



**INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN /
INGENIERÍA EN SISTEMAS COMPUTACIONALES
EN COMPETENCIAS PROFESIONALES**



ASIGNATURA DE FUNDAMENTOS DE PROGRAMACIÓN ORIENTADA A OBJETOS

PROPÓSITO DE APRENDIZAJE DE LA ASIGNATURA	El alumno codificará algoritmos computacionales a través del paradigma de programación orientado a objetos para resolver problemas del ámbito cotidiano.				
CUATRIMESTRE	Quinto				
TOTAL DE HORAS	PRESENCIALES	NO PRESENCIALES	HORAS POR SEMANA	PRESENCIALES	NO PRESENCIALES
	90	0		6	

UNIDADES DE APRENDIZAJE	HORAS DEL SABER		HORAS DEL SABER HACER		HORAS TOTALES	
	P	NP	P	NP	P	NP
I. Fundamentos del paradigma de Programación Orientado a Objetos	6	0	12	0	18	0
II. Análisis y diseño orientado a objetos	10	0	20	0	30	0
III. Desarrollo de clases y métodos	12	0	30	0	42	0
TOTALES	28		62		90	

ELABORÓ:	Comité de Directores de la Ingeniería en Tecnologías de la Información / Ingeniería en Sistemas Computacionales	REVISÓ:	Dirección Académica
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre 2018

COMPETENCIA A LA QUE CONTRIBUYE LA ASIGNATURA

De acuerdo con la metodología de diseño curricular de la CGUTyP, las competencias se desagregan en dos niveles de desempeño: Unidades de Competencias y Capacidades.

La presente asignatura contribuye al logro de la competencia y los niveles de desagregación de los contenidos a continuación:

COMPETENCIA: Desarrollar soluciones innovadoras de integración de tecnologías de la información mediante metodologías de desarrollo de software, diseño de base de datos, seguridad de la información y administración de proyectos; con base en los estándares aplicables para atender las áreas de oportunidad, resolver las necesidades y optimizar los procesos y recursos de la organización.

UNIDADES DE COMPETENCIA	CAPACIDADES	CRITERIOS DE DESEMPEÑO
Modelar software con base en los requerimientos usando estándares y técnicas para desarrollar una solución que satisfaga las necesidades específicas de la organización.	Determinar requerimientos de software mediante el uso de estándares y técnicas establecidas y el análisis de procesos y necesidades específicas de la organización para el modelado de software.	Integra el documento de especificación de requerimientos del sistema el cual incluye: <ul style="list-style-type: none"> - Introducción: propósitos, ámbito del sistema, definiciones, acrónimos y abreviaturas, referencias y visión general del documento. - Descripción general: perspectiva del producto, funciones del producto, características de los usuarios, restricciones, suposiciones y dependencias y requisitos futuros. - Requerimientos específicos: interfaces externas, funciones, requisitos de rendimiento, restricciones de diseño, atributos del sistema. - Apéndices - Índice
	Formular modelos de software con base en los requerimientos determinados empleando metodologías y herramientas formales para desarrollar una solución que satisfaga las necesidades específicas de la organización.	Integra el documento de diseño del sistema incluyendo: <ul style="list-style-type: none"> - Diagramas de Casos de uso, - Diagrama de clases, - Diagrama de interacción, - Diagrama de componentes, - Diagramas de estado.

ELABORÓ:	Comité de Directores de la Ingeniería en Tecnologías de la Información / Ingeniería en Sistemas Computacionales	REVISÓ:	Dirección Académica
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre 2018

UNIDADES DE COMPETENCIA	CAPACIDADES	CRITERIOS DE DESEMPEÑO
Desarrollar soluciones de software con base en metodologías, procedimientos y técnicas para satisfacer las necesidades específicas de la organización.	Programar software con base en un modelo de solución empleando metodologías, herramientas de programación y pruebas para la optimización de procesos de la organización.	<p>Entrega un software de aplicación que incluya:</p> <ul style="list-style-type: none"> - Archivos de código fuente programado y documentado de acuerdo a los estándares establecidos, - Archivos de código ejecutable, - Archivos de recursos complementarios: librerías, controladores, multimedia y licencias. - Diseño de casos de prueba, - Manual técnico - Reporte de ejecución de aplicación de casos de prueba, - Normatividad y estándares de referencia.
	Implementar soluciones de software mediante procedimientos de instalación, capacitación del usuario y entrega formal de la documentación para satisfacer las necesidades específicas de la organización.	<p>Diseña y ejecuta un Plan de implementación de la solución de software que incluya :</p> <ul style="list-style-type: none"> - Cronograma de actividades: - Lista de verificación de cumplimiento de los requisitos de instalación - Programa de capacitación para los usuarios - Procedimiento de instalación y puesta en marcha del software de aplicación - Manual de usuario - Formato de entrega del software de aplicación
	Realizar mantenimiento de software mediante técnicas de análisis, pruebas y con base en los estándares establecidos, considerando las observaciones del usuario para corregir errores, mejorar rendimiento y atributos.	<p>Ejecuta mantenimiento a software acuerdo a estándares establecidos y lo documenta en un reporte que incluya :</p> <ul style="list-style-type: none"> - Requerimiento de mantenimiento - Tipo de mantenimiento - Procedimiento del mantenimiento realizado - Resultados de pruebas de funcionamiento - Validación del usuario final - Actualización del historial de versiones

