COLEGIO UNIVERSITARIO ISESA

Diplomado en Desarrollo de Software

Marzo 2020

Nombre de la institución solicitante:
Colegio Universitario ISESA

Nombre del director de la carrera:
Roger Segnini Esquivel, Máster en Ciencias de la Administración

Tipo de Carrera:
Diplomado Parauniversitario

Nombre de la carrera propuesta

Diplomado en Desarrollo de Software

Matrícula probable y sus proyecciones

La matrícula probable proyectada para este plan de estudios en los primeros cuatro años es la siguiente:

Año	Cuatrimestre I	Cuatrimestre II	Cuatrimestre III
2020		**15**	**15**
2021	**15**	**15**	**15**

Descripción de la carrera

La carrera de Diplomado en Desarrollo de Software tiene como objetivo la formación de futuros profesionales en el campo de la informática.

Su enfoque es práctico y profesional, para que los graduados tengan la capacidad de implementar sus altos conocimientos en el mundo laboral.

Con los conocimientos adquiridos, los titulados en desarrollo de software podrán trabajar con tecnologías de internet, tecnologías móviles, tecnologías de almacenamiento y gestión de la información, desarrollo de software y sistemas.

Los estudiantes tienen la oportunidad de gestionar proyectos integrales e interdisciplinarios, lo cual los convierte en profesionales con un alto compromiso ético, con valores necesarios para la innovación para lograr adaptarse a los cambios y transformaciones que exigen las sociedades que se encuentran en constante desarrollo.

El plan de estudios de la carrera provee a los estudiantes de competencias y conocimientos específicos, que darán la respuesta a la demanda profesional que se requiere en la actualidad.

La formación se encuentra relacionada con cuestiones metodológicas, técnicas y tecnológicas prácticas como el desarrollo de aplicaciones móviles, análisis, diseños y construcciones de software, programación de datos, controles de calidad, brindar capacitación a usuarios, creación y aplicación de webs, administración de sistemas, soporte técnico, márquetin digital, entre otros. Además, se brindan otros elementos que permiten el desarrollo de habilidades interpersonales y el trabajo colaborativo, responsabilidad y disciplina en la realización de proyectos en equipo, así como la puesta en práctica de la ética profesional de la carrera.

El plan de estudios consta de seis (6) cuatrimestres. La metodología pedagógica utilizada para el desarrollo del programa combina la guía del profesor con el aprendizaje colaborativo e interactivo que se desarrolla por medio del trabajo en equipo; además de un aprendizaje orientado en la resolución de situaciones reales por medio de la formulación y desarrollo de proyectos.

Los cursos se inician con el desarrollo de conocimientos impartidos por el profesor y se consolidan por medio de tareas, prácticas dirigidas en clase, laboratorios y proyectos.

Al finalizar todos los cursos de la malla curricular, el estudiante podrá realizar su p**ráctica supervisada** como requisito de opción de evaluación final de graduación.

El plan de estudios consta de 24 cursos teórico-prácticos distribuidos en 6 cuatrimestres, de 15 semanas de duración cada uno. Cada cuatrimestre consta de 4 materias, para una carga académica mínima de 12 créditos y una máxima de 15 créditos. La carrera tiene un total de 86 créditos y una duración total de 2 años.

Especificaciones generales de la carrera.

1. Nombre del título	Diplomado en Desarrollo de Software
2. Duración de la carrera	2 años, divididos en seis cuatrimestres con una carga académica total de cuatro materias por cuatrimestre.
3. Requisitos de ingreso	Título de educación diversificada o su preparación equivalente.
4. Total de créditos	86

Modalidad

Este plan de estudios de Diplomado en Desarrollo de software se impartirá en la modalidad presencial.

Perfil profesional

El graduado de la carrera de Diplomado en Desarrollo de software estará capacitado para participar en cualquiera de las fases técnicas de un proyecto de desarrollo de software, en donde podrá asumir diferentes papeles como:

- Capacidad de liderar un equipo de desarrollo.
- Interactuar con clientes para identificar los requisitos software de la aplicación.
- Prestar una especial atención a la gestión de proyectos software.
- Definir arquitecturas software para aplicaciones empresariales que favorezcan su evolución y mantenimiento.
- Desarrollar sistemas con una gran cantidad de datos persistentes en el tiempo, que incluyen unas complejas reglas de negocio, y tienen importantes necesidades de integración con otras aplicaciones software.

Caracterización del Diplomado

El graduado de esta carrera estará capacitado para laborar como:

- Programador de sistemas software.
- Diseñador/integrador de sistemas.
- Consultor de sistemas.
- Especialista en soluciones TIC.
- Diseñador de web.
- Programador de aplicaciones.
- Especialista en mantenimiento software.
- Gestor de información.
- Especialista en integración y pruebas.
- Analista y especificador de sistemas de software.
- Diseñador de sistemas de software sobre diversas plataformas tecnológicas.
- Programador de aplicaciones en lenguajes de alto nivel.
- Programador de componentes en lenguajes de alto nivel y de bajo nivel (como el ensamblador).
- Analista e investigador de nuevas tecnologías, métodos y técnicas de desarrollo de software.

El graduado del Diplomado en Desarrollo de Software se podrá desempeñar en:

- Empresa desarrolladora de software.
- Empresa o institución que aplica tecnologías de la información.
- Empresa que integra tecnologías de la información.
- Empresa propia, consultoría independiente.
- Representación, capacitación y venta de productos de software.

Perfil del Diplomado

A continuación, se presentan los conocimientos, habilidades, destrezas y actitudes que poseerán los graduados del Diplomado en Desarrollo de Software.

Los graduados de esta carrera tendrán, en general, capacidad de:

- Diseñar, desarrollar, seleccionar y evaluar, aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a los principios éticos y a la legislación y normativa vigente.
- Planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social.
- Elaborar el pliego de condiciones técnicas de una instalación informática que cumpla los estándares y normativas vigentes.
- Conocer, administrar y brindar mantenimiento sistemas, servicios y aplicaciones informáticas.
- Conocer y aplicar de los procedimientos matemáticos básicos de las tecnologías informáticas para diseñar soluciones a situaciones planteadas.
- Conocer, diseñar y utilizar de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema.
- Analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
- Conocer, comprender y evaluar la estructura y arquitectura de los computadores, así como los componentes básicos que los conforman.
- Conocer las características, funcionalidades y estructura de los Sistemas Operativos y diseñar e implementar aplicaciones basadas en sus servicios.
- Conocer y aplicar de las características, funcionalidades y estructura de los sistemas distribuidos, las redes de computadores e internet y diseñar e implementar aplicaciones basadas en ellas.
- Conocer y aplicar las características, funcionalidades y estructura de las bases de datos, que permitan su adecuado uso, y el diseño y el análisis e implementación de aplicaciones basadas en ellos.

- Conocer y aplicar las herramientas necesarias para el almacenamiento, procesamiento y acceso a los sistemas de información, incluidos los basados en web.
- Conocer y aplicar los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real.
- Conocer y aplicar los principios, metodologías y ciclos de vida del desarrollo de software.
- Diseñar y evaluar interfaces persona-computador que garanticen la accesibilidad y usabilidad a los sistemas, servicios y aplicaciones informáticas.
- Capacidad para comprender la importancia de la negociación, los hábitos de trabajo efectivos, el liderazgo y las habilidades de comunicación en todos los entornos de desarrollo de software.
- Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.
- Capacidad para realizar un proyecto en el ámbito de la Ingeniería Informática, con énfasis en la tecnología específica del Desarrollo de Software, de naturaleza profesional en el que se sinteticen e integren las competencias adquiridas en las enseñanzas.

Nivel Cognitivo

Los conocimientos en cuanto a elementos técnicos incluirán:

Matemática

- Lógica básica para resolución de problemas.
- Estructuras matemáticas discretas que permiten el modelado de sistemas computacionales y estructuras de información.
- Sistemas y técnicas de razonamiento formal.

Tecnología de la información

- Elementos de la arquitectura de una computadora.
- Organización y servicios de un sistema operativo.

Paradigmas, lenguajes y tecnologías de desarrollo de software

- Programación metódica en los paradigmas orientados a objetos e imperativo/estructurado.
- Dominio intermedio-avanzado de los lenguajes Python y de Java, bajo un enfoque de orientación a objetos y de desarrollo de componentes.
- Desarrollo de aplicaciones web en arquitecturas de múltiples capas.
- Dominio intermedio del lenguaje C++, para la programación de algoritmos y estructuras de datos comunes.
- Dominio intermedio de PHP, HTML y Java Script.
- Conocimiento aplicado de las plataformas de componentes y de acceso a bases de datos vía componentes.
- Conceptos y técnicas de programación en lenguaje ensamblador.

Bases de datos y modelado de datos

- Modelado de datos mediante el enfoque entidad-relación extendido y su traducción sistemática al modelo relacional.
- Normalización de bases de datos relacionales.
- Dominio avanzado del lenguaje SQL para definición y manipulación de bases de datos relacionales.
- Construcción y manipulación de bases de datos SQL y no SQL.
- Disparadores (triggers) y procedimientos almacenados.

Análisis y diseño de sistemas de software

- Principios de educción y especificación de requerimientos.
- Análisis y estructuración de requerimientos mediante casos de uso, historias de usuario.
- Modelado conceptual y orientado a objetos.
- Principios, técnicas y reglas usuales para el diseño de software, particularmente para el diseño orientado a objetos.
- Diseño, especificación, modelado y construcción de arquitecturas de software de aplicación.
- Patrones de diseño de software (detallado y conceptual).
- Algunos marcos (frameworks) de componentes.

Procesos coadyuvantes al desarrollo de software de calidad

- Procesos sistemáticos, incrementales e iterativos de desarrollo de software.
- Elementos de la planificación del trabajo individual y grupal.
- Técnicas de seguimiento y control individual (bitácoras, cuaderno del desarrollador) y grupal (minutas, agendas, informes de avance).
- Valoración de riesgos en proyectos y sus entregables.
- Aseguramiento de la calidad del software: planificación, especificación, técnicas de verificación, técnicas de validación.
- Planificación, identificación y control de la configuración.

Contexto profesional: empresa, comunicación, desarrollo humano, ética

- Principios y criterios éticos para el desempeño laboral.
- Conceptos básicos de empresa y su organización por procesos y funciones.
- Contexto empresarial en que se aplica la tecnología de información.
- Valores sociales, profesionales y las leyes que rigen las sociedades.
- Impacto social y económico de la tecnología de la información.
- Principios y técnicas de comunicación personal y empresarial.
- Factores humanos del trabajo en equipo.
- Inteligencia emocional para el desarrollo individual.

Actitudes y valores.

Además de las destrezas y conocimientos técnicos, los graduados desarrollarán las cualidades siguientes:

- Actitud para el trabajo colaborativo.
- Principios éticos y morales en el cumplimiento del trabajo.
- Disciplina de trabajo sistemático, orientada a planificar el trabajo (individual y colectivo).
- Disposición al cambio y a innovar.
- Uso creativo de recursos para la solución de problemas.
- Autodidacta y autocrítico para la búsqueda continua de la superación.
- Conciencia y cuidado por el ambiente físico y social en que se desenvuelve.

Objetivos de la carrera

Objetivo general

Formar estudiantes especializados en el desarrollo de software incluyendo el diseño, programación, administración y soporte a los sistemas, mediante la aplicación de tecnologías que les faculte en el desenvolvimiento en empresas de software con estándares de calidad mundial.

Objetivos específicos

- Analizar los fundamentos del proceso lógico en el ámbito del desarrollo de software que tengan por objeto la concepción, el desarrollo o la explotación de sistemas, servicios y aplicaciones informáticas.
- Dirigir las actividades objeto de los proyectos del desarrollo de software.
- Desarrollar la accesibilidad, ergonomía, usabilidad y seguridad de los sistemas, servicios y aplicaciones informáticas, así como de la información que gestionan.
- Definir plataformas de hardware y software para el desarrollo y la ejecución de sistemas, servicios y aplicaciones informáticas.
- Desarrollar sistemas o arquitecturas en el ámbito del diseño de software centralizadas o distribuidas integrando hardware, software y redes.
- Conocer las materias básicas y tecnologías, que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones.
- Resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad.
- Conocimientos para la realización de mediciones, cálculos, valoraciones, tasaciones, peritaciones, estudios, informes, planificación de tareas y otros trabajos análogos del desarrollo del software.
- Capacidad para analizar y valorar el impacto social y medioambiental de las soluciones técnicas, comprendiendo la responsabilidad ética y profesional.

- Aplicación de elementos básicos de matemáticas y de gestión de recursos humanos, organización y planificación de proyectos, así como la legislación, regulación y normalización en el ámbito de los proyectos del desarrollo de software.
- Valorar las necesidades de los clientes y especificar los requisitos software para satisfacer estas necesidades.
- Capacitar para la dirección de las actividades objeto de los proyectos del ámbito del desarrollo de software.
- Administrar el sistema y las bases de datos vinculadas, sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales, evaluando los riesgos potenciales asociados que pudieran presentarse.
- Participar en equipos de trabajo, integrando sinérgicamente su labor a los objetivos del equipo y aprovechando sus habilidades para comunicarse y relacionarse armoniosamente con colegas de su profesión al igual que con otras personas.

Especificaciones que identifiquen la carrera

El desarrollo del software es una disciplina que se ocupa de proporcionar un marco adecuado para la gestión de los proyectos software, y para el diseño de aplicaciones empresariales con una arquitectura software que favorezca su mantenimiento y evolución a lo largo del tiempo.

Los conceptos de modelo de proceso, planificación temporal, trabajo en equipo, gestión de riesgos, garantía de calidad, gestión de la configuración, especificación de requisitos software, arquitectura software, modelo de diseño, patrón de diseño, prueba, integración, evolución y mantenimiento, son claves en esta disciplina.

La carrera de Desarrollo de Software es un diplomado orientado específicamente a estudiantes que deseen profundizar y actualizar en metodologías y tecnologías modernas de la construcción de software y la inclusión de este como eje de desarrollo social, empresarial y tecnológico del país y de otras partes del mundo.

También fomenta la capacitación en aptitud, pericia y competencias en lo que se refiere al campo de la programación y desarrollo de software. Emplea el estudio minucioso de los lenguajes propios de la construcción en dicha área, así como también, de la caracterización de los diseños necesarios para la utilización de herramientas a profundidad; se forman técnicos que puedan proponer soluciones tecnológicamente viables, a los cambios difíciles propuestos por el mercado y que se basan específicamente, en su formación fundamental con relación a los lenguajes básicos y avanzados de programación, modelado y desarrollo de software.

Plan de Estudios

A continuación, se presenta el plan de estudios de Diplomado en Desarrollo de Software propuesto:

Plan de Estudios Diplomado en Desarrollo de Software

El plan de estudios incluye 24 cursos teórico-prácticos. En estos cursos teórico-prácticos se trabaja alrededor de núcleos de conocimiento que se imparten de manera magistral y se reafirman mediante laboratorios, prácticas en clase, tareas programadas y proyectos.

La nota mínima para la aprobación de todos los cursos es de 70 puntos en una escala de 0 a 100 puntos, donde cada uno tiene su evaluación de los aprendizajes específica.

El plan de estudios de esta carrera tiene como requisito de opción final de graduación la **Práctica Supervisada**, que corresponde a 160 horas de trabajo en una empresa de desarrollo de software o afín, supervisadas por un delegado del Consejo Superior de Educación.

La práctica supervisada sólo puede ser realizada por los estudiantes que hayan aprobado la totalidad de los cursos del plan de estudios, es decir, después de haber aprobado los 24 cursos.

Código	Nombre de curso y Periodo	Requisitos	Créditos
I CUATRIMESTRE			
DDS 100	Fundamentos de Informática	NT	3
DDS 101	Principios de Programación	NT	3
MAT 102	Matemática General	NT	2
ING 103	Inglés I	NT	
	Total		**12**
II CUATRIMESTRE			
ING 200	Inglés II	ING 103	3
EP 201	Ética Profesional	NT	3
DDS 202	Programación I	DDS 100- DDS 101	4
DDS 203	Principios de Bases de Datos	DDS 100- DDS 101	4
	Total		**14**
III CUATRIMESTRE			
ING 300	Ingles III	ING 103-ING 200	3
DDS 301	Programación II	DDS 202	4
DDS 302	Proyecto de Software I	DDS 202	4
DDS 303	Fundamentos Eléctricos	MAT 102	4
	Total		**15**
IV CUATRIMESTRE			
AE 400	Autogestión Empresarial	NT	3
DDS 401	Programación III	DDS 202	4
DDS 402	Programación de Base de Datos	DDS203	4
DDS 403	Proyecto de Software II	DDS 101	4
			15
V CUATRIMESTRE			
DDS 500	Fundamentos de Sistemas Operativos	DDS 101-DDS 202 DDS 402	4
DDS 501	Diseño y Construcción de Componentes	DDS 101-DDS 202 DDS 402	4
DDS 502	Programación Orientada a Objetos	DDS 101-DDS 202 DDS 402	4
RHE 503	Relaciones Humanas y Comunicación Eficaz	NT	3
	Total		**15**
VI CUATRIMESTRE			
IVA 600	Investigación aplicada	DDS 402-DDS 502	4
DDS 601	Estructura de Datos	DDS 203-DDS 403	4
DDS 602	Proyecto de Software III	DDS 501	4
DDS 603	Diseño Conceptual de Software	DDS 502	4
	Total		**15**
	Total, General		**86**

