retroalimentación docente.

Recursos de apoyo:

- Bootstrap
- API pública
- Computadora
- Internet

Unidad IV. Servicios REST e Integración

Práctica 8: Desarrollo de un Servicio REST e Integración Front-Back

Duración

04 horas

Procedimiento

1. Atiende a la explicación docente sobre servicios REST y Express.
2. Instala Node.js y configura un proyecto básico.
3. Implementa endpoints GET y POST que trabajen con datos locales (JSON).
4. Prueba los endpoints con POSTMAN.
5. Conecta la página web creada previamente al servicio REST usando Fetch API.
6. Muestra datos consultados desde el backend en el front-end e implementa envío de datos.
7. Documenta el flujo completo (capturas + explicación del código).
8. Entregar reporte en PDF.
9. Recibir retroalimentación docente.

Recursos de apoyo:

- Node.js
- Editor de código
- Computadora
- Internet

Práctica 9: Integración Final y Despliegue del Proyecto

Duración

04 horas

Procedimiento

1. Atiende a la explicación docente sobre despliegue web.
2. Realiza la versión final de tu aplicación completa.
3. Publica el front-end en GitHub Pages, Netlify o Vercel.
4. Prueba el proyecto final desde distintos dispositivos.
5. Documenta la URL de despliegue, capturas y conclusiones.
6. Entregar en PDF en plataforma.
7. Recibir retroalimentación docente.

Recursos de apoyo:

- Plataforma de despliegue
- Editor de código
- Computadora
- Internet

MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Aprendizaje basado en problemas
- Técnica expositiva
- Método de proyectos

Estrategia de aprendizaje

- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Ejercicios con pseudocódigos

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario, cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	30%
– Tareas y prácticas de laboratorio	30%
– Aplicación web	40%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Brooks, D. R. (2024). *Guide to HTML, JavaScript and PHP: For scientists and engineers* [eBook]. Springer.

Fernández Casado, P. E. (2023). *Construcción y diseño de páginas web con HTML, CSS y JavaScript*. RC Libros / Alfaomega.

Gascón Gonzáles, U. (2025). *El gran libro de Node.js*. Alfaomega – Marcombo.

Vigouroux, C. (2022). *Aprender a desarrollar con JavaScript: Aspectos principales del uso de frameworks* (4ª ed.). Ediciones ENI.

Zuckarelli, J. L. (2022). *Learn coding with Python and JavaScript: A practical introduction for beginners* [eBook]. Apress.

Recursos de acceso libre o complementarios externos

Brown, E. (2019). *Web Development with Node and Express: Leveraging the JavaScript Stack* (2ª ed.). O'Reilly Media.

DesarrolloWeb.com. (s.f.). *Desarrollo web*. <https://desarrolloweb.com/>

World Wide Web Consortium. (s.f.). *W3Schools Online Web Tutorials*. <https://www.w3schools.com/tutorials/>

World Wide Web Consortium. (s.f.). *Developers*. <https://www.w3.org/developers/>

XII. PERFIL DOCENTE

Contar con formación en el área de Ciencias Computacionales, Ingeniería en Ciencia de Datos, Ingeniería de Software o área afín; preferentemente con estudios de posgrado y dos años de experiencia de práctica docente, con amplio dominio de los contenidos temáticos y experiencia en la conducción de cursos en línea. Ser una persona responsable y proactiva que proporcione al estudiante las herramientas que le permitan desarrollar las aptitudes y competencias descritas para este curso.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Programación de Sistemas Empotrados
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 03 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:07
Etapas de formación:	Disciplinaria
Carácter:	Optativa
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	Eloísa del Carmen García Canseco Miguel Ángel Alonso Arévalo
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La unidad de aprendizaje tiene como propósito desarrollar en cada estudiante la capacidad de diseñar, programar e implementar soluciones tecnológicas basadas en sistemas empujados modernos. Ante la convergencia de la instrumentación científica y el Internet de las Cosas (IoT), es fundamental que las y los estudiantes de Física y Ciencias Computacionales dominen la interacción entre el software y el mundo físico.

En esta unidad de aprendizaje, cada estudiante transitará desde los fundamentos de la arquitectura de hardware y la programación eficiente en C++, pasando por el prototipado rápido de instrumentación con MicroPython, hasta la implementación de sistemas conectados e inteligentes (TinyML). Proporciona las competencias técnicas necesarias para el desarrollo de prototipos de automatización, monitoreo remoto e instrumentación inteligente aplicables en la industria y la investigación.

Esta unidad de aprendizaje es de carácter optativo en la etapa disciplinaria y pertenece al área de conocimiento de Arquitectura de Computadoras.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Diseñar e implementar sistemas empujados conectados e inteligentes, a través de la selección de arquitecturas de hardware modernas y la aplicación de técnicas de programación híbrida y algoritmos de aprendizaje automático en el borde, para resolver problemas de adquisición de datos y control automatizado con rigor metodológico y eficiencia técnica.

IV. EVIDENCIA DE APRENDIZAJE

Portafolio de prácticas que documente la resolución de prácticas de laboratorio. Este portafolio deberá evidenciar el proceso de desarrollo técnico mediante la inclusión del código fuente comentado, diagramas de conexión esquemática y registros audiovisuales (video o captura de pantalla) que demuestren el correcto funcionamiento de los sensores, actuadores e interfaces de comunicación.

Proyecto final de un prototipo funcional de un sistema empujado que resuelva una problemática específica de monitoreo o de control. Esta evidencia se materializa en dos entregables: el dispositivo físico operando de forma autónoma (que debe integrar la adquisición de señales, el procesamiento lógico o inteligente mediante TinyML y la conectividad IoT para la transmisión de datos) y un reporte técnico formal. Dicho reporte debe detallar la justificación de componentes, la metodología de diseño, las pruebas de validación realizadas y el manual de operación.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Excelencia.

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos de Arquitectura y Programación de Hardware

Competencia

Configurar los puertos de entrada/salida y recursos internos del microcontrolador, por medio del uso de entornos de desarrollo profesional y programación estructurada en C++, para establecer el control directo del hardware y la gestión eficiente de tiempos, con orden lógico y rigor técnico.

Contenido

1.1. Aplicaciones de sistemas empuotrados

1.1.1. Definición y características de un sistema empuotrado

1.1.2. Análisis de casos de estudio sobre el internet de las cosas, dispositivos vestibles e inteligencia artificial en el borde.

1.2. Arquitectura de sistemas empuotrados modernos

1.2.1. Comparativa entre microcontroladores y microprocesadores

1.2.2. Análisis de Arduino (AVR), ESP32 (Xtensa) y plataformas con enfoque industrial (STM32, etc.)

1.3. Entorno de desarrollo y lenguajes de programación: instalación, compilación y carga.

1.4. Simulación y prototipado virtuales para pruebas sin hardware.

Duración

08 horas

Unidad II. Instrumentación, Sensores y Prototipado Rápido

Competencia

Integrar sensores físicos y actuadores al sistema digital, a través de la aplicación de protocolos de comunicación estándar e interfaces de prototipado rápido, para la adquisición precisa de

datos del entorno y su validación funcional, con cuidado y responsabilidad en el manejo de componentes electrónicos.

Contenido

- 2.1. Manejo de entradas y salidas digitales
- 2.2 Adquisición de señales analógicas: ADC y sensores físicos.
- 2.3. Protocolos de comunicación: UART, I2C y SPI.
- 2.4. Actuadores y potencia: control PWM y drivers.
- 2.5. Prototipado Rápido con MicroPython: Validación de sensores y manejo de datos.
- 2.6 Gestión de eventos: uso de interrupciones para el manejo de eventos en tiempo real

Duración

08 horas

Unidad III. Internet de las Cosas (IoT) e Inteligencia Artificial en el Borde (TinyML)

Competencia

Implementar mecanismos de conectividad inalámbrica y modelos de inteligencia artificial, por medio del uso de protocolos de Internet de las Cosas, IoT, y librerías de Aprendizaje de Máquina ligero, para dotar al sistema de capacidades de monitoreo remoto y toma de decisiones autónoma, promoviendo la innovación tecnológica y la eficiencia.

Contenido

- 3.1. Conectividad Inalámbrica: WiFi, Cliente HTTP y API REST.
- 3.2. Protocolos IoT Industriales: MQTT y Dashboards en la nube.
- 3.3. Introducción a TinyML: flujo de trabajo y Edge AI.
- 3.4. Inferencia en el microcontrolador: clasificación y reconocimiento de patrones.

Duración

10 horas

Unidad IV. Prototipos de sistemas empotrados

Competencia

Desarrollar un prototipo funcional de sistema empotrado, por medio de la integración de hardware, software y conectividad, bajo una metodología de diseño ingenieril, para resolver una problemática específica de monitoreo o control, con compromiso con la calidad y claridad en la documentación técnica.

Contenido

- 4.1. Definición y Diseño: Requerimientos y diagramas de bloques.
- 4.2. Implementación: Construcción del hardware y firmware.
- 4.3. Validación y Pruebas: Depuración funcional.
- 4.4. Documentación y presentación del proyecto.

Duración

06 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Fundamentos de Arquitectura y Programación de Hardware

Práctica 1: Simulación y arquitectura virtual

Duración

04 horas

Procedimiento

1. Atiende instrucciones para el desarrollo de la práctica y el uso de simuladores (Wokwi/Tinkercad).
2. Configura un entorno virtual seleccionando la placa ESP32 y analiza su diagrama de pines (Pinout).
3. Diseña un circuito virtual básico e implementa el código en C++ para verificar las estructuras setup() y loop().
4. Utiliza herramientas de depuración virtual (monitor serie) para imprimir mensajes y variables.
5. Simula una falla de hardware (p. ej., un LED mal conectado) y documenta el proceso de identificación del error.
6. Recibe retroalimentación en la bitácora de la práctica.

Recursos de apoyo

- Computadora con acceso a internet
- Simulador Woki o Tinkercad

Práctica 2: Configuración del entorno de desarrollo

Duración

04 horas

Procedimiento

1. Atiende las instrucciones para el desarrollo de la práctica.
2. Instala el IDE (Arduino/PlatformIO) y los drivers necesarios para el ESP32.

3. Configura y carga un programa "Hola Mundo" serial para verificar la comunicación con la PC.
4. Implementa un circuito con botones y LEDs externos en el protoboard.
5. Programa la lectura de entradas digitales (botones) aplicando técnicas de eliminación de rebotes por software.
6. Controla las salidas digitales según la lógica condicional planteada por su docente.
7. Recibe retroalimentación en la bitácora de la práctica.

Recursos de apoyo

- Multímetro
- Placa de desarrollo Arduino o ESP32
- cable de datos USB
- componentes electrónicos como LEDs, botones, resistencias de 330 ohms
- protoboard

Unidad II. Instrumentación, Sensores y Prototipado Rápido

Práctica 3: Gestión de Entradas/Salidas e Interrupciones

Duración

05 horas

Procedimiento

1. Atiende las instrucciones para el desarrollo de la práctica.
2. Implementa un circuito con múltiples entradas (botones) y salidas digitales (LEDs).
3. Programa la lectura de los botones y resuelve el problema del "rebote" mecánico mediante software.
4. Configura una Rutina de Servicio de Interrupción (ISR) para detectar un evento crítico (pulsación de emergencia) que interrumpa el flujo normal del programa.
5. Compara el comportamiento del sistema usando "polling" vs "interrupciones".
6. Recibe retroalimentación en la bitácora de la práctica

Recursos de apoyo

- Computadora con el IDE instalado
- Placas de desarrollo Arduino o ESP32
- Componentes electrónicos
- Computadora con el IDE instalado

Práctica 4: Señales analógicas y control de potencia

Duración

05 horas

Procedimiento

1. Atiende las instrucciones para el desarrollo de la práctica.

2. Conecta sensores analógicos (termistor/LDR) y configura el ADC (resolución y atenuación).
3. Procesa la señal leída (mapeo de valores) para controlar la intensidad de un LED o velocidad de un motor usando PWM.
4. Utiliza drivers de potencia (transistores/puente H) para manejar cargas que superen la corriente del pin del microcontrolador.
5. Visualiza la señal PWM teórica frente a la real en un osciloscopio.
6. Recibe retroalimentación en la bitácora de la práctica

Recursos de apoyo

- Computadora con el IDE instalado
- Placas de desarrollo Arduino o ESP32
- Sensores analógicos, motor DC o servomotor, driver de potencia (L293D o transistor).
- Osciloscopio

Práctica 5: Instrumentación digital (I2C/SPI)

Duración

05 horas

Procedimiento

1. Atiende las instrucciones para el desarrollo de la práctica.
2. Identifica las direcciones de los dispositivos en el bus I2C mediante un escáner de software.
3. Interconecta un sensor ambiental o inercial (p. ej., BMP280/MPU6050) y una pantalla OLED.
4. Desarrolla el firmware para leer los registros del sensor y mostrar los valores físicos en la pantalla local.
5. Implementa el manejo de errores para detectar si un sensor se desconecta.
6. Recibe retroalimentación en la bitácora de la práctica.

Recursos de apoyo

- Computadora con el IDE instalado
- Placas de desarrollo Arduino o ESP32
- Sensores digitales I2C/SP2 y pantalla OLED

Práctica 6: Prototipado híbrido con MicroPython

Duración

04 horas

Procedimiento

1. Atiende las instrucciones para el desarrollo de la práctica.
2. Instala el firmware de MicroPython en el SoC.

3. Utiliza la consola REPL para interactuar directamente con los sensores conectados en prácticas anteriores sin recompilar.
4. Escribe un script en Python que lea el sensor y guarde los datos en una lista o en un archivo local.
5. Analiza las ventajas del scripting para la validación rápida de hardware frente a C++.
6. Recibe retroalimentación en la bitácora de la práctica.

Recursos de apoyo

- Computadora con un IDE para Python
- Firmware MicroPython
- Sensores
- Placas de desarrollo ESP32

Unidad III. Internet de las Cosas (IoT) e Inteligencia Artificial en el Borde (TinyML)

Práctica 7: Conectividad Web y API REST

Duración

05 horas

Procedimiento

1. Atiende las instrucciones para el desarrollo de la práctica.
2. Configura el microcontrolador en modo Station (STA) para acceder a internet.
3. Implementa un cliente HTTP para realizar peticiones GET y POST a una API pública o servidor de prueba.
4. Crea un servidor web local asíncrono en el chip para controlar actuadores desde un navegador móvil.
5. Recibe retroalimentación en la bitácora de la práctica.

Recursos de apoyo

- Red wifi, teléfono inteligente y/o computadora
- Placa de desarrollo ESP32

Práctica 8: Telemetría industrial con MQTT

Duración

05 horas

Procedimiento

1. Atiende las instrucciones para el desarrollo de la práctica.
2. Configura la conexión a un broker MQTT (Mosquitto/HiveMQ/Thingsboard).
3. Estructura los datos de los sensores en formato JSON.
4. Publica la telemetría en un tópico específico y suscríbete a un tópico de control para recibir comandos remotos.
5. Visualiza los datos en un dashboard en la nube en tiempo real.

6. Recibe retroalimentación en la bitácora de la práctica

Recursos de apoyo

- Computadora con acceso a internet
- Placas de desarrollo ESP32
- Sensores diversos
- Plataforma IoT en la nube

Práctica 9: TinyML - Inferencia de IA en Hardware

Duración

05 horas

Procedimiento

1. Atiende las instrucciones para el desarrollo de la práctica.
2. Realiza la recolección de datos (dataset) usando los sensores del microcontrolador (p. ej., gestos de movimiento).
3. Entrena un modelo de clasificación en una plataforma de Edge AI (p. ej., Edge Impulse).
4. Despliega el modelo optimizado como una librería C++ en el microcontrolador.
5. Valida la inferencia en tiempo real (clasificación correcta de gestos o patrones) sin conexión a internet.
6. Recibe retroalimentación en la bitácora de la práctica.

Recursos de apoyo

- Computadora con acceso a internet
- Cuenta en plataforma Edge AI
- Placas de desarrollo ESP32
- Sensores IMU

Unidad IV. Prototipos de sistemas empotrados

Práctica 10: Integración y validación del proyecto final

Duración

06 horas

Procedimiento

1. Atiende las instrucciones para el desarrollo de la práctica.
2. Presenta la integración final de los módulos de hardware y de software del proyecto.
3. Ejecuta pruebas funcionales completas y de estrés (funcionamiento continuo).
4. Realiza el ajuste de parámetros y la calibración final de los sensores.
5. Presenta el prototipo en funcionamiento autónomo y defiende las decisiones de diseño ante el grupo.
6. Recibe retroalimentación en la bitácora de la práctica

Recursos de apoyo

- Computadora
- Fuentes de alimentación
- Instrumentos de medición
- Materiales del proyecto

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer la forma de trabajo, los criterios de evaluación, la calidad de los trabajos académicos y los derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Estudio de caso
- Método de proyectos
- Aprendizaje basado en proyectos
- Aprendizaje basado en casos
- Aprendizaje basado en preguntas
- Técnica expositiva
- Discusión guiada
- Instrucción guiada
- Foros

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Mapas mentales
- Mapas conceptuales
- Cuadros comparativos
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a los exámenes ordinarios y extraordinarios, cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación está en una escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Portafolio de prácticas	40%
– Tareas	10%
– Proyecto final	50%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Ganguly, A. K. (2014). *Embedded systems: Design, programming and applications*. eBook Academic Collection (EBSCOhost). [clásica]
- Iodice, G.M. (2022). *TinyML Cookbook : Combine Artificial Intelligence and Ultra-low-power Embedded Devices to Make the World Smarter*. Packt Publishing.
- Oshana, R., & Kraeling, M. (Eds.). (2013). *Software engineering for embedded systems: Methods, practical techniques, and applications*. Newnes. [clásica]
- Pan, T., & Zhu, Y. (2018). *Designing embedded systems with Arduino: A fundamental technology for makers*. Springer Singapore. [clásica]
- Salas Arriarán, S. (2025). *Aprendizaje significativo de sistemas embebidos: De la teoría a la práctica*. Universidad Peruana de Ciencias Aplicadas.

Recursos de acceso libre o complementarios externos

- Autodesk. (2024). *Tinkercad Circuits: Simulador de electrónica y Arduino*. [Software en línea]. <https://www.tinkercad.com/>
- Edge Impulse. (2024). *Edge Impulse Documentation: Machine Learning on Embedded Devices*. <https://docs.edgeimpulse.com/>
- Kurniawan, A. (2019). *Internet of Things Projects with ESP32: Build exciting and powerful IoT projects using the highly popular ESP32 chip*. Packt Publishing.
- MicroPython. (2024). *MicroPython Documentation: Libraries and References*. <https://docs.micropython.org/en/latest/>
- PlatformIO. (2024). *PlatformIO: A new generation ecosystem for embedded development*. <https://platformio.org/>
- Wokwi. (2024). *Wokwi: Online ESP32, STM32, Arduino Simulator*. [Software en línea]. <https://wokwi.com/>

XII. PERFIL DOCENTE

Formación en física, electrónica o áreas afines, preferentemente con estudios de posgrado. Con conocimientos sólidos en el área de instrumentación y programación de sistemas empuotrados, así como experiencia en la resolución analítica de problemas y en el uso de herramientas

computacionales de apoyo. Experiencia mínima de 2 años en docencia. Disposición para promover el aprendizaje crítico y autónomo del estudiantado.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Desarrollo de Aplicaciones Móviles
Clave:	
Distribución horaria:	Horas Clase: 01 Horas Taller: 00 Horas Laboratorio: 04 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 01 Créditos:06
Etapas de formación:	Disciplinaria
Carácter:	Optativa
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	María Victoria Meza Kubo Roberto Adolfo Romero Hernández
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

El propósito de esta unidad de aprendizaje es formar al estudiante en el diseño, desarrollo e implementación de aplicaciones móviles, mediante la integración progresiva de tecnologías, herramientas y metodologías actuales, que incluyen el desarrollo de interfaces gráficas, manejo de datos, consumo de servicios web, uso de sensores y bibliotecas gráficas, con el fin de generar soluciones funcionales, interactivas y contextualizadas a problemáticas reales.

Esta unidad de aprendizaje es optativa de la etapa disciplinaria del área de Programación e Ingeniería de Software.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Desarrollar aplicaciones móviles, empleando metodologías y herramientas de desarrollo de aplicaciones móviles, para resolver problemas de manejo de información y funcionalidad contextual de manera eficiente, creativa y responsable.

IV. EVIDENCIA DE APRENDIZAJE

Desarrolla una aplicación móvil, que incluya el prototipo funcional de la aplicación, la documentación de la metodología empleada y los manuales de usuario correspondientes.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Innovación social.

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos del desarrollo de aplicaciones móviles

Competencia

Analizar la evolución, el estado actual y las tecnologías del desarrollo de aplicaciones móviles, mediante la revisión histórica y técnica de dispositivos, sistemas operativos y enfoques de

desarrollo, para seleccionar plataformas y herramientas adecuadas en la construcción de aplicaciones móviles, con actitud crítica y disposición.

Contenido

- 1.1. Evolución de los dispositivos móviles
- 1.2. Actualidad y tendencias futuras de los dispositivos móviles
- 1.3. El inicio de la revolución móvil
- 1.4. Sistemas operativos para dispositivos móviles
- 1.5. Dispositivos móviles emergentes (tabletas y otros)
- 1.6. Enfoques de desarrollo: nativo, web y multiplataforma

Duración

02 horas

Unidad II. Ambiente de desarrollo

Competencia

Desarrollar aplicaciones para dispositivos móviles sencillas, empleando un ambiente de desarrollo y otros componentes, para familiarizarse con las herramientas adecuadas para programar aplicaciones para móviles, con creatividad y responsabilidad.

Contenido

- 2.1. Selección del ambiente de desarrollo
- 2.2. Instalación y configuración
- 2.3. Componentes del emulador
- 2.4. Herramientas para depuración
- 2.5. Primer programa

Duración

02 horas

Unidad III. Interfaz gráfica

Competencia

Programar la interfaz gráfica de la aplicación móvil, mediante la realización de prácticas de laboratorio en las que se utilizan componentes gráficos, eventos y la comunicación entre ventanas, para comprender el manejo de componentes y producir interfaces graficas amigables y funcionales, con iniciativa y creatividad.

Contenido

- 3.1. Estructura de la aplicación
- 3.2. Preparando la interfaz de usuario

- 3.3. Componentes visuales
- 3.4. Manejo de eventos
- 3.5. Transición entre ventanas
- 3.6. Notificación de mensajes

Duración

03 horas

Unidad IV. Almacenamiento de datos

Competencia

Evaluar los diferentes tipos de recursos de almacenamiento incluidos en un dispositivo móvil, mediante la revisión de información técnica y la elaboración de prácticas de laboratorio, para determinar el procedimiento más adecuado en el tratamiento de los espacios de almacenamiento, con creatividad y responsabilidad.

Contenido

- 4.1. Manejo de datos en los dispositivos con bajos recursos
- 4.2. Utilizando archivos para almacenar datos
- 4.3. Integración de bases de datos
- 4.4. Uso de datos mediante Web Services

Duración

02 horas

Unidad V. Aplicaciones Web para dispositivos móviles

Competencia

Experimentar los conceptos involucrados en el desarrollo de aplicaciones Web para móviles, mediante la revisión y análisis de técnicas y herramientas, así como de la elaboración de prácticas de laboratorio, para precisar si la solución óptima deberá ser realizada orientada a Web o nativa, con actitud crítica y creatividad.

Contenido

- 5.1. La importancia del desarrollo Web para móviles
- 5.2. HTML5 y CSS3
- 5.3. Técnicas recomendadas para desarrollo Web en ambientes móviles
- 5.4. Ambientes de desarrollo multiplataforma

Duración

03 horas

Unidad VI. Sensores

Competencia

Experimentar con los diversos sensores incluidos en un dispositivo móvil, mediante la revisión de la información técnica sobre sensores y la realización de prácticas de laboratorio, para elaborar aplicaciones que requieren la funcionalidad de un sensor específico, con responsabilidad y creatividad.

Contenido

6.1. Introducción a los sensores.2. Tipos de sensores
6.3. Uso del sensor Acelerómetro

Duración

02 horas

Unidad VII. Bibliotecas gráficas

Competencia

Programar aplicaciones con la biblioteca gráfica OpenGL, mediante la realización de prácticas de laboratorio, para desarrollar aplicaciones que permitan la generación y manipulación de gráficas en dos y tres dimensiones, con creatividad y responsabilidad.

Contenido

7.1. Gráficos en 2D o en 3D
7.2. OpenGL
7.3. Especialización en videojuegos para móviles

Duración

02 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Fundamentos del desarrollo de aplicaciones móviles

Práctica 1: Análisis del ecosistema móvil y definición de aplicación

Duración:

04 horas

Procedimiento:

1. Atiende a la explicación de la persona docente sobre la evolución de los dispositivos móviles.
2. Investiga las principales plataformas móviles (Android, iOS, multiplataforma).
3. Analiza tres aplicaciones móviles populares considerando funcionalidad y tipo de desarrollo.
4. Identifica tendencias actuales en aplicaciones móviles.
5. Define una problemática a resolver mediante una aplicación móvil.
6. Elabora una propuesta inicial de aplicación (objetivo, usuarios, funcionalidades).
7. Integra un reporte técnico con el análisis realizado.
8. Genera un PDF y súbelo a la plataforma del curso.
9. Recibe retroalimentación docente.

Recursos de apoyo:

- Computadora
- Acceso a internet
- Artículos y sitios especializados

Práctica 2: Selección de plataforma y herramientas

Duración:

04 horas

Procedimiento:

1. Atiende a la explicación sobre enfoques de desarrollo (nativo, web, multiplataforma).
2. Compara al menos dos frameworks de desarrollo móvil.
3. Analiza ventajas, desventajas y requerimientos técnicos.
4. Selecciona la tecnología a utilizar en el curso.
5. Justifica la elección en función del proyecto propuesto.
6. Integra un documento técnico con la comparación y decisión.
7. Genera un PDF y súbelo a la plataforma del curso.

Recibe retroalimentación docente.

Recursos de apoyo:

- Computadora
- Librerías para móviles

Unidad II. Ambiente de desarrollo

Práctica 3: Configuración del entorno de desarrollo

Duración:

04 horas

Procedimiento:

1. Atiende a la explicación sobre entornos de desarrollo móvil.
2. Instala el IDE correspondiente (Android Studio, Flutter u otro).
3. Configura el emulador o dispositivo físico.
4. Verifica dependencias y herramientas necesarias.
5. Realiza pruebas de ejecución.
6. Documenta el proceso de instalación.
7. Genera evidencias (capturas).
8. Sube el reporte a la plataforma.
9. Recibe retroalimentación docente.

Recursos de apoyo:

- Computadora
- IDE seleccionado

Práctica 4: Desarrollo de la aplicación base

Duración:

06 horas

Procedimiento:

1. Crea un nuevo proyecto móvil.
2. Explora la estructura del proyecto.
3. Identifica archivos principales.
4. Ejecuta la aplicación en el emulador.
5. Modifica elementos básicos (texto, colores).
6. Realiza pruebas de ejecución para la comunicación entre ventanas.
7. Documenta cambios realizados.
8. Entrega evidencias funcionales.
9. Recibe retroalimentación docente.

Recursos de apoyo:

- Computadora
- Documentación sobre el manejo de ventanas del sitio oficial
- IDE
- Emulador

Unidad III. Interfaz gráfica

Práctica 5: Prototipado de interfaz

Duración:

04 horas

Procedimiento:

1. Atiende a la explicación sobre diseño de interfaces móviles.
2. Diseña prototipos (wireframes) de la aplicación.
3. Define pantallas principales.
4. Establece flujo de navegación.
5. Utiliza herramientas de prototipado.
6. Integra propuesta visual.
7. Entrega prototipo documentado.
8. Recibe retroalimentación docente.

Recursos de apoyo:

- Computadora
- Sitio oficial de la tecnología
- Aplicaciones de apoyo para la elaboración de interfaces gráficas

Práctica 6: Implementación de interfaz gráfica

Duración:

04 horas

Procedimiento:

1. Implementa las pantallas diseñadas.
2. Utiliza componentes visuales.
3. Configura layouts responsivos.
4. Aplica estilos básicos.
5. Verifica visualización en distintos dispositivos.
6. Realiza pruebas de interfaz.
7. Documenta resultados.
8. Entrega aplicación funcional.
9. Recibe retroalimentación docente.

Recursos de apoyo:

- Computadora
- Documentación sobre componentes gráficos del sitio oficial
- Referencias (libro) relacionadas con interfaces.

Práctica 7: Navegación y eventos

Duración:

04 horas

Procedimiento:

1. Implementa navegación entre pantallas.
2. Programa eventos de interacción.
3. Gestiona paso de datos entre vistas.
4. Agrega notificaciones básicas.
5. Realiza pruebas funcionales.
6. Documenta comportamiento de la app.
7. Entrega avances.
8. Recibe retroalimentación docente.

Recursos de apoyo:

- Computadora
- Documentación sobre componentes gráficos del sitio oficial
- Referencias (libro) relacionada con interfaces.
- IDE

Unidad IV. Almacenamiento de datos

Práctica 8: Persistencia de datos en aplicaciones móviles

Duración:

04 horas

Procedimiento:

1. Atiende a la explicación sobre almacenamiento local.
2. Implementa almacenamiento con archivos o preferencias.
3. Diseña estructura de datos.
4. Implementa operaciones básicas (guardar, leer).
5. Prueba persistencia de datos.
6. Documenta resultados.
7. Entrega evidencias.
8. Recibe retroalimentación docente.

Recursos de apoyo:

- Computadora
- Información sobre manejo de archivos
- Real, Shared Preferences, SQL (SQLite), XML, XML-RPC, SOAP y otras tecnologías

Práctica 9: Uso de base de datos**Duración:**

04 horas

Procedimiento:

1. Implementa base de datos local (SQLite u otra).
2. Diseña tablas.
3. Implementa operaciones CRUD.
4. Integra base de datos con interfaz.
5. Realiza pruebas.
6. Documenta funcionamiento.
7. Entrega aplicación actualizada.
8. Recibe retroalimentación docente.

Recursos de apoyo:

- Computadora
- Información sobre manejo de archivos
- Real, Shared Preferences, SQL (SQLite), XML, XML-RPC, SOAP y otras tecnologías

Unidad V. Aplicaciones Web para dispositivos móviles**Práctica 10: Consumo de servicios web****Duración:**

04 horas

Procedimiento:

1. Atiende a la explicación sobre APIs REST.
2. Consume un servicio web.
3. Procesa datos JSON/XML.
4. Muestra datos en la interfaz.
5. Maneja errores básicos.
6. Documenta resultados.
7. Entrega evidencia funcional.
8. Recibe retroalimentación docente.

Recursos de apoyo:

- Computadora
- API pública
- IDE

Práctica 11: Integración completa cliente-servidor**Duración:**

08 horas

Procedimiento:

1. Integra servicios web en la aplicación.
2. Sincroniza datos locales y remotos.
3. Optimiza consultas.
4. Implementa actualización dinámica.
5. Realiza pruebas completas.
6. Documenta arquitectura de la solución.
7. Entrega avance funcional robusto.
8. Recibe retroalimentación docente.

Recursos de apoyo:

- Computadora
- API
- IDE

Unidad VI. ensos**Práctica 12: Uso de sensores del dispositivo****Duración:**

08 horas

Procedimiento:

1. Atiende a la explicación sobre sensores móviles.
2. Implementa lectura de acelerómetro u otro sensor.
3. Procesa datos obtenidos.
4. Integra funcionalidad en la aplicación.
5. Diseña caso de uso (ej. movimiento, orientación).
6. Realiza pruebas en dispositivo o emulador.
7. Documenta resultados.
8. Entrega aplicación con funcionalidad contextual.
9. Recibe retroalimentación docente.

Recursos de apoyo:

- Computadora
- Referencia del sitio oficial
- Dispositivo móvil
- IDE

Unidad VII. Bibliotecas gráficas

Práctica 13: Integración de gráficos 2D/3D

Duración:

08 horas

Procedimiento:

1. Atiende a la explicación sobre bibliotecas gráficas.
2. Implementa gráficos 2D o 3D básicos.
3. Integra elementos visuales dinámicos.
4. Relaciona gráficos con datos de la aplicación.
5. Realiza pruebas de rendimiento.
6. Documenta implementación.
7. Entrega aplicación final integrada.
8. Recibe retroalimentación docente.

Recursos de apoyo:

- Computadora
- OpenGL o librería gráfica
- Dispositivo móvil
- IDE

X. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer de forma clara y formal:

- La metodología de trabajo.
- Criterios de evaluación y rúbricas.
- Reglas sobre la calidad y entrega de trabajos académicos.
- Derechos y obligaciones del docente y los estudiantes.

Esto debe quedar registrado y comunicado por medios institucionales (como Blackboard o correo institucional).

Estrategia de enseñanza

- Aprendizaje basado en problemas
- Técnica expositiva
- Método de proyectos

Estrategia de aprendizaje

- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Ejercicios con pseudocódigos
- Diagramas de flujo
- Análisis de problemas científicos

XI. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario, cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	10%
– Tareas y prácticas de laboratorio	45%
– Exposición	05%

– Aplicación para dispositivos móviles	40%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Darwin, I. F. (2020). *Android cookbook: Problems and solutions for Android developers* (2nd ed.). O'Reilly Media.
- Friesen, Jeff (2014). *Learn Java for Android Development*. APress, ISBN 978-1-4302-6454-5 [clásica]
- Gargenta, Marko; Masumi Nakamura (2014). *Learning Android*. O'Reilly, ISBN 978-1-4493-1923-6 [clásica]
- Liles, Steve (2013). *Asynchronous Android*. Packt Publishing ISBN 978-1-78328-688-1 [clásica]
- Matthews, R. (2011). *Beginning Android tablet programming: Starting with Android Honeycomb for tablets*. Springer. <https://libcon.rec.uabc.mx:2180/book/10.1007/978-1-4302-3784-6>
- Morris, Jason (2011). *Android User Interface Development*. Packt Publishing, ISBN 978-1-849514-48-4 [clásica]

Recursos de acceso libre o complementarios externos

- Android Developers. (s.f.). *Android overview*. <https://developer.android.com>
- Bill Phillips, C. S., & Hardy, B. (2021). *Android programming: The big nerd ranch guide* (5th ed.). Big Nerd Ranch.
- Firebase. (s.f.). *Realtime database & Firestore*. <https://firebase.google.com>
- OpenGL. (s.f.). *OpenGL ES documentation*. <https://www.khronos.org/opengl>
- Phillips, B., Stewart, C., Hardy, B., & Marsicano, K. (2019). *Android programming for beginners*. Packt Publishing.

XII. PERFIL DOCENTE

Profesionista con formación en el área de computación o áreas afines; con experiencia en docencia de al menos dos años y en el diseño y desarrollo de aplicaciones para dispositivos móviles. Debe ser una persona creativa, que fomente el trabajo colaborativo y el pensamiento crítico en sus estudiantes.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales Licenciatura en Matemáticas Aplicadas Licenciatura en Física
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Procesamiento Digital de Imágenes
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 04 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:08
Etapas de formación:	Disciplinaria
Carácter:	Optativa
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	José Ángel González Fraga Selene Solorza Calderón
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Esta unidad de aprendizaje tiene como propósito proporcionar al estudiantado los conceptos y fundamentos, así como las principales técnicas del procesamiento digital de imágenes, desarrollando habilidades para aplicar transformaciones lineales y no lineales, filtrado, restauración, segmentación y análisis en diferentes contextos. Asimismo, fomenta actitudes críticas, responsables y creativas que le permitan integrar estas competencias en la solución de problemas relacionados con la computación y disciplinas afines.

Se ubica en la etapa terminal del programa de la Licenciatura en Ciencias Computacionales y pertenece al área de conocimiento de Interacción Hombre Máquina y es de carácter optativo.

Además, se ubica en la etapa terminal de la Licenciatura en Física, en el área de conocimiento Computacional; en la Licenciatura en Matemáticas Aplicadas, es de carácter optativo y pertenece al área de conocimiento de Cómputo Científico.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Evaluar técnicas de procesamiento digital de imágenes, mediante el estudio de las bases teóricas y la implementación práctica de metodologías de filtrado, restauración, color y morfología, para mejorar, analizar e interpretar información contenida en imágenes digitales, con actitud crítica, responsable y creativa.

IV. EVIDENCIA DE APRENDIZAJE

Elabora un portafolio digital que integre los programas desarrollados para el procesamiento digital de imágenes, acompañado del análisis de resultados, conclusiones y referencias empleadas, presentado en formato electrónico, con lenguaje formal y claro, que muestre dominio de los fundamentos, técnicas y aplicaciones vistas en la unidad de aprendizaje.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar y evaluar soluciones de cómputo inteligente aplicando principios de inteligencia artificial, minería de datos, e integración de herramientas computacionales, con el propósito de apoyar la toma de decisiones en situaciones complejas propias de contextos científicos, sociales, industriales o tecnológicos, actuando con responsabilidad ética, pensamiento crítico, y colaboración interdisciplinaria.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Compromiso con la responsabilidad social, Excelencia, Vanguardia e Innovación Social

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos del procesamiento digital de imágenes

Competencia

Examinar los fundamentos del procesamiento digital de imágenes, mediante la revisión de lecturas, análisis de conceptos básicos y resolución de ejercicios, para comprender el proceso de adquisición y formación de imágenes digitales, con actitud crítica y organizada.

Contenido

- 1.1. Introducción, historia y aplicaciones
- 1.2. Componentes de un sistema de procesamiento digital de imágenes
- 1.3. Fundamentos de la imagen digital: percepción visual y espectro electromagnético
- 1.4. Adquisición de imágenes: sensado, muestreo y cuantización
- 1.5. Relaciones entre píxeles y operaciones básicas

Duración

04 horas

Unidad II. Procesamiento en el dominio espacial

Competencia

Aplicar técnicas de procesamiento espacial en imágenes digitales, mediante transformaciones de intensidad, análisis de histogramas y uso de filtros de suavizamiento y realce, para mejorar la calidad visual y resaltar características relevantes, con actitud analítica y responsable.

Contenido

- 2.1. Transformaciones de intensidad
- 2.2. Procesamiento del histograma
- 2.3. Convolución y filtrado espacial
- 2.4. Filtros de suavizamiento (promedios, mediana)
- 2.4. Detección de bordes (Sobel, Prewitt, Canny)

Duración

07 horas

Unidad III. Procesamiento en el dominio de las frecuencias

Competencia

Analizar imágenes digitales en el dominio de la frecuencia, mediante la aplicación de la transformada de Fourier y el diseño de filtros pasa bajas, pasa altas y pasa bandas, para comparar resultados con técnicas espaciales y optimizar el procesamiento, con actitud crítica y organizada.

Contenido

- 3.1. Transformada de Fourier en 2D
- 3.2. Filtrado pasa bajas
- 3.3. Filtrado pasa altas
- 3.4. Filtrado pasa bandas
- 3.5. Comparación de los filtrados en el dominio espacial y el dominio de las frecuencias

Duración

06 horas

Unidad IV. Restauración de imágenes

Competencia

Implementar técnicas de restauración de imágenes, mediante el estudio de modelos de degradación y ruido, así como el uso de filtros espaciales y de frecuencias, para recuperar información visual y mejorar la interpretación de las imágenes, con actitud crítica y responsable.

Contenido

- 4.1. Modelos de degradación y ruido
- 4.2. Filtros de restauración: inverso y Wiener
- 4.3. Restauración en el dominio espacial y en el dominio de las frecuencias

Duración

05 horas

Unidad V. Procesamiento de imágenes a color

Competencia

Evaluar y aplicar técnicas de procesamiento digital de imágenes a color, mediante el análisis de fundamentos perceptuales y modelos de representación, para realizar transformaciones y filtrado en imágenes en falso color y color real, con actitud creativa y responsable.

Contenido

- 5.1 Fundamentos del color y percepción
- 5.2 Modelos de color RGB, CMY, HSV, YCbCr

5.3 5.3 Transformaciones y filtrado en imágenes a color

5.4 Aplicaciones en falso color y color real

Duración

05 horas

Unidad VI. Transformaciones morfológicas

Competencia

Aplicar operadores morfológicos en imágenes binarias y en escala de grises, mediante el uso de elementos estructurantes y transformaciones, para extraer, analizar y segmentar estructuras de interés, con actitud analítica y crítica.

Contenido

6.1 Fundamentos de las transformaciones morfológicas

6.2 Operaciones básicas y compuestas: erosión, dilatación, apertura, cierre

6.3 Transformadas morfológicas

6.4 Aplicaciones en análisis y segmentación de imágenes

Duración

06 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Fundamentos del procesamiento digital de imágenes

Práctica 1: Operaciones básicas con imágenes digitales

Duración

06 horas

Procedimiento

1. Atiende las orientaciones docentes sobre los fundamentos del procesamiento digital de imágenes.
2. Explora el entorno de un software especializado o librerías para el PDI.
3. Importa imágenes en escala de grises y color.
4. Realiza operaciones de recorte, rotación y escalamiento.
5. Implementa operaciones aritméticas entre imágenes.
6. Analiza los resultados obtenidos y los entrega en formato digital para su evaluación.

Recursos de apoyo

- Computadora
- Software especializado como MATLAB, Python u OpenCV
- Imágenes de prueba en escala de grises, indexadas y en el espacio de color RGB

Unidad II. Procesamiento en el dominio espacial

Práctica 2: Transformaciones de intensidad e histogramas

Duración

08 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el tema de transformaciones de intensidad e histogramas.
2. Implementa transformaciones lineales y logarítmicas.
3. Genera e interpreta histogramas.
4. Aplica la ecualización del histograma.
5. Evalúa los resultados con las métricas MSE y PSNR, y los entrega en formato digital para su evaluación.

Recursos de apoyo

- Computadora.
- Software especializado como MATLAB, Python u OpenCV.
- Imágenes médicas, de las ciencias naturales o ingeniería.

Práctica 3: Filtrado espacial y detección de bordes

Duración

08 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el tema de filtrado espacial detección de bordes.
2. Implementa filtros de suavizamiento lineal y de rango.
3. Aplica filtros de realce.
4. Realiza detección de bordes con Sobel, Prewitt y Canny.
5. Comparar el desempeño de cada método con las métricas usuales, y lo entrega en formato digital para su evaluación.

Recursos de apoyo

- Computadora
- Software especializado como MATLAB, Python u OpenCV
- Imágenes de las ciencias naturales o ingeniería con y sin ruido

Unidad III. Procesamiento en el dominio de las frecuencias

Práctica 4: Transformada de Fourier y filtrado en el dominio de las frecuencias

Duración

08 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el tema de filtrado en el dominio de las frecuencias.
2. Calcula la transformada rápida 2D de Fourier (FFT) de una imagen.
3. Visualiza el espectro de magnitud y el espectro de fase.
4. Implementa filtros pasa bajas, pasa altas y pasa bandas.
5. Compara los resultados que se obtuvieron con los filtros en el dominio de las frecuencias con los obtenidos con los filtros espaciales, usando métricas estándar.
6. Entrega los resultados en formato digital para su evaluación.

Recursos de apoyo

- Computadora
- Software especializado como MATLAB, Python u OpenCV
- Imágenes sintéticas y de las ciencias naturales o ingeniería

Unidad IV. Restauración de imágenes

Práctica 5: Modelos de degradación y restauración

Duración

08 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el tema de modelos de degradación y restauración.
2. Simula la degradación de una imagen al incluir ruido con diversas densidades de probabilidad.
3. Aplica filtros para eliminar la presencia de ruido en imágenes digitales.
4. Simula la degradación de una imagen usando transformaciones de desenfoque a partir de modelos lineales.
5. Implementa filtros de restauración: inverso y Wiener.
6. Evaluar la recuperación de la imagen digital de forma cuantitativa usando el PSNR.
7. Entrega los resultados en formato digital para su evaluación.

Recursos de apoyo

- Computadora

- Software especializado como MATLAB, Python u OpenCV
- Imágenes sintéticas y de las ciencias naturales o ingeniería

Unidad V. Procesamiento de imágenes a color

Práctica 6: Transformación de modelos de color

Duración

08 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el tema de procesamiento de imágenes a color.
2. Transforma una imagen en el espacio de color RGB al espacio de color HSV y al YCbCr, y viceversa.
3. Manipula canales de color de manera individual.
4. Implementa técnicas de realce en cada modelo de color.
5. Entrega los resultados en formato digital para su evaluación.

Recursos de apoyo

- Computadora
- Software especializado como MATLAB, Python u OpenCV
- Imágenes a color de las ciencias naturales o ingeniería

Unidad VI. Transformaciones morfológicas

Práctica 7: Transformadas morfológicas y aplicaciones

Duración

08 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el tema de las transformaciones morfológicas.
2. Aplica operadores morfológicos básicos con diferentes elementos estructurantes.
3. Implementa los algoritmos: *Boundary Extraction, Hole Filling, Extraction of Connected Components, Convex Hull, Thinning, Thickening, Skeletons, Pruning*.
4. Analiza los efectos del ruido y estructuras sobre las imágenes.
5. Entrega los resultados en formato digital para su evaluación.

Recursos de apoyo

- Computadora
- Software especializado como MATLAB, Python u OpenCV
- Imágenes de documentos, de las ciencias naturales o ingeniería

Práctica 8: Proyecto Integrador de PDI

Duración

10 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el planteamiento del problema a desarrollar:
 - a. Selecciona un conjunto de imágenes (médicas, satelitales, industriales, de seguridad, etc.).
 - b. Plantea el objetivo del procesamiento digital de la imagen (mejora visual, eliminación de ruido, segmentación de estructuras, etc.).
2. Aplica preprocesamiento y fundamentos
 - a. Aplica operaciones básicas, transformaciones de intensidad e histogramas.
3. De ser necesario realiza filtrado espacial o de frecuencias
 - a. Implementa filtros de suavizamiento y realce.
 - b. Compara el desempeño de los filtros espaciales y de los filtros en el dominio de las frecuencias.
4. De ser necesario aplica algoritmos de restauración y procesamiento en color
 - a. Introduce degradación y aplica filtros de restauración.
 - b. Procesa imágenes en color o genera imágenes en falso color.
5. Si aplica, usa morfología y segmentación
 - a. Utiliza operadores morfológicos para la extracción de características.
 - b. Aplica segmentación para separar regiones de interés.
6. Elabora la documentación y presentación
 - a. Integra resultados en un portafolio electrónico con código, imágenes originales, procesadas y el análisis de los resultados.
 - b. Realiza presentación oral con demostración de resultados.

Recursos de apoyo

- Computadora
- Acceso a internet
- Lenguaje de programación MATLAB o Python (con librerías OpenCV, scikit-image, NumPy, Matplotlib)
- Conjuntos de imágenes (médicas, satelitales, industriales o libres de bases públicas como ImageNet, COCO, Kaggle Datasets)
- Software de control de versiones (Git/GitHub/GitLab)
- Plataforma de trabajo colaborativo (Google Colab, Jupyter Notebooks o MATLAB Live Scripts)
- Bibliografía y manuales técnicos (González & Woods, documentación de librerías, artículos científicos)

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Técnica expositiva
- Debates
- Discusión guiada
- Instrucción guiada
- Foros

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Visitas a campo
- Mapas mentales
- Mapas conceptuales
- Cuadros comparativos
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Dos evaluaciones parciales	30%
– Prácticas de laboratorio	35%
– Proyecto Integrador	35%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Gonzalez, R. C., & Woods, R. E. (2018). *Digital image processing* (4th ed.). Pearson.
https://catalogocimarron.uabc.mx/cgi-bin/koha/opac-detail.pl?biblionumber=263740&query_desc=kw%2Cwrdl%3A%20%22digital%20image%20processing%20%22 [classic]
- Gonzalez, R. C., Woods, R. E., & Eddins, S. L. (2010). *Digital image processing using MATLAB* (2nd ed.). Gatesmark Publishing. [classic]
- Najarian, K., & Splinter, R. (2012). *Biomedical signal and image processing*. Taylor & Francis. [classic]
- Russ, J., & Neal, F. (2017). *The image processing handbook* (7th ed.). CRC Press. [classic]
- Zhang, Y. J. (2024). *3D computer vision: Foundations and advanced methodologies*. Springer.
https://catalogocimarron.uabc.mx/cgi-bin/koha/opac-detail.pl?biblionumber=260183&query_desc=kw%2Cwrdl%3A%20Digital%20image%20processing

Recursos de acceso libre o complementarios externos

- MATLAB, *Image Processing and Computer Vision*. (s. f.). MATHWORKS.
<https://la.mathworks.com/help/overview/image-processing-and-computer-vision.html>
- OpenCV. (s. f.). *OpenCV tutorials*. OpenCV.
https://docs.opencv.org/4.x/d9/df8/tutorial_root.html
- Szeliski, R. (2022). *Computer vision: Algorithms and applications*. Springer Nature.
<https://szeliski.org/Book/>

XII. PERFIL DOCENTE

Contar con formación profesional en áreas afines a la Inteligencia Artificial, como Ciencias Computacionales, Ingeniería en Computación, Matemáticas Aplicadas o Física, preferentemente con estudios de posgrado. Debe poseer experiencia práctica en el desarrollo e implementación de algoritmos para el procesamiento digital de imágenes. Así como tener conocimiento en lenguajes y entornos de programación como Python o MATLAB, uso de librerías especializadas como OpenCV, scikit-image, numpy y matplotlib.

Debe contar con competencias pedagógicas que le permitan fomentar el aprendizaje activo, la resolución de problemas, la creatividad y el trabajo colaborativo, así como guiar a sus estudiantes en el diseño de proyectos prácticos e innovadores con responsabilidad y pensamiento crítico.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Emprendedores
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 02 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:06
Etapas de formación:	Terminal
Carácter:	Optativa
Requisito:	Ninguno.
Modalidad de impartición:	Presencial
Equipo de diseño	Adrián Enciso Almanza Julio Enrique Valencia Suarez
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La asignatura de **Emprendedores**, ubicada en la etapa terminal y de carácter optativo, integra teoría, práctica y desarrollo de proyectos. Se estructura en cuatro unidades: la primera aborda los fundamentos del emprendimiento, las características del emprendedor y la identificación de problemas como punto de partida; la segunda se centra en la generación, validación y experimentación de ideas de negocio mediante metodologías ágiles; la tercera profundiza en el análisis del mercado, la propuesta de valor, el diseño de estrategias de comercialización y el desarrollo del modelo de negocio; y la cuarta integra la gestión organizacional, legal y financiera para consolidar la viabilidad del proyecto emprendedor.

Esta asignatura permite comprender y aplicar principios, técnicas y herramientas modernas de emprendimiento, con la finalidad de que las y los estudiantes puedan diseñar, validar y estructurar modelos de negocio innovadores que respondan a necesidades sociales y del mercado, fomentando la actitud crítica, creatividad y responsabilidad. Pertenece al área de conocimiento Entorno Social.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Diseñar un modelo de negocio innovador mediante el uso de metodologías, técnicas y herramientas de emprendimiento, para generar productos y/o servicios que respondan a las problemáticas y necesidades de la población, con actitud crítica, creativa y responsabilidad social.

IV. EVIDENCIA DE APRENDIZAJE

Elabora un **modelo de negocio documentado** en el que se integren los aspectos necesarios para la creación de una empresa. El documento deberá incluir: identificación del problema y cliente, propuesta de valor, desarrollo de prototipo o producto mínimo viable (MVP), análisis de mercado y competencia, estrategias de comercialización, estructura organizacional y legal, proyecciones financieras y plan de crecimiento.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Pensamiento Crítico

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos del emprendimiento y generación de ideas

Competencia

Analizar oportunidades de negocio, mediante la aplicación del pensamiento crítico y creatividad al identificar los problemas y necesidades del cliente, para generar ideas innovadoras en contextos tecnológicos, científicos, sociales y cotidianos, actuando con ética y responsabilidad social.

Contenido

- 1.1. Empezar y características del emprendedor.
- 1.2. Creatividad e innovación para la generación de ideas de negocio.
- 1.3. Identificación de problemas y necesidades del cliente.
- 1.4. Definición del perfil y segmentos de cliente.
- 1.5. Validación del problema con el cliente.
- 1.6. Propuesta de valor.

Duración

6 horas

Unidad II. Validación y desarrollo de la solución

Competencia

Validar las ideas de negocio mediante el uso de experimentos, desarrollo de prototipos y contacto con clientes, para determinar su viabilidad técnica y de mercado, con actitud crítica, creativa y orientada a la mejora continua.

Contenido

- 2.1. Modelo de negocio (Business Model Canvas).
- 2.2. Desarrollo de prototipo y MVP.
- 2.3. Validación de la idea solución con clientes reales.
- 2.4. Análisis de la competencia y del entorno.
- 2.5. Análisis del problema considerando las tendencias y evolución de la industria.
- 2.6. Análisis de la cadena de valor y barreras a la entrada

Duración

12 horas

Unidad III. Mercado, organización y aspectos legales

Competencia

Diseñar estrategias de comercialización y administración mediante el uso de herramientas de investigación de mercado, planeación empresarial y cumplimiento normativo, para establecer la estructura organizacional y legal que garantice la operación formal, competitiva y sostenible de la empresa, con compromiso ético y responsabilidad social.

Contenido

- 3.1. Investigación y análisis de mercado.
- 3.2. Segmentación de mercado.
- 3.3. Estrategia de marketing y ventas.
- 3.4. Estructura organizacional y roles principales.
- 3.5. Aspectos legales básicos: constitución de empresa, trámites fiscales y laborales, propiedad intelectual.

Duración

8 horas

Unidad IV. Finanzas e integración del modelo de negocio

Competencia

Diseñar estrategias financieras y de gestión mediante el uso de proyecciones económicas, análisis de rentabilidad y alternativas de financiamiento, para integrar un modelo de negocio completo que sustente la viabilidad económica y la sostenibilidad del proyecto, con honestidad, objetividad y visión de largo plazo.

Contenido

- 4.1. Sistema contable básico y flujo de caja.
- 4.2. Proyecciones financieras simplificadas.
- 4.3. Indicadores clave de desempeño (KPI) para startups.
- 4.4. Estrategias de financiamiento y costos.
- 4.5. Presentación del modelo de negocio.

Duración

6 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Fundamentos del emprendimiento y generación de ideas

Práctica 1: Mapeo de problemas y necesidades del cliente

Duración: 4 horas

Procedimiento:

1. Se reúne en equipo y eligen un segmento de mercado.
2. Utiliza entrevistas, observación o encuestas para identificar problemas reales.
3. Elabora un mapa de problemas del cliente.
4. Presenta sus hallazgos al grupo para retroalimentación.

Recursos de apoyo:

- Computadora
- Acceso a internet
- Plantilla de evaluación de las ideas de negocio.
- Guía de entrevista para identificación de problemas
- Videos y artículos: “Cómo detectar problemas de tus clientes”

Práctica 2: Construcción de propuesta de valor con Lienzo Canvas

Duración: 2 horas

Procedimiento:

1. Atiende a las orientaciones en la introducción y revisión del Lienzo Canvas.
2. Llena el lienzo a partir del problema identificado.
3. Participa en una evaluación cruzada con otro equipo para detectar áreas de mejora.
4. Atiene a la revisión final con asesoría de su docente.

Recursos de apoyo:

- Computadora
- Acceso a internet
- Plantilla de Lienzo Canvas
- Videos explicativo del llenado del lienzo Canvas
- Videos de ejemplos de lienzos Canvas.

Unidad II. Validación y desarrollo de la solución

Práctica 3: Desarrollo de prototipo o MVP (Producto Mínimo Viable)

Duración: 6 horas

Procedimiento:

1. Atiende a la Introducción breve al concepto de MVP.
2. Realiza el diseño y construcción rápida de un prototipo funcional (puede ser digital, físico o visual).
3. Utiliza la prueba del MVP con al menos 3 clientes reales o simulados.
4. Usa la recopilación de retroalimentación para iterar.

Recursos de apoyo:

- Computadora
- Acceso a internet
- Materiales de prototipado (cartón, papel, apps de diseño, etc.)
- Herramientas como Figma, Canva, Mockflow, Marvel
- Video: “Cómo crear tu MVP”

Práctica 4: Análisis del entorno y competencia

Duración: 4 horas

Procedimiento:

1. Atiende a la Introducción breve sobre competencia.
2. En equipo (emprendimiento) investiga a tres competidores directos e indirectos.
3. Elabora una matriz comparativa con factores clave (precio, diferenciador, mercado).
4. Construye un mapa de posicionamiento.
5. Presenta sus hallazgos.

Recursos de apoyo:

- Computadora

- Acceso a internet
- Plantilla de análisis competitivo
- Herramientas: Google Trends, SEMrush (versión free), páginas web de competidores
- Lectura breve: “Análisis de competencia para emprendedores”

Unidad III. Mercado, organización y aspectos legales

Práctica 5: Segmentación de mercado y perfil de cliente ideal.

Duración: 4 horas

Procedimiento:

1. Aplica los criterios de segmentación (geográficos, demográficos, psicográficos, conductuales).
2. Crea uno o dos Buyer Personas por equipo.
3. Elabora la representación visual y presentación del perfil al grupo.

Recursos de apoyo:

- Computadora
- Acceso a internet
- Plantilla de Buyer Persona
- Herramienta: MakeMyPersona (HubSpot gratuito)
- Guía de segmentación del mercado

Práctica 6: Simulación de constitución legal de una empresa

Duración: 4 horas

Procedimiento:

1. Realiza previo a la práctica una investigación de los pasos legales para constituir una empresa en México.
2. Elabora una simulación del llenado de formatos (acta constitutiva, SAT, IMSS, marca, etc.).
3. Participa en la ronda de preguntas y reflexiona sobre la importancia del marco legal.

Recursos de apoyo:

- Computadora
- Acceso a internet
- Sitio web del SAT y tuempresa.gob.mx
- Formatos descargables de ejemplo (INEGI, INPI, IMSS)

- Video: “Cómo registrar legalmente tu empresa en México”

Unidad IV. Finanzas e integración del modelo de negocio

Práctica 7: Elaboración de flujo de caja proyectado

Duración: 4 horas

Procedimiento:

1. Atiende a las orientaciones sobre Introducción al flujo de efectivo y su importancia.
2. Realiza la construcción de una tabla de ingresos, egresos y saldo mensual.
3. Elabora el análisis de rentabilidad simple (punto de equilibrio, margen de utilidad).
4. Elabora la simulación de decisiones financieras: qué pasa si bajan ventas o suben costos.

Recursos de apoyo:

- Computadora
- Acceso a internet
- Plantilla en Excel o Google Sheets
- Video tutorial: “Cómo hacer un flujo de caja en Excel”
- Lectura: conceptos básicos de contabilidad para emprendedores

Práctica 8: Pitch de negocio y retroalimentación final

Duración: 4 horas

Procedimiento:

1. Participa en la preparación de presentación tipo pitch.
2. Realiza un ensayo con apoyo del docente.
3. Participa en exposición ante jurado o grupo (puede ser grabada).
4. Recibe la retroalimentación con rúbrica de evaluación.

Recursos de apoyo:

- Computadora
- Acceso a internet
- Plantilla de Pitch Deck.
- Rúbrica de pitch.
- Ejemplos de pitch.

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- **Método de proyectos** – guía la creación del modelo de negocio completo.
- **Aprendizaje basado en problemas** – plantea desafíos reales de emprendimiento.
- **Estudio de caso** – analiza ejemplos de empresas reales o startups.
- **Discusión guiada** – promueve el análisis crítico de ideas y estrategias.
- **Técnica expositiva** (*selectiva y contextual*) – introduce conceptos clave (CANVAS, finanzas, trámites legales).
- **Instrucción guiada** – para prácticas como llenado del lienzo CANVAS o uso de herramientas de validación.
- **Foros** – para discusión asincrónica sobre tendencias, análisis de casos o retroalimentación entre pares.
- **Exposición de resoluciones de casos prácticos** – los estudiantes presentan avances o soluciones.

Estrategia de aprendizaje

- **Investigación documental** – para análisis de mercado, competencia, aspectos legales.
- **Investigación empírica** – encuestas, entrevistas y validación con clientes reales.
- **Trabajo colaborativo** – desarrollo del modelo de negocio en equipo.
- **Desarrollo de prototipo o MVP** – creación y prueba de una solución inicial.
- **Creación de infografías** – para presentar la propuesta de valor, segmentación o resultados.
- **Mapas conceptuales** – organización de componentes del modelo de negocio.
- **Cuadros comparativos** – análisis de competencia, precios, canales.
- **Participación en debates y foros** – defensa de ideas, comparación de enfoques.
- **Presentación de pitch de negocio** – comunicar y defender su propuesta final.

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	20%
– Prácticas y Ejercicios	20%
– Modelo de negocios	60%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Alcaraz Rodríguez, R. (2011). *El Emprendedor de Éxito*. (4a ed.). Editorial McGraw-Hill. [clásica]
- Blank, S. y Bob, D. (2013). *El manual del emprendedor: La guía paso a paso para crear una gran empresa*. Grupo Planeta Spain.[clásica]
- Osterwalder, A., Yves P., Smith, A., Greg, B. y Papadakos, P. (2015). *Diseñando la propuesta de valor*. Centro Libros PAPP. [clásica]
- Ries, E. (2012). *El método Lean Startup: Cómo crear empresas de éxito utilizando la innovación continua*. Grupo Planeta. [clásica]

Recursos de acceso libre o complementarios externos

- Ferrell, O.C. y Hirt, Geoffrey. (2004). *Introducción a los Negocios en un Mundo Cambiante*, (4a ed.). Editorial McGraw-Hill. [clásica]
- Maurya, A. (2012). *Running Lean*, (2nd ed.). O'REILLY.
- Varela Villegas, R. (2008). *Innovación empresarial arte y ciencia en la creación.*, Prentice Hall. [clásica]
- IMPI Búsquedas tecnológicas (http://www.impi.gob.mx/wb/IMPI/herramientas_del_sitio)
- Organización Mundial de la Propiedad Intelectual (OMPI). Sus siglas en Inglés son WIPO ([http:// www.wipo.int/portal/es/](http://www.wipo.int/portal/es/))

XII. PERFIL DOCENTE

Licenciatura en Ciencias Computacionales, Ingeniería en Sistemas Computacionales o área afín, preferentemente con posgrado en Emprendimiento, Innovación o Administración. Cuenta con conocimientos en metodologías de emprendimiento, innovación y modelos de negocio, así como experiencia práctica en el ámbito empresarial o en el acompañamiento de proyectos emprendedores. Posee al menos dos años de experiencia docente y facilita el aprendizaje activo, promoviendo el pensamiento crítico y creativo, la identificación de problemas reales y el desarrollo de modelos de negocio viables, actuando con ética, responsabilidad social y enfoque interdisciplinario.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales Licenciatura en Matemáticas Aplicadas
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Visión por Computadora
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 04 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:08
Etapas de formación:	Terminal
Carácter:	Optativa
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	José Ángel González Fraga Selene Solorza Calderón
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La visión por computadora es un área fundamental de la inteligencia artificial, ya que permite a las máquinas interpretar y comprender imágenes y videos del mundo real para resolver problemas complejos en contextos científicos, industriales y sociales. Esta unidad de aprendizaje tiene como propósito proporcionar al estudiante los conceptos, fundamentos y técnicas esenciales de la visión por computadora, de manera que desarrolle habilidades para aplicar métodos de extracción de características, geometría epipolar, segmentación, reconocimiento y aprendizaje profundo en imágenes digitales. Asimismo, fomenta actitudes críticas, responsables y creativas que le permitan integrar estas competencias en el diseño de soluciones innovadoras en diversas áreas del conocimiento y la tecnología.

La asignatura se ubica en la etapa terminal del programa de Licenciatura en Ciencias Computacionales, pertenece al área de conocimiento de Interacción Hombre Máquina y es de carácter optativo.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Evaluar técnicas de visión por computadora, mediante el estudio de las bases teóricas y la implementación práctica de metodologías de características locales, geometría epipolar, segmentación, reconocimiento y aprendizaje profundo, para analizar e interpretar información contenida en imágenes y videos, con actitud crítica, responsable e innovadora.

IV. EVIDENCIA DE APRENDIZAJE

Desarrolla un proyecto integrador de visión por computadora que incluya el planteamiento del problema, la selección de bases de datos de imágenes, la implementación de una o varias de las técnicas abordadas en la unidad de aprendizaje (características locales, geometría epipolar, segmentación, reconocimiento o aprendizaje profundo), el análisis de resultados y las conclusiones. El proyecto deberá incluir un reporte y una presentación o infografía en alguna paquetería como PowerPoint, canva, genially, beamer, que muestre el funcionamiento de la solución y su impacto en un contexto aplicado, con lenguaje formal, claro y fundamentado en referencias bibliográficas.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Esta unidad de aprendizaje promueve competencias genéricas de impacto en los diferentes perfiles de egreso de los programas educativos de la **Facultad de Ciencias**.

Licenciatura en Ciencias Computacionales: Diseñar y evaluar soluciones de cómputo inteligente aplicando principios de inteligencia artificial, minería de datos, e integración de herramientas computacionales, con el propósito de apoyar la toma de decisiones en situaciones complejas propias de contextos científicos, sociales, industriales o tecnológicos, actuando con responsabilidad ética, pensamiento crítico, y colaboración interdisciplinaria.

Licenciatura en Matemáticas Aplicadas: Adaptar modelos matemáticos determinísticos y estocásticos mediante el planteamiento de hipótesis, la construcción de relaciones funcionales y la implementación de métodos de simulación y optimización, para la resolución de problemas en los sectores económico, social, tecnológico y científico, de manera colaborativa, crítica y creativa.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Compromiso con la Responsabilidad Social; Excelencia y Vanguardia e Innovación Social.

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos de la visión por computadora

Competencia

Identificar los fundamentos de la visión por computadora, mediante el estudio de la formación de imágenes y el análisis de diversas aplicaciones, para reconocer el potencial de la visión en distintos contextos, con actitud crítica y responsable.

Contenido

- 1.1. Historia y aplicaciones de la visión por computadora.
- 1.2. Formación de imágenes: geometría de la cámara, proyecciones y calibración.
- 1.3. Flujo de trabajo en visión por computadora.
- 1.4. Técnicas básicas de procesamiento de imágenes.

Duración

05 horas

Unidad II. Características locales y correspondencia

Competencia

Aplicar técnicas de detección y descripción de características locales, mediante el uso de puntos de interés y descriptores, para realizar tareas de registro y coincidencia de imágenes, con actitud analítica, organizada y comprometida.

Contenido

- 2.1. Detección de esquinas y puntos de interés usando las metodologías Harris y Shi-Tomasi.
- 2.2. Descriptores locales usando las metodologías SIFT y ORB.
- 2.3. Correspondencia de características.
- 2.4. Aplicaciones en registro y mosaicos de imágenes.