ELABORÓ:	Comité de Directores de la Ingeniería en Tecnologías de la Información / Ingeniería en Sistemas Computacionales	REVISÓ:	Dirección Académica
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre 2018

UNIDADES DE APRENDIZAJE

UNIDAD DE APRENDIZAJE	I. Fundamentos del paradigma de Programación Orientado a Objetos							
PROPOSITO ESPERADO	El alumno determinará los entornos de desarrollo para codificar algoritmos bajo el paradigma de programación orientado a objetos.							
HORAS TOTALES	P	NP	HORAS DEL SABER	P	NP	HORAS DEL SABER HACER	P	NP
	18	0		6	0		12	0

TEMAS	SABER DIMENSIÓN CONCEPTUAL	SABER HACER DIMENSION ACTUACIONAL	SER DIMENSIÓN SOCIAFECTIVA
Introducción al paradigma de programación orientado a objetos	<p>Definir el concepto de paradigma de programación orientado a objetos.</p> <p>Describir características del paradigma de programación orientado a objetos:</p> <ul style="list-style-type: none"> -Clases y objetos. -Atributos y métodos. -Relaciones estáticas y dinámicas entre objetos. -Herencia. -Polimorfismo: <ul style="list-style-type: none"> - Sobrecarga. - Sobre-escritura. -Encapsulamiento. -Instanciamiento de objetos. 		Responsable. Ordenado. Sistemático. Gestión de la Información.
Entornos de desarrollo	<p>Describir las características del entorno de desarrollo de los lenguajes de programación orientado a objetos:</p> <ul style="list-style-type: none"> - Editor de código fuente. - Control de ejecución del programa. - Control de compilación. - Control de revisión de sintaxis. - Control del modo depuración. 	Proponer el entorno de desarrollo del paradigma de programación orientado a objetos.	Analítico. Responsable. Ordenado. Sistemático. Gestión de la Información. Creativo.

ELABORÓ:	Comité de Directores de la Ingeniería en Tecnologías de la Información / Ingeniería en Sistemas Computacionales	REVISÓ:	Dirección Académica
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre 2018

	Explicar el funcionamiento del entorno de desarrollo de los lenguajes de programación orientado a objetos.		
--	--	--	--

PROCESO DE EVALUACIÓN		TÉCNICAS SUGERIDAS DE ENSEÑANZA Y APRENDIZAJE	ESPACIO DE FORMACIÓN			MATERIALES Y EQUIPOS
EVIDENCIA DE DESEMPEÑO	INSTRUMENTO EVALUACIÓN		AULA	TALLER	OTRO	
Elabora un reporte digital de caso práctico de codificación mediante un entorno de desarrollo bajo el paradigma de programación orientado a objetos que contenga: <ul style="list-style-type: none"> - Resumen. - Introducción. - Desarrollo: <ul style="list-style-type: none"> - Clases. - Atributos. - Métodos. - Polimorfismo. - Herencia. - Captura de pantallas de la operación del programa. - Conclusiones. 	Caso práctico. Rúbrica.	Prácticas de laboratorio. Trabajo de Investigación. Solución de problemas.		X		Equipo de Cómputo. Proyector. Pintarrón. Internet.