Malla Curricular

Código	Requisitos	Nombre de curso y Periodo	Créditos	Hrs Teórica	Hrs estudio	Total Hrs
I Cuatrimestre						
DDS 100	NT	Fundamentos de Informática	3	3	6	9
DDS 101	NT	Principios de Programación	3	3	6	9
MAT 102	NT	Matemática General	3	3	6	9
ING 103	NT	Ingles	3	3	6	9
II Cuatrimestre						
ING 200	ING 103	Ingles II	3	3	6	9
EP 201	NT	Ética Profesional	3	3	6	9
DDS 202	DDS 101	Programación I	4	3	6	12
DDS 203	DDS 101	Principios de Bases de Datos	4	3	6	12
III Cuatrimestre						
ING 300	ING 200	Ingles III	3	3	6	9
DDS 301	DDS 202	Programación II	4	3	6	12
DDS 302	DDS 203	Proyecto de Software I	4	3	6	12
DDS 303	MAT 102	Fundamentos Eléctricos	4	3	6	12
IV Cuatrimestre						
DDS 400	NT	Autogestión Empresarial	3	3	6	9
DDS 401	DDS 301	Programación III	4	3	6	12
DDS 402		Programación de Base de Datos	4	3	6	12
DDS 403	DDS 302	Proyecto de Software II	4	3	6	12
V Cuatrimestre						
DDS 500	DDS 101 DDS 202 DDS 401	Fundamentos de Sistemas Operativos	4	3	9	12
DDS 501	DDS 101 DDS 202 DDS 401	Diseño y Construcción de Componentes	4	3	9	12
DDS 502	DDS 101 DDS 202 DDS 401	Programación orientada a Objetos	4	3	9	12
RHE 503	NT	Relaciones Humanas y Comunicación Eficaz	3	3	6	9
VI Cuatrimestre						
DDS 600	VA 402 DDS 502	Investigación aplicada	4	3	9	12
DDS 601	DDS 203 DDS 403	Estructura de Datos	4	3	9	12
DDS 602	DDS 501	Proyecto de Software III	4	3	9	12
DDS 603	DDS 502	Diseño Conceptual de Software	4	3	9	12
Total			86	72	216	288

Nómina del Personal de la Carrera del Diplomado en Desarrollo de Software

Código	***Nombre curso***	***Nombre del profesor***
	I Cuatrimestre	
DDS 100	Fundamentos de Informática	**Jareth Rivera Pérez**
DDS 101	Principios de Programación	**Luis Mayorga Muñoz**
MAT102	Matemática General	**Adriana May Collado**
ING 103	Inglés	**Lic. Luis Abarca Colato**
	II Cuatrimestre	
ING 200	Inglés II	**Karen Jiménez Gómez**
EP 201	Ética Profesional	**Lic. Estela May Arroyo**
DDS 202	Programación I	**Bach. Wainer Marín Corrales**
DDS 203	Principios de Base de Datos	**Jareth Rivera Pérez**
	III *Cuatrimestre*	
ING 300	Inglés III	**Karen Jiménez Gómez**
DDS 301	Programación II	**Luis Mayorga Muñoz**
DDS 302	Proyecto de Software I	**Mag. Leonardo Valerio Rodríguez**
DDS 303	Fundamentos Eléctricos	**Ing. Luis Diego Montero Azofeifa**
	IV *Cuatrimestre*	
DDS 400	Autogestión Empresarial	**Lic. Estela May Arroyo**
DDS 401	Programación III	**Luis Mayorga Muñoz**
AF 402	Programación de Base de Datos	**Mag. Leonardo Valerio Rodríguez**
DDS 403	Proyecto de Software II	**Mag. Leonardo Valerio Rodríguez**
	V *Cuatrimestre*	
DDS 500	Fundamentos de Sistemas Operativos	**Luis Mayorga Muñoz**
AF 501	Diseño y Construcción de Componentes	**Jareth Rivera Pérez**
DDS 502	Programación Orientada a Objetos	**Bach. Wainer Marín Corrales**
DDS 503	Relaciones Humanas y Comunicación Eficaz	**Dra. Sileny Hernández**
	VI *Cuatrimestre*	
DDS 600	Investigación Aplicada	**Jareth Rivera Pérez**
AF 601	Estructura de Datos	**Luis Mayorga Muñoz**
DDS 602	Proyecto de Software III	**Mag. Leonardo Valerio Rodríguez**
DDS 603	Diseño Conceptual de Software	**Mag. Leonardo Valerio Rodríguez**

Programa	Diplomado en Desarrollo de Software
Curso	DDS-100 - Fundamentos de Informática
Créditos	3
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	I
Periodo	Cuatrimestral
Horas	3

DESCRIPCIÓN DEL CURSO

El curso ofrece al estudiante una descripción general sobre las tecnologías de información y comunicación, llamadas TIC; para que el estudiante aprenda sobre la evolución tecnológica, los componentes físicos y lógicos de las computadoras, su funcionamiento individual y su interrelación en sistemas complejos de hardware, software y comunicaciones.

OBJETIVOS

Objetivo general

- Analizar la evolución y los principales componentes de las tecnologías de información y comunicación, reconociendo los elementos tecnológicos y sus elementos asociados, tanto físicos como lógicos, de Hardware y de Software para valorar la función de estas tecnologías puestas al servicio del ser humano.

Objetivos específicos

- Conocer los conceptos fundamentales sobre los que se sustentan los modelos de las tecnologías de información y comunicación, distinguiendo entre dato, información, comunicación, sistema, hardware, software para emplear los conocimientos en sistemas de información y telecomunicación.
- Conocer información acerca del desarrollo histórico de las tecnologías computacionales, analizando los elementos claves que intervinieron en dicho desarrollo para conocer el contexto de las tecnologías de información y comunicación actuales.

- Identificar los diferentes sistemas numéricos y de codificación presentes en las tecnologías de información y comunicación, utilizando sistemas de numeración y codificación, para representar información.
- Comprender el funcionamiento básico de las computadoras, de las redes de computadoras y de sus componentes, analizando la interrelación entre el hardware y el software que las componen y los conceptos fundamentales de interconexión para asegurar la correcta operación entre todos los elementos que las conforman.
- Aplicar conceptos de bases de datos, identificando los elementos fundamentales para formular consultas y generar reportes sencillos.
- Conocer el papel que desempeñan las tecnologías de información y comunicación, analizando su posicionamiento en el mercado actual para identificar el aprovechamiento que le da el ser humano en su vida personal y profesional.
- Determinar el impacto social y económico de las tecnologías de información y comunicación, evaluando los diferentes grupos de usuarios, en especial en las personas con discapacidad, que los utilizan para descubrir sus beneficios en favor del desarrollo humano.
- Aplicar técnicas de indagación, identificando las tendencias en materia de tecnologías de información y comunicación, respetando los derechos de autor, para conocer acerca de nuevas tecnologías.

CONTENIDOS

Tema 1. Introducción a las tecnologías de información y comunicación

- Programación - computación - comunicación.
- La revolución de las computadoras en relación con la revolución industrial.
- Conceptos básicos:
    - Dato – Información – Comunicación.
    - Sistemas de Información.
    - Tecnología de la Información.
    - Abstracción.
    - Software.
    - Hardware.
    - Tecnología de telecomunicaciones.

- Usos de las tecnologías de información y comunicación.
- Interacción de las tecnologías de información y comunicación con comunidades de usuarios.

Tema 2. Evolución de las computadoras y de las telecomunicaciones

- Personajes y hechos importantes.
- Generaciones de las computadoras.
- Computadoras actuales.
- Tecnologías para las telecomunicaciones.
- Tecnologías digitales y convergencia del cómputo, comunicaciones, almacenamiento y multimedios.
- Tecnologías digitales para personas con discapacidad.

Tema 3. Organización y funcionamiento de las computadoras digitales

- Concepto de computadora digital.
- Sistema de cómputo y sus subsistemas.
    - Entrada.
    - Salida.
    - Almacenamiento secundario.
    - Procesamiento.
- Otros componentes.
- Características del procesador.
- Funcionamiento de la computadora.
- Relación entre los diferentes componentes.

Tema 4. Representación de datos y sistemas de numeración

- Bit y Byte.
- Códigos simbólicos.
- Unidades de medidas.
- Sistemas numéricos.
    - Sistema binario.
    - Representación de números negativos.
    - Otros sistemas de numeración posicional: decimal, hexadecimal, octal.
- Conversiones entre sistemas de numeración y operaciones básicas.
- Notación de punto flotante.

Tema 5. Conceptos básicos de sistemas operativos y máquinas virtuales

- Importancia de los sistemas operativos.
- Objetivos fundamentales y funciones de los sistemas operativos.
- El sistema operativo en el sistema de cómputo.
- Máquina virtual (abstracta o extendida).
- Utilidades del sistema operativo.
- Evolución de los sistemas operativos.

Tema 6. Clasificación del software

- Importancia.
- Clasificación general.
    - Software de Aplicación
    - Software del Sistema

Tema 7. Programación de computadoras

- Concepto de programa.
- Lenguajes de alto nivel y lenguajes de bajo nivel.
- Máquinas multinivel.
- Traducción y ejecución de programas.
- Lenguajes de programación.

Tema 8. Introducción a las redes de computadoras

- Concepto de red.
- Componentes de software y de hardware.
- Clasificación de las redes.
- Redes alámbricas e inalámbricas.
- Internet y sus aplicaciones básicas.

Tema 9. Introducción a las bases de datos

- Conceptos fundamentales de bases de datos.
- Sistemas de archivos versus sistemas de bases de datos.
- Bases de datos relacionales.
- Lenguaje de consultas.

Tema 10. Tecnologías de información y la interacción humanos- tecnología

- Usos empresariales de las tecnologías de información y comunicación.

- Impactos sociales y económicos de la tecnología
- Administración de la información.
- Desarrollo de software y sistemas de información.
- Administración de redes y plataformas tecnológicas.
- Administración de servicios habilitados por tecnologías de información y comunicación.
- Sub disciplinas en las tecnologías de la información
- Aplicaciones de las tecnologías de información y comunicación.

METODOLOGÍA

- El trabajo en clase incluye exposiciones magistrales en las cuales los temas son expuestos por el profesor, el realiza una retroalimentación con los estudiantes y tiene la posibilidad de aclarar dudas y fomentar la participación de los estudiantes. De la mano va el trabajo en clase también incluye prácticas individuales, en donde los estudiantes pueden reafirmar y aplicar el conocimiento adquirido.

  El profesor y los estudiantes pueden usar en sus lecciones recursos computacionales para ejemplificar algunos temas del curso.

ESTRATEGIAS DE APRENDIZAJE

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra-clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

RECURSOS DIDÁCTICOS

- Proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras con el software debidamente instalado y configurado para realizar las prácticas en clase, uso de diferentes partes de computadoras para ejemplificar el tema de organización y funcionamiento de las computadoras digitales. Para el tema de redes, se utilizan diversos componentes de software y hardware de red, que facilitan que los estudiantes aprendan a identificarlos y entiendan mejor su funcionamiento; libros y manuales que están en la biblioteca para consulta

de los estudiantes, y adicionalmente los estudiantes utilizan la Web para sus investigaciones.

EVALUACIÓN

Concepto	Ponderación
Prácticas en clase (laboratorio)	10%
Tareas extra-clase	10%
Trabajo de Investigación	20%
Examen 1	25%
Examen 2	25%

BIBLIOGRAFÍA

Referencias principales:

- Brookshear, J. G. (2018). Computer Science: An Overview. (13th Edition). Boston, Estados Unidos: Addison-Wesley.

Texto clásico:

- Norton, P. (2007). Introducción a la Computación. México: McGraw-Hill Interamericana.
- Suárez y Alonso, Ramón Carlos (2007). Tecnologías de la información y la comunicación: Introducción a los sistemas de información y telecomunicación. España. Editorial: Ideas Propias

Programa	Diplomado en Desarrollo de Software
Curso	DDS-101 Principios de Programación
Créditos	3
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	I
Periodo	Cuatrimestral
Horas	3

DESCRIPCIÓN DEL CURSO

En este curso los estudiantes elaboran diagramas de flujo como herramientas de diseño y construcción de soluciones, lo que les permite ir aumentando la complejidad de los problemas a resolver.

Aprenderán a crear algoritmos, hacer uso de estructuras condicionales, secuenciales y otras abstracciones procedimentales para la creación de aplicaciones de software.

Los estudiantes aprenden los elementos fundamentales que componen el pensamiento lógico que estructura la programación, para posteriormente aterrizar estos conceptos en ejercicios prácticos

OBJETIVO GENERAL

- Capacitar al estudiante en la solución de problemas por medio de la aplicación de algoritmos y programas de software, expresados en estructuras secuenciales, estructuras condicionales, estructuras iterativas, arreglos y abstracción procedimental, usando diagramas de flujo y un lenguaje de programación, para elaborar pequeñas aplicaciones de software de calidad desde su computador.

OBJETIVOS ESPECÍFICOS

- Aplicar una metodología para la solución de problemas usando conceptos de algoritmia y programación., con la finalidad de que sea más claro y fácil la elaboración de los programas.

- Desarrollar algoritmos que utilicen estructuras de decisión e iterativas para ser incorporadas como métodos en las soluciones de problemas
- Desarrollar algoritmos que utilicen estructuras de decisión e iterativas para ser incorporadas como métodos en las soluciones de problemas.
- Usar un lenguaje de programación para implementar los algoritmos planteados en la solución de problemas.
- Diseñar e implementar interfaces gráficas de usuario sencillas, usando un lenguaje de programación.
- Formular programas de software utilizando el paradigma de programación estructurada y un lenguaje de programación, a fin de que los algoritmos elaborados sean implementados y probados en la computadora.
- Utilizar buenas prácticas en la elaboración de programas de software tales como el uso de estándares, buen estilo de codificación y técnicas básicas, para el aseguramiento de la calidad de los programas de software.

CONTENIDOS

Tema 1. CONCEPTOS BÁSICOS DE COMPUTACIÓN

- Partes del computador
- Concepto de algoritmo y programa
- Conceptos de editar, compilar, interpretar, encadenar y ejecutar

Tema 2. MODELADO Y SOLUCIÓN ALGORÍTMICA DE PROBLEMAS

- Estructuras Secuenciales
- Estructuras Condicionales
- Estructuras Iterativas
- Valores, Tipos, Operaciones y Expresiones

Tema 3. EL AMBIENTE DE PROGRAMACIÓN

- Introducción a la historia del lenguaje de programación
- Estructura del lenguaje de programación

- Instalación del compilador y del editor del lenguaje de programación

Tema 4. FLUJO DE CONTROL EN EL LENGUAJE DE PROGRAMACIÓN

- Flujo de control secuencial
- Flujo de control condicional
- Flujo de control iterativo
- Análisis básico de programas

Tema 5. CALIDAD DE LA PROGRAMACIÓN

- Estándar de codificación propuesto para el lenguaje de programación
- Estilo de programación
- Listas de revisión
- Revisión conjunta de algoritmos y programas
- Formulación de casos de prueba
- Proceso básico de prueba de programas
- Depuración básica de programas

Tema 6. ABSTRACCIONES

- Abstracción procedimental
- Arreglos
- Ordenamiento y búsquedas en arreglos

METODOLOGÍA

El trabajo en clase incluye exposiciones magistrales en las cuales los temas son expuestos por el profesor, el realiza una retroalimentación con los estudiantes y tiene la posibilidad de aclarar dudas y fomentar la participación de los estudiantes. De la mano va el trabajo en clase también incluye prácticas individuales, en donde los estudiantes pueden reafirmar y aplicar el conocimiento adquirido.

El profesor y los estudiantes pueden usar en sus lecciones recursos computacionales para ejemplificar algunos temas del curso.

ESTRATEGIAS DE APRENDIZAJE

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra-clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

RECURSOS DIDÁCTICOS

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

EVALUACIÓN

Concepto	**Ponderación**
Asistencia	10%
Prácticas y laboratorios	20%
Proyecto final	25%
Examen 1	25%
Examen 2	25%
Total	**100%**

BIBLIOGRAFÍA

Bibliografía obligatoria

- Lutz, M. (2013). Learning Python: Powerful Object-Oriented Programming (5ª Edición). O' Reilly.
- Farrell, J. (2014). Programming Logic and Design, Comprehensive (8 ª Edición). Estados Unidos de América: Cengage Learning.

Bibliografía clásica

- Farrel, J. (2000). Iniciación a la Programación Lógica y Diseño. Paraninfo.

Programa	
Curso	MAT-102 Matemática General
Créditos	2
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	I
Periodo	Cuatrimestral
Horas	3

DESCRIPCIÓN DEL CURSO

Las bases matemáticas son fundamentales para la formación de cualquier diplomado, y su aprendizaje contribuye a la resolución de problemas de dicha disciplina. Por tanto, se hace necesaria una nivelación de dichos conocimientos de manera que puedan ser utilizados en la actividad laboral.

En este curso el estudiante adquiere conocimientos de lógica formal elemental y de matemática numérica y simbólica básica, para desarrollar capacidades de razonamiento lógico y cuantitativo, mediante la aplicación de técnicas de razonamiento lógico-matemático y lógico-verbal, así como el estudio de las propiedades fundamentales de los números, el álgebra, las funciones, la geometría y la trigonometría, a fin de plantear y resolver diversos problemas informáticos, cimentar el aprendizaje de cursos superiores y aplicar métodos cuantitativos simbólicos y numéricos.

OBJETIVOS

Objetivo general

- Conocer los conceptos básicos de la aritmética en el conjunto de los números reales, el álgebra, la teoría de funciones y la geometría para la solución de problemas en situaciones reales

Objetivos específicos

- Operar expresiones numéricas que involucran números reales, potencias y radicales.

- Resolver inecuaciones lineales de una incógnita, de grado mayor o igual que dos con una incógnita y las que involucran fracciones racionales., Ecuaciones e inecuaciones.

- Aplicar el concepto de valor absoluto para resolver ecuaciones e inecuaciones que involucran al valor absoluto.

- Calcular áreas y volúmenes de figuras planas y algunos sólidos tales como paralelepípedos, esferas, conos y cilindros.