Duración

06 horas

Unidad III. Geometría epipolar y reconstrucción 3D

Competencia

Analizar la geometría epipolar y técnicas de reconstrucción, mediante el estudio de correspondencias estéreo y movimiento, para modelar estructuras tridimensionales a partir de imágenes, con actitud crítica y responsable.

Contenido

- 3.1. Estéreo y disparidad.
- 3.2. Epipolaridad, matrices fundamental y esencial.
- 3.3. Estructura a partir del movimiento SfM.
- 3.4. Introducción a la reconstrucción 3D.

Duración

06 horas

Unidad IV. Segmentación y reconocimiento

Competencia

Implementar metodologías básicas de segmentación y reconocimiento, mediante el uso de técnicas de agrupamiento, bordes y detección de objetos, para identificar regiones y patrones en imágenes, con actitud analítica y responsable.

Contenido

- 4.1. Umbralización y clustering con K-means, GMM.
- 4.2. Segmentación basada en bordes y regiones.
- 4.3. Detección de objetos.
- 4.4. Introducción al reconocimiento facial.

Duración

05 horas

Unidad V. Aprendizaje profundo en visión

Competencia

Aplicar arquitecturas de redes convolucionales, mediante el uso de software especializado, para resolver problemas de clasificación y detección de objetos, con actitud innovadora, responsable y ética profesional.

Contenido

- 5.1. Redes neuronales convolucionales y arquitecturas base.
- 5.2. Aprendizaje por transferencia en visión.
- 5.3. Detección de objetos usando YOLO y SSD.
- 5.4. Aplicaciones en visión moderna.

Duración

06 horas

Unidad VI. Aplicaciones avanzadas

Competencia

Analizar aplicaciones avanzadas de la visión por computadora, mediante el estudio de casos en robótica, vehículos autónomos, imágenes médicas y realidad aumentada, para reconocer su impacto en la sociedad, con actitud crítica, responsable y ética profesional.

Contenido

- 6.1. Visión en robótica.
- 6.2. Visión en vehículos autónomos.
- 6.3. Visión en imágenes médicas.
- 6.4. Realidad aumentada y realidad virtual.

Duración

04 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Fundamentos de la visión por computadora

Práctica 1: Formación de imágenes y cámara pinhole

Duración

06 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el modelo de cámara pinhole.
2. Implementa la proyección de puntos 3D en coordenadas 2D.
3. Varía los parámetros de la cámara (focal, centro óptico) y analiza los cambios en la imagen.
4. Realiza una calibración básica con un patrón (chessboard).
5. Documenta el conjunto de imágenes procesadas con las metodologías indicadas y lo entrega en formato digital para su evaluación.

Recursos de apoyo

- Computadora

- Software especializado como MATLAB o Python con OpenCV
- Imágenes sintéticas de calibración

Unidad II. Características locales y correspondencia

Práctica 2: Detección de puntos de interés

Duración

06 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el tema de características locales.
2. Utiliza imágenes naturales en escala de grises.
3. Implementa el detector de esquinas Harris.
4. Implementa el detector Shi-Tomasi.
5. Visualiza los puntos detectados con ambas técnicas.
6. Compara los puntos de interés detectados con ambas técnicas.
7. Documenta la visualización de los puntos de interés sobre las imágenes de prueba y lo entrega en formato digital para su evaluación.

Recursos de apoyo

- Computadora.
- Software como MATLAB o Python con OpenCV.
- Imágenes de las ciencias naturales o ingeniería.

Práctica 3: Descriptores y correspondencia

Duración

06 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el tema de descriptores.
2. Utiliza al menos dos imágenes de la misma escena capturada desde distintas perspectivas.
3. Extrae descriptores locales con al menos dos de las siguientes técnicas SIFT, SURF y ORB.
4. Aplica un algoritmo de emparejamiento: Brute Force o FLANN.
5. Analiza las correspondencias usando la prueba del cociente o razón de Lowe.
6. Construye un mosaico de imágenes a partir de las correspondencias válidas.
7. Documenta el mosaico de imágenes obtenido y lo entrega en formato digital para su evaluación.

Recursos de apoyo

- Computadora
- Software como MATLAB o Python con OpenCV

- Conjunto de imágenes de escenas

Unidad III. Geometría epipolar y reconstrucción 3D

Práctica 4: Estéreo y disparidad

Duración

06 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el tema de mapas de disparidad.
2. Utiliza un par de imágenes estéreo.
3. Calcula la correspondencia entre píxeles.
4. Obtén el mapa de disparidad.
5. Visualiza el mapa de profundidad resultante.
6. Documenta el mapa de profundidad resultante y lo entrega en formato digital para su evaluación.

Recursos de apoyo

- Computadora
- Software como MATLAB o Python con OpenCV
- Base de datos de imágenes estéreo (KITTI o Middlebury)

Práctica 5: Epipolaridad y reconstrucción

Duración

06 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el tema de reconstrucción 3D.
2. Utiliza dos imágenes con diferente punto de vista.
3. Calcula la matriz fundamental con correspondencias de puntos.
4. Verifica la condición epipolar dibujando líneas epipolares.
5. Reconstruye una nube de puntos básica (SfM).
6. Documenta la nube de puntos resultante y lo entrega en formato digital para su evaluación.

Recursos de apoyo

- Computadora
- Software como MATLAB o Python con OpenCV
- imágenes de escenas estructuradas

Unidad IV. Segmentación y reconocimiento

Práctica 6: Segmentación básica

Duración

06 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el tema de segmentación.
2. Utiliza imágenes en color y escala de grises.
3. Implementa umbralización global y adaptativa.
4. Aplica segmentación basada en color con K-means.
5. Compara resultados de los métodos.
6. Documenta el análisis de resultados y lo entrega en formato digital para su evaluación.

Recursos de apoyo

- Computadora
- Software como MATLAB o Python con OpenCV y librerías especializadas scikit-learn, scikit image
- Imágenes biomédicas o naturales

Práctica 7: Detección de objetos clásicos

Duración

06 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el tema de detección de objetos.
2. Extrae características HOG de un conjunto de imágenes.
3. Entrena un clasificador SVM con ejemplos positivos y negativos.
4. Aplica el detector en imágenes de prueba.
5. Evalúa precisión y recall de la detección.
6. Documenta el análisis de resultados y lo entrega en formato digital para su evaluación.

Recursos de apoyo

- Computadora
- Software como MATLAB o Python con OpenCV y librerías especializadas scikit-learn, scikit image
- Base de datos de imágenes de objetos

Unidad V. Aprendizaje profundo en visión

Práctica 8: Clasificación con CNNs

Duración

06 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el tema de clasificación.
2. Selecciona una base de datos pequeña (MNIST o CIFAR-10).
3. Define una CNN sencilla en PyTorch o TensorFlow.
4. Entrena el modelo con un conjunto de entrenamiento.
5. Evalúa la precisión en un conjunto de pruebas.
6. Documenta el análisis de resultados con las métricas estándar y gráficas, y lo entrega en formato digital para su evaluación.

Recursos de apoyo

- Computadora con GPU (local o Colab)
- Software como MATLAB o Python con OpenCV y librerías especializadas (PyTorch/TensorFlow)
- Bases de datos MNIST o CIFAR-10

Práctica 9: Detección con aprendizaje profundo

Duración

06 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el tema de detección.
2. Utiliza un modelo pre entrenado (YOLO, SSD).
3. Emplea aprendizaje por transferencia con una base de datos reducida.
4. Evalúa el detector en imágenes nuevas.
5. Mide la precisión y el *recall* en el conjunto de prueba.
6. Documenta el análisis de resultados con las métricas estándar y gráficas, y lo entrega en formato digital para su evaluación.

Recursos de apoyo

- Computadora con GPU (local o Colab)
- Software como MATLAB o Python, con librerías especializadas como (PyTorch/TensorFlow)
- Datasets COCO u objetos propios

Unidad VI. Aplicaciones avanzadas

Práctica 10: Proyecto Integrador de Visión por computadora

Duración

10 horas

Procedimiento

1. Atiende las orientaciones docentes sobre el planteamiento del problema a desarrollar.
2. Plantea un problema (ej. visión médica, vehículos autónomos, robótica).
3. Selecciona una base de datos pública o propia.
4. Aplica técnicas comprendidas en esta unidad de aprendizaje (características, geometría, segmentación, CNNs).
5. Elabora un reporte que incluya el planteamiento del problema, la selección de bases de datos de imágenes, la implementación de una o varias de las técnicas abordadas en la unidad de aprendizaje (características locales, geometría epipolar, segmentación, reconocimiento o aprendizaje profundo), el análisis de resultados y las conclusiones.
6. Elabora una presentación o infografía en alguna paquetería como powerpoint, canva, genially, beamer.
7. Realiza la presentación frente a grupo.

Recursos de apoyo

- Computadora
- Acceso a internet
- Lenguaje de programación MATLAB o Python (con librerías OpenCV, PyTorch/TensorFlow)
- Conjuntos de imágenes (médicas, satelitales, industriales o libres de bases públicas como ImageNet, COCO, Kaggle Datasets)
- Software de control de versiones (Git/GitHub/GitLab)
- Plataforma de trabajo colaborativo (Google Colab, Jupyter Notebooks o Matlab Live Scripts)
- Bibliografía y manuales técnicos (documentación de librerías, artículos científicos)

IX. MÉTODO DE TRABAJO

Encuadre

El primer día de clase, cada docente debe establecer de forma clara y formal:

- La metodología de trabajo.
- Criterios de evaluación y rúbricas.
- Reglas sobre la calidad y entrega de trabajos académicos.
- Derechos y obligaciones del docente y los estudiantes.

Esto debe quedar registrado y comunicado por medios institucionales (como Blackboard o correo institucional).

Estrategia de enseñanza

- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Técnica expositiva
- Debates
- Discusión guiada
- Instrucción guiada

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Cuadros comparativos
- Infografías

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Dos evaluaciones parciales	30%
– Prácticas de laboratorio	35%
– Proyecto Integrador	35%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

Ekman, M. (2021). *Learning deep learning: Theory and practice of neural networks, computer vision, natural language processing, and transformers using TensorFlow*. Addison-Wesley Professional.

- Gonzalez, R. C., & Woods, R. E. (2018). *Digital image processing* (4th ed.). Pearson.
https://catalogocimarron.uabc.mx/cgi-bin/koha/opac-detail.pl?biblionumber=263740&query_desc=kw%2Cwrdl%3A%20%22digital%20image%20processing%20%22
- Gonzalez, R. C., Woods, R. E., & Eddins, S. L. (2010). *Digital image processing using MATLAB* (2nd ed.). Gatesmark Publishing.
- Li, S. Z., Jain, A. K., & Deng, J. (2024). *Handbook of face recognition* (3rd ed.). Springer.
<https://libcon.rec.uabc.mx:2496/book/10.1007/978-3-031-43567-6>
- Russ, J., & Neal, F. (2017). *The image processing handbook* (7th ed.). CRC Press.
- Zhang, Y. J. (2024). *3D computer vision: Foundations and advanced methodologies*. Springer.
https://catalogocimarron.uabc.mx/cgi-bin/koha/opac-detail.pl?biblionumber=260183&query_desc=kw%2Cwrdl%3A%20Digital%20image%20processing

Recursos de acceso libre o complementarios externos

- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.
<https://www.deeplearningbook.org/>
- MATLAB, *Image Processing and Computer Vision*. (s. f.).
<https://la.mathworks.com/help/overview/image-processing-and-computer-vision.html>
- OpenCV. (s. f.). *OpenCV tutorials*. https://docs.opencv.org/4.x/d9/df8/tutorial_root.html
- Szeliski, R. (2022). *Computer vision: Algorithms and applications*. Springer Nature.
<https://szeliski.org/Book/>
- Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2023). *Dive into deep learning*. Cambridge University Press. <https://d2l.ai/index.html>

XII. PERFIL DOCENTE

La persona docente que imparta la asignatura de Visión por Computadora debe contar con formación profesional en áreas afines a la Inteligencia Artificial, como Ciencias Computacionales, Ingeniería en Computación, Matemáticas Aplicadas o Física, preferentemente con estudios de posgrado. Debe poseer experiencia práctica en el desarrollo e implementación de algoritmos de visión por computadora, incluyendo formación de imágenes, detección y descripción de características locales, geometría epipolar, segmentación, reconocimiento de objetos y técnicas de aprendizaje profundo aplicadas a imágenes y videos. Así como tener conocimiento en lenguajes y entornos de programación como Python o MATLAB, uso de librerías y frameworks especializados como OpenCV, scikit-learn, scikit-image, PyTorch y TensorFlow. Debe contar con competencias pedagógicas que le permitan fomentar el aprendizaje activo, la resolución de problemas, la creatividad y el trabajo colaborativo, así como guiar a los estudiantes en el diseño de proyectos prácticos e innovadores con responsabilidad y pensamiento crítico.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Diseño de Interacción
Clave:	
Distribución horaria:	Horas Clase: 01 Horas Taller: 00 Horas Laboratorio: 04 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 01 Créditos:06
Etapas de formación:	Terminal
Carácter:	Optativa
Requisito:	Ninguno
Modalidad de impartición:	Presencial
Equipo de diseño:	María Victoria Meza Kubo Alberto Leopoldo Morán y Solares
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Introducir al estudiantado en los fundamentos, métodos y prácticas del diseño de interacción para concebir, diseñar, prototipar y evaluar sistemas interactivos centrados en los usuarios. En esta asignatura se permitirá comprender cómo las características cognitivas, sociales, emocionales y físicas de los seres humanos influyen en la experiencia de uso, y cómo estas consideraciones pueden integrarse en el proceso de diseño de aplicaciones modernas, incluyendo aquellas basadas en tecnologías emergentes. Se encuentra en la etapa disciplinaria, es de carácter optativo y pertenece al área de conocimiento de Interacción Hombre Máquina. Se recomienda haber cursado la asignatura de Interacción Humano Computadora.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Desarrollar aplicaciones interactivas centradas en el usuario, a través de teorías, métodos y técnicas del diseño de interacción para generar experiencias útiles, eficientes, accesibles y emocionalmente significativas para diversas poblaciones de usuarios, con una actitud creativa y responsable.

IV. EVIDENCIA DE APRENDIZAJE

Desarrolla una aplicación interactiva funcional acompañada de manuales de usuario y del sistema, así como el análisis de los resultados de una evaluación de experiencia de uso realizada a la aplicación con usuarios reales.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Inclusión y Pensamiento Crítico.

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos del diseño de interacción

Competencia

Analizar los fundamentos del diseño de interacción, mediante el entendimiento de los elementos que influyen en la experiencia de usuario, como base para el diseño de nuevas aplicaciones, con una actitud crítica.

Contenido

- 1.1. Evolución del diseño de interacción
- 1.2. Principios fundamentales (visibilidad, retroalimentación, restricciones, affordances)
- 1.3. Modelos conceptuales y mapeos
- 1.4. Procesos de diseño centrado en el usuario (DCU, Design Thinking)
- 1.5. Introducción a accesibilidad y diseño universal
- 1.6. Consideraciones éticas: sesgos, privacidad, diseño responsable

Duración

03 horas

Unidad II. Modalidades de interacción y diseño emocional**Competencia**

Analizar los mecanismos de interacción para ajustarse a las capacidades y necesidades del usuario, a través del entendimiento del ser humano como un ser social y emocional, para seleccionar las formas de interacción más adecuadas a la población, con actitud crítica e iniciativa creativa.

Contenido

- 2.1. Interacción humano-humano mediada por tecnología
- 2.2. Diseño emocional y modelos de afecto en UX
- 2.3. Microinteracciones y retroalimentación afectiva
- 2.4. Interfaces multimodales
 - 2.4.1. Voz (VUI)
 - 2.4.2. Gestos
 - 2.4.3. Interfaces tangibles
- 2.5. Interfaces naturales y realidades extendidas (AR / VR)

Duración

03 horas

Unidad III. Obtención y análisis de requerimientos**Competencia**

Definir los requerimientos para el diseño de un sistema interactivo utilizando técnicas de obtención, análisis e interpretación de los datos, para identificar las necesidades reales de los usuarios objetivo, con creatividad y disposición.

Contenido

- 3.1. Estrategias de investigación con usuarios
- 3.2. Técnicas de recolección de datos:
 - 3.2.1. Entrevistas semiestructuradas
 - 3.2.2. Shadowing y observación contextual
 - 3.2.3. Cuestionarios
 - 3.2.4. Heurísticas
- 3.3. Análisis de datos:
 - 3.3.1. Afinidades
 - 3.3.2. Patrones y hallazgos clave
 - 3.3.3. Personas y arquetipos
 - 3.3.4. Journey mapping
- 3.4. Presentación de hallazgos y definición de requerimientos

Duración

03 horas

Unidad IV. Diseño y prototipado

Competencia

Realizar diseños de prototipos interactivos, por medio de técnicas de diseño para la construcción de aplicaciones de sistemas interactivos apegadas a las necesidades de los usuarios, con creatividad e iniciativa.

Contenido

- 4.1. Diseño conceptual: tareas, flujos, casos de uso
- 4.2. Diseño físico: pantallas, layouts, patrones de interacción
- 4.3. Prototipos
 - 4.3.1. Papel
 - 4.3.2. Digital
 - 4.3.3. Prototipos interactivos en herramientas (Figma, Balsamiq, etc.)
- 4.4. Guiones y escenarios
- 4.5. Diseño de microinteracciones y feedback

Duración

03 horas

Unidad V. Evaluación de sistemas interactivos

Competencia

Evaluar un prototipo de sistema interactivo por medio de técnicas de evaluación de experiencia de usuario que permitan identificar las principales fortalezas y debilidades del prototipo, de manera crítica y ordenada.

Contenido

- 5.1. ¿Por qué y qué evaluar?
- 5.2. Métodos de evaluación
 - 5.2.1. Pruebas con usuarios
 - 5.2.2. Análisis heurístico (Nielsen refinado)
 - 5.2.3. Pruebas A/B (conceptual)
 - 5.2.4. Métricas UX (eficiencia, error, satisfacción)
- 5.3. Plan de evaluación: tareas, participantes, métricas
- 5.4. Análisis de resultados y comunicación
- 5.5. Ética en la evaluación con usuarios

Duración

04 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Fundamentos del diseño de interacción

Práctica 1: Reconociendo las capacidades humanas en la interacción

Duración

04 horas

Procedimiento

1. Atiende a la explicación de la persona docente sobre percepción, memoria de corto plazo, atención y carga cognitiva.
2. Se organiza en equipos de trabajo.
3. Realiza ejercicios guiados de memoria, reconocimiento visual y atención selectiva.
4. Registra resultados individuales y grupales.
5. Analiza cómo las limitaciones humanas influyen en el diseño de menús, iconos y mensajes.
6. Identifica ejemplos de interfaces que respetan o violan estas capacidades.
7. Redacta conclusiones con recomendaciones de diseño.
8. Presenta resultados y recibe retroalimentación.

Recursos de apoyo

- Listas de palabras/imágenes.
- Cronómetro.
- Hojas de registro.

Práctica 2: Análisis de principios de diseño en interfaces reales

Duración

04 horas

Procedimiento

1. Atiende a la explicación de la persona docente sobre principios de diseño de interacción.
2. Selección de interfaces reales (web, móvil, sistemas físicos).
3. Análisis estructurado de visibilidad, consistencia, retroalimentación y restricciones.
4. Documentación de buenas y malas prácticas.
5. Propuesta de mejoras basadas en principios teóricos.
6. Discusión grupal.
7. Recibe retroalimentación docente y grupal.

Recursos de apoyo

- Dispositivos.
- Plantillas de análisis.

Práctica 3: Modelos conceptuales y metáforas de interacción

Duración

04 horas

Procedimiento

1. Atiende a la explicación de la persona docente sobre modelos conceptuales y metáforas.
2. Selección de un sistema interactivo.
3. Identificación de acciones, objetos y resultados esperados.
4. Construcción de modelos conceptuales mediante diagramas.
5. Evaluación del modelo desde la perspectiva del usuario.
6. Discusión grupal.
7. Recibe retroalimentación docente y grupal.

Recursos de apoyo

- Dispositivos.
- Plantillas de análisis.

Unidad II. Modalidades de interacción

Práctica 4: Interacción social cara a cara vs. mediada por tecnología

Duración

04 horas

Procedimiento

1. Atiende a la explicación introductoria de la persona docente sobre interacción social cara a cara vs. remota.
2. Se reúne en equipo de trabajo.

3. Realiza dos mini-actividades:
 - a) Conversación breve cara a cara.
 - b) Conversación equivalente mediante videollamada.
4. Documenta diferencias y similitudes en expresiones, turnos conversacionales, fluidez y claridad.
5. Asocia las diferencias encontradas con implicaciones para el diseño de sistemas colaborativos o de mensajería.
6. Redacta conclusiones.
7. Presenta los hallazgos al grupo.
8. Recibe retroalimentación docente y grupal

Recursos de apoyo

- Laptops o teléfonos con videollamada.
- Guía de observación.

Práctica 5: Diseño emocional en interfaces digitales

Duración

04 horas

Procedimiento

1. Atiende a la explicación de la persona docente sobre diseño emocional y retroalimentación afectiva.
2. Se reúne en equipos pequeños.
3. Selecciona dos aplicaciones y observa cómo expresan emociones (colores, microinteracciones, sonidos, animaciones).
4. Registra situaciones donde la interfaz genera satisfacción, frustración o confusión.
5. Clasifica los elementos como expresivos, neutrales o frustrantes.
6. Propón mejoras de diseño basadas en emociones objetivo.
7. Presenta las conclusiones al resto del grupo.
8. Recibe retroalimentación docente y grupal.

Recursos de apoyo

- Aplicaciones digitales.
- Hojas de registro.

Práctica 6: Interfaces naturales y multimodales

Duración

04 horas

Procedimiento

1. Atiende a la explicación de la persona docente sobre diseño emocional y retroalimentación afectiva.
2. Se reúne en equipos pequeños.

3. Selecciona dos aplicaciones y observa cómo expresan emociones (colores, microinteracciones, sonidos, animaciones).
4. Registra situaciones donde la interfaz genera satisfacción, frustración o confusión.
5. Clasifica los elementos como expresivos, neutrales o frustrantes.
6. Propón mejoras de diseño basadas en emociones objetivo.
7. Presenta las conclusiones al resto del grupo.
8. Recibe retroalimentación docente y grupal.

Recursos de apoyo

- Aplicaciones digitales.
- Hojas de registro.

Unidad III. Definición de requerimientos de sistemas interactivos

Práctica 7: Planeación y ejecución de entrevistas con usuarios

Duración

04 horas

Procedimiento

1. Atiende a la explicación de la docente sobre técnicas de obtención de datos.
2. Se organiza en equipos y selecciona un tema.
3. Diseña una mini-entrevista (5 a 7 preguntas).
4. Entrevista a 2 o 3 compañeros o usuarios disponibles.
5. Registra observaciones sobre comportamientos y dificultades.
6. Identifica necesidades, problemas y oportunidades de diseño.
7. Comparte sus hallazgos con el grupo.
8. Recibe retroalimentación docente.

Recursos de apoyo

- Hojas para notas.
- Grabadora del teléfono.
- Guía de entrevista.

Práctica 8: Observación contextual y análisis de tareas

Duración

04 horas

Procedimiento

1. Atiende a la explicación del docente.
2. Observación de usuarios realizando tareas.
3. Registro de dificultades y comportamientos.
4. Identificación de oportunidades de mejora.
5. Presentación de hallazgos

6. Presenta análisis al grupo.
7. Recibe retroalimentación docente.

Recursos de apoyo

- Hojas de observación.
- Cámara de video.

Práctica 9: Análisis de datos y definición de requerimientos

Duración

04 horas

Procedimiento

1. Atiende a la explicación de la persona docente.
2. Agrupación de datos mediante mapas de afinidad.
3. Construcción de personas y escenarios.
4. Redacción de requerimientos funcionales y de experiencia.
5. Presenta su propuesta al grupo.
6. Recibe retroalimentación.

Recursos de apoyo

- Notas adhesivas.
- Rotafolio.
- Marcadores.
- Hojas de análisis.

Unidad IV. Diseño de sistemas interactivos

Práctica 10: Prototipado de baja fidelidad

Duración

04 horas

Procedimiento

1. Atiende a la explicación de la persona docente.
2. Diseña los flujos de interacción del proyecto.
3. Construye prototipos en papel.
4. Pruebas rápidas entre equipos.
5. Ajustes al diseño
6. Comparte sus hallazgos con el grupo.
7. Recibe retroalimentación docente.

Recursos de apoyo

- Hojas.

- Marcadores.

Práctica 11: Prototipado digital navegable

Duración

04 horas

Procedimiento

1. Atiende a la explicación de la persona docente.
2. Digitalización del prototipo.
3. Creación de navegación entre pantallas.
4. Revisión de consistencia y claridad.
5. Presenta prototipo al curso.
6. Recibe retroalimentación.

Recursos de apoyo

- Herramientas de prototipado.

Práctica 12: Diseño de microinteracciones

Duración

04 horas

Procedimiento

1. Atiende a la explicación de la persona docente.
2. Identificación de microinteracciones clave.
3. Diseño de retroalimentación visual y textual.
4. Integración al prototipo.
5. Evaluación interna
6. Presenta sus resultados al grupo.
7. Recibe retroalimentación.

Recursos de apoyo

- Software para desarrollo.

Unidad V. Evaluación de sistemas interactivos

Práctica 13: Planeación de la evaluación con usuarios

Duración

04 horas

Procedimiento

1. Atiende a la explicación de la persona docente.
2. Define objetivos y métricas.
3. Diseña tareas y guiones.
4. Prepara instrumentos
5. Presenta frente al grupo.
6. Recibe retroalimentación docente.

Recursos de apoyo

- Plantillas de evaluación.

Práctica 14: Evaluación heurística y pruebas piloto

Duración

04 horas

Procedimiento

1. Atiende a la explicación de la persona docente.
2. Aplicación de heurísticas de usabilidad.
3. Identificación de problemas.
4. Prueba piloto con usuarios.
5. Ajustes al instrumento.
6. Recibe retroalimentación.

Recursos de apoyo

- Listado de heurísticas.

Práctica 15: Diseño de microinteracciones

Duración

04 horas

Procedimiento

1. Atiende a la explicación de la persona docente.
2. Ejecución de pruebas de usabilidad.
3. Registro de tiempos, errores y comentarios.
4. Organización de resultados
5. Presenta sus resultados al grupo.
6. Recibe retroalimentación.

Recursos de apoyo

- Prototipo.

- Hojas de registro.

Práctica 16: Análisis de resultados y rediseño

Duración

04 horas

Procedimiento

1. Atiende a la explicación de la persona docente.
2. Análisis de datos obtenidos.
3. Priorización de problemas.
4. Propuesta de rediseño.
5. Presentación final al grupo.
6. Recibe retroalimentación.

Recursos de apoyo

- Prototipo actualizado.
- Reportes.

IX. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

- Estudio de caso
- Método de proyectos
- Aprendizaje basado en problemas
- Aprendizaje basado en casos
- Técnica expositiva
- Debates

Estrategia de aprendizaje

- Investigación documental
- Investigación empírica
- Exposición de resoluciones de casos prácticos
- Trabajo colaborativo
- Técnica expositiva
- Visitas a campo
- Mapas mentales

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	20%
– Prácticas laboratorio	20%
– Tareas	10%
– Desarrollo de la aplicación	50%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- Cooper, A., Reimann, R., Cronin, D., & Noessel, C. (2014). *About face: the essentials of interaction design*. John Wiley & Sons. [clásica]
- Benyon, D. (2019). *Designing user experience: A guide to HCI, UX and interaction design* (3rd ed.). Pearson. [classic]
- Norman, D. A. (2013). *The design of everyday things* (Revised and expanded ed.). Basic Books. [clásica]
- Preece, J., Rogers, Y., & Sharp, H. (2019). *Interaction design: Beyond human-computer interaction* (5th ed.). Wiley. [clásica]
- Tolino, U., & Mariani, I. (2024). *Design behind interaction: Interfaces, technology, meanings*. Springer. <https://doi.org/10.1007/978-3-031-67416-7>

Recursos de acceso libre o complementarios externos

- Balsamiq Studios. (2024). *Balsamiq wireframes*. <https://balsamiq.com>
- Figma, Inc. (2024). *Figma: Collaborative interface design tool*. <https://www.figma.com>
- Maze. (2024). *Rapid testing platform for UX prototypes*. <https://maze.com>
- Nielsen Norman Group. (2024). *Articles and reports on usability and UX*. <https://www.nngroup.com>

XII. PERFIL DOCENTE

Profesionista en Computación o áreas afines con al menos 2 años de experiencia en la docencia, tener conocimientos en interacción humano-computadora y preferentemente con posgrado en el área. Ser una persona creativa, tener un eficiente manejo grupal, fomentar el pensamiento crítico y creativo en sus estudiantes.

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

Unidad académica:	Facultad de Ciencias, Ensenada
Programa educativo:	Licenciatura en Ciencias Computacionales
Plan de Estudios:	2027-1
Nombre de la unidad de aprendizaje:	Calidad y Pruebas de Software
Clave:	
Distribución horaria:	Horas Clase: 02 Horas Taller: 00 Horas Laboratorio: 02 Horas Práctica de Campo: 00 Horas Clínicas: 00 Horas Extra-clase: 02 Créditos:06
Etapas de formación:	Terminal
Carácter:	Optativa
Requisito:	Ninguna
Modalidad de impartición:	Presencial
Equipo de diseño:	Omar Alvarez Xochihua María Victoria Meza Kubo
Subdirección que aprueba el PUA:	Priscilla Elizabeth Iglesias Vázquez
Fecha:	21 de mayo de 2026
Aprobación y registro:	

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La unidad de aprendizaje de Calidad del Software es de carácter optativa dentro de la etapa terminal. Su área de conocimiento es Programación e Ingeniería de Software, donde fortalece el aprendizaje de modelos nacionales e internacionales para la evaluación y mejora continua del proceso de desarrollo de sistemas. Esta asignatura es teórico práctica, es importante porque busca reafirmar los conocimientos teóricos mediante la solución y discusión de casos de estudio reales. Se recomienda haber cursado y aprobado las unidades de Metodología de la Programación e Ingeniería de Software.

La finalidad de esta unidad de aprendizaje es capacitar al estudiante en el uso de los fundamentos y estándares calidad de desarrollo de software, que le permitan mejorar y evaluar los procesos de desarrollo, mantenimiento y operación de sistemas.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Evaluar la calidad de productos y procesos de software mediante la aplicación de estándares internacionales, métricas cuantitativas y estrategias de verificación y validación, para asegurar que las soluciones tecnológicas satisfagan las necesidades del cliente y las normativas vigentes, con una actitud ética y profesional.

IV. EVIDENCIA DE APRENDIZAJE

Elabora un reporte de auditoría diagnóstica y plan de mejora de procesos sobre un caso de estudio que contenga: identificación de brechas entre la situación actual y los estándares CMMI o Moprosoft, propuestas de solución integrando el ciclo de vida de ISO/IEC 12207 y la gestión de servicios de ITIL, junto con un análisis crítico final que contraste la viabilidad técnica, operativa y económica de los modelos seleccionados para optimizar la capacidad de desarrollo de la organización.

V. COMPETENCIA PROFESIONAL DEL PERFIL DE EGRESO A LA QUE CONTRIBUYE LA UNIDAD DE APRENDIZAJE

Diseñar, desarrollar y gestionar soluciones de software sustentables, seguras y centradas en el usuario, aplicando tecnologías emergentes, principios de ingeniería de software y metodologías alineadas con estándares de calidad nacionales e internacionales para brindar soluciones innovadoras e integrales a problemas específicos y demandas de la sociedad en diversos contextos, con visión empresarial, sentido ético y responsabilidad social.

VI. EJE TRANSVERSAL AL QUE CONTRIBUYE

Pensamiento Crítico

VII. DESARROLLO POR UNIDADES

Unidad I. Fundamentos y Cultura de la Calidad

Competencia

Analizar la evolución y los principios de la calidad de software, valorando el impacto ético y económico de la "no calidad", para fundamentar su importancia en la competitividad de las organizaciones tecnológicas, con una actitud analítica y crítica.

Contenido

- 1.1. Evolución histórica y conceptos de calidad de software.
- 1.2. El costo de la "No Calidad" e impacto en el negocio.
- 1.3. Cultura de calidad en equipos ágiles y tradicionales.
- 1.4. Ética y actitud crítica del ingeniero de calidad

Duración

04 horas

Unidad II. Modelos de calidad de software

Competencia

Evaluar los marcos de referencia nacionales e internacionales (como ISO/IEC 12207, CMMI y Moprosoft), mediante el análisis comparativo de sus procesos, niveles de madurez y requisitos normativos, para proponer estrategias de mejora en la capacidad de desarrollo de una organización, con una actitud analítica y crítica.

Contenido

- 2.1. Gestión por procesos: Familia ISO 9000 y 90003.
- 2.2. Mejora de procesos: CMMI y NMX-I-059/02 (Moprosoft).
- 2.3. Estándares de ciclo de vida: ISO/IEC/IEEE 12207.
- 2.4. Calidad de servicios: Framework ITIL e ISO 20000

Duración

06 horas

Unidad III. Calidad del Producto y Medición

Competencia

Evaluar la calidad de un producto de software utilizando modelos de calidad (ISO 25000) y métricas técnicas, para facilitar la toma de decisiones basada en datos y la mejora del código, con creatividad y disposición.

Contenido

- 3.1. Modelos de calidad de producto: McCall, Boehm e ISO 25000 (SQuaRE).
- 3.2. Atributos de calidad internos (mantenibilidad, código) y externos (usabilidad, confiabilidad).
- 3.3. Establecimiento de métricas y medición de la calidad.
- 3.4. Análisis de causa raíz y prevención de defectos

Duración

06 horas

Unidad IV. El Proceso de Aseguramiento de la Calidad (SQA)

Competencia

Diseñar Planes de Aseguramiento de la Calidad (SQAP), integrando técnicas de revisión, auditoría y control de configuración, para asegurar la trazabilidad y el cumplimiento de los requisitos durante el ciclo de vida, con una actitud crítica y honesta.

Contenido

- 4.1. El rol del Ingeniero de Calidad vs. Auditor.
- 4.2. Planeación del Plan de Aseguramiento de la Calidad (SQA Plan).
- 4.3. Técnicas de revisión: Revisiones técnicas formales, auditorías e inspecciones de código.
- 4.4. Gestión de riesgos y configuración (control de versiones y trazabilidad).

Duración

06 horas

Unidad V. Estrategias de Verificación y Validación (Pruebas)

Competencia

Ejecutar ciclos de pruebas de software utilizando diversas técnicas y niveles de evaluación (unitarias, integración, sistema) para identificar defectos de manera temprana y garantizar la funcionalidad y seguridad del producto, con una actitud honesta y responsable.

Contenido

- 5.1. Niveles de prueba: Unitarias, integración, sistema y aceptación.
- 5.2. Tipos de prueba: Funcionales, estrés (rendimiento) y seguridad.
- 5.3. Introducción a la Automatización de Pruebas (Test Automation Engineer).
- 5.4. Gestión de defectos: Reporte, seguimiento y cierre.

Duración

06 horas

Unidad VI. Calidad en la Gestión y Mejora Continua

Competencia

Implementar mecanismos de retroalimentación y mejora continua en los procesos de gestión, evaluando el desempeño final del proyecto para proponer acciones correctivas y preventivas en futuros desarrollos, de manera lógica, eficiente y ordenada.

Contenido

- 6.1. Gestión de proyectos bajo modelos de calidad (tiempo, forma y presupuesto).
- 6.2. Monitoreo, control y gestión con base cuantitativa.
- 6.3. Evaluación final y liberación del producto al cliente.
- 6.4. Retroalimentación e implementación de oportunidades de mejora.

Duración

04 horas

VIII. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

Unidad I. Fundamentos y Cultura de la Calidad

Práctica 1: Diagnóstico de Cultura y Madurez Organizacional

Duración

04 horas

Procedimiento

- 1. Atiende las orientaciones docentes para el análisis de casos de estudio organizacionales.

2. Diseña un instrumento de recolección de datos enfocado en la filosofía y ética de la calidad.
3. Aplica el instrumento para identificar el nivel de madurez inicial y las barreras culturales de la organización.
4. Identifica las brechas detectadas respecto a una mentalidad de aseguramiento de calidad (QA Mindset).
5. Elabora y entrega un reporte diagnóstico sobre la situación ética y profesional de la empresa analizada.

Recursos de apoyo:

1. Formatos de encuesta
2. Casos de estudio
3. Guías de ética profesional.

Unidad II. Modelos y Estándares de Procesos

Práctica 2: Auditoría Diagnóstica de Modelos (ISO/CMMI/Moprosoft)

Duración:

04 horas

Procedimiento:

1. Atiende las orientaciones docentes para la ejecución de una auditoría de procesos.
2. Selecciona un proceso de desarrollo específico para ser evaluado contra un estándar nacional o internacional.
3. Determina el tipo de auditoría a realizar (diagnóstica o de cumplimiento) según el modelo CMMI o Moprosoft.
4. Ejecuta la auditoría mediante el contraste de la práctica actual versus los requisitos normativos.
5. Identifica hallazgos, conformidades y no conformidades encontradas durante el ejercicio.
6. Elabora y entrega un reporte de auditoría que incluya el análisis de brechas (Gap Analysis).
7. Recursos: Estándares ISO/IEC 12207, Guías CMMI-DEV, NMX-I-059/02.

Recursos de apoyo:

4. Estándares ISO/IEC 12207
5. Guías CMMI-DEV, NMX-I-059/02

Unidad III. Calidad del Producto y Medición

Práctica 3: Evaluación de Atributos de Calidad del Producto

Duración:

04 horas

Procedimiento:

1. Atiende las orientaciones docentes para el uso de modelos de calidad de producto (ISO 25010).
2. Selecciona un producto de software funcional para evaluar sus atributos internos y externos.
3. Aplica métricas técnicas para cuantificar la mantenibilidad, seguridad y usabilidad del sistema.
4. Utiliza software especializado para realizar el análisis estático de código y detectar malas prácticas.
5. Compara los resultados obtenidos con los umbrales de calidad establecidos en el estándar SQuaRE.
6. Elabora y entrega un reporte de calidad del producto con recomendaciones técnicas de mejora.

Recursos de apoyo:

6. Norma ISO/IEC 25000
7. Herramientas de análisis de código (SonarQube o similares).

Unidad IV. Gestión del Aseguramiento de la Calidad (SQA)

Práctica 4: Planificación de SQA y Gestión de la Configuración

Duración:

04 horas

Procedimiento:

- I) Atiende las orientaciones docentes para la planeación del aseguramiento bajo el estándar IEEE 730.
- II) Define las actividades de verificación y validación para el ciclo de vida de un proyecto de software.
- III) Establece los mecanismos de control de versiones y gestión de cambios para asegurar la integridad de los artefactos.
- IV) Documenta los roles y responsabilidades dentro del Plan de Aseguramiento de la Calidad (SQAP).
- V) Elabora y entrega el documento de planificación y control de configuración del proyecto.

Recursos de apoyo:

- Estándar IEEE 730
- Git/GitHub
- Plantillas de gestión

Práctica 5: Revisión Técnica Formal (Inspección de Software)**Duración:**

04 horas

Procedimiento:

- Atiende las orientaciones docentes para la integración de equipos de inspección y asignación de roles.
- Selecciona un artefacto técnico (documento de requisitos o código) para ser revisado formalmente.
- Ejecuta la reunión de inspección siguiendo el protocolo de detección de defectos de Fagan.
- Registra cada defecto detectado especificando su ubicación, tipo y severidad.
- Realiza el análisis de los hallazgos y determina el estado de aprobación del artefacto inspeccionado.
- Elabora y entrega el reporte de la revisión técnica formal y el acta de acuerdos.

Recursos de apoyo:

- Listas de verificación (checklists)
- Formatos de reporte de defectos

Unidad V. Verificación y Validación: Estrategias de Prueba

Práctica 6: Diseño y Ejecución de Pruebas Dinámicas

Duración:

04 horas

Procedimiento:

- Atiende las orientaciones docentes para el diseño de casos de prueba de caja negra y caja blanca.
- Identifica las funcionalidades críticas y rutas lógicas del software para ser probadas.
- Ejecuta las pruebas dinámicas registrando los resultados esperados contra los obtenidos.
- Identifica discrepancias y fallos en la lógica o interfaz del sistema.
- Compara el comportamiento del software bajo condiciones normales y de error.
- Elabora y entrega un informe de ejecución de pruebas y cobertura de código.

Recursos de apoyo:

- Frameworks de pruebas (JUnit, PyTest)
- Entornos de desarrollo (IDE)

Práctica 7: Automatización de Pruebas y Gestión de Defectos

Duración:

04 horas

Procedimiento:

1. Atiende las orientaciones docentes para la automatización de suites de pruebas.
2. Codifica scripts de prueba automáticos para procesos de regresión y funciones críticas.
3. Ejecuta la suite automatizada y documenta los fallos encontrados en una herramienta de seguimiento.

4. Gestiona el ciclo de vida del defecto (reporte, asignación y validación del cierre).
5. Analiza la eficiencia de la automatización comparada con la ejecución manual.
6. Elabora y entrega un reporte de métricas de defectos y estado de la suite automatizada.

Recursos de apoyo:

- Selenium
- Gestores de defectos (Jira/Bugzilla)

Unidad VI. Mejora Continua y Cierre

Práctica 8: Auditoría Final y Evaluación de Mejora Continua

Duración:

04 horas

Procedimiento:

1. Atiende las orientaciones docentes para el cierre de los procesos de calidad del proyecto.
2. Evalúa el grado de cumplimiento final respecto a los objetivos establecidos en el Plan de Calidad inicial.
3. Efectúa una auditoría de cierre sobre los artefactos y procesos ejecutados durante el curso.
4. Identifica lecciones aprendidas y oportunidades de mejora para futuros ciclos de desarrollo.
5. Documenta las acciones correctivas y preventivas sugeridas tras el análisis final.
6. Elabora y entrega el reporte final de liberación de calidad y cierre de auditoría.

Recursos de apoyo:

- Guía ISO 19011
- Portafolio de evidencias del estudiante

XI. MÉTODO DE TRABAJO

Encuadre

El primer día de clase cada docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-estudiante.

Estrategia de enseñanza

Seleccione las estrategias correspondientes al proceso de enseñanza-aprendizaje, puede agregar o quitar

- Técnica expositiva para la introducción a los conceptos formales.
- Aprendizaje basado en problemas.
- Ejercicios prácticos para el diseño y aseguramiento de la calidad.
- Aprendizaje basado en proyectos.

Estrategia de aprendizaje

- Investigación.
- Trabajo en equipo.
- Realización de prácticas de laboratorio para la planeación del aseguramiento de la calidad.
- Elaboración de reportes de resultados y seguimiento de problemas.
- Elaboración de plan de aseguramiento de la calidad.

X. CRITERIOS DE EVALUACIÓN

Criterios de acreditación

Para tener derecho a examen ordinario y extraordinario cada estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente. La calificación es en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

– Evaluaciones parciales	40%
– Prácticas de laboratorio	40%
– Plan de aseguramiento de la calidad ..	20%
Total	100%

XI. REFERENCIAS

Recursos de la biblioteca física y digital de la UABC

- S. Goericke. (2020). The Future of Software Quality Assurance. Cham, Switzerland; Springer Open.
- S. Singh. (2020). Modeling of Test Suite Optimization Framework: An Enhancement of Regression Testing Techniques. Beau. Bassin, Mauritius; LAP LAMBERT Academic Publishing.
- R. S. Mann. (2019). Software Testing and Quality Assurance. Beau Bassin, Mauritius; LAP LAMBERT Academic Publishing.
- R. Cummings-John, y O. Peer. (2019). Leading Quality: How Great Leaders Deliver High-Quality Software and Accelerate. Growth. ROI Press.
- C. Laporte, y A. April. (2018). Software Quality Assurance. Hoboken, NJ, USA; Wiley IEEE Press.
- R. Chopra (2018). Software Quality Assurance: A Self-Teaching Introduction. Boston, Massasuchetts; Mercury Learning and Information LLC.

Recursos de acceso libre o complementarios externos

- J.J. Shen. (2019). Software Testing: Techniques, Principles, and Practices. Wuxi City, China; Jianjun Shen.
- A. S. Mahfuz. (2016). Software Quality Assurance: Integrating Testing, Security, and Audit (Internal Audit and IT Audit). Boca Raton, Florida, USA; CRC Press, Taylor & Francis Group.
- L. Lancu. (2019). QA Quality Assurance & Software Testing Fundamentals. Liliana Lancu.
- A. Pollner, M. Fewster, I. Schieferdecker. (2020). Test Automation Engineer: Guide to the ISTQB Advanced Level Certification. Rocky Nook.
- O. Ilhan. (2021). Software Testing Masterclass (2021) - From Novice to Expert. Disponible en <https://www.udemy.com/course/software-testing-masterclass-from-novice-to-expert/>; accesado en abril 06, 2026.
- W. Virgen. (2021). ISTQB Certified Tester - Foundation Level 2021 - Accredited. Disponible en <https://www.udemy.com/course/accredited-istqb-certified-testerfoundation-level-course/>; accesado en Febrero 2021.

XII. PERFIL DOCENTE

El docente de esta asignatura deberá contar con un título de licenciatura en Ciencias Computacionales o área afín; con experiencia en planeación, diseño e implementación de proyectos reales de desarrollo de software, haber participado en equipos de desarrollo basados en estándares de calidad y tener experiencia docente.



UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
Coordinación General de Formación Profesional

11.3. Anexo 3.

Evaluación externa e interna del programa educativo de Licenciado en Ciencias Computacionales

Dra. Edna Luna Serrano
Dra. Violeta García Arizmendi
Responsables Técnicos

Ensenada, Baja California, abril de 2024.

Contenido

Sección I. Evaluación externa del programa educativo de Licenciado en Ciencias	
Computacionales	10
Introducción.....	10
Objetivo general	10
Método general	11
Resultados.....	15
Estudio 1.1 Evaluación de la pertinencia social	15
1.1.1 Análisis de necesidades sociales.....	15
1.1.2 Análisis del mercado laboral	24
1.1.3 Estudio de egresados	31
1.1.4 Análisis de oferta y demanda.....	36
Matrícula de los programas educativos iguales o afines	43
Demanda vocacional a nivel estatal	60
Conclusiones Estudio 1.1 Evaluación de la pertinencia social.....	64
Estudio 1.2 Evaluación de referentes	68
1.2.1 Análisis de la profesión y su prospectiva	68
1.2.2 Análisis comparativo de programas educativos.....	80
1.2.3 Análisis de organismos nacionales e internacionales	105
Conclusiones Estudio 1.2 Evaluación de referentes	108
Conclusiones generales de la evaluación externa de el programa de LCC	114
Pertinencia social.....	114
Referentes estratégicos	115
Sección II. Evaluación interna del programa educativo de Licenciado en Ciencias	
Computacionales	117
Introducción.....	117
Estudio 2.1. Evaluación de fundamentos y condiciones de operación del programa educativo.....	118
2.1.1 Propósitos del programa, misión y visión	119

2.1.2. Condiciones generales de operación del programa educativo.....	126
Conclusiones Estudio 2.1. Evaluación de fundamentos y condiciones de operación del programa educativo	131
Estudio 2.2. Evaluación del currículo.....	133
2.2.1 Modelo educativo y plan de estudios	134
2.2.2 Actividades para la formación integral.....	144
Conclusiones Estudio 2.2. Evaluación del currículo	148
Estudio 2.3 Evaluación de la trayectoria escolar de los estudiantes en el programa educativo.....	150
2.3.1 Proceso de ingreso al programa educativo	151
2.3.2 Indicadores de trayectoria escolar	152
2.3.3 Participación de los estudiantes en programas de apoyo.....	155
2.3.4 Resultados de los estudiantes	158
Conclusiones Estudio 2.3 Evaluación de la trayectoria escolar de los estudiantes por el programa educativo	159
Estudio 2.4 Evaluación del personal académico, infraestructura y servicios	162
2.4.1 Personal académico	164
2.4.2 Infraestructura académica.....	168
2.4.3 Infraestructura física	170
2.4.4 Servicios de apoyo.....	171
Conclusiones Estudio 2.4 Evaluación del personal académico, infraestructura y servicios	173
Conclusiones generales de la evaluación interna del programa de LCC.....	176
Fundamentos y condiciones de operación del programa educativo	176
Currículo.....	176
Trayectoria escolar de los estudiantes en el programa educativo.....	177
Personal académico, infraestructura y servicios.....	178
Referencias	179
Apéndices	186

Apéndice A. Cuestionario dirigido a académicos para retroalimentar la evaluación externa e interna del programa educativo de Licenciado en Ciencias Computacionales	186
Apéndice B. Cuestionario dirigido a egresados para retroalimentar la evaluación externa e interna del programa educativo de Licenciado en Ciencias Computacionales.....	195
Apéndice C. Cuestionario dirigido a empleadores para retroalimentar la evaluación externa e interna del programa educativo de Licenciado en Ciencias Computacionales	204
Apéndice D. Cuestionario dirigido a estudiantes para retroalimentar la evaluación externa e interna del programa educativo de Licenciado en Ciencias Computacionales.....	209

Índice de Tablas, Sección I

Tabla 1.1 Objetivos generales y específicos de los estudios en la Evaluación externa	11
Tabla 1.2 Participantes de la Evaluación interna del programa educativo del programa de LCC	12
Tabla 1.3 Características generales de la muestra de egresados	12
Tabla 1.4 Características generales de la muestra de empleadores.....	13
Tabla 1.5 Objetivos del Análisis de necesidades sociales.....	15
Tabla 1.6 Opinión de egresados sobre fundamentos sobresalientes en la carrera	22
Tabla 1.7 Opinión de egresados sobre aspectos sobresalientes en la carrera	22
Tabla 1.8 Objetivos del Análisis del mercado laboral	25
Tabla 1.9 Opinión de empleadores sobre la formación profesional de los egresados del programa de LCC	28
Tabla 1.10 Opinión de empleadores sobre la formación profesional de los egresados del programa de LCC	29
Tabla 1.11 Objetivos del Estudio de egresados.....	32
Tabla 1.12 Sector económico del empleo de los egresados	33
Tabla 1.13 Nombre de las empresas y descripción de labores del escenario de trabajo de los egresados	33
Tabla 1.14 Ingresos mensuales de los egresados	34
Tabla 1.15 Opinión de egresados sobre la concordancia de sus estudios en el programa de LCC y su actividad laboral	35
Tabla 1.16 Objetivos del Análisis de oferta y demanda.....	36
Tabla 1.17 Oferta de Programas iguales al programa de LCC en México	37
Tabla 1.18 Oferta de Programas afines al programa de LCC en México	40
Tabla 1.19 Porcentaje de crecimiento de la matrícula de LCC (2018-2019 a 2022-2023)..	43
Tabla 1.20 Oferta de Programas iguales o afines al programa de LCC en España	45
Tabla 1.21 Oferta de Programas iguales o afines al programa de LCC en Argentina	51
Tabla 1.22 Oferta de Programas iguales o afines al programa de LCC en Chile	58
Tabla 1.23 Oferta de Programas iguales o afines al programa de LCC en Colombia	59
Tabla 1.24 Objetivos del Análisis de la profesión y su prospectiva	68

Tabla 1.25 Objetivos del Análisis comparativo de programas educativos	80
Tabla 1.26 Programas Educativos Nacionales iguales a LCC	83
Tabla 1.27 Programas Educativos Internacionales iguales a LCC	96
Tabla 1.28 Objetivos del Análisis de organismos nacionales e internacionales	105
Tabla 1.29 Estructura del EGEL Plus COMPU	106

Índice de Tablas, Sección II

Tabla 2.1 Elementos estructurales de la Evaluación de fundamentos y condiciones de operación del programa educativo.....	118
Tabla 2.2 Elementos del perfil de ingreso al programa de LCC.....	122
Tabla 2.3 Competencias de las Etapas Básica, Disciplinaria y Terminal del programa de LCC	124
Tabla 2.4 Modalidades de aprendizaje y objetivos	125
Tabla 2.5 Indicadores y estadísticas de estudiantes del programa de LCC	128
Tabla 2.6 Estructura organizacional de la Facultad de Ciencias.....	130
Tabla 2.7 Descripción de elementos estructurales de la Evaluación del currículo	133
Tabla 2.8 Etapas de formación del programa de LCC.....	135
Tabla 2.9 Distribución de asignaturas por áreas del conocimiento del programa de LCC	137
Tabla 2.10 Distribución de los créditos por etapa de formación y tipo de asignatura del programa de LCC	138
Tabla 2.11 Asignaturas del área de conocimiento Entorno Social.....	139
Tabla 2.12 Asignaturas del área de conocimiento Matemáticas	140
Tabla 2.13 Asignaturas del área de conocimiento Programación e Ingeniería de Software	140
Tabla 2.14 Asignaturas del área de conocimiento Arquitectura de Computadoras	141
Tabla 2.15 Asignaturas del área de conocimiento Redes.....	142
Tabla 2.16 Asignaturas del área de conocimiento Software de Base.....	143
Tabla 2.17 Asignaturas del área de conocimiento Interacción Humano-Computadora.....	143
Tabla 2.18 Asignaturas del área de conocimiento Tratamiento de la Información.....	144

Tabla 2.19 Descripción de elementos estructurales de la Evaluación de la trayectoria escolar de los estudiantes por el programa educativo.....	150
Tabla 2.20 Rezago por reprobación del programa de LCC	152
Tabla 2.21 Deserción del programa de LCC.....	153
Tabla 2.22 Eficiencia terminal de titulación y eficiencia de titulación por egreso en el programa de LCC	155
Tabla 2.23 Tipos de becas otorgadas a estudiantes del programa de LCC	156
Tabla 2.24 Atención del área de Orientación Educativa y Psicopedagógica.....	157
Tabla 2.25 Descripción de elementos estructurales de la Evaluación del currículo	163
Tabla 2.26 Características generales del cuerpo docente.....	165
Tabla 2.27 Listado de colaboradores del Programa de Licenciado en Ciencias Computacionales	165

Índice de Figuras, Sección I

Figura 1.1 Estructura de los estudios de la Evaluación externa del programa educativo de LCC	10

Figura 1.2 Matrícula de LCC de los últimos 5 años.....	44

Índice de Figuras, Sección II

Figura 2.1 Estructura de los estudios de la Evaluación interna del programa educativo de LCC	117

Figura 2.2 Identificación, solicitud y funcionamiento de las asesorías académicas	155

Presentación

Las universidades tienen el objetivo de formar personas con las habilidades, conocimientos y actitudes necesarios para promover el desarrollo social y económico a largo plazo en cada país (Ferreyra, et al., 2017), con responsabilidad social (Vallaey, 2014). Este objetivo es conseguido a través de tres funciones principales de las universidades como son la investigación, la enseñanza y la difusión de la cultura. También, tienen la responsabilidad de formar profesionistas y especialistas requeridos en todos los sectores de la producción, que tengan la oportunidad de insertarse en el mercado laboral (Hernández-Laos et al., 2012).

En este sentido, la Universidad Autónoma de Baja California es una institución comprometida con la calidad educativa en apego a lineamientos rigurosos, sistemáticos y continuos que favorezcan los procesos educativos. Una de las tareas que se realizan es la evaluación de programas educativos con el fin de identificar las fortalezas y áreas de mejora. Este documento es una evaluación realizada al programa educativo de la Licenciado en Ciencias Computacionales, con el propósito de brindar información que sustente la modificación o actualización del programa educativo.

La evaluación está dividida en dos secciones: Evaluación externa y Evaluación interna. Cada una está conformada por objetivos, método, resultados organizados con base en la estructura de las secciones —estudios, análisis o categorías— y conclusiones. A continuación, se detalla cada sección.

La Sección I contiene la Evaluación externa, misma que se integra por dos estudios como son la evaluación de la pertinencia social y la evaluación de referentes. Por su parte, la Sección II abarca la Evaluación interna, la cual se constituye de cuatro estudios, el primero de estos es la evaluación de fundamentos y condiciones de operación del programa educativo; el segundo, corresponde a la evaluación del currículo; el tercero es el referente a la evaluación de la trayectoria escolar de los estudiantes del programa educativo; por último, la evaluación del personal académico, infraestructura y servicios.

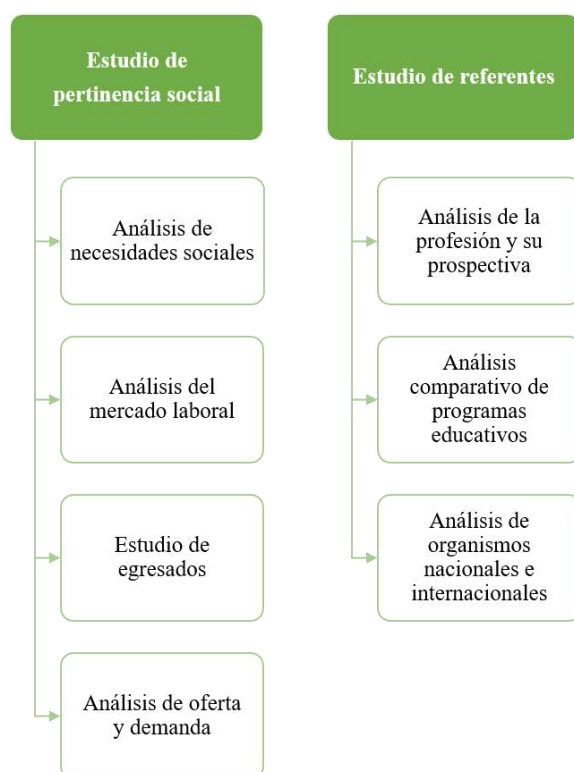
Sección I. Evaluación externa del programa educativo de Licenciado en Ciencias Computacionales

Introducción

La Evaluación externa del programa educativo de Licenciado en Ciencias Computacionales (LCC) se conforma de dos estudios: 1.1 Estudio de pertinencia social y 1.2 Estudio de referentes. Cada estudio compromete análisis particulares, tal como se muestra en la Figura 1.1.

Figura 1.1

Estructura de los estudios de la Evaluación externa del programa educativo de LCC



Nota. De “Metodología de los estudios de fundamentación para la creación, modificación y actualización de programas educativos de licenciatura”, por A. Serna, y A. Castro, 2018, Universidad Autónoma de Baja California. CC BY-NC-ND.

Objetivo general

El objetivo general de la Evaluación externa es valorar la pertinencia social del programa educativo y analizar los referentes nacionales e internacionales que fundamenten la

creación, modificación o actualización. La Tabla 1.1 muestra los objetivos generales de los dos estudios pertenecientes a esta evaluación

Tabla 1.1

Objetivos generales y específicos de los estudios en la Evaluación externa

Objetivo general	Análisis específicos	
1.1 Estudio de pertinencia social		
Evaluar la pertinencia social que fundamente la creación, modificación o actualización de programas educativos de licenciatura.	1.1.1	Análisis de necesidades sociales
	1.1.2	Análisis del mercado laboral
	1.1.3	Estudio de egresados
	1.1.4	Análisis de oferta y demanda
1.2 Estudio de referentes		
Analizar referentes estratégicos donde se inscribe la profesión del programa educativo, los programas iguales o afines y los referentes nacionales e internacionales que permiten fundamentar la creación, modificación o actualización del programa.	1.2.1	Análisis de la profesión y su prospectiva.
	1.2.2	Análisis comparativo.
	1.2.3	Análisis de organismos nacionales e internacionales.

Nota. De “Metodología de los estudios de fundamentación para la creación, modificación y actualización de programas educativos de licenciatura”, por A. Serna, y A. Castro, 2018, Universidad Autónoma de Baja California. CC BY-NC-ND.

Método general

Este apartado siguió la propuesta de Serna y Castro (2018) con el fin de sustentar la actualización o modificación del programa educativo del programa de LCC. De esta manera, se integró una investigación documental y una investigación empírica para dar respuesta a los objetivos específicos de los dos estudios de la evaluación externa.

Participantes

Los participantes involucrados fueron una muestra no probabilística de 15 docentes de la licenciatura en Ciencias Computacionales, 15 egresados de este programa educativo y una muestra no probabilística de 10 empleadores (ver Tabla 1.2).

Tabla 1.2*Participantes de la Evaluación interna del programa educativo del programa de LCC*

Participantes	Población total	Muestras	
		Frecuencia	%
Docentes	28	17	60.0
Egresados	71	16	23.0
Empleadores	21	10	48.0

De la muestra de egresados 12 (75%) pertenecen al sexo masculino y 4 (25%) al sexo femenino. El rango de edad con mayor frecuencia fue de 41 a 45 años (25%) y la nacionalidad mexicana es reportada por el 100% de dicha muestra. En cuanto al lugar de residencia, 13 egresados (81.3%) indicaron vivir en Baja California, dos residen en otro estado de la República Mexicana y un egresado radica fuera del país en Seattle, Washington. Por otra parte, el estado civil de 9 egresados (56.3%) es soltero (a) (ver Tabla 1.3).

Tabla 1.3*Características generales de la muestra de egresados*

Ítems	Respuestas	Frecuencias	%
Sexo	Masculino	12	75
	Femenino	4	25
Edad	25 años o menos	2	12.5
	26-30 años	3	18.8
	31-35 años	1	6.2
	36-40 años	3	18.8
	41-45 años	4	25.0
	46-50 años	2	12.5
	51-55 años	1	6.2
	56 años o más	0	--
Nacionalidad	Mexicana	16	100
	Extranjero	0	--
Lugar de residencia actual	Baja California	13	81.3
	Otro estado de la república mexicana	2	12.5
	Fuera de México	1	6.2
Estado civil	Casado(a)	6	37.5
	Soltero(a)	9	56.3
	Divorciado(a)	0	--
	Unión libre	1	6.2
Hijos	Sí	5	31.2
	No	11	68.8

Respecto a los empleadores de la muestra, nueve de ellos (80.0%) eran de sexo femenino y 1 (10.0%) de sexo masculino; respecto a la edad, tres (27.2%) empleadores se ubicaron entre las edades de 26 a 30 años, igualmente entre las edades de 41 a 45 años (27.7%). El nivel máximo de estudios de seis empleadores (60.0%) correspondió a Licenciatura o Ingeniería; y, la empresa u organización en la que laboraban pertenecía en mayor parte al sector privado 7 (70.0%) (ver Tabla 1.4).

Tabla 1.4

Características generales de la muestra de empleadores

Ítem	Respuestas	Frecuencias	%
Sexo	Femenino	6	81.8
	Masculino	4	18.1
Edad	25 años o menos	0	--
	26-30 años	4	40.0
	31-35 años	1	10.0
	36-40 años	1	10.0
	41-45 años	2	20.0
	46-50 años	1	10.0
	51-55 años	1	10.0
	56 años o más	--	--
	Sin respuesta	--	--
Máximo nivel de estudios	Licenciatura o Ingeniería	6	60.0
	Maestría	4	40.0
	Doctorado	--	--
	Posdoctorado	--	--
Sector al que pertenece la empresa u organización	Público	2	20.0
	Privado sin ánimo de lucro (Asociación)	1	10.0
	Privado	7	70.0
	Otro	--	--

Materiales

- 1) Anuarios Estadísticos de Educación Superior: Ciclo escolar 2021-2022 (Asociación Nacional de Universidades e Instituciones de Educación Superior [ANUIES], 2023).
- 2) Plan Estatal de Desarrollo de Baja California 2022-2027 (Gobierno del Estado de Baja California, 2022).
- 3) Indicadores de características educativas de la población México en cifras (Instituto Nacional de Estadística, Geografía e Informática [INEGI], 2021).
- 4) Bases de datos: EBSCOhost, Web of Science. y Scopus.

- 5) Páginas web: Observatorio laboral del Gobierno de México, Hays, Forbes México, Banco Mundial, Consejo Nacional de Acreditación de Informática y Computación (CONAIC), Centro Nacional de Evaluación para la Educación Superior (Ceneval).
- 6) Metodología para realizar estudios que fundamentan la creación, modificación o actualización de programas educativos (Serna y Castro, 2018).
- 7) Cuestionario dirigido a egresados adaptado del Documento ejecutivo para la elaboración de la Evaluación Externa e Interna para la modificación o actualización de Planes de Estudio de Licenciatura de la UABC (Propuesta de instrumentos para estudios) (UABC, 2020a) (Apéndice B).
- 8) Cuestionario dirigido a empleadores adaptado del Documento ejecutivo para la elaboración de la Evaluación Externa e Interna para la modificación o actualización de Planes de Estudio de Licenciatura de la UABC (Propuesta de instrumentos para estudios) (UABC, 2020a) (Apéndice C).

Procedimiento

El procedimiento consistió en dos tipos de investigación: documental y empírica. La investigación documental comprometió el análisis de diversos documentos y páginas institucionales en línea especializadas en la profesión de ciencias computacionales, algunos de estos fueron: los Anuarios estadísticos de la ANUIES (2023), el Plan Estatal de Desarrollo de Baja California 2022-2027 (Gobierno del Estado de Baja California, 2022), el INEGI (2021), bases de datos, el Observatorio laboral del Gobierno de México, Forbes México y el Banco Mundial.

La investigación empírica fue un diseño transversal de tipo descriptivo que involucró la participación de coordinadores de carrera, egresados y empleadores del programa de LCC.

Coordinadores: se tuvo una entrevista por videoconferencia con una docente del programa de LCC, la finalidad fue atender temas referentes a principios teóricos, programas afines y organismos reguladores para la formación en ciencias computacionales. Posteriormente, la

vía de comunicación fue el correo electrónico institucional y reuniones presenciales, medio por el cual se compartieron materiales, dudas y comentarios.

Egresados y empleadores: la aplicación de los cuestionarios se administró y supervisó por medio del *software LimeSurvey* versión 1.92+ *build* 120919. De esta forma fueron enviados los cuestionarios y posteriormente cinco recordatorios. Asimismo, se visitó en su oficina a los empleadores que no abrieron el cuestionario en línea, para verificar que la empresa existiera o asegurar sus respuestas y comentarios.

El procesamiento de la información conllevó análisis descriptivos de los datos resultantes de los ítems cerrados y análisis de contenido de la información derivada de los ítems abiertos.

Resultados

Los resultados se exponen de acuerdo con la estructura de los dos estudios de la Evaluación Externa (ver Figura 1). Es importante señalar que, en el análisis de las respuestas se contó con la participación de toda la muestra.

Estudio 1.1 Evaluación de la pertinencia social

El objetivo general fue evaluar la pertinencia social que fundamente la creación, modificación o actualización de programas educativos. Para tal fin, se abordaron cuatro análisis: 1.1.1 Análisis de necesidades sociales, 1.1.2 Análisis del mercado laboral, 1.1.3 Estudio de egresados y 1.1.4 Análisis de oferta y demanda.

