

CONSEJO NACIONAL DE RECTORES

Oficina de Planificación de la Educación Superior

División Académica

DICTAMEN SOBRE LA SOLICITUD DE REDISEÑO DEL BACHILLERATO EN INGENIERÍA EN COMPUTACIÓN DE LA UNIVERSIDAD NACIONAL (con salida lateral de Diplomado en Desarrollo de Software)



Alexander Cox Alvarado

OPES; no. 19-2026

378

C877d

Cox Alvarado, Alexander

Dictamen sobre la solicitud de rediseño del bachillerato en ingeniería en computación de la Universidad Nacional : con salida lateral de diplomado en desarrollo de software / Alexander Cox Alvarado. -- San José, C.R. : CONARE - OPES, 2026.

(OPES; no. 19-2026) 1 recurso en línea (61 páginas): archivos de texto PDF, 750 KB

ISBN 978-9977-77-717-7

1. INGENIERÍA EN COMPUTACIÓN. 2. DESARROLLO DE SOFTWARE. 3. DIPLOMADO UNIVERSITARIO. 4. BACHILLERATO UNIVERSITARIO. 5. PERFIL PROFESIONAL. 6. PLAN DE ESTUDIOS. 7. PERSONAL DOCENTE. 8. UNIVERSIDAD NACIONAL (COSTA RICA). I. Título. II. Serie.

Información del autor

Alexander Cox Alvarado. <https://orcid.org/0009-0001-9724-6175>

Esta obra se comparte bajo la licencia
Reconocimiento – No Comercial – Compartir Igual
(CC-BY-NC-SA)

Permite usar una obra para crear otra obra o contenido, modificando o no la obra original, siempre que se cite al autor, la obra resultante se comparta bajo el mismo tipo de licencia y no tenga fines comerciales



PRESENTACIÓN

El estudio que se presenta en este documento (OPES; no. 19-2026) se refiere al dictamen sobre la solicitud de rediseño del Bachillerato en *Ingeniería en Computación* de la Universidad Nacional.

El dictamen fue realizado por el M.Sc. Alexander Cox Alvarado, investigador de la División Académica de la Oficina de Planificación de la Educación Superior (OPES) con base en los documentos *Plan de estudios Carrera Bachillerato en Ingeniería en Computación, Universidad Nacional, 2026* y *Resumen ejecutivo Carrera Bachillerato en Ingeniería en Computación, Universidad Nacional, 2026*. La revisión del documento estuvo a cargo de la Mag. Johanna Jiménez Bolaños, Jefa a.i. de la División Académica y la edición del documento fue realizada por la señora. Sandra Guillén Guardado, asistente de la División citada.

El presente dictamen fue aprobado por el Consejo Nacional de Rectores en la Sesión Ordinaria No.19-2026, celebrada el 9 de junio de 2026, en el artículo 9, inciso a).



Katalina Perera Hernández
Directora a.i de la OPES

Tabla de Contenido

1. Introducción	1
2. Datos generales	2
3. Objeto de estudio	2
4. Justificación del rediseño de la carrera	3
5. Objetivos de la carrera	4
6. Perfil académico-profesional	5
7. Campo de inserción laboral de los graduados.....	8
8. Carreras afines en la Educación Superior	9
9. Datos de empleabilidad del Observatorio Laboral de Profesiones	11
10. Requisitos de ingreso y permanencia.....	11
11. Requisitos de graduación	11
12. Actividades de formación académica	11
13. Descripción de las actividades de formación académica de la carrera	12
14. Correspondencia del equipo docente con las actividades académicas.....	12
15. Ficha de información para la gestión de datos	12
16. Conclusiones	13
17. Recomendaciones	13
ANEXO A.....	14
ANEXO B.....	17
ANEXO C	55
ANEXO D	58

1. Introducción

La solicitud de rediseño del *Bachillerato en Ingeniería en Computación* con salida lateral en *Diplomado en Desarrollo de Software* en la Universidad Nacional (UNA) fue presentada al Consejo Nacional de Rectores por el Rector Jorge Enrique Herrera Murillo, en nota UNA-R-OFIC-514-2026. El rediseño incluye el cambio de nombre de la carrera de *Ingeniería en Sistema de Información* por el mencionado.

Cuando se crean o rediseñan carreras, ya sea de grado o de posgrado, según lo señalado en el documento *Lineamientos para la creación y el rediseño de carreras universitarias estatales*¹ se estudian los siguientes temas que son la base del estudio que realiza la OPES:

- Datos generales
- Objeto de estudio
- Justificación del rediseño de la carrera
- Objetivos de la carrera
- Perfil académico-profesional
- Campo de inserción laboral del graduado
- Carreras afines en la Educación Superior
- Datos de empleabilidad del Observatorio Laboral de Profesiones
- Requisitos de ingreso y de permanencia
- Requisitos de graduación
- Actividades de formación académica de la carrera
- Descripción de las actividades de formación académica de la carrera
- Correspondencia del equipo docente con las actividades de formación académica.

¹ Aprobado por el Consejo Nacional de Rectores en la sesión N°41-2022 celebrada el 18 de octubre de 2022

2. Datos generales

La apertura de la carrera fue autorizada por el Consejo Nacional de Rectores en la sesión 22-1992, artículo 7, celebrada el 6 de octubre de 1992. Se impartirán dos ciclos por año. El total de ciclos es de ocho. La carrera recibirá estudiantes anualmente y el número de promociones será de 20. Esta carrera se impartirá en los siguientes campus:

- Sede Interuniversitaria de Alajuela
- Sección Regional Sarapiquí
- Campus Liberia
- Campus Nicoya
- Campus Pérez Zeledón
- Campus Coto

El diploma otorgará el siguiente grado y título:

- Diplomado en Desarrollo de Software
- Bachillerato en Ingeniería en Computación.

La Universidad Nacional afirma en el resumen ejecutivo que cuenta con los recursos presupuestarios y financieros (talento humano, infraestructura y equipo) necesarios para ofrecer la carrera.

3. Objeto de estudio

Lo siguiente es lo que la Universidad Nacional presentó sobre el objeto de estudio de la carrera.

Las Ciencias de la Computación son una disciplina formal que estudia los principios, métodos, modelos y tecnologías para representar, procesar, almacenar y comunicar información mediante sistemas computacionales, así como el diseño y construcción de soluciones algorítmicas y de ingeniería que aporten a la solución de problemas de manera eficiente.

La adquisición de conocimientos y habilidades está orientada para analizar, diseñar, implementar y evaluar soluciones en diversos contextos, mediante

el uso ético y responsable de las tecnologías de la información y la comunicación, con el fin de contribuir a la optimización de las organizaciones y la mejora del quehacer humano y del entorno social.

El objeto de estudio de las Ciencias de la Computación es la información y la computación, incorpora los procesos, métodos, técnicas y herramientas que permiten la resolución de problemas mediante sistemas computacionales; abarcando la teoría, el diseño, la implementación y las aplicaciones de sistemas informáticos, tanto de hardware como software.

Este campo trata con la construcción de modelos matemáticos y técnicas de análisis cuantitativos, así como el uso de computadoras para analizar y resolver diversos problemas. Se centra en los procesos algorítmicos para crear, procesar, almacenar, transformar y comunicar información codificada en datos, lenguajes de programación, sistemas operativos; así como el hardware, software y sistemas inteligentes que hacen la interacción entre humanos y computadoras, todo con un enfoque en la automatización y el procesamiento eficiente de datos para resolver problemas complejos.

Esta disciplina abarca el estudio de fundamentos teóricos, técnicas prácticas y aplicaciones en temáticas como los lenguajes de programación, algoritmos, estructuras de datos, análisis de datos, sistemas operativos, inteligencia artificial, infraestructura y ciberseguridad.

Las organizaciones operan procesos que con el avance de la tecnología se hacen más complejos. Se requieren profesionales capaces de encargarse de implementar estrategias que contribuyan con la optimización de los procesos de transformación digital y se hagan más competitivos que los del mercado, para que logren cumplir con las metas y objetivos trazados en la organización.

Los profesionales en Computación deben ser personas creativas, disciplinadas, que dominen las matemáticas, la estadística y conozcan desde la programación hasta el correcto manejo de los procesos digitales de la organización. Esto hace que sean muy solicitados por cualquier tipo de empresa en el mundo contemporáneo de los negocios

La División Académica de la OPES considera que la definición del objeto de estudio de Ingeniería en Computación es clara.

4. Justificación del rediseño de la carrera

Lo siguiente es justificación del rediseño de la carrera:

Dada la creciente internacionalización de la demanda laboral en esta área, es indispensable pensar en un proceso de formación que abra nuevos mercados laborales para los graduados, más allá de las fronteras nacionales. Para abordar esta transformación de manera efectiva, es crucial proporcionar opciones educativas en áreas que respondan a las necesidades actuales de las organizaciones, la industria y el mercado laboral, en las que se promueva el rigor científico junto con habilidades técnicas en matemáticas, estadísticas y tecnológicas, así como el dominio de un segundo idioma.

La carrera de Bachillerato en Ingeniería en Computación con salida lateral al Diplomado en Desarrollo de Software se propone para responder a las tendencias de desarrollo de la profesión y a las demandas del mercado laboral. Se plantean tres áreas de disciplinares de profundización, a saber:

la ingeniería de datos, inteligencia artificial, ciberseguridad. Un área de fundamentación con tres subáreas: matemática, infraestructura computacional y desarrollo de software y el área complementaria que busca la formación integral de los estudiantes al potenciar conocimientos, habilidades y valores que favorezcan el desempeño académico, profesional y personal, no siempre limitándose al área disciplinar.

Entendiendo que las Ciencias de la Computación estudian los fundamentos teóricos, los métodos y las tecnologías para representar, procesar y comunicar información mediante sistemas computacionales. Su práctica contemporánea es sociotécnica: integra teoría, ingeniería y consideraciones éticas, legales y de impacto ambiental y se despliega considerando las subdisciplinas de ingeniería de datos, inteligencia artificial, ciberseguridad, y el ciclo de vida del desarrollo de software entre otras, que son considerados a su vez en la Computer Science Curricula 2023 (CS2023), donde se orientan los planes de estudio y perfiles de egreso de ACM/IEEE, promoviendo la convergencia entre disciplinas.

La División Académica de la OPES considera que la justificación para el rediseño del Bachillerato en Ingeniería en Computación es apropiada y está debidamente fundamentada, pues evidencia la necesidad de formación académica especializada en el área, así como la demanda creciente de personas profesionales capacitadas en ciencia de datos.

5. Objetivos de la carrera

Los objetivos del Diplomado son los siguientes:

Objetivos generales

- Formar diplomados en desarrollo de software con una base sólida en los fundamentos teóricos, metodológicos y técnicos de la disciplina, capaces del diseño y desarrollo de aplicaciones informáticas que respondan a necesidades del entorno.
- Promover el desarrollo de habilidades profesionales que integren la ética, la colaboración y la capacidad de adaptación a tecnologías emergentes, para un ejercicio responsable y actualizado de la disciplina.

Objetivos específicos

- Aplicar los principios de programación, algoritmos y metodologías de desarrollo de software para la construcción de aplicaciones informáticas eficientes, seguras y sostenibles.
- Analizar los principios de arquitectura de computadoras, sistemas operativos, redes y virtualización, como soporte esencial para el diseño y desarrollo de aplicaciones informáticas robustas, seguras y adaptables.
- Desarrollar competencias profesionales que integren la ética, el trabajo colaborativo y la adaptación a tecnologías emergentes, para un ejercicio disciplinar responsable, pertinente y actualizado.

Los objetivos del Diplomado son los siguientes:

Objetivo general

Formar profesionales con una base sólida en los fundamentos teóricos, metodológicos y técnicos de la disciplina, capaces del diseño, desarrollo y evaluación de soluciones que respondan a las necesidades del entorno.

Objetivos específicos

- Aplicar los conocimientos y habilidades de avanzada, como la inteligencia artificial, ciberseguridad y la ingeniería de datos para la resolución de problemas en diversos sectores impulsando la innovación tecnológica.
- Diseñar e implementar sistemas de software siguiendo metodologías actuales (ágiles, iterativas o tradicionales), integrando herramientas de ingeniería apropiadas y considerando criterios de calidad, usabilidad y sostenibilidad.
- Integrar equipos de trabajo colaborativos, asumiendo el liderazgo con pensamiento crítico, disposición al aprendizaje autónomo, iniciativa, ética y responsabilidad profesional.

La División Académica de la OPES estima que los objetivos de la carrera son claros y, además, coherente con el objeto de estudio presentado por la Universidad Nacional.

6. Perfil académico-profesional

La Universidad Nacional envió el siguiente perfil académico-profesional del Diplomado:

Conocimientos (saber -saber)

- Domina los fundamentos matemáticos relevantes para la computación (matemática básica, cálculo aplicado a la computación, estructuras discretas, álgebra lineal), y aplica el método científico como base para el pensamiento computacional.
- Comprende los principios fundamentales de las ciencias de la computación, incluyendo algoritmos, estructuras de datos, arquitectura, fundamentos de programación, diseño de software y gestión de datos.
- Domina los fundamentos teóricos y prácticos del desarrollo de software y la infraestructura computacional.
- Desarrolla aplicaciones móviles mediante el uso correcto de los recursos y herramientas facilitadas por las plataformas para este fin.
- Domina infraestructura computacional para la ejecución de aplicaciones de software, incluyendo la arquitectura de computadores, sistemas operativos, redes de comunicación, computación paralela y distribuida, fundamentos de sistemas y aspectos de seguridad.
- Identifica los principios de los sistemas de gestores de bases de datos abiertos y propietarios, los lenguajes de definición, consulta, manipulación y transacción de datos.

Habilidades y destrezas (saber hacer)

- Analiza problemas de mediana complejidad y formula soluciones computacionales adecuadas, considerando aspectos técnicos, éticos y sociales.
- Aplica el ciclo de vida del desarrollo de software, utilizando metodologías modernas y herramientas de desarrollo.
- Administra infraestructuras computacionales, incluyendo redes, sistemas distribuidos y servicios en la nube.
- Aplica los fundamentos de las Ciencias de la Computación para la programación, diseño de software y gestión de datos.
- Interpreta los conceptos y estructuras lógico-matemáticas en la programación de componentes de software para la resolución de problemas del contexto.
- Demuestra el desarrollo de habilidades de liderazgo para la resolución de los problemas y conflictos que se presentan cotidianamente en las organizaciones.
- Comunica información en forma efectiva tanto oral como escrita, en los idiomas de español e inglés.
- Participa en proyectos colaborativos y multidisciplinarios.

Actitudes y valores (saber ser)

- Actúa con integridad ética y responsabilidad profesional, respetando el impacto social, ambiental y cultural de sus desarrollos tecnológicos.
- Demuestra una actitud de aprendizaje permanente, actualizando sus conocimientos y habilidades conforme evoluciona la disciplina.
- Demuestra compromiso con la calidad, la excelencia y la mejora continua en su práctica profesional.
- Respeta la diversidad y promueve la inclusión en los entornos de trabajo y en la sociedad.
- Desarrolla su capacidad hacia la interdisciplinariedad y la colaboración, reconociendo el valor de múltiples perspectivas en la solución de problemas.
- Aplica la proactividad en la prevención de riesgos, en la promoción de la seguridad y la privacidad en los sistemas que diseña y opera.

La Universidad Nacional envió el siguiente perfil académico-profesional del Bachillerato:

Conocimientos (saber-saber)

- Domina los fundamentos matemáticos relevantes para la computación (fundamentos de matemática, cálculo, estructuras discretas, álgebra lineal) y comprensión del método científico como base para el pensamiento computacional.
- Comprende los principios de las Ciencias de la Computación, incluyendo algoritmos, estructuras de datos, arquitectura, fundamentos de programación, diseño de software e ingeniería de datos para crear soluciones tecnológicas, considerando aplicaciones, sistemas operativos, bases de datos y redes, entre otros
- Domina los conocimientos para el diseño, aplicación y optimización de algoritmos que resuelven problemas computacionales específicos.
- Identifica los principios teóricos y prácticos del desarrollo de software, la infraestructura, la ingeniería de datos, la ciberseguridad, la inteligencia artificial y el desarrollo de aplicaciones específicas.

- Comprende los principios para la selección y utilización de lenguajes de programación para resolver problemas específicos.
- Domina los principios teóricos de las ciencias de la computación y del desarrollo de software para crear soluciones tecnológicas, considerando aplicaciones, sistemas operativos, bases de datos y redes, entre otros.
- Identifica los fundamentos para el desarrollo de mecanismos en la resolución de problemas relacionados con la seguridad, el almacenamiento, el procesamiento y el envío de información a través de redes de computadoras.
- Domina los principios básicos del aprendizaje automático y los modelos generativos, incluyendo redes neuronales, representación de datos, entrenamiento, sesgo, generalización y aplicaciones en texto, imagen, código y sonido.

Habilidades y destrezas (saber hacer)

- Demuestra el desarrollo de habilidades en la implementación del ciclo de vida para el desarrollo de software, aplicando metodologías modernas.
- Aplica distintos paradigmas computacionales para la resolución de problemas en las organizaciones.
- Administra infraestructuras computacionales, incluyendo redes, sistemas distribuidos, servicios en la nube y ciberseguridad.
- Implementa prácticas de seguridad en todo el ciclo de desarrollo de software.
- Aplica los principios computacionales de la inteligencia artificial, ingeniería de datos y ciberseguridad en la optimización del recurso de computacional que posee las organizaciones.
- Utiliza herramientas y sistemas basados en inteligencia artificial generativa para resolver problemas computacionales.
- Aplica los principios éticos en la práctica profesional, incluyendo privacidad, seguridad, propiedad intelectual y responsabilidad social.
- Demuestra el desarrollo de habilidades de liderazgo en los equipos de trabajo inter, multi y transdisciplinarios.
- Demuestra una comunicación asertiva oral y por escrito, en español e inglés en el ambiente laboral.
- Demuestra pensamiento crítico en la resolución de problemas de manera efectiva.
- Actúa con ética, responsabilidad y compromiso social en el desempeño de sus funciones.

