



A. IDENTIFICACIÓN

CARRERA:	INGENIERÍA INFORMÁTICA
ASIGNATURA:	INGENIERÍA DE SOFTWARE II
SIGLA:	INF 3811
DURACIÓN:	Un semestre académico (20 semanas)
HORAS SEMANALES:	Teóricas: 4, Laboratorio: 1, Prácticas: 1, TOTAL: 6
PLAN DE ESTUDIOS:	2011

B. CONTRIBUCIÓN AL PERFIL

Objetivos:

El Estudiante obtendrá al final de esta asignatura, conocimientos sólidos en el empleo los métodos en el análisis y diseño de soluciones software orientado a objetos, la aplicación práctica de tales métodos en proyectos reales. El estudiante tendrá la capacidad de aplicar UML en el análisis y diseño de desarrollo de aplicaciones. Además de establecer claramente el alcance y los requerimientos del software, así como también será capaz de escribir y comprender un documento formal del software, basado es el Proceso Unificado de Desarrollo de Software.

Unidades de competencia:

- Habilidad para la utilización de los métodos en el análisis y diseño de soluciones basadas en software y la aplicación práctica de tales métodos en proyectos reales de construcción de aplicaciones de Software.
- Capacidad para aplicar algunos métodos de ingeniería de software para el análisis de requerimientos en el desarrollo de aplicaciones.

C. CONTENIDO PROGRAMÁTICO

Contenido mínimo:

Introducción y el modelado con UML.- Requerimientos.- Análisis y diseño orientado a objetos.- Implementación.- Pruebas y desarrollo de un proyecto de software.

Contenido analítico:

Tema 1: Introducción y el modelado (U.M.L.).

1.1 Introducción a UML.

1.1.1 Concepción de UML.

1.1.2 Orientación a objetos.

1.1.3 Clasificadores, relaciones, generalizaciones, realizaciones, dependencias y restricciones.

1.2 Área Estructural.

1.2.1 Diagrama de clases.

1.2.2 Diagrama de Casos de Usos.

1.2.3 Diagrama de Componentes.



- 1.2.4 Diagramas de Despliegue.
- 1.2.5 Diagrama de Objetos.
- 1.3 Área Dinámica.
 - 1.3.1 Diagrama de Secuencia.
 - 1.3.2 Diagrama de Comunicación.
 - 1.3.3 Diagrama de Estado.
 - 1.3.4 Diagrama de Actividad.
 - 1.3.5 Diagrama de Tiempo.
- 1.4 Otras características.
 - 1.4.1 Paquetes.
 - 1.4.2 Notas.
 - 1.4.3 Estereotipos.
- 1.5 El Proceso unificado de desarrollo de software.
- 1.6 Flujo de trabajo fundamental.
- 1.7 Desarrollo iterativo e incremental.

Tema 2: Requerimientos.

- 2.1 Requerimientos del software.
 - 2.1.1 Requerimientos funcionales.
 - 2.1.2 Requerimientos No funcionales.
- 2.2 Modelo del dominio.
- 2.3 Modelo del negocio.
- 2.4 Comportamiento del sistema.

Tema 3: Análisis Orientado a Objetos.

- 3.1 Modelo del análisis.
- 3.2 Modelo de Casos de usos.
- 3.3 Clase del análisis.
- 3.4 Descripción de la arquitectura.

Tema 4: Diseño Orientado a Objetos.

- 4.1 Modelo de diseño.
- 4.2 Casos de uso reales.
- 4.3 Clases del diseño.
- 4.4 Diseño de la arquitectura (multicapa).
- 4.5 Modelo de despliegue.
- 4.6 Diseño de interfaces de usuarios.

Tema 5: Implementación.

- 5.1 Modelo de implementación.
- 5.2 Modelo de componentes.
- 5.3 Implementación de la arquitectura.
- 5.4 Refinamiento y reutilización del software.
- 5.5 Casos prácticos.

Tema 6: Pruebas Orientadas a Objetos.



- 6.1 Modelo pruebas.
- 6.2 Casos de pruebas.
- 6.3 Plan de pruebas.
- 6.4 Evaluar pruebas.

Tema 7: Desarrollo de un Proyecto de Software.

- 7.1 Elaboración de Proyecto Software.

D. REFERENCIAS BIBLIOGRÁFICAS

- [1] Jacobson, I.; Booch, G.; Rumbaugh, J. (2000). **El Lenguaje Unificado de Modelado, Manual de referencias**. Madrid: Ed. Addison-Wesley.
- [2] Schmuller, J (2001). **Aprendiendo UML en 24 Horas**. Ed. Prentice Hall.
- [3] Jacobson I.; Booch G.; Rumbaugh J. (1999). **El Proceso Unificado de Desarrollo de Software**. Madrid: Ed. Addison-Wesley.
- [4] Fowler, M (1999). **UML gota a gota**. México: Ed. Addison-Wesley.
- [5] Sommerville, I. (2005). **Ingeniería de Software** (7ma. Edición). Editorial Addison-Wesley.