1.1.1 Análisis de necesidades sociales

Objetivo

El objetivo general y los objetivos específicos se exponen en la Tabla 1.5.

Tabla 1.5

Objetivos del Análisis de necesidades sociales

Objetivo general	Objetivo específico
Determinar las necesidades y problemáticas sociales estatales, regionales, nacionales e internacionales	Identificar el contexto geográfico, demográfico, social, cultural, económico y político tanto regional como nacional e internacional en el que se inscribe el programa educativo.
	Analizar las necesidades y problemáticas sociales estatales,

Objetivo general	Objetivo específico
(actuales y futuras) que atenderá o atiende el programa educativo y los egresados.	regionales, nacionales e internacionales que atenderá o atiende el programa educativo. Realizar un análisis prospectivo de las necesidades y problemáticas sociales que atenderá o atiende el programa educativo.

Nota. De “Metodología de los estudios de fundamentación para la creación, modificación y actualización de programas educativos de licenciatura”, por A. Serna, y A. Castro, 2018, *Universidad Autónoma de Baja California*. CC BY-NC-ND.

Metodología

Se realizó una investigación documental y un diseño no experimental. Para alcanzar los objetivos planteados se revisaron fuentes secundarias de organismos nacionales e internacionales como la OCDE, Naciones Unidas, Foro Económico Mundial, Encuesta Delloite, Oxford Internet Institute (OII), Dell Technologies, INEGI, CONEVAL, Encuesta de jóvenes en México, DOF, Plan Nacional de Desarrollo 2019-2014, y el Plan Estatal de Desarrollo de Baja California 2022-2027.

Resultados

La población mundial sumó 8,000 millones de personas en el año 2022 y se estima que para el 2050 asciendan a 8,500 ciudadanos. De estos, 662 millones habitan en América Latina y el Caribe, y la esperanza de vida se ubica en los 72.2 años (Naciones Unidas, 2024). En el tema económico, durante la pandemia por Covid-19, el mundo enfrentó la peor crisis económica, con afectaciones en la economía formal, el turismo, el servicio de aviación, y la pérdida general de 400 millones de empleos; esta situación ocasionó que más de 71 millones de personas cayeran en la categoría de pobreza extrema. En el ámbito educativo, los cierres de las escuelas dejaron sin escolarización al 90% de los estudiantes, revirtiendo años de progreso (Naciones Unidas, 2020).

De acuerdo con la OCDE (2022), el aprendizaje de adultos, también conocido como aprendizaje permanente, contribuye con las personas a progresar en sus carreras y adaptarse a un mundo incierto y cambiante. En promedio, en los países de la OCDE, el 14% de los adultos ha participado en educación y capacitación formales o no formales. El aprendizaje y la capacitación en línea están en aumento, sin embargo, serán diferentes para quienes tienen empleo o no. Los que tienen empleo están poniendo mayor énfasis en los cursos de

desarrollo personal. Aquellos que están desempleados han puesto mayor interés en mejorar sus habilidades digitales, por lo que toman cursos de análisis de datos, informática y manejo de la tecnología de la información y la comunicación.

En el caso de los jóvenes, su situación es la siguiente. En el mundo habitan 1,200 millones entre 15 y 24 años de edad. En ellos recae parte importante del desarrollo económico, político, social y cultural de sus países y del mundo. Por ello, es importante conocer las condiciones en que viven y las oportunidades que requieren. La “Encuesta Deloitte”, que analiza periódicamente las inquietudes de los jóvenes de 46 países, reveló que sus mayores preocupaciones son las finanzas diarias y el no poder jubilarse cómodamente. Cuando se trata de elegir un nuevo empleo, buscan el equilibrio entre trabajo y vida personal, así como oportunidades de aprendizaje y desarrollo. En el cuidado del medio ambiente, la gran mayoría de los jóvenes se esfuerzan por reducir su impacto personal. Desde su punto de vista, las empresas y los gobiernos no están tan comprometidos, y presionan a sus propios empleadores para que tomen medidas en beneficio del medio ambiente. En el tema de la salud mental, los niveles de estrés y agotamiento son altos, más aún en las mujeres, y se complica la retención para los empleadores (Deloitte, 2022).

Contexto nacional

En la República Mexicana habitan 126,014,024 de personas, el 51.2% son mujeres y el 48.8% son hombres. La edad mediana es de 29 años (INEGI, 2020a). La esperanza de vida hacia el 2050 será de 76.7 años en los hombres y de 82.6 años para el caso de las mujeres. Se prevé un incremento de la población mayor de 65 años y representarán el 16.8% para el año 2050. Esto significará un doble reto para el gobierno y la sociedad mexicana. Por un lado, la cobertura de pensiones y, por el otro, la generación de condiciones para que se mantengan activos, saludables e independientes (CONAPO, 2019).

De acuerdo con el INEGI (2020a) en el área educativa, los grupos que más asistieron a la escuela fueron de 6 a 11 años (95.5%), y el grupo de 12 a 14 años (90.5%). En contraparte, los grupos de edad que menos asistieron al nivel educativo correspondió a los niños y niñas de 3 a 5 años (63.3%), así como el grupo de 15 a 24 años (45.3%) en edad

de cursar la educación media superior y superior. El rezago educativo en México, para el año 2020, fue del 19.2% el equivalente a 24.4 millones de personas (CONEVAL, 2022).

En el rubro económico para el año 2022, la población económicamente activa en México fue de 59.6 millones de personas. El ingreso corriente promedio al mes de los hogares mexicanos presentó ciertas diferencias, por ejemplo, en el decil más alto el ingreso fue de 54,427 pesos/mes, en el medio 11,122 pesos/mes, y en el bajo 3,312 pesos/mes. Por su parte, el mayor gasto corriente monetario promedio mensual es en alimentos, bebidas y tabaco (3,793 pesos); seguido de transporte y comunicaciones (1,850 pesos); vivienda y combustible (1,905 pesos); y educación y esparcimiento y cuidados personales (770 pesos). En lo que menos se gasta es en vestido y calzado (297 pesos) [INEGI, 2020].

El pronóstico para la población en edad de trabajar (30 a 64 años) pasarán del 43.8% en 2030, al 45.8% en el año 2050. Los esfuerzos de política pública deberán encaminarse hacia la generación de condiciones que les permita gozar de un retiro digno de la actividad laboral y servicios de salud (CONAPO, 2019).

El porcentaje de la población mexicana en situación de pobreza para el año 2020 fue de 43.9%. Los estados de Chiapas (75.5%), Guerrero (66.4%), Puebla (62.4%) y Oaxaca (61.7%) presentan un porcentaje alto de su población en esta condición. En contraparte, se encuentran los estados de Baja California (22.5%), Nuevo León (24.3%), Chihuahua (25.3%), y Coahuila (25.6%). En el caso de la pobreza extrema, el porcentaje de personas que viven en esta condición en México es del 8.5%. De acuerdo con el CONEVAL (2020), la emergencia sanitaria derivada de la pandemia por Covid-19 aumentó los desafíos en salud, educación y alimentación de los mexicanos. Por lo tanto, entre los principales retos de política pública estarán la recuperación del ingreso de las personas y la atención a grupos y regiones más afectados.

En cuanto a los jóvenes mexicanos, acontece lo siguiente. En México viven 30.7 millones y representan el 24.6% de la población (INEGI, 2020a). En la Encuesta de jóvenes en México 2019, realizada por la Fundación SM, se mencionó que el trabajo es la principal actividad de los jóvenes de entre 23 y 29 años (60%). Entre aquellos que trabajan, el 24.5% son empleados, el 7.7% tiene trabajo informal, y el 6.5% tiene negocio propio. Además,

una quinta parte de los jóvenes de hasta 22 años (21%) realizan doble actividad, es decir, estudian y trabajan. Entre los principales motivos para trabajar se encuentran: ayudar a mantener a la familia (36%), tener más dinero para gastos personales (17.1%), contar con suficientes recursos para estudiar (14.5%) o independizarse (13.1%). El 33.8% tuvo un ingreso mensual de 3,100 pesos o menos, y el 32.7% entre 3,101 y 5000 pesos (Fundación SM, 2019).

Lo que más importa a los jóvenes es su familia (58.3%), la salud (55.7%) y ganar dinero (51.8%). En contraparte, lo que menos les importa es la política (23.9%) y la religión (22.5%). En su tiempo libre, el recurso que más utilizan los jóvenes de entre 15 y 29 años es el internet (72%). Los dispositivos que más utilizan son el teléfono celular (76.7%) y la computadora (16.8%). Las principales actividades que realizan cuando navegan en internet son usar redes sociales como Facebook, Twitter o Instagram (58.9%), y comunicarse vía WhatsApp y Messenger (51.9%). Las actividades que realizan con mayor frecuencia son escuchar música (66%), ver televisión (65%), estar con pareja o familia (61%), amigos (55%) y leer (55%). Aquéllas que menos realizan son viajar (28.5%), hacer trabajo voluntario (29%), asistir a museos, exposiciones o espectáculos en vivo (32%) (Fundación SM, 2019).

Problemáticas y necesidades sociales que la profesión ayudará a resolver.

La era de la tecnología digital avanza a pasos agigantados, hoy en día gozamos de la inteligencia artificial (IA) en nuestra vida cotidiana. Por ejemplo, existen aplicaciones que nos informan sobre el tráfico en la ciudad y nos recomienda la mejor ruta para llegar a nuestro destino. Tenemos predictores del clima, y también de nuestros gustos e intereses; contamos con asistentes de voz para planificar tareas, reproducir música, o buscar información. Las aerolíneas organizan mejores rutas gracias a la disponibilidad de datos de otros vuelos en tiempo real. Los agricultores pueden identificar problemas en sus cultivos, con ayuda de ciertas aplicaciones. En la medicina, la IA ayuda a analizar radiografías y realizar diagnósticos de manera más rápida y precisa (OII, 2023).

Sin embargo, esta no es una realidad para el 50% de la población a nivel mundial, debido a que no tienen acceso a la misma, lo que demanda el afrontamiento de ciertos retos

y desafíos que permitan a toda la humanidad gozar de sus beneficios. Aunado a ello, aún se requiere crear contenido digital y aplicaciones que sean acordes a las necesidades locales, junto con el incipiente desarrollo de seguridad digital (Naciones Unidas, 2019).

Tampoco es un hecho para todas las empresas. En opinión de los líderes empresariales, la mayoría de las compañías se encuentra luchando contra el cambio digital, y reconocen que aún muchas de estas no gozan de sus beneficios. Las empresas requieren visión y estrategia digital, fuerza laboral capacitada, tecnología, tiempo y presupuesto. Por ello, consideraban la necesidad de activar la transformación digital en sus empresas y esperaban obtener una defensa efectiva de ciberseguridad, crear una oferta de productos como servicio, evolucionar a empresa digital, avanzar con ayuda de la I+D, utilizar la IA para anticiparse a las demandas del cliente, y ofrecer experiencias hiperconectadas con realidad virtual (Dell Technologies, 2018). Sin embargo, la recesión por el Covid-19 transformó las tareas, las habilidades y los trabajos de manera emergente. El 84% de los empleadores se consideraron dispuestos a digitalizar rápidamente los procesos de trabajo, y el 44% de la fuerza laboral operó de forma remota (Foro Económico Mundial, 2020).

Las Naciones Unidas informaron que muchos países en desarrollo aún no cuentan con servicios básicos como carreteras, agua, saneamiento, energía eléctrica y tecnologías de la información y las comunicaciones. En cuanto al uso de internet, únicamente dos de cada 10 personas que viven en países en vías de desarrollo tuvieron acceso en 2019, comparado con 9 de cada 10 ciudadanos de los países desarrollados. Es necesario invertir en I+D, y financiar la infraestructura económica de los países en desarrollo para impactar en el nivel de vida y contribuir a la disminución de la pobreza mediante la generación de oportunidades de empleo, atención sanitaria, saneamiento y acceso a la educación (Naciones Unidas, 2015).

En este sentido, las Naciones Unidas promueven un crecimiento y desarrollo de los países más equitativo y sostenible, a través de 17 objetivos globales alcanzables hacia el año 2030. Los Objetivos de Desarrollo Sostenibles (ODS) que puede atender la licenciatura en Ciencias Computacionales son los siguientes: El objetivo 4 “Educación de calidad” y el objetivo 8 “Trabajo decente”, y el objetivo 9 “Industria, innovación e infraestructura”. El objetivo 4 se centra en garantizar una educación inclusiva, equitativa y de calidad y

promover oportunidades de aprendizaje durante toda la vida para todos. Entre las metas hacia el año 2030 se encuentran aumentar el número de jóvenes y adultos con competencias para acceder al empleo, trabajo decente y emprendimiento; eliminar las disparidades de género en la educación, así como asegurar igualdad de oportunidades para las personas más vulnerables; promover la educación en desarrollo sostenible, entre otros. Mientras, a través del objetivo 8 se espera lograr hacia el 2030 un empleo pleno, productivo y decente para todos; igualdad de remuneración por trabajo del mismo valor; así como erradicar el trabajo forzoso, esclavitud y la trata de personas (Naciones Unidas, 2015).

Por su parte, el Objetivo 9. Industria, innovación e infraestructura, consiste en construir infraestructuras resilientes, promover la industrialización sostenible y fomentar la innovación. Una de las metas hacia el año 2030 es aumentar la investigación científica y mejorar la capacidad tecnológica de los sectores industriales, mediante la innovación y el aumento de investigadores en esta área (Naciones Unidas, 2015).

Aunado a lo anterior, en el Plan Nacional de Desarrollo 2019-2024 se estableció, como parte de la Política Social, el objetivo de que todos los mexicanos puedan vivir hacia el año 2024 en un entorno de bienestar. Esto implica la necesidad de incluir a la población vulnerable en el desarrollo económico, político, educativo y social del país para avanzar hacia la construcción de la modernidad, sin excluir a nadie. Para ello se desarrolló, entre muchas otras, la Estrategia Nacional Digital 2019-2024 (END). Esta tiene la finalidad de lograr una transformación digital y tecnológica a nivel nacional mediante dos ejes principales de acción: (a) la Política Digital en la Administración Pública Federal, la cual pretende mejorar y transparentar los servicios del gobierno aprovechando y utilizando las TIC; y (b) la Política Social Digital, proyecta la cobertura de internet en todo México, incluyendo las comunidades más pobres, para integrarlos a la cultura digital que impera a nivel local y mundial (DOF, 2019). La END atiende el derecho de las personas para acceder a las TIC, la radiodifusión y las telecomunicaciones, estipulado en el artículo 6° de la Constitución Política de los Estados Unidos Mexicanos (DOF, 2023).

A su vez, el Plan Estatal de Desarrollo de Baja California 2022-2027 indica, en el apartado de “Ciencia, Tecnología e Innovación Educativa,” la necesidad de crear plataformas educativas digitales que brinden capacitación a docentes de educación básica

en el uso de herramientas digitales aplicadas a la enseñanza. Asimismo, señala la necesidad de destinar recursos para crear infraestructura y equipamiento especializado que permita desarrollar tecnologías de punta, y vincular la relación universidades-empresas para crear proyectos de innovación tecnológica y generar programas de incubadoras de empresas.

Opinión de los egresados: Respecto a los fundamentos considerados sobresalientes en la carrera; el 56.2% de los egresados valoró en total acuerdo y el 31.2% de acuerdo en que los aprendizajes obtenidos en la carrera son el fundamento para empezar a trabajar y mejorar sus perspectivas profesionales (ver Tabla 1.6).

Tabla 1.6

Opinión de egresados sobre fundamentos sobresalientes en la carrera

Ítem	Respuestas (%)				
	Totalmente de acuerdo	De acuerdo	Ni de acuerdo ni en desacuerdo	En desacuerdo	Totalmente en desacuerdo
<i>Considero que gracias a mi carrera tengo una buena base para:</i>					
Empezar a trabajar	56.2	31.2	--	6.2	6.2
Aprender en el trabajo	68.7	31.2	--	--	--
Realizar las tareas de tu trabajo actual	62.5	25.0	--	6.2	6.2
Mejorar tus perspectivas profesionales	56.2	31.2	6.2	--	6.2
Mejorar tu desarrollo personal	68.7	18.7	6.2	--	6.2

Asimismo, los egresados coincidieron en acuerdo con 50.0% que el enfoque del plan de estudios es general, el 62.5% señaló que el programa de LCC cuenta con reconocimiento académico, de igual forma, el 62.5% puntualizó que la secuencia de asignaturas es adecuada y el 50.0% consideró pertinente la proporción de teoría y práctica del programa educativo. Por otra parte, el 43.7% tuvo una postura imparcial —ni de acuerdo ni en desacuerdo— sobre la especialización del plan de estudios (ver Tabla 1.7).

Tabla 1.7

Opinión de egresados sobre aspectos sobresalientes en la carrera

Ítem	Respuestas (%)

	Totalmente de acuerdo	De acuerdo	Ni de acuerdo ni en desacuerdo	En desacuerdo	Totalmente en desacuerdo
<i>Considero que en mi carrera:</i>					
El enfoque del plan de estudios es general	25.0	50.0	18.7	--	6.2
El enfoque del plan de estudios es especializado	6.2	31.2	43.7	12.5	6.2
Cuenta con prestigio académico	31.2	62.5	6.2	--	--
La secuencia de las asignaturas del plan de estudios es adecuada	6.2	62.5	18.7	6.2	6.2
La proporción de teoría y práctica del programa educativo es adecuada	12.5	50.0	12.5	12.5	12.5

Conclusiones de las Necesidades sociales

-En el mundo habitan 8,000 millones de personas, en América Latina viven 662 millones y la esperanza de vida es de 72.2 años.

-El aprendizaje y la capacitación en línea de los adultos están en aumento, quienes trabajan prefieren cursos de desarrollo personal y los desempleados optan por cursos de análisis de datos, informática y manejo de las TIC.

-Los jóvenes prefieren un empleo que compagine con su vida personal, se preocupan por las finanzas diarias, la jubilación y el medio ambiente (Deloitte, 2022)

-En México habitan 126,014,024 personas, el 51.2% son mujeres y el 48.8% hombres; la edad media es de 29 años (INEGI, 2020).

-Hacia el 2050 se prevé un incremento del 16.8% en la población mayor de 65 años (CONAPO, 2019).

-La población económicamente activa fue de 59.6 millones de personas en el año 2022, el ingreso promedio mensual oscila entre \$54,427^{oo} y los \$3,312^{oo} pesos (INEGI, 2020).

- El mayor porcentaje de la población mexicana en condiciones de pobreza se presentó en Chiapas, Guerrero, Puebla y Oaxaca. Mientras el menor porcentaje de dio en Baja California, Nuevo León, Chihuahua y Coahuila (CONEVAL, 2020).

- En México habitan 30.7 millones de jóvenes y representó el 24.6% de la población (INEGI, 2020).
- El trabajo es la principal actividad de los jóvenes entre 23 y 29 años de edad.
- El recurso que más utilizan los jóvenes de 15 a 29 años es el teléfono celular (76.7%).
- El uso de la inteligencia artificial está en aumento en la vida cotidiana de las personas (OII, 2023).
- Sin embargo, el 50% de la población no tiene acceso a la misma, se requiere crear contenido digital y aplicaciones acordes a las necesidades locales y la seguridad digital es un tema pendiente (Naciones Unidas, 2019).
- Algunas empresas aún se resisten al cambio digital, aunque están convencidas de la necesidad de transformarse digitalmente y requieren del desarrollo de la ciberseguridad (Dell Technologies, 2018).
- Las Naciones Unidas promueven un crecimiento y desarrollo mundial más equitativo y sostenible, por lo que proponen 17 Objetivos de Desarrollo Sostenible hacia el 2030. Los egresados de la Licenciatura en Ciencias Computacionales podrían contribuir con los objetivos 4 “Educación de calidad”, objetivo 8 “Trabajo decente” y objetivo 9 “Industria, innovación e infraestructura”.
- A nivel nacional se impulsa la Estrategia Nacional Digital 2019-2024, con la finalidad de lograr una transformación digital y tecnológica en México.
- A nivel estatal, en el Plan Estatal de Desarrollo de Baja California 2022-2027, se indica la necesidad de crear plataformas educativas digitales para capacitar a los docentes de educación básica; crear infraestructura y equipamiento para desarrollar tecnología de punta; lograr una mayor vinculación universidad-empresas en proyectos de innovación tecnológica.

1.1.2 Análisis del mercado laboral

Objetivos

La Tabla 1.8 presenta los objetivos de este análisis.

Tabla 1.8

Objetivos del Análisis del mercado laboral

Objetivo general	Objetivo específico
Determinar las necesidades y problemáticas (actuales y futuras) del mercado laboral (estatal, regional, nacional y global) que atenderá o atiende el egresado del programa educativo.	Analizar el mercado laboral actual y futuro donde se insertará el egresado del programa educativo. Identificar los requerimientos del mercado laboral, en términos del perfil que debe tener el egresado.

Nota. De “Metodología de los estudios de fundamentación para la creación, modificación y actualización de programas educativos de licenciatura”, por A. Serna, y A. Castro, 2018, Universidad Autónoma de Baja California. CC BY-NC-ND.

Metodología

El tipo y diseño de investigación que se realizó en esta apartado es una investigación documental y una investigación empírica. La investigación documental permitirá analizar el mercado laboral actual y futuro en donde se insertará el egresado del programa de LCC. La investigación empírica permitirá identificar y analizar los requerimientos del mercado laboral y determinar el perfil que debe tener el egresado del programa de LCC (Serna y Castro, 2018).

Para realizar la investigación documental se revisó la información disponible en las páginas web de diversos organismos nacionales e internacionales, tales como el Foro Económico Mundial, Banco Interamericano de Desarrollo, LinkedIn, Dell Technologies, Hays, EBV Consulting & Learning, Observatorio laboral e INEGI. En lo referente a la investigación empírica se detallan las respuestas del Cuestionario para empleadores de la UABC.

Resultados

De acuerdo con el Foro Económico Mundial (2020), la automatización y la recesión por el Covid-19 transformó rápidamente las tareas, las habilidades y los trabajos. El 43% de las empresas encuestadas indicaron una reducción de su fuerza laboral debido a la integración de tecnología, mientras el 34 % tuvo un incremento de la misma. Por su parte, el 41% expandió el uso de contratistas para realizar tareas especializadas. Asimismo, los empleadores mencionaron un grupo de habilidades que se requieren mejorar hacia el 2025, entre estas se incluyen el pensamiento crítico y el análisis, así como la resolución de

problemas, la autogestión y el aprendizaje activo, la resiliencia, tolerancia al estrés y la flexibilidad (Foro Económico Mundial, 2020).

Entre las Megatendencias a desarrollarse en la próxima década se encuentra la transformación digital, la cual tiene un impacto a nivel personal y empresarial. Por ejemplo, cada vez más personas incluyen la tecnología en su vida cotidiana, para mejorar su autonomía y capacidad de gestión con ayuda de aplicaciones y la tecnología del internet. Las empresas incorporan estas tecnologías digitales en todas las áreas con la finalidad de mejorar sus modelos de negocios, procesos, operaciones y la experiencia de sus clientes con valor personalizado (EBV Consulting & Learning, 2023). Algunos líderes empresariales consideraron que algunas tareas de las compañías podrán delegarse a las tecnologías digitales como la gestión del inventario, la administración financiera, la logística, administración de reuniones o entrada de datos, el diseño de productos y el servicio al cliente (Dell Technologies, 2018).

En consecuencia, esta transformación digital generará otra megatendencia que son las nuevas competencias en el ámbito laboral, lo cual requiere de personal especializado para desarrollar tareas que ayuden a las empresas a superar las barreras que les impiden incursionar como un negocio digital exitoso, tales como la falta de visión y estrategia digital, falta de preparación de la fuerza laboral, limitaciones tecnológicas, de tiempo y presupuesto, así como la falta de leyes y reglamentos que regulen las actividades digitales (Dell Technologies, 2018).

En relación con lo anterior, en el “listado de empleos en auge 2023” de LinkedIn, menciona que serán 15 ocupaciones las que tendrán mayor demanda: Ingeniero(a) SER (Site Reliability Engineer), Representante de desarrollo de ventas, Responsable de crecimiento, Ingeniero de datos, Representante de desarrollo de negocios, Relación con clientes o especialista de éxito con el cliente, Desarrollador(a) de Java/Script, Investigador(a) de experiencia del usuario, Especialista en ventas, Asociado de operaciones, Ingeniero(a) de nube, Desarrollador(a) de Python, Ingeniero en ciberseguridad, Director(a) de alianzas, y Experto(a) en logística (LinkedIn, 2023).

En México, el Observatorio Laboral reportó, con cifras del primer trimestre del 2023, que los profesionistas ocupados con licenciatura en Ciencias de la computación

sumaron 747, 005 (el 71.8% son hombres y el 28.2% mujeres) y ganaron un promedio 14, 876 pesos mensuales (Observatorio Laboral, 2023).

Las tendencias en el mercado laboral en México, en el caso del sector privado, apuntan hacia el requerimiento de profesionistas especializados, la necesidad de innovación constante, la evolución del marketing digital, y el aumento en el uso del teléfono móvil. Las PyMEs se encuentran en crecimiento, por lo que será una opción laboral para varios profesionistas. En lo referente al sector público, la contratación temporal o por honorarios se volverá más constante con el objetivo de cubrir necesidades urgentes. El panorama del mercado laboral es prometedor para los profesionistas con conocimientos en inteligencia artificial, machine learning, Cloud computing y Blockchain. Los cargos altamente demandados son: data analysts, software developers, cyber security experts, infrastructure architects, CRM specialists, automation y cloud experts, IT project managers (Hays, 2023).

En este sentido, el Foro Económico Mundial (2020) especificó que, los trabajos de alta demanda en México apuntan hacia especialistas en inteligencia artificial, analistas de seguridad de la información, directores de proyectos y científicos de datos. De igual forma, las nuevas habilidades que los empleadores demandan serán la solución de problemas complejos, el aprendizaje activo, pensamiento analítico, diseño y programación de tecnología, y resolución de problemas.

En cuanto al tipo de empresa, el observatorio laboral informó que la tendencia apunta hacia una mayor contratación en las pequeñas y medianas empresas (PyMES), más que en las grandes o multinacionales. De esta manera, todos aquéllos que buscan empleo deberán considerar a este tipo de empresa. En lo que respecta al sector público, el tipo de contratación muestra una propensión hacia la contratación temporal y por honorarios, más que por tiempo indefinido o por plazas (Observatorio laboral, 2023).

Las empresas de Consultoría aportaron el 1.8% de la producción bruta del país, colocándola dentro de las 17 actividades más importantes de la economía en México, entre un total de 88 subsectores. Este tipo de empresas contratan un promedio de 15 personas, posicionándolas por su tamaño en la categoría de microempresas. En el año 2022 ofrecieron sus servicios un total de 69, 247 empresas de consultoría y mostraron, desde 2003, un

aumento en promedio por año del 2.4%. Aún durante la pandemia por Covid-19 progresaron con un promedio de crecimiento anual del 3.3% (INEGI, 2022).

Cabe resaltar que, entre las actividades de consultoría más importantes por su producción bruta se encuentran: Administración de negocios (31.4%), Consultoría en administración (15.3%), Diseño de sistemas (11.1%), Contabilidad y auditoría (8.5%), Servicios de ingeniería (6.3%), Bufetes jurídicos (5.7%), Servicios de minería (3.7%), Procesamiento electrónico (3.3%), Laboratorios de pruebas (1.8%), Servicios de arquitectura (1.8%), Resto de actividades (11.1%). Como puede observarse, dos de las actividades destacadas en este tipo de empresas tienen relación con el área de Ciencias Computacionales: (a) Diseño de sistemas, que abarca lo relacionado con servicios en el campo de las TIC por medio de actividades de planeación y diseño de sistemas de cómputo, asesorías de instalación de redes y equipos informáticos, administración de centros de cómputo o instalación de software; y (b) Procesamiento electrónico de información, hospedaje y otros servicios, mismos que abarcan diversos servicios como procesamiento de tarjetas de crédito para tiendas virtuales, reservaciones, tiempo compartido de instalaciones de *mainframe*, hospedaje de páginas web y aplicaciones, servicios de *streaming*, así como de microfilmación y escaneo óptico (INEGI, 2022).

El tema de las certificaciones también se destaca como una tendencia en el área de TIC, ya sea que las mismas empresas busquen capacitar a sus empleados, con la finalidad de colocarlos a la vanguardia del conocimiento que contribuya en la mejorar su desempeño o que los propios profesionistas busquen la mejora de sus credenciales (BID, 2022).

Opinión de los empleadores: el 40.0% estuvo de acuerdo en que la formación profesional del egresado del programa de LCC corresponde a las exigencias actuales del campo laboral. Sin embargo, la opinión se divide en el tema referente a la formación del egresado frente a las demandas del mercado laboral del futuro, el 30% está de acuerdo y otro 30% en desacuerdo (ver Tabla 1.9).

Tabla 1.9

Opinión de empleadores sobre la formación profesional de los egresados del programa de LCC

Ítems	Respuestas (%)				
	Totalmente de acuerdo	De acuerdo	Ni de acuerdo ni en desacuerdo	En desacuerdo	Totalmente en desacuerdo
La formación profesional de los egresados de la carrera en LCC es acorde a las demandas del mercado laboral actual:	10.0	40.0	30.0	20.0	--
La formación profesional de los egresados de la carrera en LCC es acorde a las demandas del mercado laboral futuro:	20.0	30.0	20.0	30.0	--

Asimismo, el 62.5% de los empleadores estimó una alta demanda actual del programa de LCC y el 75.0% consideró una alta demanda futura. Además, el 62.5% opinó que el empleo de hoy solicita bastantes conocimientos y habilidades de los egresados del programa de LCC (ver Tabla 1.10).

Tabla 1.10

Opinión de empleadores sobre la formación profesional de los egresados del programa de LCC

Ítems	Respuestas (%)		
	Alta	Media	Baja
En la actualidad, considera que la demanda de la carrera en LCC es:	62.5	37.5	--
En el futuro, considera que la demanda de la carrera en LCC será:	75.0	12.5	12.5

Ítems	Respuestas (%)			
	Nada	Poco	Bastante	Mucho
¿En qué medida, el trabajo actual requiere más conocimientos y habilidades de los egresados de la carrera en LCC?	--	12.5	62.5	25.0

Además, los empleadores comentaron lo siguiente:

Fortalezcan las bases, pero enseñen a aplicar las bases en un ambiente laboral. No dividan tanto el plan de estudios, enfóquense en un área, ya sea de investigación o de implementación, desarrollo y manejo de sistemas. Pero no ambos. (Em_1)

Proyectos donde apliquen los conocimientos adquiridos al desarrollo de capacidades de liderazgo, trabajo en equipo, cumplimiento de metas y que tengan un impacto social. (Em_3)

Es importante trabajar las habilidades blandas de los estudiantes, ética, lógica, valores, responsabilidad. (Em_4)

Se podrían mejorar las materias dedicadas al desarrollo de software. (Em_5)

Conclusiones del Mercado laboral

- Durante la pandemia por Covid-19 se transformaron las tareas, habilidades y los empleos. Los empresarios destacaron un grupo de habilidades que requieren mejorarse hacia el 2025: pensamiento crítico y análisis, resolución de problemas, autogestión y aprendizaje activo, resiliencia, tolerancia al estrés y flexibilidad (WEF, 2023).
- Entre las mega tendencias a desarrollarse hacia el año 2030 se enlistan la transformación digital que impactará a nivel personal y empresarial; las empresas requerirán personal especializado que ayuda a las empresas a transformarse en un negocio digital (Dell Technologies, 2018).
- En México, los profesionistas ocupados con licenciatura en ciencias computacionales en 2023 sumaron 747,005 (71.8% hombres y 28.2% mujeres) y ganaron un promedio mensual de \$14,876° pesos (INEGI, 2023).
- El mercado laboral es prometedor para profesionistas con conocimientos en inteligencia artificial, machine learning, Cloud computing y Blockchain (Hays, 2023).
- Los cargos altamente demandados serán: data analysts, software developers, cyber security experts, infraestructura architects, CRM specialists, automation and cloud experts, IT project managers (Hays, 2023).

- En México, los empleos altamente demandados serán para especialistas en inteligencia artificial, analista de seguridad de la información, directores de proyectos y científicos de datos.
- Las tendencias en contrataciones en el sector privado serán hacia las pequeñas y medianas empresas; mientras en el sector público serán por honorarios y temporales.
- Las empresas de consultoría se colocan dentro de las 17 actividades más importantes en México. Entre las actividades realizadas por este tipo de empresas, se destacan las relacionadas con el programa de LCC: diseño de sistemas, y procesamiento electrónico de información, hospedaje y otros servicios.
- Las certificaciones se acentúan como una tendencia de capacitación y actualización en el área de las TIC.
- Entre las respuestas al Cuestionario de empleadores de la UABC, el 40% estuvieron de acuerdo en que la formación profesional de los egresados del programa de LCC corresponde con las exigencias actuales del mercado laboral. Sin embargo, las opiniones se dividen al considerar esta formación acorde con las demandas del mercado laboral futuro.
- Los empleadores recomiendan aplicar las bases en el ambiente laboral, evitar dividir el plan de estudios, enfocarse en un área, aplicar los conocimientos en proyectos que desarrollan habilidades de liderazgo, trabajo en equipo, cumplimiento de metas, ética, lógica, valores, responsabilidad; así como mejorar materias dedicadas al desarrollo de software.

1.1.3 Estudio de egresados

Objetivos

El objetivo general del estudio de egresados fue analizar las experiencias de los egresados del programa de LCC según su desempeño en el mercado laboral para retroalimentar el programa educativo (ver Tabla 1.11).

Tabla 1.11*Objetivos del Estudio de egresados*

Objetivo general	Objetivo específico
Analizar las expresiones de los egresados de acuerdo con su desempeño en el mercado laboral con la finalidad de retroalimentar el programa educativo.	Identificar la situación sociodemográfica y laboral de los egresados del programa educativo.
	Explicar el nivel de satisfacción de los egresados con la formación recibida, para resolver necesidades y problemáticas del mercado laboral y de la sociedad.
	Estimar la relación de las diversas modalidades de aprendizaje con la formación integral del egresado.
	Identificar el interés del egresado por su desarrollo profesional.
	Establecer las nuevas competencias predominantes y emergentes requeridas por el mercado laboral.
	Analizar las recomendaciones de los egresados para mejorar el programa educativo.

Nota. De “Metodología de los estudios de fundamentación para la creación, modificación y actualización de programas educativos de licenciatura”, por A. Serna, y A. Castro, 2018, *Universidad Autónoma de Baja California*. CC BY-NC-ND.

Metodología

El tipo y diseño de investigación realizado en este estudio corresponde con una investigación empírica, por lo que se aplicó el Cuestionario para egresados de la UABC. A continuación, se desglosan los principales resultados.

Resultados

El licenciado en Ciencias Computacionales es el profesional dedicado a gestionar, ofrecer soluciones innovadoras en diversos entornos, derivadas del análisis, modelado, diseño, implementación y evaluación de proyectos complejos para la automatización de procesos que pueden ser tratados con sistemas de cómputo y asegurando la integridad de la información (UABC, 2017). Por lo tanto, el Plan de estudios 2017 de la licenciatura en Ciencias Computacionales estipula que un egresado de esta licenciatura desarrollará las siguientes competencias:

-Gestionar y desarrollar software aplicando metodologías acordes a los estándares nacionales e internacionales para satisfacer las demandas del mercado con sentido ético, visión empresarial y responsabilidad social.

- Administrar los recursos de cómputo y telecomunicaciones, utilizando los protocolos y estándares de comunicación para garantizar la integridad de la información en las organizaciones con sentido de responsabilidad social.

-Diseñar y desarrollar soluciones computacionales que modelen y simulen procesos de las Ciencias, mediante algoritmos y modelos matemáticos que permitan analizar el comportamiento de fenómenos complejos, con creatividad e innovación.

Opinión de los egresados: Respecto a la ocupación laboral, del total de la muestra (16) el 100% se encontraban laborando; de los cuales 15 (93.7%) ejercen su profesión, y solo en el caso de 1 (6.3%) el empleo no se relaciona con su carrera, trabaja en una empresa del sector salud. Los egresados que se emplearon al finalizar los estudios fueron 14 (87.5%).

Asimismo, 1 (6.3%) trabaja de manera independiente y 15 (93.8%) forman parte de una empresa. En relación con el tipo de sector económico al que pertenece la empresa en que laboran, 6 egresados (37.5%) se encuentran en servicios profesionales y técnicos (ver Tabla 1.12).

Tabla 1.12

Sector económico del empleo de los egresados

Ítems	Respuestas	Frecuencias	%
Sector económico al que pertenece la empresa o institución en que trabaja:	Servicios profesionales y técnicos	6	37.5
	Educación	3	18.7
	Gobierno	2	12.5
	Investigación	2	12.5
	Industria	2	12.5
	Otro	1	6.3

El 100% de los egresados indicó el nombre del centro donde laboran, de estos, el 68.8% señaló la actividad que realizaban. La Tabla 1.13 presenta las empresas y la descripción de las labores mencionadas por los egresados.

Tabla 1.13

Nombre de las empresas y descripción de labores del escenario de trabajo de los egresados

Nombre de la empresa	Descripción de labores
iTijuana	Desarrollador Android/iOS sr, scrum master

Nombre de la empresa	Descripción de labores
Cigom	Desarrollo y soporte en aplicaciones para distribución y representación de datos
Microsoft Corp	Trabajando como desarrollador de software full stack
Centro de Investigación Científica y de Educación Superior de Ensenada, Baja California	Técnico desarrollador y administrador de bases de datos, Departamento de Informática, División de Telemática
UABC	--
Outsourcing - Baja Technologies	--
Hospital General de Ensenada	Estadística y Desarrollo de aplicaciones
Gobierno del Estado de Baja California	--
Potentor	Desarrollo y mantenimiento de la plataforma
Thermofisher Scientific	--
Search rebel	Desarrollador web
Centro de Investigación Científica y de Educación Superior de Ensenada, Baja California	Gestor de propiedad intelectual
Ubilogix	Desarrolladora
Baja Agro International	--
UABC	Programador web como prestador de servicios
UABC	Administrador de Redes, servidores, soporte técnico y Docencia

El 62.5% de los egresados, el porcentaje más alto, recibe un ingreso mensual de más de \$20,000 pesos, en esta muestra de egresados ninguno tiene ingresos menores a \$8,000 pesos (ver Tabla 1.14). Así, el 93.8% de los egresados estimó que sus ingresos son suficientes para cubrir sus necesidades, ante el 6.3% que los consideró insuficientes.

Tabla 1.14

Ingresos mensuales de los egresados

Ítem	
Indique su ingreso mensual neto actual (incluyendo bonos y prestaciones):	
	%
Más de 20,000 pesos	62.5
Entre 17,000 y 20,000 pesos	25.0
Entre 13,000 y 16,000 pesos	6.3
Entre 9,000 y 12,000 pesos	6.3
Entre 5,000 y 8,000 pesos	--

Por otra parte, el 68.7% de los egresados indicó que sus actividades laborales se alinean mucho a su formación profesional en el programa de LCC (ver Tabla 1.15). Además, el 56.3% destacó posibilidades de ascenso en su trabajo, el 18.7% no tiene oportunidades de ascenso y el 25.0% no estaba seguro.

Tabla 1.6

Opinión de egresados sobre la concordancia de sus estudios en el programa de LCC y su actividad laboral

Ítems	Respuestas (%)			
	Mucho	Parcialmente	Poco	Muy poco
¿En qué medida coincide su actividad laboral con sus estudios en el programa de LCC?	68.7	18.7	6.3	6.3

El 87.5% mencionaron que después de egresar del programa de LCC demoraron menos de un año para emplearse, el 6.3% tardó entre uno y dos años y el 6.3% tardó cinco años o más en conseguir empleo. Es importante mencionar que, el 81.25% de los egresados consideraron que el emprendimiento no se fomentó durante sus estudios, mientras el 68.75% opinó que el perfil de emprendedor es relevante en esta carrera profesional.

Respecto al grado de estudios para el empleo, el 56.3% de los egresados consideró que los estudios de licenciatura son adecuados para su trabajo actual, el 25.0% indicó que la maestría es requerida en su empleo, el 12.3% precisó conveniente los estudios doctorales para el ejercicio de su empleo actual y el 6.3% opinó no ser necesarios los estudios universitarios para desempeñar su empleo.

Conclusiones del estudio de egresados

- Respecto a la situación laboral de la muestra de egresados, el 100% se encuentra trabajando; el 93.7% considera que ejerce su profesión, y el 87.5% se emplearon al finalizar los estudios de licenciatura.
- El 93.8% forma parte de una empresa y el 6.3% trabaja de manera independiente; el 37.5% se desempeña en el sector de servicios profesionales y técnicos.

- El 62.5% recibe un ingreso mensual mayor a \$20,000° pesos, ninguno percibe un ingreso mensual menor a \$8,000° pesos; el 93.8% de los egresados consideraron que sus ingresos son suficientes para cubrir sus necesidades.
- El 68.7% indicó que sus actividades laborales se alinean mucho con su formación profesional.
- El 87.5% demoró menos de un año para emplearse.
- Respecto a la formación en emprendimiento, el 82.2% consideró que no se fomentó durante sus estudios; el 68.7% opinó que el perfil de emprendedor es relevante en esta carrera.
- Referente al grado de estudios para el empleo, el 56.3% consideró que la licenciatura es adecuada, mientras el 25% indicó que su empleo requiere el grado de maestría.

1.1.4 Análisis de oferta y demanda

Objetivos

En la tabla 1.7 se especifican los objetivos plasmados en la Metodología de Serna y Castro (2018) para el análisis de la oferta y demanda del programa educativo en estudio.

Tabla 1.7

Objetivos del Análisis de oferta y demanda

Objetivo general	Objetivo específico
Analizar la oferta de programas educativos y la demanda vocacional para cursar el programa educativo.	Identificar la oferta de programas educativos iguales y/o afines al programa educativo que se pretende crear, modificar o actualizar con el propósito de determinar la posición competitiva que se pretende lograr con el programa educativo.
	Analizar la matrícula de programas educativos iguales y/o afines al programa educativo que se creará, modificará o actualizará con el fin de identificar el comportamiento de la oferta educativa.
	Determinar la demanda vocacional a nivel estatal para cursar el programa educativo.

Objetivo general	Objetivo específico
	Caracterizar a la población estudiantil potencial a ingresar al programa educativo.
	Establecer la relación de la demanda y la oferta en los programas educativos afines o iguales.

Nota. De “Metodología de los estudios de fundamentación para la creación, modificación y actualización de programas educativos de licenciatura”, por A. Serna, y A. Castro, 2018, Universidad Autónoma de Baja California. CC BY-NC-ND.

Metodología

El tipo y diseño de investigación en este apartado corresponde con una investigación documental que permitirá analizar la oferta educativa nacional e internacional, y una investigación empírica que ayudará a determinar la demanda vocacional en el Estado de Baja California para cursar el programa educativo (Serna y Castro, 2018).

Para realizar la investigación documental se revisó la información disponible en las páginas web de diversos organismos nacionales e internacionales, tales como ANUIES, Ministerios de Educación de España, Argentina, Chile y el Sistema Nacional de Información para la Educación Superior en Colombia. En lo referente a la investigación empírica se consultó la información arrojada en el “Estudio de preferencias vocacionales y demanda de carreras profesionales de los estudiantes de educación media superior de Baja California” (UABC, 2020).

Resultados

En México, la oferta de *programas iguales* a la licenciatura en Ciencias Computacionales, para el ciclo escolar 2022-2023, se ubicó en 8 universidades públicas. La oferta se distribuyó por zonas educativas de la siguiente manera: (a) Noroeste, Universidad Autónoma de Baja California, Universidad Autónoma de Sonora; (b) Noreste, Universidad Autónoma de Nuevo León; (c) Centro, Universidad Nacional Autónoma de México, Universidad Autónoma del Estado de Hidalgo, Benemérita Universidad Autónoma de Puebla; y (d) Sur-Sureste, Universidad Juárez Autónoma de Tabasco y Universidad Autónoma de Yucatán (ver tabla 1.17).

Tabla 1.17

Oferta de Programas iguales al programa de LCC en México

Universidad	Nombre del Plantel	Nombre del Programa	Matrícula	Nuevo ingreso	Lugares ofertados	Solicitudes	Tipo de sostenimiento
Noroeste							
Universidad Autónoma de Baja California	Facultad de Ciencias	Licenciatura en Ciencias Computacionales	116	16	16	29	Público
Universidad de Sonora	Departamento de Matemáticas,	Licenciatura en Ciencias de la Computación	188	55	40	265	
Noreste							
Universidad Autónoma de Nuevo León	Facultad de Ciencias Físico-Matemáticas	Licenciatura en Ciencias Computacionales	948	211	260	241	Público
Centro							
Universidad Nacional Autónoma de México		Licenciatura en Ciencias de la Computación	756	173	867	867	Público
Universidad Autónoma del Estado de Hidalgo	Escuela Superior de Huejutla	Licenciatura en Ciencias Computacionales	23	0	0	0	
	Instituto de Ciencias Básicas e Ingeniería	Licenciatura en Ciencias Computacionales	638	100	100	402	
Benemérita Universidad Autónoma de Puebla	Facultad de Ciencias de la Computación	Licenciatura en Ciencias de la Computación	602	141	150	332	
Sur-sureste							
Universidad Juárez Autónoma de Tabasco	División Académica de Ciencias Básicas	Licenciatura en Ciencias Computacionales	50	19	35	27	Público
Universidad Autónoma de Yucatán	Facultad de Matemáticas	Licenciatura en Ciencias de la Computación	108	32	34	43	

Nota. Adaptado de “Anuarios Estadísticos de Educación Superior, Ciclo escolar 2022-2023.” Derechos de Autor por ANUIES, 2023. La información corresponde al Subsistema de Universidades Públicas y al Campo específico de formación de implementación de las tecnologías de la información y la comunicación.

En cuanto a la oferta de *programas afines* a la licenciatura en Ciencias Computacionales, se identificaron 27 programas en 22 universidades (ANUIES, 2023). La Ingeniería en

Ciencias de la Computación destacó como el programa afín con la mayor cantidad de estudiantes de nuevo ingreso con 456 estudiantes (ver tabla 1.18).

Tabla 1.18*Oferta de Programas afines al programa de LCC en México*

Universidad	Nombre del Plantel	Nombre del Programa	Matrícula	Nuevo ingreso	Lugares ofertados	Solicitudes	Tipo de sostenimiento
Noroeste							
Universidad Autónoma de Chihuahua	Facultad de Ingeniería	Licenciatura en Ingeniería en Ciencias de Computo	278	95	130	122	Público
Centro de Enseñanza Técnica y Superior	Dirección de Educación Superior e Investigación (Mexicali)	Ingeniería en Ciencias Computacionales	103	32	31	37	Privado
	Dirección de Educación Superior e Investigación (Tijuana)	Ingeniería en Ciencias Computacionales	140	49	50	49	
Centro de Estudios Universitarios del Norte A.C	Centro de Estudios Universitarios del Norte A.C	Ingeniería en Ciencias Computacionales	3	0	20	0	
Noreste							
Universidad Autónoma de San Luis Potosí	Facultad de Ingeniería	Ingeniería en Sistemas Inteligentes	544	140	140	509	Público
Universidad Insuco	Universidad Insuco	Ingeniero en Sistemas Computacionales	42	2	2	2	Privado
	Campus Apodaca	Ingeniero en Sistemas Computacionales	52	25	50	25	
	Campus La Fe	Ingeniero en Sistemas Computacionales	23	5	5	5	
Instituto Metropolitano de Estudios Superiores de Tamaulipas	Instituto Metropolitano de Estudios Superiores de Tamaulipas	Ingeniero en Sistemas Computacionales	28	28	50	50	
Universidad del Atlántico	Campus Reynosa, Jarachina Sur	Ingeniero en Sistemas Computacionales	47	20	30	20	
	Campus Reynosa, Ribereña	Ingeniero en Sistemas Computacionales	101	0	20	0	

Universidad	Nombre del Plantel	Nombre del Programa	Matrícula	Nuevo ingreso	Lugares ofertados	Solicitudes	Tipo de sostenimiento
	Campus Río Bravo	Ingeniero en Sistemas Computacionales	5	0	15	0	
Universidad del Noreste	Universidad del Noreste	Ingeniería en Sistemas Computacionales y Electrónicos	68	36	36	36	
Universidad del Norte de Tamaulipas	Universidad del Norte de Tamaulipas	Ingeniero en Sistemas Computacionales	193	51	116	96	
Occidente							
Universidad Autónoma de Aguascalientes	Centro de Ciencias Básicas	Ingeniería en Computación Inteligente	225	48	50	165	Público
Universidad de Colima	Facultad de Ingeniería Mecánica y Eléctrica	Ingeniería en Computación Inteligente	194	61	65	61	
Universidad de Guanajuato	Campus Guanajuato, División de Ciencias Naturales y Exactas	Licenciatura en Computación Matemática	60	11	40	34	
Universidad de Guanajuato	Campus Irapuato-Salamanca, División de Ingenierías	Licenciatura en Ingeniería de Datos e Inteligencia Artificial	20	20	30	25	
Universidad de Guadalajara	Centro Universitario de Tonalá	Ingeniería en Ciencias Computacionales	549	90	90	198	
Centro							
Universidad Autónoma Metropolitana	Universidad Autónoma Metropolitana - Unidad Lerma	Licenciatura en Ingeniería en Computación y Telecomunicaciones	186	51	62	137	Público
Universidad Autónoma del Estado de Morelos	Instituto de Investigación en Ciencias Básicas y Aplicadas	Licenciatura en Ciencias Área Terminal en Ciencias Computacionales y Computación Científica	31	0	0	0	

Universidad	Nombre del Plantel	Nombre del Programa	Matrícula	Nuevo ingreso	Lugares ofertados	Solicitudes	Tipo de sostenimiento
Benemérita Universidad Autónoma de Puebla	Facultad de Ciencias de la Computación	Ingeniería en Ciencias de la Computación	2,141	456	500	984	
Instituto de Estudios Superiores en Arquitectura y Diseño, A.C.	Escuela de Humanidades	Ingeniería en Ciencias de la Computación	73	35	35	37	Privado
Instituto Poblano de Estudios Superiores A.C.	Área de Ciencias Sociales	Ingeniería en Ciencias de la Computación	27	17	30	17	
Sur-sureste							
Universidad Autónoma de Guerrero	Facultad de Ingeniería	Ingeniero en Computación	467	68	70	124	Pública
Universidad del Caribe	Universidad del Caribe	Ingeniería en Datos e Inteligencia Organizacional	341	82	150	126	Privada
Universidad del Sur	Universidad del Sur	Ingeniería en Ciencias Computacionales	44	31	31	31	

Nota. Adaptado de “Anuarios Estadísticos de Educación Superior, Ciclo escolar 2022-2023.” Derechos de Autor por ANUIES, 2023. La información corresponde al Subsistema de Universidades Públicas y al Campo específico de formación de implementación de las tecnologías de la información y la comunicación.

En el siguiente apartado se ofrece información detallada del comportamiento de la matrícula de Ciencias Computacionales durante los últimos cinco años en México.

Matrícula de los programas educativos iguales o afines

De acuerdo con la ANUIES (2023), la matrícula del ciclo escolar 2022-2023 correspondiente al subsistema de universidades públicas estatales y del campo detallado de formación en Ciencias Computacionales fue de 9, 414 estudiantes (76.09% hombres y 23.91% mujeres). Para el ciclo escolar mencionado se tuvieron 5, 096 solicitudes, se ofertaron 3, 350 lugares e ingresaron 2, 200 estudiantes.

En lo que respecta a los programas iguales a la licenciatura en Ciencias Computacionales, la matrícula fue de 3, 429 (equivalente al 36.43% de la matrícula total); el 80.99% fueron hombres y el 19.01% mujeres. Por su parte, la matrícula de los programas afines a dicha licenciatura fue de 5, 985 (63.57% de la matrícula total).

Durante los últimos cinco años, del ciclo escolar 2018-2019 al 2022-2023, la matrícula del campo detallado de Ciencias Computacionales ha presentado los siguientes cambios: del ciclo 2021-2022 al 2022-2023 tuvo el mayor crecimiento con el 15.34% y el menor crecimiento de 1.23% entre el ciclo escolar 2019-2020 al 2020-2021. (ver tabla 1.19).

Tabla 1.19

Porcentaje de crecimiento de la matrícula de LCC (2018-2019 a 2022-2023)

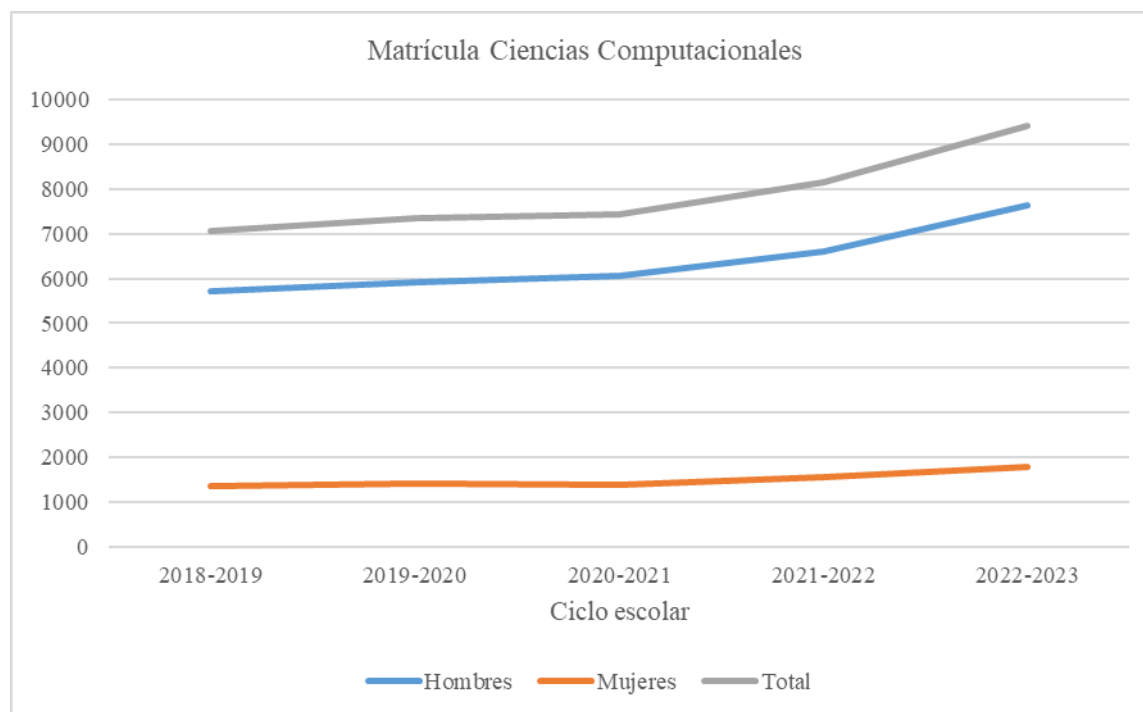
Ciclo escolar	Hombres		Mujeres		Total	% de crecimiento
	Matrícula	% de crecimiento	Matrícula	% de crecimiento		
2018-2019	5, 708	-	1, 368	-	7, 076	-
2019-2020	5, 928	3.85	1, 415	3.44	7, 343	3.77
2020-2021	6, 045	1.97	1, 388	-1.91	7, 433	1.23
2021-2022	6, 607	9.30	1, 555	12.03	8, 162	9.81
2022-2023	7, 628	15.45	1, 786	14.86	9, 414	15.34

Nota. El porcentaje de crecimiento de la matrícula se calculó entre ciclos escolares. Se tomó en consideración la matrícula del subsistema de Universidades Públicas Estatales; campo específico de formación en Innovación en tecnologías de la información y la comunicación; campo detallado de formación en ciencias computacionales. Elaboración propia a partir de “Anuarios Estadísticos de la Educación Superior”. ANUIES, (2023).

En términos globales, entre el ciclo escolar 2018-2019 al 2022-2023, la matrícula creció un 33.04%; es decir, hubo un aumento de 2, 338 estudiantes en cinco años. En este mismo sentido, el crecimiento de la matrícula de hombres fue del 33.64% y de las mujeres del 30.55%. En la figura 1.2 se observa el panorama general del comportamiento de la matrícula de la licenciatura en Ciencias Computacionales.

Figura 1.2

Matrícula de LCC de los últimos 5 años



Nota. Adaptado de “Anuarios Estadísticos de la Educación Superior”. ANUIES, 2023. Se tomó en consideración la matrícula del subsistema de Universidades Públicas Estatales; campo específico de formación en Innovación en tecnologías de la información y la comunicación; campo detallado de formación en ciencias computacionales.

En el siguiente apartado se presenta la oferta de los programas educativos iguales y afines a la licenciatura en Ciencias Computacionales de algunos países de Iberoamérica, tales como España, Argentina, Chile y Colombia. De manera general, se apreciará la

información respecto al nombre de los programas, institución que la oferta, plazas disponibles, créditos solicitados y tipo de sostenimiento.

En España, el Sistema Universitario Español (SUE) se integra por 86 universidades, 50 son de sostenimiento público y 36 corresponde al privado. En el año 2022 se ofrecieron 3, 112 programas (Ministerio de Educación de España, 2023a). El programa de estudios afín a la licenciatura en Ciencias Computacionales, en general, se denomina Ingeniería en Informática. Entre las características generales se resume lo siguiente: 82 universidades ofertan programas educativos afines, la mayoría son de sostenimiento público (70.73%). En total, se reportó la oferta de 9, 467 plazas en el año 2023; y los créditos que conforman el plan de estudios pueden ser en algunas universidades de 180 y en otras de 240 (Ministerio de Educación de España, 2023b). En la tabla 1.20 se muestran la información completa.

Tabla 1.20

Oferta de Programas iguales o afines al programa de LCC en España

Universidad	Centro/Facultad	Nombre del programa	Plazas ofertadas	Tipo de sostenimiento	Créditos
Universidad Miguel Hernández de Elche	Escuela Politécnica Superior de Elche	Ingeniería Informática en Tecnologías de la Información	50	Pública	240
Universidad de Salamanca	Facultad de Ciencias (Salamanca)	Ingeniería Informática	120		240
Universidad Pompeu Fabra	Departamento de Medicina y Ciencias de la Vida (Barcelona)	Bioinformática	-		180
	Escuela Superior Politécnica Tecnocampus (Barcelona)	Ingeniería Informática de Gestión y Sistemas de Información	50		240
	Escuela Superior de Comercio Internacional (Barcelona)	Bioinformática	40		180
Universidad de Murcia	Facultad de Informática (Murcia)	Ingeniería Informática	200		240
Universidad Autónoma de Madrid	Escuela Politécnica Superior (Madrid)	Ingeniería Informática	81		240
Universidad de Burgos	Escuela Politécnica Superior (Burgos)	Ingeniería Informática	175		240
Universidad de Las Palmas de Gran Canaria	Escuela de Ingeniería Informática (Las Palmas)	Ingeniería Informática	200		240
Universidad de Alicante	Escuela Politécnica Superior (Alicante)	Ingeniería Informática	190		240
Universidad de Huelva	Escuela Técnica Superior de	Ingeniería Informática	150		240

Universidad	Centro/Facultad	Nombre del programa	Plazas ofertadas	Tipo de sostenimiento	Créditos
Universidad de Sevilla	Ingeniería (Huelva) Escuela Técnica Superior de Ingeniería Informática (Sevilla)	Ingeniería Informática	120		240
Universidad Autónoma de Barcelona	Escuela de Ingeniería (Barcelona)	Ingeniería Informática	190		240
Universidad Pública de Navarra	Escuela Técnica Superior de Ingeniería Industrial, Informática y Telecomunicación (Navarra)	Ingeniería Informática	100		240
Universidad Complutense de Madrid	Facultad de Informática (Madrid)	Ingeniería Informática	115		240
Universidad Nacional de Educación a Distancia	Escuela Técnica Superior de Ingeniería Informática (Madrid)	Ingeniería Informática			240
Universidad de Girona	Escuela Politécnica Superior (Girona)	Ingeniería Informática	100		240
Universidad de Alcalá	Escuela Politécnica Superior (Madrid)	Ingeniería Informática	50		240
Universidad de Cádiz	Escuela Superior de Ingeniería (Cádiz)	Ingeniería Informática	120		240
Universidad de Málaga	Escuela Técnica Superior de Ingeniería Informática (Málaga)	Ingeniería Informática	140		240
Universidad de Salamanca	Escuela Politécnica Superior de Zamora (Zamora)	Ingeniería Informática en Sistemas de Información	42		240
Universidad de Córdoba	Escuela Politécnica Superior de Córdoba (Córdoba)	Ingeniería Informática	110		240
Universidad de Oviedo	Escuela Politécnica de Ingeniería de Gijón (Asturias)	Ingeniería Informática en Tecnologías de la Información	115		240
Universidad de A Coruña	Facultad de Informática (A Coruña)	Ingeniería Informática	264		240
Universidad de Santiago de Compostela	Escuela Técnica Superior de Ingeniería (A Coruña)	Ingeniería Informática	75		240
Universidad Politécnica de Valencia	Escuela Técnica Superior de Ingeniería Informática (Valencia)	Ingeniería Informática	400		240
	Escuela Politécnica Superior de Alcoy (Alicante)	Ingeniería Informática	75		240
Universidad de Jaén	Escuela Politécnica Superior (Jaén) (Jaén)	Ingeniería Informática	150		240
Universidad de León	Escuela de Ingenierías Industrial, Informática y	Ingeniería Informática	100		240

Universidad	Centro/Facultad	Nombre del programa	Plazas ofertadas	Tipo de sostenimiento	Créditos
Universidad de Lleida	Aeroespacial (León) Escuela Politécnica Superior (Lleida)	Ingeniería Informática	105		240
Universidad de Castilla-La Mancha	Escuela Superior de Informática (Ciudad Real)	Ingeniería Informática	160		240
	Facultad de Ciencias Sociales y Tecnologías de la Información (Toledo)	Ingeniería Informática	60		240
	Escuela Superior de Ingeniería Informática (Albacete)	Ingeniería Informática	160		240
Universidad de Vigo	Escuela Superior de Ingeniería Informática (Ourense)	Ingeniería Informática	147		240
Universidad de Extremadura	Escuela Politécnica (Cáceres)	Ingeniería Informática en Ingeniería del Software	80		240
Universidad Jaume I de Castellón	Escuela Superior de Tecnología y Ciencias Experimentales (Castellón)	Ingeniería Informática	90		240
Universidad Rovira i Virgili	Escuela Técnica Superior de Ingeniería (Tarragona)	Ingeniería Informática	110		240
Universidad de Zaragoza	Escuela Universitaria Politécnica (Teruel)	Ingeniería Informática	25		240
	Escuela de Ingeniería y Arquitectura (Zaragoza)	Ingeniería Informática	110		240
Universidad Pompeu Fabra	Escuela de Ingeniería (Barcelona)	Ingeniería en Informática	55		240
Universidad de Barcelona	Facultad de Matemáticas e Informática (Barcelona)	Ingeniería Informática	70		240
Universidad de Granada	Facultad de Educación, Economía y Tecnología de Ceuta (Ceuta)	Ingeniería Informática	50		240
	Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación (Granada)	Ingeniería Informática	260		240
Universidad de Almería	Escuela Superior de Ingeniería (Almería)	Ingeniería Informática	150		240
Universitat de València (Estudi General)	Escuela Técnica Superior de Ingeniería (Valencia)	Ingeniería Informática	50		240
Universidad de La Laguna	Escuela Superior de Ingeniería y	Ingeniería Informática	155		240

Universidad	Centro/Facultad	Nombre del programa	Plazas ofertadas	Tipo de sostenimiento	Créditos
Universidad de Sevilla	Tecnología (Santa Cruz de Tenerife) Escuela Técnica Superior de Ingeniería Informática (Sevilla)	Ingeniería Informática	225		240
Universidad Carlos III de Madrid	Escuela Politécnica Superior. Sección Colmenarejo (Madrid)	Ingeniería Informática / Licenciatura en Ciencias e Ingeniería Informática	25		240
	Escuela Politécnica Superior (Madrid)	Ingeniería Informática / Licenciatura en Ciencias e Ingeniería Informática	180		240
Universidad Rey Juan Carlos	Escuela Técnica Superior de Ingeniería Informática. Campus de Madrid (Madrid)	Ingeniería Informática	50		240
	Escuela Técnica Superior de Ingeniería Informática. Campus de Móstoles (Madrid)	Ingeniería Informática	33		240
Universidad de la Rioja	Facultad de Ciencia y Tecnología de la Universidad de La Rioja (La Rioja)	Ingeniería Informática	50		240
Universidad Politécnica de Catalunya	Escuela Universitaria Politécnica de Mataró (Barcelona)	Ingeniería Informática	-		240
	Escuela Politécnica Superior de Ingeniería de Vilanova i la Geltrú (Barcelona)	Ingeniería Informática	50		240
	Facultad de Informática (Barcelona)	Ingeniería Informática	400		240
	Facultad de Informática (Barcelona)	Bioinformática	-		180
Universidad Pablo de Olavide	Escuela Politécnica Superior (Sevilla)	Ingeniería Informática en Sistemas de Información	75		240
Universidad de las Illes Balears	Escuela Politécnica Superior (Illes Balears)	Ingeniería Informática	150		240
Universidad Autónoma de Barcelona	Escuela Universitaria de Informática Tomàs Cerdà (Barcelona)	Informática y Servicios	50		240
Universidad de Valladolid	Escuela de Ingeniería Informática de Segovia (Segovia)	Ingeniería Informática de Servicios y Aplicaciones	50		240
	Escuela de Ingeniería	Ingeniería	115		240

Universidad	Centro/Facultad	Nombre del programa	Plazas ofertadas	Tipo de sostenimiento	Créditos
	Informática de Valladolid (Valladolid)	Informática			
Universidad del País Vasco/Euskal Herriko Unibertsitatea	Escuela de Ingeniería de Bilbao (Bizkaia)	Ingeniería Informática de Gestión y Sistemas de Información	70		240
	Escuela de Ingeniería de Vitoria-Gasteiz (Araba/Álava)	Ingeniería Informática de Gestión y Sistemas de Información	45		240
Universidad de Sevilla	Escuela Técnica Superior de Ingeniería Informática (Sevilla)	Ingeniería Informática	145		240
Universidad de Extremadura	Centro Universitario de Mérida (Badajoz)	Ingeniería Informática en Tecnologías de la Información	40		240
	Escuela Politécnica (Cáceres)	Ingeniería Informática en Ingeniería de Computadores	80		240
Universidad de Oviedo	Escuela de Ingeniería Informática (Asturias)	Ingeniería Informática del Software	150		240
Universidad del País Vasco/Euskal Herriko Unibertsitatea	Facultad de Informática (Gipuzkoa)	Ingeniería Informática	135		240
Universidad de Cantabria	Facultad de Ciencias (Cantabria)	Ingeniería Informática	60		240
Universidad Politécnica de Madrid	Escuela Técnica Superior de Ingenieros Informáticos (Madrid)	Matemáticas e Ingeniería Informática	50		240
			280		240
Universidad de Barcelona	Facultad de Biología (Barcelona)	Bioinformática	-		180
Universidad Politécnica de Valencia	Escuela Técnica Superior de Ingeniería Informática (Valencia)	Informática Industrial y Robótica	-		240
	Escuela Politécnica Superior de Alcoy (Alicante)	Industrial y Robótica	-		240
Universidad Autónoma de Barcelona	Facultad de Biociencias (Barcelona)	Bioinformática	-		180
Universidad Oberta de Catalunya	Universitat Oberta de Catalunya	Ingeniería Informática	1600	Privado	240
Universidad Internacional Isabel I de Castilla	Facultad de Ciencias y Tecnología (Burgos)	Ingeniería Informática	-		240
Universidad Europea de Madrid	Escuela de Arquitectura, Ingeniería y Diseño. Campus Villaviciosa y Alcobendas	Ingeniería Informática	-		240

Universidad	Centro/Facultad	Nombre del programa	Plazas ofertadas	Tipo de sostenimiento	Créditos
	(Madrid)				
Universidad Europea del Atlántico	Escuela Politécnica Superior (Cantabria)	Ingeniería Informática	-		240
Universidad San Jorge	Facultad de Ciencias de la Salud (Zaragoza)	Bioinformática	-		180
	Escuela de Arquitectura y Tecnología (Huesca)	Ingeniería Informática	-		240
Mondragon Unibertsitatea	Escuela Politécnica Superior (Gipuzkoa)	Ingeniería en Informática	-		240
Universidad de Deusto	Facultad de Ingeniería (Bizkaia)	Ingeniería Informática	-		240
Universidad Antonio de Nebrija	Escuela Politécnica Superior (Madrid)	Ingeniería Informática	-		240
Universidad a Distancia de Madrid	Facultad de Ciencias Técnicas e Ingeniería (Madrid)	Ingeniería Informática	-		240
Universidad Alfonso X El Sabio	Escuela Politécnica Superior (Madrid)	Ingeniería Informática	-		240
Universidad Pontificia de Salamanca	Escuela Superior de Ingeniería y Arquitectura (Madrid)	Ingeniería Informática	-		240
	Facultad de Informática (Salamanca)	Ingeniería Informática	-		240
Universidad Loyola Andalucía	Escuela Técnica Superior de Ingeniería (Sede de Sevilla) (Sevilla)	Ingeniería Informática y Tecnologías Virtuales	-		240
Universidad Internacional Valenciana	Escuela Superior de Ingeniería, Ciencia y Tecnología (Valencia/València)	Ingeniería Informática	-		240
Universidad Europea Miguel de Cervantes	Escuela Politécnica Superior (Valladolid)	Ingeniería Informática	-		240
Universidad Católica San Antonio	Escuela Politécnica Superior (Murcia)	Ingeniería Informática	-		240
Universidad Internacional de la Rioja	Escuela Superior de Ingeniería y Tecnología (La Rioja)	Ingeniería Informática	-		240
Universidad Católica Santa Teresa de Jesús de Avila	Facultad de Ciencias y Artes (Avila)	Bioinformática	-		180
Universidad Ramón Llull	Escuela Técnica Superior de Ingeniería La Salle (Barcelona)	Ingeniería Informática	-		240
Universidad Francisco de Vitoria	Escuela Politécnica Superior (Madrid)	Ingeniería Informática	-		240
Universidad Internacional de la Empresa	Facultad de Ciencias Sociales Aplicadas y de la Comunicación (Madrid)	Ingeniería Informática	-		240
Universidad Camilo José Cela	Facultad de Tecnología y Ciencia (Madrid)	Ingeniería Informática Biomédica	-		240
		Ingeniería Informática	-		240
CUNEF Universidad	Escuela Politécnica	Ingeniería	-		240

Universidad	Centro/Facultad	Nombre del programa	Plazas ofertadas	Tipo de sostenimiento	Créditos
Universidad San Pablo-CEU	Superior (Madrid) Facultad de Medicina (Madrid)	Informática Bioinformática y Datos Masivos (Big Data)	-		240
Universidad Católica Santa Teresa de Jesús de Ávila	Facultad de Ciencias y Artes (Ávila)	Ingeniería Informática - Sistemas de Información	-		240

Nota. Adaptado de “Qué estudiar y dónde en la Universidad,” (<http://siiu.universidades.gob.es/QEDU/>). Derechos de Autor por Ministerio de Educación de España, 2023.