Actitudes y valores (saber ser)

- Posee la capacidad de liderazgo para el trabajo en equipo, manteniendo una actitud vigilante para que se apliquen los principios que rigen a la UNA.
- Asume como parte integral de su quehacer, una ética centrada en el desarrollo humano, laboral y conservación de la naturaleza.
- Demuestra disposición para resolver problemas con los recursos disponibles a su alcance con calidad y excelencia en su práctica profesional.
- Aplica las normas de salud ocupacional y ambiente para el ejercicio laboral según los requerimientos de ejecución de su trabajo.
- Demuestra una actitud de aprendizaje permanente, actualizando sus conocimientos y habilidades conforme evoluciona la disciplina.

La División Académica de la OPES considera que dicho perfil está en consonancia con el objeto de estudio y los objetivos de la carrera presentados anteriormente; se enfoca en la formación de personas profesionales altamente capacitadas en el uso de tecnologías computacionales. La valoración técnica-profesional de la OPES sobre el perfil y otros elementos curriculares, como la estructura curricular y los contenidos del plan de estudios es que esos elementos son congruentes entre sí.

Esta Oficina ha constatado que se cumple con los requerimientos para el grado académico de Bachillerato en el Marco de Cualificaciones para la Educación Superior Centroamericana (MCESCA) ².

7. Campo de inserción laboral de los graduados

La UNA envió la siguiente información sobre el particular:

Los graduados de Diplomado pueden ocupar puestos de trabajo tales como:

- Desarrollador de Software general: Encargado de los requerimientos, diseño, programación, prueba y mantenimiento de software.
- Desarrollador de aplicaciones en plataformas Web y móvil: Diseña y desarrolla el frontend y el backend para aplicaciones en este tipo de plataformas
- Especialista de infraestructura computacional: Encargado de apoyar el desarrollo de infraestructura en la nube, virtualización y plataformas tecnológicas a escala empresarial.

Ámbitos de empleo para el diplomado

- Organizaciones especializadas en ingeniería de software, desarrollo de aplicaciones y soluciones de TI.
- Organizaciones públicas o privadas con áreas internas de TI, involucradas en el ciclo de vida del desarrollo de software (SDLC).
- Proveedores de servicios de infraestructura, nube, virtualización y plataformas tecnológicas a escala empresarial.

Los graduados de Bachillerato pueden ocupar puestos de trabajo como:

- Ingeniero en computación: Diseña, desarrolla, mejora sistemas y aplicaciones integrando software y hardware para resolver problemas en

² CSUCA, Marco de Cualificaciones para la Educación Superior Centroamericana, 2018.

el área disciplinar, considerando aspectos como ciberseguridad; inteligencia artificial y automatización de procesos; la ingeniería de datos y el desarrollo software.

- Desarrollador de software y aplicaciones en plataformas especializadas: Se encarga de los requerimientos, diseño, programación, prueba, despliegue y mantenimiento de software para diversas plataformas especializadas, tales como móviles y web.
- Especialista en Ciberseguridad: Trabaja en la protección de sistemas y datos contra accesos no autorizados, fraudes y otros ataques informáticos.
- Especialista en Infraestructura y redes: Se especializa en la administración de servidores, redes y la infraestructura tecnológica general de una organización.
- Ingeniero de datos: Diseña e implementa la infraestructura necesaria para recolectar, procesar y almacenar grandes volúmenes de datos, asegurando su calidad y fiabilidad.
- Especialista en inteligencia artificial: Aplica técnicas de inteligencia artificial y aprendizaje automático para el desarrollo de modelos que extraigan información valiosa de esos datos, creando sistemas capaces de tomar decisiones automáticas y resolver problemas complejos.
- Evaluador de software: verifica que un programa funcione correctamente, detecta errores y asegura que cumpla con los requisitos de calidad y usabilidad antes de su lanzamiento al público.
- Consultor de TI: analiza las necesidades de la organización, propone e implementa soluciones tecnológicas, mejora la seguridad digital, propone el uso de inteligencia artificial en los procesos, y optimiza el manejo de los datos.

Los graduados en Bachillerato en Ingeniería en Computación pueden desempeñarse en:

- Organizaciones especializadas en ingeniería en computación, desarrollo de aplicaciones y soluciones de TI.
- Organizaciones públicas o privadas con áreas internas de TI, involucradas en el ciclo de vida del desarrollo de software (SDLC), ciberseguridad, inteligencia artificial e ingeniería de datos.
- Proveedores de servicios de infraestructura computacional, nube, virtualización y plataformas tecnológicas a escala organizacional.
- Proveedores de servicios en ingeniería de datos, inteligencia de negocios y visualización de información.
- Entidades especializadas en ciberseguridad y gestión de riesgos digitales.
- Industrias intensivas en computación, que demandan automatización, inteligencia artificial y manejo avanzado de datos en entornos interdisciplinarios.

Esta Oficina considera que lo anterior es claro.

8. Carreras afines en la Educación Superior

En Computación había 51 carreras autorizadas en 2025. La lista completa se presenta en el Cuadro N°1. Es bastante amplia la oferta.

CUADRO N°1

CARRERAS DE GRADO EN COMPUTACIÓN AUTORIZADAS EN COSTA RICA, POR AÑO DE CREACIÓN, GRADOS Y UNIVERSIDAD. 2025

Carrera	Universidad	Grado	Año
Informática	Universidad de Costa Rica	B	1974
Computación	Universidad de Costa Rica	B	1974
Ingeniería en Computación	Instituto Tecnológico de Costa Rica	B	1976
Sistemas de Computación	Universidad Autónoma de Centro América	B	1978
Computación	Universidad Nacional	D	1978
Ingeniería en Computadoras	Universidad Autónoma de Centro América	L	1978
Sistemas de Computación	Universidad Federada de Costa Rica	B	1980
Ingeniería de Software	Universidad Veritas	B	1980
Informática Empresarial	Universidad de Costa Rica	B	1987
Ingeniería en Informática	Universidad Internacional de las Américas	BL	1987
Ingeniería en Sistemas	Universidad Magister	B	1989
Sistemas de Computación	Universidad Panamericana	B	1989
Ingeniería Informática	Universidad Latinoamericana de Ciencia y Tecnología	BL	1989
Ingeniería Informática	Universidad Central	BL	1990
Ingeniería en Sistemas de Información	Universidad Nacional	B	1993
Ingeniería Informática	Universidad Hispanoamericana	BL	1993
Ingeniería de Sistemas Informáticos	Universidad Latina de Costa Rica	BL	1993
Ingeniería Informática	Universidad Estatal a Distancia	BL	1994
Ingeniería en Sistemas de Computación	Universidad Fidélitas	BL	1994
Informática	Universidad Estatal a Distancia	D	1994
Ingeniería de Sistemas	Universidad Católica de Costa Rica Anselmo Llorente y Lafuente	BL	1995
Ingeniería de Sistemas	Universidad de Ciencias Empresariales	B	1997
Ingeniería de Sistemas Computacionales	Universidad Latina de Costa Rica	BL	1997
Ingeniería de Sistemas	Universidad Americana	BL	1998
Ingeniería Informática	Universidad del Valle	BL	1998
Ingeniería Informática	Universidad Metropolitana Castro Carazo	BL	1998
Ingeniería de Sistemas	Universidad Adventista de Centro América	BL	1999
Ingeniería en Sistemas Computacionales	Universidad Tecnológica Costarricense	BL	1999
Ingeniería de Sistemas	Universidad Autónoma de Centro América	B	2000
Ingeniería Telemática	Universidad Latina de Costa Rica	B	2004
Ingeniería del Software	Universidad Latina de Costa Rica	B	2004
Ingeniería en Sistemas	Universidad Internacional San Isidro Labrador	BL	2004
Ingeniería en Sistemas de Información	Universidad Internacional de las Américas	BL	2007
Ingeniería en Desarrollo de Software	Universidad Internacional de las Américas	BL	2007
Ingeniería en Computadores	Instituto Tecnológico de Costa Rica	L	2008
Administración de Tecnologías de Información	Instituto Tecnológico de Costa Rica	L	2008
Programación de Aplicaciones Informáticas	Universidad Nacional	D	2009
Sistemas de Información	Universidad Latina de Costa Rica	L	2009
Tecnologías Estratégicas de Información y Comunicación	Universidad Metropolitana Castro Carazo	L	2009
Ingeniería del Software	Universidad Cenfotec	B	2011
Ingeniería en Tecnologías de Información y Comunicación	Universidad Cenfotec	B	2011
Ingeniería en Tecnologías de Información	Universidad Técnica Nacional	BL	2011
Ingeniería del Software	Universidad Técnica Nacional	DBL	2011
Ingeniería en Tecnologías de Información	Universidad Técnica Nacional	DBL	2011
Desarrollo de Software	Universidad Latina de Costa Rica	L	2011
Administración de Infraestructuras Tecnológicas	Universidad Latina de Costa Rica	L	2011
Informática y Tecnología Multimedia	Universidad de Costa Rica	B	2013
Ingeniería en Gestión de la Calidad de Software	Universidad Fidélitas	L	2013
Tecnologías en Información y Comunicación Empresarial	Universidad INVENIO	L	2013
Informática	Universidad Nacional	L	2018

Nota: D: Diplomado B: Bachillerato L: Licenciatura.

Fuente: Base de datos División Académica, OPES-CONARE, 2022

9. Datos de empleabilidad del Observatorio Laboral de Profesiones

Según el Observatorio Laboral de Profesiones, en Computación el porcentaje de desempleo es de 1,9 %, el porcentaje de subempleo es de 0,0 % y el porcentaje de personas cuyos empleos no están relacionados con sus estudios es de 2,6 %. Estos indicadores evidencian una inserción laboral sólida y sostenida en el campo de la informática.

10. Requisitos de ingreso y de permanencia

Según la Universidad Nacional, los requisitos de ingreso son los siguientes:

- Poseer el grado de Bachillerato en Educación Media.
- Cumplir con lo señalado en los estatutos y reglamentos sobre el proceso de admisión a la Universidad Nacional.

En cuanto a los requisitos de permanencia, se establece que son los que establece al respecto la Universidad Nacional. Esta Oficina considera que los requisitos de ingreso a la carrera planteados, así como los de permanencia son apropiados y se ajustan a la normativa vigente.

11. Requisitos de graduación

Para graduarse se requiere aprobar todos los cursos de la estructura curricular y cumplir con otros requisitos administrativos y financieros establecidos por la Universidad Nacional.

Los requisitos de graduación planteados son apropiados.

12. Actividades de formación académica

La estructura curricular de la carrera, presentada en el Anexo A, consta de 141 créditos para el Bachillerato.

Se cumple con la normativa relativa a la duración, el número de créditos por ciclo lectivo y el total de créditos de la carrera respecto al grado de Bachillerato.

13. Descripción de las actividades de formación académica de la carrera

Los programas de las actividades de formación académica de la carrera se muestran en el Anexo B.

14. Correspondencia del equipo docente con las actividades académicas

En el Anexo C, se indican personal docente de cada uno de los cursos de la carrera. En el Anexo D se muestran sus grados académicos. Todos cumplen con el requisito de poseer al menos el grado de Licenciatura y sus diplomas o su experiencia laboral son afines con la asignatura que impartirá cada uno de ellos.

15. Ficha de información para la gestión de datos

DIVISIÓN ACADÉMICA

FICHA DE INFORMACIÓN PARA GESTIÓN DE DATOS

Nombre de la carrera:	Ingeniería en Computación
Universidad	Universidad Nacional
Grados académicos	Diplomado y Bachillerato
Nombre de la titulación:	Diplomado en Desarrollo de Software Bachillerato en Ingeniería en Computación

Clasificación carreras STEM

Carrera STEM

☒ Sí

☐ No

Número de créditos totales

141

Número de
periodos
totales

8

Tipo de ciclo o periodo
Semestral

Clasificación Campos de Educación y Formación (CINE-F 2013), UNESCO:

Campo amplio (área)

06 Tecnologías de la información y la comunicación

Campo específico (disciplina)

061 Tecnologías de la información y la comunicación

Campo detallado (carrera)

0613 Desarrollo de software y aplicaciones

Observaciones Generales

Es un rediseño.

16. Conclusiones

La propuesta curricular planteada cumple con los requisitos formales, con la normativa aprobada por el CONARE en el *Convenio para unificar la definición de crédito en la Educación Superior* ³ y en el *Convenio para crear una nomenclatura de grados y títulos de la Educación Superior Estatal* ⁴, con los requerimientos para el grado académico de Bachillerato en el Marco de Cualificaciones para la Educación Superior Centroamericana (MCESCA) ⁵, así como con los procedimientos establecidos por el documento *Lineamientos para la creación y el rediseño de carreras universitarias estatales* ⁶.

17. Recomendaciones

Con base en las conclusiones del presente estudio, se recomienda lo siguiente:

- Que la Universidad Nacional proceda con el rediseño del Bachillerato *en Ingeniería en Computación con salida lateral de Diplomado en Desarrollo de Software*, de acuerdo con los términos expresados en este dictamen.
- Se recomienda que la Universidad Nacional implemente mecanismos periódicos de evaluación y mejora continua de esta carrera, con la participación activa de egresados, empleadores y docentes, así como de la población estudiantil, especialmente en relación con las necesidades regionales de los distintos territorios donde se imparte la carrera, que aseguren su buena marcha, desarrollo, actualización y pertinencia.

³ Aprobada por el CONARE en la sesión del 10 de noviembre de 1976.

⁴ Aprobado por el CONARE y ratificado por los Consejos Universitarios e Institucional. Publicado en La Gaceta (Diario Oficial) 190 de 16 de octubre de 2023, páginas 42 a 46.

⁵ CSUCA, Marco de Cualificaciones para la Educación Superior Centroamericana, 2018.

⁶ Aprobado por el Consejo Nacional de Rectores en la sesión N°41-2022 celebrada el 18 de octubre de 2022.

ANEXO A

**ESTRUCTURA CURRICULAR DEL BACHILLERATO EN INGENIERÍA EN
COMPUTACIÓN DE LA UNIVERSIDAD NACIONAL
(con salida lateral de Diplomado en Desarrollo de Software)**

ANEXO A

ESTRUCTURA CURRICULAR DEL BACHILLERATO EN INGENIERÍA EN COMPUTACIÓN DE LA UNIVERSIDAD NACIONAL (con salida lateral de Diplomado en Desarrollo de Software)

CICLO Y CURSO	CRÉDITOS
<u>Primer ciclo</u>	<u>17</u>
Infraestructura I	3
Desarrollo de Software I	4
Computación y Modelado Matemático I	4
Estudios Generales I	3
Estudios Generales II	3
<u>Segundo ciclo</u>	<u>17</u>
Infraestructura II	3
Desarrollo de Software II	4
Computación y Modelado Matemático II	4
Estudios Generales III	3
Estudios Generales IV	3
<u>Tercer ciclo</u>	<u>18</u>
Infraestructura III	3
Desarrollo de Software III	4
Computación y Modelado Matemático III	4
Gestión Datos I	3
Inglés I	4
<u>Cuarto ciclo</u>	<u>18</u>
Infraestructura IV	3
Desarrollo de Software IV	4
Computación y Modelado Matemático IV	4
Gestión de Datos II	3
Inglés II	4
<u>Quinto ciclo</u>	<u>18</u>
Algoritmia Avanzada	4
Ingeniería de Software	3
Computación y Modelado Matemático V	4
Comunicación Oral y Escrita para Computación	3
Fundamentos de Lenguajes de Programación	4

CICLO Y CURSO	CRÉDITOS
<i>Total del Diplomado</i>	<i>88</i>
<u>Sexto ciclo</u>	<u>18</u>
Inteligencia Artificial I	4
Ciberseguridad I	4
Desarrollo para Plataformas	3
Ingeniería de Datos I	4
Interfaz Humano-Computadora	3
<u>Sétimo ciclo</u>	<u>18</u>
Inteligencia Artificial II	4
Ciberseguridad II	4
Optativa I (libre)	3
Ingeniería de Datos II	4
Seminario I	3
<u>Octavo ciclo</u>	<u>17</u>
Práctica Profesional Supervisada	5
Optativa II (disciplinar)	3
Optativa III (disciplinar)	3
Optativa IV (disciplinar)	3
Seminario II	3
<i>Total del Bachillerato</i>	<i>141</i>

Lista de optativos (todos de tres créditos):

Cursos Optativos Disciplinarios

Informática Forense
 Métodos Numéricos Aplicados a la Ingeniería
 Modelado Multidimensional e Inteligencia de Negocios
 Computación en la Nube Avanzado
 Introducción al Análisis de Imágenes Digitales

Optativos Libres

Introducción a la Creación de Empresas
 Las TIC en el ámbito jurídico de Costa Rica
 Propiedad Intelectual y su aplicación práctica
 Aseguramiento de la Calidad de Software

ANEXO B

TEMÁTICAS DE LOS CURSOS DEL BACHILLERATO EN INGENIERÍA EN COMPUTACIÓN DE LA UNIVERSIDAD NACIONAL

ANEXO B

TEMÁTICAS DE LOS CURSOS DEL BACHILLERATO EN INGENIERÍA EN COMPUTACIÓN DE LA UNIVERSIDAD NACIONAL

Infraestructura I

I. Descripción del Curso

Este curso introductorio presenta la organización básica de un sistema de cómputo (hardware y software), el rol de los sistemas operativos, los principios elementales de redes, prácticas elementales de seguridad y bienestar digital, y los fundamentos de virtualización y computación en la nube. Se promueve el uso práctico de la línea de comandos (CLI) para realizar tareas sencillas, diagnosticar problemas comunes y adoptar hábitos de trabajo ordenados y seguros. Reconocerá componentes y arquitecturas comunes de PC, servidores y dispositivos conectados.

