



SYLLABUS

1. INFORMACIÓN GENERAL

– Nombre de la Asignatura	:	INGENIERÍA DE SOFTWARE (E)
– Nro. de créditos	:	4
– Nro. de Horas (teoría y práctica)	:	4 (2h. Teoría, 2h Laboratorio)
– Pre-requisito	:	Diseño de Sistemas
– Carácter	:	Electivo
– Ciclo Académico	:	X
– Semestre Académico	:	2016B

2. SUMILLA

Utilizar la teoría de otras disciplinas y aplicarlas de forma efectiva en la solución de problemas complejos de sistemas de software. El software no solo significa "programas de computadora" asociados con una aplicación o producto. Además de esto, implica la documentación necesaria para dar seguimiento, la instalación, el uso y el mantenimiento del software. En sistemas complejos de gran tamaño, el esfuerzo necesario para escribir esta documentación es a menudo tanto como lo es el desarrollo del software

3. OBJETIVOS GENERALES

Objetivo General:

Capacitar al alumno para que pueda desarrollar aplicaciones de software para la empresa, y que le permita comprender los conceptos, métodos técnicas y herramientas en el desarrollo de aplicaciones para la empresa. Se hace énfasis en aplicaciones reales que pueden ser desarrolladas por el Ingeniero de Sistemas; y, se complementa con una capacitación práctica en herramientas orientadas al desarrollo de aplicaciones para la empresa.

Objetivo Específico

1. Consolidar los conceptos relacionados a la ingeniería de software.
2. Desarrollar una cultura de ingeniería de software.
3. Reforzar la importancia de la resolución de problemas haciendo un uso racional de los recursos.
4. Fortalecer las habilidades en el proceso de desarrollo (construcción o adaptación) de software.
5. Desarrollar habilidades para mejorar las especificaciones y requisitos de software.
6. Establecer consideraciones para mejorar las especificaciones y requisitos de software.
7. Valorar el proceso de verificación y validación de software.

4. METODOLOGÍA

La metodología empleada, será activa, que favorezca la participación del alumno, complementando con la práctica de la inducción específicamente en la resolución de los problemas propuestos.

Se estimulará el interés por la investigación de temas planteados. La programación del curso considera, para el cumplimiento de los objetivos las siguientes técnicas de trabajo:

- Exposición y diálogo.
- Exposición sobre el avance del proyecto final.
- Proyectos Final en Grupos.
- Investigación en el Laboratorio de Computación y Sistemas.
- Uso intensivo de Internet, correo electrónico y discos virtuales.
- Uso intensivo del campus virtual.

5. CRITERIOS DE EVALUACIÓN

N1	:	Nota de Examen Parcial
N2	:	Notas de Trabajos Prácticos
N3	:	Nota de Examen Final
N4	:	Nota de Proyecto Final

La nota del Examen Sustitutorio reemplaza a la nota más baja.

TA: Tarea académica: $(N2+N4)/2$

Promedio General = $((3 * TA) + (3 * N1) + (4 * N3)) / 10$

La nota mínima final aprobatoria es 11.

6. MATERIAL EDUCATIVO

- Distribución de copias y separatas del curso.
- Pizarra para la teoría y Pc's para la parte práctica.
- USB para portar trabajos.
- Software de Modelamiento Data Modeler, ERWIN, Star UML.
- Laboratorio equipado con computadoras personales.
- Campus Virtual.

7. CONTENIDO TEMÁTICO

SEMANA 1

- La Ingeniería de Software.
- Modelos del proceso de software (modelo lineal, de construcción de prototipos, modelo DRA, modelos evolutivos de proceso de software, desarrollo basado en componentes).
- El enfoque de Sistemas.
- El paradigma orientado a objetos.
- Introducción a la Ingeniería de Software y la gestión de la información.
- El UML. El Lenguaje de Modelado Unificado.
- Modelado con el UML. Herramientas.