ELABORÓ:	Comité de Directores de la Ingeniería en Tecnologías de la Información / Ingeniería en Sistemas Computacionales	REVISÓ:	Dirección Académica
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre 2018

UNIDAD DE APRENDIZAJE	II. Análisis y diseño orientado a objetos							
PROPÓSITO ESPERADO	El alumno representará análisis y diseños de problemas bajo el paradigma de programación orientado a objetos para codificar algoritmos.							
HORAS TOTALES	P	NP	HORAS DEL SABER	P	NP	HORAS DEL SABER HACER	P	NP
	30	0		10	0		20	0

TEMAS	SABER DIMENSIÓN CONCEPTUAL	SABER HACER DIMENSION ACTUACIONAL	SER DIMENSIÓN SOCIAFECTIVA
Introducción a UML	<p>Definir el concepto de modelado en el paradigma de programación orientado a objetos.</p> <p>Describir las características del lenguaje de modelado unificado.</p> <p>Identificar componentes del lenguaje de modelado unificado.</p> <p>Describir bloques de construcción del lenguaje de modelado unificado.</p> <ul style="list-style-type: none"> - Elementos. - Relaciones. - Diagramas. <p>Explicar el proceso de análisis del lenguaje de modelado unificado.</p>	Realizar la interpretación del lenguaje de modelado unificado.	<p>Analítico.</p> <p>Responsable.</p> <p>Ordenado.</p> <p>Sistemático.</p> <p>Gestión de la Información.</p> <p>Creativo.</p>
Diagramas UML	<p>Definir el concepto de diagrama del lenguaje de modelado unificado.</p> <p>Describir los tipos de diagramas del lenguaje de modelado unificado.</p>	Construir diagramas del lenguaje de modelado unificado.	<p>Analítico.</p> <p>Responsable.</p> <p>Ordenado.</p> <p>Sistemático.</p> <p>Gestión de la Información.</p> <p>Creativo.</p>

ELABORÓ:	Comité de Directores de la Ingeniería en Tecnologías de la Información / Ingeniería en Sistemas Computacionales	REVISÓ:	Dirección Académica
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre 2018

	<p>Describir las características de los diagramas del lenguaje de modelado unificado:</p> <ul style="list-style-type: none"> - Clases. - Objetos. - Componentes. - Despliegue. - Casos de uso. - Secuencia. - Colaboración. - Estados. - Actividades. <p>Explicar el proceso de diseño de modelos del lenguaje de modelado unificado.</p> <p>Explicar el proceso de construcción de diagramas del lenguaje de modelado unificado.</p>		
--	--	--	--

ELABORÓ:	Comité de Directores de la Ingeniería en Tecnologías de la Información / Ingeniería en Sistemas Computacionales	REVISÓ:	Dirección Académica
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre 2018

PROCESO DE EVALUACIÓN		TÉCNICAS SUGERIDAS DE ENSEÑANZA Y APRENDIZAJE	ESPACIO DE FORMACIÓN			MATERIALES Y EQUIPOS
EVIDENCIA DE DESEMPEÑO	INSTRUMENTO EVALUACIÓN		AULA	TALLER	OTRO	
Elabora un reporte digital de caso práctico de construcción del lenguaje de modelado unificado que incluya: <ul style="list-style-type: none"> - Resumen. - Introducción - Desarrollo: <ul style="list-style-type: none"> - Diagramas del lenguaje de modelado unificado. - Justificación de modelos del lenguaje de modelado unificado. - Descripción de los modelos del lenguaje de modelado unificado obtenidos. - Conclusiones. 	Caso práctico. Rúbrica.	Prácticas de laboratorio. Trabajo de Investigación. Solución de problemas.		X		Equipo de Cómputo. Proyector. Pintarrón. Internet.

ELABORÓ:	Comité de Directores de la Ingeniería en Tecnologías de la Información / Ingeniería en Sistemas Computacionales	REVISÓ:	Dirección Académica
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre 2018

UNIDAD DE APRENDIZAJE	III.Desarrollo de clases y métodos							
PROPÓSITO ESPERADO	El alumno representará clases en lenguaje de programación orientado a objetos para codificar el análisis y diseño de modelos en UML.							
HORAS TOTALES	P	NP	HORAS DEL SABER	P	NP	HORAS DEL SABER HACER	P	NP
	42	0		12	0		30	0