II. Objetivos

Objetivo general:

Reconocer los conceptos y procesos fundamentales para la instalación, operación y mantenimiento de sistemas computacionales, mediante la aplicación de metodologías específicas, con el fin de desarrollar una comprensión profunda que permita resolver problemas tecnológicos de manera efectiva.

Objetivos específicos:

- Identificar los componentes principales de hardware y su función dentro de la arquitectura de un sistema de cómputo, a través del análisis de cada componente (como la CPU, la memoria y los dispositivos de entrada/salida), para comprender cómo contribuyen al procesamiento de datos, el almacenamiento y la comunicación.
- Explicar el rol del sistema operativo en la gestión de procesos, memoria, archivos y permisos, mediante el análisis de cómo coordina y asigna recursos a estos elementos, para garantizar la eficiencia en la ejecución de programas.
- Ejecutar tareas desde la línea de comandos para navegar por el sistema de archivos, gestionar procesos, realizar redirecciones y consultas básicas, mediante el uso de un entorno de línea de comandos adecuados (Shell) apropiado, para interactuar de forma eficiente con el sistema operativo.
- Interpretar la representación de datos (bits/bytes, binario/hexadecimal, complemento a dos y codificación de caracteres) en ejercicios prácticos, aplicando métodos teóricos y prácticos adecuados para analizar e identificar patrones en la información de manera clara y efectiva.
- Describir los parámetros esenciales de red (IPv4, puertos y DNS, etc.), así como las herramientas de diagnóstico y su utilización en entornos locales y virtualizados, para la comprensión de su función en la gestión de redes.

II Contenidos:

1. Fundamentos de electricidad: conceptos básicos de carga, corriente y voltaje; leyes de Ohm y Kirchhoff; potencia eléctrica; componentes electrónicos y circuitos lógicos básicos en sistemas computacionales.
2. Componentes y arquitectura de computadoras personales: unidades funcionales principales; jerarquía y niveles de memoria; mecanismos de entrada/salida.
3. Representación de datos en sistemas digitales: bits y bytes; sistemas binario y

- hexadecimal; codificación de caracteres (ASCII/UTF-8).
4. Sistemas operativos: fundamentos y funciones principales: procesos e hilos (noción); gestión de memoria (visión general); sistemas de archivos y permisos; relación hardware–sistema operativo–aplicaciones; GUI vs. CLI.
 5. Comandos básicos en la interfaz de línea de comandos (CLI): rutas; creación y gestión de archivos, y directorios; variables de entorno; visualización y terminación de procesos.
 6. Conceptos básicos de redes de computadoras: direccionamiento IPv4, máscara, puerta de enlace, puertos, DNS; herramientas básicas de diagnóstico (ping, nslookup, etc.).
 7. Seguridad informática básica: principios de confidencialidad, integridad y disponibilidad; mínimo privilegio; contraseñas y MFA; actualizaciones, copias de seguridad y ética digital.
 8. Fundamentos de virtualización: concepto y casos de uso; máquinas virtuales vs. contenedores (visión general); escenarios de laboratorio.
 9. Conceptos introductorios de computación en la nube: modelos de servicio (IaaS/PaaS/SaaS a alto nivel); recursos virtuales; consideraciones básicas de costo y seguridad.

Desarrollo de Software I

I. Descripción del Curso

Este primer curso en desarrollo de software y programación introduce al estudiante en los fundamentos del desarrollo de software, mediante el aprendizaje de los aspectos básicos de la programación de computadores y la aplicación, a nivel elemental, de todas las fases del ciclo de vida del desarrollo de software (CVDS-nivel 1). Se realizan proyectos de alcance moderado donde se integran no solo los aspectos de programación, sino también conceptos básicos de requerimientos, diseño, pruebas, versionado, despliegue y administración del proceso.

II. Objetivos

Objetivo general:

Desarrollar proyectos de software de alcance moderado, con un enfoque profesional integrando tanto aspectos fundamentales de programación, como del ciclo de vida de desarrollo de software.

Objetivos específicos:

- Comprender las etapas del ciclo de vida de desarrollo de software.
- Aplicar el pensamiento lógico y el concepto de algoritmo en la solución de problemas.
- Resolver problemas de programación usando estructuras de datos y de flujo de control apropiadas.
- Crear soluciones de software de complejidad moderada aplicando, a nivel elemental, todas las etapas del ciclo de vida de desarrollo de software y utilizando en cada caso las herramientas apropiadas.
- Demostrar capacidad de trabajo en equipo y de comportamiento ético en el cumplimiento de sus responsabilidades

III. Contenidos

Ciclo de vida de desarrollo de software, pensamiento algorítmico, flujo de control, estructuras de datos simples (arreglos, listas, diccionarios), entrada/salida (consola, archivos, API), algoritmos de búsqueda y ordenamiento y uso de recursividad, todo dentro de un enfoque estructurado. Diseño de programas usando abstracción y encapsulamiento, pruebas unitarias, versionado y despliegue de software.

Computación y Modelado Matemático I

I. Descripción del Curso

Este curso introduce los conceptos fundamentales de matemática necesarios para el estudio de las ciencias de la computación, con énfasis en el desarrollo del pensamiento computacional. Se enfoca en desarrollar la capacidad de formular problemas, Utilizar herramientas computacionales para resolverlos y comprender los fundamentos lógicos y algebraicos que sustentan la computación moderna. El curso adopta un enfoque práctico donde los estudiantes aprenden a usar la computación como herramienta para explorar y resolver problemas matemáticos.

II. Objetivos

Objetivo general:

Desarrollar el pensamiento computacional y las competencias matemáticas fundamentales necesarias para las ciencias de la computación.

Objetivos específicos:

- Desarrollar habilidades de pensamiento computacional para descomponer y resolver problemas complejos.
- Utilizar herramientas computacionales para explorar y validar conceptos matemáticos.
- Aplicar principios de lógica proposicional y de predicados en la formulación y verificación de algoritmos.
- Modelar situaciones reales usando teoría de conjuntos, relaciones y funciones.
- Desarrollar la teoría de vectores y matrices vinculando estos conceptos en contextos computacionales.
- Aplicar la teoría de sucesiones y relaciones de recurrencia mediante el empleo de herramientas de software.
- Utilizar inducción matemática en la demostración de igualdades que involucran sumatorias y/o productorias.

III. Contenidos:

1. Introducción al pensamiento computacional: descomposición, reconocimiento de patrones, abstracción y algoritmos. Formulación de problemas. Uso de herramientas computacionales para exploración matemática. Modelado matemático de situaciones reales.
2. Sucesiones aritméticas y geométricas. Sumatorias y productorias. Relaciones de recurrencia lineales y particularmente homogéneas. Método característico. Exploración de patrones usando herramientas computacionales.
3. Principio de inducción matemática (enfaticando identidades con sumatorias y/o productorias).
4. Proposiciones y valores de verdad. Conectivas lógicas: conjunción, disyunción, implicación, bicondicional y negación. Construcción de tablas de verdad usando herramientas computacionales. Leyes lógicas: conmutatividad, asociatividad, distributividad, De Morgan. Tautologías y contradicciones. Inferencia lógica y reglas de deducción. Cuantificadores universales y existenciales. Negación de cuantificadores. Cuantificadores anidados. Uso en la creación de instrucciones para programación (if, for each, switch, entre otras).
5. Definición y notación de conjuntos. Operaciones con conjuntos: unión, intersección, diferencia, complemento. Leyes de conjuntos. Cardinalidad. Producto cartesiano. Principio de inclusión-exclusión. Conjuntos potencia. Visualización y manipulación de conjuntos usando software (uso de select, entre otras). Aplicaciones en estructuras de datos y bases de datos relacionales (casos simples).

6.Relaciones binarias. Propiedades de relaciones: reflexividad, simetría, transitividad. Relaciones de equivalencia y orden. Funciones en una y varias variables: dominio, codominio, rango. Tipos de funciones: inyectivas, sobreyectivas, biyectivas. Composición de funciones. Función inversa. Análisis gráfico de funciones usando herramientas computacionales (intersecciones con los ejes, identificación de extremos locales y globales, asíntotas, entre otras). Modelado del tiempo de ejecución de un algoritmo. Monotonía. Funciones pares e impares. Gráficas de superficies usando software.

7.Vectores en el plano y el espacio. Operaciones con vectores: suma, resta, producto escalar. Matrices: definición, tipos y notación. Operaciones con matrices: suma, resta, multiplicación, transpuesta. Determinantes y propiedades básicas. Matriz inversa. Sistemas de ecuaciones lineales y métodos de solución. Aplicaciones en transformaciones geométricas, gráficos computacionales y representación de datos.

Infraestructura II

I. Descripción del Curso

Este curso fundamental del área de comunicaciones y redes introduce al estudiante en los principios, modelos y tecnologías de las redes de computadoras. Se estudian los fundamentos de la comunicación digital, las arquitecturas de red, los protocolos y los servicios de Internet, así como conceptos de seguridad y sostenibilidad en redes. El curso integra teoría y práctica mediante simulaciones, de manera que los estudiantes adquieran la capacidad de diseñar, analizar y administrar redes modernas, considerando aspectos de rendimiento, confiabilidad y seguridad.

II. Objetivos

Objetivo general:

Analizar los principios, modelos y tecnologías de redes de computadoras, mediante el estudio de protocolos, dispositivos, servicios, y la implementación de prácticas de diseño, simulación y configuración, para el desarrollo de soluciones de red seguras, eficientes y escalables en contextos reales de infraestructura computacional.

Objetivos específicos:

- Analizar los fundamentos de la comunicación digital y los modelos de referencia OSI/TCP-IP, la función de cada capa y los protocolos que la integran, con el fin de comprender la organización lógica de las redes y su funcionamiento como base para el diseño e implementación de infraestructuras de comunicación.
- Implementar redes de computadoras, mediante el uso de simuladores y herramientas prácticas, aplicando protocolos de enrutamiento, direccionamiento IPv4/IPv6 y técnicas de seguridad básica, para la garantía de la conectividad, funcionalidad y protección de la infraestructura de red en diferentes escenarios.
- Diseñar soluciones de red seguras y escalables, integrando técnicas de seguridad, criterios de rendimiento y adaptación tecnológica, con el fin de la protección de la infraestructura de comunicación y la garantía de su confiabilidad en diversos escenarios operativos.

III. Contenidos

1. Fundamentos de Comunicación de Datos

Introduce los conceptos esenciales de transmisión de datos, modelos de comunicación, medios y principios de codificación y modulación.

2. Diseño e implementación de redes de área

Cubre los principios y prácticas para planificar, desplegar y configurar redes locales y de área amplia que soporten servicios y aplicaciones de red.

3. Soporte de Fiabilidad
Explora mecanismos y protocolos que permiten mantener la continuidad operativa de la red, incluyendo redundancia, tolerancia a fallos y recuperación ante errores.
4. Enrutamiento y Reenvío
Aborda cómo se seleccionan y aplican rutas para el envío eficiente de paquetes, incluyendo algoritmos, tablas de enrutamiento y protocolos intra/interdominio.
5. Comunicación de Salto Único
Analiza la comunicación directa entre nodos, enfocándose en medios físicos, enlaces punto a punto y control de acceso.
6. Seguridad de Red
Examina amenazas, vulnerabilidades y mecanismos de protección en redes, incluyendo autenticación, cifrado, integridad, control de acceso y defensa ante ataques.
7. Movilidad
Analiza cómo las redes dan soporte a nodos móviles y a la movilidad de usuarios, manteniendo conectividad continua, handovers eficientes y calidad de servicio.
8. Temas Emergentes
Presenta tecnologías, arquitecturas y paradigmas de redes emergentes (SDN, NFV, virtualización avanzada, IA en redes, 5G/6G, etc.), enfatizando tendencias actuales y futuras.

Desarrollo de Software II

I. Descripción del Curso

Este curso inicial sobre estructuras de datos y análisis de algoritmos aplica el paradigma de la programación orientada a objetos (POO). Se aprenden los conceptos de la programación orientada a objetos (clases, objetos, herencia, polimorfismo) y se aplican en la construcción de estructura de datos clásicas (colas, pilas, listas, árboles y grafos), estudiando también la complejidad algorítmica de las operaciones sobre dichas estructuras. En los proyectos se aplican, a un nivel intermedio de rigurosidad, todas las fases del ciclo de vida del desarrollo de software (CVDS-nivel 2), siguiendo una versión simplificada de una metodología ágil como Scrum.

II. Objetivos

Objetivo general:

Desarrollar, analizar y aplicar estructuras de datos clásicas en proyectos de software de alcance medio, usando apropiadamente la programación orientada a objetos siguiendo una metodología que cubra todo el ciclo de vida de desarrollo de software.

Objetivos específicos:

- Aplicar estructuras de datos clásicas, tales como colas, pilas, listas, árboles y grafos; en el desarrollo de proyectos de software.
- Analizar el rendimiento, en cuanto a tiempo y espacio, de estructuras de datos clásicas.
- Aplicar el paradigma de programación orientada a objetos, incorporando los conceptos de: clase, objeto, herencia, polimorfismo, composición y asociación.
- Ejecutar proyectos de desarrollo de software siguiendo una metodología ágil como Scrum.
- Demostrar capacidad de trabajo en equipo y de comportamiento ético en el cumplimiento de sus responsabilidades.

III Contenidos:

1. Introducción al paradigma de la programación orientada a objetos (POO):

- clases, objetos, atributos y métodos.
- 2. Estructuras de datos lineales: pilas, colas y listas. Introducción al análisis de algoritmos.
- 3. Aspectos más elaborados sobre POO: herencia, polimorfismo, composición y asociación.
- 4. Estructuras de datos no lineales: árboles y grafos.
- 5. Metodología ágil Scrum.
- 6. Versionamiento y despliegue en contenedores.

Computación y Modelado Matemático II

I. Descripción del Curso

Este curso profundiza en las estructuras matemáticas discretas fundamentales para las ciencias de la computación, con énfasis en el análisis y diseño de algoritmos. Se centra en desarrollar competencias avanzadas en análisis de algoritmos, teoría de grafos, árboles y autómatas, aplicando el pensamiento computacional para resolver problemas complejos en inteligencia artificial, desarrollo de software, redes y ciberseguridad, e ingeniería de datos.

II. Objetivos

Objetivo general:

Diseñar y optimizar soluciones computacionales eficientes para problemas complejos en áreas fundamentales y de especialización de la disciplina, utilizando herramientas computacionales modernas y metodologías de pensamiento computacional.

Objetivos específicos:

- Validar y visualizar soluciones usando herramientas computacionales apropiadas para explorar patrones y verificar resultados.
- Analizar la eficiencia de algoritmos usando notaciones asintóticas y herramientas computacionales.
- Aplicar teoría de grafos para el modelaje y resolución de problemas de redes y su optimización.
- Utilizar estructuras de árboles para el diseño de algoritmos eficientes de búsqueda y organización.
- Emplear autómatas finitos para el modelaje de sistemas de estados y procesamiento de cadenas.
- Resolver problemas de enumeración aplicando principios fundamentales de conteo, permutaciones, combinaciones y el principio del palomar.
- Resolver problemas de probabilidad computacional combinando técnicas de conteo para la determinación de espacios muestrales y el cálculo de probabilidades.

III. Contenidos:

1. Criterios de análisis: corrección, eficiencia, legibilidad. Notaciones asintóticas: O , Ω y Θ . Análisis gráfico de las notaciones asintóticas usando herramientas computacionales. Propiedades de las notaciones asintóticas. Ejemplos de análisis de algoritmos utilizando notaciones asintóticas.
2. Concepto de grafo, dígrafos y sus propiedades. Tipos de grafos: simples, múltiples, completos, bipartitos, planares. Caminos y circuitos: aplicaciones en informática y redes. Circuitos de Euler, teoremas fundamentales y aplicaciones en *routing*. Circuitos de Hamilton, teoremas relacionados (Dirac y Ore) y aplicaciones sobre el problema del agente viajero. Matriz de adyacencia e incidencia y sus propiedades computacionales. Concepto de grafo ponderado. Algoritmo del camino más corto (Dijkstra) y sus implementaciones.
3. Concepto de árbol y sus propiedades fundamentales. Árboles binarios y sus propiedades estructurales. Árboles generadores o de expansión y algoritmos:

búsqueda primero a lo ancho (BFS) y a lo profundo (DFS). Operaciones de los árboles binarios: búsqueda, inserción y recorridos preorden, inorden y postorden. Árboles con etiquetas y sus aplicaciones sobre la representación de expresiones algebraicas. Árboles de expansión mínima y algoritmos para su generación (Kruskal o Prim). Problemas y aplicaciones clásicas de árboles en bases de datos y sistemas de archivos.

4. Concepto y propiedades de un autómata de estado finito determinista y no determinista. Recorridos de cadenas en un autómata de estado finito. Construcción de autómatas de estado finito para reconocimiento de patrones. Expresiones regulares.
5. Principio fundamental de conteo. Permutaciones y combinaciones. Principio del palomar. Combinaciones con repetición. Exploración de problemas combinatorios usando herramientas computacionales (para conteo, cálculo numérico, aproximaciones, entre otras).
6. Espacios de probabilidad discretos. Distribuciones: binomial, geométrica, Poisson. Valor esperado y varianza. Simulación de eventos aleatorios usando software. Exploración de distribuciones de probabilidad mediante herramientas computacionales. Aplicaciones básicas en análisis de algoritmos randomizados.