SEMANA 2

- El Proceso Unificado Rational (RUP) o Proceso Unificado de Desarrollo de Software. Definiciones Generales.
- Fases de RUP (Inicio, Elaboración, Construcción y Transición) e Iteraciones. Componentes del Proceso.
- Características del Proceso Unificado.
- Modelo del Negocio. Modelo de Casos de Uso de Negocio

SEMANA 3

- Modelo de Análisis de Negocio
- Realizaciones de Casos de Uso de Negocio
- Diagrama de actividades del negocio.
-

SEMANA 4

- Modelos de objetos del negocio
- Diagramas de secuencia Objetos de Negocio.
- Diagramas de colaboración Objetos de Negocio.

SEMANA 5

- Modelo de Requerimientos.
- Requerimientos funcionales.
- Requerimientos no funcionales.
- Diagrama de Arquitectura de Paquetes.
- Documentando los escenarios.

SEMANA 6

- Diagrama de casos de uso de Sistemas.
- Documentación de casos de uso de Sistema.
- Priorización de casos de uso de Sistema

SEMANA 7

- Definición de Objetos. Definición de Clases. Encontrando clases.
- Herencia. Generalización y especialización.
- Encapsulamiento y polimorfismo.
- Instanciación y Generalización.
- Composición y agregación.
- Modelo Conceptual de Clases

SEMANA 8

- EXAMEN PARCIAL

SEMANA 9

- Modelo de Análisis
- Diagrama de clases de análisis.
- Diagramas de colaboración clases de Análisis.
- Diagramas de Secuencia de clases de Análisis.

SEMANA 10

- Desarrollo del Modelo Lógico.
- Clases Asociativa.
- Consideraciones para pasar del Modelo Conceptual al Modelo Lógico Clases.
- Analizando el comportamiento de los objetos
- Diagramas de Estados.

SEMANA 11

- Modelo de Diseño.
- Diagrama de clases de diseño.
- Casos de uso de realización.
- Diseño de interfaz de usuario.
- Diagramas de secuencia.
- Diagramas de actividades.

SEMANA 12

- Modelo Físico de Clases. El Data Modeler.
- Consideraciones para pasar del Modelo Lógico de Clases al Modelo Físico.

SEMANA 13

- Arquitectura lógica. Arquitectura física. Hardware. Software.
- Diagramas de componentes.
- Diagrama de despliegue.

SEMANA 14

- Diseño de casos de uso de prueba.
- Métodos de prueba aplicables a nivel de clase.
- Diseño de casos de prueba a nivel de clase.
-

SEMANA 15

- Métricas medidas e indicadores. Métricas para la calidad de software.
- Establecimiento de un sistema de métricas de software.

SEMANA 16

- CMMI, SCAMPI, MOPROSOFT y Otros Estándares.

SEMANA 17

- EXAMEN FINAL.

8. BIBLIOGRAFÍA

1. PRESSMAN, Roger. *“Ingeniería del Software. Un Enfoque Práctico”*. 5ª ed. México: McGraw-Hill Latinoamericana, 2002. ISBN: 8448132149.
2. BOOCH, Grady et tal. *“El Proceso Unificado de Desarrollo de Software”*. 1ª ed. España: Editorial Addison-Wesley.
3. LARMAN, Craig. *UML y Patrones – Introducción al Análisis y Diseño Orientado a Objetos*. 1ª ed. España: Pearson Educación.
4. SENN, James. *“Análisis y Diseño de Sistemas de Información”*. México: Mc Graw Hill. ISBN: 9684229917.
5. BRAUDE, J. *“Ingeniería de Software: Una Perspectiva Orientada a Objetos”* Rama. ISBN: 8478975756. ISBN-13: 9788478975754.
6. SOMMERVILLE, Ian *“Ingeniería de Software: Un enfoque practico”*, Eddison Wesley, México, 692 p.