CONTENIDOS

Tema 1. Aritmética entera y racional

- Números enteros ($\mathbb{Z}$)
- Números racionales ($\mathbb{Q}$)
- Operaciones fundamentales con números enteros y racionales
- Aritmética modular
- Comparación de cantidades
- Cálculo de porcentajes
- Proporciones y promedios

Tema 2. Números reales

- Subconjuntos de los números reales ($\mathbb{R}$)
- Propiedades de la suma y la multiplicación en $\mathbb{R}$
- Orden en $\mathbb{R}$
- Números irracionales y aproximación racional de números reales
- Desigualdad e intervalos
- Valor absoluto y sus propiedades
- Operaciones con números reales

Tema 3. Razonamiento lógico-verbal

- Silogismos
- Comprensión de lectura
- Completar oraciones
- Sinónimos y antónimos

Tema 4. Razonamiento lógico-proposicional

- Conectivos básicos y tablas de verdad
- Formas normales (conjuntiva y disyuntiva)
- Implicación y equivalencia lógica

Tema 5. Conjuntos

- Concepto de conjunto
- Elementos, pertenencia, subconjuntos

- Diagramas de Venn
- Operaciones sobre conjuntos
- Álgebra de conjuntos, dualidad
- Conjunto potencia

Tema 6. Técnicas de conteo

- Principios básicos de conteo
- Funciones: factorial, coeficientes binomiales, triángulo de Pascal
- Permutaciones
- Combinaciones
- Aplicaciones

Tema 7. Razonamiento lógico-matemático

- Problemas aritméticos
- Problemas numéricos: números. Balanzas, enigmas, análisis de proposiciones
- Problemas de recuento sistemático
- Problemas de razonamiento inductivo

Tema 8. Geometría

- Perímetros y áreas de figuras geométricas
- Propiedades de los triángulos (teorema de Pitágoras y otros teoremas)
- Propiedades de rectas paralelas y perpendiculares
- Áreas de figuras planas y de sólidos
- Volúmenes de sólidos
- Vistazo a la geometría analítica
- Aplicaciones

Tema 9. Trigonometría

- Razones trigonométricas, triángulos especiales
- Circunferencia trigonométrica
- Medidas de ángulos en grados y radianes
- Gráfica de las funciones trigonométricas
- Identidades trigonométricas, resolución de triángulos
- Aplicaciones

Tema 10. Expresiones algebraicas

- Definiciones básicas
- Operaciones algebraicas: suma, resta, multiplicación (productos notables), división algebraica y división sintética
- Factorización de polinomios en $\mathbb{Q}$
- Simplificación de fracciones algebraicas racionales
- Graficación en el plano cartesiano

Tema 11. Ecuaciones e inecuaciones

- Ecuaciones lineales
- Ecuaciones cuadráticas
- Ecuaciones con valor absoluto, con un radical, racionales y polinomiales

- Inecuaciones lineales, cuadráticas, con valor absoluto, con un radical, racionales y polinomiales
- Aplicaciones

Tema 12. Funciones

- Conceptos básicos: dominio, ámbito, imágenes, pre-imágenes, intersección con los ejes, intervalos de monotonía, signos
- Análisis de gráficas de funciones
- Gráfica estándar: identidad, cuadrática, cúbica, raíz cuadrada, valor absoluto, exponencial, logarítmica
- Máximo dominio
- Composición de funciones
- Función lineal: pendiente, puntos de intersección con los ejes, rectas paralelas, perpendiculares
- Clasificación de funciones: inyectivas, sobreyectivas, biyectivas
- Aplicaciones

Tema 13. Expresiones exponenciales y logarítmicas

- Expresiones exponenciales
- Expresiones logarítmicas comunes y naturales
- Propiedades de los logaritmos
- Cambio de base
- Ecuaciones exponenciales
- Ecuaciones logarítmicas
- Aplicaciones

Tema 14. Sucesiones y series numéricas

- Formación de términos de acuerdo con una ley
- Acumulación y sumatorias
- Aplicaciones

METODOLOGÍA DE ENSEÑANZA

Este curso se basa en la resolución de problemas y la creación de modelos lógico-matemáticos sencillos, para contribuir a desarrollar en los estudiantes el razonamiento formal y cuantitativo. El profesor ilustra los conceptos gráficamente, a efectos de estimular la comprensión, aprovechando la tecnología computacional para mostrar las propiedades geométricas, algebraicas, trigonométricas y computacionales de las expresiones, las funciones, las series y las matrices.

ESTRATEGIAS DE APRENDIZAJE

Este curso se abre espacios al desarrollo de la imaginación y el pensamiento del estudiante, fundamentalmente al vincular conceptos formales con sus aplicaciones.

Los retos planteados estimulan el desarrollo de habilidades de pensamiento, razonamiento lógico-matemático y estrategias de solución de problemas.

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra-clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

RECURSOS DIDÁCTICOS

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

EVALUACIÓN

Concepto	**Ponderación**
Prácticas en clase	25%
Asignaciones y Tareas	20%
Mini-proyectos	20%
Investigación	15%
Examen	20%
Total	**100%**

BIBLIOGRAFÍA

- Murillo, M. (2015). *Introducción a la matemática discreta, 4ª ed*. Cartago, Costa Rica: Editorial Tecnológica de Costa Rica
- Timmons, D., Johnson, C. & McCook, S. (2013). *Fundamentals of Algebraic Modeling, 6th ed*. Belmont, Estados Unidos: Brooks Cole.

Bibliografía complementaria

- Camacho, L. (2015). *Introducción a la lógica*. Cartago, Costa Rica: Libro Universitario Regional (Editorial Tecnológica de Costa Rica).
- Dawkins, P. (2018). *Algebra and trigonometry*. Beaumont, Estados Unidos: Lamar University Department of Mathematics. Disponible libremente en: http://tutorial.math.lamar.edu/getfile.aspx?file=B,12,N
- Dawkins, P. (2017). *College algebra*. Beaumont, Estados Unidos: Lamar University Department of Mathematics. Disponible libremente en: http://tutorial.math.lamar.edu/Classes/Alg/Alg.aspx
- Stewart, J., Redlin, L. & Watson, S. (2017). *Precálculo. Matemáticas para el cálculo. Septima Edición*. México D.F., México: Cengage Learning Editores.

Textos clásicos

- Copi, I. & Cohen, C. (2013). *Introducción a la lógica, 2da. ed.* México D.F., México: Editorial Limusa.
- Lipschutz, S. (2007). *Schaum's outline of discrete mathematics*. Nueva York, Estados Unidos: McGraw-Hill.

Programa	Diplomado en Desarrollo de Software
Curso	ING-103 Inglés I
Créditos	3
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	I
Periodo	Cuatrimestral
Horas	3

DESCRIPCIÓN DEL CURSO

Este curso provee al estudiante de la formación básica para iniciarse en el conocimiento de vocabulario y estructuras gramaticales.

Se provee preparación en cuanto a destrezas básicas para la comprensión de lectura. Se proporciona también formación introductoria para escribir comunicaciones básicas

OBJETIVOS

Objetivo General

- Desarrollar en el estudiante habili-dades y es-tra-tegias de comprensión de literatura escrita en el idioma in-glés, que le permitan la capacidad para utilizar el vocabulario básico en contexto

Objetivos específicos

- Determinar en una oración, la función y posición de las diversas estructuras gramaticales.
- Interpretar en un contexto determinado, el significa-do de un sustantivo núcleo.
- Aplicar las reglas de los determinantes en el aná-lisis de o-raciones.
- Traducir al español oraciones en inglés.
- Interpretar oraciones insertadas y de reporte del mensa-je.

- Aplicar nuevas estructuras gramaticales y vocabulario básico para expresarse, en forma oral en situaciones cotidianas.

CONTENIDOS

1.- Las estructuras gramaticales:

- Sustantivos: singular, plural, como su-jetos y p-r-e-d-i-c-a-d-os, com-puestos, plurales irr-egulares, conta-b-les, no con-ta-bles, con "e-no-ugh", "much" y "many".
- Pronombres: En posición de sujeto, demos-trativos, refle-xi-vos, obsesivos, con "some", cuantificadores "a lot of", "mu-ch", “many", "a little", "a few", "several".
- Pronombres: En posición de sujeto, demos-trativos, re-fle-- xivos, obsesivos, con "some" y "any".
- Determinantes: “an” y "the", adjetivos posesivos, núme-ros cuan-tificadores "some", "any" y "no", cuantificado-res "a lot of", "much", "many", "a little", "a few", “se-ve-ral".
- Adjetivos: De lugar, nacionalidad, compa-rativos, su-per-lati-vos, entre otros.
- Preposiciones: De lugar, de tiempo, de posición, en-tre otros.
- Verbos: Presente de "be", impera-tivo, presente simple, pre-sente progresivo, modales, verbos "to like" y "ha-ve to", el futuro, el pasado perfecto, la voz pasiva.
- Adverbios: Principales adverbios de lugar, tiempo y fre-cuen-cia con la terminación "by” y las correspondientes ex-cep-ciones, comparativos y superlativos, frases con "f-or", "ago", "since".
- Correctores: "and", "but", conceptos expre-sados por los correc-tores (tiempo, concesión).

2. Las oraciones:

- Oraciones con "let's", "there is", "here are", entre otras, para todos los tiempos verbales intro-du-ci-dos, afirmativas y negati-vas.
- Respuestas cortas afirmativas y negativas en todos los tiem-pos y estudiados.
- Oraciones exclamativas con "How", "What", entre otras, oracio-nes complejas con el presente real e irreal con "if", cláusulas rela-tivas, oraciones con "Wi-sh", ora-ciones con "Would", en el pasado irreal.
- Oraciones insertadas y de reporte del mensaje.

3. Comprensión y análisis de lectura:

- Traducción al español. Identificación de, idea central y secundaria.

4. Construcción y pronunciación de oraciones en forma oral sobre situaciones cotidianas que le permitan desarrollarse en el ámbito laboral.

METODOLOGÍA:

En concordancia con lo anterior, el docente se concibe como un facilitador, colaborador y consejero de los procesos de aprendizaje del estudiante, por lo que asume una gran responsabilidad por la calidad de la enseñanza, dentro de un proceso de carácter formativo y permanente, en el que procura armonizar los tres elementos del desarrollo humano: el factor cognitivo, el factor socio-afectivo y el factor psico-motriz.

Se utilizarán técnicas participativas para lograr que el alumno se comunique adecuadamente en forma oral y escrita.

RECURSOS EDUCATIVOS:

Para desarrollar este curso se requiere algunos capítulos o secciones de los libros mencionados, de acuerdo con el ordenamiento de la temática anotada. El contenido servirá para trabajar en clase o para realizar prácticas en el hogar.

En cuanto al equipo, se requieren dispositivos de tipo audio-visual, tales como la videograbadora, cámaras, DVD y el televisor, además de la radio grabadora.

EVALUACIÓN:

Concepto	**Ponderación**
2 evaluaciones parciales	40%
1 evaluación final	30%
Prácticas	10%
Quices y asignaciones	20%
Total	**100%**

BIBLIOGRAFÍA CLÁSICA

- Remacha, S. (2008). Infotech: English for Computer Users, Fourth Edition - Student's Book. Cambridge, Reino Unido: Cambridge University Press.
- Glendinning, E. (2007). Oxford English for Careers: Technology 1: Student's Book. Oxford, Reino Unido: Oxford University Press
- Marco, E.; Remacha, S. (2007). Professional English in Use: ICT. Cambridge, Reino Unido: Cambridge University Press.
- Marco, E.; Remacha, S. (2008). English for Computer users. Student´s book. Cambridge, Reino Unido: Cambridge University Press.

Programa	Diplomado en Desarrollo de Software
Curso	ING-200 Ingles II
Créditos	3
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	II
Periodo	Cuatrimestral
Horas	3

Descripción del curso

El curso tiene como finalidad fortalecer el desarrollo de habilidades para comunicarse adecuadamente en forma oral y escrita, mediante la preparación y presentación de reportes orales y escritos; dirigidos y enfocados en el área de los seguros, así como mejorar la fluidez en la expresión de opiniones y puntos de vista personales, tanto guiados como libres.

OBJETIVOS

Objetivo General

Incrementar el conocimiento del idioma inglés, fortaleciendo las habilidades de lectura y escritura en ese idioma dentro de los términos utilizados a nivel internacional en el campo de los seguros.

Objetivos Específicos

- Mejorar la riqueza y fluidez en el habla del idioma inglés.
- Fortalecer el uso de vocabulario técnico para comunicarse adecuadamente con el consumidor, intermediarios y compañías de seguros y reaseguros.
- Desarrollar habilidades de síntesis en la preparación de reportes orales y escritos a nivel técnico.

Contenidos:

1. Pasos para elaborar un reporte de trabajo escrito.
2. Elaboración de reportes sobre asuntos relacionados con la cotidianidad:
3. Elaboración de mensajes, invitaciones, actividades a realizar.
4. Exposiciones orales con referencia a**:** Descripción e información en términos relacionados a seguros. Descripción de las diferentes acciones que realizan los actores en la actividad aseguradora. Descripción de procedimientos. Expresar agradecimientos y despedidas. Dar y recibir mensajes.

Metodología

Los contenidos proporcionan apoyo para el desarrollo de habilidades receptivas y de producción, de ahí que las actividades se enfoquen en lograr la fluidez conversacional; por tanto, será indispensable la participación activa del alumno y el profesor. Entre las actividades que se realizarán, se pueden citar: group work, pair work, role play, brainstorming, warm up activities.

Recursos

Para desarrollar este curso se requiere algunos capítulos o secciones de los libros mencionados, de acuerdo con el ordenamiento de la temática anotada. El contenido servirá para trabajar en clase o para realizar prácticas en el hogar.

En cuanto al equipo, se requieren dispositivos de tipo audio-visual, tales como la videograbadora, cámaras, DVD y el televisor, además de la radio grabadora.

Evaluación

Concepto	**Ponderación**
2 examenes parciales	40%
1 evaluación final	30%
Prácticas	10%
Quices y asignaciones	20%
Total	**100%**

Bibliografía clásica

- Alcaraz Enrique, **Diccionario de Términos de Seguros**, Editorial Ariel, 2013.
- Anderson, N. (2013). **Active. Skills for Reading: Book 1.** Singapore: Thomson.
- Del Pozo, Juan Ramón. **Diccionario bilingüe de expresiones y términos de seguros: inglés-español, español-inglés.** FUNDACIÓN MAPFRE, Instituto de Ciencias del Seguro. 2012. España
- **Diccionario Español-inglés, inglés-Español**. McGrraw-Hill, USA, 2012.
- **Fundamentals of English Grammar**, 3Edition Betty Schrampfer Azar. Editorial Logman. 2014

Programa	Diplomado en Desarrollo de Software
Curso	EP-201 Ética Profesional
Créditos	3
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	II
Periodo	Cuatrimestral
Hora	3

Descripción del curso

Actualmente nuestro país ha ido adelantando en muchos campos del desarrollo tanto tecnológico, como científico. De ahí, que el presente curso contribuirá con la formación integral de los futuros graduados, al integrar la técnica, los conocimientos, la moral y el desarrollo social de los individuos.

Objetivo general

Propiciar en los estudiantes una cultura, tecnológica y humanística respecto a su carrera y al medio ambiente que los rodea.

Objetivos específicos

- Conocer los conceptos filosóficos que están relacionados con el concepto de moralidad y con el orden social.
- Identificar las principales corrientes del pensamiento que pretendan explicar el problema ético.
- Describir algunos de las principales características del problema moral y consecuente, de ética, como respuestas racionales de tal problema.
- Reconocer la importancia del conocimiento de los problemas éticos, con base para la aplicación específica del ejercicio de la profesión.

CONTENIDOS

Tema 1: PENSAMIENTO CRÍTICO

- El pensamiento crítico
- Ética en los negocios
- Metodologías para la toma de decisiones éticas
- Trabajo académico correcto
- Integridad versus fraude académico

Tema 2: LA INFORMÁTICA COMO PROFESIÓN

- Concepto de profesión
- Desarrollo histórico de la profesión Informática y sus sub-disciplinas
- El papel de las asociaciones profesionales
- Los cuerpos de conocimientos en Informática y referentes internacionales
- Acreditación, certificación y licenciamiento: esquemas
- Estándares técnicos y profesionales en las disciplinas informáticas
- Perfiles profesionales y ocupacionales en Informática
- Desarrollo de una carrera profesional informática
- Actualización y desarrollo profesional
- Planificación de la carrera profesional
- Estado de madurez profesional de las disciplinas informática

Tema 3: ÉTICA PROFESIONAL INFORMÁTICA

- Conceptos y marco histórico de la Ética
- Cumplimiento de responsabilidades éticas y profesionales
- Valores éticos dentro de la Informática
- Códigos de ética, conducta y práctica relacionados con la Informática
- Problemas éticos del profesional en Informática

Tema 4: ASPECTOS LEGALES DE LA INFORMÁTICA

- Conceptos generales
- Propiedad intelectual
- Licenciamiento de software
- Seguridad y privacidad de la información
- Usos inadecuados de la Informática
- Legislación en el comercio electrónico (‘*e-commerce*’, en Inglés)
- Tipos de contratos

TEMA 5: EL TRABAJO PROFESIONAL EN INFORMÁTICA

- Naturaleza del trabajo
- Perfiles laborales en Informática
- Estándares de calidad y responsabilidades según el perfil
- Preparación del currículum vitae y entrevistas de trabajo
- Hábitos de la gente eficaz
- El trabajo como emprendedor
- El trabajo como agente libre ('*freelance'*, en Inglés)

Tema 6: RELACIONES LABORALES E INFORMÁTICA

- Elementos esenciales de las relaciones laborales y del Derecho Laboral
- Contratos laborales Contratos por servicios profesionales Derechos derivados
- Acuerdos de no divulgación ('*non-disclosure agreements*', en Inglés)
- Remuneración
- Equidad de género. El acoso y sus variedades
- Teletrabajo
- Seguridad e higiene laboral

METODOLOGÍA

- Con el propósito de estimular que los estudiantes generen criterio en torno a las temáticas tratadas y puedan establecer las relaciones pertinentes entre lo tecnológico y lo social, se utilizarán métodos didácticos entre los que destaca la discusión fundamentada como medio para reforzar el aprendizaje, la argumentación y la reflexión.
- Se emplearán exposiciones magistrales, análisis de casos y discusión de temas utilizando diferentes técnicas como mesas redondas, debates y charlas de especialistas invitados. También se asignan lecturas e indagaciones previas a la discusión, ya sea en pequeños grupos o como trabajo individual.
- Los estudiantes escribirán ensayos con análisis de casos y preparan exposiciones en que podrán interactuar con sus compañeros y el profesor, confrontando sus puntos de vista y defendiendo sus argumentos.

ESTRATEGIAS DE APRENDIZAJE

Las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso son:

- Lecturas: Uno de los recursos que más se utilizará es la lectura y el análisis de textos sobre temas contemporáneos, obtenidos de libros especializados y publicaciones periódicas, así como de revistas y diarios.
- Ensayos: La escritura de ensayos es una importante herramienta para la exploración de ideas, el desarrollo de argumentos y la formación de opinión. Los ensayos permitirán a los estudiantes investigar, analizar, contrastar y profundizar en temas de interés ético, profesional y social, manifestando su posición. Se considerarán, entre otros aspectos: apego al tema abordado (planteamiento y debate), búsqueda de información, claridad de expresión de las ideas (capacidad explicativa y creatividad), sustento argumental, organización del escrito, y corrección ortográfica, gramatical y de redacción.
- Discusiones, mesas redondas y debates: Son actividades en que los estudiantes, con la conducción del profesor, realizan su participación en clase discutiendo los análisis de casos que ellos han estudiado en la literatura consultada o desarrollada en sus ensayos.
- Carteles: Los estudiantes deberán presentar carteles sobre temas estudiados en el curso y en cada clase deberán traer un tema noticioso para discusión en procura de fomentar el análisis crítico y el debate de ideas en torno a un tema específico, siempre dentro de un marco profesional.
- Exposiciones: Son actividades en que los estudiantes presentan formalmente al grupo, para discusión, el trabajo desarrollado en uno de sus ensayos esto con el fin de generar capacidad expositiva y defensa de ideas dentro de un ambiente ético y profesional.