TEMAS	SABER DIMENSIÓN CONCEPTUAL	SABER HACER DIMENSIÓN ACTUACIONAL	SER DIMENSIÓN SOCIAFECTIVA
Construcción de clases	<p>Definir el concepto de clase y objeto.</p> <p>Describir las características y propiedades de clases y objetos:</p> <ul style="list-style-type: none"> - Atributos y métodos. - Relaciones estáticas y dinámicas entre objetos. - Herencia. - Polimorfismo. <ul style="list-style-type: none"> - Sobrecarga. - Sobre-escritura. - Encapsulamiento. - Instanciamiento de objetos. <p>Explicar el proceso sintáctico de codificación de clases y métodos.</p>	Codificar clases y métodos.	<p>Analítico.</p> <p>Responsable.</p> <p>Ordenado.</p> <p>Sistemático.</p> <p>Ético.</p> <p>Honestidad.</p> <p>Gestión de la Información.</p> <p>Creativo.</p>
Generación de código fuente de un diseño UML	<p>Explicar la relación de los diagramas del lenguaje de modelado unificado en la codificación de clases y objetos.</p> <p>Explicar el proceso de equivalencias semántico-sintácticas de diagramas del lenguaje de modelado unificado en lenguajes de programación.</p>	Codificar diseños del lenguaje de modelado unificado.	<p>Analítico.</p> <p>Responsable.</p> <p>Ordenado.</p> <p>Sistemático.</p> <p>Ético.</p> <p>Honestidad.</p> <p>Gestión de la Información.</p> <p>Creativo.</p>

ELABORÓ:	Comité de Directores de la Ingeniería en Tecnologías de la Información / Ingeniería en Sistemas Computacionales	REVISÓ:	Dirección Académica
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre 2018

PROCESO DE EVALUACIÓN		TÉCNICAS SUGERIDAS DE ENSEÑANZA Y APRENDIZAJE	ESPACIO DE FORMACIÓN			MATERIALES Y EQUIPOS
EVIDENCIA DE DESEMPEÑO	INSTRUMENTO EVALUACIÓN		AULA	TALLER	OTRO	
Elabora un reporte digital de caso práctico de la implementación de diseños del lenguaje de modelado unificado que incluya: <ul style="list-style-type: none"> - Resumen. - Introducción. - Desarrollo: <ul style="list-style-type: none"> - Diagramas de lenguaje de modelado unificado. - Código fuente de los diagramas. - Conclusiones. 	Caso práctico. Rúbrica.	Prácticas de laboratorio. Trabajo de Investigación. Solución de problemas.		X		Equipo de Cómputo. Proyector. Pintarrón. Internet. Software de compilación.

ELABORÓ:	Comité de Directores de la Ingeniería en Tecnologías de la Información / Ingeniería en Sistemas Computacionales	REVISÓ:	Dirección Académica
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre 2018

REFERENCIAS BIBLIOGRÁFICAS

AUTOR	AÑO	TÍTULO DEL DOCUMENTO	LUGAR DE PUBLICACIÓN	EDITORIAL	ISBN
Paul J. Deitel, Harvey M. Deitel	2012	Cómo programar en Java	Mexico	Pearson Educacìon	9786073211505
Harvey M. Deitel, Paul J. Deitel	2013	Còmo Programar en C/C++ y Java	Madrid, España	Pearson Educacìon	9702605318
Llorens Agost, María Luisa Gómez Adrian, Jon Ander Galiano Ronda, Isabel Remedios Herrero Cuco, Carlos Marqués Hernández, Francisco Casanova Faus, Assumpció González Mollá, Jorge Moltó Martínez, Germán Martínez Hinarejos, Carlos David Prieto Saez, Natividad Piris Ruano, Francisco Javier	2016	Empezar A Programar Usando Java	España	Editorial Universitat Politècnica de València	9788490485422
Pablo Sznajdleder	2017	Programación estructurada a fondo implementacion de algoritmos en C	México	Alfaomega Grupo Editor	9789873832284
Robert W. Sebesta	2012	Concepts Of Programming Languages	United States of America	Pearson Education	9780131395312

ELABORÓ:	Comité de Directores de la Ingeniería en Tecnologías de la Información / Ingeniería en Sistemas Computacionales	REVISÓ:	Dirección Académica
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre 2018

REFERENCIAS ELECTRÓNICAS

AUTOR	TÍTULO DEL DOCUMENTO	FECHA DE RECUPERACIÓN	VÍNCULO
Manuel Collado	Entornos de programación Concepto, funciones y tipos	15/09/2016	http://lml.ls.fi.upm.es/ep/entornos.html#toc6
J. Steven Perry	Introducción a la programación Java, parte 1 Conceptos básicos del lenguaje Java Programación orientada a objetos en la plataforma Java	26/10/2017	https://www.ibm.com/developerworks/ssa/java/tutorials/j-introjava1/index.html
Luis R. Izquierdo	Introducción a la Programación Orientada a Objetos	30/11/2017	http://luis.izqui.org/resources/ProgOrientadaObjetos.pdf

ELABORÓ:	Comité de Directores de la Ingeniería en Tecnologías de la Información / Ingeniería en Sistemas Computacionales	REVISÓ:	Dirección Académica
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre 2018