Asimismo, se indagó la oferta respectiva en Argentina, de las 131 instituciones registradas en el Ministerio de Educación, la licenciatura en Ciencias Computacionales se ofrece en 50 universidades de sostenimiento público y 29 del privado. La mayoría tienen una duración entre cuatro y cinco años (ver tabla 1.21).

Tabla 1.21

Oferta de Programas iguales o afines al programa de LCC en Argentina

Universidad	Centro/Facultad	Nombre del programa	Duración	Tipo de sostenimiento
Instituto Tecnológico de Buenos Aires	Departamento de Informática	Ingeniero en Informática	5	Privado
Instituto Universitario de Gendarmería Nacional	Unidad Académica General Don Martín Miguel de Güemes	Licenciado en Gestión y Seguridad de las Tecnologías de la Información	4	Privado
Instituto Universitario Escuela Argentina de Negocios	Unidad Académica Capital Federal - Sede Central	Licenciado en Tecnología Informática	4	Privado
	Unidad Académica Provincia de Buenos Aires - Sede Martínez	Licenciado en Tecnología Informática	4	
Pontificia Universidad Católica Argentina Santa María de los Buenos Aires	Facultad de Ingeniería y Ciencias Agrarias	Ingeniero en Informática	5	Privado
		Licenciado/a en Ciencias de Datos	4	
		Ingeniero en Sistemas Informáticos	5	
		Licenciado/a en Producción de Simuladores y Videojuegos	4	
Universidad Abierta Interamericana	Facultad de Tecnología Informática	Licenciado en Gestión de Tecnología Informática	4	Privado
		Ingeniero en Sistemas Informáticos	5	

Universidad	Centro/Facultad	Nombre del programa	Duración	Tipo de sostenimiento
Universidad Adventista del Plata	Facultad de Ciencias Económicas y de la Administración	Ingeniero en Sistemas de Información	5	Privado
Universidad Argentina de la Empresa	Facultad de Arquitectura y Diseño	Licenciado/a en Diseño de Productos Interactivos y Realidad Virtual	4	Privado
		Ingeniero en Informática	5	
		Licenciado/a en Desarrollo de Videojuegos	4	
		Licenciado en Sistemas	5	
Universidad Argentina John F. Kennedy	Escuela de Sistemas	Ingeniero en Informática	5	Privado
Universidad Atlántida Argentina	Anexo Mar del Plata	Licenciado en Informática	4	Privado
		Ingeniero en Informática	5	
Universidad Austral	Facultad de Ingeniería	Ingeniero en Informática	5	Privado
Universidad Blas Pascal	Vicerrectorado de Asuntos Académicos	Ingeniero Informático	5	Privado
		Ingeniero en Sistemas	5	
		Licenciado en Gestión de Sistemas y Negocios	4	
Universidad CAECE	Departamento de Sistemas	Licenciado en Sistemas	5	Privado
		Ingeniero de Sistemas	5	
		Ingeniero en Computación	5	
Universidad Católica de Córdoba	Facultad de Ingeniería	Ingeniero en Computación	5	Privado
Universidad Católica de La Plata	Facultad de Ciencias Exáctas e Ingeniería	Licenciado en Sistemas	5	
Universidad Católica de Salta	Facultad de Ingeniería	Ingeniero en Informática	5	
Universidad Católica de Santa Fe	Facultad de Arquitectura	Licenciado/a en Diseño de Videojuegos y Tecnologías Creativas	4	Privado
		Analista de Sistemas	4	
		Ingeniero en Informática	5	
Universidad Católica de Santiago del Estero	Departamento Académico Rafaela	Ingeniero en Informática	5	Privado
	Facultad de Ciencias para la Innovación y el Desarrollo	Ingeniero en Informática	5	
Universidad Champagnat	Facultad de Informática y Diseño	Licenciado en Sistemas de Información	5	Privado
		Ingeniero en Informática	5	
Universidad de Belgrano	Facultad de Ingeniería Y Tecnología Informática	Licenciado en Sistemas de Información	5	Privado
		Ingeniero en Informática	5	

Universidad	Centro/Facultad	Nombre del programa	Duración	Tipo de sostenimiento
Universidad del Aconcagua	Facultad de Ciencias Sociales y Administrativas	Licenciado en Informática y Desarrollo de Software	4	Privado
Universidad de la Cuenca del Plata	Facultad de Ingeniería y Tecnología	Ingeniero en Sistemas de Información	5	Privado
		Licenciado en Sistemas de Información	5	
Universidad de la Fraternidad de Agrupaciones Santo Tomás de Aquino	Facultad de Ingeniería	Ingeniero en Informática	5	Privado
Universidad de la Marina Mercante	Facultad de Administración y Economía	Licenciado en Administración de Sistemas	4	Privado
	Facultad de Ingeniería	Ingeniero en Sistemas	5	
Universidad del CEMA	Departamento de Ingeniería y Matemática	Ingeniero en Informática	5	Privado
Universidad del Centro Educativo Latinoamericano	Facultad de Química	Ingeniero en Sistemas de Información	5	--
Universidad del Norte Santo Tomás de Aquino	Centro Universitario Concepción	Ingeniero en Informática	5	Privado
	Complejo Educativo Nuestra Señora del Pilar	Ingeniero en Informática	5	
Universidad del Salvador	Complejo Educativo Nuestra Señora del Pilar	Licenciado en Sistemas de Información	4	Privado
		Ingeniero en Computación	5	
Universidad de Mendoza	Facultad de Ingeniería	Ingeniero en Informática	5	Privado
		Ingeniero en Informática	5	
Universidad de Morón	Escuela Superior de Ingeniería, Informática y Ciencias Agroalimentarias	Ingeniero en Informática	5	Privado
		Licenciado en Sistemas	5	
		Ingeniero en Informática	5	
		Licenciado/a en Ciberseguridad	4	
Universidad de Palermo	Facultad de Ingeniería	Licenciado/a en Inteligencia Artificial	4	Privado
		Licenciado en Informática	4.5	
		Licenciado en Redes y Comunicación de Datos	4	
		Ingeniero en Software	4.5	
Universidad Empresarial Siglo 21		Licenciado/a en Ciencias de Datos	4	Privado
		Licenciado/a en	4	

Universidad	Centro/Facultad	Nombre del programa	Duración	Tipo de sostenimiento
		Inteligencia Artificial y Robótica		
		Licenciado/a en Seguridad Informática	4	
Universidad Gastón Dachary	Departamento de Ingeniería y Ciencias de la Producción	Ingeniero en Informática	5	Privado
Universidad Maimónides	Escuela de Comunicación Multimedial	Licenciada/o en Tecnología de Videojuegos y Diseño Interactivo	4	Privado
Universidad Metropolitana para la Educación y el Trabajo	Facultad de Tecnología Aplicada	Licenciado en Informática	5	--
Universidad Nacional Arturo Jauretche	Instituto de Ingeniería y Agronomía	Ingeniero en Informática	5	Público
Universidad Nacional de Avellaneda	Departamento de Tecnología y Administración	Ingeniero en Informática	5	Público
Universidad Nacional de Catamarca	Facultad de Tecnología y Ciencias Aplicadas	Ingeniero en Informática	5	Público
Universidad Nacional de Chilecito	Escuela de Ingeniería	Ingeniero en Sistemas	5	Público
		Licenciado/a en Sistemas	5	
Universidad Nacional de Córdoba	Facultad de Ciencias Exactas, Físicas y Naturales	Ingeniero en Computación	5	Público
	Facultad de Matemática, Astronomía, Física y Computación	Licenciado en Ciencias de la Computación	5	
		Licenciado en Ciencias de la Computación	5	
Universidad Nacional de Cuyo	Facultad de Ingeniería			Público
Universidad Nacional de Entre Ríos	Facultad de Ciencias de la Administración	Licenciado en Sistemas	5	Público
Universidad Nacional de General Sarmiento	Instituto de Industria	Licenciado en Sistemas	5	Público
Universidad Nacional de Hurlingham	Instituto de Tecnología e Ingeniería	Licenciado en Informática	5	Público
Universidad Nacional de Jujuy	Extensión Aúlica Libertador General San Martín	Ingeniero en Informática	5	Público
	Facultad de Ingeniería	Licenciado en Sistemas	5	
Universidad Nacional de La Matanza	Departamento de Ingeniería e Investigaciones Tecnológicas	Ingeniero en Informática	6	Público
Universidad Nacional de Lanús	Departamento de Desarrollo Productivo y Tecnológico	Licenciado en Sistemas	5	Público
Universidad Nacional de La Pampa	Facultad de Ingeniería	Ingeniero en Computación	5	Público
		Ingeniero en Sistemas	5	
Universidad Nacional de la Patagonia Austral	Unidad Académica Caleta Olivia	Ingeniero en Sistemas	5	Público

Universidad	Centro/Facultad	Nombre del programa	Duración	Tipo de sostenimiento
Universidad Nacional de la Patagonia San Juan Bosco	Unidad Académica Río Gallegos	Licenciado en Sistemas	5	Público
	Facultad de Ingeniería	Licenciado en Informática	5	
		Licenciado en Sistemas	5	
Universidad Nacional de La Plata	Facultad de Informática	Licenciado en Informática	5	Público
	Facultad de Informática e Ingeniería	Licenciado en Sistemas	5	
		Ingeniero en Computación	5	
Universidad Nacional de La Rioja	Departamento Académico de Ciencias Exactas, Físicas y Naturales	Ingeniero en Sistemas de Información	5	Público
		Licenciado en Sistemas de Información	5	
		Ingeniero de Sistemas	5	
Universidad Nacional del Centro de la Provincia de Buenos Aires	Facultad de Ciencias Exactas	Ingeniero en Sistemas	5	Público
Universidad Nacional del Chaco Austral	Departamento de Ciencias Básicas y Aplicadas	Ingeniero en Sistemas de Información	5	Público
Universidad Nacional del Comahue	Facultad de Informática	Licenciado en Ciencias de la Computación	5	Público
		Licenciado en Sistemas de Información	5	
		Licenciada/o en Ciencia de Datos	5	
Universidad Nacional del Litoral	Facultad de Ingeniería Química	Ingeniero en Informática	5	Público
	Facultad de Ingeniería y Ciencias Hídricas	Licenciado en Sistemas de Información	5	
	Facultad de Ciencias Exactas y Naturales y Agrimensura	Ingeniero en Informática	5	
Universidad Nacional del Nordeste	Escuela de Tecnología	Licenciado en Sistemas	5	Público
		Licenciado en Informática	5	
		Licenciado en Sistemas	5	
Universidad Nacional del Oeste	Escuela de informática	Licenciado en Informática	5	Público
		Ingeniero en Computación	5	
		Ingeniero en Sistemas de Información	5	
Universidad Nacional del Sur	Departamento de Ciencias e Ingeniería de la Computación	Licenciado en Ciencias de la Computación	10 Cuatrimestres	Público
		Ingeniero en Sistemas de Computación	5	
		Licenciado en Sistemas de Información	11 Cuatrimestres	
Universidad Nacional de Luján	Centro Regional Chivilcoy	Ingeniero en Computación	5	Público
Universidad Nacional de Mar del Plata	Facultad de Ingeniería	Ingeniero en	5	Público

Universidad	Centro/Facultad	Nombre del programa	Duración	Tipo de sostenimiento
Universidad Nacional de Misiones	Facultad de Ciencias Exactas, Químicas y Naturales	Informática Licenciado en Sistemas de Información	5	Público
	Facultad de Ingeniería	Ingeniero/a en Computación	5	
Universidad Nacional de Quilmes	Departamento de Ciencia y Tecnología	Licenciado en Informática	5.5	Público
		Ingeniero/a en Computación	5	Público
Universidad Nacional de Rafaela	Secretaría Académica	Licenciado/a en Agroinformática	4.5	
		Licenciado en Producción de Videojuegos y Entretenimiento Digital	4.5	
		Licenciado en Ciencias de la Computación	5	Público
Universidad Nacional de Río Cuarto	Facultad de Ciencias Exactas, Físico-Químicas y Naturales	Ingeniero/a en Computación	5	Público
Universidad Nacional de Río Negro		Licenciado en Sistemas	5	
		Licenciado en Ciencias de la Computación	5	Público
Universidad Nacional de Rosario	Facultad de Ciencias Exactas, Ingeniería y Agrimensura	Licenciado en Ciencias de la Computación	5	Público
Universidad Nacional de Salta	Facultad de Ciencias Exactas	Licenciado en Análisis de Sistemas	5	Público
Universidad Nacional de San Antonio de Areco	Escuela de Desarrollo Productivo y Tecnológico	Licenciado en Informática	5	Público
	Sede San Antonio de Areco	Licenciado en Informática	5	
Universidad Nacional de San Juan	Facultad de Ciencias Exactas, Físicas y Naturales	Licenciado en Ciencias de la Computación	5	Público
		Licenciado en Sistemas de Información	5	
		Ingeniero en Computación	5	Público
Universidad Nacional de San Luis	Facultad de Cs. Físico-Matemáticas y Naturales	Ingeniero en Informática	5	
		Licenciado en Ciencias de la Computación	5	
Universidad Nacional de San Martín	Escuela de Ciencia y Tecnología	Licenciada/o en Ciencia de Datos	4	Público
Universidad Nacional de Santiago del Estero	Facultad de Ciencias Exactas y Tecnologías	Licenciado en Sistemas de Información	5	Público
Universidad Nacional de Tierra del Fuego, Antártida e Islas del Atlántico Sur	Instituto de Desarrollo Económico e Innovación	Licenciado/a en Sistemas	5	Público
	Sede Ushuaia	Licenciado/a en Sistemas	5	
Universidad Nacional de Tres de Febrero	Secretaría Académica	Ingeniero en Computación	5	Público
		Licenciado en	4	

Universidad	Centro/Facultad	Nombre del programa	Duración	Tipo de sostenimiento
Universidad Nacional de Tucumán	Facultad de Ciencias Exactas y Tecnología	Sistemas de Información Geográfica	5	Público
		Ingeniero/a en Computación		
		Ingeniero en Informática		
Universidad Nacional de Villa Mercedes	Escuela de Ingeniería y Ciencias Ambientales	Licenciado en Informática	5	Público
		Ingeniero en Sistemas de Información		
Universidad Nacional Guillermo Brown	Rectorado	Licenciado/a en Ciencias de Datos	4.5	Público
		Licenciada/o en Gestión de Tecnologías de la Información		
Universidad Nacional José C. Paz	Departamento de Economía, Producción e Innovación Tecnológica	Licenciado/a en Producción y Desarrollo de Videojuegos	5	Público
		Ingeniero en Sistemas de Información		
Universidad Tecnológica Nacional	Facultad Regional Buenos Aires	Licenciado en Seguridad en Tecnologías de la Información y Comunicaciones	4	Público
Universidad Autónoma de Entre Ríos	Facultad de Ciencia y Tecnología	Licenciado en Sistemas de Información	5	Público
Universidad de Buenos Aires	Facultad de Ciencias Exactas y Naturales	Licenciado/a en Ciencias de Datos	5.5	Público
	Facultad de Ingeniería	Licenciado en Ciencias de la Computación	10 Cuatrimestres	
		Ingeniero en Informática	12 Cuatrimestres	
Universidad de la Defensa Nacional	Centro Regional Universitario Córdoba IUA	Licenciado en Análisis de Sistemas	9 Cuatrimestres	Público
		Ingeniero en Informática	5	
Universidad del Chubut	Ambiente, Producción y Desarrollo Sostenible	Licenciado en Redes y Telecomunicaciones	10 Cuatrimestres	Público

Nota. Adaptado de “Guía de Carreras –Universitarias. Carreras de Pregrado y Grado,” (https://guiadecarreras.siu.edu.ar/carreras_de_pregrado_y_grado.php). Derechos de Autor por Ministerio de Educación de Argentina, 2023.

En Chile, de las 138 Instituciones de Educación Superior, se identificaron dos universidades de sostenimiento público que ofertan la Licenciatura en Ciencias Computacionales: Universidad de Santiago de Chile y Universidad de la Serena. La duración de los programas es de 5 años (ver tabla 1.22).

Tabla 1.22

Oferta de Programas iguales o afines al programa de LCC en Chile

Universidad	Centro/Facultad	Nombre del programa	Duración/Años	Tipo de sostenimiento
Universidad de Santiago de Chile	Casa Central	Licenciatura en Ciencias de la Computación	5	Público
Universidad de la Serena	Escuela de Ingeniería en Computación	Licenciatura en Ciencias de la Computación	5	
Universidad de Bío-Bío	Facultad de Ciencias Empresariales	Ingeniería de Ejecución en Computación e Informática	4	
Universidad Tecnológica Metropolitana	--	Ingeniería en Informática	4	
Universidad Adolfo Ibáñez	--	Ingeniería en Ciencias Computacionales	5	Privado
Universidad Bernardo O'Higgins	Facultad de Ingeniería	Ingeniería en Informática	4	
Universidad Católica Cardenal Raúl Silva Henríquez	Facultad de Ingeniería y Empresas	Ingeniería en Tecnologías de la Información y Comunicaciones	5	
Universidad de las Américas	Facultad de Ingeniería y Negocios	Ingeniería de Ejecución en Informática	4	
Universidad Gabriela Mistral	Escuela de Negocios y Tecnología	Ingeniería en Informática	--	
Universidad Mayor	--	Ingeniería en Informática	4	
Universidad Tecnológica de Chile INACAP	--	Ingeniería en Informática	4	

Nota. Adaptado de “¿Dónde y qué estudiar?,” (<https://www.mifuturo.cl/buscador-de-carreras/?tipo=carrera>). Derechos de Autor por Ministerio de Educación de Chile, 2023.

Por último, en Colombia, los programas afines a la licenciatura en Ciencias Computacionales se ubicaron en 29 universidades, la mayoría ofertan Ingenierías en

Sistemas Computacionales; el 44.83% son de sostenimiento público y el 55.17% privado. Los créditos oscilan entre 134 y 183. Por su parte, el plan de estudios está diseñado para implementarse entre cuatro y seis años (ver tabla 1.23).

Tabla 1.23

Oferta de Programas iguales o afines al programa de LCC en Colombia

Universidad	Nombre del programa	Sostenimiento	Créditos	Duración/se mestres
Escuela Tecnológica Instituto Técnico Central	Ingeniería de Sistemas	Público	177	11
Universidad de Caldas	Ingeniería de Sistemas y Computación	Público	175	10
Universidad de Cartagena	Ingeniería de Sistemas	Público	179	10
Universidad de Nariño	Ingeniería de Sistemas	Público	167	10
Universidad Francisco de Paula Santander	Ingeniería de Sistemas	Público	167	10
Universidad Distrital-Francisco José de Caldas	Ingeniería en Telemática	Público	172	10
Universidad del Magdalena - Unimagdalena	Ingeniería de Sistemas	Público	160	10
Universidad Pedagógica y Tecnológica de Colombia - UPTC	Ingeniería de Sistemas y Computación	Público	175	10
Universidad Tecnológica de Pereira - UTP	Ingeniería de Sistemas y Computación	Público	183	10
Universidad del Quindío	Ingeniería de Sistemas y Computación	Público	163	10
Escuela Militar de Aviación Marco Fidel Suárez	Ingeniería Informática	Público	165	8
Politécnico Colombiano Jaime Isaza Cadavid	Ingeniería Informática	Público	166	10
Colegio Mayor del Cauca	Ingeniería Informática	Público	167	10
Universidad del Norte	Ingeniería de Sistemas y Computación	Privado	155	10
Universidad de San Buenaventura	Ingeniería de Sistemas	Privado	160	10
Universidad ICESI	Ingeniería de Sistemas	Privado	167	10
Universidad Autónoma de Occidente	Ingeniería Informática	Privado	156	9
Fundación Universitaria Konrad Lorenz	Ingeniería de Sistemas	Privado	142	9
Institución Universitaria Salazar y Herrera	Ingeniería de Sistemas	Privado	170	10
Universidad Escuela Colombiana de Ingeniería Julio Garavito	Ingeniería de Sistemas	Privado	158	10
Universidad Católica de Colombia	Ingeniería de Sistemas y Computación	Privado	149	10
Universidad ICESI	Ingeniería Telemática	Privado	167	10
Universidad Autónoma de Manizales	Ingeniería de Sistemas	Privado	175	12
Pontificia Universidad Javeriana	Ingeniería de Sistemas y Computación	Privado	170	10
Universidad de La Sabana	Ingeniería Informática	Privado	175	10
Universidad de Los Andes	Ingeniería de Sistemas y	Privado	134	8

	Computación			
Universidad Antonio Nariño	Ingeniería de Sistemas y Computación	Privado	157	10
Universidad Tecnológica de Bolívar	Ingeniería de Sistemas y Computación	Privado	162	10
Universidad EIA	Ingeniería de Sistemas y Computación	Privado	177	10

Nota. Adaptado de “Sistema Nacional de Información para la Educación Superior en Colombia 2022”. Ministerio de Educación de Colombia (2022).

<https://hecaa.mineducacion.gov.co/consultaspublicas/programas>

Demanda vocacional a nivel estatal

La Universidad Autónoma de Baja California realizó en el año 2020 el “Estudio de preferencias vocacionales y demanda de carreras profesionales de los estudiantes de educación media superior de Baja California”. En este participaron 17, 024 estudiantes de educación media superior de los cinco Municipios (el 54% fueron mujeres y el 46% hombres). Los instrumentos utilizados fueron los que a continuación se mencionan: (a) Escala de Autopercepción de Competencias, en la que evaluaron las habilidades de los participantes en cuatro competencias: tecnológica, lingüística, matemática y socioafectiva; (b) Inventario de Intereses y Preferencias Profesionales, en el que valoraron 14 campos profesionales; (c) listado de carreras, en la que seleccionaron y ordenaron las tres carreras profesionales que más les gustaría estudiar; y (d) listado de preferencias de modalidad de estudios.

Entre las características de los participantes, se especificó que la edad media fue de 17 años (73%). La distribución de la población encuestada por subsistemas de Educación Media Superior fue la siguiente: COBACH (43%), CECyTE (22%), UEMSTIS (15%), Federal (9%), CONALEP (6%), y DGECyTM (5%). La distribución por Municipio: Tijuana (45%), Mexicali (33%), Ensenada (12%), Tecate (5%), y Playas de Rosarito (5%). En cuanto a la escolaridad de los padres, el nivel promedio de estudio fue la secundaria (madres 38.5%, padres 33%), seguido de la preparatoria (madres 29.7% y padres 32%). La expectativa de estudios del 97.2% de los estudiantes fue dar continuidad con su preparación profesional al finalizar el bachillerato. El 69% indicó que no trabaja, mientras el 31% sí reportó un trabajo remunerado, ya sea en negocio familiar (11%), empleado en negocio de servicios (7%), o empleado de un comercio (7%).

En lo referente a los resultados de la escala de autopercepción de competencias, en específico de las competencias tecnológicas, la mayoría de los estudiantes refirieron sentirse capaces para realizar tareas en las que usen las tecnologías (comunicarse con otras personas, realizar descargas en internet, navegar en internet, realizar documentos escritos con procesador de textos, y realizar búsquedas bibliográficas). En contraparte, el 27% de los participantes estimaron no sentirse capaces para instalar y desinstalar programas informáticos en un ordenador, y el 23% tampoco cree tener capacidad para diseñar, crear y modificar hojas de cálculo con algún programa informático. En la competencia lingüística, el 61% consideró tener capacidad para expresarse con claridad de manera oral y el 59% para comunicarse de manera escrita. Por el contrario, al 33% les cuesta expresarse en idioma inglés y obtener conclusiones a partir de otras ideas. En la competencia matemática, el 41% de los participantes consideraron tener capacidad para formular y resolver problemas aplicando diferentes enfoques; sin embargo, el 27% no se sienten capaces para argumentar la solución obtenida de un problema con métodos numéricos, gráficos o analíticos e interpretar las situaciones reales, hipotéticas o formales que requiere utilizar el pensamiento matemático. Por último, en las competencias socio afectivas, la mayoría señaló poseer la capacidad para plantearse metas (77%), aprovechar al máximo sus recursos (73%), solicitar apoyo y tomar decisiones que les generen bienestar (66%); únicamente el 12% puntualizó no sentirse capaz de cultivar relaciones sanas (UABC, 2020).

En cuanto a la modalidad de estudio, el 67.1% de los estudiantes eligieron como primera opción la modalidad presencial; y como segunda opción prefirieron la modalidad a distancia, seguida de la modalidad en línea, modalidad semipresencial y Modalidad MOOCs.

En el tema de la demanda vocacional a nivel estatal, y de acuerdo con el inventario de intereses y preferencias, el campo profesional científico experimental ocupó el décimo lugar en las categorías de selección Muy Alto y Alto, entre 14 campos profesionales expuestos. Específicamente, la licenciatura en Ciencias Computacionales fue seleccionada como primera opción por el 2.3% de la población encuestada (n=388); y fue la segunda carrera de mayor preferencia dentro de este campo profesional.

De acuerdo con los datos proporcionados por ANUIES, la relación que tiene la oferta de los programas educativos iguales o afines y la demanda vocacional para cursar el programa fue la siguiente: la UABC ofertó, en el ciclo escolar 2022-2023, 16 espacios para estudiar la licenciatura en Ciencias Computacionales, se registraron 29 solicitudes, e ingresaron 16 estudiantes (ANUIES, 2023); y la información correspondiente al estudio de preferencias vocacionales en Baja California, indicó que 388 estudiantes mostraron interés en cursar dicho programa (UABC, 2020). A nivel nacional, se observó que, entre el ciclo escolar 2018-2019 al 2022-2023, la matrícula creció un 33.04%; es decir, hubo un aumento de 2,338 estudiantes en cinco años.

Conclusiones de la oferta y demanda

- En México, la oferta de programas iguales a el programa de LCC se ubicó en ocho universidades públicas. En la zona noroeste únicamente se oferta en la Universidad Autónoma de Baja California y en la Universidad de Sonora (ANUIES, 2023).
- Las universidades que presentaron la mayor oferta y demanda del programa de LCC fueron la Universidad Autónoma de Nuevo León con 260 lugares disponibles y 211 inscritos en primer año; UNAM con 867 lugares y 173 aceptados; Universidad Autónoma de Hidalgo con 100 lugares ofertados mismos que se inscribieron; y la Benemérita Universidad Autónoma de Puebla con 150 lugares disponibles y 141 ingresos (ANUIES, 2023).
- Los programas afines ubicados fueron tres y se ofertaron en 22 universidades de México (50% públicas y 50% privadas) son los siguientes: la Ingeniería en Ciencias Computacionales, Ingeniería en Sistemas Computacionales, e Ingeniería en Sistemas Inteligentes (ANUIES, 2023).
- De estos programas afines, las instituciones que presentaron la mayor oferta y demanda fueron Universidad Autónoma de Chihuahua con 130 lugares y 95 aceptados; Universidad Autónoma de San Luis Potosí con 140 lugares disponibles y misma cantidad de inscritos; y Universidad de Guadalajara con 90 lugares ofertados e inscritos (ANUIES, 2023).
- La matrícula nacional del programa de LCC en el ciclo escolar 2022-2023 fue de 3,429 (80.99% hombres y 19.01% mujeres).

- Durante los últimos cinco años, la matrícula del campo detallado de formación en ciencias computacionales tuvo el mayor crecimiento durante los ciclos escolares 2021-2022 y el 2022-2023 (15.34%); y el menor crecimiento entre el 2019-2020 y el 2020-2021 (1.23%) [ANUIES, 2023].
- En términos generales, la matrícula del campo detallado de formación en ciencias computacionales aumentó un 33.04% durante cinco años.
- En España, de las 86 universidades registrada, 82 ofertaron la Ingeniería en Informática y en el año 2023 se ofertaron 9,467 lugares; y los créditos oscilaron entre 180 y 240 (Ministerio de Educación de España, 2023).
- En Argentina, de las 131 universidades existentes, el programa de LCC se ofertó en ocho universidades de sostenimiento público (Ministerio de Educación de Argentina, 2023).
- En Chile, existen 138 universidades, el programa de LCC se ofertó en dos de estas (Ministerio de Educación de Chile, 2023).
- En Colombia, se identificaron 29 universidades que ofertaron programas afines a el programa de LCC, de manera general, fue la Ingeniería en Sistemas Computacionales (Ministerio de Educación de Colombia, 2022).
- En Baja California, la demanda vocacional del campo profesional científico experimental ocupó el décimo lugar en las categorías de preferencia Muy alta y Alta (UABC, 2020).
- El programa de LCC fue seleccionada como primera opción por el 2.3% de la población encuestada (n=388), y la segunda de mayor preferencia dentro del campo científico experimental (UABC, 2020).
- Entre las características de la población estudiantil potencial a ingresar a este programa educativo, destaca lo siguiente: La edad media es de 17 años; el 43% proviene del COBACH; el 45% viven en Tijuana, el 33% en Mexicali, el 12% en Ensenada, 5% en Tecate y 5% en Rosarito; la modalidad de preferencia del 67.1% es la presencial (UABC, 2020).
- El 92.7% tiene como expectativa de estudios dar continuidad a su preparación profesional al concluir el bachillerato; el 69% se dedica a los estudios (UABC, 2020).

- En cuanto a la percepción de competencias tecnológicas, el 50% se sienten capaces de realizar tareas en las que usen las TIC (UABC, 2020).
- En la competencia lingüística, el 61% consideró tener capacidad para expresarse con claridad de manera oral; el 59% para comunicarse de manera escrita; al 33% le cuesta expresarse en inglés (UABC, 2020).
- En la competencia matemática, el 41% consideró tener capacidad para formular y resolver problemas aplicando diferentes enfoques (UABC, 2020).
- En la competencia socioafectiva, el 77% tiene capacidad para ponerse metas; el 66% puede solicitar apoyo y tomar decisiones que le generen bienestar; y el 12% puntualizó no sentirse capaz de cultivar relaciones sanas (UABC, 2020).
- En el programa de LCC de la UABC, en el ciclo escolar 2022-2023, se registraron 29 solicitudes, y se ofertaron y ocuparon 16 lugares.

Conclusiones Estudio 1.1 Evaluación de la pertinencia social

En el mundo habitan 8,000 millones de personas, en América Latina viven 662 millones y la esperanza de vida es de 72.2 años. El aprendizaje y la capacitación en línea de los adultos están en aumento, quienes trabajan prefieren cursos de desarrollo personal y los desempleados optan por cursos de análisis de datos, informática y manejo de las TIC. Los jóvenes prefieren un empleo que compagine con su vida personal, se preocupan por las finanzas diarias, la jubilación y el medio ambiente (Deloitte, 2022)

En México habitan 126,014,024 personas, el 51.2% son mujeres y el 48.8% hombres; la edad media es de 29 años. La población económicamente activa fue de 59.6 millones de personas en el año 2022, el ingreso promedio mensual oscila entre \$54,427^{oo} y los \$3,312^{oo} pesos (INEGI, 2020). El mayor porcentaje de la población mexicana en condiciones de pobreza se presentó en Chiapas, Guerrero, Puebla y Oaxaca; y el menor porcentaje de dio en Baja California, Nuevo León, Chihuahua y Coahuila (CONEVAL, 2020). Los jóvenes representaron el 24.6% de la población; el trabajo es la principal actividad; el recurso que más utilizan es el teléfono celular (76.7%) (INEGI, 2020).

El uso de la inteligencia artificial está en aumento en la vida cotidiana de las personas (OII, 2023). Sin embargo, el 50% de la población no tiene acceso a la misma, se

requiere crear contenido digital y aplicaciones acordes a las necesidades locales y la seguridad digital es un tema pendiente (Naciones Unidas, 2019). Algunas empresas aún se resisten al cambio digital, aunque están convencidas de la necesidad de transformarse digitalmente y requieren del desarrollo de la ciberseguridad (Dell Technologies, 2018).

Las Naciones Unidas promueven un crecimiento y desarrollo mundial más equitativo y sostenible, por lo que proponen 17 Objetivos de Desarrollo Sostenible hacia el 2030. Los egresados de la Licenciatura en Ciencias Computacionales podrían contribuir con los objetivos 4 “Educación de calidad”, objetivo 8 “Trabajo decente” y objetivo 9 “Industria, innovación e infraestructura”.

En el tema tecnológico en México actualmente se impulsa la Estrategia Nacional Digital 2019-2024, con la finalidad de lograr una transformación digital y tecnológica en México. Mientras a nivel estatal, en el Plan Estatal de Desarrollo de Baja California 2022-2027, se indica la necesidad de crear plataformas educativas digitales para capacitar a los docentes de educación básica; crear infraestructura y equipamiento para desarrollar tecnología de punta; lograr una mayor vinculación universidad-empresas en proyectos de innovación tecnológica (Gobierno de Baja California, 2023).

Con relación al mercado laboral, durante la pandemia por Covid-19 se transformaron las tareas, habilidades y los empleos. Los empresarios destacaron un grupo de habilidades que requieren mejorarse hacia el 2025: pensamiento crítico y análisis, resolución de problemas, autogestión y aprendizaje activo, resiliencia, tolerancia al estrés y flexibilidad (WEF, 2023). Entre las megatendencias a desarrollarse hacia el año 2030 se enlistan la transformación digital que impactará a nivel personal y empresarial; las empresas requerirán personal especializado que ayuda a las empresas a transformarse en un negocio digital (Dell Technologies, 2018). El mercado laboral es prometedor para profesionistas con conocimientos en inteligencia artificial, machine learning, Cloud computing y Blockchain (Hays, 2023). Los cargos altamente demandados serán: data analysts, software developers, cyber security experts, infraestructura architects, CRM specialists, automation and cloud experts, IT project managers (Hays, 2023).

En México, los profesionistas ocupados con licenciatura en ciencias computacionales en 2023 sumaron 747,005 (71.8% hombres y 28.2% mujeres) y ganaron

un promedio mensual de \$14,876^{oo} pesos (INEGI, 2023). Entre los empleos altamente demandados serán para especialistas en inteligencia artificial, analista de seguridad de la información, directores de proyectos y científicos de datos. Mientras las tendencias en contrataciones en el sector privado serán hacia las pequeñas y medianas empresas; y en el sector público serán por honorarios y temporales. Por su parte, las empresas de consultoría se colocan dentro de las 17 actividades más importantes en México. Entre las actividades realizadas por este tipo de empresas, se destacan las relacionadas con el programa de LCC: diseño de sistemas, y procesamiento electrónico de información, hospedaje y otros servicios. Asimismo, las certificaciones se acentúan como una tendencia de capacitación y actualización en el área de las TIC.

La muestra de empleadores encuestados en este estudio señaló lo siguiente, el 40% estuvieron de acuerdo en que la formación profesional de los egresados del programa de LCC corresponde con las exigencias actuales del mercado laboral. Sin embargo, las opiniones se dividen al considerar esta formación acorde con las demandas del mercado laboral futuro. Estos recomiendan aplicar los conocimientos en el ambiente laboral, evitar dividir el plan de estudios y enfocarse en una sola área, aplicar los conocimientos en proyectos que desarrollan habilidades de liderazgo, trabajo en equipo, cumplimiento de metas, ética, lógica, valores, responsabilidad; así como mejorar materias dedicadas al desarrollo de software.

En relación con el estudio de egresados, el 100% mencionaron que se encuentra trabajando; el 93.7% considera que ejerce su profesión, y el 87.5% se emplearon al finalizar los estudios de licenciatura. En lo referente al lugar en que trabajan señalaron que, el 93.8% forma parte de una empresa y el 6.3% trabaja de manera independiente; el 37.5% se desempeña en el sector de servicios profesionales y técnicos; el 87.5% demoró menos de un año para emplearse. El salario mensual que recibe el 62.5% de los egresados es mayor a \$20,000^{oo} pesos, ninguno percibe un ingreso mensual menor a \$8,000^{oo} pesos. Por lo que un porcentaje alto (93.8%) de los egresados consideraron que sus ingresos son suficientes para cubrir sus necesidades. En lo que concierne a su formación profesional, el 68.7% indicó que sus actividades laborales se alinean mucho con la misma. En lo que atañe a la formación en emprendimiento, el 82.2% consideró que no se fomentó durante sus estudios;

el 68.7% opinó que el perfil de emprendedor es relevante en esta carrera. Referente al grado de estudios para el empleo, el 56.3% consideró que la licenciatura es adecuada, mientras el 25% indicó que su empleo requiere el grado de maestría.

En el tema de oferta y demanda de programas iguales y afines a el programa de LCC en México se concluye lo siguiente, la oferta de programas iguales a el programa de LCC se ubicó en ocho universidades públicas. Las universidades que presentaron la mayor oferta y demanda del programa de LCC fueron la Universidad Autónoma de Nuevo León, UNAM, Universidad Autónoma de Hidalgo y la Benemérita Universidad Autónoma de Puebla (ANUIES, 2023). En lo que corresponde a los programas afines, por nombre se ubicaron tres (Ingeniería en Ciencias Computacionales, Ingeniería en Sistemas Computacionales, e Ingeniería en Sistemas Inteligentes) y se ofertaron en 22 universidades de México (ANUIES, 2023).

La matrícula nacional del programa de LCC en el ciclo escolar 2022-2023 fue de 3,429 (80.99% hombres y 19.01% mujeres). Durante los últimos cinco años, la matrícula de todo el campo detallado de formación en Ciencias Computacionales tuvo el mayor crecimiento durante los ciclos escolares 2021-2022 y el 2022-2023 (15.34%); y el menor crecimiento entre el 2019-2020 y el 2020-2021 (1.23%) [ANUIES, 2023]. En términos generales, la matrícula del programa de LCC aumentó un 33.04% durante cinco años.

En el tema internacional, la oferta y demanda se observó tal como se describe a continuación. En España, de las 86 universidades registrada, 82 ofertaron la Ingeniería en Informática y en el año 2023 se ofertaron 9,467 lugares; y los créditos oscilaron entre 180 y 240 (Ministerio de Educación de España, 2023). En lo que corresponde a Argentina, de las 131 universidades existentes, el programa de LCC se ofertó en ocho universidades de sostenimiento público (Ministerio de Educación de Argentina, 2023). En Chile, existen 138 universidades, el programa de LCC se ofertó en dos de estas (Ministerio de Educación de Chile, 2023). Por último, en Colombia, se identificaron 29 universidades que ofertaron programas afines a el programa de LCC, de manera general, fue la Ingeniería en Sistemas Computacionales (Ministerio de Educación de Colombia, 2022).

La demanda vocacional del campo profesional científico experimental en Baja California ocupó el décimo lugar en las categorías de preferencia Muy alta y Alta. El

programa de LCC fue seleccionada como primera opción por el 2.3% de la población encuestada (n=388), y la segunda de mayor preferencia dentro del campo científico experimental. Entre las características de la población estudiantil potencial a ingresar a este programa educativo, destaca lo siguiente: La edad media es de 17 años; el 43% proviene del COBACH; el 45% viven en Tijuana, el 33% en Mexicali, el 12% en Ensenada, 5% en Tecate y 5% en Rosarito; la modalidad de preferencia del 67.1% es la presencial. En cuanto a la percepción de competencias tecnológicas, el 50% se sienten capaces de realizar tareas en las que usen las TIC. En la competencia lingüística, el 61% consideró tener capacidad para expresarse con claridad de manera oral; el 59% para comunicarse de manera escrita; al 33% le cuesta expresarse en inglés. En la competencia matemática, el 41% consideró tener capacidad para formular y resolver problemas aplicando diferentes enfoques. En la competencia socioafectiva, el 77% tiene capacidad para plantearse metas; el 66% puede solicitar apoyo y tomar decisiones que le generen bienestar; y el 12% puntualizó no sentirse capaz de cultivar relaciones sanas (UABC, 2020).

Estudio 1.2 Evaluación de referentes

El objetivo general fue analizar los referentes estratégicos donde se inscribe la profesión del programa educativo, los programas iguales o afines y los referentes nacionales e internacionales que permiten fundamentar la creación, modificación o actualización del programa. Para tal fin, se abordaron tres análisis: 1.2.1 Análisis de la profesión y su prospectiva, 1.2.2 Análisis comparativo de programas educativos y 1.2.3 Análisis de organismos nacionales e internacionales.

1.2.1 Análisis de la profesión y su prospectiva

Objetivos

El objetivo general y los objetivos específicos de este análisis se muestran en la Tabla 1.24.

Tabla 1.24

Objetivos del Análisis de la profesión y su prospectiva

Objetivo general	Objetivo específico
Analizar la profesión, su evolución y sus campos de acción a nivel	Describir el entorno de la profesión del programa educativo.
	Identificar las prácticas de la profesión.

Objetivo general	Objetivo específico
nacional e internacional con el fin de fundamentar la creación, modificación o actualización del programa educativo.	Analizar las profesiones afines con las que comparte su ejercicio y práctica profesional. Analizar la evolución y prospectiva de la profesión en el contexto nacional e internacional

Nota. De “Metodología de los estudios de fundamentación para la creación, modificación y actualización de programas educativos de licenciatura”, por A. Serna, y A. Castro, 2018, Universidad Autónoma de Baja California. CC BY-NC-ND.

Metodología

Se realizó una investigación documental, con un diseño no experimental. Las fuentes de información utilizadas fueron: las Bases de datos de Scopus, Ebsco EJS, y Google académico; las páginas de organismos nacionales e internacionales como UNESCO, OCDE, World Economic Forum; IEEE Computer Society; Association for Computing Machinery; INEGI; CONAIC; Academia Mexicana de Computación; ANUIES; CINVESTAV; el libro de Hernández-Laos, et al., 2012, que abarca el tema del Mercado laboral de profesionistas en México; así como el libro de Oropeza-García, 2019 referente al Desarrollo industrial hacia el año 2050.

Resultados

Entorno de la profesión

El entorno en que se desarrolla el programa de LCC está fundamentado en un mundo globalizado, interconectado y cambiante. De esta manera, se precisa que los avances tecnológicos hacia el 2030 generarán el mayor progreso tecnológico sin precedentes en los últimos 100 años. De acuerdo con McKinsey & Company, las diez principales megatendencias tecnologías para los próximos 10 años serán las siguientes (WEF, 2021):

1. *Automatización y virtualización de procesos* tendrá impacto en la reducción de procesos y costos en el sector industrial.
2. *Conectividad del futuro* será proporcionada a través de la quinta generación de tecnología celular inalámbrica o 5G que beneficiará la economía y la vida de la mayoría de las personas, a través de ciudades, empresas y casas inteligentes.
3. *Infraestructura distribuida* permitirá a las empresas que el acceso y manejo de datos sean desde la nube, disminuyendo costos, proporcionando velocidad en los procesos y altos niveles de seguridad.

4. *Computación de próxima generación* beneficiará el desempeño de las empresas de finanzas, viajes, logística, energía, e industria avanzada.
5. *Inteligencia artificial aplicada* mejorará más del 75% de los servicios digitales de contacto, como los asistentes de voz.
6. *Futuro de la programación* marcará tendencia mediante la aplicación del Software 2.0.
7. *Arquitectura de confianza* ayudará a las empresas a mejorar todos los procesos relacionados con la ciberseguridad.
8. *Biorevolución*, los avances en la ciencia biológica impactarán la vida de las personas a través del desarrollo de terapias genéticas, medicina hiperpersonalizada y estudios genéticos; de igual forma afectará la agricultura, la energía, los materiales y los bienes de consumo.
9. *Materiales de última generación*, alcanzarán una mayor eficiencia en el sector farmacéutico, energético, transporte, manufactura entre otros, por ejemplo, materiales como el grafeno y el disulfuro de molibdeno prometen revolucionar la industria electrónica.
10. *Tecnologías limpias*, existe una tendencia en los países y las empresas por alcanzar un desarrollo sostenible, por lo que se busca un consumo de agua y energía renovable, así como un sistema de transporte ecológico (WEF, 2021).

De esta manera, se señala que las profesiones inmersas en el área de las ciencias computacionales se caracterizan por la rápida innovación, la colaboración internacional, la movilidad laboral y la necesidad de adaptar y crear cuestiones éticas y sociales a escala mundial.

Aunado a lo anterior, la pandemia por covid-19 marcó una situación sin precedentes en la vida de las personas, el trabajo, las cadenas de suministro y la imprevisibilidad de las operaciones y los mercados. Por lo que se espera que la tecnología continúe con un papel central en futuras pandemias, guerras y desastres naturales. En este sentido, se generaron 19 predicciones tecnológicas para los próximos 15 años (IEEE, 2023):

1. Atención médica remota y dispositivos portátiles.
2. Realidad aumentada.

3. Software para Edge2Cloud Continuum.
4. Hardware abierto.
5. DevOps asistido por IA.
6. Impresión 3D en atención sanitaria personalizada.
7. Inteligencia artificial generativa.
8. Tecnología para la sostenibilidad.
9. Vehículos autónomos.
10. Fabricación distribuida digital, ayudará a reducir la huella energética y medioambiental y beneficiará la resiliencia en las cadenas de suministro.
11. Computación confiable, los proveedores de servicios confiables serán líderes del mercado.
12. Redes neurales gráficas amplias, aumentarán su uso en el aprendizaje automático.
13. Productos farmacéuticos adaptativos y generativos se combinarán áreas como la nanotecnología y la creación de vacunas de manera eficiente.
14. Robots autónomos, adopción generalizada de plataformas robóticas.
15. Inteligencia general artificial, sistemas que podrán aprender y desaprender tareas intelectuales que realiza el ser humano.
16. Digitalización global de las transacciones monetarias.
17. TIC espacial.
18. Fabricación espacial sostenible.
19. Detección/corrección de la desinformación.

En términos de avance del desarrollo tecnológico en 2023 e impacto para la humanidad, la mayor predicción de éxito a corto plazo la tendrá la atención médica remota y los dispositivos portátiles; mientras las oportunidades a largo plazo se señalan en las tecnologías espaciales y la digitalización global de las transacciones monetarias.

Las oportunidades de desarrollo se distinguen por áreas. En el caso de la industria serán la realidad aumentada, la industria de la salud, la tecnología para la sustentabilidad y el hardware abierto. Los gobiernos tendrán la oportunidad de trabajar en la detección de la desinformación y los productos farmacéuticos generativos. Mientras para la academia se marca todo lo relacionado con el desarrollo de tecnologías espaciales. Por último, las

organizaciones profesionales podrán focalizarse en la inteligencia artificial sostenible y la tecnología para el espacio (IEEE, 2023).

Prácticas de la profesión

A nivel nacional, el Sistema Nacional de Clasificación de Ocupaciones (SINCO) 2019, es el documento que agrupa y describe las ocupaciones que se realizan en México. De esta manera, las actividades que realizan los investigadores y profesionistas del área de sistemas computacionales se clasifican en dos grupos: (a) Desarrolladores y analistas de software y multimedia, y (b) Administradores de bases de datos y redes de computadora (INEGI, 2019).

Los Desarrolladores y analistas de software y multimedia, son los profesionistas que “desarrollan, crean y modifican programas para aplicaciones informáticas generales o programas utilitarios especializados; investigan, diseñan, desarrollan y programan sitios de Internet e Intranet” (p. 128). Pertenecen a este subgrupo las siguientes ocupaciones: Analista de sistemas y programador de software; Asesor y consultor informático; Diseñador de páginas web, Diseñador de programas de computación, Ingeniero en sistemas y desarrollador de software.

Por su parte, los Administradores de bases de datos y redes de computadora son los profesionistas realizan las siguientes actividades:

[...] diseñan, desarrollan, y administran soluciones de gestión de datos utilizando un software de gestión de base de datos; coordinan el mantenimiento y actualización de los sistemas y redes informáticas, prueban e implantan bases de datos aplicando los conocimientos de los sistemas de gestión de bases de datos; desarrollan y aplican políticas, normas y modelos para la administración de bases de datos. (p. 129)

Las ocupaciones que corresponden a este subgrupo son: Administrador de bases de datos, Administrador de redes de computación, e Ingeniero administrador de sistemas de cómputo (INEGI, 2019).

Análisis de las profesiones afines con las que comparte su ejercicio y práctica profesional

De acuerdo con el Consejo Nacional de Acreditación en Informática y Computación A.C. (CONAIC), las profesiones que tienen relación con la computación son cuatro: Licenciatura en informática, Ingeniería de software, la Ingeniería en ciencias computacionales y Licenciatura en ciencias computacionales (CONAIC, 2022):

- La licenciatura en informática, tiene un perfil profesional, con amplias posibilidades de dar continuidad a estudios de posgrado. Es definida como el profesional con conocimientos en el área de las TIC aplicadas al proceso administrativo de las organizaciones.
- Ingeniería de Software, también es considerada como un perfil profesional con oportunidad de continuar a estudios de posgrado. Es el especialista en la producción de sistemas de software de calidad; formula, planea, implementa y da mantenimiento a los sistemas de información; tiene una formación en técnicas de análisis, diseño de sistemas de información, manejo de herramientas de programación e ingeniería de software.
- Ingeniería computacional, es el profesional encargado de construir, configurar, evaluar y seleccionar obras y entornos de servicios computacionales; genera nueva tecnología, implementa soluciones eficientes de cómputo en las organizaciones; domina los principios teóricos, prácticos y metodológicos que dan sustento al diseño y desarrollo de sistemas complejos, a la arquitectura del hardware y la configuración de redes de cómputo.
- Licenciatura en ciencias computacionales, tiene un perfil académico con capacidad para estudios de posgrado. Es definida como el profesional que estudia y desarrolla las ciencias computacionales; conceptualiza y desarrolla ambientes, facilidades y aplicaciones innovadoras de computación en diversos entornos; elabora modelos de realidades complejas con rigor teórico y práctico.

Evolución de la profesión

Las universidades tienen el objetivo de formar personas con las habilidades, conocimientos y actitudes necesarios para promover el desarrollo social y económico a largo plazo en cada país (Ferreyra, et al., 2017), con responsabilidad social (Vallaey, 2014). Este objetivo es conseguido a través de tres funciones principales de las universidades como son la

investigación, la enseñanza y la difusión de la cultura. También, tienen la responsabilidad de “formar a los profesionistas y especialistas requeridos por los sectores de la producción y los servicios en todos los ámbitos del saber” y que les permita insertarse en el mercado laboral (Hernández-Laos et al., 2012, p. 110).

En México, el Instituto Politécnico Nacional (IPN) ofreció, desde 1965, las carreras de técnico en computación electrónica y técnico en mantenimiento de equipos de computación y electrónica. También el IPN impartió, en el mismo año, la maestría en ciencias con especialidad en computación. Mientras la primera licenciatura en computación la impartió el Instituto Tecnológico y de Estudios Superiores de Monterrey en el año 1968, denominada Ingeniería en Sistemas Computacionales (CINVESTAV, s.f.).

Los datos más recientes ofrecen información respecto a la evolución de la oferta y demanda de los profesionistas en este país durante el periodo 2001-2009 y una prospectiva hacia el 2020. La indagación permitió destacar que, en dicho periodo de análisis, egresaron del nivel superior 3,705,243 estudiantes, la licenciatura representó el 81.5% del total. En el área de Ingenierías, se observó un crecimiento de egresados del 36%; mientras el área la artística obtuvo el mayor crecimiento en este periodo con el 196.7%. El mayor incremento en los egresados de las Ingenierías se obtuvo en la rama de metalurgia y extractiva con un 132%, mientras en computación e informática fue del 17%. La evolución de los egresados de esta área se encuentra estrechamente relacionada con la dinámica de crecimiento del sector industrial. Por lo que, su bajo crecimiento fue impactado por las crisis del 2001 y 2008 procedentes de Estados Unidos (Hernández-Laos et al., 2012).

En lo referente al tema de profesionistas ocupados con nivel de licenciatura, durante dos periodos analizados 2000 y 2009 se observó que, ocho de cada diez profesionistas formaron parte de la población económicamente activa, y comenzó una mayor participación de mujeres profesionales en el mercado laboral. El crecimiento en el empleo se obtuvo en cinco áreas del conocimiento, las Ingenierías ocuparon el segundo lugar con el 17.3% en 2000 y el 17.7% en 2009; las carreras que destacaron fueron las Ingeniería mecánica, Ingeniería industrial, Ingeniería en computación e informática. Se advierte que, en esta década a pesar del crecimiento en el empleo de profesionistas, aún no es equiparable con los países exitosos en las exportaciones de manufactura y de servicios de alta tecnología,

quienes insertan a sus egresados en estos sectores especializados (Hernández-Laos et al., 2012).

La Ingeniería en computación e informática ocupó el quinto lugar en 2009 entre 16 carreras (de 45) que aglutinaron al 65% de los egresados de licenciatura. En un escenario analizado para la década del 2010 al 2020, el balance entre oferta y demanda se observó favorable para esta profesión, situación que no lograron el resto de las carreras de la misma área. Por lo que se previó un equilibrio entre la oferta-demanda entre el 2009 y 2015. Sin embargo, se pronosticó hacia el 2020, una demanda superior a la oferta de estos profesionistas, aunque no sería para empleos de calidad (Hernández-Laos et al., 2012).

Actualmente, en el ciclo escolar 2022-2023, la licenciatura en ciencias computacionales se ofrece en ocho universidades públicas y registró una matrícula de 3,429 estudiantes, donde el 21.78% fueron de nuevo ingreso. Los programas afines, como la Ingeniería en ciencias computacionales o la Ingeniería en computación, se imparten en 21 universidades públicas y privadas (ANUIES, 2023).

Prospectiva de la profesión a nivel nacional e internacional.

Las megatendencias tecnológicas pronosticadas hacia el año 2030 mantienen un avance inédito, comparado con el desarrollo realizado durante la época de la Revolución industrial, que nos encamina a la llamada Revolución digital, cuyo impacto directo se observará en la vida cotidiana de las personas, en la industria, el medio ambiente y los gobiernos. Aunque se desconoce el resultado futuro de estas megatendencias, la transformación de la industria del siglo XXI presentará una evolución tecnológica continua (Oropeza-García, 2019).

Debido a que México sobresale como la quinceava economía a nivel mundial, se pronostica como una próxima economía emergente. La rápida evolución de la industria tecnológica abre las oportunidades de desarrollo industrial para nuestro país, y se espera la colaboración del sector industrial, el gobierno y la academia (Buenrostro, 2019).

Entre las áreas concebidas para cada uno de los sectores se menciona que compete a la industria trabajar con la realidad aumentada; a la industria de la salud, la tecnología para la sustentabilidad y el hardware abierto; a los gobiernos la detección de la desinformación y los productos farmacéuticos generativos; y a la academia lo relacionado con la inteligencia artificial sostenible y la tecnología para el espacio (IEEE, 2023).

Las megatendencias y la tecnología impulsarán la generación de nuevos empleos en todas las categorías laborales e industrias, aunque se espera que otros desaparezcan, principalmente aquellos de servicio cara a cara. Entre los empleos que crecerán un 30-35% hacia el 2027 serán los analistas y científicos de datos, especialistas en *Big Data*, analistas en inteligencia empresarial, profesionales de bases de datos y redes e Ingenieros de datos; todos estos impulsados por los avances y el crecimiento en la adopción de tecnologías de vanguardia que dependen de Big Data. Las industrias que esperan un alto crecimiento en estos roles incluyen los de servicios financieros (31%), venta minorista y mayorista de bienes de consumo (37%), y cadena de suministro y transporte (42%); la expectativa es medida para la industria de tecnología de la información y comunicaciones digitales con el 8% (WEF, 2020).

De frente a estos nuevos empleos, los empresarios señalan nuevas habilidades en el trabajo. En opinión del 40% de los empresarios encuestados por el Foro Económico Mundial, las siguientes habilidades tendrán importancia durante el 2023: (a) cognitivas, como el pensamiento analítico y el pensamiento creativo; (b) autoeficacia, entre las que destacan la resiliencia, flexibilidad y agilidad, motivación y conciencia de sí mismo, curiosidad y aprendizaje a lo largo de la vida, confiabilidad y atención al detalle; (c) tecnológicas, como la literacidad tecnológica; (d) trabajo con otros, empatía y escucha activa, liderazgo e influencia social; y (e) habilidades de gestión, como el control de calidad. Hacia el 2027, las habilidades que las empresas esperan aumenten, dada su importancia en empleo seguirán siendo las cognitivas como el pensamiento creativo (73.2%) y el pensamiento analítico (71.6%), así como la literacidad tecnológica (67.7%) (WEF, 2020).

Asimismo, dados los cambios vertiginosos en el desarrollo de tecnología, los empresarios señalan la necesidad de mantener un aprendizaje continuo o a lo largo de la vida (WEF, 2020). En promedio, en los países de la OCDE, el 14% de los adultos ha participado en educación y capacitación formales o no formales, hasta ahora los cursos de desarrollo personal son preferidos entre quienes tienen empleo, mientras los desempleados muestran interés en mejorar sus habilidades digitales, por lo que toman cursos de análisis de datos, la informática y manejo de la tecnología de la información y la comunicación (OCDE, 2022).

Ante este avance tecnológico y otros sucesos económicos y sociales, las Naciones Unidas invitan a los países a mantener un crecimiento sostenible, por lo que resulta importante que los universitarios tengan presente los 17 Objetivos de Desarrollo Sostenible (ODS) y los pongan en práctica en cada una de sus profesiones. La finalidad de los ODS, es combatir la pobreza, disminuir las desigualdades dentro y entre países, y combatir el cambio climático. En general, se trata de vivir en este planeta con respeto, para que las próximas generaciones lo reciban en condiciones favorables (Naciones Unidas, 2015).

Conclusiones del Análisis de la profesión y su prospectiva

Entorno de la profesión

- Los avances tecnológicos hacia el 2030 generarán el mayor progreso tecnológico sin precedentes en los últimos 100 años.
- Las profesiones inmersas en el área de las ciencias computacionales se caracterizan por la rápida innovación, la colaboración internacional, la movilidad laboral y la necesidad de adaptar y crear cuestiones éticas y sociales a escala mundial.
- A partir de la pandemia por Covid-19 se espera que la tecnología continúe con un papel central en futuras pandemias, guerras y desastres naturales. En este sentido, se generaron 19 predicciones tecnológicas para los próximos 15 años (IEEE, 2023).
- En términos de avance del desarrollo tecnológico en 2023 e impacto para la humanidad, la mayor predicción de éxito a corto plazo la tendrá la atención médica remota y los dispositivos portátiles; mientras las oportunidades a largo plazo se señalan en las tecnologías espaciales y la digitalización global de las transacciones monetarias.
- Las oportunidades de desarrollo se distinguen por áreas: (a) industria, con la realidad aumentada, la salud, la tecnología para la sustentabilidad y el hardware abierto; (b) gobiernos, detección de la desinformación y los productos farmacéuticos generativos; (c) academia, todo lo relacionado con el desarrollo de tecnologías espaciales; (d) organizaciones profesionales, podrán focalizarse en la inteligencia artificial sostenible y la tecnología para el espacio (IEEE, 2023).

Prácticas de la profesión

- Las actividades que realizan los investigadores y profesionistas del área de sistemas computacionales se clasifican en dos grupos: (a) Desarrolladores y analistas de software y multimedia, y (b) Administradores de bases de datos y redes de computadora (INEGI, 2019).

Profesiones afines con las que comparte su ejercicio y práctica profesional

- Las profesiones que tienen relación con la computación son cuatro: Licenciatura en informática, Ingeniería de software, la Ingeniería en ciencias computacionales y Licenciatura en ciencias computacionales (CONAIC, 2022).

Evolución de la profesión

- En México, el Instituto Politécnico Nacional (IPN) ofreció, desde 1965, las carreras de técnico en computación electrónica y técnico en mantenimiento de equipos de computación y electrónica, y la maestría en ciencias con especialidad en computación. La primera licenciatura en computación la impartió el Instituto Tecnológico y de Estudios Superiores de Monterrey en el año 1968, denominada Ingeniería en Sistemas Computacionales (CINVESTAV, s.f.).
- Durante el periodo 2001-2009, egresaron del nivel superior 3,705,243 estudiantes, la licenciatura representó el 81.5% del total. En el área de Ingenierías, se observó un crecimiento de egresados del 36%; el mayor incremento correspondió a la rama de metalurgia y extractiva con un 132%, mientras en computación e informática fue del 17%. La evolución de los egresados de esta área se encuentra estrechamente relacionada con la dinámica de crecimiento del sector industrial (Hernández-Laos et al., 2012).
- Durante el mismo periodo, el crecimiento en el empleo se obtuvo en cinco áreas del conocimiento, las Ingenierías ocuparon el segundo lugar con el 17.3% en 2000 y el 17.7% en 2009; las carreras que destacaron fueron las Ingeniería mecánica, Ingeniería industrial, Ingeniería en computación e informática (Hernández-Laos et al., 2012).
- El porcentaje de egresados de la Ingeniería en computación e informática ocupó el quinto lugar en 2009 entre 16 carreras (de 45) que aglutinaron al 65% de los egresados de licenciatura.

- El pronóstico hacia el 2020 que habría una demanda superior a la oferta de estos profesionistas, aunque no sería para empleos de calidad (Hernández-Laos et al., 2012).
- Actualmente, en el ciclo escolar 2022-2023, la licenciatura en ciencias computacionales se ofrece en ocho universidades públicas y registró una matrícula de 3,429 estudiantes, donde el 21.78% fueron de nuevo ingreso. Los programas afines, como la Ingeniería en ciencias computacionales o la Ingeniería en computación, se imparten en 21 universidades públicas y privadas (ANUIES, 2023).

Prospectiva de la profesión

- Nos encamina a la llamada Revolución digital, con impacto directo en la vida cotidiana de las personas, en la industria, el medio ambiente y los gobiernos. La transformación de la industria del siglo XXI presentará una evolución tecnológica continua (Oropeza-García, 2019).
- México sobresale como la quinceava economía mundial, y se pronostica como una próxima economía emergente (Buenrostro, 2019).
- La rápida evolución de la industria tecnológica abre las oportunidades de desarrollo industrial para nuestro país, y se espera la colaboración entre la industrial, el gobierno y la academia (Buenrostro, 2019).
- Los empleos que crecerán un 30-35% hacia el 2027 serán los analistas y científicos de datos, especialistas en Big Data, analistas en inteligencia empresarial, profesionales de bases de datos y redes e Ingenieros de datos (WEF, 2020).
- La expectativa de crecimiento es medida para la industria de tecnología de la información y comunicaciones digitales con el 8% (WEF, 2020).
- El tipo de habilidades para el trabajo necesarias en 2023 serán: (a) cognitivas, pensamiento analítico y el pensamiento creativo; (b) autoeficacia, resiliencia, flexibilidad y agilidad, motivación y conciencia de sí mismo, curiosidad y aprendizaje a lo largo de la vida, confiabilidad y atención al detalle; (c) tecnológicas, literacidad tecnológica; (d) trabajo con otros, empatía y escucha activa, liderazgo e influencia social; y (d) habilidades de gestión, como el control de calidad (WEF, 2020).