Infraestructura III

I. Descripción del Curso

Este curso fundamental del área de sistemas introduce al estudiante en los principios y prácticas de la arquitectura y organización de computadoras. Su propósito es que el estudiantado comprenda los fundamentos de los sistemas computacionales modernos, desde la representación de datos y la organización de instrucciones, hasta la jerarquía de memoria, la eficiencia energética, la seguridad y las arquitecturas emergentes. El curso combina teoría con práctica mediante proyectos aplicados, de manera que los estudiantes analicen, evalúen y desarrollen soluciones que integren aspectos de hardware y software en el funcionamiento interno de la computadora.

II. Objetivos

Objetivo general:

Analizar los principios de diseño y funcionamiento de las funciones digitales en sistemas computacionales, mediante el estudio de la arquitectura de hardware y la implementación de prácticas de diseño físicas y simuladas, así como el uso de lenguajes de bajo nivel, para la profundización en la interacción entre el software y el hardware.

Objetivos específicos:

- Analizar los principios fundamentales de las funciones digitales, el diseño lógico y la composición de sistemas digitales, para la explicación de la relación entre estructuras combinatoriales y secuenciales y su integración en arquitecturas computacionales básicas, mediante prácticas.
- Examinar la estructura interna de las arquitecturas de CPU, considerando la gestión de memoria, los formatos de instrucción y el procesamiento en *pipeline*, mediante emuladores especializados, para vincular teoría y práctica en el desarrollo de funciones en lenguaje de bajo nivel.
- Aplicar instrucciones en lenguaje ensamblador x86, mediante simuladores y emuladores de bajo nivel, para manipular registros, ALU, memoria y operaciones internas de la CPU.
-

III. Contenidos:

1. Fundamentos de Semiconductores
Introduce los principios físicos y electrónicos que sustentan la implementación de compuertas lógicas y circuitos digitales. Este tema explora los materiales base de la computación moderna y se centra en cómo los semiconductores, como el silicio, se utilizan para crear transistores.
2. Representación de Datos
Examina cómo los datos se representan internamente, incluyendo números binarios, enteros con signo, punto flotante, caracteres y estructuras, así como los errores y limitaciones asociados a las distintas representaciones.
3. Lógica Digital y Sistemas Digitales
Cubre el diseño y análisis de circuitos combinacionales y secuenciales, así como su integración en sistemas digitales básicos.
4. Arquitectura del Procesador
Examina la estructura interna de la CPU, los conjuntos de instrucciones, la ejecución en pipeline, el control y la organización de componentes.
5. Jerarquía de Memoria
Estudia la organización jerárquica de memorias (registros, caché, memoria principal y secundaria) y sus implicaciones en el desempeño. La jerarquía va desde la memoria más rápida y pequeña (registros del procesador y caché) hasta la más lenta y grande (RAM y almacenamiento en disco).
6. Interfaz y Comunicación
Analiza cómo los procesadores interactúan con periféricos y otros sistemas mediante buses, controladores, protocolos de E/S e interrupciones.
7. Rendimiento y Energía
Aborda métodos de evaluación y optimización del rendimiento de sistemas, junto con el análisis del consumo energético y la eficiencia.
8. Arquitecturas Modernas
Explora arquitecturas heterogéneas (multicore, GPU, aceleradores) y mecanismos de seguridad a nivel arquitectónico frente a amenazas modernas.
9. Tendencias Futuras (Introducción a Arquitecturas Cuánticas)
Presenta los conceptos fundamentales de cómputo cuántico y su potencial impacto en las arquitecturas futuras, así como otras tecnologías y tendencias emergentes.

Desarrollo de Software III

I. Descripción del Curso

Este curso avanzado en desarrollo de software usa programación concurrente y distribuida que establece, coordina y controla múltiples cómputos que ocurren al mismo tiempo y en diferentes lugares. Se introducen los conceptos y retos de la programación concurrente y distribuida que se aplican al desarrollo de proyectos de software utilizando recursos computacionales de manera eficiente y escalable. Se aplica una metodología ágil que cubra, a un nivel intermedio de rigurosidad, todas las etapas del ciclo de vida del desarrollo de software (CVDS-nivel 2).

II Objetivos

Objetivo general:

Desarrollar sistemas de software de alto rendimiento que aprovechen recursos de hardware con capacidad de paralelismo, usando técnicas y tecnologías de programación concurrente y distribuida, con la finalidad de cubrir todas las etapas del ciclo de vida de desarrollo de software.

Objetivos específicos:

- Comprender los conceptos de paralelismo, concurrencia y distribución.
- Utilizar procesos e hilos para implementar programación concurrente.
- Aplicar mecanismos de sincronización y comunicación adecuados en procesamiento concurrente.
- Resolver problemas de procesamiento distribuido usando los algoritmos y mecanismo de comunicación adecuados.
- Demostrar capacidad de trabajo en equipo y de comportamiento ético en el cumplimiento de sus responsabilidades.

III Contenidos:

1. Conceptos de concurrencia, paralelismo y distribución. Arquitecturas paralelas.
2. Procesos e hilos, condiciones de carrera, exclusión mutua, interbloqueo e inanición.
3. Sincronización y comunicación, memoria compartida y paso de mensajes. Diseño y análisis de algoritmos concurrentes.
4. Sistemas distribuidos: modelos de comunicación, algoritmos distribuidos, modelos de difusión y de consenso.
5. Plataformas (middleware) modernas de procesamiento paralelo y distribuido.

Computación y Modelado Matemático III

I. Descripción del Curso

Este curso de Computación y Modelado Matemático III, se refiere al estudio del cálculo aplicado a la computación. Introduce al estudiante en los principios fundamentales del cálculo diferencial e integral en una y varias variables, enfocados en la resolución de problemas computacionales. Se emplea software especializado para explorar funciones, límites, derivación e integración, tanto simbólica como numéricamente, y para modelar fenómenos reales. El enfoque es práctico y promueve el pensamiento computacional como herramienta para el análisis matemático aplicado.

El curso capacita al estudiante para utilizar herramientas digitales, formular modelos matemáticos, implementar soluciones algorítmicas y justificar decisiones técnicas en contextos reales de su área de especialización, integrando técnicas de cálculo con enfoques computacionales aplicados.

II. Objetivos

Objetivo general:

Desarrollar competencias para la comprensión, aplicación, análisis y evaluación de conceptos de cálculo diferencial e integral en una y varias variables, orientados a la resolución de problemas computacionales.

Objetivos específicos:

- Relacionar entornos computacionales para la realización de cálculo simbólico, gráfico y numérico.
- Analizar el concepto de límite y su relación con la continuidad y la derivación.
- Modelar procesos de acumulación, cálculo de áreas y fenómenos físicos computacionales a través del uso de integrales.
- Resolver problemas matemáticos mediante métodos simbólicos y numéricos, seleccionando herramientas digitales adecuadas.
- Analizar situaciones reales mediante funciones continuas y discontinuas, formulando modelos matemáticos pertinentes.
- Interpretar gráficamente el comportamiento de funciones, límites, derivadas e integrales, evaluando su significado en contextos aplicados.

III. Contenidos:

1. Exploración Computacional Inicial: Instalación, configuración y uso básico de software matemático.
2. Límites y Continuidad: Introducción intuitiva mediante tablas de valores. Cálculo manual y computacional de límites. Propiedades, indeterminaciones y continuidad. Aplicaciones computacionales.
3. Derivadas: Reglas de derivación y su interpretación geométrica. Tasas de cambio y derivación asistida por software. Regla de la cadena, derivadas parciales y gradiente.
4. Aplicaciones de la Derivada: Optimización. Crecimiento y decaimiento. Modelado de trayectorias.
5. Integrales Definidas e Indefinidas: Antiderivación y técnicas básicas (sustitución y partes). Sumas de Riemann. Propiedades e interpretación de la integral definida. Teorema fundamental del cálculo.
6. Aplicaciones de la Integral: Cálculo de áreas y volúmenes. Trabajo físico. Simulación de procesos físicos.
7. Métodos Numéricos de Integración: Interpolación y recta de mejor ajuste. Algoritmo de Newton-Raphson. Derivación numérica. Reglas del trapecio y Simpson. Algoritmo del descenso del gradiente.
8. Temas adicionales usando software: Gráficas animadas (sliders). Gráficas de desigualdades. Coordenadas polares. Curvas paramétricas en 2D y 3D. Gráficas de superficies.
9. Series de Taylor: Definición de serie de Taylor. Serie de Maclaurin como caso especial. Derivadas sucesivas y coeficientes. Aproximación de funciones con polinomios

Gestión de Datos I

I. Descripción del Curso

En este curso los estudiantes aprenderán los fundamentos de las bases de datos relacionales, abarcando desde el modelado conceptual de datos, la estructura y diseño de tablas hasta el uso del lenguaje SQL para realizar consultas y manipulación de datos. Se abordarán también conceptos esenciales como la integridad de datos, procedimientos almacenados y optimización de consultas.

II Objetivos

Objetivo general:

Diseñar e implementar bases de datos relacionales, utilizando herramientas y lenguajes adecuados que permitan gestionar eficientemente la información en distintos contextos informáticos.

Objetivos específicos:

Al finalizar el curso, el estudiante será capaz de:

- Comprender los principios fundamentales de las bases de datos relacionales.
- Diseñar y crear estructuras de bases de datos eficientes y normalizadas.
- Utilizar el lenguaje SQL para definir, consultar y manipular datos.
- Implementar funciones, procedimientos y disparadores (*triggers*).
- Establecer conexiones con bases de datos desde aplicaciones externas.
- Aplicar técnicas básicas de optimización de consultas.
-

- II. Contenidos:
 - 1. Introducción a las Bases de Datos Relacionales
 - 2. Modelado conceptual de datos
 - 3. Diseño de Bases de Datos Relacionales
 - 4. Creación de Tablas y Relaciones
 - 5. Lenguaje SQL (Lenguaje No Procedimental)
 - 6. Lenguaje Procedimental
 - 7. Conexiones a Bases de Datos
 - 8. Optimización de Consultas

Infraestructura IV

I. Descripción del Curso

Este curso profundiza en los fundamentos de los sistemas operativos (gestión de memoria, procesos y almacenamiento persistente) y los articula con prácticas actuales de desarrollo y operaciones. Se estudian las abstracciones del sistema operativo y su extensión mediante virtualización y contenedores, con experiencias prácticas en tecnologías de contenedores y orquestación. El estudiantado aplica integración y entrega continua para automatizar compilación, pruebas y despliegue, incorpora infraestructura como código y establece controles básicos de seguridad (gestión de secretos y endurecimiento). Además, introduce monitoreo y observabilidad para medir disponibilidad, desempeño y errores en producción. Al finalizar el curso, la persona es capaz de diseñar, desplegar y operar servicios confiables y escalables, justificando decisiones según criterios técnicos.

II. Objetivos

Objetivo general:

Diseñar plataformas operativas para el despliegue y la operación de servicios de software mediante virtualización y contenedores, integración y entrega continua, con controles de seguridad orientadas a la confiabilidad y escalabilidad.

Objetivos específicos:

- 1. Evaluar los mecanismos de gestión de memoria virtual y procesos, almacenamiento persistente y concurrencia en sistemas operativos modernos, a través del análisis de casos prácticos y la comparación entre diferentes enfoques de administración de recursos, para la comprensión de su impacto en el rendimiento y la eficiencia de los sistemas.
- 2. Analizar las características de la virtualización y de los contenedores para el despliegue de aplicaciones, por medio de la identificación de sus aportes en aislamiento, uso de recursos, tiempo de aprovisionamiento, portabilidad, configuración de red y almacenamiento, explicando su integración en arquitecturas con orquestación e infraestructura como código.
- 3. Aplicar los principios de integración y entrega continuas para la automatización del ciclo de vida del software, a través de la construcción de flujos de trabajo para el aseguramiento de la calidad, seguridad y eficiencia de los despliegues.
- 4. Implementar contenedores para el despliegue de aplicaciones, mediante herramientas de orquestación que permitan su escalabilidad, manejo automatizado y alta disponibilidad, para la garantía de entornos de ejecución flexibles, confiables y reproducibles.
- 5. Evaluar el rendimiento y la seguridad de diferentes configuraciones de sistemas y despliegues de aplicaciones, mediante la aplicación de métricas de desempeño, prácticas de observabilidad y pruebas de endurecimiento en entornos controlados, para la optimización de la operación y fortalecimiento de la protección de los servicios.

III Contenidos:

1. Principios de Sistemas Operativos
Gestión de memoria (paginación, memoria virtual), sistemas de archivos (*journaling*, RAID) y modelo de procesos.
2. Virtualización y Contenedores
Diferencias entre máquinas virtuales y contenedores. Aislamiento de recursos. Uso de tecnologías de contenedores para la creación y gestión imágenes. Orquestación de contenedores.
3. Principios de DevOps
Cultura, automatización, medición y colaboración (CAMS). Infraestructura como Código (IaC).
4. Ciclo de Vida del Software Automatizado (CI/CD)
Herramientas de integración continua. Creación de flujos de trabajo para compilación, pruebas y despliegue automáticos.
5. Seguridad y Cloud Computing
Modelos de seguridad en la nube (responsabilidad compartida). Aseguramiento de contenedores. Gestión de secretos. Monitoreo y observabilidad en sistemas distribuidos.

Desarrollo de Software IV

I. Descripción del Curso

El curso se enfoca en el desarrollo de aplicaciones escalables para plataformas especializadas Web y Móvil. Introduce los fundamentos comunes a estos tipos de plataformas, como la separación entre cliente (*frontend*) y servidor (*backend*) y los protocolos de comunicación entre ellos, el desarrollo de APIs basados en microservicios con seguridad y balanceo de carga, y la persistencia de datos. Incursiona en el desarrollo de la parte cliente (*frontend*), tanto para Web como para aplicaciones móviles. En los proyectos se aplican, a un nivel pleno de rigurosidad, todas las fases del ciclo de vida del desarrollo de software (CVDS-nivel 3).

II Objetivos

Objetivo general:

Desarrollar aplicaciones para plataformas web y móviles que sean robustas, escalables y seguras, aplicando una metodología de desarrollo ágil.

Objetivos específicos:

- Implementar la parte servidor de una aplicación usando microservicios con balanceo de carga en el desarrollo de los APIs
- Aplicar protocolos de comunicación, tales como REST y SOAP, para acceder a los APIs del servidor.
- Implementar aplicaciones web usando plataformas cliente adecuadas, tales como React o Angular.
- Implementar aplicaciones móviles usando plataformas cliente apropiadas, tales como Android Studio o React Native.
- Demostrar capacidad de trabajo en equipo y de comportamiento ético en el cumplimiento de sus responsabilidades.

III Contenidos

Conceptos de cliente (*frontend*) y servidor (*backend*). Plataforma para servidor (*backend*). APIs implementados como microservicios. Servicios REST y SOAP. Desarrollo del cliente (*frontend*) como una aplicación de una sola página (*Single Page Application*). Plataforma para cliente web (*React* o *Angular*). Desarrollo

cliente tipo aplicación móvil. Plataforma para móviles (*Android Studio* o *React Native*). Acceso y uso a los recursos del móvil (cámara, gps, etc.). Tiendas de aplicaciones móviles.

Computación y Modelado Matemático IV

I. Descripción del Curso

Este curso proporciona los principios fundamentales del álgebra lineal, enfocados en su uso práctico en ciencias de la computación. Mediante el uso de software especializado y un enfoque práctico orientado al desarrollo del pensamiento computacional, el estudiante aprenderá no solo los conceptos teóricos (vectores, espacios vectoriales, autovalores, ortogonalidad, etc.), sino también cómo implementarlos algorítmicamente y aplicarlos en contextos reales en áreas como análisis de datos y gráficos por computadora, entre otras.

II. Objetivos

Objetivo general:

Desarrollar el pensamiento computacional utilizando los fundamentos del álgebra lineal y sus principios computacionales para la formulación de modelos matemáticos y el desarrollo de soluciones algorítmicas eficientes, orientadas a la resolución de problemas complejos en ciencias de la computación.

Objetivos específicos:

- Resolver sistemas de ecuaciones lineales mediante métodos algorítmicos (reducción por filas, factorización de matrices), y el análisis de su eficiencia considerando la complejidad algorítmica asociada.
- Caracterizar espacios vectoriales identificando independencia lineal, bases y dimensión, utilizando estos elementos en procesos de reducción de dimensiones.
- Aplicar conceptos de ortogonalidad, proyección y mínimos cuadrados, usándolos para el ajuste de modelos y aproximación de soluciones en problemas de regresión lineal.
- Aplicar los conceptos de autovalores y autovectores en áreas como análisis de datos y gráficos por computadora, entre otros.
- Aplicar los conocimientos a proyectos reales en áreas como: análisis de datos y gráficos por computadora, entre otros.
- Evaluar la estabilidad numérica de algoritmos de álgebra lineal (Gauss, LU, Cholesky, mínimos cuadrados, métodos iterativos) y los errores derivados del cálculo en aritmética de punto flotante.

III. Contenidos:

Exploración Computacional Inicial: Instalación, configuración y uso básico orientado a la temática del curso.

Ecuaciones matriciales y transformaciones: Ecuaciones de la forma $Ax = b$. Interpretación geométrica en 2D y 3D. Transformaciones geométricas con matrices (rotaciones, escalamiento, reflexiones). Manipulación de imágenes y figuras mediante matrices utilizando software.

Resolución de sistemas de ecuaciones: Implementación, comparación de eficiencia, estabilidad numérica y uso práctico de los algoritmos de resolución de sistemas de ecuaciones mediante la reducción por filas (Gauss, Gauss-Jordan) y la factorización matricial (LU y Cholesky), en programación. Aplicación computacional: resolución de sistemas en análisis de datos.

Bases y dimensión en espacios vectoriales: Dependencia e independencia lineal. Espacios y subespacios vectoriales. Conjunto generador de un espacio vectorial. Definición y construcción de bases de un espacio vectorial. Aplicación de la teoría

de subespacios vectoriales para la clasificación y organización de información en contextos computacionales.