RECURSOS DIDÁCTICOS

Recursos

Para desarrollar este curso se requiere algunos capítulos o secciones de los libros mencionados, de acuerdo con el ordenamiento de la temática anotada. El contenido servirá para trabajar en clase o para realizar prácticas en el hogar.

En cuanto al equipo, se requieren dispositivos de tipo audio-visual, tales como la videograbadora, cámaras, DVD y el televisor, además de la radio grabadora.

Evaluación

Concepto	**Ponderación**
2 ensayos (15% c/u)	30%
Participación: Discusiones, mesas redondas y debates	30%
Conversatorio	15%
Exposición	25%
Total	**100%**

BIBLIOGRAFÍA

Referencias principales:

- Baase, S. (2017). *A Gift of Fire: Social, Legal, and Ethical Issues for Computing and the Internet, 5th Edition*. Upper Saddle River, Estados Unidos: Prentice Hall.
- Colegio de Profesionales en Informática y Computación de Costa Rica (2013). *Código de ética profesional*. San José, Costa Rica: CPIC.

Referencias complementarias:

- Association for Computing Machinery (2018). *ACM Code of Ethics and Professional Conduct*. Nueva York, Estados Unidos: Association for Computing Machinery. https://ethics.acm.org/ (Consultado el 24 de septiembre del 2019)
- British Computer Society (2015). *Code of Conduct for BCS Members*. Londres, Reino Unido: The British Computer Society.

Programa	Desarrollo de Software
Curso	DDS-202 Programación I
Créditos	4
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	II
Periodo	Cuatrimestral
Requisitos	DDS-101 Principios de Programación
Horas	3

DESCRIPCIÓN DEL CURSO

El estudio de los paradigmas de programación orientados a objetos permite al estudiante resolver problemas más complejos. La orientación a objetos permite programar de una manera similar a lo que sucede en la vida real. En este curso los estudiantes experimentarán el uso de ambientes de programación orientado a objetos, utilizarán diagramas de flujo como herramienta de diseño y construcción de las soluciones, a fin de establecer las bases para la resolución de problemas más complejos mediante la programación.

OBJETIVOS

Objetivo general

- Resolver problemas medianos mediante la abstracción de programación orientada a objetos, usando diagramas de flujo y un lenguaje de programación, para elaborar pequeñas aplicaciones de software de calidad.

Objetivos específicos

- Conocer el ambiente de programación de un lenguaje orientado a objetos para poder hacer uso de las diferentes estructuras de control y otros elementos necesarios para la elaboración de aplicaciones de software.

- Elaborar programas en un lenguaje de programación orientado a objetos aplicando fundamentos sistemáticos de resolución de problema, usando los conceptos de abstracciones y con división en capas

CONTENIDOS

Tema 1. EL AMBIENTE DE PROGRAMACIÓN

- Introducción a la historia del lenguaje de programación
- Estructura del lenguaje de programación
- Instalación del editor del lenguaje de programación

Tema 2. FLUJO DE CONTROL EN EL LENGUAJE DE PROGRAMACIÓN

- Flujo de control secuencial
- Flujo de control condicional
- Flujo de control iterativo

Tema 3. ABSTRACCIONES

- Matrices
- Separación del programa en capas
- Manejo de hileras
- Manejo de colecciones

Tema 4. INTRODUCCIÓN A LA ORIENTACIÓN A OBJETOS

- Transición desde el paradigma procedimental.
- Conceptos del paradigma de programación orientado a objetos

Tema 5. DESARROLLO ORIENTADO A OBJETOS

- Abstracción.
- Encapsulamiento.
- Modularidad.

METODOLOGÍA

El trabajo en clase incluye exposiciones magistrales en las cuales los temas son expuestos por el profesor, el realiza una retroalimentación con los estudiantes y tiene la posibilidad de aclarar dudas y fomentar la participación de los estudiantes. De la mano va el trabajo en clase también incluye prácticas individuales, en donde los estudiantes pueden reafirmar y aplicar el conocimiento adquirido.

El profesor y los estudiantes pueden usar en sus lecciones recursos computacionales para ejemplificar algunos temas del curso.

ESTRATEGIAS DE APRENDIZAJE

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra-clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

RECURSOS DIDÁCTICOS

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

EVALUACIÓN

Concepto	**Ponderación**
Prácticas y laboratorios	25%
Proyecto final	25%
Examen 1	25%
Examen 2	25%
Total	**100%**

BIBLIOGRAFÍA

Referencias principales:

- Ceballos, F. (2010)··· *Java 2: Curso de Programación (4ª Edición)*. Editorial Alfa omega Ra-Ma.
- Farrell, J. (2017). *Programming Logic and Design*, *Comprehensive (9 ª Edición)*. Estados Unidos de América: Course Technology Cengage Learning

Referencias clásica:

- Farrel, J. (2000). Iniciación a la Programación Lógica y Diseño. Paraninfo

Programa	Diplomado en Desarrollo de Software
Curso	DDS-203 Principios de Base de Datos
Créditos	4
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	II
Periodo	Cuatrimestral
Requisitos	DDS-101 Principios de Programación
Horas	3

DESCRIPCIÓN DEL CURSO

Este curso prepara al estudiante para conocer la teoría de bases de datos que incluye los principios formales para definir y manipular datos estructurados e interrelacionados. Para definir los datos se utiliza un modelo de datos y para su manipulación un lenguaje. Diferentes modelos de datos se han propuesto buscando un mayor nivel expresivo para representar el mundo real.

Poder realizar el diseño de modelos de datos y su implementación en bases de datos relacionales, será posible por medio del uso de modelos conceptuales y lógicos.

OBJETIVOS

Objetivo general

Diseñar una estrategia optimizada del manejo de datos, mediante el uso de notaciones estándar de modelaje y las facilidades de programación de un Sistema Administrador de Base de Datos Relacional, que brinde solución a situaciones pequeñas y medianas de almacenamiento y manipulación de datos, desde la perspectiva de un desarrollador de software transaccional.

Objetivos específicos

- Comprender los diferentes modelos de bases de datos, y en concreto dominar el modelo relacional.
- Ser capaz de realizar el modelo de una base de datos relacional, a partir de la especificación de requerimientos de un proyecto, comprendiendo y aplicando los conceptos y transformaciones implicados.
- Conocer el uso y administración de dos de los gestores de bases de datos relacionales más populares en el ámbito del software libre: PostgreSQL y MySQL.
- Tener experiencia en el desarrollo de aplicaciones en conexión con bases de datos en varios lenguajes.

CONTENIDOS

Tema 1. Sistemas de información y almacenamiento de datos

- Conceptos en sistemas de información.
- Estilos arquitectónicos para el diseño de sistemas de información.
- Administración de datos dentro de los sistemas de información.

Tema 2. Conceptos de bases de datos

- Conceptos de las bases de datos relacionales: relación (tabla), atributo (campo), tupla (registro), llave, índice.
- Proceso de construcción de una base de datos.

Tema 3. Modelado de datos

- Análisis del modelo del negocio.
- · Consideraciones de diseño de una Base de Datos (volumen de datos, escalabilidad, distribución de nodos, carga de la red).
- Diseño de un modelo de datos.
- Modelo de Entidad – Relación (modelo de Chen).
- Paso del modelo lógico al modelo físico.
- Formas normales y normalización de una base de datos relacional.
- Herramienta de Ingeniería del Software Asistida por Computadora (CASE) para el modelado de datos

Tema 4. Sistema administrador de bases de datos

- Características de un Sistema Administrador de Bases de Datos Relacionales (SABDR).
- Instalación del Sistema Administrador de Bases de Datos Relacionales (SABDR).
- Comandos Principales.
- Tipos de Datos.
- Seguridad – Usuarios.
- Interfaz Gráfica de Usuario (GUI, por sus siglas en inglés).

Tema 5. Álgebra relacional

- El modelado relacional de datos desde una perspectiva matemática.
- Selección.
- Proyección.
- Reunión ('*Join'* en inglés).
- Unión.
- Intersección.
- Diferencia.
- Producto Cartesiano.
- División.

Tema 6. Definición y manipulación de datos

- Conceptos básicos
- Lenguaje de consulta estructurado (SQL).
- Definición de tablas, llaves, relaciones, índices.
- Manipulación de datos
- Consulta, inserción, actualización y borrado de datos.
- Programación del Lenguaje de consulta estructurado (SQL)
- Creación de vistas, procedimientos almacenados, disparadores ('*triggers'* en inglés), funciones de usuario, cursores y transacciones

Tema 7. Integridad de datos

- Variedades de integridad de datos.
- Técnica para la implementación de la integridad (restricciones, reglas, valores predeterminados).

METODOLOGÍA

El trabajo en clase incluye exposiciones magistrales en las cuales los temas son expuestos por el profesor, el realiza una retroalimentación con los estudiantes y tiene la posibilidad de aclarar dudas y fomentar la participación de los estudiantes. De la mano va el trabajo en clase también incluye prácticas individuales, en donde los estudiantes pueden reafirmar y aplicar el conocimiento adquirido.

El profesor y los estudiantes pueden usar en sus lecciones recursos computacionales para ejemplificar algunos temas del curso.

ESTRATEGIAS DE APRENDIZAJE

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra-clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

RECURSOS DIDÁCTICOS

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

EVALUACIÓN

Concepto	Porcentaje
Laboratorio 1	10%
Laboratorio 2	10%
Examen 1	20%
Examen 2	20%
Prácticas en clase	10%
Tareas extra-clase	10%
Proyecto final	20%
Total	**100%**

BIBLIOGRAFÍA

Refrencias principales:

- Elmasri, R. (2015). *Fundamentos de Sistemas de Bases de Datos (7 Edición).* Pearson Education. (Addison- Wesley)
- González, C. (2009). *Sistemas de Bases de Datos.* Editorial Tecnológica de Costa Rica.

Programa	Diplomado en Desarrollo de Software
Curso	ING-301 inglés III
Créditos	3
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	II
Periodo	Cuatrimestral
Horas	3

Descripción del curso

Este curso provee al estudiante de la formación intermedia del idioma inglés para continuar en el conocimiento de vocabulario y estructuras gramaticales para iniciar un proceso de conversación aceptable del idioma inglés en temas propios de la carrera.

Objetivo General

Desarrollar en el estudiante habilidades y estrategias de compresión de literatura escrita en el idioma inglés, de herramientas y técnicas que le permitan sostener una conversación aceptable. Se estimulará la interacción del idioma desde el primer momento.

Objetivos Específicos

1. Desarrollar y mejorar las habilidades de comunicación oral del idioma.
2. Mantener e iniciar conversaciones en el idioma inglés, siendo complemento los cursos anteriores.
3. Enriquecer el vocabulario.
4. Mejorar la pronunciación.
5. Adquirir fluidez en la expresión oral.

6. Trabajar en parejas y grupos para eliminar las limitaciones propias mentales y dar paso a actitudes positivas y de superación personal.

Contenidos

1. El profesor elaborará estrategias para establecer grupos de trabajo en función del tiempo del curso. Deberá preparar la documentación impresa y digital para ser entregada a cada estudiante con el fin de dar un seguimiento al avance conversacional.

2. El profesor deberá elaborar dinámicas para ir gradualmente aumentando las habilidades de comunicación oral del idioma. Los estudiantes deberán ir aumentando su gramática, vocabulario y pronunciación con técnicas útiles para la expresión oral. Para ello se trabajará con una metodología ecléctica que incorpora el uso de material audiovisual, plataformas computacionales y actividades que desarrollarán la comprensión lectora y auditiva, la habilidad oral y escrita.

3. El procedimiento a utilizar será la de establecer combinación de trabajo en grupo o parejas, debates en equipos y presentaciones orales individuales. Todos los estudiantes se distribuirán en niveles en función del aprendizaje obtenido con anterioridad.

4. Vinculación de inglés con disciplina de textos adecuados

Metodología

Los contenidos proporcionan apoyo para el desarrollo de habilidades receptivas y de producción, de ahí que las actividades se enfoquen en lograr la fluidez conversacional; por tanto, será indispensable la participación activa del alumno y el profesor. Entre las actividades que se realizarán, se pueden citar: group work, pair work, role play, brainstorming, warm up activities.

Recursos

Para desarrollar este curso se requiere algunos capítulos o secciones de los libros mencionados, de acuerdo con el ordenamiento de la temática anotada. El contenido servirá para trabajar en clase o para realizar prácticas en el hogar.

En cuanto al equipo, se requieren dispositivos de tipo audio-visual, tales como la videograbadora, cámaras, DVD y el televisor, además de la radio grabadora.

Evaluación

Concepto	Porcentaje
2 evaluaciones parciales	40%
1 presentación final sostenible de 20 minutos-dinámica grupos	30%
Tareas y ejercicios auditivos	5%
Pruebas cortas	10%
Ensayos cortos	5%
Tareas y ejercicios escritos	5%
Ejercicios orales	5%

Bibliografía

Libro de Texto

Abranhamson, Richard y otros. A to Zoo. Estados Unidos de América: Editorial Harcourt Brace, 1997.

Libros de Clásico

Banathy, Bela, Lange, Dale. A Design for Foreign Language Curriculum. U.S.A.: DC Heath and Company, 1992.

Collins, Harold, et. Al. Educational measurement and evaluation. 2a. ed. Usa: Scott, Foresman and Co., 1996.

Doll C, Ronald. Curriculum improvement. United States: Allyn and Bacon, 1996.

Green C., Alan. Educational facilities with new media. 5a. ed. USA: Association for Educational Communications and Technology, 1994.

Richards, Jack. Inter-intro English for international communication. US: Cambridge University Press, 1994.

Schrampfer, Betty Fundamentals of English Grammar. Second Edition. Englewood Cliffs, New Jersey: Prentice Hall Regents, 1992.

Programa	Diplomado en Desarrollo de Software
Curso	DDS-302 Programación II
Créditos	4
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	III
Periodo	Cuatrimestral
Requisitos	DDS- 202 Programación I

DESCRIPCIÓN DEL CURSO

La computadora es una herramienta fundamental, por lo tanto, es necesario enseñar al estudiante su uso y aplicación práctica en la solución de problemas de mediana y alta complejidad. Para ello se deben desarrollar habilidades en el uso de las técnicas actuales que simplifican y agilizan la tarea de diseño y construcción de los programas computacionales requeridos, así como el conocimiento de los distintos tipos de datos abstractos que se pueden emplear en el desarrollo de sistemas programados.

OBJETIVOS

Objetivo general

Desarrollar en el estudiante la habilidad para el diseño y especificación usando diagramas de flujo y diseños en un lenguaje de programación de alto nivel.

Objetivos específicos

- Conocer y manipular con destreza el concepto de abstracción de datos.
- Diseñar, especificar e implementar tipos de datos abstractos.

- Dominar con destreza los paradigmas de la orientación por objetos.
- Conocer técnicas para la especificación, diagramación y desarrollo de sistemas programados orientados por objetos.
- Diseñar e implantar la solución de problemas de mediana complejidad utilizando los distintos tipos de datos abstractos.
- Conocer los distintos tipos de almacenamiento en memoria secundaria.
- Conocer y manejar las distintas organizaciones de archivos para el almacenamiento de datos persistentes.
- Dominar los constructos de un lenguaje de programación de alto nivel orientado por objetos.

CONTENIDOS

Tema 1. EL AMBIENTE DE PROGRAMACIÓN

- Introducción a la historia del lenguaje de programación
- Estructura del lenguaje de programación
- Instalación del editor del lenguaje de programación

Tema 2. FLUJO DE CONTROL EN EL LENGUAJE DE PROGRAMACIÓN

- Flujo de control secuencial
- Flujo de control condicional
- Flujo de control iterativo

Tema 3. ABSTRACCIONES

- Matrices
- Separación del programa en capas
- Manejo de hileras
- Manejo de colecciones

Tema 4. INTRODUCCIÓN A LA ORIENTACIÓN A OBJETOS

- Transición desde el paradigma procedimental.
- Conceptos del paradigma de programación orientado a objetos

Tema 5. DESARROLLO ORIENTADO A OBJETOS

- Abstracción.
- Encapsulamiento.
- Modularidad.

METODOLOGÍA

El trabajo en clase incluye exposiciones magistrales en las cuales los temas son expuestos por el profesor, el realiza una retroalimentación con los estudiantes y tiene la posibilidad de aclarar dudas y fomentar la participación de los estudiantes. De la mano va el trabajo en clase también incluye prácticas individuales, en donde los estudiantes pueden reafirmar y aplicar el conocimiento adquirido.

El profesor y los estudiantes pueden usar en sus lecciones recursos computacionales para ejemplificar algunos temas del curso.

ESTRATEGIAS DE APRENDIZAJE

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra-clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

RECURSOS DIDÁCTICOS

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

EVALUACIÓN

Concepto	Ponderación
Trabajo grupal	5%
Prácticas y laboratorios	20%
Proyecto final	25%
Examen 1	25%
Examen 2	25%
Total	100%

BIBLIOGRAFÍA

Referencias principales:

- Farrell, J. (2017). *Programming Logic and Design*, *Comprehensive (9 ª Edición)*. Estados Unidos de América: Course Technology Cengage Learning

Referencias clásicas:

- Ceballos, F. (2010) ... *Java 2: Curso de Programación (4ª Edición)*. Editorial Alfaomega Ra-Ma.
- Farrel, J. (2000). Iniciación a la Programación Lógica y Diseño. Paraninfo

Programa	Diplomado en Desarrollo de Software
Curso	DDS-303 Proyecto de Software 1
Créditos	4
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	III
Periodo	Cuatrimestral
Requisito	DDS 202 Programación I

DESCRIPCIÓN DEL CURSO

Durante el curso se desarrolla un proyecto siguiendo las buenas prácticas de calidad, cuyo producto es una aplicación web con su correspondiente base de datos. Busca que el estudiante se maneje en las disciplinas de trabajo que se utilizan en diferentes empresas donde aplican procesos de desarrollo de software y crean de programas para Internet en busca del desarrollo de habilidades y competencias en áreas interdisciplinarias, con el fin de construir una aplicación de software basada en Web, que realizará validaciones de datos en el cliente y accede a una base de datos ubicada en un servidor; todo esto aplicando las etapas del proceso de desarrollo de software.