- Hacia el 2027, las habilidades que las empresas esperan aumenten, dada su importancia en empleo seguirán siendo las cognitivas como el pensamiento creativo (73.2%) y el pensamiento analítico (71.6%), así como la literacidad tecnológica (67.7%) (WEF, 2020).
- El 14% de los adultos ha participado en educación y capacitación formales o no formales, los cursos de desarrollo personal son preferidos entre quienes tienen empleo; quienes no tienen empleo prefieren cursos de análisis de datos, informática y manejo de la tecnología de la información y la comunicación (OCDE, 2022).
- Es importante que los universitarios conozcan y den seguimiento a los ODS desarrollados por las Naciones Unidas para vivir en este planeta con respeto, y entregarlo a las próximas generaciones en condiciones favorables (Naciones Unidas, 2015).

1.2.2 Análisis comparativo de programas educativos

Objetivos

El objetivo general del presente análisis, junto con los objetivos específicos se enuncian en la Tabla 1.25.

Tabla 1.25

Objetivos del Análisis comparativo de programas educativos

Objetivo general	Objetivo específico
Realizar un análisis comparativo para identificar las mejores prácticas en los programas educativos nacionales e internacionales de acuerdo con criterios de calidad, trascendencia y reconocimiento.	Comparar las prácticas, características y estrategias desarrolladas por instituciones educativas en los mejores programas educativos nacionales e internacionales reconocidos por organismos acreditadores o por rankings nacionales y/o internacionales.
	Determinar las prácticas y estrategias destacables de los mejores programas educativos nacionales e internacionales para ser considerados en la creación, modificación o actualización del programa educativo.

Nota. De “Metodología de los estudios de fundamentación para la creación, modificación y actualización de programas educativos de licenciatura”, por A. Serna, y A. Castro, 2018, Universidad Autónoma de Baja California. CC BY-NC-ND.

Metodología

El tipo de diseño e investigación corresponde con una investigación comparada y documental (Serna y Castro, 2018). Por lo tanto, en este apartado se describen los mejores diez programas de licenciatura en Ciencias Computacionales, a nivel nacional e internacional. En este sentido, se muestran las características principales (objetivos, duración del programa, créditos, perfil de egreso, y campo laboral) así como la estructura del plan de estudios y las mejores prácticas identificadas en las mejores instituciones de educación superior. En un primer bloque se muestra la información referente a las siguientes universidades mexicanas: Universidad de Sonora (UNISON), Benemérita Universidad Autónoma de Puebla (BUAP), Universidad Autónoma de Yucatán, Universidad Nacional Autónoma de México (UNAM) y Universidad Autónoma de Guadalajara. En un Segundo bloque se agrupa la información respectiva de las siguientes universidades internacionales: Massachusetts Institute of Technology (MIT), Stanford University, Carnegie Mellon University, University of California-Berkeley, University of Oxford.

Resultados

En el tema de la duración de la licenciatura en Ciencias Computacionales se identificaron tres tipos: 8 semestres (UNISON y UNAM), 9 semestres (Universidad Autónoma de Yucatán), y 10 semestres (BUAP y Universidad Autónoma de Nuevo León).

En cuanto a los créditos solicitados para completar el plan de estudios, se observó que la UNISON y la UNAM requieren 383 y 376 créditos respectivamente. La Universidad Autónoma de Yucatán solicita 360. Mientras la BUAP y la Universidad de Nuevo León precisan 293 y 226 créditos cada una.

Los conocimientos, capacidades y habilidades que las instituciones de educación superior a las cuales se comprometen las instituciones que ofertan esta licenciatura, son los siguientes: Los egresados adquirirán conocimientos teóricos de las ciencias de la computación, fundamentos matemáticos, lenguajes de programación, así como capacidad para desarrollar la creatividad e innovación. En cuanto a las actitudes y valores, se comprometen a formar a los estudiantes con apego ético, liderazgo, trabajo en equipo,

calidad, y responsabilidad social. La BUAP, la Universidad Autónoma de Yucatán, y la UNAM se interesan en la aplicación de conocimientos sobre el diseño e implementación de software y hardware eficientes. Además, la UNAM perfila a sus estudiantes para aprender sobre la inteligencia artificial. La Universidad Autónoma de Nuevo León y la UNAM promueven habilidades para solucionar problemas con rigor científico y realizar propuestas académicas con apego a estándares internacionales. De manera particular, la Universidad de Guadalajara forma Ingenieros en Computación, por lo cual promueve los conocimientos en algoritmos de computación, arquitectura de computadoras y redes de paradigmas modernos de cómputo y tecnologías emergentes.

En el campo laboral se especifica la Consultoría e investigación en empresas públicas y privadas, el sector empresarial en el desarrollo de hardware y software, así como la Docencia. En la BUAP adicionalmente, se puntualiza el desarrollo de aplicaciones web y videojuegos. En la UNAM, además, se capacita a los estudiantes para dar continuidad a los estudios de posgrado. Mientras en la Universidad de Nuevo León, los egresados se pueden desempeñar también como analistas de datos. En la tabla X se presenta la información completa.

Tabla 1.26

Programas Educativos Nacionales iguales a LCC

Programa	Objetivos	Campo laboral	Duración	Perfil de egreso	Estructura
Licenciatura en Ciencias de la Computación. <i>Universidad de Sonora</i>	Formar egresados con entrenamiento en análisis y abstracción, habilidades que le permitirán generar nuevos conocimientos, métodos de trabajo, y herramientas.	Carrera académica. Ejerciendo de forma independiente su profesión Desarrollo de hardware o software en diversas empresas.	8 semestres	Conocimientos: Lenguajes de programación, Análisis y diseño de sistemas computacionales. Fundamentos matemáticos. Conocimientos matemáticos para modelar y simular fenómenos de diferente naturaleza. Fundamentos teóricos de las Ciencias de la Computación. Habilidades: Divulgación de sus conocimientos de manera oral y escrita. Trabajo interdisciplinario y/o en equipo. Capacidad de abstracción, para diseñar sistemas computacionales a nivel de hardware y software. Capacidad para adquirir nuevos conocimientos, adaptarse a los cambios de la computación. Actitudes: Emprendimiento y visión. Liderazgo. Autoformación. Compromiso con la sociedad.	El total de créditos 383, distribuidos en 5 ejes: Básico (159), Profesionalizante (128), Especializante (40), Integrador (40), Común (16).
Licenciatura en Ciencias de la	Formar de manera integral Licenciados en Ciencias de la	Investigación en instituciones públicas	10 semestres	Conocimientos: Fundamentos Matemáticos	Créditos: 293 máximo y 266 mínimo.

Programa	Objetivos	Campo laboral	Duración	Perfil de egreso	Estructura
Computación. <i>Benemérita Universidad Autónoma de Puebla</i>	Computación que realicen análisis, diseño e implementación de sistemas de software eficientes, con la finalidad de generar soluciones innovadoras a problemas del entorno, aplicando los conocimientos de la disciplina en los ámbitos nacional e internacional, con sentido ético y compromiso social.	o privadas. Docencia. Administración de proyectos de software. Jefe de Departamento de Sistemas. Consultoría a empresas. Desarrollo de sistemas inteligentes. Diseño e implementación de sistemas de software. Desarrollo de aplicaciones web y/o videojuegos.		de Ciencias de la Computación. Conceptos principales de las teorías de la Ciencia de la Computación. Sistemas inteligentes. Desarrollo de investigación. Paradigmas de programación. Innovaciones en las formas de cómputo centrado en la red. Metodologías de Ingeniería de Software. Habilidades: Análisis y generación de modelos matemáticos. Interacción con usuarios y especialistas. Razonamiento computacional. Desarrollo y aplicación de metodologías (análisis, diseño e implementación de sistemas de cómputo). Comunicación de ideas, tanto en lengua materna como en lengua extranjera. Actitudes y Valores: Actitud positiva ante los cambios científicos y tecnológicos. Actualización disciplinar permanente. Trabajo en equipo. Liderazgo. Responsabilidad.	Dividido en: Currículo correlacionado: nivel básico (106) y formativo (127), y asignaturas complementarias (33). Currículo transversal.

Programa	Objetivos	Campo laboral	Duración	Perfil de egreso	Estructura
				Solidaridad. Compromiso social y con el medio ambiente. Principios éticos.	
Licenciatura en Ciencias de la Computación. <i>Universidad Autónoma de Yucatán.</i>	Formar profesionales en el área de ciencias de la computación capaces de desarrollar y gestionar tecnología computacional, realizar investigación, desarrollar software tanto de base como de aplicación, con ética profesional y servicio a la sociedad.	Automatización de procesos y asistencia técnica especializada (Instituciones Públicas o Empresas). Docencia. Compañías especializadas en desarrollo de software. Consultoría en computación y desarrollo de aplicaciones informáticas. Centros de comercialización de bienes informáticos.	9 semestres	Competencias de egreso: Desarrollo de software de aplicación. Desarrollo de software de base. Gestión tecnológica. Computación Científica. Competencias disciplinares: Interpretación adecuada de tablas, gráficas, diagramas y textos con lenguaje matemático. Construcción de modelos matemáticos. Diseño de algoritmos computacionales complejos. Solución de problemas matemáticos y computacionales.	360 créditos: Asignaturas obligatorias (268) Prácticas profesionales (8) Servicio social (12) Optativas (54) Libres (18)
Licenciatura en Ciencias de la Computación. <i>UNAM</i>	Prepara a los estudiantes para: Analizar, diseñar y construir sistemas de software complejos. Resolver problemas de seguridad, almacenamiento y envío de información. Estudiar áreas como la inteligencia artificial.	Departamentos de sistemas y desarrollo de empresas o instituciones Consultoría Docencia Estudios de posgrado	8 semestres	Habilidades: Solucionar problemas con rigor y efectividad Capacidad de análisis y síntesis, observación y deducción Creatividad e inventiva Conocimientos: Teóricos para analizar, diseñar e implementar aplicaciones Matemáticos Organización de la	Total 376 créditos: Asignaturas obligatorias (292) Optativas (60) Eje de inglés (24)

Programa	Objetivos	Campo laboral	Duración	Perfil de egreso	Estructura
				información Actitudes: Respeto a la propiedad intelectual Ética en su actividad profesional	
Licenciatura en Ciencias de la Computación. <i>Universidad Autónoma de Nuevo León.</i>	Formar profesionales en ciencias de la computación con capacidad para: Diseñar soluciones computacionales para empresas del sector público y privado, mediante razonamiento crítico, ética profesional, responsabilidad social y trabajo multidisciplinario. Analizar, diseñar, evaluar, implementar, e innovar tecnologías computacionales en diversos ámbitos. Brindar soporte técnico especializado. Asesorar procesos de transformación digital. Administrar proyectos de investigación y desarrollo tecnológico de vanguardia.	Desarrollo de software Soporte técnico y redes de telecomunicaciones Análítica de datos Investigación y desarrollo Administración de proyectos tecnológicos	10 semestres	Competencias Instrumentales: Aplicar estrategias de aprendizaje autónomo. Utilizar distintos lenguajes para comunicar ideas de diversa índole. Manejar las tecnologías de la información, comunicación, conocimiento y aprendizajes digitales. Dominar lengua materna de forma oral y escrita. Dominar un segundo idioma. Emplear el pensamiento lógico, crítico y creativo en la toma de decisiones. Elaborar propuestas académicas con apego a estándares mundiales de calidad. Personales: Compromiso y respeto a la diversidad de prácticas sociales y culturales. Intervenir en los retos sociales contemporáneos con actitud crítica, compromiso social y desarrollo sustentable. Integradoras:	Total 226 créditos:

Programa	Objetivos	Campo laboral	Duración	Perfil de egreso	Estructura
				Liderazgo en el cambio social pertinente. Construir propuestas innovadoras. Adaptarse a los ambientes sociales profesionales de incertidumbre.	
Ingeniería en Computación. <i>Universidad Autónoma de Guadalajara</i>	Formar ingenieros en computación con una cultura científica, tecnológica y humanística capaces de adaptar e incorporar los avances científicos y tecnológicos a su entorno, con compromiso y responsabilidad social.	Empresas públicas, privadas, empresa propia, diversas ramas de la ingeniería en el desarrollo de software, y realización de aplicaciones.	9 semestres	Conocimientos en: Área de algoritmos de computación. Diseño de software. Arquitectura de computadoras Redes de paradigmas modernos de cómputo y tecnologías emergentes. Capacidad para: Aplicar principios matemáticos y técnicas básicas de programación. Diseñar sistemas de software e información, para solucionar problemas reales. Aplicar diversas arquitecturas de computadoras. Implementar tecnologías de redes computacionales. Trabajar en equipo, con responsabilidad, ética, calidad, innovación y liderazgo.	Total 375 créditos: Formación básica común (149) Formación básica particular (144) Formación especializante obligatoria (50) Formación especializante selectiva (16) Formación optativa abierta (16)

Nota. Elaboración propia con base en Universidad de Sonora (2023); Benemérita Universidad Autónoma de Puebla (2022); Universidad Autónoma de Yucatán (2023); Universidad Nacional Autónoma de México (2023); Universidad Autónoma de Guadalajara (2023).

A continuación, se muestran algunas de las mejores prácticas identificadas en los programas nacionales que ponen a disposición de los interesados la licenciatura en ciencias computacionales.

La **Universidad de Sonora (UNISON)** tiene 35,026 estudiantes inscritos, el 97% pertenecen al nivel de licenciatura, mismos que son atendidos por 1,654 docentes; se ubica en el lugar 201-250 del *Latin America University Rankings* (QS, 2023). El Departamento de matemáticas coordina el programa educativo de Licenciatura en Ciencias de la Computación y tiene 25 años de instauración. Entre los recursos e instalaciones se encuentra un Centro de cómputo, el Laboratorio de Redes, el Laboratorio de ingeniería de software, y el Área de cómputo de alto rendimiento, el Laboratorio de Investigación y Desarrollo de Cómputo Avanzado, el programa de emprendedores y el programa de incubación de empresas, todo esto puesto a disposición de la comunidad en general, ya sea con fines de investigación o asesoría. La gama de opciones de pasantía que tienen los estudiantes de esta licenciatura son: cubrir la totalidad de los créditos del programa; presentar el Examen General de Egreso de Licenciatura (EGEL); y cumplir con el Servicio Social Universitario. La UNISON ofrece como opciones de titulación un promedio mayor o igual a 90.0; Examen nacional de calidad del CENEVAL; experiencia profesional, curso de titulación, tesis profesional, trabajo profesional, servicio social comunitario, y/o prácticas profesionales.

La **Benemérita Universidad Autónoma de Puebla (BUAP)**, es una universidad pública que registra 94, 692 estudiantes, el 97% son de licenciatura. El cuerpo docente está integrado por 2, 645 profesores. Esta universidad se encuentra en el lugar 74 del *Latin America University Rankings* (QS, 2023). La Licenciatura en Ciencias Computacionales, forma parte de los programas instaurados en esta institución, misma que se encuentra acreditada ante el Consejo Nacional para la Acreditación de Informática y Computación (CONAIC).

Las asignaturas del Nivel Básico son: Formación Humana y Social, Matemáticas Elementales, Álgebra Superior, Cálculo Diferencial, Álgebra Lineal, Cálculo Integral, Metodología de la Programación, Ensamblador, Programación I, Programación II,

Estructuras Discretas, Circuitos Eléctricos, Lógica Matemática, Lenguajes Formales y Autómatas, Estructuras de Datos, Lengua Extranjera I, II, III, IV.

Las asignaturas del Nivel Formativo: Circuitos lógicos, Probabilidad y Estadística, Sistemas Operativos I, Programación Concurrente y Paralela, Sistemas Operativos II, Fundamentos de Lenguajes de Programación, Programación Distribuida, Computabilidad, Inteligencia Artificial, Arquitectura Funcional de Computadoras, Redes de Computadoras, Graficación, Bases de Datos, Seguridad en Redes, Recuperación de la Información, Ingeniería de Software, Análisis y Diseño de Algoritmos, Administración de Proyectos, Servicio Social y Prácticas Profesionales.

Del total de Asignaturas Optativas, los estudiantes deben cursar cinco. Entre las opciones se encuentran: Aplicaciones WEB, Web Semántica, Aprendizaje Automático y Heurísticas, Animación por Computadora, Compiladores, Criptografía, Demostración Automática de Teoremas, Estadística Avanzada, Métodos Formales, Programación de Dispositivos Móviles, Redes Avanzadas, Minería de Datos, Ingeniería de Software Avanzada, Bases de Datos Avanzadas, Interacción Humano Computadora, Procesamiento Digital de Imágenes, Big Data, Bases de Datos Distribuidas, Programación de Videojuegos. Optativas Complementarias: Tópicos Selectos de la Computación I, Tópicos Selectos de la Computación II, Microprocesadores.

En la **Universidad Autónoma de Yucatán** se brinda servicio a 25, 527 estudiantes, de licenciatura y cuenta con 745 profesores de tiempo completo. Por su calidad, se ubica 151-160 del Latin America University Rankings (QS, 2023). Entre los 113 programas educativos que ofrece se encuentra la Licenciatura en Ciencias de la Computación. Este programa se encuentra acreditado por el Consejo Nacional de Acreditación en Informática y Computación, A. C. [CONAIC]; CIIES nivel 1; y Consejo para la Acreditación de la Educación Superior [COPAES].

Esta licenciatura tiene el objetivo de cubrir 360 créditos, los cuales se destinan en su mayoría para asignaturas obligatorias (268), asignaturas optativas (54), libres (18), servicio social (12), y prácticas profesionales (8).

Se integra por asignaturas del área de Matemáticas, Computación y Estadística, optativas y otras como la Cultura maya y Responsabilidad social universitaria. En el primer grupo de matemáticas se observan: Álgebra superior, Álgebra avanzada, Álgebra lineal, Álgebra intermedia y Matemáticas discretas; Geometría analítica, Cálculo diferencial, Cálculo integral, Cálculo vectorial y Ecuaciones diferenciales; Algoritmia y programación estructurada. En el segundo grupo de asignaturas se encuentran las siguientes: Estructura de datos, Teoría de la computación, Arquitectura y organización de computadoras, Programación orientada a objetos, Análisis de algoritmos, probabilidad Teoría de lenguajes de programación, Modelado de datos, Inferencia estadística, Sistemas operativos, Análisis y diseños de software, Compiladores, Metodología de la investigación, Métodos numéricos, Desarrollo y mantenimiento de software, Redes de computadoras, Cómputo científico, Sistemas distribuidos, Gestión de tecnologías de la información, Gráficas por computadora, Inteligencia artificial, Administración de proyectos tecnológicos.

La Facultad de Ciencias de la **Universidad Nacional Autónoma de México** (UNAM), tiene la licenciatura en Ciencias Computacionales con sede en la Ciudad de México. Esta carrera es catalogada por la propia universidad como alta demanda, debido a que el número de aspirantes que desean ingresar es de 1, 108 y de los cuales únicamente serán aceptados 120. La UNAM recomienda como alternativa institucional a la UABC, la Benemérita Universidad Autónoma de Puebla y a la Universidad Autónoma de Yucatán.

Este programa cuenta con la acreditación del Consejo Nacional de Acreditación de Informática y Computación A. C. (CONAIC). El plan de estudios avalado por este organismo acreditador, se estructura en 40 asignaturas agrupadas en siete ejes obligatorios, un eje de inglés y un grupo de asignaturas optativas. Los ejes obligatorios son: fundamentos matemáticos, estructuras discretas, ingeniería de software, programación, computación teórica, integración teórica-práctica y organización de sistemas de cómputo. Los estudiantes pueden elegir entre una amplia oferta de asignaturas optativas que refuerzan o amplían los conocimientos, y se agrupan en áreas que van desde lenguajes de programación, redes y seguridad, computación teórica, hasta bioinformática, entre muchos más.

En esta licenciatura se ofrece una variedad de opciones de titulación: Tesis, apoyo a la investigación, apoyo a la docencia, seminario de titulación, experiencia profesional,

extensión del servicio social, proyecto de apoyo a la divulgación, reporte de taller de experiencia profesional, resultados sobresalientes en concursos internacionales de computación, y alto rendimiento académico. La UNAM ofrece cuatro reconocimientos de carreras técnicas a los estudiantes que no culminaron la totalidad de créditos: técnico en desarrollo de aplicaciones de software, técnico en bases de datos, técnico en coordinación de procesos de software, y técnico en infraestructura de software para sistemas de cómputo. Estas opciones respaldan las habilidades adquiridas, hayan o no culminado el programa de licenciatura.

La **Universidad Autónoma de Nuevo León** es una de las universidades públicas más grandes de México. Por su calidad, se ubica en el lugar 48 en el Latin American University Rankings 2023 (QS, 2023). En esta institución se atienden a 132, 813 estudiantes distribuidos en 38 preparatorias y 26 facultades.

La Facultad de Ciencias Físico-Matemáticas de esta universidad oferta la licenciatura en Ciencias Computacionales. Este programa cuenta con la acreditación nacional de los Comités Interinstitucionales para la Evaluación de la Educación Superior A. C. (CIEES), así como de la AcreditAcción, agencia evaluadora de Chile.

En el plan de estudios se observan grupos de asignaturas de matemáticas, estadística y probabilidad, liderazgo y emprendimiento en los negocios y empresas, tecnologías computacionales, programación, redes, ciberseguridad, y diversos laboratorios. A partir del sexto semestre y hasta el noveno, se cursan ocho asignaturas optativas denominadas del área fundamental y del área y profesional integradora; dentro de esta última se incluyen las prácticas profesionales, estancias de investigación, así como la movilidad académica y educación continua. En el décimo semestre, conjuntamente con otras asignaturas específicas, también forman a los estudiantes para el examen de egreso y para el desempeño profesional. Como requisitos de egreso se establecen haber cubierto la totalidad de los créditos, el servicio social, presentar el examen de egreso, y el fortalecimiento del idioma inglés. A continuación, se mencionan las asignaturas del plan de estudios por semestre y los créditos correspondientes.

Primer semestre: Metodología de la programación, Cálculo diferencial, Geometría Analítica y Álgebra (4 créditos c/u); Responsabilidad social y desarrollo sustentable, Cultura de Paz (2 créditos c/u).

Segundo Semestre: Cálculo integral, Programación básica, Tópicos de álgebra, Mecánica traslacional y rotacional, (4 créditos c/u); Ética y cultura de legalidad, Cultura de género, Liderazgo, emprendimiento e innovación (2 créditos c/u).

Tercer Semestre: Matemáticas discretas, Álgebra lineal, Física para computación, Fundamentos de sistemas operativos, Programación estructurada, (4 créditos c/u); Laboratorio de programación estructurada, Laboratorio de física para computación (2 créditos c/u).

Cuarto semestre: Estructura de datos, Teoría de autómatas, Teoría de la información, (4 créditos c/u); Circuitos digitales, Fundamentos de redes, (3 créditos c/u); Laboratorio de circuitos digitales, Laboratorio de fundamentos de redes (2 créditos c/u).

Quinto semestre: Análisis numérico para programación, Teoría de la información aplicada (4 créditos c/u); Bases de datos, Algoritmia y optimización, Programación orientada a objetos (3 créditos c/u); Laboratorio de base de datos, Laboratorio de programación orientada a objetos (2 créditos c/u).

Sexto semestre: Programación lineal, Inteligencia artificial, Inglés para tecnologías (4 créditos c/u). *Séptimo semestre:* Investigación de operaciones, Cómputo en la nube, Fundamentos de seguridad informática, Compiladores, Minería de datos (3 créditos c/u).

Octavo semestre: Investigación y desarrollo (4 créditos), Análisis de sistemas (3 créditos).

Noveno semestre: Cómputo de alto rendimiento (4 créditos), Ingeniería de software (3 créditos), Servicio social (16 créditos). *Décimo semestre:* Administración de proyectos tecnológicos, (4 créditos); Modelo de negocios, Transformación digital, Analítica de datos e inteligencia de negocio, Taller para examen de egreso (3 créditos c/u); Seminario para el desempeño profesional, (2créditos).

La **Universidad de Guadalajara** es una institución pública que atiende a más de 140,000 estudiantes, el 96% pertenecen al nivel de licenciatura, y son atendidos por 11, 703 docentes. En el QS Latin America University Rankings 2023 se ubica en el lugar 42 (QS,

2023). Entre los 130 programas que ofrece se encuentra la Ingeniería en Computación. Esta tiene el objetivo de formar ingenieros capaces de adaptar e incorporar los avances científicos y tecnológicos a su entorno, con compromiso y responsabilidad social. Para lograrlo, este programa educativo se estructura en 9 semestres, a través de las siguientes áreas de formación y seminarios.

Asignaturas del área de formación básica común: Matemática discreta, Teoría de la computación, Algoritmia, Programación, Estructuras de datos I, II y III, Bases de datos, Programación para Internet, Ingeniería de software I, Métodos matemáticos I, II y III, Estadística y procesos estocásticos (8 créditos c/u).

Asignaturas del área de formación básica particular: Fundamentos filosóficos de la computación, Arquitectura de computadoras, Traductores de lenguajes I y II, Sistemas operativos, Redes de computadoras y protocolos de comunicación, Sistemas operativos de red, Sistemas concurrentes y distribuidos, Computación tolerante a fallas, Seguridad, Inteligencia artificial I y II, Simulación por computadora (cada una con valor de 8 créditos).

Asimismo, ofrece *Seminarios de solución de problemas* de diversas temáticas, entre las que se encuentran arquitectura de computadoras, traductores de lenguajes I y II, Sistemas operativos, redes de computadoras y protocolos de comunicación, Sistemas operativos de red, Inteligencia artificial I y II, Algoritmia, Programación, estructuras de datos I y II, Bases de datos, Ingeniería de software I y Métodos matemáticos I, II y III, (con valor de 5 créditos c/u)

Formación especializada obligatoria: proyecto de arquitectura y programación de sistemas (20 créditos), proyecto de sistemas inteligentes y proyecto de sistemas distribuidos (15 créditos c/u). Por último, se ofrece al Área de formación especializada selectiva y el Área de formación optativa abierta (16 créditos c/u).

En el apartado que se presenta a continuación, se expondrán las características de los programas internacionales: duración, objetivos, estructura del plan de estudios, perfil de egreso y campo laboral. Las instituciones a las que pertenecen los programas analizados son: Massachusetts Institute of Technology, Stanford University, Carnegie Mellon University, University of California-Berkeley, y University of Oxford. En la tabla X se aprecia que la duración de los programas en Ciencias Computacionales es de cuatro años, excepto por el MIT que establece su plan de estudios para tres años.

En el **MIT** se observó que, en comparación con lo declarado por el resto de las universidades, forman a sus estudiantes con habilidades, conocimientos y creatividad para demostrar liderazgo en la solución de problemas complejos que competen a esta área del conocimiento.

La **Universidad de California, Berkeley** tiene el objetivo de preparar a sus egresados para que se desenvuelvan con liderazgo técnico o para dar continuidad a los estudios de posgrado. En esta universidad se muestra una capacitación con centros y laboratorios de investigación, tanto propios como de otras universidades y una colaboración con el sector empresarial.

En la **Universidad de Oxford** les interesa que sus estudiantes logren vincular la teoría con aspectos de la práctica. Para alcanzar su objetivo tienen un entrenamiento constante en la solución de problemas, que primero realizan con la guía de sus docentes y después en conjunto con empresas. También tienen una capacitación con distinguidos expertos mundiales, lo que en conjunto logra en el estudiante una dotación de habilidades de pensamiento riguroso y práctico sobre el comportamiento de los sistemas informáticos.

La **Universidad de Stanford** no declara objetivos de su programa, sin embargo, busca que sus egresados tengan un acercamiento con el sector empresarial y que tengan habilidades y conocimientos para continuar con estudios de posgrado. De igual forma, busca que sus estudiantes puedan desenvolverse con equipos de otras disciplinas dentro de la misma universidad.

La **Universidad Carnegie Mellon** se centra en la inteligencia artificial, gráficos y sonido, ingeniería de software y en los tipos de interacción de los seres humanos y las computadoras.

En el campo laboral, esta universidad tiene un vínculo con las empresas de tecnología de alto prestigio mundial dedicadas al desarrollo de hardware y software, por lo que sus estudiantes están formados de manera muy cercana con estas empresas durante todo su trayecto escolar y logran posicionarse en puestos laborales. Los egresados también pueden trabajar en sus propias empresas de consultoría, en la docencia o continuar con estudios de posgrado.

Tabla 1.27*Programas Educativos Internacionales iguales a LCC*

Programa	Objetivos	Campo Laboral	Duración	Perfil de egreso
Bachelor of Science in Computer Science and Engineering. <i>MIT</i>	Los egresados serán capaces de aportar los enfoques físicos, analíticos y computacionales aprendidos para encontrar la estructura esencial, origen de incertidumbre y aplicación de modelos, herramientas técnicas y evaluaciones apropiados. Los egresados serán capaces de demostrar liderazgo e influir en sus equipos, empresas y campos. Los egresados aplicarán habilidades, conocimiento y talento creativo y productivamente en los campos previstos o no por el plan de estudios. Los graduados demostrarán sensibilidad a los diversos contextos, compromiso ético y aprendizaje permanente para ser un miembro efectivo de su comunidad.	Iniciativa privada Docencia	Tres años	Capacidad para: Resolver problemas complejos de ingeniería aplicando los principios de ingeniería, ciencias y matemáticas. Aplicar el diseño de ingeniería para generar soluciones que satisfagan necesidades tomando en consideración los diversos factores. Comunicación efectiva en diversos contextos profesionales. Actuar con ética y profesionalismo. Funcionar efectivamente creando entorno colaborativo. Desarrollar, analizar, e interpretar adecuadamente los resultados de experimentos en ingeniería. Diseñar, implementar y evaluar una solución informática. Aplicar la teoría de las ciencias de la computación y los desarrollos del software para aplicar soluciones basados en la computación.
Computer Science. <i>Stanford University</i>	No se declara	Diversas empresas de tecnología Docencia Estudios de posgrado	4 años	No se declara
Computer Science. <i>Carnegie Mellon University</i>	No se declara	Empresas de consultoría Compañías de Software Corporaciones Fortune 500 Agencias gubernamentales	4 años	Habilidades: Identificar, usar, diseñar, desarrollar y analizar abstracciones y algoritmos para resolver problemas. Solucionar problemas en dominios como

Programa	Objetivos	Campo Laboral	Duración	Perfil de egreso
				inteligencia artificial, gráficos y sonido, ingeniería de software e interacción humano-computadora. Razonar e implementar programas en varios lenguajes y paradigmas de programación. Describir, especificar y desarrollar sistemas de software abiertos a gran escala. Comunicar efectivamente material técnico para todo tipo de audiencias. Trabajar en equipo y de manera individual. Reconocer el impacto social de la informática y la responsabilidad de considerar las implicaciones legales, morales y éticas de las tecnologías informáticas.
Computer Science. <i>University of California, Berkeley</i>	Preparar a los estudiantes para un liderazgo técnico o para los estudios de posgrado.	Ingeniería de software Desarrollador web Analista de datos Consultoría Estudios de posgrado	4 años	No presenta el perfil de egreso.
Computer Science. <i>University of Oxford</i>	El curso se concentra en crear vínculos sólidos entre la teoría y la práctica; cubre una amplia gama de hardware y software.	Fabricantes de computadoras Empresas de software Consultoría de gestión Finanzas Docencia Estudios de posgrado	4 años	Habilidades de razonamiento rigurosos sobre el comportamiento de los programas y sistemas informáticos. Habilidades para la resolución de problemas y diseño de programas.

Las mejores prácticas y estrategias de los programas de licenciatura en ciencias computacionales ubicadas a nivel internacional, se muestran a continuación. En el departamento de Ingeniería Eléctrica y Ciencias Computacionales del **MIT** se encuentran inscritos más de 1,300 estudiantes universitarios, y 750 de posgrado. Los estudiantes tienen acceso a diversos laboratorios como el Laboratorio de Sistemas de Información y Decisión del MIT, el Laboratorio de Ciencias de la Computación e Inteligencia Artificial, el Laboratorio de Tecnología de Microsistemas, y el Laboratorio de Investigación de Electrónica, todos estos laboratorios se encuentran a la vanguardia.

El currículo de este programa se encuentra diseñado para cursar en 6 semestres y se distribuye de la siguiente manera: tres o cuatro asignaturas de introducción (habilidades de programación, introducción a la ingeniería y ciencias computacionales, y matemáticas discretas); tres asignaturas fundamentales (estructuras computacionales, programación y algoritmos); cuatro asignaturas especializantes (*header subjects*) (inteligencia artificial y aprendizaje automático, sistemas computacionales, construcción de software, algoritmos y computación, y laboratorio.); dos asignaturas avanzadas (dominio profundo de áreas de interés); una asignatura de comunicación y seis optativas. La licenciatura en Ciencias Computacionales e Ingeniería del MIT se encuentra certificada por el ABET, organismo acreditador estadounidense de programas de licenciatura y maestría del área de ciencias naturales y aplicadas, y programas de computación, ingeniería y tecnología de ingeniería.

Los estudiantes universitarios tienen la oportunidad de unirse a la investigación; participar en el programa industrial; realizar estudios en el extranjero (principalmente con ETH Zurich y el Imperial Collage London); o unirse a grupos de estudiantes (USAGE grupo asesor de estudiantes de licenciatura, ETA Kappa Nu, MIT IEE/ACM Club) cuya finalidad es asesorar a otros estudiantes, organizar eventos académicos o discutir desde el plan de estudios tanto del departamento de ingeniería, como el plan general del MIT, discutir y defender la diversidad e inclusión, entre otros temas.

En la licenciatura de Ciencia de la Computación de la **Universidad de Stanford**, el plan de estudios se estructura en cinco áreas: Introducción, especialización, materias optativas y proyecto senior. Los estudiantes toman seis asignaturas de introducción a la informática. Después, se especializan en alguna área entre inteligencia artificial,

biocomputación, ingeniería informática, gráficos, interacción persona-ordenador, información, sistemas, o teoría; mediante un máximo de cinco asignaturas. Además, deben cursar de 2 a 4 asignaturas optativas, relacionadas o no con la especialidad, es posible que puedan tomar cursos en otros departamentos de Stanford. Finalmente, los estudiantes deben entregar un proyecto senior.

Este proyecto se obtiene con alguna de las siguientes opciones: la primera consiste en desarrollar una aplicación de programación, la cual exponen al personal escolar, estudiantes y empresas locales. La segunda opción es desarrollar un proyecto en dos trimestres con algún socio corporativo (Facebook, Yahoo! o BMW), para ello tienen asignado un presupuesto, un consejo asesor y retos reales del mundo laboral. Una tercera opción es realizar una investigación que puede ser individual (aplicación de software o investigación publicable) en conjunto con un asesor de la facultad; tesis dirigida, en donde el estudiante se asocia a una investigación más grande en curso; o tesis de investigación con honores, en la que realiza investigación dirigida por un profesor durante un año. Estas variantes de proyecto senior son recomendadas para quienes deseen continuar estudios de posgrado.

Los estudiantes de la Facultad de Ingeniería, dentro del cual se encuentra el Departamento de Ciencias Computacionales, cuentan con varios laboratorios, grupos de investigación y conexiones con empresas. En el listado de laboratorios tienen el de inteligencia artificial de Stanford, y el Laboratorio de información de Stanford. Tres grupos de investigación como el Grupo de Teoría de Stanford, Grupo de Interacción Humano-Computadora, y el de Sistemas Informáticos. Los egresados también se benefician de una amplia cartera de empresas dentro y fuera de Silicon Valley, en las que pueden asociarse desde que cursan la carrera y con amplias posibilidades laborales: Apple, Bloomberg, ebay, Fujitsu, Google, Hitachi, IBM, Meta, Microsoft, Samsung, Zoom, Uber, el propio Stanford Engineering Computer Science, entre otros. Destaca el edificio Gates, con dos salas equipadas con herramientas y equipo de presentación y de cómputo; también cuenta con dos laboratorios informáticos y diversos espacios destinados para docentes y estudiantes.

La **Carnegie Mellon University** es una institución privada, con sede principal en Pittsburgh, Pensilvania. Actualmente, atiende a 12,692 estudiantes, el 50% son de

licenciatura. En el departamento de Ciencias Computacionales de esta universidad se ofrece la licenciatura en Ciencias Computacionales, la estructura del Plan de Estudios es como se muestra a continuación.

- 1) Ciencias computacionales (12 cursos, 125 créditos).
- 2) Matemáticas (5 cursos, 48 créditos).
- 3) Comunicación técnica (1 curso, 9 créditos).
- 4) Ciencias/Ingeniería (4 cursos, 36 créditos).
- 5) Humanidades/Artes (7 cursos, 63 créditos).
- 6) Especialidad (75 créditos).
- 7) Computación Carnegie Mellon (1 curso, 3 créditos).
- 8) Seminario del primer año (1 curso, 1 crédito).

Los estudiantes interesados en la investigación tienen la opción de presentar tesis de investigación, para lo cual trabajan durante el último año. En cuanto a los temas, los estudiantes tienen una amplia gama de opciones que van desde gráficos de computadora, sistemas informáticos, la interacción persona-ordenador, tecnologías del lenguaje, aprendizaje automático, algoritmos y complejidad, principios de los lenguajes de programación, robótica, seguridad y privacidad, ingeniería en software, hasta biología computacional.

En la **Universidad de Berkeley** se registraron 32,479 estudiantes de licenciatura en el otoño del 2022, de los cuales, el 6% pertenecen a los programas de Ciencias Computacionales.

En el Departamento de Ingeniería Eléctrica y Ciencias Computacionales se ofrecen estudios de posgrado, así como la Ingeniería en Electrónica e Informática y la Licenciatura en Ciencias de la Información. Esta última está diseñada para otorgar a los estudiantes tanto conocimiento teórico de la computación, como técnico y aplicado de los métodos de procesamiento de la información, diseño de hardware, software y aplicaciones; tomando en consideración los diversos contextos sociales y humanos y las implicaciones éticas para recopilar, analizar y utilizar los datos.

Para cumplir con sus objetivos de formación, este departamento se encuentra afiliado a más de 70 centros y laboratorios de investigación. También colabora con otros

departamentos dentro de la institución y con universidades como el MIT, Stanford, UC San Diego. Además, participa y se asocia con agencias de gobierno, y compañías privadas entre las que se puede mencionar AT&T, Google, Oracle, Pixar, Nokia, Samsung, Toyota, entre muchas más. Así, los estudiantes estarán preparados para informar y desarrollar soluciones ante una variedad de retos y oportunidades que suceden en el mundo laboral.

La **Universidad de Oxford** es una de las instituciones públicas de alto prestigio a nivel mundial y se encuentra en el Reino Unido. En el QS World University Rankings 2024 se posiciona en el tercer lugar. En la actualidad atiende a 21,972 estudiantes, el 56% se registran en licenciatura. Cuenta con una planta docente de 6,650 elementos, el 45% son extranjeros (QS, 2023).

En la Universidad de Oxford, se advierte a los interesados en la licenciatura en ciencias computacionales sobre la manera distinta que tienen de formar a sus estudiantes. La Licenciatura comienza el primer año con cursos, conferencias y tutorías. Los cursos pueden variar en temáticas, sin embargo, abarcan las matemáticas, programación, sistemas digitales, álgebra, o probabilidad; la evaluación del 100% se otorga a cuatro exámenes. Las conferencias son impartidas, entre 8 y 9 por semana, a través de expertos mundiales en temas sobre el funcionamiento de los sistemas informáticos. Las tutorías se realizan tres veces por semana, por pares de estudiantes, con la finalidad de revisar las soluciones a problemas establecidos por el docente. En el segundo año, los estudiantes ejecutan una práctica grupal en conjunto con una empresa, en donde diseñarán una solución a un problema práctico, será equivalente al 50% de la calificación; también se incluyen cursos de algoritmos, programación, modelos computacionales que serán evaluados mediante ocho exámenes y representan el restante 50%. Asimismo, durante el tercer y cuarto año, se imparten cursos y presentarán de cinco a seis exámenes equivalentes al 67-62% de la calificación. El proyecto individual es guiado por un asesor experto, por medio del cual profundizan en alguna temática del curso y representa el restante 33-38%.

De igual forma, aunque no forma parte del curso, se imparten seminarios; estos son impartidos principalmente por representantes de la industria y abarca temas como seguridad cibernética y sistemas de información y verificación. Durante el verano, los estudiantes pueden acercarse al mundo laboral. Esto sucede mediante pasantías ofrecidas por las

diferentes empresas como Bloomberg, Google, Amazon, entre otras. De esta manera, se ilustra la estructura y funcionamiento del plan de estudios ofrecido en esta universidad de prestigio mundial.

Conclusiones del Análisis comparativo de programas

- La duración de la licenciatura en Ciencias Computacionales es de tres tipos: 8 semestres (UNISON y UNAM), 9 semestres (Universidad Autónoma de Yucatán), y 10 semestres (BUAP y Universidad Autónoma de Nuevo León).
- Los créditos solicitados para completar el plan de estudios oscilan en la UNISON y la UNAM entre 383 y 376 créditos respectivamente. La Universidad Autónoma de Yucatán solicita 360. Mientras la BUAP y la Universidad de Nuevo León precisan 293 y 226 créditos cada una.
- Los conocimientos, capacidades y habilidades engloban de manera general los siguientes: La BUAP, la Universidad Autónoma de Yucatán, y la UNAM se interesan en la aplicación de conocimientos sobre el diseño e implementación de software y hardware eficientes. Por su parte, la UNAM perfila a sus estudiantes para aprender sobre la inteligencia artificial. La Universidad Autónoma de Nuevo León y la UNAM promueven habilidades para solucionar problemas con rigor científico y realizar propuestas académicas con apego a estándares internacionales. De manera particular, la Universidad de Guadalajara forma Ingenieros en Computación, por lo cual promueve los conocimientos en algoritmos de computación, arquitectura de computadoras y redes de paradigmas modernos de cómputo y tecnologías emergentes.
- En el campo laboral se especifican la Consultoría e investigación en empresas públicas y privadas, el sector empresarial en el desarrollo de hardware y software, así como la Docencia. En la BUAP adicionalmente, se puntualiza el desarrollo de aplicaciones web y videojuegos. En la UNAM, además, se capacita a los estudiantes para dar continuidad a los estudios de posgrado. Mientras en la Universidad de Nuevo León, los egresados se pueden desempeñar también como analistas de datos.

- En la UNISON, el Departamento de matemáticas coordina el programa educativo de Licenciatura en Ciencias de la Computación y tiene 25 años de instauración. Cuenta con diversos laboratorios de cómputo, el programa de emprendedores y el programa de incubación de empresas.
- En la Benemérita Universidad Autónoma de Puebla, El programa de LCC se encuentra acreditada por el Consejo Nacional de Acreditación en Informática y Computación, A. C. (CONAIC).
- En la Universidad Autónoma de Yucatán, la licenciatura en Ciencias de la Computación. está acreditada por el CONAIC; CIIES nivel 1; y Consejo para la Acreditación de la Educación Superior (COPAES).
- La UNAM, tiene el programa de LCC con sede en la Ciudad de México. Esta carrera es catalogada como alta demanda, debido a que el número de aspirantes que desean ingresar es de 1,108 y de los cuales únicamente serán aceptados 120. La UNAM recomienda como alternativa institucional a la UABC, la Benemérita Universidad Autónoma de Puebla y a la Universidad Autónoma de Yucatán. Resalta la variedad de opciones de titulación: Tesis, apoyo a la investigación, apoyo a la docencia, seminario de titulación, experiencia profesional, extensión del servicio social, proyecto de apoyo a la divulgación, reporte de taller de experiencia profesional, resultados sobresalientes en concursos internacionales de computación, y alto rendimiento académico.
- La Universidad Autónoma de Nuevo León oferta LCC. Este programa cuenta con la acreditación nacional de los Comités Interinstitucionales para la Evaluación de la Educación Superior A. C. (CIEES), así como de la AcreditAcción, agencia evaluadora de Chile. En el plan de estudios se observan grupos de asignaturas de matemáticas, estadística y probabilidad, liderazgo y emprendimiento en los negocios y empresas, tecnologías computacionales, programación, redes, ciberseguridad, y diversos laboratorios.
- La Universidad de Guadalajara ofrece la Ingeniería en Computación, tiene el objetivo de formar ingenieros capaces de adaptar e incorporar los avances

científicos y tecnológicos a su entorno, con compromiso y responsabilidad social; se cursa en nueve semestres.

- En el MIT se centran en formar a sus estudiantes con habilidades, conocimientos y creatividad para demostrar liderazgo en la solución de problemas complejos.
- En la Universidad de California, Berkeley dan prioridad a la formación en liderazgo técnico y la continuidad hacia los estudios de posgrado. En esta universidad se muestra una capacitación con centros y laboratorios de investigación, tanto propios como de otras universidades y una colaboración con el sector empresarial.
- En la Universidad de Oxford les interesa vincular la teoría con la práctica, a través de un entrenamiento constante en la solución de problemas, con sus docentes y empresas. También tienen una capacitación con distinguidos expertos mundiales para dotar a sus estudiantes con habilidades de pensamiento riguroso y práctico sobre el comportamiento de los sistemas informáticos.
- La Universidad de Stanford busca que sus egresados tengan un acercamiento con el sector empresarial y que tengan habilidades y conocimientos para continuar con estudios de posgrado. De igual forma, busca que sus estudiantes trabajen de forma interdisciplinaria dentro de la misma universidad.
- La Universidad de Carnegie Mellon tiene un vínculo con las empresas de tecnología de alto prestigio mundial dedicadas al desarrollo de hardware y software, por lo que sus estudiantes están formados de manera muy cercana con estas empresas y logran posicionarse en puestos laborales. Los egresados también pueden trabajar en sus propias empresas de consultoría, en la docencia o continuar con estudios de posgrado.

1.2.3 Análisis de organismos nacionales e internacionales

Objetivos

La Tabla 1.28 presenta el objetivo general y los objetivos específicos correspondientes en este análisis.

Tabla 1.28

Objetivos del Análisis de organismos nacionales e internacionales

Objetivo general	Objetivo específico
Analizar los referentes nacionales e internacionales que señalan competencias, contenidos de dominio y prácticas que deben cubrirse para apoyar la creación, modificación o actualización del plan de estudios	Identificar las competencias, contenidos de dominio y prácticas de la profesión que señalan o recomiendan los organismos nacionales, como COPAES, CIEES u otras asociaciones, cámaras, colegios, —entre otras—, e internacionales para que sean incluidos en la creación, modificación o actualización del programa educativo.
	Identificar las competencias, contenidos de dominio y prácticas de la profesión de las evaluaciones internas y externas como el Examen General para el Egreso de la Licenciatura (EGEL) que realiza CENEVAL, para considerarlos en la creación, modificación o actualización del programa educativo.

Nota. De “Metodología de los estudios de fundamentación para la creación, modificación y actualización de programas educativos de licenciatura”, por A. Serna, y A. Castro, 2018, *Universidad Autónoma de Baja California*. CC BY-NC-ND.

Metodología

En este apartado se realizó una investigación documental. Con la finalidad de alcanzar los objetivos generales y específicos, se revisaron las páginas web del Consejo para la Acreditación de la Educación Superior (COPAES) y el Centro Nacional de Evaluación para la Educación Superior (Ceneval).

Resultados

El Consejo Nacional de Acreditación en Informática y Computación A. C. (CONAIC) es un organismo cuyo objetivo es evaluar con fines de acreditación aquellos programas académicos del nivel superior del área de computación, informática, y tecnologías de la información y telecomunicación (CONAIC, 2017). El CONAIC es supervisado por el Consejo para la Acreditación de la Educación Superior (COPAES). El COPAES es la única organización autorizada por el Gobierno de México para asegurar la calidad de la

educación superior, mediante el reconocimiento y supervisión de organizaciones acreditadoras de programas educativos del nivel superior en cualquier modalidad (COPAES, 2023).

Las categorías establecidas por el CONAIC para evaluar y emitir un dictamen de acreditación, son diez: personal académico, estudiantes, plan de estudios, evaluación del aprendizaje, formación integral, servicios de apoyo para el aprendizaje, vinculación-extensión, investigación, infraestructura y equipamiento, y gestión administrativa y financiamiento (CONAIC, 2017). Actualmente, el programa de LCC de la Facultad de Ciencias es un programa acreditado por el CONAIC con vigencia al 23 de mayo del 2026 (CONAIC, 2022).

Por otra parte, el Centro Nacional de Evaluación para la Educación Superior (Ceneval) es una agencia de evaluación educativa no gubernamental dedicada a la elaboración de conocimientos, habilidades y competencias para el ingreso y egreso de diversos niveles educativos, además de diagnósticos, acreditaciones y certificaciones profesionales de distintas áreas profesionales (Ceneval, 2023). En el caso de los Exámenes Generales de Egreso de el programa de LCC (EGEL Plus COMPU) el objetivo es evaluar el aprendizaje del estudiante al concluir o próximo a concluir un plan de estudios de nivel licenciatura; a través de los siguientes propósitos: Determinar el nivel de desempeño de los egresados, establecer el grado de dominio de los conocimientos y habilidades al término de la formación académica relacionada con su profesión y conocer su nivel de dominio en habilidades de lenguaje y comunicación —comprensión lectora y redacción— (Ceneval, 2023).

El EGEL Plus COMPU se conforma de dos secciones: Disciplinar, específica de la profesión y Transversal de Lenguaje y comunicación, común a todas las profesiones; cada sección se divide por áreas y subáreas con un número total de 200 ítems (Ceneval, 2023) (ver Tabla 1.29).

Tabla 1.29

Estructura del EGEL Plus COMPU

Secciones	Áreas	Subáreas	Temas	No. ^a

Secciones	Áreas	Subáreas	Temas	No. ^a
Disciplinar específica de la profesión	1. Algoritmia	1.1 Análisis y diseño de algoritmos		10
		1.2 Estructura de datos		10
		1.3 Matemáticas discretas		10
		1.4 Lógica computacional		10
	2. Desarrollo de software de base	2.1 Arquitectura de computadoras y sistemas operativos		10
		2.2 Compiladores		10
		2.3 Redes de computadoras		10
	3. Desarrollo de software de aplicación	3.1 Ingeniería de software		10
		3.2 Lenguajes de programación		10
		3.3 Bases de datos		10
		3.4 Seguridad informática		10
	4. Soluciones de cómputo inteligente	4.1 Inteligencia artificial		10
		4.2 Minería de datos		10
		4.3 Cómputo distribuido		10
Transversal de Lenguaje y comunicación, común a todas las profesiones	1. Comprensión lectora	1.1 Ámbito de estudio	• Identificación de información	12
		1.2 Ámbito literario	• Interpretación	12
		1.3 Ámbito de participación social	• Evaluación de la forma y el contenido	6
	2. Redacción indirecta	1.1 Ámbito de estudio	• Dimensión comunicativa	15
		1.2 Ámbito de participación social	• Dimensión gramatical y semántica	15
			• Dimensión ortográfica	
Total de ítems				200

Nota. ^a No.=Número de ítems.

Conclusiones del Análisis de organismos nacionales e internacionales

- El Consejo Nacional de Acreditación en Informática y Computación A. C. (CONAIC) es un organismo cuyo objetivo es evaluar con fines de acreditación aquellos programas académicos del nivel superior del área de computación, informática, y tecnologías de la información y telecomunicación (CONAIC, 2017). El CONAIC es supervisado por el Consejo para la Acreditación de la Educación Superior (COPAES).

- Las categorías establecidas por el CONAIC para evaluar y emitir un dictamen de acreditación, son diez: personal académico, estudiantes, plan de estudios, evaluación del aprendizaje, formación integral, servicios de apoyo para el aprendizaje, vinculación-extensión, investigación, infraestructura y equipamiento, y gestión administrativa y financiamiento (CONAIC, 2017).
- Actualmente, el programa de LCC de la Facultad de Ciencias es un programa acreditado por el CONAIC con vigencia al 23 de mayo del 2026 (CONAIC, 2022).
- El Examen General de Egreso del programa de LCC (EGEL Plus COMPU) tiene los siguientes propósitos: Determinar el nivel de desempeño de los egresados, establecer el grado de dominio de los conocimientos y habilidades al término de la formación académica relacionada con su profesión y conocer su nivel de dominio en habilidades de lenguaje y comunicación —comprensión lectora y redacción— (Ceneval, 2023).
- El EGEL Plus COMPU se conforma de dos secciones y se estructura por un total de 200 ítems: (a) Disciplinar, es específica de la profesión y evalúa cuatro áreas de las ciencias computacionales: Algoritmia, Desarrollo de software de base, Desarrollo de software de aplicación y Soluciones de cómputo inteligente; y (b) Transversal de Lenguaje y comunicación, común a todas las profesiones (Ceneval, 2023).

Conclusiones Estudio 1.2 Evaluación de referentes

1.2.1 Análisis de la profesión y su prospectiva

En el entorno de la profesión de ciencias computacionales se indica que los avances tecnológicos hacia el 2030 generarán el mayor progreso tecnológico sin precedentes en los últimos 100 años. Asimismo, las profesiones inmersas en el área de las ciencias computacionales se caracterizan por la rápida innovación, la colaboración internacional, la movilidad laboral y la necesidad de adaptar y crear cuestiones éticas y sociales a escala mundial. Desde la pandemia por Covid-19 se espera que la tecnología continúe con un papel central en futuras pandemias, guerras y desastres naturales. En este sentido, se

generaron 19 predicciones tecnológicas para los próximos 15 años (IEEE, 2023). En términos de avance del desarrollo tecnológico en 2023 e impacto para la humanidad, la mayor predicción de éxito a corto plazo la tendrá la atención médica remota y los dispositivos portátiles; mientras las oportunidades a largo plazo se señalan en las tecnologías espaciales y la digitalización global de las transacciones monetarias. Aunado a lo anterior, las oportunidades de desarrollo se distinguen por áreas: (a) industria, con la realidad aumentada, la salud, la tecnología para la sustentabilidad y el hardware abierto; (b) gobiernos, detección de la desinformación y los productos farmacéuticos generativos; (c) academia, todo lo relacionado con el desarrollo de tecnologías espaciales; (d) organizaciones profesionales, podrán focalizarse en la inteligencia artificial sostenible y la tecnología para el espacio (IEEE, 2023).

Entre *las prácticas de la profesión* que realizan los investigadores y profesionistas del área de sistemas computacionales se clasifican en dos grupos: (a) Desarrolladores y analistas de software y multimedia, y (b) Administradores de bases de datos y redes de computadora (INEGI, 2019). Las profesiones afines al programa de LCC con las que comparte su ejercicio y práctica profesional son cuatro: Licenciatura en informática, Ingeniería de software, la Ingeniería en ciencias computacionales y Licenciatura en ciencias computacionales (CONAIC, 2022).

La Evolución del programa de LCC muestra que, en México, el Instituto Politécnico Nacional (IPN) ofreció, desde 1965, las carreras de técnico en computación electrónica y técnico en mantenimiento de equipos de computación y electrónica, y la maestría en ciencias con especialidad en computación. La primera licenciatura en computación la impartió el Instituto Tecnológico y de Estudios Superiores de Monterrey en el año 1968, denominada Ingeniería en Sistemas Computacionales (CINVESTAV, s.f.). Durante el periodo 2001-2009, egresaron del nivel superior 3,705,243 estudiantes, la licenciatura representó el 81.5% del total. En el área de Ingenierías, se observó un crecimiento de egresados del 36%; el mayor incremento correspondió a la rama de metalurgia y extractiva con un 132%, mientras en computación e informática fue del 17%. La evolución de los egresados de esta área se encuentra estrechamente relacionada con la dinámica de crecimiento del sector industrial. Durante el mismo periodo, el crecimiento en el empleo se obtuvo en cinco áreas del conocimiento, las Ingenierías ocuparon el segundo lugar con el

17.3% en 2000 y el 17.7% en 2009; las carreras que destacaron fueron las Ingeniería mecánica, Ingeniería industrial, Ingeniería en computación e informática. El porcentaje de egresados de la Ingeniería en computación e informática ocupó el quinto lugar en 2009 entre 16 carreras (de 45) que aglutinaron al 65% de los egresados de licenciatura. El pronóstico hacia el 2020 que habría una demanda superior a la oferta de estos profesionistas, aunque no sería para empleos de calidad (Hernández-Laos et al., 2012). Actualmente, en el ciclo escolar 2022-2023, la licenciatura en ciencias computacionales se ofrece en ocho universidades públicas y registró una matrícula de 3,429 estudiantes, donde el 21.78% fueron de nuevo ingreso. Los programas afines, como la Ingeniería en ciencias computacionales o la Ingeniería en computación, se imparten en 21 universidades públicas y privadas (ANUIES, 2023).

El análisis de la Prospectiva del programa de LCC nos encamina hacia la llamada Revolución digital, la cual tendrá un impacto directo en la vida cotidiana de las personas, en la industria, el medio ambiente y los gobiernos. La transformación de la industria del siglo XXI presentará una evolución tecnológica continua (Oropeza-García, 2019). México sobresale como la quinceava economía mundial, y se pronostica como una próxima economía emergente (Buenrostro, 2019). Por lo que, la rápida evolución de la industria tecnológica abre las oportunidades de desarrollo industrial para nuestro país, y se espera la colaboración entre la industrial, el gobierno y la academia (Buenrostro, 2019).

De acuerdo con el Foro Económico Mundial, los empleos que crecerán un 30-35% hacia el 2027 serán los analistas y científicos de datos, especialistas en Big Data, analistas en inteligencia empresarial, profesionales de bases de datos y redes e Ingenieros de datos. La expectativa de crecimiento es medida para la industria de tecnología de la información y comunicaciones digitales con el 8%. El tipo de habilidades para el trabajo necesarias en 2023 serán: (a) cognitivas, pensamiento analítico y el pensamiento creativo; (b) autoeficacia, resiliencia, flexibilidad y agilidad, motivación y conciencia de sí mismo, curiosidad y aprendizaje a lo largo de la vida, confiabilidad y atención al detalle; (c) tecnológicas, literacidad tecnológica; (d) trabajo con otros, empatía y escucha activa, liderazgo e influencia social; y (d) habilidades de gestión, como el control de calidad. Hacia el 2027, las habilidades que las empresas esperan aumenten, dada su importancia en empleo

seguirán siendo las cognitivas como el pensamiento creativo (73.2%) y el pensamiento analítico (71.6%), así como la literacidad tecnológica (67.7%) (WEF, 2020).

1.2.2 Análisis comparativo de programas educativos

El análisis comparativo de programas educativos nacionales e internacionales permitió observar que la duración de la licenciatura en Ciencias Computacionales es de tres tipos: 8 semestres (UNISON y UNAM), 9 semestres (Universidad Autónoma de Yucatán), y 10 semestres (BUAP y Universidad Autónoma de Nuevo León). Los créditos solicitados para completar el plan de estudios oscilan en la UNISON y la UNAM entre 383 y 376 créditos respectivamente. La Universidad Autónoma de Yucatán solicita 360. Mientras la BUAP y la Universidad de Nuevo León precisan 293 y 226 créditos cada una.

Los conocimientos, capacidades y habilidades engloban de manera general los siguientes: La BUAP, la Universidad Autónoma de Yucatán, y la UNAM se interesan en la aplicación de conocimientos sobre el diseño e implementación de software y hardware eficientes. Por su parte, la UNAM perfila a sus estudiantes para aprender sobre la inteligencia artificial. La Universidad Autónoma de Nuevo León y la UNAM promueven habilidades para solucionar problemas con rigor científico y realizar propuestas académicas con apego a estándares internacionales. De manera particular, la Universidad de Guadalajara forma Ingenieros en Computación, por lo cual promueve los conocimientos en algoritmos de computación, arquitectura de computadoras y redes de paradigmas modernos de cómputo y tecnologías emergentes.

En lo referente al campo laboral se especifican la Consultoría e investigación en empresas públicas y privadas, el sector empresarial en el desarrollo de hardware y software, así como la Docencia. En la BUAP adicionalmente, se puntualiza el desarrollo de aplicaciones web y videojuegos. En la UNAM, además, se capacita a los estudiantes para dar continuidad a los estudios de posgrado. Mientras en la Universidad de Nuevo León, los egresados se pueden desempeñar también como analistas de datos.

Entre las características que destacaron en cada programa analizado son por ejemplo en la UNISON, LCC tiene 25 años de instauración. Cuenta con diversos laboratorios de cómputo, el programa de emprendedores y el programa de incubación de empresas. En la Benemérita Universidad Autónoma de Puebla, El programa de LCC se encuentra acreditada

por el Consejo Nacional de Acreditación en Informática y Computación, A. C. (CONAIC). En la Universidad Autónoma de Yucatán, el programa de LCC está acreditada por el CONAIC; CIES nivel 1; y Consejo para la Acreditación de la Educación Superior (COPAES). En la UNAM, el programa de LCC es catalogada como alta demanda, por lo que a aquéllos aspirantes que quedaron fuera les recomienda como alternativa institucional a la UABC, a la Benemérita Universidad Autónoma de Puebla y a la Universidad Autónoma de Yucatán. En la UNAM, resalta la variedad de opciones de titulación. Mientras en la Universidad Autónoma de Nuevo León el programa de LCC está acreditada por el CIEES, así como de la Acreditación, agencia evaluadora de Chile. En el plan de estudios se observan grupos de asignaturas de matemáticas, estadística y probabilidad, liderazgo y emprendimiento en los negocios y empresas, tecnologías computacionales, programación, redes, ciberseguridad, y diversos laboratorios. La Universidad de Guadalajara ofrece la Ingeniería en Computación, tiene el objetivo de formar ingenieros capaces de adaptar e incorporar los avances científicos y tecnológicos a su entorno, con compromiso y responsabilidad social; se cursa en nueve semestres.

En el análisis comparativo de los programas internacionales destacó lo siguiente: el MIT se centran en formar a sus estudiantes con habilidades, conocimientos y creatividad para demostrar liderazgo en la solución de problemas complejos. En la Universidad de California, Berkeley dan prioridad a la formación en liderazgo técnico y la continuidad hacia los estudios de posgrado, sus docentes y estudiantes se capacitan en centros y laboratorios de investigación, tanto propios como de otras universidades y tienen una colaboración cercana con el sector empresarial. En la Universidad de Oxford les interesa vincular la teoría con la práctica, También tienen una capacitación con distinguidos expertos mundiales para dotar a sus estudiantes con habilidades de pensamiento riguroso y práctico sobre el comportamiento de los sistemas informáticos. En la Universidad de Stanford busca que sus egresados tengan un acercamiento con el sector empresarial y que tengan habilidades y conocimientos para continuar con estudios de posgrado. De igual forma, busca que sus estudiantes trabajen de forma interdisciplinaria dentro de la misma universidad. Finalmente, la Universidad de Carnegie Mellon tiene un vínculo con las empresas de tecnología de alto prestigio mundial dedicadas al desarrollo de hardware y software, por lo que sus estudiantes están formados de manera muy cercana con estas

empresas y logran posicionarse en puestos laborales. Los egresados también pueden trabajar en sus propias empresas de consultoría, en la docencia o continuar con estudios de posgrado.

El Consejo Nacional de Acreditación en Informática y Computación A. C. (CONAIC) es un organismo cuyo objetivo es evaluar con fines de acreditación aquellos programas académicos del nivel superior del área de computación, informática, y tecnologías de la información y telecomunicación (CONAIC, 2017). El CONAIC es supervisado por el Consejo para la Acreditación de la Educación Superior (COPAES). Las categorías establecidas por el CONAIC para evaluar y emitir un dictamen de acreditación, son diez: personal académico, estudiantes, plan de estudios, evaluación del aprendizaje, formación integral, servicios de apoyo para el aprendizaje, vinculación-extensión, investigación, infraestructura y equipamiento, y gestión administrativa y financiamiento (CONAIC, 2017). Actualmente, el programa de LCC de la Facultad de Ciencias es un programa acreditado por el CONAIC con vigencia al 23 de mayo del 2026 (CONAIC, 2022). El Examen General de Egreso del programa de LCC (EGEL Plus COMPU) se conforma de dos secciones y se estructura por un total de 200 ítems: (a) Disciplinar, es específica de la profesión y evalúa cuatro áreas de las ciencias computacionales: Algoritmia, Desarrollo de software de base, Desarrollo de software de aplicación y Soluciones de cómputo inteligente; y (b) Transversal de Lenguaje y comunicación, común a todas las profesiones (Ceneval, 2023).

Conclusiones generales de la evaluación externa del programa de LCC

Pertinencia social

- El uso de la inteligencia artificial está en aumento en la vida cotidiana de las personas, se requiere crear contenido digital y aplicaciones acordes a las necesidades locales y la seguridad digital es un tema pendiente. Algunas empresas requieren transformarse digitalmente, por lo que solicitarán personal especializado.
- Las Naciones Unidas promueven un crecimiento y desarrollo mundial más equitativo y sostenible.
- En México actualmente se impulsa la Estrategia Nacional Digital 2019-2024. A nivel estatal, se indica la necesidad de crear plataformas educativas digitales para capacitar a los docentes de educación básica; crear infraestructura y equipamiento para desarrollar tecnología de punta; lograr una mayor vinculación universidad-empresas en proyectos de innovación tecnológica.
- Los empresarios destacaron un grupo de habilidades que requieren mejorarse hacia el 2025: pensamiento crítico y análisis, resolución de problemas, autogestión y aprendizaje activo, resiliencia, tolerancia al estrés y flexibilidad.
- Los cargos altamente demandados en el mercado laboral internacional serán: data analysts, software developers, cyber security experts, infraestructura architects, CRM specialists, automation and cloud experts, IT project managers.
- En México, los profesionistas ocupados con licenciatura en ciencias computacionales en 2023 sumaron 747,005 (71.8% hombres y 28.2% mujeres) y ganaron un promedio mensual de \$14,876° pesos.
- Entre los empleos altamente demandados serán para especialistas en inteligencia artificial, analista de seguridad de la información, directores de proyectos y científicos de datos.
- Las tendencias en contrataciones en el sector privado serán hacia las pequeñas y medianas empresas; y en el sector público serán por honorarios y temporales.
- Las empresas de consultoría se colocan dentro de las 17 actividades más importantes en México, destacan las actividades relacionadas con el programa de

LCC: diseño de sistemas, y procesamiento electrónico de información, hospedaje y otros servicios.