Ortogonalidad y mínimos cuadrados: Producto punto y ortogonalidad. Proyección de vectores. Método de mínimos cuadrados para aproximación de datos. Bases ortogonales y ortonormales. Regresión lineal y ajuste de modelos.

Autovalores y autovectores: Definiciones y propiedades. Cálculo de autovalores y autovectores propios. Diagonalización de matrices. Interpretación geométrica. Análisis de estabilidad en algoritmos numéricos iterativos de la forma $x_{(n+1)} = Bx_n + c$.

Gestión de Datos II

I. Descripción del Curso

Este curso aborda los fundamentos teóricos y prácticos de la gestión de datos no relacionales en entornos modernos de desarrollo de software. Se estudian los modelos de datos alternativos al relacional —clave-valor, documento, columna y grafo—, sus mecanismos de almacenamiento, consulta y escalabilidad, así como los principios de consistencia, replicación y particionamiento en sistemas distribuidos. El curso enfatiza la comparación con el modelo relacional, el diseño orientado a requerimientos particulares, y la implementación práctica en motores NoSQL como MongoDB, Cassandra y Neo4j.

II. Objetivos

Objetivo general

Diseñar e implementar soluciones de almacenamiento y consulta basadas en modelos de datos no relacionales, aplicando principios de escalabilidad, consistencia y rendimiento.

Objetivos específicos:

- Analizar los fundamentos conceptuales y arquitectónicos de los modelos NoSQL.
- Comparar los diferentes modelos de datos no relacionales en función de sus casos de uso.
- Diseñar esquemas de datos y consultas optimizadas en motores NoSQL.
- Implementar sistemas distribuidos de almacenamiento con tolerancia a fallos.
- Aplicar principios de gestión de datos en la nube y prácticas de DevOps para despliegue y mantenimiento.

III. Contenidos:

1. Introducción y fundamentos: Limitaciones del modelo relacional. Principios CAP (consistencia, disponibilidad, particionamiento). Taxonomía de bases de datos NoSQL.
2. Modelos de datos no relacionales: Modelo clave-valor. Modelo de documentos. Modelos columnares. Modelo de grafos.
3. Escalabilidad y replicación: Fragmentación, replicación y consistencia eventual. Particionamiento horizontal y vertical. *Caching* y rendimiento.
4. Administración y seguridad: Respaldo y restauración. Autenticación y autorización. Monitoreo y métricas de rendimiento.
5. Aplicaciones prácticas: Integración de NoSQL en aplicaciones web/móviles. Comparación con sistemas relacionales híbridos. Casos de uso reales: datos masivos, internet de las cosas, analítica

Algoritmia Avanzada

I. Descripción del Curso

Este curso presupone conocimientos elementales de programación, diseño y análisis de algoritmos, estructuras de datos y teoría de la probabilidad. Se estudian diversos patrones de diseño de algoritmos, tales como división y conquista, programación dinámica, algoritmos voraces, algoritmos de vuelta atrás (*backtracking*). Se revisan algoritmos avanzados en áreas específicas tales como la criptografía, identificación de patrones, aprendizaje automático, simulación computacional, optimización y resolución de problemas complejos. También se estudian jerarquías de complejidad computacional.

II. Objetivos

Objetivo general:

Analizar y aplicar algoritmos avanzados en la solución de problemas computacionales complejos, tanto algoritmos que siguen patrones de diseño reconocidos como algoritmos avanzados en áreas específicas.

Objetivos específicos:

- Diseñar e implementar soluciones algorítmicas utilizando patrones algorítmicos avanzados.
- Aplicar métodos formales de análisis de algoritmos avanzados para la comparación de su eficiencia en tiempo y espacio.
- Explorar e integrar algoritmos especializados en áreas como criptografía, búsqueda de patrones, aprendizaje automático, optimización y simulación computacional.
- Usar apropiadamente el ciclo de vida de desarrollo de software para la construcción de software que aplique los algoritmos estudiados.
- Demostrar capacidad de comunicación e interacción efectiva con usuarios, dentro de un marco de trabajo en equipo y de comportamiento ético.

III. Contenidos:

1. Patrones de diseño algorítmico: división y conquista, programación dinámica, algoritmos voraces, vuelta atrás (*backtracking*) y búsqueda en espacio de estados.
2. Algoritmos avanzados en áreas específicas tales como la criptografía, identificación de patrones, aprendizaje automático, simulación computacional, optimización y resolución de problemas complejos.
3. Complejidad computacional, jerarquías de complejidad (P, NP, NP-completo), aprendizaje automático.

Ingeniería de Software

I. Descripción del Curso

En este curso se estudian, desde una perspectiva más conceptual, y se aplican, en un proyecto integrador, todas las etapas del ciclo de vida del desarrollo del software (CVDS-nivel 3): requerimientos, diseño, construcción, verificación y validación, despliegue y evolución. Los estudiantes interactuarán con usuarios reales, enfatizando el manejo de la relación con dichos usuarios y el trabajo en equipo. Se da continuidad y profundidad a lo aplicado en los cursos anteriores en Desarrollo de software. La aplicación para desarrollar deberá ser escalable y se deberá desarrollar para una plataforma especializada, tipo web o móvil.

II. Objetivos

Objetivo general:

Desarrollar un proyecto de ingeniería de software robusto, escalable y seguro,

interactuando y atendiendo las necesidades de usuarios reales.

Objetivos específicos:

- Examinar con mayor amplitud las etapas del ciclo de vida de desarrollo de software.
- Aplicar técnicas de elicitación y calidad de los requerimientos en interacción con los usuarios.
- Asegurar la calidad del software, desarrollando y aplicando un plan de pruebas comprensivo.
- Diseñar, implementar y desplegar el software requerido por los usuarios, usando técnicas de integración y entrega continuas.
- Demostrar capacidad comunicación e interacción efectiva con usuarios, dentro de un marco de trabajo en equipo y de comportamiento ético.

III Contenidos:

1. Interacción formal con usuarios.
2. Elicitación y validación de requerimientos.
3. Diseño de la arquitectura y de los servicios.
4. Implementación para plataformas especializadas: web y móvil.
5. Verificación y validación de software.
6. Integración y entrega continuas. Entrega y cierre de proyectos.

Computación y Modelado Matemático V

I. Descripción del Curso

Este curso introductorio establece los fundamentos matemáticos y estadísticos necesarios para el aprendizaje automático (*Machine Learning - ML*). Se estudian los principios de probabilidad, estadística inferencial y álgebra lineal aplicada, integrados con modelos de ML supervisados y no supervisados básicos. Incluye laboratorios prácticos en Python (o equivalentes), donde se implementan modelos sencillos y se analizan resultados.

Este curso sirve como base conceptual y práctica para Inteligencia Artificial I y II.

I. Objetivos

Objetivo general:

Implementar y evaluar modelos básicos de ML mediante la aplicación de los fundamentos de probabilidad y estadística al aprendizaje automático.

Objetivos específicos:

- Aplicar conceptos fundamentales de probabilidad y estadística — distribuciones más comunes, variables aleatorias básicas, estimación elemental e inferencia introductoria — en el contexto de problemas de ML.
- Aplicar álgebra lineal elemental en la formulación de modelos.
- Implementar modelos supervisados básicos (regresión lineal, regresión logística, k-NN).
- Aplicar métodos no supervisados básicos (*k-means*, *clustering* jerárquico, PCA).
- Evaluar modelos mediante métricas estándar (*accuracy*, precisión, recall, F1-score. Matrices de contingencia. ROC).
- Utilizar prácticas de validación cruzada y división de conjuntos de datos (*training/testing/validation*).
- Reflexionar sobre riesgos, limitaciones y aspectos éticos del uso de ML en aplicaciones iniciales.

III Contenidos:

- Fundamentos de probabilidad y estadística aplicados a ML.
- Álgebra lineal básica aplicada a ML.

- Modelos supervisados elementales (regresión, clasificación básica).
- Modelos no supervisados básicos (*clustering*, reducción de dimensionalidad).
- Validación de modelos y métricas de desempeño.
- Aspectos éticos y limitaciones de ML básico.

Comunicación Oral y Escrita para Computación

I Descripción del Curso

El estudiante desarrolla el empleo de forma clara y eficaz del lenguaje verbal en sus tres dimensiones fundamentales: la lectura, la escritura y la oralidad, para que, en su formación profesional, cuente con los medios lingüísticos y visuales (tecnológicos, gráficos) para la comunicación de sus pensamientos, elabore críticas, plante o solvante un problema entre pares o ante un público no especializado. De la misma manera, comprenda, enlace ideas y exponga su pensamiento crítico ante los discursos (orales y escritos) de otros.

El curso es teórico-práctico y responde, según el perfil profesional a la habilidad de comunicar en forma efectiva tanto oral como escrita, en el idioma de español.

II Objetivos

Objetivo general:

Dominar los conocimientos y habilidades de la comunicación escrita, oral y gestual necesarias para comunicarse en las diferentes áreas de las ciencias de la computación, considerando las reglas y normas del idioma español.

Objetivos específicos:

- Aplicar los procesos de lectura, escritura, escucha, habla e integración de los recursos pertinentes que se desarrollan en una competencia comunicativa para la resolución de las dificultades discursivas y de escritura en el ámbito profesional.
- Mejorar la capacidad de redacción para facilitar el proceso de comunicación con usuarios, colegas, y superiores
- Transmitir efectivamente sus ideas, tanto en forma oral como escrita.
- Hacer presentaciones efectivas tanto en forma oral como escrita.

III Contenidos

1. Proceso comunicativo (elementos constitutivos y elementos que obstaculizan el proceso).
2. Reglas ortográficas, estructura morfológica, sintáctica y léxica del texto como parte constitutiva de la estructura textual.
3. Estructura morfológica, sintáctica y léxica del texto.
4. Procesos cognitivos y elementos socioculturales implicados en el proceso de comprensión de lectura.
5. Los párrafos: una herramienta para ordenar la información.
6. Producción de textos: informes, ensayos, sistemas de citación y referencia
7. Habilidades en la comunicación oral.

Fundamentos de Lenguajes de Programación

I. Descripción del Curso

Este curso explora los fundamentos de los lenguajes de programación desde perspectivas computacionales tanto teóricas como prácticas. Se estudian modelos formales de sintaxis y semántica, así como paradigmas dominantes y sus bases lógico-matemáticas: programación funcional y cálculo- λ , programación lógica y lógicas de Horn, orientación a objetos y teorías de objetos. Se analizan sistemas de análisis estático de código y sus implicaciones para la seguridad, eficiencia y expresividad de los lenguajes. Se estudian

lenguajes y paradigmas que sirvan de modelo para la comprensión de la teoría cubierta. Asimismo, se contempla un bloque que ayude a conectar los fundamentos formales clásicos con los avances en modelos de lenguaje, mostrando cómo gramáticas, semánticas y sistemas de tipos siguen siendo relevantes para comprender y mejorar las capacidades de la IA generativa en generación de código.

El curso incluye prácticas de construcción de *parsers*, intérpretes y fragmentos de compiladores que permitan al estudiante aplicar los conceptos formales en el análisis, transformación y la implementación de lenguajes.

II. Objetivos

Objetivo general:

Implementar mecanismos de análisis, síntesis, compilación y evaluación de programas para la seguridad, eficiencia y expresividad de los lenguajes.

Objetivos específicos:

- Describir los componentes de la arquitectura de un lenguaje computacional en términos de su entorno de ejecución, incluyendo mecanismos de alcance, estrategias de ligadura (*binding*) y gestión de memoria y manejo del(os) procesador(es).
- Reconocer la evolución histórica de lenguajes “icónicos”, entendiendo las fuerzas técnicas y conceptuales que motivaron su creación y su influencia en los lenguajes contemporáneos.
- Explicar el diseño y propósito de características particulares (“*features*”) destacables de los lenguajes estudiados, valorando cómo reflejan decisiones de diseño pragmáticas y/o principios teóricos subyacentes.
- Analizar formalmente la sintaxis y semántica de lenguajes mediante gramáticas y sistemas formales de reglas.
- Comprender las nociones de complejidad hasta computabilidad en el contexto de lenguajes formales (máquinas abstractas, lenguajes formales, decidibilidad y límites de lo computable en el área de análisis, compilación y ejecución).
- Comprender los fundamentos del cálculo- λ y su papel en la programación funcional.
- Estudiar la programación lógica a partir de lógicas de Horn y la ejecución basada en resolución.
- Estudiar los fundamentos teóricos de la orientación a objetos y modelos para la teoría de objetos (pe., Abadi & Cardelli u comparables).
- Implementar analizadores de sintaxis y semántica, intérpretes y componentes de compiladores sencillos para experimentarlos con los conceptos estudiados.
- Explicar con criterio la relación entre fundamentos de lenguajes y modelos contemporáneos de IA generativa aplicados a la programación como posible paradigma.
- Desarrollar la capacidad de comparación de lenguajes y paradigmas desde perspectivas tanto teóricas como prácticas.

III Contenidos: Elementos básicos de Lenguajes Formales y Decidibilidad.

Gramáticas formales y análisis sintáctico. Generadores de Parsers.

Modelamiento formal de semántica de lenguajes de programación.

Programación funcional: cálculo- λ , funciones de orden superior, estrategias de evaluación (*eager*, *lazy*). Programación lógica: lógicas de Horn, unificación, resolución, semántica.

Orientación a objetos: teoría de objetos, encapsulamiento y subtipado. Paradigmas y requerimientos en programación reactiva, concurrente o asincrónica.

Sistemas de tipos: tipado estático y dinámico, polimorfismo, inferencia de tipos. Implementación práctica: *parsers*, intérpretes,

typers, fragmentos de compiladores. Estudio de lenguajes contemporáneos o no comunes, como estudio de caso (ej. Scala, Haskell, Elixir, OCaml, Elm, Go, Rust). Lenguajes formales e IA generativa, fundamentos de gramáticas, semántica y tipos en la IA generativa; comparación entre procesamiento clásico y generación estadística de lenguajes. Generación automática de código y verificación de sintaxis y tipos.

Inteligencia Artificial I

I. Descripción del Curso

Este curso introductorio a los fundamentos de la IA integra los enfoques denominados clásicos y modernos. Se establece el concepto de agente inteligente, se revisa la evolución histórica. Se estudian problemas de búsqueda y planificación básica, representación y razonamiento lógico, razonamiento probabilístico y la conexión con ML supervisado y no supervisado (partiendo de los conocimientos previos).

Como actualización, se incorpora la discusión de *arquitecturas híbridas* que combinen representaciones simbólicas (basados en reglas, KBs) con modelos fundacionales (ejemplo, LLMs), destacando cómo estas tendencias actuales buscan recuperar explicabilidad y control en agentes inteligentes modernos. Se incluyen laboratorios de manera opcional en un lenguaje de programación apropiado, mediante algoritmos de búsqueda, razonamiento y experimentos con modelos sencillos.

II. Objetivos

Objetivo general:

Comprender los fundamentos de la IA, conectando las aproximaciones simbólicas y estadísticas, con conciencia, responsabilidad profesional, ética y social.

Objetivos específicos:

- Conocer los problemas fundamentales de la IA y su evolución histórica.
- Implementar algoritmos de búsqueda (BFS, DFS, A*, heurísticas).
- Representar conocimiento mediante lógica proposicional y de predicados.
- Aplicar razonamiento probabilístico básico (redes bayesianas simples).
- Relacionar la IA simbólica con los enfoques modernos de ML destacando cómo los principios de búsqueda y razonamiento inspiran prácticas actuales como optimización y afinamiento.
- Analizar implicaciones éticas y sociales del uso de IA.

III. Contenidos:

1. Fundamentos de la IA y su historia.
2. Algoritmos de búsqueda y su conexión con la optimización en ML (ej. descenso de gradiente, búsqueda de hiperparámetros).
3. Representación y razonamiento con lógica.
4. Razonamiento probabilístico.
5. Vínculos entre IA simbólica y estadística.
6. Ética y responsabilidad social.

Ciberseguridad I

I. Descripción del Curso

Este curso introduce los fundamentos conceptuales y prácticos de la ciberseguridad, abarcando la identificación, análisis y mitigación de vulnerabilidades en sistemas informáticos y redes. Se enfatiza la comprensión de los principios de confidencialidad, integridad, disponibilidad y autenticación, así como la aplicación de estrategias de defensa en profundidad, controles físicos y digitales, y políticas de seguridad organizacional.

Además, se promueve el trabajo colaborativo y el pensamiento crítico para diseñar soluciones de protección alineadas con los estándares actuales del sector.

II. Objetivos

Objetivo general:

Analizar los principios, amenazas y mecanismos fundamentales de la ciberseguridad, mediante el análisis de conceptos clave, escenarios reales y estrategias de defensa, para el fortalecimiento de la protección de infraestructuras tecnológicas y datos en entornos organizacionales.

Objetivos específicos:

- Analizar el impacto en la protección de la información con la identificación de los principios fundamentales de la ciberseguridad, así como las principales amenazas, vulnerabilidades y riesgos en sistemas y redes, mediante el análisis de sus conceptos clave y casos introductorios.
- Evaluar estrategias básicas de prevención y respuesta mediante el análisis de mecanismos de defensa en profundidad y controles físicos, técnicos y administrativos.
- Aplicar buenas prácticas, políticas de seguridad y marcos normativos relevantes, utilizando herramientas y lineamientos actualizados, para el propósito del fortalecimiento de la protección de infraestructuras tecnológicas y datos en entornos organizacionales.

III Contenidos:

1. Fundamentos de la Gestión de Ciberseguridad
Conceptos fundamentales de la seguridad de la información y la importancia de la gestión. Normativas, legislación, clasificación de la información y gestión de riesgos e incidentes, como base para políticas organizacionales.
2. Vulnerabilidades y tipos de ataques
Vulnerabilidades en redes, hardware, software y acceso físico, así como ataques internos y externos, para comprender cómo se originan y evolucionan las amenazas en entornos tecnológicos.
3. Políticas de seguridad de la información
Creación, mantenimiento y aplicación de políticas, procedimientos y normas que estructuran la seguridad informática dentro de una organización, asegurando coherencia y cumplimiento normativo.
4. Seguridad en redes
Tecnologías y mecanismos como firewalls, IDS, VPN, VLAN e IPSec, para establecer defensas en la infraestructura de red, garantizando confidencialidad, integridad y disponibilidad en las comunicaciones.
5. Seguridad operacional
Protección física y humana mediante gestión de instalaciones, seguridad del personal, defensa en profundidad, controles físicos, sistemas de detección y respaldo eléctrico, como capas esenciales de seguridad.
6. Introducción al *Ethical Hacking*
Técnicas de evaluación de seguridad, recopilación de información, escaneo, ingeniería social y análisis de protocolos, con el fin de identificar vulnerabilidades desde una perspectiva defensiva.
7. Tecnologías de seguridad
Herramientas y arquitecturas avanzadas como firewalls, IDS y PKI. Certificados digitales y firma electrónica, para fortalecer la autenticación, el control de acceso y la protección de redes.
8. Planes de contingencia y continuidad del negocio
Conceptos y procesos de recuperación ante desastres y continuidad operativa,

integrando análisis de vulnerabilidades y planificación estratégica para mantener servicios críticos tras incidentes

Desarrollo para Plataformas

I. Descripción del Curso

Este curso aborda el diseño y la integración de ecosistemas de software que abarcan plataformas embebidas e interactivas, extendiendo los principios de ingeniería de software hacia entornos donde confluyen hardware y software. Se enfatizan aspectos como la gestión eficiente de recursos, la comunicación entre dispositivos, la interacción persona-computadora avanzada (voz, gestos, tacto) y la interoperabilidad entre plataformas. A través de proyectos prácticos, el estudiantado diseña y desarrolla soluciones que integran sensores, actuadores y mecanismos de interacción natural, aplicando principios de seguridad, eficiencia energética y accesibilidad, con un enfoque de arquitectura escalable y adaptable a entornos híbridos (embebidos, IoT e interfaces tangibles).

II. Objetivos

Objetivo general:

Diseñar e implementar soluciones multiplataforma que integren componentes embebidos e interactivos, garantizando interoperabilidad, seguridad, eficiencia y usabilidad, mediante la aplicación de arquitecturas modulares y técnicas de optimización de recursos.

Objetivos específicos:

- Analizar los componentes comunes y restricciones de plataformas embebidas e interactivas, incluyendo gestión de procesos, comunicación y consumo energético.
- Diseñar arquitecturas modulares que integren hardware, software y servicios distribuidos, asegurando interoperabilidad y escalabilidad.
- Implementar aplicaciones básicas en sistemas embebidos, con acceso a sensores y controladores mediante protocolos estándar (IoT, MQTT, BLE, etc.).
- Integrar interfaces naturales de interacción (voz, gestos, tacto, realidad aumentada) que mejoren la experiencia del usuario.
- Evaluar el rendimiento, consumo energético y seguridad de las aplicaciones desarrolladas, aplicando métricas de desempeño y pruebas de usabilidad.

III Contenidos:

1. Fundamentos de plataformas embebidas e interactivas: Concepto de plataforma y middleware ligero. Ciclo de vida y gestión de procesos en sistemas embebidos. Comunicación entre dispositivos (IoT, MQTT, HTTP, Bluetooth, ZigBee). Persistencia ligera de datos y sincronización. Seguridad, autenticación y permisos en entornos restringidos.
2. Plataformas embebidas: Arquitectura de sistemas embebidos (microcontroladores, Raspberry Pi, Arduino). Programación eficiente y gestión de energía. Interacción con sensores y actuadores. Control de rendimiento y tolerancia a fallos.
3. Plataformas interactivas: Fundamentos de computación e interacción natural. Diseño de interfaces tangibles, hápticas y multimodales. Integración de hardware interactivo. Introducción a entornos inmersivos y realidad aumentada. Pruebas de usabilidad y accesibilidad.
4. Integración multiplataforma: Diseño de arquitecturas híbridas que integren dispositivos embebidos e interfaces interactivas. Comunicación entre sistemas y sincronización de datos. Evaluación del desempeño, eficiencia y experiencia de usuario. Despliegue en entornos físicos o distribuidos.

Ingeniería de Datos I

I Descripción del Curso

Este curso introductorio a la Ingeniería de Datos abarca conceptos básicos de manejo y transformación de datos, infraestructura de datos y flujos ETL/ELT. Se trabaja con herramientas de código abierto y servicios en la nube. Incluye un trabajo práctico semanal en ambientes Linux y entornos virtualizados.

II Objetivos

Objetivo general:

Analizar el rol del ingeniero de datos en un ecosistema de AI y analítica a través de la construcción, integración y gestión de flujos de datos.

Objetivos específicos:

- Comprender los principios de adquisición, limpieza y almacenamiento de datos.
- Implementar flujos ETL/ELT simples con Python y SQL.
- Aplicar buenas prácticas de documentación, versionado y control de calidad de datos.
- Reconocer la importancia de la ética y la privacidad en la manipulación de datos.
- Conocer las bases del diseño multidimensional (hechos, dimensiones, granularidad).

III. Contenidos:

1. datos estructurados/no estructurados
2. Bases de datos relacionales y NoSQL
3. Introducción a data *lakes* y *warehouses*, *pipelines* básicos, diseño multidimensional (visión general).

Interfaz Humano-Computador

I. Descripción del Curso

En este curso se presentan los principios, modelos y métodos para el diseño, la implementación y la evaluación de sistemas de información interactivos, con énfasis en la usabilidad y la experiencia del usuario. Los estudiantes exploran los factores humanos, el diseño de interfaces, la accesibilidad, la creación de prototipos y las técnicas de evaluación. Se hace hincapié en los procesos de diseño centrados en el usuario que integran perspectivas cognitivas, sociales y culturales. Mediante actividades prácticas, los estudiantes desarrollan habilidades para diseñar y evaluar interfaces que mejoran la eficacia, la eficiencia y la satisfacción, preparándolos para crear tecnología que se ajuste a las necesidades y comportamientos humanos.

II. Objetivos

Objetivo general:

Diseñar, implementar y evaluar sistemas de información interactivos a través de procesos iterativos y centrados en el usuario.

Objetivos específicos:

- Analizar el proceso iterativo de diseño y desarrollo centrado en el usuario para un proyecto de desarrollo de sistemas de información interactivos.
- Identificar y analizar críticamente los posibles impactos de un diseño en la sociedad y las comunidades relevantes para el abordaje de cuestiones como la sostenibilidad, la inclusión, la seguridad, la privacidad, los daños y el impacto desigual.

- Identificar los desafíos de accesibilidad que enfrentan las personas con capacidades diferentes y especificar las tecnologías accesibles y de asistencia que pueda dar soporte a la experiencia de este tipo de usuarios.
- Analizar las diferencias entre la evaluación formativa y sumativa y su papel en la evaluación del diseño de un sistema de información interactivo.

III. Contenidos:

1. Comprensión del usuario (características físicas y cognitivas del usuario, poblaciones de usuarios diversas, colaboración y comunicación).
2. Diseño centrado en el usuario, metodologías de diseño centrado en el usuario (técnicas, impacto del diseño, ética en los métodos y soluciones de diseño, requisitos en el diseño, diseño sensible al valor, persuasión a través del diseño).
3. Generalidades del apoyo social, legal y obligaciones con las personas con discapacidad (estándares de accesibilidad, marcos de diseño inclusivo (IDF)).
4. Diseño universal, diseño de UX (experiencia de usuario) y prácticas de investigación que facilitan la inclusión y la accesibilidad).
5. Métodos formativos y sumativos para la evaluación de la usabilidad y la experiencia de usuario. Métodos formativos y sumativos para la inspección de la usabilidad y la experiencia de usuario.
6. Planificación de estudios de usabilidad y experiencia de usuario, técnicas y herramientas para el análisis cualitativo y cuantitativo.

Inteligencia Artificial II

I. Descripción del Curso

Este curso avanzado explora las técnicas contemporáneas de IA: *deep learning*, aprendizaje reforzado, NLP, visión por computadora y agentes inteligentes. Se enfatiza la tendencia actual de *Agentic Programming*, mostrando cómo los agentes se orquestran con modelos modernos (incluyendo LLMs) para planificar y ejecutar tareas en forma de agentes. Se integran metodologías de MLOps para el ciclo de vida de modelos en producción, junto con reflexiones sobre ética, gobernanza y seguridad. El curso pretende ofrecer una visión integrada de las principales técnicas contemporáneas de IA, con prácticas seleccionadas que permitan al estudiante experimentar con arquitecturas y aplicaciones representativas. Los laboratorios utilizan herramientas apropiadas, con casos aplicados a texto, imágenes y entornos simulados.

II Objetivos

Objetivo general:

Diseñar, entrenar y desplegar modelos y agentes en entornos profesionales con responsabilidad social, a través de la aplicación de competencias avanzadas en IA moderna.

Objetivos específicos:

- Construir y entrenar redes neuronales profundas (CNN, RNN, *Transformers*).
- Aplicar algoritmos de aprendizaje reforzado (RL) en entornos simulados.
- Implementar técnicas de NLP (*embeddings*, *Transformers*) y visión (CNN).
- Implementar y evaluar modelos generativos (GANs, VAEs, *Diffusion Models* o comparables) y explorar sus aplicaciones.
- Explorar *Agentic Programming* como evolución de la planificación y los agentes clásicos y el uso de *frameworks* y protocolos de integración de agentes.
- Diseñar pipelines de MLOps para entrenamiento, despliegue y monitoreo de modelos.

- Analizar implicaciones éticas, de seguridad y gobernanza en sistemas de IA.
- III Contenidos:
1. Redes neuronales profundas: arquitecturas avanzadas (CNN, RNN, *Transformers*) y sus aplicaciones en visión y NLP.
 2. Aprendizaje reforzado (RL): agentes en entornos simulados, exploración/explotación, aplicaciones en control.
 3. Modelos generativos: GANs, VAEs, *Diffusion Models* (o comparables) y sus aplicaciones en síntesis de datos, imágenes y texto.
 4. Procesamiento de lenguaje natural (NLP): *embeddings*, *Transformers*, LLMs y aplicaciones en comprensión y generación de lenguaje.
 5. Visión por computadora: reconocimiento de imágenes, detección de objetos y aplicaciones industriales.
 6. *Agentic Programming*: evolución de la planificación clásica hacia agentes modernos orquestados con LLMs y protocolos de integración.
 7. MLOps: *pipelines* de entrenamiento, despliegue, monitoreo y actualización de modelos en producción.
 8. Ética, gobernanza y seguridad: sesgos, transparencia, uso responsable de modelos avanzados.

Ciberseguridad II

I. Descripción del Curso

El curso introduce principios y prácticas aplicadas de ciberseguridad para proteger infraestructuras tecnológicas y activos digitales. Aborda amenazas, vulnerabilidades, políticas y controles de seguridad, así como estándares internacionales y prácticas de defensa técnica. Mediante actividades teórico-prácticas, el estudiantado desarrolla competencias para analizar riesgos, aplicar mecanismos de protección y responder ante incidentes de seguridad.

II. Objetivos

Objetivo general:

Aplicar políticas, estándares, técnicas y herramientas de ciberseguridad, con el fin de fortalecer la protección de los activos de información y la resiliencia organizacional frente a incidentes.

Objetivos específicos:

- Formular estrategias de mitigación efectivas mediante el análisis de amenazas y vulnerabilidades en entornos tecnológicos empresariales, a través de la identificación de vectores de ataque internos y externos.
- Diseñar políticas y controles de seguridad coherentes con los objetivos organizacionales, mediante la gestión de accesos, el control de puertos y la aplicación de normativas, para la protección de la infraestructura y la información.
- Aplicar los principios fundamentales de la ciberseguridad (confidencialidad, integridad, disponibilidad y autenticación) a través de ejercicios prácticos en redes y sistemas, para el establecimiento de entornos seguros y confiables.
- Desarrollar actitudes éticas, investigativas y colaborativas, con el fin de promover el aprendizaje continuo y el fortalecimiento de la capacidad de respuesta ante incidentes reales de ciberseguridad.

III. Contenidos:

1. Gobernanza y gestión de la seguridad

Introduce los principios, normativas y prácticas para la administración estratégica de la seguridad de la información, incluyendo gestión de riesgos,

- cumplimiento regulatorio y definición de políticas organizacionales.
2. Amenazas, vulnerabilidades y ataques
Analiza las principales amenazas y vulnerabilidades que afectan redes, sistemas y aplicaciones, abordando vectores de ataque como malware, phishing, fuerza bruta y denegación de servicio.
 3. Políticas y gestión de la seguridad de la información
Aborda la formulación, implementación y evaluación de políticas y procedimientos que regulan el uso, protección y gestión de la información dentro de una organización.
 4. Seguridad en redes y comunicaciones
Profundiza en mecanismos y protocolos avanzados para proteger redes cableadas e inalámbricas, incluyendo firewalls, IDS/IPS, cifrado, certificados digitales y seguridad en tecnologías emergentes (IoT, 5G).
 5. Seguridad operacional de infraestructuras
Estudia las prácticas y controles destinados a asegurar la operación continua y segura de sistemas tecnológicos, abarcando monitoreo, control de accesos, *hardening* y mantenimiento seguro.
 6. *Hacking* ético y pruebas de penetración
Introduce metodologías de pruebas de penetración y análisis ético de vulnerabilidades para identificar y mitigar riesgos de seguridad, siguiendo marcos legales y buenas prácticas profesionales.
 7. Planes de contingencia y continuidad del negocio
Presenta estrategias y procedimientos para garantizar la disponibilidad de servicios críticos ante incidentes, incluyendo recuperación ante desastres, respaldos, redundancia y respuesta organizada.
 8. Tecnologías y mecanismos de seguridad
Explora soluciones y herramientas técnicas clave, como cortafuegos, sistemas de detección y prevención de intrusiones, VPN, PKI y mecanismos de cifrado, aplicadas a la protección de infraestructuras tecnológicas.

Ingeniería de Datos II

I Descripción del Curso

Este curso es en sistemas distribuidos, Big Data y prácticas avanzadas de DE. Se enfoca en el diseño de *pipelines* escalables, procesamiento en *streaming* y almacenamiento optimizado en la nube. Incluye laboratorios en Spark y un proyecto final grupal con *datasets* reales.

II Objetivos

Objetivo general:

Implementar soluciones de ingeniería de datos escalables y sostenibles, aplicables a entornos empresariales y científicos.

Objetivos específicos:

- Implementar pipelines de datos en *batch* y *streaming* con herramientas distribuidas (p.ej. Spark, Kafka).
- Diseñar esquemas de almacenamiento eficientes en entornos de la nube (*cloud*).
- Incorporar pruebas, monitoreo y control de calidad de datos en proyectos de DE.
- Integrar prácticas de gobernanza y seguridad en la gestión de datos.
- Analizar el rol del *data warehouse* moderno frente a *data lakes* y *lakehouses*

III. Contenidos:

1. Procesamiento distribuido

2. Arquitectura lambda/kappa
3. *Data lakes* versus *warehouses*
4. Orquestación avanzada (*Airflow/Dagster*)
5. Calidad de datos
6. Seguridad en entornos *cloud*.

Seminario I

I. Descripción del Curso

Este primer curso de Seminario busca introducir al estudiante en el proceso de investigación en ciencias de la computación, desarrollando competencias para identificar un área de interés, realizar búsquedas bibliográficas, estudiar, reseñar y presentar artículos académicos sobre el área seleccionada, para finalmente plantear un proyecto de investigación propio.

II. Objetivos

Objetivo general:

Plantear formalmente un proyecto de investigación relacionado con una de las áreas disciplinares del plan de estudios.

Objetivos específicos:

- Comprender los pasos fundamentales del proceso de investigación científica.
- Dominar las técnicas de búsqueda, análisis y síntesis de literatura científica.
- Analizar los aspectos éticos, sociales y profesionales de la investigación.
- Redactar un protocolo inicial de investigación con objetivos, hipótesis y metodología preliminar.

III. Contenidos:

1. El proceso de investigación científica.
2. Formulación de un proyecto de investigación: objetivos, hipótesis y metodología.
3. Lectura, reseña y exposición de artículos científicos.

Práctica Profesional Supervisada

I. Descripción del Curso

El curso consiste en establecer una estrategia de vinculación con el sector empresarial y público, mediante la realización de una práctica profesional y, a la vez, fortalecer las habilidades socioemocionales del estudiante. Esta práctica tendrá una supervisión y seguimiento por parte de la Universidad a través de la designación de un responsable del curso durante el ciclo lectivo. Durante la práctica el estudiante deberá desenvolverse en un ambiente de trabajo, con una asignación de tiempo de al menos 20 horas presenciales, virtuales o mixto por semana, según las condiciones de la organización.

II. Objetivos

Objetivo general:

Aplicar los conocimientos y las habilidades obtenidas en la carrera de Bachillerato en Ingeniería en Computación, mediante la inserción en un ambiente profesional real dentro de una organización.

Objetivos específicos:

- Aplicar las habilidades y los conocimientos adquiridos en la carrera.
- Implementar una solución tecnológica en un ambiente real.
- Ejercitar sus conocimientos y habilidades de relacionamiento interpersonal en un

entorno productivo y de trabajo en equipo en una organización.

- Adquirir experiencia a partir de una situación real, bajo una cultura organizacional, una jerarquía superior e interactuando con usuarios reales para satisfacer una necesidad en una organización y su contexto.
- Elaborar un plan para la administración de su proyecto en la práctica profesional.
- Reflexionar respecto al planteamiento y administración de los proyectos de los otros estudiantes del curso.