Simula condiciones reales de trabajo, donde se aplican roles y pautas para el desarrollo del proyecto.

OBJETIVOS

Objetivo general

Crear una aplicación de software mediante la aplicación de procesos adecuados de planificación, análisis, diseño, implementación y pruebas, la

permita a los cibernautas hacer tareas básicas por medio de una página Web interactiva.

Objetivos específicos

Conceptualizar y planificar un proyecto de software usando métodos, técnicas y herramientas adecuadas al contexto, para elaborar el Plan de proyecto.

Aplicar el modelo de ciclo de vida iterativo-incremental durante la construcción de la aplicación, mediante el cumplimiento de las etapas de cada uno de los ciclos, con el objetivo de mejorar el programa que se construye.

Realizar las actividades de cada etapa del proceso de desarrollo de software, tales como: requerimientos, análisis, diseño, construcción y validación; con el fin de documentar todos los elementos y especificaciones del software

Usar conceptos básicos de la administración de proyectos de software, utilizando una herramienta de administración de proyectos, para el manejo de tareas, horas y responsabilidades de cada miembro del equipo.

Emplear estándares y técnicas en cada una de las etapas y actividades del proyecto, para asegurar la calidad del software.

Hacer una aplicación de software basada en Web, utilizando el Lenguaje de Marcado de Hipertexto (HTML), las hojas de estilo en cascada (CSS), un lenguaje de programación que permita la interacción con el usuario y acceso a bases de datos relacionales, permitiéndole al usuario realizar tareas por medio de una página Web.

Demostrar buenas prácticas de redacción de documentos técnicos y de exposición de trabajos realizados en equipo, para facilitar la comprensión de los temas y logren comunicar de una manera sencilla y concisa lo que se desea transmitir.

CONTENIDOS

Tema 1. FUNDAMENTOS DE INGENIERÍA DEL SOFTWARE

- Introducción.
- Ciclos de vida de software.
- Técnicas y herramientas para el registro del tiempo.

Tema 2. ASEGURAMIENTO DE LA CALIDAD DEL SOFTWARE

- Generalidades y principios.
- Características de calidad.
- Validación (Revisiones y pruebas)

Tema 3. ESTÁNDARES DE DOCUMENTACIÓN

- Importancia para los proyectos de software.
- Plantilla general de documentación.
- Plantillas para la elaboración de agendas y minutas de reunión.

Tema 4. MODELOS DE DESARROLLO DE SOFTWARE

- Modelos tradicionales
- Modelos emergentes

Tema 5. ESPECIFICACIÓN DE REQUERIMIENTOS DEL SOFTWARE

- Importancia.
- Requerimientos funcionales.
- Requerimientos no funcionales.
- Restricciones.
- Técnicas para la efectiva obtención de requerimientos.
- Aseguramiento de la calidad del software
- Plantilla para la Especificación de Requerimientos del Software (ERS, por sus siglas en español).

Tema 6. ANÁLISIS DEL SOFTWARE

- Lenguaje de Modelado Unificado (UML, por sus siglas en inglés).
- Casos de uso.
- Diagramas de actividad.
- Análisis de riesgos.
- Aseguramiento de la calidad del software
- Documentos de análisis.
- Plantilla para el Informe de Análisis del Software (IAS, por sus siglas en español).

Tema 7. DISEÑO DEL SOFTWARE

- Elementos de diseño.
- Arquitectura del software.
- Aseguramiento de la calidad del software
- Plantilla para el Informe de Diseño del Software (IDS, por sus siglas en español).
- Diseño de escenarios de prueba para el sistema.

Tema 8. LENGUAJE DE MARCADO DE HIPERTEXTO (HTML, por sus siglas en inglés)

- Aspectos básicos.
- Formularios.

Tema 9. HOJAS DE ESTILO EN CASCADA (CSS, por sus siglas en inglés).

- Aspectos básicos.

Tema 10. LENGUAJES DE PROGRAMACIÓN INTERPRETADOS

- Introducción.
- Características de los lenguajes de programación interpretados (lenguajes 'script', en inglés).
- Uso de los lenguajes de programación interpretados para la creación de páginas Web.
- Estándar de nomenclatura.
- Validación de formularios.

Tema 11. DISEÑO DE SITIOS WEB

- Elementos de diagramación.
- Manejo del color.
- Diseño básico de interfaces de usuario (apariencia y sensación).

Tema 12. CONSTRUCCIÓN DEL SOFTWARE

- Publicación y administración de sitios Web en una Intranet.
- Manejo de datos.
- El lenguaje de programación interpretado: PHP (por sus siglas en inglés)
- Diseño y realización de pruebas unitarias.
- Plantilla para el Informe Final del Proyecto (IFP, por sus siglas en español).

Tema 13. TÉCNICAS DE COMUNICACIÓN EMPRESARIAL

- Comunicación escrita efectiva.
- Comunicación en equipos de trabajo.
- Comunicación oral.
- Comunicación en aplicaciones Web.

Tema 14. ADMINISTRACIÓN DE PROYECTOS DE SOFTWARE

- Conceptos básicos de administración de proyectos de software.

Tema 15. TÉCNICAS PARA DESARROLLAR COMPETENCIAS DE EFECTIVIDAD PERSONAL

- Introducción a la inteligencia emocional en el contexto de TI
- Percepción de sí mismo
- Expresión de sí mismo

METODOLOGÍA

- El trabajo en clase se distribuye en sesiones teóricas y sesiones prácticas. Las sesiones teóricas consisten en exposiciones magistrales por parte del profesor seguidas por interrogatorios didácticos a los estudiantes en los que el profesor tiene la oportunidad de aclarar dudas y fomentar la participación de los estudiantes. El profesor también compara y contrasta diferentes ejemplos en cada tema para ayudar a aclarar los conocimientos y facilitar su comprensión.
- En las sesiones prácticas se hacen laboratorios en computadora, individuales, que giran alrededor de asignaciones fijadas por el profesor, diseñadas también para reforzar los conocimientos y aplicarlos. Durante los laboratorios, los estudiantes trabajan bajo la supervisión del profesor, teniéndolo a su disposición para ser consultado cuando se requiera. También se realiza trabajo en grupo, que se genera de las asignaciones que el profesor le hace a cada equipo de estudiantes.

ESTRATEGIAS DE APRENDIZAJE

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

RECURSOS DIDÁCTICOS

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

EVALUACIÓN

Concepto	**Ponderación**
Tareas	5%
Laboratorios	10%
Exámenes	30%
Documentación del proyecto	20%
Presentación del proyecto	25%
Avances del proyecto (2)	10%
Total	100%

BIBLIOGRAFÍA

Referencias clásica

- Fowler, M. (2000). *UML gota a gota*. Addison Wesley Longman.
- Lerdorf, R.; Tatroe, K.; y MacIntyre, P. (2013). *Programming PHP*. O'Reilly.
- Conallen, J. (2002). *Building Web Applications with UML*. Addison Wesley.

Programa	Diplomado en Desarrollo de Software
Curso	DDS-304 Fundamentos Eléctricos
Créditos	4
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	III
Periodo	Cuatrimestral
Requisitos	MAT 102 Matemática General

Descripción

Con este curso se pretende que el participante adquiera una formación en el campo del análisis de circuitos en corriente directa, así como en la manipulación de las magnitudes eléctricas básicas. Además, desarrollará habilidades en la utilización de teoremas de análisis de circuitos eléctricos y en los procesos de experimentación en el campo de la electricidad.

Objetivo General:

Conocer los fenómenos eléctricos en corriente continua, así como los diferentes instrumentos utilizados en electricidad.

Objetivos Específicos

1. Realizar lecturas con instrumentos específicos de parámetros que intervienen en circuitos de corriente continua.

2. Conocer los fenómenos de la corriente eléctrica a través de un conductor, así como los factores y unidades que lo determinan.

3. Conocer y calcular potencia eléctrica en circuitos de corriente continua.

4. Conocer las magnitudes eléctricas elementales utilizadas en corriente continua.

5. Realizar cálculos y mediciones en circuitos resistivos sencillos en corriente continua.

6. Conocer los diferentes teoremas que permiten el cálculo y análisis de circuitos mixtos en corriente continua.

7. Describir los fenómenos de capacitancia e inductancia eléctrica.

Contenido:

I **LA CORRIENTE ELECTRICA**.

1.1. Composición de la materia
1.2. La carga electroestática.
1.3. El flujo de corriente: conductores y aisladores, la batería, fuentes de DC y el circuito eléctrico.
1.4. Medición de la corriente.

II **EL VOLTAJE**

2.1. La fuerza electroestática.
2.2. La producción de una fuerza electromotriz EMF.
2.3. Tipos de baterías y sus conexiones.
2.4. Caídas de tensión y rizados de voltaje.
2.5. Concepto de tierra.
2.6. Medición de voltaje.

III RESISTENCIA.

3.1. El ohmio.
3.2. Factores que determinan la resistencia.
3.3. Tipos de resistores.
3.4. Tolerancia y potencia de las resistencias.
3.5. Conexiones serie y paralelo.
3.6. Conexión mixta
3.7. Medición de resistencia.

IV LA LEY DE OHM.

4.1. Uso de la Ley de Ohm.
4.2. Definición de potencia.

V MAGNETISMO.

5.1. El campo magnético.
5.2. Electricidad y magnetismo.
5.3. El flujo magnético.
5.4. Inducción magnética, generadores.

VI ANALISIS DE CIRCUITOS EN DC.

6.1. Circuitos resistivos, tipo puente, división de extensión y de corriente.
6.2. Leyes de Kirchoff.
6.3. Teorema de Thevenin y Norton.
6.4. Máxima transferencia de potencia.

VII INDUCTANCIA Y CAPACITANCIA.

7.1 Inductancia e inductores.
7.2 Constantes de tiempo en circuitos RL.
7.3 Capacitancia y tipos de capacitores
7.4 Capacitores en serie y paralelo.
7.5 Constante de tiempo en circuitos RC.

METODOLOGÍA

El trabajo en clase incluye exposiciones magistrales en las cuales los temas son expuestos por el profesor, el realiza una retroalimentación con los estudiantes y tiene la posibilidad de aclarar dudas y fomentar la participación de los estudiantes. De la mano va el trabajo en clase también incluye prácticas individuales, en donde los estudiantes pueden reafirmar y aplicar el conocimiento adquirido.

El profesor y los estudiantes pueden usar en sus lecciones recursos computacionales para ejemplificar algunos temas del curso.

ESTRATEGIAS DE APRENDIZAJE

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

RECURSOS DIDÁCTICOS

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

Evaluación

Concepto	Ponderación
I Examen parcial	15%
II Examen parcial	15%
Laboratorio	30%
Examen final	30%
Trabajos extracurriculares	10%

Bibliografía clásica

- Boylestad, Robert L... ANALISIS INTRODUCTORIO DE CIRCUITOS'. Editorial Trillas. México. 1998. X edición.
- Slurzberg y Osterheld. "FUNDAMENTOS DE ELECTRICIDADELECTRONICA". Editorial Maccraw-Hill. 1970. III edición.
- Fitzgerald y otros. "FUNDAMENTOS DE INGENIERIA ELECTRICA". Editorial Maccraw-Hill. México. 1975.

Programa	Diplomado en Desarrollo de Software
Curso	AE-401 Autogestión Empresarial
Créditos	3
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	IV
Periodo	Cuatrimestral
Requisitos	
Horas	4

DESCRIPCIÓN DEL CURSO

Este curso permite a los estudiantes adquirir fundamentos teóricos y metodológicos esenciales para la comprensión de las funciones de la administración general y de las actividades administrativas fundamentales que llevan a cabo las empresas, como lo son: producción, recursos humanos, mercadeo, contabilidad y finanzas, para comprender la dinámica de las organizaciones y el uso que éstas dan a la información y la tecnología.

OBJETIVO GENERAL

Analizar las funciones y las actividades que llevan a cabo las organizaciones, usando la teoría administrativa y los sistemas de información como marcos de referencia a fin de que comprender el uso y la importancia de la información y la tecnología en las organizaciones.

OBJETIVOS ESPECÍFICOS

- Comprender el concepto de organización, sus diversos tipos y sus procesos típicos, mediante el estudio de los procesos y de ejemplos concretos, a fin de que el estudiante entienda cómo funcionan las empresas.
- Diferenciar las actividades básicas de la administración: organización, control, planeación y dirección de las organizaciones, para comprender qué tipo de

información es importante y qué tipo de tecnología sirve de apoyo a las actividades de una empresa, que le permita ser competente.

- Reconocer la actividad financiera de las organizaciones, mediante el estudio de los procesos y los sistemas de información que los apoyan, a fin de comprender la importancia de la información contable para determinar el estado financiero de una organización.
- Diferenciar diversas formas de organizar la producción en las empresas, mediante el estudio de los procesos de compras, logística y control de inventarios, para el uso de la información en las formas de organizar las operaciones de manera eficiente en las empresas.
- Clasificar las actividades requeridas para el mercadeo de productos y servicios en una organización, mediante el estudio del proceso de la mercadotecnia y de ejemplos exitosos y no exitosos, a fin de que se utilice la información adecuada para hacer mercadeo en las empresas.

CONTENIDOS

Tema 1: INTRODUCCIÓN A LOS PROCESOS ORGANIZATIVOS

- Significado de administración de procesos empresariales.
- ¿Qué son las organizaciones?
- Cambios en el entorno de las organizaciones.
- Trabajo directivo en las organizaciones.
- Diversidad en los tipos de organización.

Tema 2: Las funciones administrativas en las organizaciones

- Planeación.
- Organización.
- Dirección.
- Control.

Tema 3: PLANEACIÓN DE PROCESOS EMPRESARIALES

- Propósito de la planeación.
- Objetivos de la planeación.
- Estrategias, políticas y premisas.
- Toma de decisiones y uso de la información.

Tema 4: ORGANIZACIÓN DE LA EMPRESA

- Naturaleza de la organización.

- Estructura organizacional: departamentalización.
- Autoridad de línea y descentralización.
- Cultura organizacional.
- Los procesos en la organización.
- Uso de la información en la organización.

Tema 5: ADMINISTRACIÓN DE OPERACIONES

- Administración de la producción y de la productividad.
- Administración de materiales y control de inventarios.
- Compras y logística de abastecimiento.
- Canales y logística de distribución.
- Administración de la tecnología.
- Uso de los sistemas de información en la Administración de Operaciones.

Tema 6: ADMINISTRACIÓN DE LA MERCADOTECNIA

- Etapas de la Mercadotecnia.
- Orientación hacia el mercado.
- Factores controlables y no controlables.
- Segmentación e investigación de mercados.
- Uso de los sistemas de información en la Mercadotecnia.

Tema 7: ADMINISTRACIÓN DEL TALENTO HUMANO

- Administración y selección de personal.
- Evaluación del desempeño y estrategia del desarrollo profesional.
- Administración del cambio organizacional.
- Los factores humanos y la motivación.
- Uso de los sistemas de información en la Administración del Talento Humano.

Tema 8: CONTABILIDAD Y ADMINISTRACIÓN FINANCIERA

- Información contable.
- Tipos de contabilidad.
- La función financiera en la empresa.
- Las funciones del administrador financiero.
- Los sistemas de información financiera-contable.

METODOLOGÍA

El trabajo en clase incluye exposiciones magistrales en las cuales los temas son expuestos por el profesor, el realiza una retroalimentación con los estudiantes y tiene la posibilidad de aclarar dudas y fomentar la participación de los estudiantes. De la mano va el trabajo en

clase también incluye prácticas individuales, en donde los estudiantes pueden reafirmar y aplicar el conocimiento adquirido.

El profesor y los estudiantes pueden usar en sus lecciones recursos computacionales para ejemplificar algunos temas del curso.

ESTRATEGIAS DE APRENDIZAJE

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

RECURSOS DIDÁCTICOS

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

EVALUACIÓN

Concepto	**Ponderación**
Examen parcial 1	20%
Examen parcial 2	20%
Tareas cortas	10%
Interrogatorios didácticos	2%
Discusiones dirigidas	8%
Proyectos en grupo	20%
Proyecto final	20%
Total	100%

BIBLIOGRAFÍA

Clásica:

- Robbins, S. & Coulter, M. (2010). *Administración*. México: Pearson.

Referencias complementarias:

- Jones, G. (2018). *Administración Contemporánea, 10 Edición*. México: McGraw-Hill Interamericana

Programa	Diplomado en Desarrollo de Software
Curso	DDS -401 Programación III
Créditos	4
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	IV
Periodo	Cuatrimestral
Requisitos	DDS- 301 Programación II

DESCRIPCIÓN DEL CURSO

Este curso introduce a los estudiantes en el paradigma de programación orientado a objetos, que incluye los principios fundamentales de análisis, diseño y programación orientada a objetos, como lo son la abstracción, encapsulamiento, modularidad y jerarquía de clases, para construir aplicaciones de software de múltiples capas en el lenguaje de programación Java.

OBJETIVOS

Objetivo general

Crear aplicaciones de software pequeñas y medianas, de múltiples capas y persistentes, con énfasis en el análisis y el diseño, aplicando el análisis y diseño detallado orientado a objetos, usando el Lenguaje de Modelado Unificado, el lenguaje de programación Java y las bases de datos relacionales, a fin de que las aplicaciones de software cumplan criterios de calidad.

Objetivos específicos

- Elaborar el análisis y diseño detallado de las aplicaciones de software, usando la notación del Lenguaje de Modelado Unificado (ó UML por sus

siglas en inglés), para que las aplicaciones desarrolladas sean fáciles de entender, fáciles de corregir y fáciles de modificar.

- Formular la programación de las aplicaciones de software, mediante el uso del lenguaje de programación Java, a fin de que los análisis y diseños elaborados sean implementados y probados en la computadora.
- Emplear el modelo de arquitectura de múltiples capas en la construcción de aplicaciones de software, diseñando las capas de presentación, capa lógica y capa de datos, para elaborar aplicaciones de software eficiente y legible.
- Usar buenas prácticas en la elaboración de aplicaciones de software, mediante el uso de estándares, la aplicación de principios de diseño, el uso de buen estilo de codificación y la aplicación de técnicas, para asegurar la calidad de las aplicaciones de software elaboradas.
- Utilizar las bases de datos relacionales, mediante el modelo de entidad-relación y el lenguaje de consultas, para construir aplicaciones de software orientadas a objetos persistentes.