- Las certificaciones se acentúan como una tendencia de capacitación y actualización en el área de las TIC.
- Los empleadores recomiendan aplicar los conocimientos en el ambiente laboral, evitar dividir el plan de estudios y enfocarse en una sola área, desarrollar habilidades de liderazgo, trabajo en equipo, cumplimiento de metas, ética, lógica, valores, responsabilidad; así como mejorar materias dedicadas al desarrollo de software.
- El 100% de los egresados del programa de LCC tienen empleo, la mayoría gana más de \$20,000 pesos al mes; sus actividades laborales se alinean con su formación profesional, aunque requieren mayor formación en emprendimiento.
- La oferta de programas iguales a el programa de LCC se ubicó en ocho universidades públicas. Mientras los programas afines se ofertaron en 22 universidades de México.
- La matrícula nacional del campo detallado de formación en Ciencias Computacionales aumentó un 33.04% durante cinco años.
- La oferta internacional se encuentra de la siguiente manera: en España, se oferta la Ingeniería en Informática; en Argentina, el programa de LCC se ofertó en ocho universidades de sostenimiento público; en Chile, el programa de LCC se ofertó en dos; en Colombia, se identificaron 29 universidades que ofertaron la Ingeniería en Sistemas Computacionales.
- La demanda vocacional del campo profesional científico experimental en Baja California ocupó el décimo lugar en las categorías de preferencia Muy alta y Alta. El programa de LCC fue seleccionada como primera opción por el 2.3% de la población encuestada (n=388), y la segunda de mayor preferencia en este campo de estudio.

Referentes estratégicos

- Los avances tecnológicos hacia el 2030 generarán el mayor progreso tecnológico sin precedentes en los últimos 100 años.

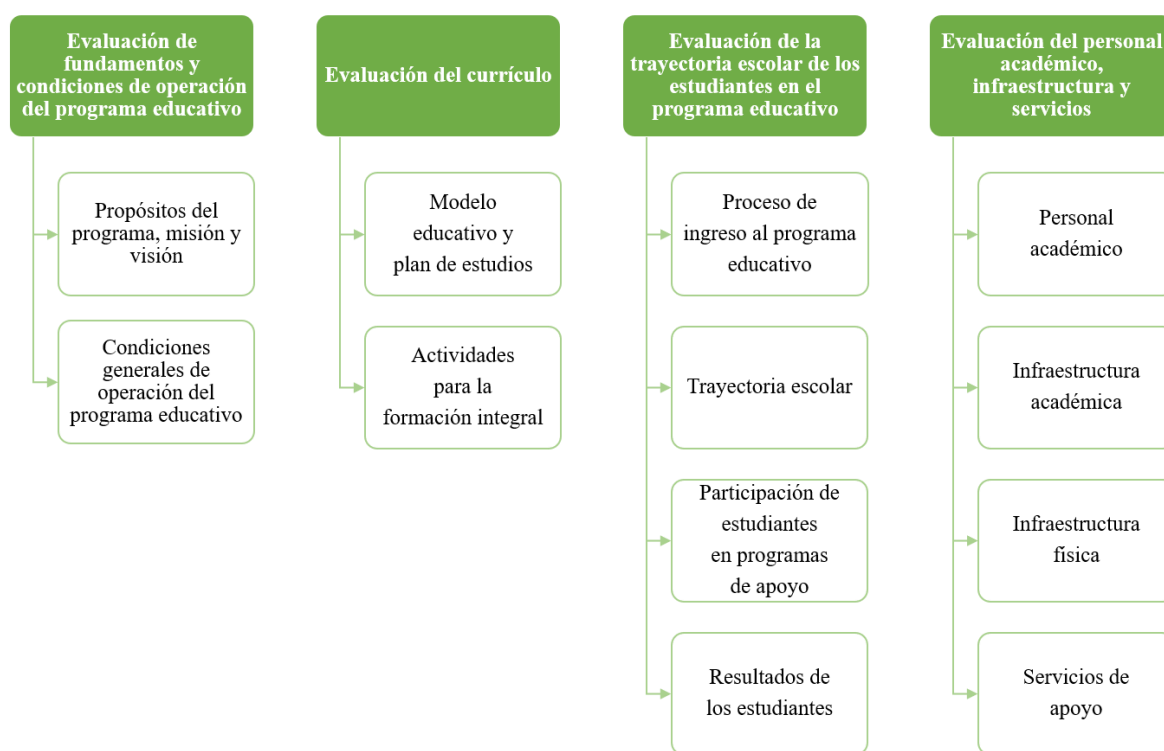
- Las profesiones inmersas en el área de las ciencias computacionales se caracterizan por la rápida innovación, la colaboración internacional, la movilidad laboral y la necesidad de adaptar y crear cuestiones éticas y sociales a escala mundial.
- Las prácticas de la profesión que realizan los investigadores y profesionistas del área de sistemas computacionales se clasifican en dos grupos: (a) Desarrolladores y analistas de software y multimedia, y (b) Administradores de bases de datos y redes de computadora.
- Las profesiones afines al programa de LCC con las que comparte su ejercicio y práctica profesional son cuatro: Licenciatura en informática, Ingeniería de software, la Ingeniería en ciencias computacionales y Licenciatura en ciencias computacionales.
- El Instituto Politécnico Nacional (IPN) en 1965 y el Instituto Tecnológico y de Estudios Superiores de Monterrey en 1968, ofrecieron las primeras carreras del área de computación.
- La prospectiva del programa de LCC señala que la transformación de la industria del siglo XXI presentará una evolución tecnológica continua. La rápida evolución de la industria tecnológica abre las oportunidades de desarrollo industrial para nuestro país, y se espera la colaboración entre la industrial, el gobierno y la academia
- Hacia el 2027, las habilidades que las empresas esperan aumenten, dada su importancia en empleo serán las cognitivas como el pensamiento creativo y el pensamiento analítico, así como la literacidad tecnológica.
- Las mejores prácticas de las IES con mayor prestigio en el ámbito de la formación profesional en ciencias computacionales pueden ser de interés para el programa de LCC de la UABC.
- Los lineamientos de los organismos acreditadores y evaluadores en el ámbito de las ciencias computacionales también sirven como referente de mejora para el programa de LCC de la UABC.

Sección II. Evaluación interna del programa educativo de Licenciado en Ciencias Computacionales

Introducción

La Evaluación interna del programa educativo de Licenciado en Ciencias Computacionales (LCC) se conforma de cuatro estudios: 2.1 Evaluación de fundamentos y condiciones de operación del programa educativo, 2.2 Evaluación del currículo, 2.3 Evaluación de la trayectoria escolar de los estudiantes en el programa educativo y 2.4 Evaluación del personal académico, infraestructura y servicios. Cada estudio compromete categorías de análisis particulares, tal como se muestra en la Figura 2.1.

Figura 2.1 Estructura de los estudios de la Evaluación interna del programa educativo de LCC



Nota. De “Metodología de los estudios de fundamentación para la creación, modificación y actualización de programas educativos de licenciatura”, por A. Serna, y A. Castro, 2018, Universidad Autónoma de Baja California. CC BY-NC-ND.

Estudio 2.1. Evaluación de fundamentos y condiciones de operación del programa educativo

Objetivo

El objetivo general de este estudio fue evaluar los fundamentos y condiciones de operación del programa educativo para fundamentar la modificación o actualización del programa de LCC. Así, se abordaron dos categorías: 2.1.1 Propósitos del programa, misión y visión y 2.1.2 Condiciones generales de operación del programa educativo (ver Tabla 2.1).

Tabla 2.1

Elementos estructurales de la Evaluación de fundamentos y condiciones de operación del programa educativo

Categoría	Objetivo específico	Indicador
2.1.1 Propósitos del programa, misión y visión	Analizar los propósitos, misión y visión del programa educativo.	2.1.1.1 Misión, visión y objetivos del programa educativo
		2.1.1.2 Perfil de ingreso
		2.1.1.3 Perfil de egreso
2.1.2 Condiciones generales de operación del programa educativo	Analizar las condiciones generales de operación del programa educativo.	2.1.2.1 Matrícula total y de primer ingreso
		2.1.2.2 Presupuesto/recursos del programa educativo
		2.1.2.3 Estructura organizacional para operar el programa educativo

Nota. De “Metodología de los estudios de fundamentación para la creación, modificación y actualización de programas educativos de licenciatura”, por A. Serna, y A. Castro, 2018, Universidad Autónoma de Baja California. CC BY-NC-ND.

Método

Este apartado se realizó una investigación de tipo documental. Para alcanzar los objetivos planteados se revisaron fuentes secundarias como Modelo Educativo de la UABC, Propuesta de modificación del programa educativo Licenciatura en Ciencias Computacionales 2017-1, Autoevaluación del Programa de Licenciado en Ciencias Computacionales para la Acreditación de Programas Académicos de Informática y Computación en Educación Superior (CONAIC, 2020); Cuarto Informe de Actividades 2019-2023 (UABC, 2022); la página del sistema integral de Información Institucional, e-

trayecto e indicadores de trayectoria escolar (UABC, 2023), así como el Sistema de Indicadores y Estadísticas Institucionales (UABC, 2021).

Resultados

2.1.1 Propósitos del programa, misión y visión

El objetivo específico es analizar los propósitos, misión y visión del programa educativo.

Misión, visión y objetivos del programa educativo

La Misión de la UABC es la siguiente:

Contribuir al desarrollo inclusivo y sostenible y al bienestar de la sociedad bajacaliforniana, la nación y del planeta, a través de la formación integral de profesionistas, de investigadoras, investigadores y ciudadanía comprometida con una cultura democrática; así como a la generación y difusión de la cultura, del conocimiento y de las tecnologías.

La Visión es:

En 2040, la UABC es una universidad líder e innovadora con reconocimiento a nivel nacional y global por ser un agente transformador de progreso social que contribuye a la realización plena del ser humano, al desarrollo incluyente y sostenible, a la profundización de la democracia y a la justicia social a través de la formación de licenciatura, posgrado y a lo largo de la vida, y de sus avances científicos, tecnológicos y culturales.

Aunado a la Misión y la Visión, la UABC señala cinco prioridades:

1. *Aprendizaje integral, flexible y a lo largo de la vida.* Brindar un aprendizaje integral, flexible y de excelencia en los programas de licenciatura, posgrado y aprendizaje a lo largo de la vida.
2. *Investigación e innovación.* Expandir el conocimiento y generar soluciones innovadoras a través de la investigación —en especial interdisciplinaria— para enfrentar los retos locales y globales de la humanidad y del planeta.

3. *Bienestar de la comunidad universitaria.* Contribuir al bienestar de la comunidad universitaria con equidad, inclusión y respeto a la diversidad.
4. *Desarrollo regional e internacionalización.* Fortalecer actividades de vinculación y extensión que contribuyan al desarrollo social, cultural, económico y ambiental a nivel regional, nacional, transfronterizo y global para un mundo mejor.
5. *Gestión y financiamiento.* Implementar una gestión oportuna, flexible, transparente y sostenible que brinde soporte y asegure el financiamiento de las funciones sustantivas de la universidad (UABC, 2024).

La Facultad de Ciencias (FC) del Campus Ensenada comenzó operaciones a partir del 21 de junio de 1977 y en 1986 inicia el programa de Licenciado en Ciencias Computacionales (LCC).

La misión de la FC es:

Formar profesionistas de excelencia y de alto nivel competitivo, capaces de aplicar sus conocimientos y habilidades para enfrentar y resolver los retos propios al entorno científico, actual y futuro. La Facultad de Ciencias impulsa la investigación científica en sus diferentes áreas. Además, busca generar conocimiento y extenderlo a la comunidad, llevándolo a su aplicación en el ámbito científico, académico y social. Todo esto con la finalidad de mejorar la calidad de vida en el entorno local, regional, nacional e internacional. Finalmente, fomenta a cada paso los valores culturales, el sentido ético, la responsabilidad social y el respeto al medio ambiente, a través de la extensión de la cultura, la difusión y divulgación del conocimiento.

La visión de la FC es:

Ser una institución reconocida por la competitividad y formación integral de sus egresados, destacada por la calidad y vigencia de sus planes de estudio que responden a la demanda del ámbito científico de los diversos sectores de la sociedad y de la industria en el entorno local y nacional. En la actividad docente, busca tener niveles elevados de formación del personal académico, con un incremento del número de docentes con grado de maestría y doctorado, aprovechando la

vinculación y convenios con sectores dedicados a la ciencia para la formación y actualización constante de los académicos, lo que tiene un impacto favorable en los estudiantes, contribuyendo así a la producción científica e intelectual con base en el nuevo modelo educativo. Al estudiante de la Facultad de Ciencias, además de su valor competitivo, lo destaca su valor que le asigna a la conducta ética, el esfuerzo permanente y el compromiso para responder a las cambiantes realidades de su entorno (UABC, 2021f, p. 115).

En lo que se refiere a la Misión del programa de LCC se enuncia lo siguiente:

El programa de Licenciado en Ciencias Computacionales tiene la misión de formar profesionales con un alto nivel de conocimientos científicos, tecnológicos y sentido de responsabilidad social, comprometido con su entorno, capaz de identificar problemas y ofrecer soluciones, promoviendo la aplicación ética y responsable de las tecnologías de la información y la comunicación en beneficio del Estado y del País (CONAIC, 2020).

La Visión señala:

El programa de Licenciado en Ciencias Computacionales tiene la visión de ser un programa acreditado, atendido por un conjunto de académicos asociados a Cuerpos Académicos consolidados, los cuales están permanentemente actualizando los planes y programas de estudio, integrando en las diferentes asignaturas los nuevos conocimientos y estrategias de aprendizaje resultado de investigaciones científicas y tecnológicas. Así, académicos, estudiante y egresados conforman profesionales multi e interdisciplinarios que promueven el uso eficaz de recursos humanos y materiales existentes e impulsan y participan en programas de movilidad institucional. Todo lo anterior les otorga a los grupos así integrados y organizados una visión holística sobre los retos, alcances y oportunidades de su profesión en un contexto regional, nacional e internacional (CONAIC, 2020, p. 8).

Los objetivos del programa educativo del programa de LCC son:

1. Formar profesionistas de las ciencias computacionales dedicados a gestionar y ofrecer soluciones innovadoras en diversos entornos, derivadas del análisis,

modelado, diseño, implementación y evaluación de proyectos complejos para la automatización de procesos que pueden ser tratados con sistemas de cómputo y asegurando la integridad de la información con alto impacto en el mercado laboral en los ámbitos local, nacional e internacional, con actitud emprendedora y capaces de transformar su entorno con responsabilidad y compromiso ético.

2. Promover la vinculación y colaboración con los sectores público, social y empresarial para la realización de proyectos innovadores de investigación y desarrollo tecnológico que contribuyan al progreso en el área de las ciencias computacionales, así como a incrementar el nivel de avance de la sociedad en los contextos nacional y global, procurando un equilibrio entre la generación del conocimiento básico, el económicamente relevante y el socialmente útil.

(CONAIC, 2020, p. 8)

Con base en la misión, visión y los objetivos del programa de LCC, la facultad se ha comprometido a desempeñar un trabajo de calidad y mejora continua, congruente con los objetivos de la educación superior y con los desarrollos actuales y futuros de la computación (CONAIC, 2020). Conjuntamente, toma en cuenta la misión y visión de la UABC y de su propia FC. Además, con el fin de garantizar su cumplimiento, se realizan actualizaciones periódicas a sus planes y programas, fundamentados en los avances científicos y en la evaluación del programa por parte de organismos acreditadores externos a la UABC.

Con la finalidad de garantizar el cumplimiento de la misión y la visión del programa de LCC, se articulan dos grandes objetivos que convergen en el perfil de egreso y en el plan de estudios, los cuales se desglosarán en los siguientes subapartados.

Perfil de ingreso

Los conocimientos, habilidades, actitudes y valores del aspirante a ingresar al programa de LCC se presentan en la Tabla 2.2.

Tabla 2.2

Elementos del perfil de ingreso al programa de LCC

Conocimientos	Habilidades	Actitudes y valores
1) Aritmética	1) Pensamiento matemático y analítico	1) Perseverancia y disciplina
2) Álgebra		2) Capacidad de organización
3) Geometría	2) Manejo de las estructuras de la lengua y comprensión lectora	3) Disposición para trabajar en equipo
4) Cálculo	3) Expresarse adecuadamente de forma oral y escrita	4) Inquietud por lograr independencia intelectual
	4) Administrar de forma adecuada tiempos y actividades	5) Sentido crítico y reflexivo
	5) Manejar tecnologías de cómputo	6) Iniciativa y propositivismo
		7) Capacidad para el trabajo bajo presión

Nota. De “Licenciatura en Ciencias Computacionales. Plan de estudios 2017-1”, por Universidad Autónoma de Baja California, 2023. CC BY-NC-ND

Entre las estrategias implementadas por la UABC para asegurar el cumplimiento de los atributos del perfil de ingreso, se ubica el estudio vocacional, mismo que fue aplicado a estudiantes de educación media superior en Baja California. A través de este se conocieron los intereses profesionales, aptitudes, y habilidades de quienes en algún momento pudieran ingresar a la universidad (UABC, 2020).

Asimismo, el Examen de Ingreso a la Educación Superior (ExIES), mide los conocimientos y habilidades de los aspirantes a ingresar a la UABC en tres áreas: lectura, matemáticas y lengua escrita. El puntaje mínimo establecido para inscribirse al programa de LCC es de 700 puntos. De igual manera, se reporta la aplicación de entrevistas y encuestas a los estudiantes de nuevo ingreso (CONAIC, 2020).

Al momento de ingresar a la UABC, se utiliza el Examen Diagnóstico del Idioma Inglés (Exadii), con la finalidad de evaluar el nivel de dominio del inglés al en tres áreas: comprensión lectora, redacción y comprensión auditiva (UABC, 2022).

Perfil de egreso

El perfil de egreso del programa de LCC declara que:

El Licenciado en Ciencias Computacionales es un profesional dedicado a gestionar y ofrecer soluciones innovadoras en diversos entornos, derivadas del análisis, modelado, diseño, implementación y evaluación de proyectos complejos para la automatización de procesos que pueden ser tratados con sistemas de cómputo y asegurando la integridad de la información. Ello implica que el Licenciado en Ciencias Computacionales egresado sea competente para: Gestionar y desarrollar software aplicando metodologías acordes a estándares nacionales e internacionales para satisfacer las demandas del mercado con sentido ético, visión empresarial y responsabilidad social; Administrar los recursos de cómputo y telecomunicaciones, utilizando los protocolos y estándares de comunicación para garantizar la integridad de la información en las organizaciones con sentido de responsabilidad profesional; y, Diseñar y desarrollar soluciones computacionales que modelen y simulen procesos de las Ciencias, mediante algoritmos y modelos matemáticos que permitan analizar el comportamiento de fenómenos complejos, con creatividad e innovación (UABC, 2023).

Con la finalidad de alcanzar el perfil de egreso del programa de LCC, el plan de estudios contempla un total de 350 créditos. El plan de estudios del programa de LCC se divide en tres etapas: Básica, Disciplinaria y Terminal. En la etapa básica los estudiantes adquieren orientación formativa, que les permita emplear conocimientos y herramientas básicas de las ciencias computacionales; mientras en la etapa disciplinaria obtienen los conocimientos teóricos y prácticos de la profesión; por último, en la etapa terminal refuerzan los conocimientos teórico-instrumentales específicos de las ciencias computacionales (ver tabla 2.3).

Tabla 2.3

Competencias de las Etapas Básica, Disciplinaria y Terminal del programa de LCC

Etapa/Periodos escolares	Orientación	Créditos	Competencias
Básica (I, II y III)	Formativo	131 (112 de UA obligatorias y 19 de optativas)	Emplear los conocimientos y herramientas básicas de las ciencias computacionales a través de la construcción de algoritmos y razonamiento lógico para la solución de problemas, con actitud creativa y

			pensamiento crítico.
Disciplinaria (IV, V y VI)	Genérico	134 (110 de UA obligatorias y 24 de optativas)	Aplicar las herramientas tecnológicas de cómputo actuales y las bases del cómputo científico a través del desarrollo y documentación de programas de cómputo de baja complejidad, para brindar los conocimientos teóricos y prácticos de su profesión, con actitudes responsables que garanticen la confiabilidad y seguridad de la información.
Terminal (VII y VIII)	Teórico-instrumental específico	85 (33 de UA obligatorias y 42 de optativas, 10 de prácticas profesionales).	Evaluar tecnologías de cómputo avanzadas a través de la implementación de soluciones innovadoras para atender problemas reales susceptibles a ser automatizadas, con responsabilidad y trabajo en equipo multidisciplinario.

Asimismo, el plan de estudios integra diversas modalidades de aprendizaje como asignaturas y actividades, mismas que en conjunto tienen la finalidad de alcanzar el perfil de egreso: unidades de aprendizaje obligatorias y optativas, cursos intersemestrales, apoyo a actividades de extensión y divulgación, proyectos de vinculación con valor en créditos (PVVC), actividades artísticas, culturales y deportivas, prácticas profesionales, programa de emprendedores universitarios, actividades para la formación en valores, movilidad e intercambio, servicio social, acreditación de idioma extranjero y diversas opciones de titulación. En la tabla 2.4 se pueden observar los objetivos de cada una de estas.

Tabla 2.4

Modalidades de aprendizaje y objetivos

Modalidad de aprendizaje	Objetivo
Unidades de aprendizaje obligatorias	Lograr el perfil de egreso, definidas y organizadas en función de las competencias genéricas, específicas y profesionales que conforman dicho perfil.
Unidades de aprendizaje optativas	Fortalecer el proyecto educativo de los estudiantes en áreas específicas: ambientes inteligentes, cómputo visual, Big Data y TIC.
Otros cursos optativos	Incorporar UA acordes a los avances científicos y tecnológicos.
Apoyo a actividades de extensión y vinculación	Fomentar relaciones universidad-comunidad.
Programas de Vinculación con Valor en Créditos (PVVC)	Fortalecer el logro de competencias específicas en los estudiantes, a través de la aplicación y generación de conocimiento, y solución de problemas,.

Modalidad de aprendizaje	Objetivo
Actividades artísticas, culturales y deportivas	Desarrollar habilidades que contribuyen con la formación integral.
Prácticas profesionales	Aplicar conocimientos teóricos en la práctica, adquirir experiencia profesional requerida al egreso y vincular universidad-sectores público y privado.
Programa de emprendedores universitarios	Fomentar los proyectos innovadores, mediante la formulación de un plan de negocios, orientación para apoyo financiero y su validación académica.
Actividades para la formación en valores	Contribuir a la formación de valores éticos y profesionales a través de visitas, foros, pláticas, entre otros).
Cursos intersemestrales	Cubrir créditos y avanzar en el plan de estudios.
Movilidad e intercambio	Cursar UA, Prácticas profesionales o actividades académicas en otras IES nacionales o extranjeras, para adquirir nuevas competencias: maduración, autonomía, maduración de los estudiantes.
Servicio social	Primera etapa (Servicio social comunitario), Fortalecer la misión social de la UABC. Segunda etapa (Servicio social profesional), fortalecer la formación académica.
Acreditación del idioma extranjero	Acreditar el conocimiento de un segundo idioma, esencial en la formación de los estudiantes.
Titulación	Indicador de calidad y eficiencia del programa educativo.

Nota. De “Propuesta de Modificación del Programa de Licenciado en Ciencias Computacionales 2017-1”, UABC, 2017.

2.1.2. Condiciones generales de operación del programa educativo

Objetivo específico. Analizar las condiciones generales de operación del programa educativo.

Matrícula total y de primer ingreso

Con base en la información disponible en el Sistema de indicadores y estadísticas institucionales, del período escolar de 2019-2 a 2023-2 la matrícula total del programa de LCC campus Ensenada fue de 469 estudiantes, 422 de sexo masculino (90%) y 47 de sexo femenino (10%). El grupo de edad de mayor frecuencia fue de 20 a 24 años con 367 estudiantes (78.2%), y el lugar de procedencia fue principalmente de Baja California con 420 estudiantes (89.5%). En términos generales, en el periodo señalado, la matrícula aumentó 152%. Se registraron algunas fluctuaciones, por ejemplo: entre el periodo escolar 2021-2 al 2022-1 hubo un decremento de la matrícula del 10.8% y entre el 2023-1 al 2023-2 del 10.0%. Sin embargo, el periodo 2021-1 al 2021-2 el aumento en la matrícula fue del

32%; asimismo entre el período 2021-2 al 2022-1 hubo un incremento del 30.0% (ver Tabla 2.5).

Tabla 2.5

Indicadores y estadísticas de estudiantes del programa de LCC

Periodo	Sexo				Grupo de edad												Ingreso				Procedencia						Total ^a		
	Hombres		Mujeres		18-19		20-24		25-29		30-34		35-39		40- ^b		18 ^c		Ni ^d		Ri ^e		BC ^f		Oe ^g			Op ^h	
	F ⁱ	%	F	%	F	%	F	%	F	%	F	%	F	%	F	%	F	%	F	%	F	%	F	%	F	%		F	%
Campus Ensenada																													
2023-2	56	88.9	7	11.1	5	7.9	50	79.4	7	11.1	1	1.6	0	0	0	0	0	0	0	0	63	100	55	87.3	8	12.7	0	0	63
2023-1	62	88.6	8	11.4	5	7.1	54	77.1	8	11.4	1	1.4	2	2.9	0	0	0	0	0	0	70	100	63	90.0	7	10.0	0	0	70
2022-2	57	86.4	9	13.6	6	9.1	50	75.8	7	10.6	1	1.5	2	3.0	0	0	0	0	0	0	66	100	60	90.9	6	9.1	0	0	66
2022-1	54	93.1	4	6.9	2	3.5	47	81.0	6	10.3	1	1.7	2	3.5	0	0	0	0	0	0	58	100	52	89.7	5	8.6	1	1.7	58
2021-2	59	90.8	6	9.2	5	7.7	52	80.0	4	6.1	1	1.5	3	4.6	0	0	0	0	0	0	65	100	58	89.2	5	7.7	2	3.1	65
2021-1	45	90.0	5	10.0	0	0	39	78.0	8	16.0	1	2.0	2	4.0	0	0	0	0	0	0	50	100	44	88.0	6	12.0	0	0	50
2020-2	37	90.0	4	10.0	0	0	32	78.0	5	12.2	2	4.9	2	4.9	0	0	0	0	0	0	41	100	39	95.0	2	5.0	0	0	41
2020-1	29	93.5	2	6.5	0	0	24	77.4	4	12.9	1	3.2	2	6.5	0	0	0	0	0	0	31	100	27	87.1	3	9.7	1	3.2	31
2019-2	23	92.0	2	8.0	0	0	19	76.0	5	20.0	0	0	1	4.0	0	0	0	0	0	0	25	100	22	88.0	2	8.0	1	4.0	25
Total ^j	422		47		23		367		54		9		16		0		0		0		469		420		44		5		469
	469										469								469						469				

Nota. ^a Total de frecuencias por periodo. ^b 40 y más. ^c Menos de 18. ^d Ni=Nuevo ingreso. ^e Ri=Reingreso. ^f Baja California. ^g Oe=Otra entidad. ^h Otro país. ⁱ F=Frecuencia. ^j Total de frecuencias por clasificación. Las celdas vacías corresponden a la ausencia de datos. De “Sistema de indicadores y estadísticas institucionales”, por Universidad Autónoma de Baja California, 2023. CC BY-NC-ND.

Presupuesto/recursos del programa educativo

El personal académico participa en la planeación del programa educativo, misma que se refleja en el Plan de Desarrollo de la unidad académica, en donde se enlistan los gastos requeridos como movilidad, compra de materiales, mantenimiento, cursos y talleres. El plan presupuestal se deriva del presupuesto anual de la UABC, y se integra considerando el presupuesto y el ejercicio del gasto del año anterior. Dicho plan es sometido a revisión por la Unidad de Presupuestos y Finanzas para verificar su apego al marco legal, finalmente se somete votación para su autorización por el H. Consejo Universitario (CONAIC, 2020).

Los ingresos provienen de diversas fuentes como son las cuotas pagadas por los estudiantes (cuotas específicas, formación integral, pro-desarrollo y prácticas de campo), pago de cursos intersemestrales durante verano e invierno, el Sorteo UABC, y recursos de fondos extraordinarios como el Programa de Fortalecimiento a la Excelencia Educativa (PROFEXCE), cuyo objetivo es otorgar apoyos económicos para implementar proyectos académicos encaminados a mejorar la calidad de los programas educativos; el Programa para el Desarrollo Profesional Docente (PRODEP), que contribuye a mejorar la infraestructura y la calidad de los programas; las Convocatorias del Consejo Nacional de Ciencia y Tecnología, y la Convocatoria interna de proyectos. Sin embargo, los apoyos del PROFEXCE y el PRODEP, se recibieron hasta el año 2020, situación que ha impactado los rubros que se impulsaban a través de estos (UABC, 2023).

En el año 2020 se aprobó un presupuesto de operación de \$167,022.00 para compra de materiales de oficina, procesamiento, gastos de conservación de equipo, transporte, pago de eventos y becas económicas para estudiantes. El programa cuenta con criterios establecidos para la determinación de gastos de mantenimiento y operación de laboratorios y talleres; este mantenimiento se realiza en coordinación con el Departamento de Servicios Administrativos de la UABC (CONAIC, 2020).

Estructura organizacional para operar el programa educativo

La estructura organizacional de la FC se elabora con base en los propósitos compartidos con la UABC; asimismo, la organización del capital humano, los objetivos académicos y financieros responden a la operación de procesos y procedimientos institucionales en atención a la oficina de Planeación y Desarrollo Institucional (CONAIC, 2020; UABC, 2022).

Tal como puede observarse en la Tabla 2.6, la estructura organizacional que permite realizar el trabajo requerido para la adecuada operación en la Facultad de Ciencias es a través de un Director, una Subdirectora, un Administrador, 18 Docentes del Programa de LCC, tres Coordinaciones, 10 Responsables de área, cinco secretarías, un Auxiliar de laboratorio, dos Almacenista, un Encargado de fotocopiado, un Chofer de autobús y seis Conserjes. Es importante señalar que, el personal administrativo de la FC trabaja de manera conjunta para brindar servicio a los programas de Licenciatura en Matemáticas Aplicadas, Licenciatura en Física y Licenciado en Ciencias Computacionales. Los 18 docentes mencionados son exclusivamente del programa de LCC, y realizan diversas actividades adicionales a las de enseñanza-aprendizaje, incluido el actual Director de la FC, quien pertenece a este programa educativo. Cabe señalar que, las actividades a desarrollar por el Responsable de Tutorías no se indican en el Manual Operativo y se encuentran en la Propuesta de Modificación del Programa de LCC 2017-1.

Tabla 2.6

Estructura organizacional de la Facultad de Ciencias

Nombre del Cargo	Cantidad	Finalidad
Director	1	Planear, coordinar, y supervisar las actividades del personal en las áreas de docencia, investigación y difusión cultural, y administrar adecuadamente los recursos
Subdirectora	1	Coordinar y controlar las actividades del personal, y verificar el cumplimiento del plan de estudios.
Personal Docente LCC	18	Aplicar adecuadamente las actividades de enseñanza-aprendizaje y promover la atención integral de los estudiantes.
Secretarías	5	Realizar actividades requeridas para el funcionamiento de la FC.
Administrador	1	Programar, organizar, integrar, dirigir, y controlar las actividades del personal a su cargo, y realizar los trámites necesarios ante las dependencias de la institución.
Coordinación de Formación Profesional	1	Organizar, supervisar y verificar el funcionamiento de los programas educativos: TSU y licenciatura.
Coordinación de Extensión y Vinculación	1	Impulsar, organizar, promover y supervisar actividades de extensión de la cultura, divulgación, vinculación y cooperación académica de la FC.
Coordinación de Investigación y Posgrado	1	Organizar, supervisar y verificar el cumplimiento de actividades de investigación, posgrado y propiedad intelectual de la FC.
Responsable del Programa Educativo	1	Coordinar, controlar las actividades del personal docente de su área.
Responsable de Laboratorio	12	Programar, coordinar, administrar y gestionar las actividades del laboratorio.

Nombre del Cargo	Cantidad	Finalidad
Responsable de Orientación Educativa y Psicopedagógica	1	Planear, organizar y desarrollar la atención a los aspirantes y estudiantes de nuevo ingreso; otorgar atención psicológica y pedagógica a los estudiantes de la FC; asesorar pedagógicamente a los docentes; y elaborar información de las actividades realizadas.
Responsable de Servicio Social	1	Coordinar y supervisar a estudiantes que solicitan información y asesoría para tramitar, desarrollar y liberar el servicio social.
Responsable de Prácticas Profesionales	1	Coordinar y supervisar a estudiantes que solicitan información y asesoría para tramitar, desarrollar y liberar las prácticas profesionales.
Responsable de Movilidad Estudiantil	1	Coordinar y fomentar el intercambio del personal docente y estudiantes, y verificar el cumplimiento de los planes de estudio.
Responsable de Difusión	1	Coordinar actividades de difusión, para compartir a la sociedad actividades de gestión, investigaciones académicas y de extensión de la FC.
Responsable de Formación y Evaluación Docente	1	Organizar, evaluar y vigilar las actividades de formación, actualización, capacitación, desarrollo y coordinar el proceso de evaluación docente.
Responsable de Seguimiento de Egresados	1	Establecer una relación recíproca, permanente y de vinculación entre la FC y los egresados.
Responsable de Laboratorio de Cómputo	1	Administrar la red de cómputo, equipo de cómputo y sistemas, asegurar su funcionamiento y asesorar a docentes y personal administrativo en su manejo.
Auxiliar de Laboratorio	1	Preparar el material de prácticas de laboratorio y asesorar a estudiantes en el manejo del material y equipo.
Almacenista	2	Atender los requerimientos de los programas académicos respecto al suministro de materiales.
Encargado de Fotocopiado y material audiovisual	1	Atender los servicios de fotocopiado y administración de equipo audiovisual para apoyar el proceso de enseñanza-aprendizaje.
Chofer de Autobús	1	Proporcionar el servicio de transporte al personal y estudiantes de la FC.
Conserje	6	Mantener limpios los edificios, oficinas, mobiliario y equipo.

Nota. De “Manual organizacional y procedimientos. Facultad de Ciencias”, por UABC 2021e.

Conclusiones Estudio 2.1. Evaluación de fundamentos y condiciones de operación del programa educativo

Después de analizar los fundamentos y condiciones de operación del programa, y con apego a la Metodología de Serna y Castro (2018), se concluye que el programa de Licenciado en Ciencias Computacionales cuenta con una misión y visión, mismas que desembocan en dos objetivos. En

conjunto, la misión, visión y objetivos se articulan de manera coherente. Para garantizar su cumplimiento se realizan actualizaciones periódicas a sus planes y programas, los cuales son sometidos a evaluación por parte de acreditadores externos.

Los aspirantes interesados en cursar el programa de LCC deben contar con conocimientos sobre aritmética, álgebra, geometría, y cálculo; así como habilidades en el manejo de las tecnologías de la información, comprensión lectora, expresión oral y escrita, y administración del tiempo; con actitud hacia el trabajo en equipo, perseverancia, independencia intelectual, sentido crítico, iniciativa y capacidad para trabajar bajo presión. Con la finalidad de asegurar su cumplimiento, la UABC estableció diversos mecanismos como el Estudio vocacional en educación media superior en Baja California, la aplicación del ExIES al ingreso, y el Examen Diagnóstico del Idioma Inglés (Exadii).

Para alcanzar el perfil de egreso, el plan de estudios contempla un total de 350 créditos distribuidos en unidades de aprendizaje obligatorias y optativas, prácticas profesionales, servicio social; además incluye proyectos de vinculación con valor en créditos, movilidad e intercambio.

En cuanto a las condiciones generales de operación del programa de LCC, la matrícula total de programa durante los periodos escolares 2019-2 al 2023-2 fue de 469 estudiantes y tuvo su mayor aumento (32.2%) entre el periodo escolar 2020-1 y 2020-2, y un decremento (10.8%) entre el periodo 2021-2 y 2022-1.

Concerniente al presupuesto destinado al programa de LCC, la cantidad asignada es para compra de materiales de oficina, procesamiento, conservación de equipo, transporte, eventos y becas de estudiantes. En lo que se refiere a gastos de mantenimiento y operación de laboratorios y talleres, este gasto se coordina con el Departamento de Servicios Administrativos de la UABC. La estructura organizacional de la Facultad de Ciencias se elabora con base en los propósitos compartidos con la UABC y su oficina de Planeación y Desarrollo Institucional (CONAIC, 2020). Los ingresos provienen de diversas fuentes, como cuotas pagadas por los estudiantes, Sorteo UABC, recursos de fondos extraordinarios como el PROFEXE y PRODEP; sin embargo, estos dos últimos dejaron de recibirse a partir del año 2020, situación que afectó los rubros que se impulsaban a través de los mismos (UABC, 2023).

Finalmente, la estructura organizacional de la FC permite compartir al personal administrativo quienes ofrecen servicio a los programas educativos de Licenciatura en Matemáticas

Aplicadas, Licenciatura en Física, Licenciado en Ciencias Computacionales y Licenciado en Ciencia de Datos.

Estudio 2.2. Evaluación del currículo

Objetivo

El objetivo general de la evaluación del currículo del programa educativo de LCC fue evaluar el currículo específico y genérico del programa educativo para fundamentar su modificación o actualización. La estructura se conformó de dos categorías: 2.2.1 Modelo educativo y plan de estudios y 2.2.2 Actividades para la formación integral (ver Tabla 2.7).

Tabla 2.7

Descripción de elementos estructurales de la Evaluación del currículo

Categoría	Objetivo específico	Indicador
2.2.1 Modelo educativo y plan de estudios	Analizar el plan de estudios, el mapa curricular, las asignaturas o UA y la tecnología educativa y de la información para el aprendizaje.	2.2.1.1 Plan de estudios
		2.2.1.2 Mapa curricular
		2.2.1.3 Programas de unidades de aprendizaje
		2.2.1.4 Tecnología educativa y de la información para el aprendizaje
2.2.2 Actividades para la formación integral	Analizar las actividades para la formación integral de los estudiantes.	2.2.2.1 Actividades complementarias para la formación integral
		2.2.2.2 Enseñanza de otras lenguas extranjeras

Nota. De “Metodología de los estudios de fundamentación para la creación, modificación y actualización de programas educativos de licenciatura”, por A. Serna, y A. Castro, 2018, *Universidad Autónoma de Baja California*. CC BY-NC-ND.

Método

Este apartado se realizó una investigación de tipo documental. Para alcanzar los objetivos planteados se revisaron fuentes secundarias como Modelo Educativo de la UABC, Propuesta de modificación del programa educativo Licenciado en Ciencias Computacionales 2017-1, Autoevaluación del Programa de Licenciado en Ciencias Computacionales para la Acreditación de Programas Académicos de Informática y Computación en Educación Superior (CONAIC, 2020); Cuarto Informe de Actividades 2019-2023 (UABC, 2022); la página del Sistema Integral de Información Institucional, e-trayecto e indicadores de trayectoria escolar (UABC, 2023), así como el Sistema de Indicadores y Estadísticas Institucionales (UABC, 2021).

Resultados

2.2.1 Modelo educativo y plan de estudios

El objetivo específico de este apartado es analizar el plan de estudios, el mapa curricular, las asignaturas o UA y la tecnología educativa y de la información para el aprendizaje.

Plan de estudios

El plan de estudios se estructura en ocho semestres, en los dos primeros se ubica el tronco común. En este se comparten 12 unidades de aprendizaje obligatorias con Física y Matemáticas Aplicadas: Comunicación Oral y Escrita, Diseño de Algoritmos, Cálculo Diferencial, Geometría Vectorial, Álgebra Superior, Historia e Impacto de la Ciencia, Formación de Valores Introducción a la Programación, Cálculo Integral, Álgebra Lineal, Mecánica, y Métodos Experimentales. Además de las UA del Tronco común, se comparten 12 UA de carácter obligatorio para Ciencias Computacionales y de carácter obligatorio u optativo para Matemáticas Aplicadas y Física; y 12 UA de carácter optativo para Ciencias Computacionales y de carácter obligatorio u optativo para Matemáticas Aplicadas y Física.

Con la finalidad de afianzar el aprendizaje de inglés, se cuenta con tres UA compartidas con Física y Matemáticas Aplicadas, las cuales incluyen de manera explícita la formación de habilidades de comunicación oral y escrita. Además, el 20% de la bibliografía básica de todas las unidades de aprendizaje es bibliografía en inglés.

El total de créditos para cubrir el plan de estudios del programa de LCC es de 350. Estos se encuentran divididos de la siguiente forma: 265 créditos son obligatorios, 85 créditos son optativos; en total 38 UA son obligatorias y 11 son optativas.

El plan de estudios establece tres competencias generales: (a) Gestionar y desarrollar software aplicando metodologías acordes a estándares nacionales e internacionales para satisfacer las demandas del mercado con sentido ético, visión empresarial y responsabilidad social; (b) Administrar los recursos de cómputo y telecomunicaciones, utilizando los protocolos y estándares de comunicación para garantizar la integridad de la información en las organizaciones con sentido de responsabilidad profesional; (c) Diseñar y desarrollar soluciones computacionales que modelen y simulen procesos de las Ciencias, mediante algoritmos y modelos matemáticos que permitan analizar el comportamiento de fenómenos complejos, con creatividad e innovación.

Las áreas de énfasis son las que recomienda la Asociación Nacional de Instituciones de Educación en Tecnologías de la Información A.C. (ANIEI): Big Data, Ambientes Inteligentes, Cómputo Visual, Tecnologías de Información y Comunicación.

El plan de estudios del programa de LCC 2017-1 se desarrolló con apego al Modelo Educativo de la UABC. El sustento filosófico y pedagógico se basa en el humanismo, el constructivismo y la educación a lo largo de la vida. El modelo define tres atributos esenciales: la flexibilidad curricular, la formación integral; y el sistema de créditos. Todos lo anterior se ve reflejado desde la Misión, Visión, Objetivos, Competencias y en el Plan de estudios del programa de LCC. Así mismo, la estructura organizacional está definida para cubrir todas las áreas que se precisan en el Modelo educativo de la UABC.

Sin embargo, dados los avances tecnológicos que el Foro Económico Mundial advierte hacia el 2030, se espera una evolución tecnológica continua y nos encaminan hacia la Revolución digital, (Oropeza-García, 2019). Así, las profesiones inmersas en el área de las ciencias computacionales se caracterizarán por la rápida innovación, la colaboración internacional, la movilidad laboral y la necesidad de adaptar y crear cuestiones éticas y sociales a escala mundial.

Aunado a lo anterior, el CONAIC (2022) define a la licenciatura en ciencias computacionales como el profesional que estudia y desarrolla las ciencias computacionales; conceptualiza y desarrolla ambientes, facilidades y aplicaciones innovadoras de computación en diversos entornos; elabora modelos de realidades complejas con rigor teórico y práctico. Además, señala que, la licenciatura en ciencias computacionales tiene un perfil académico con capacidad para estudios de posgrado.

Es por ello que, se precisa actualizar el plan de estudios 2017-1 del programa de LCC de la UABC, para adaptarse a los requerimientos actuales y futuros de la profesión.

Mapa curricular

El Mapa curricular del programa de LCC cuenta con tres etapas de formación: Básica, Disciplinaria y Terminal (Tabla 2.8).

Tabla 2.8

Etapas de formación del programa de LCC

Etapas Básicas	Etapas Disciplinarias	Etapas Terminales
Contribuye a la formación elemental e integral estudiantil de las ciencias básicas con una orientación eminentemente formativa, para la adquisición de conocimientos de las diferentes disciplinas que promueven competencias contextualizadoras, metodológicas, instrumentales y cuantitativas.	Se tiene la oportunidad de conocer, profundizar y enriquecerse de los conocimientos teórico-metodológicos y técnicos de la profesión orientados a un aprendizaje genérico del ejercicio profesional	Se refuerzan los conocimientos teórico-instrumentales específicos; se incrementan los trabajos prácticos y se desarrolla la participación en el campo profesional, explorando las distintas orientaciones a través de la integración y aplicación de los conocimientos adquiridos para enriquecerse en áreas afines y distinguir los aspectos relevantes de las técnicas y procedimientos en la solución de problemas o generación de alternativas.

Nota. De “Licenciatura en ciencias computacionales: Perfil de egreso”, por Universidad Autónoma de Baja California, 2023. CC BY-NC-ND.

La distribución de las asignaturas por etapa y área de conocimiento se encuentra de la siguiente manera: *Matemáticas* contempla la mayor carga en la etapa básica (61.5%), seguido de la etapa disciplinar (30.8%), y etapa terminal (7.7%). Por su parte, *Arquitectura de computadoras* se ubica en su mayoría en la etapa básica (66.7%) y en la etapa disciplinar (33.3%). El área de conocimiento *Interacción humano-computadora* se sitúa en su totalidad (100%) en la etapa disciplinar. *Programación e ingeniería de software* se distribuye en las tres etapas de formación, básica (40%), disciplinaria (40%) y terminal (20%). *Redes* se ubica al final de la etapa disciplinaria (50%) e inicios de la etapa terminal (50%). Las áreas de *Software de base* y *Tratamiento de la información* se encuentran 100% en la etapa disciplinaria. Por último, Entorno social se divide entre la etapa básica (75%) y terminal (25%). En la tabla 2.9 se presenta la distribución de las asignaturas por áreas del conocimiento y etapas de formación y en la tabla 2.10 se presenta distribución de los créditos por etapa de formación y tipo de asignatura.

Tabla 2.9

Distribución de asignaturas por áreas del conocimiento del programa de LCC

Área del conocimiento/ Semestre	Asignaturas							
	Etapa Básica			Etapa Disciplinaria			Etapa Terminal	
	I	II	III	IV	V	VI	VII	VIII
	Tronco común							
Matemáticas	Cálculo diferencial Álgebra superior Geometría vectorial	Cálculo integral Álgebra lineal	Cálculo vectorial Probabilidad Matemáticas discretas	Ecuaciones diferenciales ordinales Investigación de operaciones	Teoría de autómatas Métodos numéricos	-	-	Simulación
Arquitectura de computadoras	-	Mecánica Métodos experimentales	-	Organización y arquitectura de computadoras	-	-	-	-
Interacción Humano-computadora	-	-	-	-	Graficación	Inteligencia artificial	-	-
Programación de ingeniería de software	Diseño de algoritmos	Introducción a la programación	Estructura de datos y algoritmos Programación orientada a objetivos	Análisis de algoritmos Metodología de la programación	Ingeniería de software	Administración de proyectos	Sistemas distribuidos Reingeniería de procesos	-
Redes	-	-	-	-	-	Redes de datos	Seguridad de cómputo	-
Software de base	-	-	-	-	Sistemas operativos	Compiladores	-	-
Tratamiento de la información	-	-	-	Bases de datos	-	Minería de datos	-	-
Entorno social	Comunicación oral y escrita. Historia e impacto de la ciencia	Formación en valores	-	-	-	-	-	Aspectos legales, sociales y éticos de la computación

La distribución de los créditos por etapa de formación corresponde al 37.4% a la etapa básica, 38.2% a la etapa disciplinaria y 24.2% a la etapa terminal, incluidos los créditos que conciernen a las prácticas profesionales (ver tabla 2.10).

Tabla 2.10

Distribución de los créditos por etapa de formación y tipo de asignatura del programa de LCC

Etapas	Obligatorias		Optativas		Total	
	A ¹	CR ²	A ²	CR ²	A ²	%
Básica	17	112	3	19	20	37.4%
Disciplinaria	16	110	3	24	19	38.2%
Terminal	5	33	5	42	10	21.4%
Práctica profesional		10				2.8%
Total						100%

Nota. ¹Asignaturas, ²Créditos

En el siguiente apartado se muestra la distribución de los créditos por área del conocimiento.

Programas de unidades de aprendizaje

El plan de estudios del programa de LCC se desarrolla a través de cinco áreas de conocimiento recomendadas por la Asociación Nacional de Instituciones de Educación en Tecnología de la Información A. C. (ANIEI): Entorno Social, Matemáticas, Programación e Ingeniería de Software, Arquitectura de Computadoras, Redes, Software de Base, Interacción Humano-Computadora, y Tratamiento de la Información. La evaluación del aprendizaje se realiza de acuerdo con el título tercero, artículos 63 al 102 del Estatuto Escolar de la UABC (2006) en donde se especifican las escalas de calificaciones, los tipos de exámenes, revisión de exámenes y asistencia a clases (UABC, 2017).

El área de Entorno Social se integra por cinco UA obligatorias y seis optativas, los créditos obligatorios son 27. Es importante mencionar que en esta área ninguna UA requiere de seriación obligatoria. La bibliografía programada promueve un segundo idioma (ver tabla 2.11). Además, dos UA optativas se ofertan en inglés y español: Introducción a energías renovables y Negocios Tecnológicos.

Adicionalmente, en las evidencias de desempeño de cada UA se especifican reportes técnicos, ensayos académicos, exposiciones orales, plan de negocios, síntesis, portafolio de ejercicios y solución de problemas, protocolo de investigación, proyecto final, reporte técnico, y estudio de factibilidad (UABC, 2017; CONAIC, 2020).

Tabla 2.11

Asignaturas del área de conocimiento Entorno Social

Asignatura	HC	HL	HT	HPC	CR	S	% Bibliografía	
							español	inglés
Obligatorias								
Comunicación Oral y Escrita	01		03		05	I	63	37
Historia e Impacto de la Ciencia	02		01		05	I	73	27
Formación de Valores	01		03		05	II	50	50
Reingeniería de Procesos	01	01	03		06	VII	17	83
Aspectos Legales, Sociales y Éticos de la Computación	02		02		06	VIII	56	44
Optativas								
Química	02	03	03		08		40	60
Estructura Socio-Económica de México	02		02		06		78	22
Metodología de la Investigación	02		02		06		60	40
Emprendedores	02		03		07		89	11
Negocios Tecnológico	02		03		07		27	73
Introducción a Energías Renovables	03				06		0	100

Nota. De “Propuesta de Modificación del Programa de Licenciado en Ciencias Computacionales 2017-1”, UABC, 2017.

El área de Matemáticas se integra por 13 UA obligatorias y dos optativas, los créditos obligatorios son 103. En esta área las UA que requieren de seriación obligatoria son Cálculo Integral y Cálculo Vectorial, así como Teoría de Autómatas y Compiladores (esta última pertenece al área de conocimiento Software de base). La bibliografía programada promueve un segundo idioma (ver tabla 2.12).

Como evidencias de desempeño de cada UA se especifican portafolio de ejercicios y solución de problemas, desarrollo de un producto de software, compendio de prácticas de laboratorio, compendio de ejercicios, reporte de proyecto final (UABC, 2017; CONAIC, 2020).

Tabla 2.12*Asignaturas del área de conocimiento Matemáticas*

Asignatura	HC	HL	HT	HPC	CR	S	% Bibliografía	
							español	inglés
<i>Obligatorias</i>								
Cálculo Diferencial	02		04		08	I	60	40
Geometría Vectorial	01		03		05	I	25	75
Algebra Superior	02		03		07	I	18	82
Cálculo Integral	02		04		08	II		
Algebra Lineal	02		04		08	II	60	40
Cálculo Vectorial	03		02		08	III	33	67
Probabilidad	03		02		08	III	73	27
Matemáticas Discretas	02		03		07	III	17	83
Ecuaciones Diferenciales Ordinarias	03		03		09	IV	70	30
Investigación de Operaciones	02	02			06	IV	63	37
Teoría de Autómatas	01	02	02		06	V	43	57
Métodos Numéricos	02	02	02		08	V	30	70
Simulación	02	03			07	VIII	12	88
<i>Optativas</i>								
Algebra Lineal II	03		03		09		60	40
Estadística	03		03		09		58	42

Nota. De “Propuesta de Modificación de la Licenciatura en Matemáticas Aplicadas 2017-1”, UABC, 2017.

El área de Programación e Ingeniería de Software se integra por nueve UA obligatorias y siete optativas, los créditos obligatorios son 62. En esta área las UA que requieren de seriación obligatoria son Metodología de la Programación e Ingeniería de Software. La bibliografía programada promueve un segundo idioma (ver tabla 2.13).

Como evidencias de desempeño de cada UA se especifican proyecto final, desarrollo de prototipo en equipo, reporte de desarrollo de software, desarrollo de aplicación para dispositivos móviles, evaluación de al menos cuatro lenguajes de programación de diferentes paradigmas, proyecto de programación en internet, portafolio de proceso de desarrollo de software, desarrollo de programas computacionales, portafolio de ejercicios, portafolio de pruebas de proyecto de software (UABC, 2017; CONAIC, 2020).

Tabla 2.13*Asignaturas del área de conocimiento Programación e Ingeniería de Software*

Asignatura	HC	HL	HT	HPC	CR	S	% Bibliografía	
							español	inglés
Obligatorias								
Diseño de Algoritmos	02	02	01		07	I	63	37
Introducción a la Programación	02	03			07	II	-	100
Estructura de Datos y Algoritmos	02	02	01		07	III	-	100
Programación Orientada a Objetos	02	02	01		07	III	33	67
Análisis de Algoritmos	02	02	01		07	IV	-	100
Metodología de la Programación	02	03			07	IV	79	21
Ingeniería de Software	02	03			07	V	86	14
Administración de Proyectos	01	02	02		06	VI	56	44
Sistemas Distribuidos	02	03			07	VII	17	83
Optativas								
Paradigmas y Lenguajes de Programación	01	02	02		06		-	100
Introducción a la Programación en Internet	02	03			07		-	100
Desarrollo de Aplicaciones para Dispositivos Móviles	02	03			07		-	100
Física para Programadores de Videojuegos	02	01	02		07		-	100
Calidad del Software	02	01	02		07		50	50
Cómputo en la Nube	02	02	02		08		-	100
Cómputo Paralelo	02	02	02		08		-	100

Nota. De “Propuesta de Modificación del Programa de Licenciado en Ciencias Computacionales 2017-1”, UABC, 2017.

El área de Arquitectura de Computadoras se integra por tres UA obligatorias y tres optativas, los créditos obligatorios son 16. En esta área las UA no requieren de seriación obligatoria. La bibliografía programada promueve un segundo idioma (ver tabla 2.14).

Como evidencias de desempeño de cada UA se especifican portafolio de reporte de práctica, reporte final de investigación, desarrollo de un segmento de código en alto nivel y reporte técnico, portafolio de ejercicios, proyecto final, y bitácora de experimentos (UABC, 2017; CONAIC, 2020).

Tabla 2.14

Asignaturas del área de conocimiento Arquitectura de Computadoras

Asignatura	HC	HL	HT	HPC	CR	S	% Bibliografía	
							español	inglés

Asignatura	HC	HL	HT	HPC	CR	S	% Bibliografía	
							español	inglés
<i>Obligatorias</i>								
Mecánica	02		04		08	II	33	67
Métodos Experimentales	02	02			02	II	50	50
Organización y Arquitectura de Computadoras	02	02			06	IV	-	100
<i>Optativas</i>								
Programación de Sistemas Empotrados	02	02	01		07		-	100
Introducción a las Ciencias Computacionales	02		02		06		75	25
Circuitos Eléctricos	01	02			04		-	100

Nota. De “Propuesta de Modificación del Programa de Licenciado en Ciencias Computacionales 2017-1”, UABC, 2017.

El área de Redes se integra por dos UA obligatorias y tres optativas, los créditos obligatorios son 14. En esta área las UA no requieren de seriación obligatoria. La bibliografía programada promueve un segundo idioma (ver tabla 2.15).

Como evidencias de desempeño de cada UA se especifican portafolio, caso de estudio, proyecto final y reporte final (UABC, 2017; CONAIC, 2020).

Tabla 2.15

Asignaturas del área de conocimiento Redes

Asignatura	HC	HL	HT	HPC	CR	S	% Bibliografía	
							español	inglés
Obligatorias								
Redes de Datos	02	02	01		07	VI	20	80
Seguridad en Cómputo	02	03			07	VII	10	90
Optativas								
Redes Inalámbricas	02	02	01		07		33	67
Arquitecturas de Protocolos de Red	02	02	01		07		37	63
Administración de Centros de Cómputo	02	02	01		07		60	40

Nota. De “Propuesta de Modificación del Programa de Licenciado en Ciencias Computacionales 2017-1”, UABC, 2017.

El área de Software de Base se integra por dos UA obligatorias, los créditos obligatorios son 14. En esta área las UA no requieren de seriación obligatoria. La bibliografía programada promueve un segundo idioma (ver tabla 2.16).

Como evidencias de desempeño de cada UA se especifican desarrollo de compilador, proyecto final de estructura de sistemas operativos (UABC, 2017; CONAIC, 2020).

Tabla 2.16

Asignaturas del área de conocimiento Software de Base

Asignatura	HC	HL	HT	HPC	CR	S	% Bibliografía	
							español	inglés
Obligatorias								
Sistemas Operativos	02	03			07	V	50	50
Compiladores	02	02	01		07	VI	17	83

Nota. De “Propuesta de Modificación del Programa de Licenciado en Ciencias Computacionales 2017-1”, UABC, 2017.

El área de Interacción Humano-Computadora se integra por dos UA obligatorias y siete optativas, los créditos obligatorios son 14. En esta área las UA no requieren de seriación obligatoria. La bibliografía programada promueve un segundo idioma (ver tabla 2.17).

Como evidencias de desempeño de cada UA se especifican desarrollo de proyecto de aplicación, portafolio, desarrollo de sistema de software, desarrollo de aplicación interactiva funcional, proyecto final, portafolio, desarrollo de sistema de cómputo (UABC, 2017; CONAIC, 2020).

Tabla 2.17

Asignaturas del área de conocimiento Interacción Humano-Computadora

Asignatura	HC	HL	HT	HPC	CR	S	% Bibliografía	
							español	inglés
Obligatorias								
Graficación	02	03			07	V	-	100
Inteligencia Artificial	02	02	01		07	VI	43	57
Optativas								
Interacción Humano	02	02	02		08		25	75

Asignatura	HC	HL	HT	HPC	CR	S	% Bibliografía	
							español	inglés
Computadora								
Procesamiento Digital de Imágenes	02	04			08		7	93
Algoritmos Bioinspirados	02	02	01		07		-	100
Diseño de Interacción	02	02	02		08		-	100
Agentes Inteligentes	02	04			08		-	100
Reconocimiento de Patrones en Imágenes Digitales	03	03			09		-	100
Visión por Computadora	02	04			08		10	90

Nota. De “Propuesta de Modificación del Programa de Licenciado en Ciencias Computacionales 2017-1”, UABC, 2017.

El área de Tratamiento de la Información se integra por dos UA obligatorias y dos optativas, los créditos obligatorios son 13. En esta área las UA no requieren de seriación obligatoria. La bibliografía programada promueve un segundo idioma (ver tabla 2.18).

Como evidencias de desempeño de cada UA se especifican desarrollo de prototipo de Big data, desarrollo de un sistema de base de datos, proyecto final, construcción de un modelo de minería de datos (UABC, 2017; CONAIC, 2020).

Tabla 2.18

Asignaturas del área de conocimiento Tratamiento de la Información

Asignatura	HC	HL	HT	HPC	CR	S	% Bibliografía	
							español	inglés
<i>Obligatorias</i>								
Bases de Datos	02	02	01		07	IV	43	57
Minería de Datos	01	02	02		06	VI	-	100
<i>Optativas</i>								
Manipulación de Grandes Volúmenes de Datos	02	03			07		17	83
Big Data	02	02	02		08		17	83

Nota. De “Propuesta de Modificación del Programa de Licenciado en Ciencias Computacionales 2017-1”, UABC, 2017.

Tecnología educativa y de la información para el aprendizaje

El programa de LCC cuenta con Tecnología educativa y de la información para el aprendizaje. Los laboratorios equipados y puestos a disposición de estudiantes y docentes son un total de 10:

laboratorio de electrónica, cinco laboratorios de docencia, dos laboratorios de prácticas avanzadas, y

dos laboratorios de prácticas de investigación. Cada uno de estos dispone de computadoras, servicio de internet e impresión. El Laboratorio de investigación dispone de: ocho computadoras personales, una impresora Dell 2330d, 8 kits Lego Mindstorms, una Parrot Drone, una Phantom Drone, un kit Tetrix PITSCO Education, y un Wireless router lynksys WRT54G.

En el Laboratorio de investigación II se tiene a disposición de los estudiantes y docentes cinco Computadoras personales, un servidor Web, un Novint Falcon, una Leap motion interfaces, un kit de Phidgets, dos diademas emotiv+, y una diadema emotiv Insight. Por su parte, el Laboratorio de investigación III, cuenta con seis computadoras personales, un servidor Web, un Phantom Omni, una impresora 3D, dos Leap motion interfaces, dos kits de Phidgets, y dos diademas Mindwave (CONAIC, 2020).

Aunado a lo anterior, los estudiantes y docentes cuentan con servicios para el aprendizaje en línea, entre los que se mencionan: servicio de correo electrónico (<https://llave.uabc.edu.mx>); portal de alumnos (<http://alumnos.uabc.mx/>); Blackboard (<https://uabc.blackboard.com/>); El catálogo cimarrón de la Biblioteca (<http://catalogocimarron.uabc.mx/>); y el Sistema de Tutorías (<https://tutorias.uabc.mx/>) (CONAIC, 2020).

2.2.2 Actividades para la formación integral

El objetivo específico es analizar las actividades para la formación integral de los estudiantes.

Actividades complementarias para la formación integral

El Modelo educativo de la UABC indica tres atributos esenciales como son la flexibilidad curricular, la formación integral y el sistema de créditos. En específico, la formación integral apoya al estudiante a lo largo de las distintas etapas de formación con el objetivo de “incorporarse y adaptarse al entorno escolar; resolver las dificultades que enfrenta en su proceso de enseñanza-aprendizaje; tomar decisiones informadas en el ámbito académico y profesional; vincularse con su medio social, cultural y laboral; y fomentar su salud física y emocional” (UABC, 2013, p. 55). En cada una de las etapas de formación, Básica, Disciplinaria y Terminal, se especifican las actividades que contribuyen a construir el perfil profesional. Además, se establecen actividades complementarias denominadas *ejes transversales*, las cuales permean en todas las etapas, estas son: cultura y deportes; formación valoral; idioma extranjero; tutoría; y, orientación educativa y psicopedagógica (UABC, 2013).

Dentro de las actividades culturales se especifica la conmemoración del día internacional de la mujer 2020. Asimismo, la Facultad de Artes organiza eventos para toda la comunidad en general como exposiciones de arte, conciertos, obras de teatro, recitales, encuentros literarios, talleres con diversas temáticas entre los que se enlistan: creación literaria, iniciación a la plástica, dibujo artístico, guitarra popular, fotografía, danza, jazz y tap, fotoshop, cerámica (CONAIC, 2020).

Distintas actividades deportivas son diseñadas y ejecutadas por promotores deportivos asignados a las unidades académicas, por parte de la Facultad de Deportes de la UABC; estas tienen valor en créditos curriculares. Entre las actividades se encuentran deporte masivo y deporte curricular, en las que han participado un total de 27 estudiantes del programa de LCC durante los periodos 2019-1, 2020-1 y 2020-2. Es importante mencionar que, la UABC organizó actividades deportivas en línea durante la pandemia por Covid-19 para promover la activación física desde casa: #homechallenge, zumba, entrenamiento funcional, kick boxing, ritmos latinos, acondicionamiento funcional, entrenamiento virtual de futbol rápido varonil y baloncesto femenino, carrera virtual prohibido rendirse, entre otras (CONAIC, 2020).

En lo referente a actividades de orientación profesional, se señaló una participación de 93 estudiantes en Seminarios de la Facultad de Ciencias, Convenios, Proyectos de Vinculación con Valor en Créditos impartidos durante los periodos escolares: 2019-1, 2019-2, 2020-1 y 2020-2. También se realizó la “Semana de Ciencias 2019”, reportándose una asistencia de más 15,000 personas, a quienes se mostraron diversas actividades de divulgación, vinculación y promoción de la investigación; y durante el confinamiento en casa por la pandemia, este evento se efectuó en línea (CONAIC, 2020).

En relación con el tema de Emprendimiento, dos asignaturas optativas se relacionan directamente con ello: curso de Emprendedores y curso de Negocios Tecnológicos, aunque otras asignaturas solicitan un desarrollo tecnológico como proyecto final. Igualmente, se realizó el “Foro virtual de emprendimiento y empleabilidad 2020”, en el que se ejecutaron diversas actividades y videoconferencias para compartir conocimientos sobre emprendimiento y empleo como región fronteriza. A la par, cada año se organiza el Congreso internacional de ciencias computacionales (CiComp), el Foro de investigación de cuerpos académicos (FICA), y el Concurso de “Expo Emprendedores” (CONAIC, 2020).

La orientación educativa y psicopedagógica es impartida por un psicólogo adscrito a cada unidad académica, la atención está dirigida para: aspirantes, estudiantes de nuevo ingreso, y estudiantes universitarios. Entre las actividades que realizan estos profesionistas, dirigidas a los estudiantes inscritos se encuentran: orientación para favorecer el aprendizaje, orientación vocacional, atención a problemas personales, impartición de cursos y talleres sobre educación sexual, psicológica, hábitos de estudio y fomento de valores éticos y morales (CONAIC, 2020).

Enseñanza de lenguas extranjeras

El Modelo educativo de la UABC (2013), reconoce el dominio de la lengua extranjera como parte de la normatividad institucional y se delega a las unidades académicas definir el grado de conocimiento con base en el perfil profesional de la carrera. Adicionalmente, el Estatuto escolar de la UABC (2021) destaca la relevancia del conocimiento de una lengua extranjera en la formación del estudiante; convirtiéndolo en un requisito esencial de todos los planes de estudios de la universidad. El Estatuto escolar también señala el nivel de dominio, opciones y etapas de acreditación, redacción y presentación de cualquier trabajo terminal —tesis u obras similares— en lengua extranjera, mismas que serán establecidas por cada Unidad Académica (UABC, 2021).

La UABC informó sobre el fortalecimiento de los esquemas institucionales destinados para el aprendizaje del idioma inglés, a través de la impartición de 180 cursos formales en las que participaron 16,537 estudiantes y que fueron realizadas por las unidades académicas y Uniser, A.C.; se ofertaron 88 nuevas unidades de aprendizaje en idioma inglés; y se otorgaron certificados de competencias comunicativas en inglés a 37 estudiantes. Aunado a lo anterior, se señaló la aplicación del Examen Diagnóstico del Idioma Inglés (Exadii) a 11, 311 estudiantes, para evaluar el nivel de dominio del inglés al ingreso en tres áreas: comprensión lectora, redacción y comprensión auditiva (UABC, 2022).

Es importante destacar que el Plan de estudios 2017 del programa de LCC no indica en las habilidades del perfil de ingreso y egreso, las relacionadas con la comunicación oral y escrita en idioma inglés (UABC, 2017).