III Contenidos:

1. Administración de proyectos
2. Administración, diseño e implementación de bases de datos
3. Diseño, implementación y mantenimiento de redes de computadores
4. Diseño, implementación y mantenimiento de sistemas de información
5. Configuración y administración de plataformas tecnológicas
6. Gestión de la configuración del software
7. Innovación
8. *Testing* y aseguramiento de la calidad (QA).

Seminario II

I Descripción del Curso

En este segundo curso de Seminario II se desarrolla la investigación planteada en el curso de Seminario I, exponiendo regularmente los avances de la investigación en un área de la Ingeniería de Computación y la generación de resultados con su respectivo análisis. Además, dicha investigación deberá escribirse en formato de artículo científico para ser presentado en un foro académico.

II Objetivos

Objetivo general:

Desarrollar el proyecto de investigación planteado en el Seminario I, generando resultados, análisis y un artículo científico listo para ser presentado en un foro académico.

Objetivos específicos:

- Desarrollar la solución propuesta al problema de investigación planteado.
- Procesar y analizar los datos obtenidos utilizando herramientas estadísticas y computacionales adecuadas.
- Evaluar los resultados en relación con el estado del arte y discutir su relevancia.
- Comunicar los hallazgos mediante un artículo científico y presentaciones orales.

III Contenidos:

1. Análisis de datos y validación de resultados.
2. Ética en la publicación de resultados y manejo de datos sensibles.
3. Estructura de un artículo académico.
4. Preparación de presentaciones orales y carteles científicos.

Cursos Optativos Disciplinarios **Informática Forense**

I. Descripción del Curso

Este curso introduce los fundamentos teóricos, técnicos y legales de la informática forense como disciplina esencial en la investigación de incidentes de ciberseguridad y delitos informáticos. Se estudian metodologías de recolección, preservación, análisis y presentación de evidencia digital conforme a principios éticos y marcos legales nacionales e internacionales.

El curso aborda la cadena de custodia, análisis forense en sistemas operativos, redes, dispositivos móviles y entornos en la nube. El estudiantado desarrollará competencias para utilizar herramientas forenses, documentar hallazgos y presentar resultados técnicos en contextos judiciales o administrativos, con énfasis en la integridad, trazabilidad y confidencialidad de la evidencia.

II. Objetivos

Objetivo general:

Analizar la evidencia digital en distintos entornos tecnológicos, conforme a principios éticos, normativos y legales, para el apoyo de investigaciones de incidentes de seguridad y delitos informáticos.

Objetivos específicos:

- Analizar los fundamentos conceptuales, éticos y legales de la informática forense y su relación con la ciberseguridad, la evidencia digital y la cadena de custodia.
- Aplicar procedimientos estandarizados de identificación, preservación y adquisición de evidencia digital, utilizando herramientas especializadas que garanticen su integridad y trazabilidad.
- Detectar actividades sospechosas, intrusiones o fraudes digitales mediante el examen de sistemas operativos, redes y dispositivos móviles.
- Evaluar la evidencia obtenida, correlacionando eventos y datos de múltiples fuentes, con el fin de reconstruir incidentes y establecer hipótesis fundamentadas.
- Elaborar informes técnicos y periciales que documenten los hallazgos de la investigación, cumpliendo con estándares profesionales y legales de comunicación de evidencia.

III. Contenidos:

- 1- Fundamentos de la informática forense
 - Concepto, alcance y objetivos de la informática forense.
 - Ética profesional, normativa legal y marcos internacionales (ISO/IEC 27037, 27041, NIST SP 800-86).
 - Tipos de evidencia digital y cadena de custodia.
 - Rol del perito forense y relación con la ciberseguridad organizacional.
- 2- Procesos y metodologías forenses
 - Fases de la investigación forense (identificación, preservación, análisis, presentación).
 - Modelos de referencia: DFRWS, NIST, ACPO.
 - Herramientas y entornos de trabajo forense (*EnCase*, *Autopsy*, *FTK*, *Sleuth Kit*, *Volatility*, *Wireshark*).
 - Documentación y registro de evidencias.
- 3- Análisis forense de sistemas y archivos
 - Estructura de sistemas de archivos (NTFS, EXT, FAT).
 - Recuperación de archivos borrados y análisis de metadatos.
 - Forense de memoria y registro (RAM, *logs*, *hives* de Windows).
 - Evidencia en sistemas Windows, Linux y macOS.
- 4- Análisis forense de redes y dispositivos móviles
 - Captura y análisis de tráfico de red.
 - Forense de correo electrónico y mensajería instantánea.
 - Detección de intrusiones y correlación de eventos.
 - Análisis forense en Android e iOS (imágenes lógicas y físicas, extracción de datos, geolocalización).
- 5- Forense en entornos emergentes
 - Computación en la nube y retos de la evidencia distribuida.
 - Forense en IoT, entornos virtualizados y contenedores.

- Criptografía, esteganografía y recuperación de información cifrada.
- Inteligencia de amenazas y análisis de malware en contextos forenses.
- 6- Comunicación de resultados
 - Elaboración de informes técnicos y periciales.
 - Presentación de hallazgos ante audiencias técnicas y no técnicas.

Métodos Numéricos Aplicados a la Ingeniería

I Descripción del Curso

Este curso optativo aborda los métodos numéricos fundamentales para la resolución de problemas ingenieriles, centrándose en cuatro áreas esenciales: la solución de ecuaciones en una variable mediante métodos como bisección, Newton-Raphson, secante y Müller; la aproximación polinomial a través de interpolación de Lagrange, diferencias divididas y trazadores cúbicos; la derivación e integración numérica incluyendo técnicas de extrapolación de Richardson, integración de Romberg, métodos gaussianos e integrales múltiples; y la resolución de sistemas lineales mediante métodos directos con estrategias de pivoteo y métodos iterativos.

El curso combina teoría y práctica mediante el uso de herramientas computacionales interactivas como por ejemplo las basadas en lenguaje Wolfram (CDF), permitiendo a los estudiantes visualizar y experimentar con los algoritmos de forma eficiente.

II Objetivos

Objetivo general:

Analizar y aplicar métodos numéricos fundamentales en la resolución de problemas de ingeniería, integrando el uso de herramientas computacionales para la visualización, implementación y validación de los algoritmos estudiados.

Objetivos específicos:

- Aplicar métodos iterativos y cerrados (bisección, Newton-Raphson, secante, Regula-Falsi y Müller) para determinar las raíces de ecuaciones algebraicas y trascendentes, evaluando su convergencia y precisión.
- Implementar técnicas de derivación numérica de diferentes órdenes y métodos de extrapolación de Richardson para estimar derivadas de funciones con control de error.
- Calcular integrales definidas mediante métodos de cuadratura simple, compuesta, Romberg, adaptativa y gaussiana, incluyendo la evaluación numérica de integrales múltiples.
- Resolver sistemas de ecuaciones lineales utilizando métodos directos con estrategias de pivoteo y métodos iterativos (Jacobi, Gauss-Seidel), analizando la eficiencia computacional y convergencia de cada técnica.

III Contenidos:

- 1- Concepto de raíz de una ecuación y condiciones de existencia. Métodos cerrados: método de bisección, convergencia y análisis de error. Método de Regula-Falsi y comparación con bisección. Iteración de punto fijo: teoremas de convergencia y condiciones suficientes. Método de Newton-Raphson: derivación, interpretación geométrica y orden de convergencia. Aplicación de Newton-Raphson con división sintética para polinomios. Método de la secante: ventajas y desventajas respecto a Newton-Raphson. Método de Müller para raíces reales y complejas. Técnicas de aceleración de convergencia: método de Steffensen y proceso Δ^2 de Aitken. Criterios de parada y análisis de error en métodos iterativos. Implementación computacional y visualización usando herramientas de software.
- 2- Problema general de interpolación y teorema de existencia y unicidad. Interpolación de Lagrange: forma canónica, propiedades y error de interpolación. Método de

Neville para evaluación eficiente de polinomios de interpolación. Diferencias divididas de Newton: construcción recursiva y propiedades. Interpolación paramétrica usando Lagrange para curvas en el plano. Splines o trazadores: concepto y clasificación. Trazadores cúbicos: condiciones de suavidad, frontera natural, sujeta y periódica. Construcción de trazadores cúbicos mediante sistemas tridiagonales. Aplicaciones de trazadores en diseño gráfico y modelado de curvas. Análisis del error de interpolación y fenómeno de Runge. Implementación computacional de algoritmos de interpolación.

- 3- Derivación numérica simple: fórmulas de diferencias finitas hacia adelante, atrás y centradas. Derivación de orden superior: segundas derivadas y derivadas de mayor orden. Análisis de error de truncamiento en fórmulas de derivación numérica. Extrapolación de Richardson para mejorar aproximaciones de derivadas. Integración numérica simple: regla del trapecio y regla de Simpson. Fórmulas de Newton-Cotes cerradas y abiertas. Integración numérica compuesta: trapecio compuesto, Simpson compuesto y análisis de error. Integración de Romberg: tabla de Romberg y extrapolación sucesiva. Integración numérica adaptativa: estrategias de refinamiento automático. Grado de exactitud de fórmulas de integración y concepto de precisión. Integración gaussiana: polinomios de Legendre, nodos y pesos de Gauss. Cuadratura gaussiana de orden superior y aplicaciones. Cálculo numérico de integrales dobles y triples: reglas compuestas bidimensionales. Implementación y visualización de métodos de integración mediante software.
- 4- Repaso de sistemas de ecuaciones lineales y notación matricial. Métodos directos: eliminación gaussiana simple y operaciones elementales. Estrategias de pivoteo: pivoteo parcial, total y escalado para mejorar estabilidad numérica. Descomposición LU: factorización y aplicaciones en resolución de sistemas múltiples. Descomposición de Cholesky para matrices simétricas definidas positivas. Análisis de error de redondeo en métodos directos y número de condición. Métodos iterativos: conceptos generales de convergencia y criterios de parada. Método de Jacobi: formulación matricial, condiciones de convergencia y radio espectral. Método de Gauss-Seidel: ventajas sobre Jacobi y análisis de convergencia. Método de relajación sucesiva (SOR) y elección del parámetro de relajación óptimo. Criterio de diagonal dominante y teoremas de convergencia para métodos iterativos. Normas matriciales y vectoriales aplicadas al análisis de error. Comparación de eficiencia computacional entre métodos directos e iterativos. Implementación computacional y visualización de convergencia usando herramientas de software.

Modelado Multidimensional e Inteligencia de Negocios

I Descripción del Curso

Este curso optativo profundiza en el diseño de modelos multidimensionales y su aplicación en *business intelligence* (BI). Se exploran esquemas en estrella y copo de nieve, cubos OLAP, y la implementación práctica de consultas y visualizaciones en herramientas modernas de BI. Incluye laboratorios prácticos en entornos de *data warehouse* y proyectos con herramientas de análisis.

Se estudiarán los conceptos esenciales de gestión tecnológica empresarial, de las tendencias tecnológicas que interactúan con la gestión interna y externa de las principales funciones de la organización. Interesa el dominio de los conocimientos de sistemas de información gerencial, la infraestructura tecnología asociada a Internet y en la nube, así como la estrategia de negocios. Se desarrollan las habilidades para el pensamiento creativo, innovador y crítico y la capacidad para el diseño de proyectos utilizando metodologías ágiles para la toma de decisiones. También se desarrollan valores como el

respeto, la ética profesional, la no distinción de etnia, edad, género y la tolerancia para trabajar en equipo

II Objetivos

Objetivo general:

Diseñar e implementar modelos multidimensionales para la generación de información de valor en entornos empresariales.

Objetivos específicos:

- Diseñar modelos en estrella y copo de nieve.
- Implementar y consultar cubos OLAP.
- Analizar la pertinencia del modelado multidimensional frente a arquitecturas modernas de datos.
- Utilizar herramientas de BI (tipo *Power BI*, *Tableau*) para generar reportes y tableros (**dashboards**).
- Evaluar consideraciones éticas y de privacidad en BI

III Contenidos:

1. Fundamentos de modelado multidimensional, OLAP, esquemas en estrella/copo de nieve, herramientas BI, integración con *data warehouses* y *lakehouses*.
2. Inteligencia de negocios y toma de decisiones.
3. Información estratégica e inteligencia de negocios con Power BI
4. Grandes datos (*Big Data*) y el manejo de grandes cantidades de información.
5. Medios sociales, confidencialidad y noticias falsas.
6. Herramientas para inteligencia de negocios.

Computación en la Nube Avanzado

I Descripción del Curso

El curso Computación en la Nube Avanzado profundiza en el diseño, implementación, administración y seguridad de soluciones distribuidas en entornos *multicloud* e híbridos.

Se estudian arquitecturas escalables orientadas a servicios, la integración de inteligencia artificial y *big data* en la nube, el uso de contenedores y orquestadores, la automatización mediante infraestructura como código (IaC) y la optimización de costos y rendimiento.

El curso enfatiza el desarrollo de competencias técnicas y estratégicas que permitan a los estudiantes diseñar soluciones de nube sostenibles, seguras y alineadas con las tendencias internacionales

II Objetivos

Objetivo general:

Profundizar en el uso estratégico de la computación en la nube para el análisis de desafíos complejos en las áreas de ciencia de datos, ciberseguridad y gestión avanzada de bases de datos, utilizando herramientas y servicios para nube especializados.

Objetivos específicos:

- Diseñar arquitecturas *cloud* escalables y seguras para aplicaciones críticas y entornos de *big data*.
- Aplicar servicios *cloud* avanzados para el procesamiento, análisis y visualización de datos (e.g., *AWS SageMaker*, *Google BigQuery*, *Azure Machine Learning*).
- Implementar estrategias de ciberseguridad en la nube, incluyendo "*identity and access management*" (IAM), encriptación avanzada y monitorización de *threats*.
- Gestionar bases de datos *cloud* de alto rendimiento y disponibilidad, con enfoque en optimización y costos.
- Desarrollar proyectos integradores que simulen escenarios reales en ciencia de datos, ciberseguridad y desarrollo de software

- III Contenidos:
1. Arquitecturas y modelos avanzados de computación en la nube
Modelos de despliegue híbrido y *multicloud*
Contenedores y orquestadores (Docker, Kubernetes)
Microservicios y *serverless computing*
 2. Automatización y DevSecOps
Infraestructura como código (Terraform, Ansible)
Integración y entrega continua (CI/CD)
Observabilidad y *logging* distribuido
 3. Ciencia de datos e inteligencia artificial en la nube
Procesamiento y almacenamiento distribuido (Data Lake, BigQuery, Redshift)
Machine Learning y MLOps en la nube
Ética, sostenibilidad y gobernanza de datos
 4. Seguridad, cumplimiento y costos
IAM, Zero Trust y encriptación
Evaluación de riesgos y auditorías de seguridad
Optimización de costos y sostenibilidad energética
 5. Proyecto integrador
Diseño y despliegue de una solución completa en la nube
Presentación y defensa técnica

Introducción al Análisis de Imágenes Digitales

I Descripción del Curso

El curso de Análisis y Diseño de Imágenes Digitales proporciona al estudiantado una introducción integral al procesamiento, análisis y manipulación de imágenes en el dominio digital. A lo largo del curso, se exploran los fundamentos teóricos y prácticos necesarios para comprender cómo las imágenes son representadas, transformadas y procesadas digitalmente, desde los principios de adquisición y representación hasta las aplicaciones avanzadas en reconocimiento de patrones y visión por computadora.

El curso combina actividades teóricas y prácticas para que los estudiantes desarrollen competencias en el uso de herramientas y bibliotecas especializadas de programación. Además, se enfatiza el análisis crítico, la creatividad y la aplicación ética de estas tecnologías en contextos reales de ingeniería, ciencia y tecnología.

II Objetivos

Objetivo general:

Dominar los principios, técnicas y herramientas empleadas en el análisis, diseño y procesamiento digital de imágenes, para aplicarlas en la resolución de problemas reales en el ámbito de la ingeniería en computación.

Objetivos específicos:

- Comprender los fundamentos de la representación digital de imágenes y la formación del color.
- Aplicar técnicas de procesamiento digital para mejorar la calidad, el contraste y la visualización de imágenes.
- Implementar métodos de segmentación para separar objetos de interés en imágenes complejas.
- Utilizar algoritmos de extracción de características para analizar patrones, texturas y estructuras visuales.
- Emplear transformaciones geométricas para modificar la orientación, el tamaño o la perspectiva de las imágenes.
- Aplicar técnicas básicas de reconocimiento de patrones y visión por

computadora.

- Desarrollar proyectos prácticos que integren herramientas de análisis y diseño de imágenes en contextos reales.
- Aplicar el pensamiento crítico, la creatividad y la ética profesional en las tecnologías de procesamiento de imágenes.

III Contenidos

1. Introducción a la imagen digital y sistemas de adquisición.
2. Fundamentos del color y modelos de representación (RGB, HSV, CMYK, YCbCr).
3. Operaciones básicas sobre imágenes: filtrado espacial, convolución y transformaciones de intensidad.
4. Transformaciones geométricas: rotación, escalamiento, traslación y deformaciones.
5. Técnicas de mejora y restauración de imágenes.
6. Segmentación: umbralización, detección de bordes, crecimiento de regiones y *clustering*.
7. Extracción de características: forma, textura y color.
8. Introducción al reconocimiento de patrones y visión por computadora.
9. Herramientas y librerías de procesamiento digital de imágenes (Python + OpenCV).
10. Casos de estudio y aplicaciones prácticas en ingeniería, medicina, seguridad y automatización.

Optativos Libres

Introducción a la Creación de Empresas

I Descripción del Curso

Las organizaciones hoy día han cambiado su entorno de modelos de negocio analógicos a digitales y esto genera la necesidad de crear e innovar estrategias que ayuden a las empresas a delinear rutas de transformación digital y a tomar decisiones rápidas y acertadas.