CONTENIDOS

Tema 1. Conceptos del paradigma de programación orientado a objetos

- Abstracción.
- Encapsulamiento.
- Modularidad.

Tema 2. RELACIONES ENTRE OBJETOS

- Dependencia.
- Asociación.
- Agregación.
- Composición.
- Representación de relaciones.

Tema 3. RELACIONES ENTRE CLASES

- Estructura de clases.
- Herencia.
- Redefinición de métodos.
- Modificadores de acceso.
- Clases abstractas.
- Polimorfismo.
- Ligado: estático y dinámico.
- Conversión de tipos.

Tema 4. MODELADO DE APLICACIONES DE SOFTWARE

- Técnicas de Modelado.
- Contrato de Operaciones.
- Diagramas de interacción.
- Diagramas de Actividad.

Tema 5. INTERFACES, EXPRESIONES E HILOS

- Concepto, terminología e implementación.
- Comparación con clases abstractas.
- Comparación con herencia.
- Manejo de expresiones e hilos

Tema 6. ARQUITECTURA DE SOFTWARE

- Arquitectura de múltiples capas
- Conexión a bases de datos relacionales

Tema 7. EXCEPCIONES

- Concepto y utilidad.
- Manejo de excepciones.
- Implementación de excepciones en Java.
- Jerarquía de Excepciones.

Tema 8. INTERFAZ GRÁFICA

Tema 9. DISEÑO Y APLICACIÓN DE PRUEBAS Y REVISIONES

- Pruebas de Caja Negra.
- Pruebas de Unidad.
- Revisiones de escritorio.
- Revisión grupal de diseños orientados a objetos.

METODOLOGÍA

El trabajo en clase incluye exposiciones magistrales en las cuales los temas son expuestos por el profesor, el realiza una retroalimentación con los estudiantes y tiene la posibilidad de aclarar dudas y fomentar la participación de los estudiantes. De la mano va el trabajo en clase también incluye prácticas individuales, en donde los estudiantes pueden reafirmar y aplicar el conocimiento adquirido.

El profesor y los estudiantes pueden usar en sus lecciones recursos computacionales para ejemplificar algunos temas del curso.

ESTRATEGIAS DE APRENDIZAJE

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

RECURSOS DIDÁCTICOS

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los

casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

EVALUACIÓN

Concepto	Ponderación
Comprobaciones de lectura	20%
Laboratorios en clase	10%
Proyecto Final	30%
Examen 1	20%
Examen 2	20%
Total	100%

BIBLIOGRAFÍA

Bibliografía clásica:

· Ceballos, F. (2010). *Java 2: Curso de Programación (4ª Edición)*. Editorial Alfaomega Ra-Ma.

· Larman, C. (2004). *UML y Patrones: Introducción al análisis y diseño orientado a objetos*. Prentice Hall.

Referencias complementarias:

· Bruegge B.; Dutoit, A. (2013). Object-Oriented Software Engineering Using UML, Patterns, and Java (3rd Edition). Prentice Ha

Programa	Diplomado en Desarrollo de Software
Curso	DDS- 403 Programación de Base de Datos
Créditos	4
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	IV
Periodo	Cuatrimestral
Requisitos	DDS- 101 Principios de Programación, MAT- 102 Matemática General

DESCRIPCIÓN

Este curso introduce al estudiante en temas avanzados y de optimización de bases de datos, como lo son: consultas complejas, restricciones de integridad, vistas, seguridad, recuperación, concurrencia y transacciones; para que el estudiante logre optimizar el rendimiento de las bases de datos y mejorar la construcción de diseños y la programación de aplicaciones. En el curso se detallan estos temas en profundidad en dos plataformas de Sistemas Administradores de Bases de Datos Relacionales (SABDR).

OBJETIVO GENERAL

Construir bases de datos optimizadas grandes, utilizando criterios de rendimiento, de optimización, de concurrencia y de transacciones, así como programación avanzada de bases de datos relacionales, para construir aplicaciones de software persistentes y eficientes que usen y manipulen grandes volúmenes de datos.

OBJETIVOS ESPECÍFICOS

Comprender la arquitectura de los Sistemas Administradores de Bases de Datos Relacionales más usados en la industria, mediante ejemplos, ejercicios y laboratorios, a fin de que el estudiante pueda utilizar estos sistemas para resolver problemas de bases de datos.

Emplear técnicas y principios de eficiencia y rendimiento, usando procedimientos almacenados y disparadores, para mejorar el rendimiento de aplicaciones intensivas en el manejo de datos.

Aplicar técnicas de seguridad, mediante el control y administración de usuarios, para crear aplicaciones de software con bases de datos seguras.

Diseñar bases de datos, aplicando la optimización en el diseño, programación e implementación, a fin de construir aplicaciones con bases de datos eficientes.

Construir bases de datos, usando criterios de eficiencia, rendimiento, seguridad y optimización, con el fin de manejar grandes volúmenes de información en las aplicaciones de software.

Comprender temas avanzados y actuales de bases de datos, mediante la revisión de casos de estudio, para aplicarlos en el diseño y construcción de las bases de datos.

CONTENIDOS

Tema 1. ARQUITECTURA DE SISTEMAS ADMINISTRADORES DE BASES DE DATOS

- Organización del espacio físico.
- Base de datos del sistema y su impacto en tareas de administración y rendimiento.
- Estructura y funcionalidades del Sistema Administrador de Bases de Datos Relacional (SABDR): *´MS SQL Server´*.
- Estructura y funcionalidades del Sistema Administrador de Bases de Datos Relacional (SABDR): *´Oracle´*.

Tema 2. ARQUITECTURA DE APLICACIONES BASADAS EN BASES DE DATOS

- Evaluación del modelo Cliente/Servidor.
- Evaluación del modelo de múltiples capas.
- Sistema Administrador de Bases de Datos (SABD) para sistemas empotrados.

Tema 3. ELEMENTOS DE RENDIMIENTO Y OPTIMIZACIÓN

- Creación y depuración de procedimientos almacenados y disparadores (*´triggers´, en inglés).*
- Creación e impacto de índices.
- Uso y análisis de la herramienta de captura de eventos del servidor (*´profiler´, en Inglés)*.
- Mejora de consultas.

Tema 4. RECUPERACIÓN

- Cerrojos (*´locks´, en inglés)*.
- Protocolo de dos fases.
- Serialización.

Tema 5. TRANSACCIONES

- Enunciados: Aplicación de transacciones (*´Commit´, en Inglés)* y reversación (*´Rollback´, en Inglés)*.
- Creación y utilización de vistas.
- Recuperación y actualización de vistas.

Tema 6. CONCURRENCIA

- Cerrojos.
- Estampillas de Tiempo (*´Timestamp´, en inglés)*.
- Bloqueo mortal (*´Deadlock´, en Inglés)* y recuperación del bloqueo mortal.
- Bloqueo mortal y por aproximación.

Tema 7. SEGURIDAD Y AUTORIZACIÓN

- Esquema de definición de usuarios y roles.
- Administración de roles.
- Control de acceso discrecional.
- Control de acceso obligatorio.
- Cifrado de datos.

Tema 8. PROGRAMACIÓN AVANZADA DEL SISTEMA ADMINISTRADOR DE BASES DE DATOS RELACIONES (SABDR) *´ORACLE´*

- Introducción al lenguaje de transacciones de bases de datos PL/SQL.
- Tipos de datos compuestos.
- Cursores.
- Uniones (*´Joins´, en Inglés)* y consultas complejas.
- Excepciones.
- Procedimientos, funciones y paquetes.
- Lenguaje de consulta estructurado (SQL) dinámico y meta-datos.
- Consideraciones de diseño.
- Disparadores (*´triggers´, en inglés)*.
- Creación de índices.
- Vistas.

Tema 9. PROGRAMACIÓN AVANZADA DEL SISTEMA ADMINISTRADOR DE BASES DE DATOS RELACIONES (SABDR) SQL Server

- Introducción al lenguaje de transacciones de bases de datos SQL (*´Transact-SQL´, en inglés).*
- Integridad de datos.
- Manejo de errores.
- Diseño y creación de índices.
- Consulta a múltiples tablas.
- Sub-consultas.
- Transacciones.
- Datos distribuidos.
- Procedimientos almacenados.
- Tablas temporales y cursores.
- Disparadores (*´triggers´, en inglés)*.
- Manejo de perfiles y medición.
- Mejora de consultas.
- Manejo del Lenguaje de marcas extensible (XML).
- Índices.
- Vistas.

Tema 10. TEMAS AVANZADOS

- Apoyo para la toma de decisiones.
- Bases de datos móviles.
- Bases de datos en la Telaraña de cobertura mundial, la *´WWW´*.
- Bases de datos orientadas a objetos.
- Bases de datos objeto-relacionales.
- Bases de datos textuales.
- Bases de datos espaciales.

METODOLOGÍA

El trabajo en clase incluye exposiciones magistrales en las cuales los temas son expuestos por el profesor, el realiza una retroalimentación con los estudiantes y tiene la posibilidad de aclarar dudas y fomentar la participación de los estudiantes. De la mano va el trabajo en clase también incluye prácticas individuales, en donde los estudiantes pueden reafirmar y aplicar el conocimiento adquirido. El profesor y los estudiantes pueden usar en sus lecciones computacionales para ejemplificar algunos temas del curso.

ESTRATEGIAS DE APRENDIZAJE

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

RECURSOS DIDÁCTICOS

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

EVALUACIÓN

Concepto	Ponderación
Laboratorio 1	12,5%
Laboratorio 2	12,5%
Prácticas en clase	8%
Interrogatorios didácticos	2%
Proyecto	25%
Examen 1	20%
Examen 2	20%
Total	100%

BIBLIOGRAFÍA

Bibliografía obligatoria

- Harrison, G. (2009) [2]. *Oracle Performance Survival Guide: A Systematic Approach to Database Optimization*. Prentice Hall.
- Korth, H., Silberschatz, A., & Sudarshan, S. (2019). *Database System Concepts. (7th Edition)*. McGraw-Hill Science/Engineering/Math.

Bibliografía complementaria

- González, C. (2009) [3]. *Sistemas de Bases de Datos.* Editorial Tecnológica de Costa Rica.
- SQL Server 2008 R2. (2008) [4]. *SQL Server Book Online, MSDN*. Disponible libremente en: http://msdn.microsoft.com/en-us/library/ms130214.aspx (Consultado en septiembre del 2019)

Programa	Diplomado en Desarrollo de Software
Curso	DDS-404 Proyecto de Software II
Créditos	4
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	IV
Periodo	Cuatrimestral
Requisitos	DDS- 303 Proyecto de Software I

DESCRIPCIÓN DEL CURSO

Este curso profundiza al estudiante en las disciplinas de trabajo utilizadas en empresas que producen software, aplicando un proceso de desarrollo de software iterativo-incremental, utilizando aspectos técnicos de la tecnología Microsoft y desarrollando habilidades humanas y de comunicación, para crear una aplicación transaccional de calidad, de tipo local y Web. Una aplicación de tipo local consiste en un software que se ejecuta en una computadora sin necesidad de conectarse a Internet. Este software debe ser instalado y configurado en cada una de las estaciones en las que se desee utilizar. Por el contrario, una aplicación Web es un software que se ejecuta desde un navegador conectado a Internet y no se necesita instalar en la computadora, debido a que realiza las peticiones a un servidor remoto desde el navegador.

Durante el curso se conforman equipos de trabajo, de cinco a seis estudiantes, los cuales durante el periodo académico deben construir una aplicación de software transaccional. La aplicación debe consistir en dos partes, una que se podrá acceder de manera local y la otra de manera Web vía Internet. Ambas partes deben realizar validaciones de datos en el cliente, acceder una base de datos y publicar reportes. Para esto deben cumplir con las actividades de las etapas del proceso de desarrollo

de software iterativo-incremental, que son: requerimientos, análisis, diseño, construcción y validación.

El desarrollo de la aplicación se lleva a cabo en un ambiente que simula condiciones reales de trabajo, donde cada equipo trabaja como una empresa, cada miembro del equipo cumple un rol y cada rol sigue las pautas que se indican en el plan del proyecto. Así, al terminar las tareas de cada etapa del plan, el equipo habrá construido una aplicación de calidad compuesta de la parte local y la parte Web.

El proceso de desarrollo de software que debe seguir cada uno de los equipos, define los productos que deben elaborarse en las actividades del plan, los estándares que rigen su forma y contenido, y las responsabilidades de los miembros del equipo de trabajo.

Objetivo general

Crear una aplicación de software local y para la Web, de tipo transaccional, empleando un proceso de desarrollo de software iterativo-incremental, siguiendo estándares de calidad y aplicando disciplinas de proyectos orientadas al trabajo individual y en equipo, con el fin de proporcionarle una solución al usuario mediante una aplicación de software.

Objetivos específicos

Aplicar el ciclo de vida iterativo-incremental durante la construcción de la aplicación, mediante el cumplimiento de las etapas de cada uno de los ciclos, con el objetivo de mejorar el programa que se construye.

Usar conceptos de administración de proyectos de software para la elaboración del plan de trabajo, con el fin de distribuir los recursos existentes tales como el tiempo, recursos y tareas.

Emplear la notación Lenguaje de Modelado Unificado (UML, por sus siglas en inglés), mediante una herramienta para la diagramación, con la finalidad de realizar el análisis y diseño conceptual de un sistema de software Web.

Aplicar una arquitectura de múltiples capas para la construcción de la aplicación de software, siguiendo los conceptos de la arquitectura cliente-servidor.

Diseñar la aplicación de software, tanto la parte local como la parte Web, usando diseño y programación orientada a objetos, el diseño de bases de datos relacionales y el diseño de la interfaz de usuario.

Construir una aplicación de software utilizando tecnologías Microsoft, en ambiente local y otra en ambiente Web. Las mismas deben incluir casos de uso, análisis, diseño de base de datos y diseño de la interfaz de usuario, con el objetivo de brindarle al usuario final una aplicación transaccional.

Emplear estándares y técnicas en cada una de las etapas y actividades del proyecto, para asegurar la calidad del software.

Utilizar herramientas para la gestión de la configuración y de esta manera asegurar el manejo de versiones del código fuente y documentos.

Emplear técnicas para la gestión del riesgo de las diferentes actividades que se llevan a cabo durante la elaboración de la aplicación, para poder minimizar, y controlar cualquier situación que sea ajena a lo planificado.

Demostrar buenas prácticas de redacción de documentos técnicos y de exposición de trabajos realizados en equipo, para facilitar la comprensión de los temas y logren comunicar de una manera sencilla y concisa lo que se desea transmitir.

Organizar el trabajo individual y el trabajo en grupo, identificándose con el equipo de trabajo, comprometiéndose a cumplir el plan del proyecto y proponiendo, comentando y creando soluciones en conjunto para garantizar la finalización exitosa del proyecto.

Interactuar con los compañeros del equipo de trabajo, colaborando para que haya una buena comunicación en el equipo, resolviendo los posibles conflictos y fomentando la comprensión a los demás, para garantizar la finalización exitosa del proyecto.

CONTENIDOS

Tema 1. ASPECTOS BÁSICOS DE INGENIERÍA DE REQUERIMIENTOS

- Aspectos generales.
- Educción de requerimientos.

Tema 2. GESTIÓN DE LA CONFIGURACIÓN

- Elementos de configuración.
- Gestión de versiones.
- Herramienta para la gestión de la configuración.

Tema 3. GESTIÓN DE RIESGOS DEL PROYECTO

- Definición de riesgo.
- Técnicas para gestionar el riesgo.

Tema 4. PLANEACIÓN DEL PROYECTO

- Definición de tareas.
- Establecimiento de recursos.

Tema 5. LENGUAJE DE MODELADO UNIFICADO (UML)

- Modelo conceptual.
- Contratos de operaciones.
- Diagramas de secuencia.
- Diagramas de estado.
- Diagramas de colaboración.
- Diagrama de clases de diseño.

Tema 6. ARQUITECTURAS DE SOFTWARE

- Diferentes opciones de arquitecturas.
- Conceptos y desarrollo de aplicaciones en múltiples capas.

Tema 7. LENGUAJE DE PROGRAMACIÓN VISUAL BASIC EN PLATAFORMA .NET (VB.NET)

- Controles.
- Estructuras de Control.
- Modificadores de Alcances.
- Clases y Objetos.
- Excepciones.
- Componente de reportes (Crystal Reports).
- Servicio de reportes (Reporting Services).
- Componente de acceso y servicio a datos

Tema 8. PÁGINAS ACTIVAS DE SERVIDOR (ASP.NET)

- Generalidades.
- Controles Web.
- Manejo del estado.
- Controles de usuario.
- Manejo de errores.
- Seguridad.
- Manejo de memoria

Tema 9. DISEÑO DE INTERFACES DE USUARIO

- Diferentes opciones de interacción usuario – sistema.
- Creación de aplicaciones fáciles de usar.

Tema 10. TÉCNICAS DE COMUNICACIÓN ESCRITA

- Informes técnicos.
- Las partes de un informe técnico.
- Errores comunes en la escritura.
- Redacción sin errores.

- Redacción de objetivos.
- Investigación de temas.

Tema 11. TÉCNICAS DE COMUNICACIÓN ORAL, PRESENTACIONES EN PÚBLICO

- Barreras de la comunicación
- La entrevista.
- Errores de expresión oral.
- Técnicas de la exposición oral.
- Manejo de la voz.
- Manejo de las técnicas de presentación.
- Manejo del desplazamiento.
- Manejo de la expresión corporal.
- Materiales de apoyo.
- Conocimiento del público.

Tema 12. TÉCNICAS PARA EL MANEJO DE REUNIONES

- Técnicas avanzadas del manejo de reuniones.
- Seguimiento.