Conclusiones Estudio 2.2. Evaluación del currículo

El plan de estudios 2017-1 del programa de LCC es congruente con el Modelo Educativo, normativa y política institucional de la UABC. El total de créditos para cubrir el plan de estudios de este programa es de 350, durante ocho semestres. Las UA obligatorias son 38 y 11 optativas. Las competencias generales establecidas para alcanzar el perfil de egreso son tres. La distribución de las UA se observa con apego a los objetivos establecidos en cada una de las etapas básica, disciplinaria y terminal.

En lo referente a las áreas del conocimiento del plan de estudios y las UA, se diseñaron con apego a las recomendadas por la ANIEI: Entorno social, Matemáticas, Programación e Ingeniería de Software, Arquitectura de Computadoras; Redes; Interacción Humano-Computadora; Software de Base; y Tratamiento de la Información. Cada una de las UA registra bibliografía en español e inglés, e incluso algunas son impartidas en inglés, sobretodo optativas. Es importante mencionar que el área de Matemáticas presenta una mayor carga en la etapa básica, por lo que se requiere un perfil de ingreso con amplios conocimientos, habilidades y vocación hacia la misma.

Los avances tecnológicos previstos hacia el 2030 demandan una actualización y revisión de las áreas que se desarrollan en el plan de estudios 2017-1 del programa de LCC. Al respecto, se advierte sobre la rápida innovación, colaboración internacional, movilidad laboral, la necesidad de adaptar y desarrollar cuestiones éticas y la formación de estudiantes en esta área para dar continuidad a los estudios de posgrado.

Dentro de las actividades para la formación integral de los estudiantes del programa de LCC, se especifica el desarrollo de actividades culturales como la conmemoración del día internacional de la mujer 2020. Asimismo, la Facultad de Artes organiza eventos para toda la comunidad de la UABC como exposiciones de arte, conciertos, obras de teatro, talleres y cursos. Distintas actividades deportivas son diseñadas y ejecutadas por promotores deportivos asignados a las unidades académicas, por parte de la Facultad de Deportes de la UABC; y estas tienen valor en créditos curriculares. En lo que se refiere al tema del Emprendimiento, además de contar con dos asignaturas optativas, se realizaron diversos foros, congresos y concursos.

Específicamente el tema de formación en un segundo idioma, la UABC ha destinado esfuerzos y recursos para fortalecer los esquemas institucionales destinados para el aprendizaje del idioma inglés, por lo que se han impartido cursos formales, se ofertaron más unidades de aprendizaje de dicho idioma, y se otorgaron certificados de competencias comunicativas en inglés.

Además, se aplicó el Examen Diagnóstico del Idioma Inglés (Exadii) a estudiantes de nuevo ingreso, para evaluar su nivel de dominio (UABC, 2022).

Cabe señalar que el Plan de estudios 2017 del programa de LCC no indica en el perfil de ingreso y egreso, las habilidades requeridas en la comunicación oral y escrita en idioma inglés (UABC, 2017).

Estudio 2.3 Evaluación de la trayectoria escolar de los estudiantes en el programa educativo

Objetivo

El objetivo general de este estudio fue evaluar la trayectoria escolar de los estudiantes para modificar o actualizar el programa educativo. Para tal fin, se consideraron cuatro categorías: 2.3.1 Proceso de ingreso al programa educativo, 2.3.2 Indicadores de trayectoria escolar, 2.3.3 Participación de los estudiantes en programas de apoyo y 2.3.4 Resultados de los estudiantes (ver Tabla 2.19).

Tabla 2.19

Descripción de elementos estructurales de la Evaluación de la trayectoria escolar de los estudiantes por el programa educativo.

Categoría	Objetivo específico	Indicador
2.3.1 Proceso de ingreso	Describir el proceso de ingreso al programa educativo.	2.3.1.1 Estrategias de difusión, promoción y orientación del programa educativo
2.3.2 Indicadores de trayectoria escolar	Analizar indicadores de trayectoria escolar de los estudiantes que cursan el programa educativo.	2.3.2.1 Control del desempeño de los estudiantes en el programa educativo 2.3.2.2 Eficiencia terminal 2.3.2.3 Eficiencia en la titulación u obtención de grado
2.3.3 Participación de los estudiantes en programas de apoyo	Analizar la participación de los estudiantes en programas de apoyo.	2.3.3.1 Programas de asesoría académica, regularización y acciones de nivelación a los estudiantes 2.3.3.2 Programa de inclusión 2.3.3.3 Movilidad e intercambio de estudiantes 2.3.3.4 Servicios de tutoría 2.3.3.5 Servicios de orientación educativa y psicopedagógica de apoyo al estudiante 2.3.3.6 Prácticas profesionales, estancias y visitas en los diversos sectores 2.3.3.7 Servicio social
2.3.4 Resultados de los estudiantes	Evaluar los resultados de los estudiantes que cursaron el programa educativo, en exámenes, diversos eventos nacionales e internacionales, así como el dominio de otras lenguas.	2.3.4.1 Resultados en exámenes de egreso externos a la institución 2.3.4.2 Participación de estudiantes en concursos, competencias, exhibiciones y presentaciones nacionales e internacionales 2.3.4.3 Cumplimiento del perfil de egreso

Nota. De “Metodología de los estudios de fundamentación para la creación, modificación y actualización de programas educativos de licenciatura”, por A. Serna, y A. Castro, 2018, *Universidad Autónoma de Baja California*. CC BY-NC-ND.

Método

Este apartado se realizó una investigación de tipo documental. Para alcanzar los objetivos planteados se revisaron fuentes secundarias como Modelo Educativo de la UABC, Propuesta de modificación del programa educativo Licenciado en Ciencias Computacionales 2017-1, Autoevaluación del Programa de Licenciado en Ciencias Computacionales para la Acreditación de Programas Académicos de Informática y Computación en Educación Superior (CONAIC, 2020); Cuarto Informe de Actividades 2019-2023 (UABC, 2022); la página del Sistema Integral de Información Institucional, e-trayecto e indicadores de trayectoria escolar (UABC, 2023), así como el Sistema de Indicadores y Estadísticas Institucionales (UABC, 2021).

Resultados

2.3.1 Proceso de ingreso al programa educativo

El objetivo específico es describir el proceso de ingreso al programa educativo.

Estrategias de difusión, promoción y orientación del programa educativo

La Facultad de Ciencias difunde el programa de estudios (estructura del plan de estudios, objetivo, perfil, asignaturas, horas, duración, seriación y especialidades), a través de medios impresos como trípticos y medios electrónicos como la página web de la misma. De igual forma, cada tutor comparte la información mencionada a los alumnos de nuevo ingreso.

En lo referente a la promoción externa del programa educativo, se organizan diversos eventos como la Semana de ciencias realizada de forma anual en la que los estudiantes de la FC comparten, a estudiantes de todos los niveles educativos, prototipos, trabajos de investigación, cursos, talleres, seminarios, teatro, actividades recreativas. Durante la pandemia todo lo anterior se realizó de manera virtual. Otra actividad, son los seminarios organizados para estudiantes del nivel medio superior, en esta los docentes comparten el tipo de investigación que realizan y otros materiales con información del programa de LCC. Asimismo, la FC participa dos veces por año en las jornadas vocacionales organizadas por el Sistema Educativo Estatal. Otros medios de

comunicación son utilizados con el mismo fin: página en redes sociales, Expociencia y tecnología, publicación en prensa y videos promocionales (UABC, 2023).

Los padres de familia son parte importante del proceso de inducción, por lo que son convocados a una reunión informativa en la que se comparten temas relacionados con el proceso general de estudios en la UABC (presentación de autoridades de la facultad, explicación del reglamento, derechos y obligaciones de los estudiantes, y canales de información). Durante el ciclo escolar 2020-2, debido a la pandemia dicha reunión se realizó en línea.

2.3.2 Indicadores de trayectoria escolar

El objetivo específico es analizar indicadores de trayectoria escolar de los estudiantes que cursan el programa educativo.

Control del desempeño de los estudiantes en el programa educativo

En la Facultad de Ciencias se implementan diversas estrategias para atender las deficiencias académicas de los estudiantes de nuevo ingreso. En la semana de inducción se aplica un examen de diagnóstico, con los resultados arrojados han implementado las siguientes estrategias: (a) fortalecimiento de las tutorías, (b) instauración del centro de asesoría, en el que estudiantes de servicio social de las carreras de física, ciencias computacionales y matemáticas aplicadas, junto con docentes ofrecen asesorías a estudiantes inscritos en los primeros semestres; (c) asignación de tutores desde el ingreso, a partir de periodo 2017-1 los estudiantes cuentan con tutores del programa de LCC desde el tronco común; (d) creación de tutores inteligentes (CONAIC, 2020).

Respecto al rezago por reprobación en el programa de LCC de la FC, las cohortes —año de ingreso del estudiante con calificación— donde se presentaron mayor rezago nivel uno —una UA reprobada— fueron el 2019-1 y 2020-1 (2 estudiantes). En el caso del rezago nivel dos —dos UA reprobadas—, las cohortes que ubicaron más estudiantes fueron el 2018-2, 2019-1 y 2021-2 (2 estudiantes). Sobre el rezago nivel tres —tres o más UA reprobadas—, la cohorte con el mayor número de estudiantes fue el 2019-1 (10 estudiantes). Así, el nivel de rezago con un promedio general más alto fue el tres con 53.7%. También, se identifica la cohorte 2020-2 con el mayor número de estudiantes regulares (7 estudiantes) (ver Tabla 2.20).

Tabla 2.20

Rezago por reprobación del programa de LCC

Cohorte	Inscritos				Regulares				Rezago nivel 1				Rezago nivel 2				Rezago nivel 3			
	F ^a		%		F		%		F		%		F		%		F		%	
	M ^b	H ^c	M	H	M	H	M	H	M	H	M	H	M	H	M	H	M	H	M	H
2017-1	9			100	1			11	1			11	0			0	7		78	
	0	9	0	100	0	1	0	11	0	1	0	11					0	7	0	78
2017-2	5			100	2			40	0			0	0			0	3		60	
	1	4	20	80	1	1	20	20									0	3	0	100
2018-1	2			100	1			50	0			0	0			0	1		50	
	0	2	0	100	0	1	0	50									0	1	0	50
2018-2	16			100	4			25	1			6	2			13	9		56	
	0	16	0	100	0	4	0	25	0	1	0	6	0	2		13	0	9	0	56
2019-1	17			100	3			18	2			12	2			12	10		59	
	2	15	12	88	2	1	12	6	0	2	0	12	0	2		12	0	10	0	59
2019-2	15			100	5			33	2			13	1			7	7		47	
	1	14	7	93	1	4	7	26	0	2	0	13	0	1	0	7	0	7	0	47
2020-1	11			100	2			18	2			18	0			0	7		64	
	3	8	27	73	0	2	0	18	0	2	0	18					3	4	27.3	36.4
2020-2	14			100	7			50	1			7	1			7	5		36	
	4	10	29	71	1	6	7	43	0	1	0	7	0	1	0	7	3	2	21.4	14.3
2021-1	6			100	1			17	1			17	1			17	3		50	
	1	5	17	83	0	1	0	17	1	0	17	0	0	1	0	17	0	3	0	50
2021-2	9			100	2			22	1			11	2			22	4		44	
	1	8	11	89	0	2	0	22	0	1	0	11	0	2	0	22	1	3	11	33
2022-1	2			100	0			0	0			0	0			0	2		100	
	0	2	0	100													0	2	0	100
2022-2	2			100	2			100												
	0	2	0	100	0	2	0	100												
2023-1	--			--	--			--	--			--	--			--	--		--	
Promedios generales					27.8				10.2				8.3				53.7			
Promedios generales de las cohortes 2021-2 a 2023-1					30.8				7.7				15.4				46.1			

Nota. De “Indicadores de seguimiento escolar: Rezago por reprobación”, por Universidad Autónoma de Baja California, 2023. CC BY-NC-ND. ^a F=Frecuencia. ^b M=Mujer. ^c H=Hombre.

Respecto a la deserción en la FC, las cohortes 2018-2 y 2020-1 mostraron 4 bajas académicas, fueron las frecuencias más elevadas. La cohorte 2017-1 presentó 4 estudiantes potenciales a baja académica. (ver Tabla 2.21).

Tabla 2.21

Deserción del programa de LCC

Cohorte	Inscritos				Baja académica				Potenciales a baja académica			
	F ^a		%		F		%		F		%	
	M ^b	H ^c	M	H	M	H	M	H	M	H	M	H
2017-1	9			100	3			33	4			44
	0	9	0	100	0	3	0	33	0	4	0	44
2017-2	5			100	0			0	1			20

Cohorte	Inscritos				Baja académica				Potenciales a baja académica			
	F ^a		%		F		%		F		%	
	M ^b	H ^c	M	H	M	H	M	H	M	H	M	H
2018-1	1	4	20	80					0	1	0	20
	2		100		0		0		0		0	
	0	2	0	100								
2018-2	16		100		4		25		2		13	
	0	16	0	100	0	4	0	25	0	2	0	13
2019-1	17		100		3		18		2		12	
	2	15	12	88	0	3	0	18	0	2	0	12
2019-2	15		100		0		0		0		0	
	1	14	7	93								
2020-1	11		100		4		36		1		9	
	3	8	27	73	2	2	18	18	0	1	0	9
2020-2	14		100		2		14		0		0	
	4	10	29	71	1	1	7	7				
2021-1	6		100		0		0		0		0	
	1	5	17	83								
2021-2	9		100		0		0		0		0	
	1	8	11	89								
2022-1	2		100		0		0		0		0	
	0	2	0	100								
2022-2	2		100		0		0		0		0	
	0	2	0	100								
2023-1	--		--		--		--		--		--	
Promedios generales												

Nota. De “Indicadores de seguimiento escolar: Deserción (tasa)”, por Universidad Autónoma de Baja California, 2023. CC BY-NC-ND. ^a F=Frecuencia. ^b M=Mujer. ^c H=Hombre.

La FC se ha dado a la tarea de identificar las causas de las bajas académicas. Durante el tronco común se ha observado que la mayor reprobación sucede en la asignatura de matemáticas. De igual forma, en la etapa disciplinaria, las asignaturas de programación presentan la mayor incidencia de reprobación. Sin embargo, el mayor número de estudiantes reprobados se presentan en las asignaturas de cálculo diferencial, cálculo integral, mecánica, probabilidad y estadística, y ecuaciones diferenciales.

Para atender los índices de reprobación se propusieron las siguientes estrategias: fortalecimiento de las tutorías, creación del centro de asesorías, asignación de tutores al ingreso del programa, y la implementación de los tutores inteligentes (CONAIC, 2020).

Eficiencia en la titulación u obtención de grado.

En la UABC, la eficiencia en la titulación se especifica de dos formas: eficiencia terminal de titulación y eficiencia de titulación por egreso. La primera se obtiene de la relación entre el número de estudiantes inscritos por primera vez en una carrera universitaria —cohorte generacional— con los estudiantes titulados durante el transcurso del primer año de su egreso. Por su parte, la eficiencia de titulación por egreso se logra de la misma forma que la anterior, con la única diferencia en la cantidad de semestres que el estudiante requiere para titularse (UABC, 2023).

La cohorte generacional 2017-2 fue la que tuvo mayor cantidad de estudiantes en eficiencia terminal de titulación (dos) y en eficiencia de titulación por egreso (dos) (ver Tabla 2.22).

Tabla 2.22

Eficiencia terminal de titulación y eficiencia de titulación por egreso en el programa de LCC

Cohorte	Inscritos				ET ^a de titulación				E ^b de titulación por egreso			
	F ^c		%		F		%		F		%	
	M ^d	H ^e	M	H	M	H	M	H	M	H	M	H
2017-1	9			100	2			22	2			22
	0	9	0	100	0	2	0	22	0	2	0	22
2017-2	5			100	2			40	2			40
	1	4	20	80	1	1	20	20	1	1	20	20
2018-1	2			100	0			0	0			0
	0	2	0	100								
2018-2	16			100	1			6	1			6
	0	16	0	100	0	1	0	6	0	1	0	6
Totales y promedios generales												

Nota. De “Indicadores de seguimiento escolar: Eficiencia terminal”, por Universidad Autónoma de Baja California, 2023. CC BY-NC-ND. ^a ET=Eficiencia terminal. ^b E=Eficiencia. ^c F=Frecuencia. ^d M=Mujer. ^e H=Hombre.

2.3.3 Participación de los estudiantes en programas de apoyo

El objetivo específico es analizar la participación de los estudiantes en programas de apoyo.

Programas de asesoría académica, regularización y acciones de nivelación a los estudiantes

La tutoría se encuentra establecida en el Modelo educativo de la UABC como uno de los ejes transversales en las tres etapas de formación profesional: básica o tronco común, disciplinaria y terminal (UABC, 2013). En la UABC, la tutoría es un proceso en el cual un docente es designado como tutor que ofrece asesoría y orientación al estudiante durante toda su trayectoria escolar en diversos ámbitos, lo que le permitirá alcanzar las competencias básicas, genéricas y específicas de

su carrera profesional. El objetivo principal es disminuir los índices de reprobación, deserción y mejorar el índice de eficiencia terminal (UABC, s.f.).

En la Facultad de Ciencias, el proceso de designación de tutores establece que al inicio del ciclo escolar se asigna un profesor de tiempo completo como tutor de un grupo de estudiantes. El número mínimo de sesiones programadas son tres: durante las reinscripciones, a mitad de semestre y al final del mismo (UABC, s.f.).

Programa de inclusión

En la UABC se reconoce que un contexto vulnerable o en desventaja representa un desafío para los estudiantes de educación superior, comprometiendo el aseguramiento de la calidad de los programas educativos, la inclusión y la equidad (UABC, 2013). Por tanto, en apoyo a la población estudiantil en condición de vulnerabilidad, la FC promueve los apoyos extraordinarios que brinda la universidad (UABC, 2023). La Tabla 2.23 presenta los tipos de becas otorgadas en los periodos 2021-2, 2022-1, 2022-2 a los estudiantes de LCC. La beca prórroga es la de mayor solicitud y consiste en pagos diferidos del costo de inscripción, reinscripción, colegiatura y cuotas específicas; su aplicación es durante el transcurso del periodo semestral a cursar (UABC, 2023).

Tabla 2.23

Tipos de becas otorgadas a estudiantes del programa de LCC

Tipo de beca	Periodos		
	2021-2	2022-1	2022-2
Prórroga	131	120	115
Promedio	6	4	9
Investigación licenciatura	2	1	0
Económica	16	21	26
Deportiva	3	2	5

Nota. De “Histórico de becas otorgadas,” por Universidad Autónoma de Baja California, 2023.

Movilidad e intercambio de estudiantes

La movilidad y el intercambio estudiantil son todas aquellas acciones que permiten la recepción y envío de estudiantes a universidades nacionales e internacionales, favoreciendo la adquisición de competencias como la autonomía y adaptación de los estudiantes a nuevas circunstancias (UABC, 2013).

Durante el periodo escolar 2018-1 se enviaron cuatro estudiantes a otras universidades: dos nacionales (Universidad Autónoma de Nuevo León y Benemérita Universidad Autónoma de Puebla) y dos internacionales (Universidad de Granada, España y University of Regina, Canadá). En lo que respecta al periodo escolar 2019-1 se envió un estudiante a la Universidade da Coruña, España; y en el 2019-2 se recibieron dos estudiantes internacionales de la Universidad Católica del Norte, Chile. En el periodo 2020-1 un estudiante realizó su intercambio estudiantil a la Universidad Autónoma de Nuevo León (CONAIC, 2020).

Servicios de tutoría

La tutoría académica consiste en la asignación de un docente a cada estudiante desde su ingreso al programa y hasta su egreso, el tutor tiene la función de proporcionar información a sus tutorados para que planifiquen su trayecto académico y profesional, y de ser el caso, canalizarlos a las diversas instancias universitarias para su atención específica (UABC, 2013). La UABC cuenta con el sitio Web denominado Sistema Integral de Tutorías (SIT), mismo que permite el registro, seguimiento y evaluación del servicio de tutorías. Durante los periodos escolares 2019-1, 2019-2 y 2020-1 se registraron un promedio de 24 estudiantes por periodo (CONAIC, 2020).

Servicios de orientación educativa y psicopedagógica de apoyo al estudiante

En la UABC se ofrece el servicio de orientación educativa y psicológica, por lo tanto, en cada unidad académica se encuentra un especialista en psicología quien efectúa actividades encaminadas a la atención de situaciones que inciden en la adaptación, integración o trayectoria de los estudiantes (CONAIC, 2020; UABC, 2013). En la tabla 2.24 se desglosa el tipo de atención que se realiza con los aspirantes, estudiantes de nuevo ingreso, estudiantes universitarios y docentes.

Tabla 2.24

Atención del área de Orientación Educativa y Psicopedagógica

Atención	Descripción
Aspirantes	Información actual de las carreras que oferta la UABC, la aplicación del Examen de habilidades del Programa de Desarrollo Integral de la Persona (PRODIP), de la Encuesta de nuevo ingreso y Socioeconómica y requisitos de ingreso.
Estudiantes nuevo ingreso	Información sobre el curso de inducción y servicios con que brinda la UABC.
Estudiantes universitarios	Asesoría académica a los estudiantes, orientación vocacional, atención a situaciones personales e información referente a conferencias, cursos, talleres o asesorías individuales en relación con temas diversos —fomento de valores

Atención	Descripción
	éticos y morales. educación sexual, manejo del estrés, prevención del uso y abuso de sustancias legales e ilegales, orientación sobre depresión, suicidio, bulimia, anorexia, ansiedad, agresión y violencia—.
Docentes	Atención para la mejora de la práctica por medio de conocimientos y técnicas derivadas de la psicología —estimulación de habilidades del pensamiento, estrategias didácticas creativas, formación de valores en el aula y técnicas y dinámicas que favorezcan el aprendizaje.

Nota. De “Formato para la Autoevaluación. Licenciatura en Ciencias Computacionales. Plantel Ensenada. Universidad Autónoma de Baja California,” CONAIC, 2020.

Prácticas profesionales, estancias y visitas en los diversos sectores

La FC realiza colaboraciones para atender diversas problemáticas, a través de proyectos de vinculación con valor en créditos (PVVC), programas de servicio social profesional (SSP) y prácticas profesionales (PP). Durante el año 2022 se concretaron cuatro convenios de colaboración con empresas nacionales como son: Bodegas de Santo Tomás, S.A. de C.V., Hutchinson Ports Ensenada Cruiseport Village (ECV), S.A. de C.V., y Hutchinson Ports Ensenada International Terminal (EIT), S.A. de C.V. Asimismo, se cuenta con un convenio internacional con la empresa DPhi, Belgium (UABC, 2023). El programa de LCC tiene proyectos de vinculación con instituciones del nivel medio superior, secundarias y primarias para impartir conferencias que fomenten el interés por las ciencias computacionales (CONAIC, 2020).

Servicio social

El modelo educativo de la UABC integra diversas modalidades de aprendizaje y obtención de créditos que permitan promover la formación integral de sus estudiantes. El servicio social es una de estas, y son todas aquéllas actividades formativas y de aplicación del conocimiento dirigidas a los sectores vulnerables. Esta modalidad tiene dos etapas: servicio social comunitario (SSC), con una duración de 300 horas; y servicio social profesional (SSP), con 480 horas. El SSC tiene el objetivo de fortalecer la formación en valores de los estudiantes, y el SSP busca promover la aplicación de los conocimientos, habilidades, aptitudes y valores obtenidos durante su formación profesional (UABC, 2013).

2.3.4 Resultados de los estudiantes

Resultados en exámenes de egreso externos a la institución

El Estatuto escolar menciona que una de las modalidades de titulación con las que cuentan los estudiantes de licenciatura es obtener la constancia de aprobación del EGEL (UABC, 2021b). De esta manera, los egresados del programa de LCC tienen como opción de titulación el Examen General de Egreso de la Licenciatura en Computación (EGEL Plus COMPU) aplicado por el Centro Nacional de Evaluación para la Educación Superior (Ceneval) (CONAIC, 2020).

Los resultados del EGEL Plus COMPU fueron los siguientes: en el periodo 2018-2 egresaron y presentaron este examen 13 estudiantes, de los cuales 11 aprobaron. En el periodo 2019-1, 5 egresados obtuvieron su certificación; y en el periodo 2019-2 se titularon 2 egresados por esta vía (CONAIC, 2020). Derivado de estos resultados, el programa de LCC forma parte de los programas de alto rendimiento académico del Ceneval (Indicador de Desempeño Académico por Programas de Licenciatura) (CONAIC, 2020).

Participación de estudiantes en concursos, competencias, exhibiciones y presentaciones nacionales e internacionales

Los estudiantes del programa de LCC, cursan una asignatura optativa en formación de emprendedores. Además, se promueve la participación con sus proyectos en la Feria Expo Emprendedores realizada semestralmente por la Vicerrectoría Campus Ensenada. En esta, los universitarios presentan proyectos innovadores y propuestas de creación de empresas en diversas categorías como: Tecnologías de la Información y Comunicación, Agroindustria o industria alimenticia, Proyectos tecnológicos e industriales; Medio ambiente, desarrollo y energía; Artístico-culturales y de servicios; Ciencias de la Salud y Farmacéutica (CONAIC, 2020).

Conclusiones Estudio 2.3 Evaluación de la trayectoria escolar de los estudiantes por el programa educativo

Con respecto a la difusión del programa de estudios, la Facultad de Ciencias utiliza diversos medios impresos y electrónicos, y cada tutor comparte la información a los alumnos de nuevo ingreso. Para realizar promoción externa se organizan diversos eventos como la Semana de ciencias. Una actividad importante, son los seminarios organizados para estudiantes del nivel medio superior, y las jornadas vocacionales del Sistema educativo estatal en las que se participa dos veces al año. En el

curso de inducción se convoca a padres de familia y se comparte información del proceso general de estudios en la UABC.

Sobre el tema de rezago, en la FC se implementan diversas estrategias para atender las deficiencias académicas de los estudiantes de nuevo ingreso se aplica un examen de diagnóstico en la semana de inducción, con los resultados arrojados han implementado las siguientes estrategias: fortalecimiento de las tutorías, instauración del centro de asesoría, asignación de tutores desde el ingreso, creación de tutores inteligentes (CONAIC, 2021). Respecto al rezago por reprobación en el programa de LCC de la FC, las cohortes donde se presentaron mayor rezago nivel uno fueron el 2019-1 y 2020-1 (2 estudiantes). En el caso del rezago nivel dos, las cohortes que ubicaron más estudiantes fueron el 2018-2, 2019-1 y 2021-2 (2 estudiantes). Sobre el rezago nivel tres, la cohorte con el mayor número de estudiantes fue el 2019-1 (10 estudiantes). También, se identifica la cohorte 2020-2 con el mayor número de estudiantes regulares (7 estudiantes) [UABC, 2023].

Respecto a la deserción en la FC, las cohortes 2018-2 y 2020-1 mostraron 4 bajas académicas, fueron las frecuencias más elevadas. La cohorte 2017-1 presentó 4 estudiantes potenciales a baja académica. La FC se da dado a la tarea de identificar las causas de las bajas académicas y detectaron que, durante el tronco común, la mayor reprobación sucede en la asignatura de matemáticas; en la etapa disciplinaria, las asignaturas de programación presentan la mayor incidencia de reprobación; sin embargo, el mayor número de estudiantes reprobados se presentan en las asignaturas de cálculo diferencial, cálculo integral, mecánica, probabilidad y estadística, y ecuaciones diferenciales (CONAIC, 2021).

Referente al tema de eficiencia terminal, la cohorte generacional 2017-2 fue la que tuvo mayor cantidad de estudiantes en eficiencia terminal de titulación (dos) y en eficiencia de titulación por egreso (dos) [UABC, 2023].

La tutoría se encuentra establecida en el Modelo educativo de la UABC como uno de los ejes transversales en las tres etapas de formación profesional: básica o tronco común, disciplinaria y terminal (UABC, 2013). En la Facultad de Ciencias, se asigna un profesor de tiempo completo como tutor de un grupo de estudiantes, y el mínimo de sesiones programadas son tres (UABC, s.f.); el 64.6% de los docentes del programa de LCC opinaron que las actividades de asesoría y apoyo académico para los estudiantes son adecuadas.

En apoyo a la población estudiantil en condición de vulnerabilidad, la FC promueve los apoyos extraordinarios que brinda la universidad. La beca prórroga destaca como la de mayor

solicitud y consiste en pagos diferidos del costo de inscripción, reinscripción, colegiatura y cuotas específicas (UABC, 2023).

La movilidad y el intercambio estudiantil consisten en la recepción y envío de estudiantes a universidades nacionales e internacionales, para favorecer la adquisición de competencias como la autonomía y adaptación de los estudiantes a nuevas circunstancias (UABC, 2013). Durante el periodo escolar 2018-1 al 2020-1 se enviaron cuatro estudiantes a universidades nacionales y dos internacionales; asimismo en el 2019-2 se recibieron dos estudiantes internacionales (CONAIC, 2020). Al respecto, el 53.0% de los académicos coincidieron en total acuerdo sobre la contribución de la movilidad nacional o internacional a la formación profesional de los estudiantes.

En el tema de las tutorías, la UABC cuenta con el sitio Web denominado Sistema Integral de Tutorías (SIT), mismo que permite el registro, seguimiento y evaluación de este servicio. El total de estudiantes atendidos a través de este servicio fueron 24 por cada uno de los siguientes periodos escolares: 2019-1, 2019-2 y 2020-1 (CONAIC, 2020). En opinión de los estudiantes, el 80% consideraron muy bueno el programa de tutorías. En su opinión, resaltaron la eficacia en la comunicación y la disponibilidad de su tutor, así como la solución de dudas que les ayuda tomar mejores decisiones. Por su parte, el 47.0% de los académicos estuvieron de acuerdo en el beneficio de la función del tutor en la formación académica de los estudiantes.

En la UABC se ofrece el servicio de orientación educativa y psicológica, y en cada unidad académica se encuentra un especialista en psicología quien atiende situaciones que inciden en la adaptación, integración o trayectoria de los estudiantes (CONAIC, 2020; UABC, 2013). El 33.0% de los docentes estuvo de acuerdo en el beneficio de los programas y actividades realizadas por el área de OEP en el proceso formativo de los estudiantes; sin embargo, el 20.0% de los académicos señaló no contar con bases para opinar al respecto.

La FC realiza colaboraciones para atender diversas problemáticas, a través de proyectos de vinculación con valor en créditos (PVVC), programas de servicio social profesional (SSP) y prácticas profesionales (PP). Durante el año 2022 se concretaron cuatro convenios de colaboración con empresas nacionales; también, se cuenta con un convenio internacional (UABC, 2023). Asimismo, el programa de LCC tiene proyectos de vinculación con instituciones del nivel medio superior, secundarias y primarias para impartir conferencias que fomenten el interés por las ciencias computacionales (CONAIC, 2020). El 43.7% de los egresados calificaron como buena su experiencia durante las prácticas profesionales, aunque el 6.2% consideraron una experiencia mala, y un porcentaje similar con una muy mala experiencia.

El Servicio Social Comunitario tiene el objetivo de fortalecer la formación en valores de los estudiantes, y el Servicio Social Profesional busca promover la aplicación de los conocimientos, habilidades, aptitudes y valores obtenidos durante su formación profesional (UABC, 2013). El 53.3% de los académicos coincidieron en la contribución positiva del servicio social en la formación profesional de los estudiantes. El 50.0% de los egresados valoraron como buena la atención que la Coordinación de servicio social de la FC le brindó en el transcurso de formación profesional. Referente al SSC, el 25.0% de los egresados señalaron que este servicio fortaleció su formación en valores; sin embargo, el 31.2% consideró lo contrario. Mientras el 50.0% de los egresados valoraron positivamente el SSP, y opinaron que en esta modalidad pusieron en práctica los conocimientos, habilidades, actitudes y valores que hasta ese momento habían adquirido en su formación profesional.

El Estatuto escolar menciona que una de las modalidades de titulación con las que cuentan los estudiantes de licenciatura es obtener la constancia de aprobación del EGEL (UABC, 2021b). En el EGEL Plus COMPU se obtuvieron los siguientes resultados: en el periodo 2018-2 egresaron y presentaron este examen 13 estudiantes, de los cuales 11 aprobaron. En el periodo 2019-1, 5 egresados obtuvieron su certificación; y en el periodo 2019-2 se titularon 2 egresados por esta vía (CONAIC, 2020). Derivado de estos resultados, el programa de LCC forma parte de los programas de alto rendimiento académico del Ceneval (CONAIC, 2020).

Respecto a la participación de estudiantes en concursos, competencias, exhibiciones o presentaciones, los estudiantes del programa de LCC, cursan una asignatura optativa en formación de emprendedores. Además, se promueve la participación con sus proyectos en la Feria Expo Emprendedores realizada semestralmente por la Vicerrectoría Campus Ensenada (CONAIC, 2020).

Finalmente, en el cumplimiento del perfil de egreso, el 50.0% de los empleadores consideró en total acuerdo que los egresados que laboran en su empresa cumplen con el perfil de egreso de el programa de LCC. De igual forma, el 60% de los empleadores consideró aceptable el nivel de formación en conocimientos teóricos de los egresados; mientras el 40% opinó que las habilidades prácticas son mejorables; y el 30% valoró mejorables las competencias contextualizadas al campo laboral.

Estudio 2.4 Evaluación del personal académico, infraestructura y servicios

Objetivo

El objetivo general de este estudio fue evaluar la suficiencia del personal académico, la infraestructura física y académica y los servicios de apoyo al programa, a fin de fundamentar la modificación o actualización de programas educativos. La organización de este apartado incluyó cuatro categorías: 2.4.1 Personal académico, 2.4.2 Infraestructura académica, 2.4.3 Infraestructura física y 2.4.4 Servicios de apoyo (ver Tabla 2.25).

Tabla 2.25

Descripción de elementos estructurales de la Evaluación del currículo

Categoría	Objetivo específico	Indicador
2.4.1 Personal académico	Describir los niveles de formación del personal académico referentes a la composición actual del cuerpo docente, la superación disciplinaria y habilitación académica, la producción académica para el programa —antecedentes en actividades profesionales y de investigación—, las formas de organización para el trabajo académico, las líneas de generación, aplicación del conocimiento y su transferencia al programa, y la articulación de la investigación con la docencia.	2.4.1.1 Composición actual del cuerpo docente
		2.4.1.2 Desarrollo disciplinario y habilitación académica
		2.4.1.3 Producción académica para el programa
		2.4.1.4 Formas de organización para el trabajo académico
		2.4.1.5 Líneas de generación, aplicación del conocimiento y su transferencia al programa
2.4.2 Infraestructura académica	Describir la infraestructura académica, entendida como aulas y espacios para la docencia, los laboratorios y talleres específicos para la realización de prácticas, la biblioteca, los espacios destinados para profesores, los espacios para encuentros académicos y/o culturales y otras instalaciones fuera de la institución.	2.4.2.1 Aulas y espacios para la docencia y su equipamiento
		2.4.2.2 Laboratorios y talleres específicos para la realización de prácticas, y su equipamiento
		2.4.2.3 Otras instalaciones fuera de la Unidad Académica
		2.4.2.4 Biblioteca
		2.4.2.5 Espacios destinados para profesores
		2.4.2.6 Espacios para eventos académicos y culturales
2.4.3 Infraestructura física	Describir la infraestructura física como los espacios donde se imparte el programa, la seguridad de personas y bienes, las áreas deportivas, de recreación y convivencia, y la conectividad. Incluir la infraestructura para el acceso a personas con necesidades especiales.	2.4.3.1 Infraestructura física del lugar donde se imparte el programa
		2.4.3.2 Seguridad de personas y bienes
		2.4.3.3 Seguridad de personas discapacitadas
		2.4.3.4 Áreas deportivas, de recreación y convivencia
		2.4.3.5 Conectividad
2.4.4 Servicios de apoyo	Describir los servicios de apoyo entre los que se encuentran la administración escolar, los servicios estudiantiles, las becas y la orientación para el tránsito a la vida	2.4.4.1 Administración escolar
		2.4.4.2 Servicios a estudiantes
		2.4.4.3 Becas estudiantiles
		2.4.4.4 Tránsito a la vida profesional

Categoría	Objetivo específico	Indicador

profesional.

Nota. De “Metodología de los estudios de fundamentación para la creación, modificación y actualización de programas educativos de licenciatura”, por A. Serna, y A. Castro, 2018, *Universidad Autónoma de Baja California*. CC BY-NC-ND.

Método

Este apartado se realizó una investigación de tipo documental. Para alcanzar los objetivos planteados se revisaron fuentes secundarias como Modelo Educativo de la UABC, Propuesta de modificación del programa educativo Licenciado en Ciencias Computacionales 2017-1, Autoevaluación del Programa de Licenciado en Ciencias Computacionales para la Acreditación de Programas Académicos de Informática y Computación en Educación Superior (CONAIC, 2020); Cuarto Informe de Actividades 2019-2023 (UABC, 2022); la página del Sistema Integral de Información Institucional, e-trayecto e indicadores de trayectoria escolar (UABC, 2023), así como el Sistema de Indicadores y Estadísticas Institucionales (UABC, 2021).

Resultados

2.4.1 Personal académico

El objetivo específico es describir los niveles de formación del personal académico referentes a la composición actual del cuerpo docente, la superación disciplinaria y habilitación académica, la producción académica para el programa —antecedentes en actividades profesionales y de investigación—, las formas de organización para el trabajo académico, las líneas de generación, aplicación del conocimiento y su transferencia al programa, y la articulación de la investigación con la docencia.

Composición actual del cuerpo docente

En el periodo 2022-2, el programa de LCC contó con una planta académica de 10 profesores de tiempo completo (PTC), 2 técnicos académicos (TA) y 4 profesores de asignatura (PA). De manera adicional, docentes de otros programas educativos de la FC colaboran con el programa de LCC: 4 PTC, 2 TA y 2 PA (UABC, 2023).

Las características de los docentes del programa de LCC son: el máximo nivel de estudios es Maestría y compete a siete docentes (43.8%), seguido de Doctorado con nueve académicos (56.2%). En correspondencia con esta última cifra, 10 académicos (62.5%) pertenecen al SNII del Conahcyt. Los docentes laboran en el sector público. En lo referente a los años de experiencia, nueve docentes

(56.2%) cuentan con más de 21 años laborando en la educación superior, solo un académico (6.2%) tiene entre 1 y 5 años de experiencia (ver tabla 2.26).

Tabla 2.26

Características generales del cuerpo docente

Ítems	Respuestas	Frecuencias	%
Nivel de estudios	Licenciatura o Ingeniería	0	0
	Maestría	7	43.8
	Doctorado	9	56.2
	Posdoctorado	0	0
Sector de actividad profesional	Público	16	100
	Privado	0	0
	Ambos	0	0
Años de experiencia	1-5 años	1	6.2
	6-10 años	4	25.0
	11-15 años	2	12.5
	16-20 años	0	0
	Más de 21 años	9	56.2

Asimismo, en la FC la estructura organizacional muestra que los docentes participan impartiendo clases, y participan en algún cargo administrativo. (ver tabla 2.27). La planta académica de la Facultad de Ciencias al semestre 2020-1 está conformada por 91 miembros, los cuales se distribuyen de la siguiente manera: 44 profesores de tiempo completo (PTC), 1 profesor de medio tiempo (PMT), 4 técnicos académicos de tiempo completo (TATC), 42 profesores por asignatura (PA); el 21.43% de los PTC pertenece al programa de Licenciado en Ciencias Computacionales (UABC, 2021f).

Tabla 2.27

Listado de colaboradores del Programa de Licenciado en Ciencias Computacionales

Categoría	Nombre del docente	Asignaturas	Cargo
Profesores de tiempo completo titular	Dr. Leopoldo Morán y Solares	Ingeniería de Software	Director de la Facultad de Ciencias. PTC(C).
	Dr. José Ángel González Fraga	Probabilidad, Inteligencia Artificial, Optativa (Procesamiento Digital de Imágenes)	Responsable del programa educativo. PTC(C).
	Dra. Alma Rocío Cabazos Marín	Diseño de Algoritmos, Metodología de la	Responsable de seguimiento de

Categoría	Nombre del docente	Asignaturas	Cargo
		Programación, Simulación.	egresados. PTC(B).
	Dr. Omar Álvarez Xochihua	Introducción a la Programación, Programación Orientada a Objetos, Optativa (Manejo de grandes volúmenes de datos).	Responsable de Laboratorio. PTC(B).
	Dra. María Victoria Meza Kubo	Teoría de Autómatas, Optativa (Interfaz Humano-Computadora).	Coordinadora de Investigación y Posgrado. PTC(C).
	M.I. Evelio Martínez Martínez	Organización y Arquitectura de Computadoras, Sistemas Operativos, Redes de Datos, Aspectos Legales, sociales y éticos de la computación.	Responsable de Tronco común de Ciencias Exactas. PTC(B).
	Dr. Everardo Gutiérrez López	Estructura de Datos y Algoritmos, Análisis de Algoritmos, Compiladores, Optativa (Algoritmos Bioinspirados)	Responsable de Prácticas Profesionales. PTC(A).
	Dr. Luis Miguel Pellegrin Zazueta	Graficación, Minería de Datos, Optativa (Paradigmas y Lenguajes de Programación)	Responsable de Laboratorio. PTC(B).
	Dra. Eloisa del Carmen García Canseco	Matemáticas Discretas, Optativa (Programación de Sistemas Empotrados).	PTC(B).
	M.I. Judith Isabel Luna Serrano	Geometría Vectorial, Historia e Impacto de la Ciencia, Administración de Proyectos, Optativa (Negocios Tecnológicos).	Coordinadora de Extensión y Vinculación. PTC(B).
Profesores de asignatura	M.C. José Alejandro González Sarabia	Diseño de Algoritmos	Profesor de asignatura
	M.C. Eduardo Castro González	Bases de datos	Profesor de asignatura
	MTIC Marysol Aguirre Arzate	Reingeniería de Procesos	Técnico académico
	Dr. José Eleno Lozano Risk	Sistemas Distribuidos	Profesor de asignatura
Técnicos académicos	MTIC. Adrián Enciso Almanza	Diseño de algoritmos, Emprendedores.	Responsable de Difusión
	M.C. Gerardo Tovar Ramos	Introducción a los Sistemas Operativos, Seguridad en Cómputo, Redes.	Responsable de Laboratorio de Cómputo

Producción académica para el programa

La producción académica del 2017 al 2020 fue la siguiente: 25 materiales didácticos, 2 materiales de divulgación, 1 libro de texto, 8 proyectos registrados en la Coordinación General de Posgrado e Investigación desarrollados dentro del periodo mencionado. Estos últimos derivaron: una tesis de maestría, seis tesis de doctorado, una patente, cinco prototipos, 10 artículos en revistas indexadas, 11 memorias en congresos, tres capítulos de libro, y un libro (CONAIC, 2020).

Formas de organización para el trabajo académico

La FC tiene una normatividad clara y precisa que relaciona las actividades administrativas con las académicas y se establece en los manuales y reglamentos de organización y procedimientos tales como el Estatuto del Personal Académico de la UABC, el Reglamento Interior de Trabajo y el Reglamento Interno de la Facultad de Ciencias (CONAIC, 2020).

En el programa de LCC la organización para el trabajo académico se aplica a través de dos cuerpos colegiados. De igual forma, el organigrama indica tres Coordinadores (Formación profesional, Extensión y Vinculación, e Investigación y Posgrado), y nueve Responsables de área (Programa educativo, Laboratorio, Orientación educativa y psicopedagógica, Servicio social, Prácticas profesionales, Movilidad estudiantil, Difusión, Formación y evaluación docente, y Seguimiento de egresados) (CONAIC, 2020).

Líneas de generación, aplicación del conocimiento y su transferencia al programa

En el programa de LCC se tienen registrados dos Cuerpos Académicos y ocho líneas de generación y aplicación del conocimiento. En el Cuerpo académico con grado Consolidado denominado “Tecnologías para ambientes inteligentes”, el líder es el Dr. Alberto Leopoldo Morán y Solares, se conforma por siete colaboradores de la Facultad de Ciencias y la Facultad de Ingeniería de la UABC. Esta línea se dedica al análisis, diseño y desarrollo de infraestructuras para ambientes inteligentes, que proveen oportunamente servicios a los usuarios. Las líneas de generación y aplicación del conocimiento asociadas son cinco (Sistemas colaborativos, Interacción humano-computadora, Cómputo consciente del contexto, Tecnologías asistivas y Mecatrónica y Sistemas Hápticos).

En lo referente al Cuerpo Académico con grado Consolidado denominado “Tecnologías de Información y Visualización”, cuenta con dos Líneas de generación y aplicación del conocimiento:

Cómputo visual y Tecnologías de Información y Comunicación. El líder es el Dr. José Ángel González Fraga y se integra por cuatro miembros de la Facultad de Ciencias (CONAIC, 2020).

2.4.2 Infraestructura académica

El objetivo específico es describir la infraestructura académica, entendida como aulas y espacios para la docencia, los laboratorios y talleres específicos para la realización de prácticas, la biblioteca, los espacios destinados para profesores, los espacios para encuentros académicos y/o culturales y otras instalaciones fuera de la institución.

Aulas y espacios para la docencia y su equipamiento

El programa de LCC tiene disponibles 13 aulas con capacidad para atender entre 15, 35 y 40 estudiantes. Estas aulas poseen condiciones adecuadas de mobiliario, higiene, ventilación, iluminación y seguridad (CONAIC, 2020).

Laboratorios y talleres específicos para la realización de prácticas, y su equipamiento

El programa de LCC cuenta con un total de 10 laboratorios: un laboratorio de electrónica, cinco laboratorios de docencia, dos laboratorios de prácticas avanzadas, y dos laboratorios de prácticas de investigación. Cada uno de estos dispone de computadoras, servicio de internet e impresión. El Laboratorio de investigación dispone del siguiente equipo: 8 computadoras personales, una impresora Dell 2330d, 8 kits Lego Mindstorms, una Parrot Drone, una Phantom Drone, un kit Tetrix PITSCO Education, y un Wireless router lynksys WRT54G. En el Laboratorio de investigación II se tiene a disposición de los estudiantes y docentes lo siguiente: 5 Computadoras personales, un servidor Web, un Novint Falcon, una Leap motion interfaces, un kit de Phidgets, dos diademas emotiv+, y una diadema emotiv Insight. Por su parte, el Laboratorio de investigación III, cuenta con seis computadoras personales, un servidor Web, un Phantom Omni, una impresora 3D, dos Leap motion interfaces, dos kits de Phidgets, y dos diademas Mindwave (CONAIC, 2020).

Biblioteca

La UABC ofrece a la comunidad académica un sistema de bibliotecas ubicadas en cada campus: Tijuana, Ensenada y Mexicali, además de otras disponibles en diversos institutos, unidades académicas o facultades. En total se cuenta con un Acervo bibliográfico impreso de 304,739 títulos y 473,256 volúmenes (UABC, 2022).

Además, la UABC pone a disposición de los usuarios una biblioteca digital con 11 editoriales para consulta de libros electrónicos, un repositorio institucional que alberga libros, capítulos de libro, tesis, y distintas revistas entre las que se enlistan la Revista Ciencias Marinas, Revista Culturales, Revista Ciencias Tecnológicas, REDIE, y la Revista Estudios Fronterizos. También cuenta con diversas bases de datos como Clarivate Analytics, Elsevier B.V., JSTOR, Springer-Nature, vlex, entre otras (UABC, 2021c).

La Biblioteca de la Unidad Punta Morro del campus Ensenada se localiza a 80 metros del edificio del programa de LCC, tiene capacidad para atender a 380 usuarios simultáneamente y cuenta con 41,801 ejemplares de la clasificación LC “Ciencias” (CONAIC, 2020). Los usuarios internos pueden realizar consultas en la sala, usar los cubículos de estudio, acceder a la sala de lectura, utilizar equipo de cómputo, solicitar préstamo externo e interbibliotecario, consultar el catálogo en línea, acceder a la biblioteca digital, hacer visita guiada, así como recibir asesoría y capacitación para búsquedas o uso de las distintas fuentes de información (UABC, 2021d).

Espacios destinados para profesores

Los docentes de tiempo completo, técnicos académicos y docentes de asignatura disponen de un cubículo para realizar sus labores académicas (CONAIC, 2020).

Espacios para eventos académicos y culturales

Las actividades culturales, entre las que se pueden mencionar obras de teatro, eventos musicales, exposiciones de arte, danza, festivales, o conferencias, se realizan en el Teatro Universitario Benito Juárez con 407 localidades; en la Sala audiovisual Francisco Zarco con 150 espacios; en el Andador Universitario ubicado a la orilla del mar en la UABC Campus Ensenada con un aforo para 1000 personas; las salas de Audiovisual 1 y 2 de la Facultad de Ciencias, con espacios para recibir a 70 y 77 personas, respectivamente. De igual forma, en el Departamento de Informática y Bibliotecas de la Unidad Punta Morro del campus Ensenada, se ubican dos salas más para realizar este tipo de eventos, tales como la sala de usos múltiples con capacidad para 250 personas, equipada con aire acondicionado, mobiliario, proyector, pantalla, equipo de sonido y micrófono; y la sala de audiovisual con disponibilidad de 85 espacios (CONAIC, 2020).

2.4.3 Infraestructura física

El objetivo específico es describir la infraestructura física como los espacios donde se imparte el programa, la seguridad de personas y bienes, las áreas deportivas, de recreación y convivencia, y la conectividad. Incluir la infraestructura para el acceso a personas con necesidades especiales.

Infraestructura física del lugar donde se imparte el programa

La Facultad de Ciencias se ubica dentro del Campus Ensenada de la UABC, cuenta con una infraestructura de 10 edificios, mismos que se integran con los siguientes espacios: una área de oficinas administrativas, una área de cubículos para los docentes de tiempo completo y de asignatura, una área de cubículos para técnicos académicos y orientación educativa y psicopedagógica, dos áreas de sanitarios, dos salas de audiovisuales, una sala de juntas, un almacén general, 13 aulas de licenciatura, una aula para asesorías, cuatro salones y dos espacios para atención de estudiantes de posgrado, dos espacios de colecciones y jardín botánico, y 35 laboratorios (15 de ciencias exactas y 20 de ciencias naturales) (UABC, 2023).

Aunado a lo anterior, el Departamento de Informática y Bibliotecas (DIB) del campus Ensenada pone a disposición de la comunidad educativa cinco salas de cómputo para clases –las cuales incluyen entre 18 y 23 computadoras– y una sala general de cómputo –con 84 equipos de cómputo– para que los estudiantes realicen sus tareas y trabajos académicos, todas estas cuentan con proyector, pantalla, aire acondicionado, y pintarrón (CONAIC, 2020).

Seguridad de personas y bienes

La Facultad de Ciencias mantiene la seguridad y resguardo de personas y bienes a través del Sistema Integral de Seguridad Universitaria (SISU). El SISU tiene a su cargo la contratación de una empresa de seguridad privada, misma que se encarga de realizar las siguientes actividades durante todo el año y en horario continuo: monitoreo del acceso vehicular y peatonal, recorridos de vigilancia y control del sistema de video vigilancia del campus (UABC, 2023).

En el tema de seguridad e higiene, en esta facultad se cuenta con una brigada multifuncional de emergencias, conformada por personal académico y administrativo que tienen como objetivo la prevención y atención ante diversos eventos, y se integra por cuadrillas de: unidad interna de protección civil; prevención y combate de incendios; evacuación y resguardo, búsqueda y rescate; primeros auxilios; comunicaciones de emergencia; y derrames de sustancias químicas y residuos

peligrosos. Conjuntamente, se ofrece equipo para combatir sismos e incendios como extintores, alarmas contra incendios, detectores de humo, luces de emergencia, botiquines de emergencia, indicaciones para actuar en caso de sismos o incendios, y señalamientos de evacuación y salidas de emergencia (UABC, 2023).

Seguridad de personas discapacitadas

En la Facultad de Ciencias se cuenta con rampas de acceso para personas con discapacidad en los pasillos del primer piso (UABC, 2023).

Áreas deportivas, de recreación y convivencia

Los estudiantes de la Facultad de Ciencias tienen disponibles espacios para la actividad física y el deporte, entre los que se encuentran: el gimnasio universitario de la Unidad Punta Morro, así como canchas de fútbol y de basquetbol; también cuentan con el gimnasio universitario de la Unidad Valle Dorado y el centro deportivo de la UABC. Específicamente, en la Facultad de Ciencias se encuentra un jardín botánico y diversos espacios habilitados para la lectura o el descanso (CONAIC, 2020; UABC, 2023).

Conectividad

El Departamento de Informática y Bibliotecas (DIB) del campus Ensenada, proporciona conectividad a través de fibra óptica de tipo multimodo a la dirección de la Facultad de Ciencias, desde donde se distribuye al resto de los edificios. También cuentan con internet alámbrica con velocidad de subida y bajada de 500 Mbps. A lo anterior se añaden dos redes inalámbricas: EduWifi (con 11 puntos de acceso y capacidad para 70 usuarios cada uno) y UABC Extreme (para 250 usuarios) (CONAIC, 2020).

La Facultad de Ciencias también tienen a su disponibilidad un sistema de telefonía a través de un conmutador KXTDE 200 y 100 extensiones; además tiene acceso a dos líneas directas y 15 líneas mediante conmutador Alcatel OmniPCX Enterprise 4400 (CONAIC, 2020).

2.4.4 Servicios de apoyo

El objetivo específico es describir los servicios de apoyo entre los que se encuentran la administración escolar, los servicios estudiantiles, las becas y la orientación para el tránsito a la vida profesional.

Administración escolar

Para ofrecer un servicio de calidad a los usuarios del Campus Ensenada, se actualizó la certificación del ISO-9001:2015 correspondiente a 12 procesos administrativos en tres Departamentos: Servicios Estudiantiles y Gestión Escolar, Recursos Humanos, y Servicios Administrativos (UABC, 2023b)

Servicios a estudiantes

Los estudiantes de la FC disponen de un seguro facultativo el cual atiende diversas enfermedades y maternidad. Adicionalmente, en la Unidad Académica Punta Morro del campus Ensenada, se ubica un dispensario médico y de primeros auxilios, mismo que es atendido por médicos pasantes del servicio social de la UABC. Entre los servicios que ofrece se enlistan los siguientes: atención de accidentes, heridas, fracturas, emergencias cardiacas, convulsiones, sangrado nasal, dolor de cabeza, desmayo, tensión arterial, monitoreo de azúcar, picaduras de animales e insectos, y reacciones alérgicas (CONAIC, 2020).

Becas estudiantiles

El programa de becas de la UABC otorga apoyos económicos a los estudiantes en condiciones de vulnerabilidad. Estas becas son de diferentes tipos, cada una tiene requisitos específicos y son las siguientes: beca prórroga, crédito, patrocinio, promedio, mérito escolar deportiva, artística, de compensación, posgrado, y de vinculación. La beca de mayor solicitud es la beca prórroga, la cual consiste en el pago diferido de las cuotas de inscripción (UABC, 2023).

Tránsito a la vida profesional

En la UABC el tránsito de los estudiantes a la vida profesional se promueve a través proyectos de vinculación con valor en créditos, prácticas profesionales, intercambios académicos, bolsa de trabajo, ferias de empleo, talleres de emprendedores, e incubación de empresas. En este mismo sentido, en la UABC se implementó el Modelo de Vinculación e Innovación Inteligente, el cual fomenta la vinculación entre la universidad, industria, gobierno, sociedad y el medio ambiente. Este proyecto obtuvo el premio *Business Management Award 2019*, entregado por la organización estadounidense Global Business Corporation, en el rubro de la innovación educativa (CONAIC, 2020).

La FC tienen un Consejo de Vinculación, integrado por consejeros internos y externos a la misma, en este se generan estrategias de vinculación con el sector productivo y dan seguimiento a las acciones que se proponen al interior de dicho consejo. Actualmente, la FC tiene un padrón de

116 empresas e instituciones en las que se realizan proyectos de vinculación y estancias profesionales. Los convenios vigentes están relacionados con instituciones educativas y de tecnología de la información y desarrollo de software, algunos ejemplos son: Cetmar, Cobach, Probayes, Segity, Bluemeky. Entre los resultados se observa la oportunidad de realizar servicio social, prácticas profesionales, proyectos de vinculación, proyectos de investigación y desarrollo tecnológico, capacitación a docentes del nivel medio superior, y divulgación de la ciencia. De esta manera, la vinculación con el sector productivo ofrece la oportunidad a los estudiantes de realizar actividades técnicas y de investigación en el lugar de trabajo; acceder a la tecnología de vanguardia, así como aplicar y compartir sus conocimientos teóricos (CONAIC, 2020).

El intercambio académico es otra manera de promover el tránsito de los estudiantes hacia la vida profesional y la formación integral. El programa de LCC cuenta con diversos convenios de movilidad académica, ya sea con instituciones nacionales o internacionales, por ejemplo, en periodo 2018-1 algunos estudiantes realizaron movilidad académica con la Universidad de Granada, Universidad de Coruña, University of Regina, Universidad Autónoma de Nuevo León, y Universidad Autónoma de Puebla; y en el periodo 2020-1 con el Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE) y la Universidad Autónoma de Nuevo León. Los impactos que se generaron con esta actividad fueron las redes de colaboración internacionales, el desarrollo de proyectos de investigación, intercambio de experiencias y conocimientos entre estudiantes y fomento de interés de movilidad en otros universitarios (CONAIC, 2020).

En cuanto al rubro de emprendimiento, la FC ofrece cursos semestrales de emprendimiento y negocios tecnológicos, y se organiza la conferencia denominada “Redacción de patentes y registro de marcas”. En la UABC campus Ensenada se promueven cursos de búsquedas tecnológicas, y registro de marcas, diversos foros de vinculación, la “Feria de Emprendedores”, así como las funciones y apoyos otorgados a través de la incubadora Cimarrones Emprendedores (UABC, 2023).

Conclusiones Estudio 2.4 Evaluación del personal académico, infraestructura y servicios

El nivel de estudios del personal académico se distribuye de la siguiente manera: 7 de ellos (43.8%) cuentan con maestría, nueve (56.2%) con doctorado. Todos los docentes se desempeñan en el sector profesional público.

La producción académica en el programa de LCC durante el periodo 2017-2020 fue de 25 materiales didácticos, dos materiales de divulgación, un libro de texto, ocho proyectos, una tesis de maestría, seis de doctorado, una patente, cinco prototipos, 10 artículos en revistas indexadas, 11 memorias en congresos, tres capítulos de libro y un libro (CONAIC, 2020).

En la FC, las formas de organización para el trabajo académico se fundamentan en el Estatuto del Personal Académico de la UABC, manuales, reglamentos de organización, organigrama, y en su propio reglamento interno, el cual establece las funciones y actividades a desempeñar por todo el personal. Asimismo, en el programa de LCC el trabajo académico se realiza mediante dos cuerpos colegiados (CONAIC, 2020).

En lo referente a la infraestructura, el programa de LCC cuenta con 13 aulas con condiciones óptimas para el desarrollo de las actividades académicas. Adicionalmente, en el programa de LCC se dispone de 10 laboratorios equipados con materiales para distintos fines.

La comunidad académica del programa de LCC tiene a su disponibilidad un sistema de bibliotecas físicas ubicadas en cada campus de la UABC, institutos, unidades académicas y facultades. La UABC les ofrece también una biblioteca digital con editoriales, revistas, un repositorio institucional y distintas bases de datos.

La infraestructura de la FC cuenta con diversos espacios para realizar el trabajo académico – aulas, cubículos, sala de maestros, sala de audiovisuales, aula para asesorías, laboratorios, salas de cómputo para clases—. Los eventos académicos y culturales se realizan en distintos espacios como el Teatro Universitario Benito Juárez, la Sala Audiovisual Francisco Zarco, las salas de audiovisual 1 y 2 de la FC, el Andador Universitario de la UABC campus Ensenada, la sala de usos múltiples y la sala de audiovisual ubicadas en el Departamento de Informática y Bibliotecas del mismo campus (CONAIC, 2020).

La seguridad de las instalaciones se mantiene a través del Sistema Integral de Seguridad Universitaria (SISU), quien se encarga de la contratación de una empresa de seguridad privada. Aunado a ello, la FC integra (con personal académico y administrativo) una brigada de multifuncional de emergencias, con el objetivo de prevenir y atender diversos eventos de protección civil. En la FC se disponen de rampas de acceso en los pasillos del primer piso para las personas discapacitadas. En lo referente a espacios deportivos, de recreación y convivencia, los estudiantes de esta facultad tienen a su disposición el gimnasio universitario de la Unidad Punta Morro, el de la

Unidad Valle Dorado, y el centro deportivo de la UABC; y espacios destinados para la recreación y la lectura como el jardín botánico de la FC (CONAIC, 2020; UABC, 2023).

La FC cuenta con dos salas de audiovisuales y tienen a su disponibilidad, con previa reservación en el Departamento de Informática y Bibliotecas del campus Ensenada, cinco salas de cómputo para clases y una sala general equipada con 84 computadoras destinada para la realización de tareas y trabajos académicos.

El servicio médico se proporciona a los estudiantes a través de un seguro facultativo. Aunado a ello, en la Unidad Académica Punta Morro del campus Ensenada de la UABC se encuentra un dispensario médico y de primeros auxilios, en el cual se atienden diversas emergencias médicas (CONAIC, 2020).

En cuanto a las becas, la UABC ofrece una amplia gama de opciones para sus estudiantes, la beca de mayor solicitud es la beca prórroga que consiste en el pago diferido de las cuotas de inscripción durante el semestre en curso (UABC, 2023).

El tránsito a la vida profesional se promueve a través de proyectos de vinculación con valor en créditos, prácticas profesionales, intercambios académicos con universidades nacionales e internacionales, talleres y cursos semestrales asociados al emprendimiento, incubación de empresas, ferias de empleo y la bolsa de trabajo de la UABC. Asimismo, en esta universidad se implementó el Modelo de Vinculación e Innovación Inteligente; se cuenta con el Consejo de Vinculación, el cual genera estrategias de vinculación con el sector productivo (CONAIC, 2020).

Conclusiones generales de la evaluación interna del programa de LCC

Fundamentos y condiciones de operación del programa educativo

- La estructura organizacional de la FC se elabora con base en los propósitos compartidos con la UABC.
- En el programa de LCC se requiere que el Coordinador(a) obtenga mayores recursos para el desarrollo de las actividades de enseñanza-aprendizaje o que no se le considere como el encargado(a) de manejar recursos.
- En el programa de LCC, la competencia considerada como punto fuerte es el diseño y desarrollo de soluciones computacionales que modelen y simulen procesos de las ciencias. Sin embargo, se deben reforzar el resto de las competencias consideradas en el plan de estudios.
- En el programa de LCC, el Servicio Social Profesional aporta mucho al fortalecimiento de la formación profesional de los estudiantes.

Currículo

- En el programa de LCC, el plan de estudios en la etapa disciplinar es la de mayor pertinencia.
- En el programa de LCC el plan de estudios promueve la capacidad analítica y lógica, así como el desarrollo del pensamiento crítico. Sin embargo, se requiere promover la aplicación de los conocimientos adquiridos durante la carrera; las habilidades para el manejo de métodos y técnicas de trabajo en su área profesional; y los espacios curriculares para utilizar las TIC.
- En el programa de LCC los espacios curriculares son oportunos para la formación de los valores universales; y se requiere promover los conocimientos técnicos de la disciplina.
- En el programa de LCC la coordinación de la carrera apoya a los docentes con la planificación y ejecución de los programas de las UA que imparten.
- En el programa de LCC se han hecho cambios suficientes a los contenidos temáticos. Aunque, es necesario un mayor seguimiento y evaluación del cumplimiento de los mismos por parte de la coordinación de la carrera.

- En el programa de LCC el nivel y formación exigido en el plan de estudios respecto al idioma inglés es suficiente según las demandas laborales del campo profesional de la carrera, si bien se demandan más espacios curriculares para el desarrollo de esta habilidad, así como considerar en los perfiles de ingreso y egreso el nivel de habilidad requerido en comunicación oral y escrita.
- En el programa de LCC se promueven las actividades investigativas y científicas en los estudiantes.
- En el programa de LCC es conveniente aumentar la tecnología de la enseñanza para la mejora el proceso de enseñanza-aprendizaje.

Trayectoria escolar de los estudiantes en el programa educativo

- En el programa de LCC existe y se aplica el proceso de ingreso al programa educativo.
- En el programa de LCC el nivel con mayor rezago por reprobación es el tres, por lo tanto, es necesario identificar y revisar el contexto pedagógico de las UA que tienen esta condición.
- En el programa de LCC, las cohortes 2018-2 y 2020-1 mostraron 4 bajas académicas y la cohorte 2017-1 presentó 4 estudiantes potenciales a baja académica.
- En el programa de LCC las actividades de asesoría y apoyo académico para los estudiantes son adecuadas, sin embargo, se requiere reforzar las materias identificadas con mayor reprobación.
- En el programa de LCC considerar otras estrategias de apoyo a estudiantes en condición de vulnerabilidad.
- En el programa de LCC promover el intercambio y movilidad estudiantil nacional e internacional.
- En el programa de LCC fortalecer y promover el servicio de tutoría comunitaria.
- En el programa de LCC el Departamento de Orientación Educativa y Psicopedagógica debe realizar más actividades en beneficio del proceso formativo de los estudiantes, para ello se requiere analizar la cantidad de puestos destinados para este cargo.
- En el programa de LCC las prácticas profesionales contribuyen con el fortalecimiento profesional de los estudiantes y es conveniente dar seguimiento a su experiencia.
- El programa de LCC forma parte de los programas de alto rendimiento académico del EGEL-Ceneval.

- En el programa de LCC mejorar las competencias contextualizadas al campo laboral, así como las habilidades prácticas.

Personal académico, infraestructura y servicios

- En el programa de LCC promover los estudios de posgrado en los académicos.
- En el programa de LCC promover en los académicos cursos de formación pedagógica que brinda la UABC.
- En el programa de LCC analizar la idoneidad de los laboratorios, acervo bibliográfico, salas de audiovisuales, cañones de video y oficinas.
- En el programa de LCC el acceso al segundo piso es limitado para personas con discapacidad.