Este curso fortalece a los estudiantes para la toma de decisiones, el desarrollo de una mentalidad innovadora, mejora de las habilidades de liderazgo, así como que despierten la conciencia de la importancia que tienen las tecnologías de información y comunicación (TICs) para desempeñar mejor sus funciones y por lo tanto mejorar su productividad, la de sus equipos de trabajo y de la organización.

El curso utilizará recursos teórico-prácticos mediante estrategias metodológicas y pedagógicas que motiven, impulsen y potencialicen el espíritu emprendedor en los y las estudiantes, la energía creativa, así como el emprendimiento de esa creatividad en acciones empresariales y sociales.

Se brindará una visión integral de los beneficios y riesgos que conllevan el desarrollo de una cultura emprendedora dirigida hacia el desarrollo de una cultura empresarial, promoviendo los argumentos para tomar una decisión en su plan de vida personal y profesional.

Así mismo, el curso crea metodología para la integración ambiental y social de manera transversal en el desarrollo de los proyectos, promoviendo la incorporación de estos aspectos desde la formulación de la idea hasta la conclusión del plan de negocios.

II Objetivos

Objetivo general:

Desarrollar una cultura emprendedora que potencie la capacidad creativa en el planteamiento de proyectos responsables social y ambientalmente, para la

contribución al bienestar común.

Objetivos específicos:

- Analizar la importancia que tienen las tecnologías de información y comunicación (TIC) en los negocios para el óptimo desempeño de las funciones gerenciales, la mejora de la productividad, de los equipos de trabajo y de la organización considerando las nuevas estrategias y metodologías digitales
- Analizar las condiciones cambiantes del entorno nacional y externo, relacionado con la dinámica de la Mipymes y su impacto en la vida profesional y empresarial.
- Promover el intercambio de experiencias con empresarios y emprendedores sociales exitosos que permitan la reflexión sobre los retos y desafíos, así como el funcionamiento y componentes de una organización.

III Contenidos:

1. Tendencias tecnológicas actuales en el ámbito gerencial.
2. Infraestructura tecnológica e Internet.
3. Sistemas de información gerenciales en la nube.
4. Espíritu emprendedor. ¿Por qué emprender? Emprendimiento social
5. Plan de Negocio: Naturaleza del Proyecto
6. Plan de Mercadotecnia. Estudio Técnico
7. Organización y aspectos legales
8. Generalidades de Finanzas.

Las TIC en el ámbito jurídico de Costa Rica

I Descripción del Curso

El curso aborda la relación entre derecho y tecnología, analizando cómo los procesos de globalización y el cambio tecnológico transforman la vida social y generan nuevos desafíos jurídicos. Se estudia cómo distintas ramas del derecho —como el penal, comercial, laboral y de propiedad intelectual— han incorporado la tecnología en su ámbito de aplicación, y cómo esta interacción ha dado lugar a un campo especializado que examina los efectos jurídicos del desarrollo tecnológico y digital.

Además, el curso aborda los conocimientos legales fundamentales para comprender las implicaciones jurídicas del ejercicio profesional. Se revisan principios básicos de derecho laboral, administrativo, comercial y de propiedad intelectual, con el fin de que los futuros profesionales conozcan sus derechos, deberes y responsabilidades al desarrollar software, emprender proyectos tecnológicos o colaborar con instituciones públicas y privadas.

9. Objetivos

Objetivo general:

Dominar los conocimientos jurídicos básicos, para comprender las relaciones entre el Derecho, los desarrollos y aplicaciones de la Informática y del avance tecnológico.

Objetivos específicos:

- Identificar y aplicar conceptos jurídicos generales relacionados con las TIC.
- Comprender las causas y consecuencias, derivadas del Derecho, sobre la tecnología y los fenómenos sociales vinculados a ésta.
- Utilizar figuras y herramientas jurídicas en el marco de la sociedad de la información.

10. Contenidos:

a) Nociones generales

- El derecho. Sistemas normativos.
- Derecho público y derecho privado.
- La tecnología y el derecho.

- b) Conceptos básicos de derecho laboral
 - La relación laboral
 - Derechos laborales fundamentales.
 - Despido con causa justa. Despido sin causa justa.
- c) Conceptos básicos de derecho administrativo
 - La administración pública. Principio de legalidad.
 - El acto administrativo.
 - El procedimiento administrativo.
 - Contratación administrativa.
- d) Conceptos básicos de derecho comercial
 - Las personas jurídicas. Organización societaria. La empresa.
 - Los actos de comercio.
 - El contrato comercial.
 - Régimen fiscal de las empresas.
- e) Propiedad intelectual
 - Definición y tipos de protección.
 - Derechos de autor y conexos.
 - Marcas y nombres comerciales.
 - Patentes.
 - Modelos industriales.
 - Competencia desleal.
 - Cláusulas de confidencialidad. Información no divulgada.
- 7. Derecho informático
 - Comercio electrónico.
 - El documento digital.
 - Contratos electrónicos.
 - Régimen jurídico del software.
 - Firma digital.
 - Delitos informáticos.

Propiedad Intelectual y su aplicación práctica

I. Descripción del Curso

Siendo que en nuestras sociedades modernas el conocimiento se ha convertido en uno de los bienes intangibles de mayor relevancia social y económica como motor de desarrollo de los países, el presente curso tiene como propósito reconocer en qué consiste la Propiedad Intelectual como fenómeno global; sus distintas manifestaciones, su evolución en relación con las tecnologías de la información y comunicación (TIC), cuáles son sus formas y plazos de protección y su régimen sancionatorio en Costa Rica.

El curso posee un enfoque interdisciplinario que trata de que estudiantes de cualquier área del conocimiento (informática, educación, bibliotecología, química, física, biología, ciencias agrarias, etc.) identifiquen el valor del conocimiento intelectual que crean en sus distintas áreas disciplinarias, los mecanismos de su protección y explotación, así como el uso legítimo para su vida cotidiana de otros que son propiedad de terceros.

II. Objetivos

Objetivo general:

Reconocer una visión integral de la Propiedad Intelectual a través de conocimientos básicos para su desarrollo y protección, en beneficio personal, de terceros y de la sociedad en general.

Objetivos específicos:

- Describir conceptos fundamentales sobre propiedad Intelectual, sus manifestaciones y características principales, mecanismos y plazos de protección.
- Distinguir la importancia de la Propiedad Intelectual como herramienta que nos permita impulsar la innovación, la difusión y acceso a conocimientos y tecnología.
- Aplicar correctamente las TIC como herramientas indispensables para el desarrollo de sus distintas áreas disciplinarias.
- Analizar el marco regulatorio establecido en Costa Rica sobre la Propiedad Intelectual y su régimen sancionador.

III Contenidos:

1. Propiedad intelectual
2. Derechos de autor y conexos
3. Derechos de autor y su relación con las TIC's
4. Software
5. Patentes de invención
6. Marcas y otros signos distintivos
7. Otras formas de protección.

Aseguramiento de la Calidad de Software

I. Descripción del Curso

El curso de Aseguramiento de la Calidad (*Quality Assurance -QA*) introduce al estudiante en las prácticas, metodologías y herramientas utilizadas para garantizar la calidad del software a lo largo de su ciclo de desarrollo. Se abordan tanto las pruebas manuales como las automatizadas, permitiendo al estudiante comprender cuándo y cómo aplicarlas eficazmente.

A través de actividades prácticas, el estudiante aprenderá a diseñar planes de prueba, ejecutar casos manuales, desarrollar *scripts* automatizados y utilizar herramientas de gestión y automatización, integrándolas en procesos de desarrollo continuo (CI/CD) y entornos ágiles.

II Objetivos

Objetivo general

Planificar, ejecutar y automatizar pruebas de software, aplicando metodologías de aseguramiento de la calidad que garanticen la fiabilidad, funcionalidad y rendimiento de las aplicaciones desarrolladas.

Objetivos específicos

- Comprender los principios del aseguramiento de la calidad y su papel en el ciclo de vida del software.
- Diseñar y ejecutar pruebas manuales basadas en requerimientos y casos de uso.
- Aplicar herramientas modernas de automatización para pruebas *web* y de APIs.
- Integrar las pruebas dentro de entornos ágiles y pipelines de integración continua.
- Analizar resultados y métricas de pruebas para la mejora continua del proceso de desarrollo.

III Contenidos

1. Conceptos de calidad, QA vs QC vs *testing*. Modelos de desarrollo y rol del QA. Tipos y niveles de pruebas. Responsabilidad del encargado de QA.
2. Tipos de prueba (unitarias, integración, regresión, exploratoria, seguridad,

- rendimiento). Pruebas funcionales y no funcionales.
3. Pruebas Manuales. Ciclo de vida de las pruebas (STLC). Diseño de planes y casos de prueba. Reportes y gestión de defectos.
 4. Automatización de Pruebas. Fundamentos de la automatización. Herramientas para la gestión de pruebas y *frameworks*: *Selenium*, *Cypress*, *Playwright*. *Pruebas de APIs* con *Postman* y *REST Assured*.
 5. QA en Integración Continua y DevOps. *Pipelines* de pruebas en Jenkins o *GitHub Actions*. Ejecución automatizada y reportes de calidad. Métricas y mejora continua.

ANEXO C

PROFESORES DE LOS CURSOS DEL BACHILLERATO EN INGENIERÍA EN COMPUTACIÓN DE LA UNIVERSIDAD NACIONAL

ANEXO C

PROFESORES DE LOS CURSOS DEL BACHILLERATO EN INGENIERÍA EN COMPUTACIÓN DE LA UNIVERSIDAD NACIONAL

CURSO

Desarrollo de Software I
Computación y Modelado Matemático I
Infraestructura I
Desarrollo de Software II
Computación y Modelado Matemático II
Infraestructura II
Desarrollo de Software III
Computación y Modelado Matemático III
Infraestructura III
Desarrollo de Software IV
Computación y Modelado Matemático IV
Infraestructura IV
Gestión Datos I
Ingeniería de Software
Computación y Modelado Matemático V
Algoritmia Avanzada
Gestión de Datos II
Fundamentos de Lenguajes de Programación
Inteligencia Artificial I
Ciberseguridad I
Desarrollo para Plataformas
Ingeniería de Datos I
Interfaz Humano-Computadora
Inteligencia Artificial II
Ciberseguridad II
Ingeniería de Datos II
Seminario I
Práctica Profesional Supervisada (PPS)
Seminario II
Fundamentos de Lenguajes de Programación
Inteligencia Artificial I
Ciberseguridad I
Desarrollo para Plataformas
Ingeniería de Datos I
Interfaz Humano-Computadora
Inteligencia Artificial II
Ciberseguridad II

DOCENTE

José Sánchez Salazar
Juan Félix Ávila Herrera
José Pablo Calvo Suarez
Georges Alfaro Salazar
Enrique Vilchez Quesada
Majid Bayani Abbasy
Irene Hernández Ruíz
Johnny Flores Araya
Majid Bayani Abbasy
José Sánchez Salazar
Juan Félix Ávila Herrera
José Pablo Calvo Suarez
Johnny Villalobos Murillo
Rita Cortes Chavarría
Juan de Dios Murillo Morera
Eddy Ramírez Jiménez
Manuel Espinoza Guerrero
Georges Alfaro Salazar
Juan de Dios Murillo Morera
Henry Guevara Guevara
Carolina Gómez Fernández
Carlos Loría Sáenz
Karol Leitón Arrieta
Carlos Loría Sáenz
Henry Guevara Guevara
Johnny Villalobos Murillo
Ariella Quesada Rosales
Katty Vásquez Ávila
Fulvio Lizano Madriz
Georges Alfaro Salazar
Juan de Dios Murillo Morera
Henry Guevara Guevara
Carolina Gómez Fernández
Carlos Loría Sáenz
Karol Leitón Arrieta
Carlos Loría Sáenz
Henry Guevara Guevara

CURSO

Ingeniería de Datos II
Seminario I
Práctica Profesional Supervisada (PPS)
Seminario II
Informática forense
Métodos Numéricos aplicados a la Ingeniería
Modelado Multidimensional e Inteligencia de Negocios
Computación en la Nube Avanzado
Introducción al análisis de Imágenes Digitales
Aseguramiento de la Calidad de Software
Introducción a la Creación de Empresas
Las TIC en el ámbito jurídico de Costa Rica
Propiedad Intelectual y su aplicación práctica

DOCENTE

Johnny Villalobos Murillo
Ariella Quesada Rosales
Katty Vásquez Ávila
Fulvio Lizano Madriz
Henry Guevara Guevara
Enrique Vílchez Quesada

Johnny Villalobos Murillo
Manuel Espinoza Guerrero
Johnny Villalobos Murillo
Fulvio Lizano Madriz
Doris Ramírez Sandí
César Sánchez Badilla
Herschel Argüello Arce

ANEXO D

**PROFESORES DE LOS CURSOS DEL BACHILLERATO EN INGENIERÍA EN
COMPUTACIÓN DE LA UNIVERSIDAD NACIONAL Y SUS
GRADOS ACADÉMICOS**

ANEXO D

PROFESORES DE LOS CURSOS DEL BACHILLERATO EN INGENIERÍA EN COMPUTACIÓN DE LA UNIVERSIDAD NACIONAL Y SUS GRADOS ACADÉMICOS

HERSCHEL ARGÜELLO ARCE

Licenciatura en Derecho, Universidad de Costa Rica. Maestría en Propiedad Intelectual, Universidad Estatal a Distancia.

GEORGES ALFARO SALAZAR

Bachillerato en Computación e Informática, Universidad de Costa Rica. Maestría en Computación e Informática, Universidad de Costa Rica.

JUAN FÉLIX ÁVILA HERRERA

Bachillerato y Licenciatura en Computación e Informática, Universidad de Costa Rica. Maestría en Computación, Tecnológico de Costa Rica.

MAJID BAYANI ABBASY

Maestría en Computación e Informática, Universidad de Costa Rica.

JOSÉ PABLO CALVO SUAREZ

Bachillerato y Licenciatura en Computación e Informática, Universidad de Costa Rica. Maestría en Computación, Tecnológico de Costa Rica.

RITA CORTES CHAVARRÍA

Licenciatura en Informática, Universidad Nacional.

MANUEL ESPINOZA GUERRERO

Licenciatura en Informática, Universidad Nacional.

JOHNNY FLORES ARAYA

Bachillerato en Enseñanza de la Matemática, Universidad de Costa Rica. Bachillerato en Ingeniería Informática, Universidad Estatal a Distancia. Licenciatura en Docencia, Universidad de San José.

CAROLINA GÓMEZ FERNÁNDEZ

Licenciatura en Informática Educativa, Universidad Interamericana de Costa Rica. Maestría en Tecnología Educativa, Instituto Tecnológico y de Estudios Superiores de Monterrey, República Mexicana.

HENRY GUEVARA GUEVARA

Bachillerato en Ingeniería Informática, Universidad Nacional. Maestría en Computación, Tecnológico de Costa Rica. Maestría en Redes y Telemática, Universidad Latina de Costa Rica.

IRENE HERNÁNDEZ RUIZ

Bachillerato en Ingeniería de Sistemas de Información, Universidad de Costa Rica. Maestría en Administración de la Tecnología Informática, Universidad Nacional.

KAROL LEITÓN ARRIETA

Bachillerato en Informática Educativa, Universidad Nacional. Maestría en Tecnología e Innovación Educativa, Universidad Nacional.

FULVIO LIZANO MADRIZ

Bachillerato en Ingeniería Informática, Universidad Latinoamericana de Ciencia y Tecnología. Maestría en Computación, Tecnológico de Costa Rica. Doctorado en Informática, Universidad de Aalborg, Dinamarca.

CARLOS LORÍA SÁENZ

Licenciatura en Ciencias de la Computación, Universidad de Costa Rica. Doctorado en Informática, Universidad de Kaiserslautern, Alemania,

JUAN DE DIOS MURILLO MORERA

Bachillerato en Ingeniería Informática, Universidad Nacional. Maestría en Computación, Tecnológico de Costa Rica. Maestría en Computación e Informática, Universidad de Costa Rica. Doctorado en Computación e Informática, Universidad de Costa Rica.

ARIELLA QUESADA ROSALES

Bachillerato en Ingeniería Informática, Universidad Estatal a Distancia. Maestría en Gestión de la Innovación Tecnológica, Universidad Nacional.

EDDY RAMÍREZ JIMÉNEZ

Bachillerato en Ingeniería en Computación, Tecnológico de Costa Rica. Maestría en Computación, Tecnológico de Costa Rica.

DORIS RAMÍREZ SANDÍ

Bachillerato en Planificación Económica y Social, Maestría en Propiedad Intelectual, Universidad Estatal a Distancia.

CÉSAR SÁNCHEZ BADILLA

Bachillerato en Derecho, Universidad de Costa Rica. Maestría en Derecho de la Informática y las Telecomunicaciones, Universidad de Chile.

JOSÉ SÁNCHEZ SALAZAR

Bachillerato en Ingeniería Informática, Universidad Nacional. Maestría en Redes y Telemática, Universidad Latina de Costa Rica.

KATTY VÁSQUEZ ÁVILA

Bachillerato en Informática Empresarial, Universidad de Costa Rica. Maestría en Administración de la Tecnología Informática, Universidad Nacional.

ENRIQUE VÍLCHEZ QUESADA

Bachillerato en Enseñanza de la Matemática, Universidad de Costa Rica. Maestría en Tecnología e Innovación Educativa, Universidad Nacional. Maestría en Entornos Virtuales de Aprendizaje, Universidad Técnica Nacional.

JOHNNY VILLALOBOS MURILLO

Licenciatura en Informática, Universidad Nacional. Maestría en Computación e Informática, Universidad Nacional.



UCR

TEC

UNA

UNED

UTN
Universidad
Técnica Nacional



/Consejo Nacional de Rectores



www.conare.ac.cr



2519-5700



1.3 km. norte de la Embajada de los Estados Unidos. Pavas, San José, Costa Rica