Tema 13. TECNICAS PARA DESARROLLAR COMPETENCIAS DE EFECTIVIDAD SOCIAL

- Relaciones interpersonales
- Tolerancia al estrés
- Proceso para la toma de decisiones

METODOLOGÍA

- El trabajo en clase se distribuye en sesiones teóricas y sesiones prácticas. Las sesiones teóricas consisten en exposiciones magistrales por parte del profesor seguidas por interrogatorios didácticos a los estudiantes en los que el profesor tiene la oportunidad de aclarar dudas y fomentar la participación de los estudiantes. El profesor también compara y contrasta diferentes ejemplos en cada tema para ayudar a aclarar los conocimientos y facilitar su comprensión.

- En las sesiones prácticas se hacen laboratorios en computadora, individuales, que giran alrededor de asignaciones fijadas por el profesor, diseñadas también para reforzar los conocimientos y aplicarlos. Durante los laboratorios, los estudiantes trabajan bajo la supervisión del profesor, teniéndolo a su disposición para ser consultado cuando se requiera. También se realiza trabajo en grupo, que se genera de las asignaciones que el profesor le hace a cada equipo de estudiantes.

ESTRATEGIAS DE APRENDIZAJE

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

RECURSOS DIDÁCTICOS

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

EVALUACIÓN

Concepto	Ponderación
Tareas	5%
Laboratorios	10%
Exámenes	30%
Documentación del proyecto	20%
Presentación del proyecto	25%
Avances del proyecto (2)	10%

BIBLIOGRAFÍA

Referencias principales:

- Booch, G.; Rumbaugh, J.; Jacobson, I. (2017). *The Unified Modeling Language User Guide (Second Edition)*. Addison Wesley.
- Jacobson, I.; Booch, G.; Rumbaugh, J. (1999) [1].*The Unified Software Development Process*. Addison Wesley.
- Platt, D. (2003) [2]. *Introducen Microsoft .NET*. Microsoft Press.

Referencias complementarias:

- Thai, T.; Lam, H. (2003)[3]. *.NET Framework Essentials*. O'Reilly & Associates.

Programa	Diplomado en Desarrollo de Software
Curso	DDS- 501 Fundamentos de Sistemas Operativos
Créditos	4
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	V
Periodo	Cuatrimestral
Requisitos	DDS- 402 Programación III

DESCRIPCIÓN DEL CURSO

Este curso brinda al estudiante los conocimientos básicos respecto a la estructura interna y el funcionamiento de los sistemas operativos modernos en sus cuatro módulos principales: procesos, memoria, entrada-salida y sistemas de archivos, utilizando Linux como sistema operativo referencial para las prácticas de laboratorio, a fin de que el estudiante se preparare para la implementación de una solución integral de tecnologías de información a nivel empresarial.

OBJETIVO GENERA

Aplicar los conceptos básicos del diseño de sistemas operativos resolviendo problemas de sincronización y elaborando programas de manejo y administración de sistemas operativos, para implementar soluciones integrales de tecnologías de información.

OBJETIVOS ESPECÍFICOS

- Describir la historia de los sistemas operativos más representativos, revisando los sistemas monolíticos, con capas, cliente-servidor y de máquina virtual, para comprender la influencia en los sistemas operativos actuales.
- Comprender el funcionamiento de los principales módulos de un sistema operativo, examinando los procesos, la memoria, la entrada-salida y los sistemas de archivos,

a fin de distinguir las características más relevantes de diseño y estructura de los sistemas operativos distribuidos y no distribuidos.

- Elaborar programas de computadora para el manejo y la administración del sistema operativo, utilizando el paradigma de programación estructurada y usando el lenguaje de programación a nivel del intérprete de comandos.
- Resolver problemas de control de procesos de sistemas operativos, utilizando estructuras de datos y algoritmos adecuados, a fin de sincronizar las funciones más importantes.

CONTENIDOS

Tema 1. INTRODUCCIÓN

- Conceptos generales (procesos, archivos, llamadas al sistema, componentes del sistema).
- El sistema operativo como una máquina virtual.
- Estructura de los sistemas operativos.
- Sistemas monolíticos.
- Sistemas con capas.
- Sistemas cliente-servidor.
- Sistemas de máquina virtual.

Tema 2. HISTORIA DE LOS SISTEMAS OPERATIVOS

- Historia de los sistemas operativos.
- Principales sistemas operativos para dispositivos digitales personales (PDAs, por sus siglas en inglés).

Tema 3. GESTIÓN DE LOS PROCESOS

- Introducción a los procesos.
- Sincronización de procesos.
- Planificación de procesos.

- Línea de comandos.

Tema 4. GESTIÓN DE LA MEMORIA

- Asignación contigua.
- Paginación.
- Segmentación.
- Memoria Virtual.

Tema 5. SISTEMAS DE ARCHIVOS

- Archivos y directorios.
- Estructura de los directorios.
- Métodos de acceso.
- Seguridad.
- Sistemas de archivos más populares.

Tema 6. SISTEMAS OPERATIVOS DISTRIBUIDOS

- Introducción: conceptos básicos de redes de computadoras.
- Comunicación.
- Sistemas de archivos distribuidos.
- Transacciones, sincronización y coordinación.
- Hilos (´*threads*´, en inglés).

Tema 7. ALGUNAS TENDENCIAS DE LOS SISTEMAS OPERATIVOS

- Sistemas operativos de tiempo real.
- Sistemas operativos orientados a arquitecturas de multiprocesadores.
- Sistemas operativos orientados a objetos

METODOLOGÍA

- El trabajo en clase se distribuye en sesiones teóricas y sesiones prácticas. Las sesiones teóricas consisten en exposiciones magistrales por parte del profesor seguidas por interrogatorios didácticos a los estudiantes en los que el profesor tiene la oportunidad de aclarar dudas y fomentar la participación de los estudiantes. El profesor también compara y contrasta diferentes ejemplos en cada tema para ayudar a aclarar los conocimientos y facilitar su comprensión.
- En las sesiones prácticas se hacen laboratorios en computadora, individuales, que giran alrededor de asignaciones fijadas por el profesor, diseñadas también para reforzar los conocimientos y aplicarlos. Durante los laboratorios, los estudiantes trabajan bajo la supervisión del profesor, teniéndolo a su disposición para ser consultado cuando se requiera. También se realiza trabajo en grupo, que se genera de las asignaciones que el profesor le hace a cada equipo de estudiantes.

ESTRATEGIAS DE APRENDIZAJE

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

RECURSOS DIDÁCTICOS

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los

casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

EVALUACIÓN

Concepto	Ponderación
Interrogatorios didácticos	3%
Prácticas en clase	7%
Laboratorios en clase	20%
Tareas extra-clase	10%
Proyecto	20%
Examen Parcial	20%
Examen Final	20%

BIBLIOGRAFÍA

Referencias principales:

- Tanenbaum, A. (2015). *Sistemas Operativos Modernos*. Prentice-Hall.

Referencias complementarias:

- Stallings W. (2006) [1]. *Sistemas Operativos, Quinta Edición*. Prentice Hall, España.

Programa	Diplomado en Desarrollo de Software
Curso	DDS- 502 Diseño y Construcción de Componentes
Créditos	4
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	V
Periodo	Cuatrimestral
Requisitos	DDS- 402 Programación III DDS- 404 Proyecto de Software II

DESCRIPCIÓN

Este curso prepara y entrena a los estudiantes en los principios, técnicas, procesos y tecnologías que se aplican para construir componentes de software y ensamblar aplicaciones de software, a partir de componentes existentes (trozos de código en lenguaje de programación, que funcionan de manera independiente, cumpliendo un objetivo.

OBJETIVO GENERAL

Construir aplicaciones y sistemas de software medianos, aplicando principios, técnicas, procesos y tecnologías de diseño, construcción e implementación de componentes, a fin de incorporar al proceso de desarrollo de software los componentes para garantizar la calidad de las aplicaciones y los sistemas.

OBJETIVOS ESPECÍFICOS

Comprender el concepto de componente de software y el paradigma de desarrollo de aplicaciones de software por componentes, por medio de ejemplos y ejercicios en clase, a fin de que el estudiante pueda usar los componentes para construir sus aplicaciones de software.

Diseñar componentes de software aplicando técnicas de identificación, especificación, diseño, construcción, utilización y reutilización, para poder elaborar componentes de calidad.

Elaborar aplicaciones y sistemas de software basados en componentes, usando tecnologías y modelos de componentes, a fin de resolver problemas medianos de forma computacional.

Utilizar los diferentes modelos de componentes más utilizados en la industria, comprendiendo los principios, técnicas, procesos y tecnologías usadas en los modelos, para construir aplicaciones de software que los tengan en cuenta.

CONTENIDOS

Tema 1. EL CONCEPTO DE COMPONENTE

- Definición de componente.
- Evolución de los componentes.
- Separación de responsabilidades.

Tema 2. DISEÑO DE COMPONENTES

- Principios y técnicas para construir componentes.
- Diseños orientados a objetos y estructurados.
- Publicación de interfaces.
- Granularidad.

Tema 3. EL PROCESO DE DESARROLLO DE SOFTWARE CON COMPONENTES

- Proceso (requerimientos, especificación, adquisición, diseño, construcción, integración y pruebas).
- Diagrama de componentes.
- Diagrama de distribución.

Tema 4. LOS CONTROLES COMO COMPONENTES

- El control vs. El ambiente de desarrollo.
- Controles.

Tema 5. EL CONCEPTO DE PLATAFORMAS DE COMPONENTES

- Estándares de “alambrado”.
- Protocolos de computación distribuida.

Tema 6. MODELOS DE COMPONENTES

- Tecnologías y modelos para implementar componentes.

METODOLOGÍA

- El trabajo en clase se distribuye en sesiones teóricas y sesiones prácticas. Las sesiones teóricas consisten en exposiciones magistrales por parte del profesor seguidas por interrogatorios didácticos a los estudiantes en los que el profesor tiene la oportunidad de aclarar dudas y fomentar la participación de los estudiantes. El profesor también compara y contrasta diferentes ejemplos en cada tema para ayudar a aclarar los conocimientos y facilitar su comprensión.
- En las sesiones prácticas se hacen laboratorios en computadora, individuales, que giran alrededor de asignaciones fijadas por el profesor, diseñadas también para reforzar los conocimientos y aplicarlos. Durante los laboratorios, los estudiantes trabajan bajo la supervisión del profesor, teniéndolo a su disposición para ser consultado cuando se requiera. También

se realiza trabajo en grupo, que se genera de las asignaciones que el profesor le hace a cada equipo de estudiantes.

ESTRATEGIAS DE APRENDIZAJE

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

RECURSOS DIDÁCTICOS

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

EVALUACIÓN

Concepto	**Ponderación**
Interrogatorios didácticos	5%
Laboratorios en clase	20%
Tareas extra clase (en grupo)	30%
Examen 1	15%

Examen 2	15%
Examen 3	15%
Total	**100%**

BIBLIOGRAFÍA

Bibliografía clásica

- Anderson G; Anderson, P. (2002) *Enterprise JavaBeans Component Architecture: Designing and Coding Enterprise Applications*. Prentice Hall.

Bibliografía complementaria

- Arrington CT. (2001). Enterprise *Java with UML*. Wiley.
- Patel Sriganesh R.; Brose G.; Silverman M. (2006). *Mastering Enterprise JavaBeans 3.0.* Wiley.

Programa Diplomado en Desarrollo de Software

Curso DDS- 503 Programación IV

Créditos 4

Modalidad Presencial

Naturaleza Teórico-práctico

Ciclo V

Periodo Cuatrimestral

Requisitos DDS- 402 Programación IV

DESCRIPCIÓN DEL CURSO

Este curso introduce al estudiante los principios fundamentales para crear diseños de aplicaciones de software, como lo es la cohesión, el acoplamiento y el uso de interfaces, entre otros, a fin de que el estudiante aprenda a diferenciar, usar y seleccionar los patrones de diseño que conducen a crear aplicaciones de software de calidad.

Objetivo general

Construir aplicaciones de software medianas y de calidad, seleccionando principios y técnicas de patrones de diseño apropiados, a fin de garantizar que las aplicaciones tengan buen rendimiento, sean escalables, sean fáciles de mantener y sean interoperables.

Objetivos específicos

- Comprender los principios fundamentales y los beneficios de aplicar patrones de diseño, mediante prácticas en clase y laboratorios, para que el estudiante pueda incorporar en sus diseños de aplicaciones de software, los patrones de diseño.
- Diferenciar patrones de diseño de acuerdo con un conjunto de atributos de calidad, por medio de ejemplos y ejercicios, a fin de poder escoger los patrones de diseño apropiados para resolver cada problema.
- Crear diseños de aplicaciones de software teniendo en cuenta principios y técnicas de calidad, incorporando experiencias de diseños previos en contextos similares, para hacer diseños que sean fáciles de corregir y mantener.
- Elaborar aplicaciones medianas de software, basándose en diseños de patrones, para construir aplicaciones de software de calidad.

Contenidos

Tema 1: Introducción

- Concepto de diseño.
- Herramientas para la automatización de diseños.
- Ejemplos de diseños inadecuados.
- Elementos para mejorar los diseños- reorganización y patrones

Tema 2: Principios de diseño

- Abstracción.
- Encapsulamiento.
- Modularidad.
- Cohesión y Acoplamiento.

- Herencia vs. Composición
- Reutilización.
- Diseño con interfaces.
- Separación de intereses y responsabilidades.

Tema 3: Atributos de calidad en el diseño

- Rendimiento y escalabilidad.
- Seguridad.
- Tolerancia a fallas.
- Capacidad de Mantenimiento.
- Portabilidad e interoperabilidad.
- Capacidad de Utilización.

Tema 4: Patrones

- Origen de los patrones de diseño.
- Formatos de especificación de patrones.
- Apoyo de los lenguajes al uso de los patrones.
- Herramientas de apoyo al uso de patrones.
- Patrones no recomendados.

Tema 5: Categorías de patrones de diseño

- Patrones de Creación.
- Patrones de Estructura.
- Patrones de Comportamiento.
- Reorganización de código hacia patrones.

Tema 6: Pruebas de unidad y refactorización

- Aislamiento y detección de errores.

- Herramientas de prueba de unidad aplicadas a patrones.
- Prueba de patrones.
- Patrones de Prueba.

Tema 7: Marcos de trabajo

- Definición e Importancia.
- Ejemplos.

METODOLOGÍA

- El trabajo en clase se distribuye en sesiones teóricas y sesiones prácticas. Las sesiones teóricas consisten en exposiciones magistrales por parte del profesor seguidas por interrogatorios didácticos a los estudiantes en los que el profesor tiene la oportunidad de aclarar dudas y fomentar la participación de los estudiantes. El profesor también compara y contrasta diferentes ejemplos en cada tema para ayudar a aclarar los conocimientos y facilitar su comprensión.
- En las sesiones prácticas se hacen laboratorios en computadora, individuales, que giran alrededor de asignaciones fijadas por el profesor, diseñadas también para reforzar los conocimientos y aplicarlos. Durante los laboratorios, los estudiantes trabajan bajo la supervisión del profesor, teniéndolo a su disposición para ser consultado cuando se requiera. También se realiza trabajo en grupo, que se genera de las asignaciones que el profesor le hace a cada equipo de estudiantes.

ESTRATEGIAS DE APRENDIZAJE

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas

en clase, tareas extra clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

RECURSOS DIDÁCTICOS

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

Evaluación

Concepto	Ponderación
Examen 1	15%
Examen 2	15%
Exposición oral	20%
Interrogatorios didácticos	5%
Laboratorios en clase	20%
Proyecto	25%
Total	100%

Bibliografía Clásica

Referencias principales:

· Larman, C. (2004) [1]. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development (3rd Edition). Prentice-Hall.

· Gamma, E.; Helm R.; Johnson R.; Vlissides J. (2005) [2]. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison Wesley.

Referencias complementarias:

· Fowler, M. (2018). Refactoring; improving the design of existing code. Addison-Wesley.

Programa	Diplomado en Desarrollo de Software
Curso	RHE-504 Relaciones Humanas y Comunicación eficaz
Créditos	3
Modalidad	Presencial
Naturaleza	Teórico-práctico
Ciclo	V
Periodo	Cuatrimestral
Requisitos	NT

Descripción del curso

Este curso proporciona los elementos, técnicas y métodos para establecer una adecuada comunicación. Además, proporcionará al estudiante las herramientas y técnicas para lograr una buena relación interpersonales y como llevarse bien con las personas.

Objetivo general:

Reconocer la importancia de aplicar diversas técnicas en el manejo y comunicación con las personas

Objetivos específicos:

- Conocer la importancia de aplicar técnicas adecuadas, para lograr una mejor comunicación con las personas que asisten a un laboratorio.
- Desarrollar en el estudiante la sensibilidad para comprender y aceptar las diversas reacciones de los pacientes.
- Comprender la necesidad de desarrollar la empatía como un medio para relacionarse con el paciente y su familia.
- Conocer diversas técnicas y métodos sobre servicio al cliente.

Contenidos:

- Aumentar las habilidades de comunicación
- ¿Cómo eliminar la crítica y el mal genio?
- Obtener una mejor colaboración de los demás
- Estandarizar procesos de comunicación entre departamentos
- ¿Cómo cambiar personas negativas a positivas?
- ¿Cómo ganar amigos?
- ¿Cómo recordar nombres?
- ¿Cómo trabajar en equipo y mejorar el ambiente laboral

Metodología

- El trabajo en clase se distribuye en sesiones teóricas y sesiones prácticas. Las sesiones teóricas consisten en exposiciones magistrales por parte del profesor seguidas por interrogatorios didácticos a los estudiantes en los que el profesor tiene la oportunidad de aclarar dudas y fomentar la participación de los estudiantes. El profesor también compara y contrasta diferentes ejemplos en cada tema para ayudar a aclarar los conocimientos y facilitar su comprensión.
- En las sesiones prácticas se hacen laboratorios en computadora, individuales, que giran alrededor de asignaciones fijadas por el profesor, diseñadas también para reforzar los conocimientos y aplicarlos. Durante los laboratorios, los estudiantes trabajan bajo la supervisión del profesor, teniéndolo a su disposición para ser consultado cuando se requiera. También

se realiza trabajo en grupo, que se genera de las asignaciones que el profesor le hace a cada equipo de estudiantes.

Estrategias de Aprendizaje

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

Recursos didácticos

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

Evaluación

Prueba parcial teórico-práctica	30%
Exposiciones	20%
Examen final	25%
Tareas	15%
Participación	10%

Bibliografía

- Tschohl, John. Premier mundial del sistema de servicio a la clientela. REVISTA TURISTICA, 2012.
- Wilcox, Denis; Autt, Warren y Cameron Glen. Relaciones públicas: estrategias y tácticas. X edición, Editorial Pearson, México, 2012.
- Ovejero Bernal, Anastasio. Técnicas de negociación. España: Mac Graw Hill, 2011.