Referencias

- Asociación Nacional de Universidades e Instituciones de Educación Superior. (2023). *Anuarios Estadísticos de Educación Superior: Ciclo escolar 2021-2022*. ANUIES
<http://www.anui.es.mx/informacion-y-servicios/informacion-estadistica-de-educacion-superior/anuario-estadistico-de-educacion-superior>
- Association for Computing Machinery. (2024). *Advancing Computing as a Science & Profession*.
<https://www.acm.org/>
- Banco Interamericano de Desarrollo. [BID]. (2019). *El futuro del trabajo en América Latina y el Caribe. ¿Cuáles son las ocupaciones y habilidades más demandadas en la región?*
<http://dx.doi.org/10.18235/0001678>
- Banco Mundial. (2023). *Empleo y desarrollo: Descripción general*.
<https://www.worldbank.org/en/topic/jobsanddevelopment/overview>
- Benemérita Universidad Autónoma de Puebla [BUAP]. (2022). *Licenciatura en Ciencias de la Computación*. <https://www.cs.buap.mx/LCC.php>
- Berkeley University. (2023). *Computer Science Bachelor of Arts*.
<https://eecs.berkeley.edu/academics/undergraduate/cs-ba>
- Carnegie Mellon University. (2023). *Course Catalog 2022-2023*.
<http://coursecatalog.web.cmu.edu/pdf/2022-2023-Catalog.pdf>
- Centro Nacional de Evaluación para la Educación Superior. (2023). *Guía para el sustentante: Examen General para el Egreso de la Licenciatura en Ciencias Computacionales. EGEL Plus COMPU*. <https://online.flippingbook.com/view/783413193/10/>
- Centro Nacional de Evaluación para la Educación Superior. (2021, 15 de octubre). *Sobre Ceneval*.
https://ceneval.edu.mx/sobre_el_ceneval-perfil_institucional/
- Cinvestav (s.f.). *50 años de la computación en México y 25 años de la computación en el Cinvestav*. Centro de Investigación y Estudios Avanzados del Instituto Politécnico Nacional (CINVESTAV). <https://www.cs.cinvestav.mx/SemanaComputoCINVESTAV/Computo.html>
- Consejo Nacional de Acreditación de Informática y Computación, A.C. [CONAIC]. (2017). *Marco de referencia para la acreditación de programas educativos de Informática y Computación*.
<https://www.conaic.net/publicaciones/4-MARCO%20DE%20REFERENCIA%20CONAIC%20ES%20Y%20TSU%202018.pdf>
- Consejo Nacional de Acreditación de Informática y Computación [CONAIC]. (2020). *Formato para la Autoevaluación. Licenciatura en Ciencias Computacionales. Plantel Ensenada. Universidad Autónoma de Baja California*. CONAIC.
- Consejo Nacional de Acreditación de Informática y Computación, A.C. [CONAIC]. (2022). *Programas acreditados*. <https://www.conaic.net/acreditados.html>

- Consejo Nacional de Población [CONAPO]. (2019). *Proyecciones de la población de México y de las entidades federativas. Baja California*. Secretaría de Gobernación.
https://www.gob.mx/cms/uploads/attachment/file/487402/02_BCN.pdf
- Consejo Nacional de Evaluación de la Política de Desarrollo Social [CONEVAL]. (2022). *Medición multidimensional de la pobreza en México, 2016-2020*.
https://www.coneval.org.mx/Medicion/MP/Documents/MMP_2018_2020/Pobreza_multidimensional_2016_2020_CONEVAL.pdf
- Corporación Latinobarómetro (2021). *Informe Latinobarómetro 2021*. Santiago de Chile.
<https://www.latinobarometro.org/lat.jsp>
- Consejo para la Acreditación de la Educación Superior. (2023). *¿Qué es COPAES?*
<https://www.copaes.org/copaes.html#mision>
- Dell Technologies (2018). *Realizing 2030: A divided vision of the future*.
<https://www.delltechnologies.com/content/dam/delltechnologies/assets/perspectives/2030/pdf/Realizing-2030-A-Divided-Vision-of-the-Future-Summary.pdf> . [Consultado el 24 de octubre 2023].
- Deloitte (2022). *Striving for balance, advocating for change. The Deloitte global 2022 Gen Z & millennial survey*. <https://www.deloitte.com/content/dam/assets-shared/docs/deloitte-2022-genz-millennial-survey.pdf>
- Diario Oficial de la Federación [DOF]. (2019). *Estrategia Digital Nacional 2019-2024*. Cámara de Diputados.
https://dof.gob.mx/nota_detalle.php?codigo=5628886&fecha=06/09/2021#gsc.tab=0
- Diario Oficial de la Federación [DOF]. (2023). *Constitución Política de los Estados Unidos Mexicanos*. Cámara de Diputados.
<https://www.diputados.gob.mx/LeyesBiblio/pdf/CPEUM.pdf>
- EVb Consulting & Learning. (2023). *Megatendencias 2023: Hacia dónde vamos*.
<https://ebvconsultores.com/megatendencias-2023-hacia-donde-vamos/> [Consultado el 23 de octubre 2023].
- Ferreira, M. M., Avitabile, C., Botero, J., Haimovich, F. y Urzua, S. (2017). *Momento decisivo. América Latina y el Caribe. Resumen*. Banco Mundial.
<https://openknowledge.worldbank.org/bitstream/handle/10986/26489/211014ovSP.pdf?sequence=5&isAllowed=y>
- Forbes. (2020). *Tres tendencias de talento que definirán el futuro laboral*.
<https://www.forbes.com.mx/tres-tendencias-de-talento-que-definiran-el-futuro-laboral/>
- Fundación SM. (2019). *Encuesta de jóvenes en México 2019*. México Fundación SM, A.C.
http://www.voluntarios.cij.gob.mx/src/recursos/publicaciones/Encuesta_de_j%C3%B3venes_en_M%C3%A9xico_2019.pdf

- Gobierno de México (2019). *Plan Nacional de Desarrollo 2019-2024*.
<https://www.gob.mx/agricultura/documentos/plan-nacional-de-desarrollo-gobierno-de-mexico-2019-2024>
- Gobierno de México. (2023). *Tendencias del empleo profesional: primer trimestre 2023*.
Observatorio laboral-Servicio Nacional del Empleo-Gobierno de México.
https://www.observatoriolaboral.gob.mx/static/estudios-publicaciones/Tendencias_empleo.html
- Gobierno del Estado de Baja California (2022). *Plan Estatal de Desarrollo de Baja California 2022-2027*. Comité de Planeación para el Desarrollo del Estado de Baja California.
<https://www.bajacalifornia.gob.mx/Documentos/coplade/PED%20BC%20Completo%20110522.pdf>
- Hays. (2023). *Hallazgos del mercado*. <https://www.hays.com.mx/hallazgos-del-mercado/articulo/transformacion-digital-en-las-empresas-elementos-esenciales-que-necesitas-saber> [Consultado el 05 de octubre 2023].
- Hernández-Laos [Coord.]. (2012). *Mercado laboral de profesionistas en México*. ANUIES.
- IEEE. (2023). *Technology Predictions 2023*. IEEE Computer Society. <https://ieeecs-media.computer.org/media/tech-news/tech-predictions-report-2023.pdf>
- INEGI (2019). *Sistema Nacional de Clasificación de Ocupaciones 2019*. Instituto Nacional de Estadística y Geografía.
https://www.snieg.mx/Documentos/Normatividad/Vigente/SINCO_2019.pdf
- Instituto Nacional de Estadística y Geografía [INEGI]. (2020). *Demografía y sociedad*.
<https://www.inegi.org.mx/temas/>
- Instituto Nacional de Estadística y Geografía [INEGI]. (2021). *Anuario estadístico y geográfico por entidad federativa 2021*.
https://www.inegi.org.mx/contenidos/productos/prod_serv/contenidos/espanol/bvinegi/productos/nueva_estruc/889463904847.pdf
- Instituto Nacional de Estadística y Geografía [INEGI]. (2022a). *Encuesta Nacional de Seguridad Pública Urbana. Cuarto trimestre 2022*.
https://www.inegi.org.mx/contenidos/programas/ensu/doc/ensu2022_diciembre_presentacion_ejecutiva.pdf
- Instituto Nacional de Estadística y Geografía [INEGI]. (2022b). *Conociendo las empresas de consultoría*.
https://www.inegi.org.mx/contenidos/productos/prod_serv/contenidos/espanol/bvinegi/productos/nueva_estruc/889463907138.pdf
- LinkedIn (2023). *Empleos en auge en 2023*. <https://www.linkedin.com/pulse/empleos-en-auge-2023-de-linkedin-los-15-puestos-/?originalSubdomain=es>

- Ministerio de Educación. (2022). *Sistema Nacional de Información para la Educación Superior en Colombia*. <https://hecaa.mineducacion.gov.co/consultaspublicas/programas>
- Ministerio de Educación de Argentina. Departamento de Información Universitaria. (2023). *Guía de Carreras –Universitarias. Carreras de Pregrado y Grado*. https://guiadecarreras.siu.edu.ar/carreras_de_pregrado_y_grado.php
- Ministerio de Educación de Chile. (2023). *¿Dónde y qué estudiar? Subsecretaría de Educación Superior*. <https://www.mifuturo.cl/buscador-de-carreras/?tipo=carrera>
- Ministerio de Educación de España. (2023). *Qué estudiar y dónde en la universidad*. <http://siiu.universidades.gob.es/QEDU/>
- Naciones Unidas. (2015b). *Transformar nuestro mundo: la Agenda 2030 para el desarrollo sostenible*. <https://documents-dds-ny.un.org/doc/UNDOC/GEN/N15/291/93/PDF/N1529193.pdf?OpenElement>
- Naciones Unidas (2019). Noticias ONU. *¿Qué puede hacer la tecnología en beneficio del desarrollo?* <https://news.un.org/es/story/2019/06/1457461>
- Naciones Unidas. (2024). *ONU-Habitat*. <https://onuhabitat.org.mx/index.php/ya-somos-8-mil-millones-de-personas>
- Observatorio Laboral (2023). *Información estadística para el futuro académico y laboral de México*. <https://www.observatoriolaboral.gob.mx/#/> [consultado 06 de octubre 2023].
- Organización para la Cooperación y Desarrollo Económicos [OECD]. (2022). México. in *Education at a Glance 2022: OECD Indicators*. OECD, https://www.oecd-ilibrary.org/education/education-at-a-glance-2022_de781390-es
- Oropeza-García, A. [Coord.]. (2019). *Desarrollo industrial 2050. Hacia una industria del futuro*. UNAM-Instituto de Investigaciones Jurídicas.
- Oxford Internet Institute [OII]. (s.f.). *Dar sentido a la inteligencia artificial*. <https://atozofai.withgoogle.com/intl/en-US/> [Consultado en 16 de octubre del 2023].
- Quacquarelli Symons. (2023). *QS World University Rankings 2024: Top Global Universities*. <https://www.topuniversities.com/university-rankings/world-university-rankings/2024>
- Stanford University. (2019). *Undergraduate Computer Science*. [Consultado en 13 de julio del 2023]. <https://cs.stanford.edu/degrees/undergrad/>
- Serna, A., y Castro, A. (2018). Metodología para realizar estudios que fundamentan la creación, modificación o actualización de programas educativos. En *Metodología de los estudios de fundamentación para la creación, modificación y actualización de programas educativos de licenciatura* (pp. 24-73). Universidad Autónoma de Baja California. http://web.uabc.mx/formacionbasica/documentos/metodologia_con_ficha.pdf

- Universidad Autónoma de Baja California. (2013). *Modelo educativo de la UABC*.
<https://www.uabc.mx/wp-content/uploads/bsk-pdf-manager/2021/04/ModeloEducativodelaUABC.pdf>
- Universidad Autónoma de Baja California. (s.f.). *Manual de tutorías de la Facultad de Ciencias*. Facultad de Ciencias.
- Universidad Autónoma de Baja California. (2017). *Programa educativo de la Licenciatura en Ciencias Computacionales, Plan 2017-1*.
[http://web.uabc.mx/formacionbasica/FichasPE/Lic en Ciencias Computacionales.pdf](http://web.uabc.mx/formacionbasica/FichasPE/Lic_en_Ciencias_Computacionales.pdf)
- Universidad Autónoma de Baja California. (2020). *Documento ejecutivo para la elaboración de la Evaluación Externa e Interna para la modificación o actualización de Planes de Estudio de Licenciatura de la UABC (Propuesta de instrumentos para estudios)*.
http://web.uabc.mx/formacionbasica/documentos/documento_con_instrumentos.pdf
- Universidad Autónoma de Baja California. (2020). *Estudio de preferencias vocacionales*. UABC.
- Universidad Autónoma de Baja California. (2020). *Teatros*. Coordinación General de Extensión de la Cultura y Divulgación de la Ciencia. <https://cultura.uabc.mx/teatros/>
- Universidad Autónoma de Baja California. (2021a). *Estatuto escolar de la Universidad Autónoma de Baja California*.
[http://sriagral.uabc.mx/Externos/AbogadoGeneral/Reglamentos/Estatutos/03 EstatutoEscolarUABC Reforma May 202021.pdf](http://sriagral.uabc.mx/Externos/AbogadoGeneral/Reglamentos/Estatutos/03_EstatutoEscolarUABC_Reforma_May_202021.pdf)
- Universidad Autónoma de Baja California. (2021b). *Sistema de indicadores y estadísticas institucionales*. UABC. <http://indicadores.uabc.mx/indicadores/alumnos/>
- Universidad Autónoma de Baja California. (2021c). *Coordinación General de Informática y Bibliotecas*. <https://bibliotecas.uabc.mx/acerca-de-la-cgib/>
- Universidad Autónoma de Baja California. (2021d). *Reglamento General de Bibliotecas de la UABC. Gaceta Universitaria*. https://bibliotecas.uabc.mx/wp-content/uploads/2021/02/10_REGL_BIBLIO-1.pdf
- Universidad Autónoma de Baja California. (2021e). *Manual de Organización y Procedimientos. Facultad de Ciencias*. UABC.
- Universidad Autónoma de Baja California. (2021f). *Plan de Desarrollo 2020-2024. Facultad de Ciencias*. UABC. <https://ciencias.ens.uabc.mx/documentos/planes/PDFC2020-2024.pdf>
- Universidad Autónoma de Baja California. (2022a). *Cuarto informe de actividades Dr. Daniel Octavio Valdés Delgadillo 2019-2023*. UABC.
<http://planeacion.uabc.mx/documentos/informe/informe2022/Informe-de-actividades-2022.pdf>
- Universidad Autónoma de Baja California. (2022b). *Acervo bibliográfico*. Coordinación General de Informática y Bibliotecas. UABC. <https://bibliotecas.uabc.mx/wp-content/uploads/2022/12/Acervo-bibliografico.pdf>

- Universidad Autónoma de Baja California. (2023a). *Segundo informe de actividades. Facultad de Ciencias 2021-2022*. UABC.
<https://ciencias.ens.uabc.mx/documentos/informes/SegundoInformeActividades2021-2022.pdf>
- Universidad Autónoma de Baja California. (2023b). *Cuarto informe de actividades 2019-2023. Vicerrectoría Campus Ensenada*. UABC.
http://www.ens.uabc.mx/documentos/Informe_2019_2023_Vicerrectoria_Campus_Ensenada.pdf
- Universidad Autónoma de Baja California. (2020). *Estudio de preferencias vocacionales y demanda de carreras profesionales de los estudiantes de educación media superior de Baja California*. UABC.
- Universidad Autónoma de Baja California. (2023c). *Indicadores de seguimiento e-trayecto*. [Consultado en 02 de noviembre del 2023]. <https://sindi.uabc.mx/etrayecto-web/app?opcion=seguimiento-etrayecto>
- Universidad Autónoma de Baja California. (2023d). *Histórico de becas otorgadas*. Coordinación General de Servicios Estudiantiles y Gestión Escolar. UABC.
<http://cgsege.uabc.mx/web/cgsege/becas>
- Universidad Autónoma de Baja California. (2024). *Misión y Visión*. UABC.
<https://www.uabc.mx/mision-y-vision/>
- Universidad Autónoma de Nuevo León. (2023). *Licenciatura en Ciencias Computacionales*. [Consultado en 18 de julio del 2023]. <https://www.uanl.mx/oferta/licenciado-en-ciencias-computacionales/>
- Universidad Autónoma de Yucatán. (2023). *Licenciatura en Ciencias de la Computación*.
https://www.matematicas.uady.mx/planes-de-estudio/licenciaturas/licenciatura-en-ciencias-de-la-computacion?_gl=1*1fqueyo*_ga*NjEyNTA0MzA1LjE2NzkwODc4NTA.*_ga_59HXHG_MZL4*MTY4NTQ3OTYzMS4xLjEuMTY4NTQ3OTkxMi4wLjAuMA..
- Universidad de Guadalajara. (2020). *Ingeniería en Computación*. [Consultado en 19 de julio del 2023]. <http://guiadecarreras.udg.mx/ingenieria-en-computacion/#tabla3>
- Universidad de Sonora [UNISON]. (2023). *Licenciatura en Ciencias Computacionales*.
<https://ofertaeducativa.unison.mx/division-de-ciencias-exactas-y-naturales/licenciatura-en-ciencias-de-la-computacion/>
- Universidad Nacional Autónoma de México [UNAM]. (2023). *Licenciatura en Ciencias de la Computación*. [Consultado en 18 de julio del 2023].
<https://www.fcencias.unam.mx/estudiar-en-ciencias/estudios/licenciaturas/ccomputacion>
- Massachusetts Institute of Technology. (2023). *Computer Science. Curriculum*. [Consultado en 12 de julio del 2023]. <https://www.eecs.mit.edu/academics/undergraduate-programs/curriculum/6-3-computer-science-and-engineering/>

- University of Oxford. (2023). *Computer Science*. [Consultado en 18 de julio del 2023].
<https://www.ox.ac.uk/admissions/undergraduate/courses/course-listing/computer-science>
- Vallaes, F. (2014). La responsabilidad social universitaria: Un nuevo modelo universitario contra la mercantilización. *Revista Iberoamericana de la Educación Superior*, 12(5), 105-117.
<http://www.scielo.org.mx/pdf/ries/v5n12/v5n12a6.pdf>
- World Economic Forum. (2021). *Top 10 tech trends that will shape the coming decade, according to McKinsey*. <https://www.weforum.org/agenda/2021/10/technology-trends-top-10-mckinsey/>
- World Economic Forum. (2023). *Future of Jobs Report 2023*.
https://www3.weforum.org/docs/WEF_Future_of_Jobs_2023.pdf

Apéndices

Apéndice A. Cuestionario dirigido a académicos para retroalimentar la evaluación externa e interna del programa educativo de Licenciado en Ciencias Computacionales



Universidad Autónoma de Baja California

Cuestionario dirigido a académicos para retroalimentar la evaluación externa e interna del programa educativo de Licenciado en Ciencias Computacionales

Estimado académico,

Le informamos que la UABC desea identificar su percepción respecto al proceso de enseñanza-aprendizaje, la organización del plan de estudios, el proceso formativo y el perfil de egreso de las carreras en que presta su servicio.

La información que nos proporciones en este cuestionario es estrictamente confidencial, misma que será de utilidad para realizar cambios en el plan de estudios vigente y con ello incidir en la mejora de la formación profesional que ofrecemos. Te pedimos que respondas de manera honesta y objetiva.

Agradecemos su participación

Instrucciones: Seleccione la opción de respuesta que corresponda a su situación.

I. Datos generales

1.1 Seleccione su sexo

☐ a) Femenino

☐ b) Masculino

1.2 Seleccione su edad

☐ a) 25 años o menos

☐ c) Entre 31 y 35 años

☐ e) Entre 41 y 45 años

☐ g) Entre 51 y 55 años

☐ b) Entre 26 y 30 años

☐ d) Entre 36 y 40 años

☐ f) Entre 46 y 50 años

☐ h) 56 años o más

1.3 Seleccione su máximo nivel de estudios que ha adquirido

☐ a) Licenciatura o ingeniería

☐ c) Doctorado

☐ b) Maestría

☐ d) Postdoctorado

1.4 Actualmente, ¿en qué sector realiza su actividad profesional?

☐ a) Público

☐ b) Privado

☐ c) Ambos

1.5 Indique los años de experiencia profesional con los que cuenta:

☐ a) Entre 1 y 5 años

☐ d) Entre 16 y 20 años

☐ b) Entre 6 y 10 años

☐ e) Mas de 20 años

☐ c) Entre 11 y 15 años

1.6 Indique la facultad en la que labora como docente de UABC:

☐ Facultad de Ciencias

1.7 Indique los años de experiencia como docente en UABC:

☐ a) Entre 1 y 5 años

☐ b) Entre 6 y 10 años

☐ c) Entre 11 y 15 años

☐ d) Entre 16 y 20 años

☐ e) Mas de 20 años

1.8 Indique el tipo de contratación que tiene como docente en UABC:

☐ a) Profesor de carrera de tiempo completo

☐ b) Profesor de carrera de medio tiempo

☐ c) Profesor de asignatura

☐ d) Técnico académico

1.9 Desempeña alguna actividad relacionada con la organización académica de la escuela o facultad en la que labora en UABC:

☐ a) No

☐ b) Coordinador de carrera

☐ c) Coordinador de un área del conocimiento

☐ d) Coordinador del área básica

☐ e) Coordinador de Formación Profesional y Vinculación

☐ f) Coordinador de investigación

1.10 Actualmente, pertenece a alguna de las siguientes instancias (marque las que correspondan):

☐ a) Sí, a un colegio o asociación de mi gremio profesional

☐ b) Sí, al Sistema Nacional de Investigadores de CONACYT

☐ c) No

1.11 ¿En qué medida conoce el programa educativo de la UABC?

☐ a) Nada

☐ b) Poco

☐ c) Regular

☐ d) Completamente

1.12 En los últimos dos años, ¿en cuántos cursos de formación pedagógica o didáctica ha participado?

☐ a) Ninguno

☐ b) Entre 1 y 2

☐ c) Entre 3 y 5

☐ d) Entre 6 y 8

☐ e) Más de 8

1.13 En los últimos tres años, ¿ha participado en el diseño de programas bajo el enfoque por competencias?

☐ a) Sí

☐ b) No

1.14 Seleccione las opciones que reflejen la importancia que otorga a su trabajo como docente en UABC:

☐ a) Autonomía en el trabajo

☐ b) Estabilidad laboral

☐ c) Oportunidad de aprender cosas nuevas

☐ d) Ingresos elevados

☐ e) Afrontar nuevos retos

☐ f) Buenas perspectivas profesionales

☐ g) Tener tiempo para actividades de ocio y recreativas

☐ h) Reconocimiento/prestigio social

☐ i) Oportunidad de hacer algo útil para la sociedad

☐ j) Facilidad para combinar trabajo y familia

☐ k) Otro, especifique: _____

II. Proceso de enseñanza-aprendizaje

2.1 En mi experiencia, como docente de esta Facultad, considero que:

	Opciones de respuesta				
	No tengo bases para opinar	Totalmente en desacuerdo	En desacuerdo	De acuerdo	Totalmente de acuerdo
El personal docente de apoyo es adecuado para lograr los objetivos de los contenidos de la(s) unidad de aprendizaje que imparto					
El personal administrativo que brinda apoyo es adecuado para lograr los objetivos de los contenidos de la(s) unidad de aprendizaje que imparto					
El acervo bibliográfico (libros, revistas, bases de datos electrónicas y tesis) es adecuado, ya que responden a los requerimientos de las unidades de aprendizaje que imparto en la carrera					
El equipo de cómputo y software es adecuado, ya que responde a los requerimientos de las unidades de aprendizaje que imparto en la carrera					
El Internet es adecuado, ya que responde a los requerimientos de las unidades de aprendizaje que imparto en la carrera					
Los cañones de video son adecuados, ya que responden a los requerimientos de las unidades de aprendizaje que imparto en la carrera					
El equipo de laboratorio es adecuado, ya que responde a los requerimientos de las unidades de aprendizaje que imparto en la carrera					
Los salones de clases son adecuados, ya que permiten lograr los objetivos del plan de estudios					
Los laboratorios son adecuados, ya que permiten lograr los objetivos del plan de estudios					
Los talleres son adecuados, ya que permiten lograr los objetivos del plan de estudios					
Las salas audiovisuales son adecuadas, ya que permiten lograr los objetivos del plan de estudios					
La biblioteca es adecuada, ya que permite lograr los objetivos del plan de estudios					
Las oficinas son adecuadas, ya que permiten lograr los objetivos del plan de estudios					
Las salas de consulta y atención a estudiantes son adecuadas, ya que permiten lograr los objetivos del plan de estudios					

2.2 En mi experiencia, como docente de esta institución, considero que:

	Opciones de respuesta				
	No tengo bases para opinar	Totalmente en desacuerdo	En desacuerdo	De acuerdo	Totalmente de acuerdo

Cuento con el apoyo y asesoría para el desarrollo de actividades didácticas en las unidades de aprendizaje que imparto					
Tengo suficientes recursos didácticos o tecnológicos para el desarrollo de las unidades de aprendizaje que imparto					
Se han realizado los cambios y actualizaciones suficientes en los contenidos temáticos de las unidades de aprendizaje que imparto					
Se evalúa de manera sistemática mi trabajo					
Cuento con el apoyo de la coordinación del programa para la planificación y ejecución de los programas de las unidades de aprendizaje que imparto					
La coordinación del programa educativo otorga seguimiento y evaluación del cumplimiento de los contenidos temáticos de las unidades de aprendizaje					
Las actividades de asesoría y apoyo académico para los estudiantes resultan adecuadas					
La coordinación del programa educativo adquiere de manera suficiente recursos para el desarrollo de las actividades de enseñanza-aprendizaje					
La coordinación del programa educativo realiza actividades para la formación de equipos de trabajo por unidades de aprendizaje					
La coordinación del programa educativo contribuye en la capacitación pedagógica					
La coordinación del programa educativo contribuye en la capacitación científica o tecnológica en áreas específicas del plan de estudios					
Las unidades de aprendizaje del plan de estudios son pertinentes, ya que se corresponden con el perfil de egreso de la carrera					
Los créditos y horas otorgadas a las unidades de aprendizaje son pertinentes					
En los grupos de clase, la cantidad de estudiantes es adecuada para el desarrollo de sus actividades didácticas					
La mayoría de los estudiantes cuentan con suficiente información respecto al perfil profesional y plan de estudios de sus carreras					
Se involucra a los estudiantes en actividades de vinculación y actividades de investigación					

2.3 Seleccione las formas de trabajo que suele planear para la impartición de sus clases:

☐

a) Clases teóricas

☐

b) Atención personalizada a los estudiantes

- ☐ c) Seminarios- talleres
☐ e) Clases prácticas
☐ g) Prácticas externas

- ☐ d) Trabajo en grupo
☐ f) Trabajo autónomo e individual
☐ h) Otro(s), especifique: _____

2.4 En sus clases, ¿en qué medida otorga valor o pone en práctica lo siguiente?

	Opciones de respuesta				
	Nada	Muy poco	Poco	Bastante	Mucho
Asistencia a clase					
Puntualidad					
Aprendizaje cooperativo					
Contrato de aprendizaje					
Participación en proyectos de investigación					
Método expositivo					
Estudios de caso					
Resolución de ejercicios y problemas					
Prácticas en empresas instituciones o similares					
Conocimientos prácticos y metodológicos					
Teorías, conceptos o paradigmas					
Aprendizaje orientado en proyectos					
Trabajos escritos					
Aprendizaje basado en problemas					
Exposiciones orales					
Aplicación de pruebas tipo test para medir el aprendizaje					
Aplicación de evaluaciones de preguntas abiertas					
Otro, especifique: _____					

2.5 De los siguientes materiales de apoyo y recursos, indique la frecuencia con la que los utiliza en sus clases:

	Opciones de respuesta				
	Siempre	Casi siempre	Algunas veces	Casi nunca	Nunca
Ilustraciones					
Apuntes					
Pizarrón					
Pizarrón electrónico					
Presentaciones en <i>Power Point</i> o similares					
Estudio de casos concretos					
Material auditivo					
Material audiovisual					
Sitios de internet					
Consulta de bases de datos especializadas					
Otro. Especifique: _____					

2.6 En mis clases, suelo programar actividades para fomentar que los estudiantes sean:

	Opciones de respuesta				
	Siempre	Casi siempre	Algunas veces	Casi nunca	Nunca
Disciplinados					
Propositivos					
Creativos					
Emprendedores					

Respetuosos				
Tolerantes				
Dispuestos al diálogo				
Dispuestos a aprender				
Responsables de su propio aprendizaje				
Abiertos al cambio				
Interesados en encontrar soluciones				
Interesados en superar sus limitaciones				
Interesados en lograr metas				
Sensibles a las manifestaciones del grupo				
Interesados en conocer nuevas formas de enseñanza utilizando las TIC				
Creativos para optimizar recursos				
Interesados por mantenerse informados mediante lectura de periódicos, revistas y libros de interés general				
Más participativos en asuntos de interés público				
Promotores de iniciativas benéficas para la colectividad				
Sensibles a los problemas ambientales				
Respetuosos a los derechos de autor				
Solidarios con los compañeros de clase				

III. Organización del plan de estudios

3.1 Como docente, respecto al plan de estudios actual, considero que:	Opciones de respuesta				
	No tengo bases para opinar	Totalmente en desacuerdo	En desacuerdo	De acuerdo	Totalmente de acuerdo
Es pertinente la secuencia de las unidades de aprendizaje o materias					
Es adecuada la proporción de teoría y práctica					
Es pertinente la etapa de formación básica					
Es pertinente la etapa disciplinar					
Es pertinente la etapa terminal					
Es suficiente la oferta de unidades de aprendizaje, ya que permite que los estudiantes elijan las que son de su interés					
Suelo satisfacer las necesidades de los estudiantes					
Suelo satisfacer las necesidades de la institución					
Suelo satisfacer las demandas del campo laboral					
Tiendo a satisfacer las demandas de los sectores productivos					
Me caracterizo por diversificar la formación de los estudiantes					
Me caracterizo por promover la formación integral					

Me caracterizo por promover conocimientos generales de naturaleza científica
Cumpro con las demandas de la formación de graduados/egresados de la carrera

Opciones de respuesta

3.2 Como docente, considero que el plan de estudios propicia o promueve:

	No tengo bases para opinar	Totalmente en desacuerdo	En desacuerdo	De acuerdo	Totalmente de acuerdo
Conocimientos generales de naturaleza científica					
Conocimientos generales de naturaleza humanística					
Los espacios curriculares necesarios para el desarrollo de la habilidad para la comunicación oral, escrita o gráfica en español					
Los espacios curriculares necesarios para el desarrollo de la habilidad para la comunicación oral, escrita o gráfica en inglés					
Conocimientos técnicos de la disciplina					
Conocimientos teóricos amplios y actualizados del campo profesional					
Capacidad analítica y lógica					
Desarrollo del pensamiento crítico					
Toma de decisiones y resolución de problemas					
Capacidad para aplicar conocimientos adquiridos durante la carrera					
Los espacios curriculares para el desarrollo de la autonomía					
Habilidad para el manejo de métodos y técnicas de trabajo en su área profesional					
Los espacios curriculares para utilizar las TIC					
Espacios curriculares para la formación de los valores universales					

3.3 Del siguiente listado de valores (que conforman el perfil de egreso de la carrera en la cual imparte clases) señale qué tan pertinente es que los estudiantes cuenten con ellos para un adecuado ejercicio profesional:

Opciones de respuesta

	No tengo bases para opinar	Nada pertinente	Poco pertinente	Pertinente	Totalmente pertinente
Respeto					
Honestidad					
Asertividad					
Responsabilidad social					

Tolerancia					

3.4 Como docente, considero que el perfil de egreso de la carrera en la cual participo corresponde o atiende:

Opciones de respuesta

	No tengo bases para opinar	Totalmente en desacuerdo	En desacuerdo	De acuerdo	Totalmente de acuerdo
La gestión de soluciones innovadoras en diversos entornos, derivadas del análisis, modelado, diseño, implementación y evaluación de proyectos complejos para la automatización de procesos					

3.5 De las competencias que conforman el perfil de egreso del programa de Licenciado en Ciencias Computacionales, valore los puntos fuertes y puntos débiles en la formación de los estudiantes	Puntos fuertes	Puntos débiles	No tengo bases para opinar
Gestionar y desarrollar software aplicando metodologías acordes a estándares nacionales e internacionales para satisfacer las demandas del mercado con sentido ético, visión empresarial y responsabilidad social.			
Administrar los recursos de cómputo y telecomunicaciones, utilizando los protocolos y estándares de comunicación para garantizar la integridad de la información en las organizaciones con sentido de responsabilidad profesional.			
Diseñar y desarrollar soluciones computacionales que modelen y simulen procesos de las Ciencias, mediante algoritmos y modelos matemáticos que permitan analizar el comportamiento de fenómenos complejos, con creatividad e innovación.			

Otra: especifique: _____

3.6 De la siguiente lista, seleccione las principales causas que impiden que los estudiantes alcancen el perfil de egreso:

	A los estudiantes no les interesa el programa
	La selección de ingreso es inadecuada
	Los estudiantes tienen un rendimiento o implicación en su aprendizaje deficiente
	Actitudes paternalistas por parte de los docentes o de las autoridades educativas
	Los créditos obligatorios son insuficientes
	Falta de actualización del profesorado
	Falta de integración de unidades de aprendizaje
	El programa educativo actual no forma a los estudiantes para cubrir el perfil

Otras, especifique: _____

Opciones de respuesta

	No tengo bases para opinar	Totalmente en desacuerdo	En desacuerdo	De acuerdo	Totalmente de acuerdo
La formación valoral que reciben los estudiantes en su carrera promueve que conformen sus proyectos de vida					
El nivel y formación que se exige en el plan de estudios respecto al idioma inglés es suficiente de acuerdo con las demandas laborales del campo profesional de la carrera					
Las actividades culturales, deportivas, formativas y competitivas que ofrece el programa educativo son suficientes para incidir en la formación integral de los estudiantes					
La función del tutor beneficia la formación académica de los estudiantes					
El servicio social contribuye positivamente en la formación profesional de los estudiantes					
La movilidad estudiantil nacional o internacional son experiencias que contribuyen a la formación profesional de los estudiantes					
Los programas y acciones que emprende el Departamento de Orientación Educativa y Psicopedagógica beneficia el proceso formativo de los estudiantes					

3.8 Por favor, agregue todos los comentarios que considere necesarios para mejorar, ya sea el plan de estudios, las condiciones del docente para un mejor desempeño o la formación profesional del estudiante:

Apéndice B. Cuestionario dirigido a egresados para retroalimentar la evaluación externa e interna del programa educativo de Licenciado en Ciencias Computacionales



Universidad Autónoma de Baja California

Cuestionario dirigido a egresados para retroalimentar la evaluación externa e interna del programa educativo de Licenciado en Ciencias Computacionales

Estimado egresado de la Facultad de Ciencias. Le informamos que la UABC requiere conocer su percepción y satisfacción respecto a la formación profesional recibida.

La información que nos proporcione en este cuestionario es estrictamente confidencial, misma que será utilizada únicamente para valorar el plan de estudios vigente, y con ello incidir en la mejora de la formación profesional que ofrecemos. Le pedimos que responda de manera honesta y objetiva.

Agradecemos su participación

Instrucciones: Seleccione la opción de respuesta que corresponda a su situación.

I. Datos generales

1.1 Seleccione su sexo:

☐

a) Femenino

☐

b) Masculino

1.2 Seleccione su edad:

☐

a) 25 años o menos

☐

c) Entre 31 y 35 años

☐

e) Entre 41 y 45 años

☐

g) Entre 51 y 55 años

☐

b) Entre 26 y 30 años

☐

d) Entre 36 y 40 años

☐

f) Entre 46 y 50 años

☐

h) 56 años o más

1.3 Seleccione su nacionalidad:

☐

a) Mexicana

☐

b) Extranjero, especifique: _____

1.4 Lugar de residencia actual:

☐

a) Baja California

☐

c) Fuera de México, especifique: _____

☐

b) Otro estado de la república mexicana

1.5 Señale su estado civil actual:

☐

a) Casado(a)

☐

c) Divorciado(a)

☐

b) Soltero(a)

☐

d) Unión libre

1.6 ¿Tiene hijos?

☐

a) Sí

☐

b) No

En caso de sí tener hijos, mencione cuántos: _____

1.7 Indique el año de ingreso a la UABC: _____

1.8 Indique el año de egreso a la UABC: _____

1.9 Correo electrónico: _____

1.10 Matrícula como estudiante de UABC: _____

II. Experiencia laboral

2.1 ¿Se encuentra trabajando actualmente?

☐ a) Sí

☐ b) No

2.2 Si actualmente se encuentra trabajando, indique su situación:

☐ a) En mi trabajo ejerzo mi carrera

☐ b) Mi trabajo no tiene que ver con mi carrera

2.3 Si actualmente no se encuentra trabajando, indique la razón:

☐ a) No tengo necesidad de trabajar

☐ b) No he encontrado un trabajo acorde a la carrera que tengo desde que me gradué

☐ c) Actualmente estoy buscando

Otra (especifique): _____

Nota: Si no está trabajando, pase a la sección “Experiencia académica y expectativas profesionales”

2.4 ¿Trabaja desde que terminó su carrera?

☐ a) Sí

☐ b) No

2.5 Señale su modalidad de trabajo:

☐ a) Trabajo por mi cuenta; es decir, de manera independiente

☐ b) Trabajo como empleado en una empresa o institución

2.6 Señale el sector económico al que pertenece la empresa o institución en que trabaja:

☐ Alimentos y bebidas

☐ Investigación

☐ Industria

☐ Servicios profesionales y técnicos

☐ Medios de comunicación

☐ Educación

☐ Gobierno

☐ Comercio

☐ Productor independiente

☐ Otro, especifique: _____

2.7 Señale el régimen jurídico de la empresa/institución en que trabaja:

☐ a) Público

☐ b) Privado

2.8 Indique el nombre de la empresa, organización o institución en la que trabaja (como empleado o de manera independiente) y describa brevemente las actividades que realiza:

2.9 Indique su ingreso mensual neto actual (incluyendo bonos y prestaciones):

☐ a) Entre 5,000 y 8,000 pesos

☐ b) Entre 9,000 y 12,000 pesos

☐ c) Entre 13,000 y 16,000 pesos

☐ d) Entre 17,000 y 20,000 pesos

☐ e) Más de 20,000 pesos

2.10 Indique si el ingreso que percibe es suficiente para cubrir sus necesidades:

☐ a) Sí

☐ b) No

2.11 Señale el tipo de contratación que tiene en su trabajo actual:

☐ a) Por tiempo indeterminado

☐ b) Por obra/proyecto determinado

☐ c) Por tiempo determinado ☐ d) Otro, especifique _____

2.12 Indique el número de horas que labora a la semana:

☐ a) Menos de 40 horas ☐ b) Entre 40 y 48 horas
☐ c) Más de 48 horas

2.13 ¿Cuántos años tiene laborando en su trabajo actual?

☐ a) Menos de 1 año ☐ b) Entre 1 y 3 años
☐ c) Entre 4 y 6 años ☐ d) Entre 7 y 10 años
☐ e) Más de 10 años

2.14 ¿En qué medida coincide su actividad laboral con sus estudios de licenciatura?

☐ a) Mucho ☐ b) Parcialmente
☐ c) Poco ☐ d) Muy poco

2.15 ¿Tiene posibilidades de ascenso en su trabajo?

☐ a) Sí ☐ b) No
☐ c) No estoy seguro(a)

2.16 Una vez que egresó de la carrera, ¿en cuánto tiempo obtuvo su primer empleo profesional?

☐ a) Menos de 1 año ☐ b) Entre 1 y 2 años
☐ c) Entre 3 y 4 años ☐ d) 5 años o más

2.17 En su opinión, ¿cuál es el nivel de estudios más adecuado para su trabajo actual?

☐ a) No es necesario tener estudios universitarios ☐ b) Licenciatura
☐ c) Maestría ☐ d) Doctorado

III. Experiencia académica y expectativas profesionales

3.1 Actualmente, ¿cuál es su condición de egreso de la licenciatura?

☐ a) Titulado Indique el medio por el que se tituló: Promedio (automático) _____
CENEVAL _____
Otro, especifique: _____
☐ b) Aún no obtengo mi título Especifique la razón: _____

3.2 ¿Conoce los requisitos y opciones de titulación?

☐ a) Sí ☐ b) No ☐ c) No por completo

3.3 Actualmente, ¿tiene planes estudiar algún posgrado?

☐ a) Sí ☐ b) No

3.4 Actualmente, ¿qué nivel de manejo tiene del idioma inglés?

☐ a) Alto ☐ b) Intermedio ☐ c) Bajo
☐ d) Muy bajo ☐ e) Nulo

3.5 ¿Domina otro idioma distinto al inglés y español?

☐ a) Sí Indique cuál: _____ ☐ b) No

3.6 ¿Ha recibido algún reconocimiento o distinción durante su ejercicio profesional?

☐ a) Sí Indique cuál: _____ ☐ b) No

3.7 Actualmente, ¿pertenece a alguna asociación o colegio de su gremio profesional?

☐ a) Sí Indique cuál: _____ ☐ b) No

3.8 ¿Ha residido en el extranjero por motivos de estudio o trabajo?

☐ a) Sí, por motivos de trabajo ☐ b) Sí, por motivos de estudio ☐ c) No

IV. Formación académica y profesional

4.1 ¿Cómo calificaría el servicio que le brindó la Coordinación de Servicio Social de la escuela durante su formación profesional?

☐ a) Muy bueno ☐ b) Bueno ☐ c) Regular
☐ d) Malo ☐ e) Muy malo

4.2 ¿Considera que el servicio social comunitario fortaleció su formación valoral? Es decir, ¿puso en práctica valores como la justicia, libertad, solidaridad, convivencia y respeto?

☐ a) Sí ☐ b) Parcialmente ☐ c) No
☐ d) No estoy seguro

4.3 ¿Considera que, en las actividades de servicio social profesional que realizó, aplicó los conocimientos, habilidades, actitudes y valores que hasta entonces había adquirido en su formación profesional?

☐ a) Sí ☐ b) Parcialmente ☐ c) No
☐ d) No estoy seguro

4.4 ¿El trabajo que desempeña actualmente tiene relación con su servicio social profesional?

☐ a) Sí ☐ b) No

4.5 ¿Conoció los programas de prácticas profesionales y el reglamento de las mismas antes de tomar su decisión sobre dónde realizarlas?

☐ a) Sí ☐ b) No ☐ c) No me acuerdo

4.6 ¿La unidad receptora donde realizó sus prácticas profesionales ya tenía un convenio con la Facultad?

☐ a) Sí ☐ b) No

4.7 Califique su experiencia dentro del programa de prácticas profesionales donde las llevó a cabo

☐ a) Muy buena ☐ b) Buena ☐ c) Regular
☐ d) Mala ☐ e) Muy mala

4.8 Califique el servicio que se brindó por parte de la persona encargada de prácticas profesionales de la escuela

☐ a) Muy bueno ☐ b) Bueno ☐ c) Regular
☐ d) Malo ☐ e) Muy malo

4.9 ¿Considera que la unidad donde realizó sus prácticas profesionales le ayudó a especializarse en el área de su interés?

☐ a) Sí ☐ b) No

4.10 Durante sus estudios, ¿participó en algún estudio de investigación o de desarrollo tecnológico?

☐ a) Sí

☐ b) No

4.11 ¿Realizó alguna acción o actividad de movilidad estudiantil?

☐ a) Sí

☐ b) No

4.12 En la etapa terminal de su carrera, ¿participó en algún proyecto de vinculación con valor en créditos?

☐ a) Sí

☐ b) No

4.13 En caso de haber participado en un proyecto de vinculación en créditos, indique qué tan importante fue para su formación integral:

☐ a) Muy importante

☐ b) Importante

☐ c) Poco importante

☐ d) Nada importante

4.14 Durante sus estudios, ¿participó en ayudantías o estancias de investigación?

☐

☐

4.15 Durante su formación profesional, ¿considera que se fomentó el emprendedurismo?

☐ a) No sabía que el emprendedurismo era parte de mi formación

☐ b) No se fomentó

☐ c) Se fomentó poco

☐ d) Se fomentó, pero de manera limitada

☐ e) Sí se fomentó

☐ f) No tengo idea de qué es el emprendedurismo

4.16 ¿Considera que, para el ejercicio de su carrera profesional, el perfil de emprendedor es relevante?

☐ a) Sí

☐ b) No

4.17 ¿Considera que el nivel de inglés que se le exigió en tu carrera es acorde al que se demanda en el campo laboral de su profesión?

☐ a) Totalmente de acuerdo

☐ b) De acuerdo

☐ c) Ni en acuerdo ni en desacuerdo

☐ d) En desacuerdo

☐ e) Totalmente en desacuerdo

4.18 Seleccione el tipo de actividades en las que participó durante su formación en la carrera:

☐ a) Cursos y talleres culturales

☐ b) Cursos deportivos

☐ c) Cursos para el aprendizaje de un idioma

☐ d) No participé en ninguno

4.19 Durante su formación en la carrera, ¿participó en actividades en las que haya tenido la oportunidad de reflexionar sobre sus valores y su importancia para la convivencia social?

☐ a) Sí

☐ b) No

4.20 En qué medida, ¿las actividades para fomentar los valores en las que participó durante la carrera le han resultado útiles para la conformación de su proyecto de vida?

☐ a) Muy útiles

☐ b) Bastante útiles

☐ c) Poco útiles

☐ d) Nada útiles

4.21 En caso de que haya participado en alguna actividad cultural artística, recreativa o deportiva durante su formación en la carrera, valore cómo fue su experiencia:

☐ a) Muy buena

☐ b) Buena

☐ c) Regular

☐ d) Mala

☐ e) Muy mala

4.22 Indique qué tan satisfecho (a) se encuentra respecto al programa de tutoría académica que

recibió durante su carrera:

☐	a) Totalmente satisfecho
☐	c) Ni satisfecho ni insatisfecho
☐	e) Totalmente insatisfecho

☐	b) Satisfecho
☐	d) Insatisfecho
☐	f) No tengo bases para opinar; no participé en ninguna actividad de tutoría o no identifiqué a mi tutor

4.23 Indique qué tan satisfecho (a) se encuentra respecto a la orientación educativa o psicopedagógica que recibió durante su carrera:

☐	a) Totalmente satisfecho
☐	c) Ni satisfecho ni insatisfecho
☐	e) Totalmente insatisfecho

☐	b) Satisfecho
☐	d) Insatisfecho
☐	f) No tengo bases para opinar; no participé en ninguna actividad de tutoría o no identifiqué a mi tutor

4.24 Seleccione y clasifique las tres materias de la etapa básica que le han resultado más útiles y las tres menos útiles para su desempeño profesional

Listado de materias de la etapa básica	Opciones de respuesta	
	Más útiles	Menos útiles
Comunicación Oral y Escrita		
Diseño de Algoritmos		
Cálculo Diferencial		
Geometría Vectorial		
Álgebra Superior		
Historia e Impacto de la Ciencia		
Formación de Valores		
Introducción a la Programación		
Cálculo Integral		
Álgebra Lineal		
Métodos Experimentales		
Estructura de Datos y Algoritmos		
Programación Orientada a Objetos		
Cálculo Vectorial		
Probabilidad		
Matemáticas Discretas		

4.25 Seleccione y clasifique las tres materias de la etapa disciplinar que le han resultado más útiles y las tres menos útiles para su desempeño profesional

Listado de materias de la etapa disciplinar	Opciones de respuesta	
	Más útiles	Menos útiles
Análisis de Algoritmos		
Metodología de la Programación		
Organización y Arquitectura de Computadoras		
Ecuaciones Diferenciales Ordinarias		
Bases de Datos		
Investigación de Operaciones		
Teoría de Automatas		
Ingeniería de Software		
Sistemas Operativos		

Graficación		
Métodos Numéricos		
Compiladores		
Administración de Proyectos		
Redes de Datos		
Inteligencia Artificial		
Minería de Datos		

4.26 Seleccione y clasifique las tres materias de la etapa terminal que le han resultado más útiles y las tres menos útiles para su desempeño profesional

Listado de materias de la etapa terminal	Opciones de respuesta	
	Más útiles	Menos útiles
Sistemas Distribuidos		
Reingeniería de Procesos		
Seguridad en Cómputo		
Simulación		
Aspectos Legales, Sociales y Éticos de la Computación		
Prácticas Profesionales		

4.27 ¿Considera que la formación interdisciplinaria es útil para el desempeño de su profesión?

☐ a) Totalmente de acuerdo	☐ b) De acuerdo
☐ c) Ni en acuerdo ni en desacuerdo	☐ d) En desacuerdo
☐ e) Totalmente en desacuerdo	

4.28 Considera que gracias a su carrera tiene una buena base para: [puede elegir más de una respuesta]

	Opciones de respuesta				
	Totalmente de acuerdo	De acuerdo	Ni de acuerdo ni en desacuerdo	En desacuerdo	Totalmente en desacuerdo
Empezar a trabajar					
Aprender en el trabajo					
Realizar las tareas de su trabajo actual					
Mejorar sus perspectivas profesionales					
Mejorar su desarrollo personal					
Otra (especifique):					

4.29 Considera que en su carrera. [Puede elegir más de una respuesta]

	Opciones de respuesta				
	Totalmente de acuerdo	De acuerdo	Ni de acuerdo ni en desacuerdo	En desacuerdo	Totalmente en desacuerdo
Había que trabajar o estudiar mucho para aprobar las asignaturas					
El enfoque del plan de estudios es general					

El enfoque del plan de estudios es especializado

Cuenta con prestigio académico

La secuencia de las asignaturas o materias del plan de estudios es adecuada

La proporción de teoría y práctica del programa educativo es adecuada

Otra (especifique):

4.30 ¿Qué sugerencias haría para mejorar el proceso de enseñanza-aprendizaje? [Puede elegir más de una respuesta]

Marque su opción de respuesta para cada enunciado

Aumente	Disminuya	Siga igual

Que el número de horas clase...

Que el número de horas de taller...

Que el número de horas de laboratorio...

Que el material de apoyo...

Que la tecnología de la enseñanza...

Que el área de tutorías...

Otra (especifique):

4.31 ¿En qué medida se hizo énfasis en su carrera en los siguientes métodos de enseñanza y aprendizaje? [Puede elegir más de una respuesta]

Marque su opción de respuesta para cada enunciado

Mucho	Bastante	Poco	Muy poco	Nada

Asistencia a clase

Trabajos en grupo

Participación en proyectos de investigación

Prácticas en empresas, instituciones o similares

Conocimientos prácticos y metodológicos

Teorías, conceptos y paradigmas

El profesor es la principal fuente de información

Aprendizaje basado en proyectos o problemas

Trabajos escritos

Exposiciones orales

Realización de exámenes de preguntas libres

Realización de exámenes de opción múltiple

Otra (especifique):

4.32 ¿Cómo considera la calidad del grupo de docentes que le dieron clases en su carrera?

☐
☐

a) Muy buena

d) Mala

☐
☐

b) Buena

e) Muy mala

☐

c) Regular

4.33 En qué medida el plan de estudios ha contribuido en el desarrollo de sus...

Marque su opción de respuesta para cada enunciado

Mucho	Bastante	Poco	Muy poco	Nada

Competencias profesionales

Competencia de liderazgo

Competencia para el trabajo en equipo

Capacidad para la resolución de conflictos
Habilidades de pensamiento crítico
Creatividad

4.34 ¿Conocía el perfil de egreso de su carrera?

☐ a) Sí

☐ b) No

4.35 Del siguiente listado de habilidades, actitudes y valores que conforman el perfil de egreso de su carrera, seleccione los cinco que considera que más valora su empleador actual. En caso de no estar trabajando, seleccione los que cree que se valorarían más en el trabajo que desea tener:

Descripción de las habilidades, actitudes y valores	Selecciónelo marcando con una X
Habilidades de comunicación oral y escrita en español e inglés	
Manejo nivel básico de tecnologías de la información	
Creatividad	
Toma de decisiones	
Liderazgo	
Trabajo en equipo	
Vocación de servicio, facilidad de adaptación al cambio	
Disposición para el trabajo en equipo	
Emprendedor y comprometido	
Proactivo	
Respeto	
Honestidad	
Asertividad	
Responsabilidad social	
Tolerancia	

4.36 Agradecemos el tiempo que ha dedicado para resolver este cuestionario y apreciamos cualquier opinión adicional que desee agregar y que ayude en la mejora de la formación profesional de la carrera de la cual egresó:

Apéndice C. Cuestionario dirigido a empleadores para retroalimentar la evaluación externa e interna del programa educativo de Licenciado en Ciencias Computacionales



Universidad Autónoma de Baja California

Cuestionario dirigido a empleadores para retroalimentar la evaluación externa e interna del programa educativo de Licenciado en Ciencias Computacionales

Estimado empleador:

En la Facultad de Ciencias de la UABC necesitamos identificar su satisfacción respecto a la formación profesional de los egresados de nuestra institución, así como la pertinencia del plan de estudios ante las demandas laborales.

La información que nos proporcione en este cuestionario es estrictamente confidencial, misma que será utilizada únicamente para valorar el plan de estudios vigente, y con ello incidir en la mejora de la formación profesional que ofrecemos. Le pedimos que responda de manera honesta y objetiva.

Agradecemos su participación

Instrucciones: Seleccione la opción de respuesta que corresponda a su situación.

I. Datos generales

1.1 Seleccione su sexo

☐

a) Femenino

☐

b) Masculino

1.2 Seleccione su edad

☐

a) 25 años o menos

☐

b) Entre 26 y 30 años

☐

c) Entre 31 y 35 años

☐

d) Entre 36 y 40 años

☐

e) Entre 41 y 45 años

☐

f) Entre 46 y 50 años

☐

g) Entre 51 y 55 años

☐

h) 56 años o más

1.3 Seleccione su máximo nivel de estudios que ha adquirido

☐

a) Licenciatura

☐

b) Maestría

☐

c) Doctorado

☐

d) Postdoctorado

1.4 Su empresa u organización pertenece al sector:

☐

a) Público

☐

b) Privado sin ánimo de lucro (Asociación)

☐

c) Privado

☐

d) Otro, especifique: _____

1.5 Indique el nombre de la empresa y organización:

1.6 Lugar donde se ubica el centro de trabajo

Municipio: _____ Estado: _____ País: _____

1.7 Indique su cargo actual: _____

1.8 Refiera el tamaño de la empresa u organización:

☐ a) Micro (1 a 10 empleados)	☐ b) Pequeña (De 11 a 50 empleados)
☐ c) Mediana (De 51 a 250 empleados)	☐ d) Empresa grande (Más de 250 empleados)
☐ e) No aplica	

1.9 Indique la cantidad aproximada de egresados de la UABC del programa de Licenciado en Ciencias Computacionales que prestan servicio social profesional, realizan prácticas profesionales o proyectos de vinculación en su empresa u organización:

☐ a) de 1 a 3	☐ b) de 4 a 6	☐ c) de 6 a 9
☐ d) de 9 a 11	☐ e) más de 11	

II. Pertinencia del plan de estudios

2.1 Indique qué nivel de estudios requiere la empresa u organización de los egresados del programa de Licenciatura en Ciencias Computacionales (seleccione todos los que aplique):

☐ a) Licenciatura	☐ b) Maestría
☐ c) Doctorado	☐ d) Posdoctorado

2.2 ¿Qué tan relevante es que los egresados del programa de Licenciatura en Ciencias Computacionales manejen y dominen el idioma inglés para la relación del ejercicio profesional que realizan en su empresa u organización?

☐ a) Totalmente relevante	☐ b) Relevante
☐ c) Medianeramente relevante	☐ d) Poco relevante
☐ e) Irrelevante	

2.3 Indique la importancia que tienen para la empresa u organización las siguientes actitudes y valores para que los egresados del programa de Licenciatura en Ciencias Computacionales desempeñen de manera óptima su trabajo:

Marca tu opción de respuesta para cada enunciado

	Nada	Poco	Bastante	Mucho
Autonomía en el trabajo	☐	☐	☐	☐
Respeto a la diversidad cultural	☐	☐	☐	☐
Respeto a la autoridad	☐	☐	☐	☐
Buena actitud para afrontar nuevos retos	☐	☐	☐	☐
Responsable con el equipo y la organización del tiempo	☐	☐	☐	☐
Liderazgo en los equipos de trabajo	☐	☐	☐	☐
Cumplir con las metas del trabajo	☐	☐	☐	☐
Puntualidad en el trabajo	☐	☐	☐	☐
Incrementar la competitividad de la empresa	☐	☐	☐	☐
Facilidad para combinar trabajo y familia	☐	☐	☐	☐
Oportunidad de aprender cosas nuevas	☐	☐	☐	☐

Otros, favor de indicar:

2.4 ¿En cuáles de los siguientes aspectos considera que los egresados del programa de Licenciado en Ciencias Computacionales que trabajan en su empresa u organización tienen dominados y cuáles les falta mejorar?

Marca tu opción de respuesta

Dominan Les falta mejorar

Habilidades de comunicación oral y escrita en español e inglés	☐	☐
Manejo nivel básico de tecnologías de la información	☐	☐
Creatividad	☐	☐

Toma de decisiones

Liderazgo

Trabajo en equipo

Otros, favor de indicar: _____

2.5 Refiera en qué medida los egresados del programa de Licenciado en Ciencias Computacionales manifiestan en la realización de su trabajo los siguientes valores:

Marca tu opción de respuesta para cada enunciado

	Nada	Poco	Bastante	Mucho
Respeto				
Empatía				
Justicia				
Honradez				
Responsabilidad				
Lealtad				
Puntualidad				
Flexibilidad				
Tolerancia				
Tenacidad				
Solidaridad				
Perseverancia				
Conciencia ambiental y de sustentabilidad				

Indique qué tan de acuerdo se encuentra con las siguientes afirmaciones:

2.6 La formación profesional de los egresados del programa de Licenciado en Ciencias Computacionales es acorde a las demandas del mercado laboral actual:

☐ a) Totalmente de acuerdo	☐ b) De acuerdo
☐ c) Ni de acuerdo ni en desacuerdo	☐ d) En desacuerdo
☐ e) Totalmente en desacuerdo	

Si está en desacuerdo, indique el por qué: _____

2.7 La formación profesional de los egresados del programa de Licenciado en Ciencias Computacionales es acorde a las demandas del mercado laboral futuro:

☐ a) Totalmente de acuerdo	☐ b) De acuerdo
☐ c) Ni de acuerdo ni en desacuerdo	☐ d) En desacuerdo
☐ e) Totalmente en desacuerdo	

Si está en desacuerdo, indique el por qué: _____

2.8 Los egresados que laboran en su empresa u organización cumplen con el perfil de egreso de la carrera:

El Licenciado en Ciencias Computacionales es un profesional dedicado a gestionar y ofrecer soluciones innovadoras en diversos entornos, derivadas del análisis, modelado, diseño, implementación y evaluación de proyectos complejos para la automatización de procesos que pueden ser tratados con sistemas de

cómputo y asegurando la integridad de la información.

Ello implica que el Licenciado en Ciencias Computacionales egresado será competente para:

- Gestionar y desarrollar software aplicando metodologías acordes a estándares nacionales e internacionales para satisfacer las demandas del mercado con sentido ético, visión empresarial y responsabilidad social.
- Administrar los recursos de cómputo y telecomunicaciones, utilizando los protocolos y estándares de comunicación para garantizar la integridad de la información en las organizaciones con sentido de responsabilidad profesional.
- Diseñar y desarrollar soluciones computacionales que modelen y simulen procesos de las Ciencias, mediante algoritmos y modelos matemáticos que permitan analizar el comportamiento de fenómenos complejos, con creatividad e innovación.

☐

a) Totalmente de acuerdo

☐

b) De acuerdo

☐

c) Ni de acuerdo ni en desacuerdo

☐

d) En desacuerdo

☐

e) Totalmente en desacuerdo

Si está en desacuerdo, indique el por qué:

2.9 Considero que los egresados en el programa de Licenciado en Ciencias Computacionales tienen un nivel de formación en:

Marca tu opción de respuesta para cada enunciado

Pésimo Deficiente Mejorable Aceptable Excelente

Conocimientos teóricos

Habilidades prácticas

Competencias contextualizadas al campo laboral

III. Desempeño laboral

3.1 Actualmente, ¿en su empresa se ofrece un programa de capacitación previo a la contratación de los egresados del programa de Licenciado en Ciencias Computacionales?

☐ a) Sí

☐ b) No

3.2 En la actualidad, considera que la demanda del programa de Licenciado en Ciencias Computacionales es:

☐ a) Alta

☐ b) Media

☐ c) Baja

3.3 En el futuro, considera que la demanda del programa de Licenciado en Ciencias Computacionales será:

☐ a) Alta

☐ b) Media

☐ c) Baja

3.4 ¿En qué medida, el trabajo actual requiere más conocimientos y habilidades de los egresados del programa de Licenciado en Ciencias Computacionales?

☐ a) Nada

☐ b) Poco

☐ c) Bastante

☐ d) Mucho

3.5 En general, indique su grado de satisfacción con el trabajo que los egresados del programa de Licenciado en Ciencias Computacionales desempeñan en la empresa u organización:

☐ a) Muy insatisfecho

☐ b) Insatisfecho

☐ c) Ni satisfecho ni insatisfecho

☐ d) Satisfecho

☐ Muy satisfecho

Por favor, agregue todos los comentarios o sugerencias que, en su opinión, pueden ser de utilidad para favorecer la formación profesional de excelencia de los estudiantes del programa educativo:

Si desea recibir un resumen de los resultados, por favor escriba su correo electrónico:

¡Gracias por su invaluable apoyo!

Apéndice D. Cuestionario dirigido a estudiantes para retroalimentar la evaluación externa e interna del programa educativo de Licenciado en Ciencias Computacionales



Universidad Autónoma de Baja California

Cuestionario dirigido a estudiantes para retroalimentar la evaluación externa e interna del programa educativo de Licenciado en Ciencias Computacionales

Estimado estudiante de la Facultad de Ciencias. Le informamos que la UABC requiere conocer su percepción y satisfacción respecto a la formación profesional recibida.

La información que nos proporcione en este cuestionario es estrictamente confidencial, misma que será utilizada únicamente para valorar el plan de estudios vigente, y con ello incidir en la mejora de la formación profesional que ofrecemos. Le pedimos que responda de manera honesta y objetiva.

Agradecemos su participación

Instrucciones: Seleccione la opción de respuesta que corresponda a su situación.

I. Datos generales

1.1 Seleccione su sexo:

☐

a) Femenino

☐

b) Masculino

☐

c) Prefiero no decir

1.2 Periodo en el que te encuentras inscrito:

☐

a) Primer semestre

☐

c) Quinto semestre

☐

e) Décimo semestre

☐

b) Tercer semestre

☐

d) Séptimo semestre

1.3 ¿Cómo te enteraste del programa de Licenciado en Ciencias Computacionales y qué te ayudó a decidirte a estudiar esta carrera?:

☐

a) Familiares

☐

c) Amistades

☐

e) Radio

☐

g) Televisión

☐

i) Jornada de orientación vocacional

☐

k) Por haber asistido a una conferencia de ciencias computacionales

☐

c) Profesores

☐

d) Internet

☐

f) Periódico

☐

h) Semana de ciencias

☐

j) Plática de difusión de la carrera

☐

Otro:

1.4 Comparándote con tus compañeros de carrera, ¿cómo consideras tu promedio?:

☐

a) Muy bueno

☐

c) Regular

☐

e) Muy malo

☐

b) Bueno

☐

d) Malo

☐

f) Otro:

II. Formación académica y profesional

2.1 Considera que en su carrera.	Opciones de respuesta				
	Totalmente de acuerdo	De acuerdo	Ni de acuerdo ni en desacuerdo	En desacuerdo	Totalmente en desacuerdo
Hay que trabajar o estudiar mucho para aprobar las asignaturas					
El enfoque del plan de estudios es general					
El enfoque del plan de estudios es especializado					
Cuenta con prestigio académico					
Otra (especifique):					

2.2 ¿En qué medida se hizo énfasis en su carrera en los siguientes métodos de enseñanza y aprendizaje? [Puede elegir más de una respuesta]	Opciones de respuesta				
	Mucho	Bastante	Poco	Muy poco	Nada
Asistencia a clase					
Trabajos en grupo					
Participación en proyectos de investigación					
Prácticas en empresas, instituciones o similares					
Conocimientos prácticos y metodológicos					
Teorías, conceptos y paradigmas					
El profesor es la principal fuente de información					
Aprendizaje basado en proyectos o problemas					
Trabajos escritos					
Otra (especifique):					

2.3 ¿Conoce a su tutor?:

☐ a) Sí ☐ b) No

2.3.1 ¿Por qué no lo conoce? [Por favor, especifique su respuesta]

2.4 ¿Cuántas tutorías realiza en promedio al semestre? [Por favor, escriba su respuesta]

2.5 ¿Cómo considera el programa de tutorías? [Elija una sola respuesta]

☐ a) Muy bueno	☐ b) Bueno
☐ c) Regular	☐ d) Malo
☐ e) Muy malo	

2.5.1 ¿Por qué? [Especifique su respuesta anterior]

2.6 ¿Cómo considera la secuencia de asignaturas o materias de su plan de estudios? [Elija una sola respuesta]

☐ a) Muy bueno	☐ b) Bueno
☐ c) Regular	☐ d) Malo
☐ e) Muy malo	

2.7 Indique un máximo de tres competencias que considere “puntos fuertes” y otras que considere “puntos débiles” en su carrera. [Seleccione las entradas y coloque “fuerte” o “débil”, según corresponda].

Listado de materias de la etapa básica	Opciones de respuesta	
	Puntos fuertes	Puntos débiles
Dominio de tu área o disciplina		
Conocimiento de otras áreas o disciplinas		
Comportamiento ético y con responsabilidad social		
Capacidad para adquirir con rapidez nuevos conocimientos		
Capacidad para negociar de forma eficaz		
Capacidad para rendir bajo presión		
Capacidad para detectar nuevas oportunidades		
Capacidad para coordinar actividades		
Capacidad para usar el tiempo de forma efectiva		
Capacidad para movilizar las capacidades de otros		
Capacidad para hacerse entender		
Capacidad para hacer valer su autoridad		
Capacidad para utilizar herramientas informáticas		
Capacidad para encontrar nuevas ideas y soluciones		

Predisposición para cuestionar ideas propias o ajenas		
Capacidad para presentar en público productos, ideas o informes		
Capacidad para redactar informes o documentos		
Conciencia ecológica		

2.8 Considera que gracias a su carrera tiene una buena base para: [puede elegir más de una respuesta]	Opciones de respuesta				
	Totalmente de acuerdo	De acuerdo	Ni de acuerdo ni en desacuerdo	En desacuerdo	Totalmente en desacuerdo
Empezar a trabajar					
Aprender en el trabajo					
Realizar las tareas de su trabajo actual					
Mejorar sus perspectivas profesionales					
Mejorar su desarrollo personal					
Otra (especifique):					

2.9 ¿Cuáles son tus planes al terminar la carrera?

☐	Estudiar un posgrado
☐	Buscar un empleo
☐	Crear mi propio empleo
☐	Dedicarme a la docencia
☐	Otra (especifique):

3.0 Agradecemos el tiempo que ha dedicado para resolver este cuestionario y apreciamos cualquier opinión adicional que desee agregar y que ayude en la mejora de la formación profesional de la carrera que cursa:
