



Asignatura:

Obligatoria

Optativa

Horas:

Teóricas

Prácticas

Total (horas):

Semana

16 Semanas

Modalidad: Curso.

Asignatura obligatoria antecedente: Ninguna.

Asignatura obligatoria consecuente: Ninguna.

Objetivo(s) del curso:

El alumno aplicará en un proyecto procesos y herramientas mediante las cuales se analiza, diseña e implementa un proyecto de software.

Temario

NÚM.	NOMBRE	HORAS
1.	Introducción	3.0
2.	El Lenguaje de Modelación Universal (LMU)	12.0
3.	Planeación del Sistema de Programación	12.0
4.	Diseño	9.0
5.	Introducción a la Investigación de Operaciones	12.0
6.	Sistemas Críticos	7.5
7.	Verificación y Validación	6.0
8.	Administración	6.0
9.	Evolución y Tendencias	4.5
		72.0
	Prácticas de laboratorio	0.0
	Total	72.0



1 Introducción

Objetivo: El alumno explicará la importancia de la crisis del software y porque continua vigente hasta el día de hoy.

Contenido:

- 1.1 Introducción
- 1.2 Ingeniería de sistemas de software
- 1.3 Procesos del software
- 1.4 Administración de proyectos

2 El Lenguaje de modelación universal (LMU)

Objetivo: El alumno explicará los diagramas que intervienen en el diseño de la arquitectura de un sistema

Contenido:

- 2.1 Diagramas de clase y objetos
- 2.2 Diagramas de componentes
- 2.3 Diagramas de exhibición
- 2.4 Diagramas de casos de uso
- 2.5 Diagramas de secuencia
- 2.6 Diagramas de estados
- 2.7 Diagramas de actividades

3 Planeación del sistema de programación

Objetivo: El alumno explicará como se modelan los requerimientos de un sistema de software

Contenido:

- 3.1 Estudio general del sistema
- 3.2 Requerimientos del software
- 3.3 Procesos en los requerimientos
- 3.4 Modelos de sistemas
- 3.5 Prototipos de sistemas de software
- 3.6 Especificación formal

4 Diseño

Objetivo: El alumno explicará como las especificaciones de una aplicación son la base para obtener el diseño arquitectónico del sistema.

Contenido:

- 4.1 El diseño arquitectónico
- 4.2 Arquitectura de sistemas distribuidos
- 4.3 Diseño orientado a objetos
- 4.4 Diseño de software de tiempo real
- 4.5 Diseño reutilizable
- 4.6 Diseño de la interfaz de usuario final



5 Introducción a la investigación de operaciones

Objetivo: El alumno aplicará algoritmos de programación lineal a diversos casos en el contexto de la ingeniería de software.

Contenido:

- 5.1 Programación lineal
- 5.2 Algoritmos especiales
- 5.3 Pert, CPM
- 5.4 Gant

6 Sistemas críticos

Objetivo: El alumno explicará la importancia de considerar a algunos sistemas de software como críticos y la explicará los conceptos relacionados con estos.

Contenido:

- 6.1 Confiabilidad
- 6.2 Especificación de sistemas críticos
- 6.3 Desarrollo de sistemas críticos

7 Verificación y validación

Objetivo: El alumno explicará los conceptos de verificación y validación y su importancia en el desarrollo de los proyectos del software

Contenido:

- 7.1 Verificación y validación
- 7.2 Pruebas del software
- 7.3 Validación de Sistemas Críticos

8 Administración

Objetivo: El alumno explicará la importancia de la administración de los proyectos de software para un desarrollo exitoso de estos.

Contenido:

- 8.1 Administración de personal
- 8.2 Estimación del costo del software
- 8.3 Administración de la calidad
- 8.4 Mejora de procesos

9 Evolución y tendencias

Objetivo: El alumno explicará los conceptos asociados con la evolución de los sistemas de software y elaborará una prospectiva del software

Contenido:

- 9.1 Sistemas heredados
- 9.2 Cambios en el software
- 9.3 Reingeniería del software



- 9.4 Administración de la configuración
- 9.5 Tendencias futuras

Bibliografía básica:

SOMMERVILLE, Ian
Software Engineering
7th Edition
Reading, Massachusetts, U.S.A
Addison Wesley, 2004

WEITZENFELD, Alfredo
Ingeniería de Software orientada a objetos con UML, Java e Internet
México
Thomson, 2005

Bibliografía complementaria:

LARMAN, Craig
Applying UML and Patterns
New Jersey, U.S.A
Prentice Hall, 2002

Sugerencias didácticas:

Exposición oral

Exposición audiovisual

Ejercicios dentro de clase

Ejercicios fuera del aula

Seminarios

Temas para los que se recomienda:

Todos

Todos

Todos

Lecturas obligatorias

Trabajos de investigación

Prácticas de taller o laboratorio

Prácticas de campo

Otras

Forma de evaluar:

Exámenes parciales

Exámenes finales

Trabajos y tareas fuera del aula

Participación en clase

Asistencias a prácticas

Otras



Perfil profesiográfico de quienes pueden impartir la asignatura

Profesional con experiencia en ingeniería de software: teoría y práctica (metodologías, principios, conceptos, técnicas, métodos).