Programa	Diplomado en Desarrollo de Software
Curso	IVA 600 Investigación aplicada
Créditos	4
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestres	VI
Periodo	Cuatrimestral
Requisitos	DDS 402 Programación III

Descripción del curso

El curso de metodología de la investigación tiene como fin explicar los diferentes pasos lógicos o de sentido común que cualquier investigador debe seguir, con el fin de determinar, con cierto grado de confianza, que las conclusiones a que llegara en su proceso investigativo reúnen las condiciones necesarias para explicar el fenómeno que pretende describir.

Dado que en el área de la tecnología se fundamenta muchos casos, del análisis de observaciones y experiencias propias por ejemplo de la experiencia siniestral de la empresa, en diferentes riesgos, la forma en que se registra la información proveniente de la experiencia (propia o ajena), la forma en que se analiza, la forma en que se presenta y la forma en que los resultados respaldan las decisiones se fundamenta en la aplicación de ese método científico.

Objetivo General

Proveer al estudiante de los elementos fundamentales del proceso de investigación, a saber: formulación del problema, elaboración de objetivos e hipótesis, identificación de las variables, recolección de datos, análisis e interpretación de los datos y redacción del informe y aplicarlos en la elaboración de proyectos investigativos.

Objetivos Específicos

- Describir el concepto de investigación científica y comparar las relaciones entre la teoría, el método y las técnicas en el proceso de investigación.
- Practicar mediante la utilización de casos concretos, las semejanzas y diferencias entre el método cuantitativo y el método cualitativo.
- Explicar los procesos asociados a las etapas del proceso de investigación, a saber: formulación del problema de investigación, recolección, análisis e interpretación de los datos.
- Aplicar los conceptos anteriores, a un caso concreto de investigación.

Contenidos:

1. Teoría, método y técnicas de investigación.

   · La teoría, los métodos y las técnicas de investigación científicos.
   · La objetividad de la ciencia social.
   · Método cualitativo y método cuantitativo: características, semejanzas y diferencias.

2. El proyecto de investigación.

   · Formulación del problema, definición del marco teórico, tipos de estudio.
   · Objetivos, hipótesis y variables.
   · Procedimientos metodológicos: conceptos de población y muestra, recolección de datos, definición y elaboración de los instrumentos para la recolección de datos.

- Confiabilidad y validez.
- El plan de trabajo y las técnicas de investigación según el método de investigación elegido.

3. Aplicación concreta de un proyecto de investigación.

- Recolección, análisis e interpretación de la información obtenida, por medio de la aplicación de un proyecto de investigación.
- Presentación de los resultados: conclusiones, sugerencias y recomendaciones finales.

Metodología

- El trabajo en clase se distribuye en sesiones teóricas y sesiones prácticas. Las sesiones teóricas consisten en exposiciones magistrales por parte del profesor seguidas por interrogatorios didácticos a los estudiantes en los que el profesor tiene la oportunidad de aclarar dudas y fomentar la participación de los estudiantes. El profesor también compara y contrasta diferentes ejemplos en cada tema para ayudar a aclarar los conocimientos y facilitar su comprensión.
- En las sesiones prácticas se hacen laboratorios en computadora, individuales, que giran alrededor de asignaciones fijadas por el profesor, diseñadas también para reforzar los conocimientos y aplicarlos. Durante los laboratorios, los estudiantes trabajan bajo la supervisión del profesor, teniéndolo a su disposición para ser consultado cuando se requiera. También se realiza trabajo en grupo, que se genera de las asignaciones que el profesor le hace a cada equipo de estudiantes.

Estrategias de aprendizaje

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

Recursos didácticos

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

Evaluación

Exámenes parciales (2):	40%
Prácticas	10%
Trabajo de Investigación	30%
Quices y asignaciones	20%

Bibliografía bibliográfica

- Bernal Torres C, **Metodología de la Investigación**, Quinta Edición, Editorial Pearson Educación, México, 2012.
- Davis Duane, **Investigación en Administración para la toma de decisiones**, Sétima Edición, Editorial Thomson International, 2013.
- Gibson James L, **Organizaciones**, Duodécima Edición, Editorial Mc Graw Hill, 2014.
- **Investigaciones en seguros y gestión de riesgos.** Fundación Mapfre (España). Año de edición**:** 2014

Programa	Diplomado en Desarrollo de Software
Curso	DDS- 602 Estructuras de Datos
Créditos	4
Modalidad	Presencial
Naturaleza	Teórico-práctico
Cuatrimestre	VI
Periodo	Cuatrimestral
Requisitos	DDS-402 Programación III

DESCRIPCIÓN

Este curso introduce al estudiante en el estudio y uso de las estructuras de datos lineales y jerárquicos, su análisis, diseño, especificación e implementación con tipos abstractos de datos; a fin de que el estudiante seleccione y use estructuras de datos que constituyan la base de aplicaciones de software eficientes en tiempo y en recursos.

Objetivo General

Construir aplicaciones de software pequeñas y medianas, usando las estructuras de datos lineales, las estructuras de datos jerárquicas y los algoritmos de ordenamiento y búsqueda, para implementar aplicaciones de software más eficientes en tiempo y en recursos de computadora.

Objetivos Específicos

- Comprender el funcionamiento y la algorítmica de las estructuras de datos lineales y jerárquicos y de los algoritmos de ordenamiento y búsqueda, mediante el estudio de ejemplos y prácticas en clase, a fin de que el estudiante pueda conocer y distinguir las diferentes estructuras de datos.

- Usar las estructuras de datos lineales y jerárquicos, y los algoritmos de ordenamiento y búsqueda, mediante el uso de Tipos Abstractos de Datos y de lenguajes de programación, para resolver un problema con una aplicación de software pequeña o mediana.
- Analizar la eficiencia en tiempo y en espacio, de las estructuras de datos y de los algoritmos, usando la notación O, a fin de aplicar este criterio y escoger las estructuras y los algoritmos de menor complejidad.
- Diseñar estructuras de datos lineales y jerárquicos por medio de Tipos Abstractos de Datos, para construir aplicaciones de software fáciles de entender, corregir y cambiar.
- Elaborar aplicaciones de software eficientes, implementando estructuras de datos lineales y jerárquicos, y algoritmos de búsqueda y ordenamiento, que garanticen el menor uso de recursos y de tiempo.

Contenidos

Tema 1. Abstracción de datos

- Abstracción y Encapsulamiento.
- Tipos abstractos de datos.
- Diseño y especificación de los tipos abstractos de datos.
- Implementación.

Tema 2. Análisis de algoritmos

- Justificación.
- Comportamiento de los algoritmos.
- Análisis del peor de los casos y del caso promedio.
- Notación O.
- Reglas para realizar el análisis de algoritmos.

- Otros criterios de selección de estructuras de datos y algoritmos

Tema 3. Estructura Lineal: Listas

- Funcionamiento de las listas.
- Lista circular y Lista doblemente encadenada.
- Diseño y especificación con tipos abstractos de datos.
- Análisis.
- Implementación.

Ttema 4. Estructura Lineal: Pilas

- Funcionamiento de las pilas.
- Diseño y especificación con tipos abstractos de datos.
- Análisis.
- Implementación.

Tema 5. Estructura Lineal: Colas

- Funcionamiento de las colas.
- Cola de prioridad y bicola.
- Diseño y especificación con tipos abstractos de datos.
- Análisis.
- Implementación.

Tema 6. Algoritmos de búsqueda y ordenamiento

Tema 7. ESTRUCTURAS JERÁRQUICAS: ÁRBOLES

- Recursividad y algoritmos de retroceso
- Funcionamiento de los árboles.
- Árboles de Expresión: definición, construcción y procesamiento.
- Árboles balanceados AVL: definición y funcionamiento.
- Árboles n-arios.
- Análisis.
- Implementación.

Metodología

- El trabajo en clase se distribuye en sesiones teóricas y sesiones prácticas. Las sesiones teóricas consisten en exposiciones magistrales por parte del profesor seguidas por interrogatorios didácticos a los estudiantes en los que el profesor tiene la oportunidad de aclarar dudas y fomentar la participación de los estudiantes. El profesor también compara y contrasta diferentes ejemplos en cada tema para ayudar a aclarar los conocimientos y facilitar su comprensión.
- En las sesiones prácticas se hacen laboratorios en computadora, individuales, que giran alrededor de asignaciones fijadas por el profesor, diseñadas también para reforzar los conocimientos y aplicarlos. Durante los laboratorios, los estudiantes trabajan bajo la supervisión del profesor, teniéndolo a su disposición para ser consultado cuando se requiera. También se realiza trabajo en grupo, que se genera de las asignaciones que el profesor le hace a cada equipo de estudiantes.

Estrategias de aprendizaje

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

Recursos Didácticos

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

Evaluación

Concepto	Ponderación
Interrogatorios didácticos	5%
Prácticas en clase	15%
Tareas en grupo	35%
Examen 1	15%

Examen 2	15%
Examen 3	15%
Total	100%

BIBLIOGRAFÍA

Bibliografía Clásica

- Aho, A. (1999). *Estructura de Datos y Algoritmos*. Addison Wesley Longman.

- *Joyanes L. (2008). FUNDAMENTOS DE PROGRAMACIÓN Algoritmos, estructura de datos y objetos. McGraw-Hill*

Bibliografía complementaria

- Main, M., Savitch, W. (2010). *Data Structures and Other Objects using C++* .Addison Wesley.

Programa	Diplomado en Desarrollo de Software
Curso	SOFT-22 Proyecto de Software III
Créditos	4
Modalidad	Presencial
Naturaleza	Teórico-práctico
Ciclo	VI
Periodo	Cuatrimestral
Requisitos	DDS-303 Proyecto de Software II, DDS-403 Programación de Bases de Datos

Descripción

Este curso profundiza al estudiante en las disciplinas utilizadas para la producción de software, usando herramientas y plataformas abiertas, un proceso de desarrollo de software iterativo-evolutivo, técnicas para expresión escrita y oral, y técnicas de trabajo en equipo, para la construcción de una aplicación de software de calidad basada en Web para un cliente o usuario real.

Al inicio de este curso, los mismos estudiantes conforman los equipos de trabajo de cinco a seis integrantes, aprovechando que ellos han tenido la experiencia de los dos cursos de Proyecto de Ingeniería del Software anteriores. Cada equipo de trabajo simula ser una empresa de desarrollo de software, en donde cada miembro cumple un rol específico. El equipo debe construir en el periodo académico, una aplicación de software de calidad basada en tecnologías Web abiertas y en una arquitectura de múltiples capas, para un cliente o usuario externo y real. Por lo tanto, el equipo debe cumplir con las actividades de las etapas del proceso de desarrollo de software iterativo-evolutivo, que son: requerimientos, análisis, diseño, construcción y validación. Al finalizar el proceso de desarrollo, el equipo debe entregarle al cliente o usuario real la aplicación construida, que debe satisfacer las necesidades planteadas.

En este curso es importante que el estudiante reúna las habilidades técnicas y de trabajo en equipo, que le permitan un desempeño exitoso en el siguiente curso llamado Proyecto en Empresa, en donde el estudiante debe participar en un proyecto de desarrollo de software en una empresa.

OBJETIVO GENERAL

Construir una aplicación Web interactiva y de calidad para un cliente real (externo a la institución), basada en una arquitectura de múltiples capas y en plataformas tecnológicas libres, como el lenguaje de programación Java™ y el sistema manejador de base de datos MySQL; aplicando y siguiendo un proceso de desarrollo de software iterativo-evolutivo, usando características de administración de proyectos de software, y aplicando técnicas de comunicación escrita y oral y de trabajo en equipo, con el fin de elaborar un programa eficiente, bien documentado y de calidad, que cumpla con los requisitos solicitados por el cliente.

Objetivos Específicos

- Aplicar el modelo de ciclo de vida iterativo-evolutivo durante la construcción de la aplicación a un cliente real, mediante el cumplimiento de las etapas de cada uno de los ciclos, con el objetivo de mejorar el programa que se construye.
- Elaborar un plan de trabajo que defina el papel que debe cumplir cada miembro del equipo, y que permita la realización de las actividades correspondientes a las etapas del proceso de desarrollo de software (requerimientos, análisis, diseño, construcción y validación).
- ·Emplear la notación Lenguaje de Modelado Unificado (UML, por sus siglas en inglés), mediante una herramienta para la diagramación, con la finalidad de realizar el análisis y diseño conceptual de un sistema de software Web.
- Aplicar una arquitectura de múltiples capas para la construcción de la aplicación de software, siguiendo los conceptos de la arquitectura cliente-servidor para plataformas de Internet abiertas.

- Diseñar la aplicación Web para que satisfaga las necesidades del usuario, utilizando el diseño y programación de componentes de software, el diseño de bases de datos y el diseño de la navegabilidad en una aplicación Web
- Construir la aplicación Web interactiva, basada en las especificaciones definidas para tal fin, en el lenguaje de programación Java™ y usando el manejador de base de datos MySQL.
- Emplear estándares y técnicas en cada una de las etapas y actividades del proyecto, para asegurar la calidad del software.
- Comunicar de manera clara y efectiva, en forma oral y escrita, las características principales de la aplicación de software que se construya, para facilitar la comprensión de los temas y logren comunicar de una manera sencilla y concisa lo que se desea transmitir.
- Reconocer las propias habilidades técnicas y de trabajo en equipo, para comunicarlas de manera escrita y oral, como preparación para el trabajo en una empresa.
- Comprometerse con el trabajo en equipo para cumplir con todas las actividades asignadas y para relacionarse de manera efectiva con los compañeros y con los superiores, con el objetivo de garantizar la finalización exitosa del proyecto.

Contenidos

Tema 1. Metodología De Enseñanza Del Proceso De Desarrollo De Software

- Experiencias vividas en los cursos de Proyecto de Ingeniería del Software anteriores.
- Proceso de desarrollo de software iterativo-evolutivo.

Tema 2. Administración De Proyectos De Software

- Administración de Calidad.
- Administración de Riesgos.

- Administración de la Configuración.
- Definición de actividades.
- Secuenciación y estimación de la duración de actividades.
- Desarrollo y control de cronogramas.
- Herramientas para el manejo del tiempo: Diagramas.

Tema 3. Teoría De Estimación

- Técnicas para estimación de proyectos
- Ruta crítica

Tema 4. Atributos De Calidad: Especificación, Diseño Y Ejecución De Pruebas De Requerimientos No Funcionales

- Rendimiento.
- Consistencia.
- Seguridad.
- Otros atributos: completitud, protección, integridad, facilidad de uso y capacidad de ser comprobable.

Tema 5. Diseño De La Arquitectura Del Software

- Repaso de Casos de Uso.
- Repaso de Modelo Conceptual.
- Repaso de Contratos de Operación.
- Repaso de Diagramas de Interacción.
- Repaso de Diagrama Entidad – Relación.
- Repaso de Diagrama de Clases.

Tema 6. Entorno De Programación

- Características.
- Instalación.
- Entorno.

Tema 7. Marcos De Trabajo Para Aplicaciones Web

- Marco de trabajo para aplicaciones basadas en Web
- Marco de trabajo del Modelo-Vista-Controlador (Struts).
- Documentación interna de las aplicaciones de software.

Tema 12. Servidor De Aplicaciones

- Características.
- Instalación del servidor.
- Administración del servidor de aplicaciones.
- Comandos principales.
- Seguridad básica.

Tema 13. Nuevas Técnicas De Comunicación

- Comunicación escrita.
- Capacitación a usuarios.
- Presentaciones creativas.
- Carta de presentación.
- Currículum Vitae.
- Entrevista de Trabajo.

Tema 14. El Ser Humano En El Contexto De Las Empresas De Tecnologías De Información

- Bienestar social y emocional
- Negociación.

- Servicio al cliente.
- Innovación y creatividad

Metodología

- El trabajo en clase se distribuye en sesiones teóricas y sesiones prácticas. Las sesiones teóricas consisten en exposiciones magistrales por parte del profesor seguidas por interrogatorios didácticos a los estudiantes en los que el profesor tiene la oportunidad de aclarar dudas y fomentar la participación de los estudiantes. El profesor también compara y contrasta diferentes ejemplos en cada tema para ayudar a aclarar los conocimientos y facilitar su comprensión.
- En las sesiones prácticas se hacen laboratorios en computadora, individuales, que giran alrededor de asignaciones fijadas por el profesor, diseñadas también para reforzar los conocimientos y aplicarlos. Durante los laboratorios, los estudiantes trabajan bajo la supervisión del profesor, teniéndolo a su disposición para ser consultado cuando se requiera. También se realiza trabajo en grupo, que se genera de las asignaciones que el profesor le hace a cada equipo de estudiantes.

Estrategias de aprendizaje

Por mencionar algunas de las estrategias de aprendizaje que el estudiante debe adoptar para lograr cumplir los objetivos del curso podemos mencionar las prácticas en clase, tareas extra clase, investigaciones, exposiciones y algunas lecturas que enriquezcan el curso.

Recursos Didácticos

Uso del proyector digital, que contienen los conceptos correspondientes a cada sesión, un laboratorio de computadoras debidamente instalado y configurado para realizar los laboratorios en clase, con una aplicación que permite ejecutar algoritmos

expresados en diagramas de flujo y con el ambiente de programación (el compilador y un editor para los programas), recursos que están disponibles en Internet, tales como estándares de codificación o plantillas para las listas de revisión y ejecutar los casos de prueba, así como el uso de los libros que están en la biblioteca para consulta de los estudiantes.

relacionados con la metodología del proceso de desarrollo de software usado en el curso.

Evaluación

Concepto	**Ponderación**
Interrogatorios didácticos	10%
Exámenes	20%
Documentación del proyecto	30%
Presentación del proyecto	30%
Auto Evaluación	10%
Total	**100%**

BIBLIOGRAFÍA

Bibliografía obligatoria

- Pilone, T; Miles, R. (2008) [1]. *Head First Software Development*. O´Reilly.
- Somerville, I. (2016). *Software Engineering*, *(10th Edition)*. Addison Wesley.

Bibliografía complementaria

- Gorton, I. (2006) [2]. *Essential Software Architecture*. Springer.
- Holmes, J. (2006) [3]. *Struts: The Complete Reference, Second Edition*. McGraw Hill.

Nómina de profesores