



UNIVERSIDAD DEL CAUCA
FACULTAD DE INGENIERIA ELECTRÓNICA Y TELECOMUNICACIONES
PROGRAMA DE INGENIERIA DE SISTEMAS

ASIGNATURA:	INGENIERIA DE SOFTWARE I
CÓDIGO:	SIS503
MODALIDAD:	PRESENCIAL TEÓRICO
INTENSIDAD:	4 HORAS / SEMANA.
PREREQUISITOS:	ESTRUCTURAS DE DATOS II – LABORATORIO ESTRUCTURAS DE DATOS II
CORREQUISITOS:	LABORATORIO DE INGENIERIA DE SOFTWARE I
ÁREA:	INGENIERIA APLICADA
CRÉDITOS:	3

OBJETIVO GENERAL

Estudiar los conceptos fundamentales de la Ingeniería de Software en el proceso de desarrollo de proyectos informáticos en referencia a su gestión de riesgos, estimación y las técnicas más utilizadas para realizar recolección de requerimientos, análisis y diseño orientado a objetos.

OBJETIVOS ESPECIFICOS

Al finalizar el curso el estudiante estará en capacidad de:

1. Comprender los conceptos fundamentales relativos a la Ingeniería del Software que favorezcan la consolidación de su actitud crítica respecto al valor de metodologías y técnicas propias del ejercicio profesional del Ingeniero de Software.
2. Reflexionar, interpretar y poner en práctica un proceso de desarrollo de software desde tres perspectivas de profundización fundamentales que son: Comprensión de la secuencia de Fases y Actividades del proceso, generación de modelos de análisis y diseño descritos como artefactos con notación del Lenguaje Unificado de Modelado (UML) y por último la correlación o dependencia de los artefactos del análisis y el diseño.
3. Comprender los lineamientos metodológicos relativos al diseño de la interacción humano-computador para aplicarlos dentro del marco de un proceso de desarrollo de software.
4. Preparar, sintetizar y aplicar a proyectos de desarrollo de software conceptos relativos a la gestión de riesgos y estimación de esfuerzo, como parte del proceso de desarrollo de un producto software.

METODOLOGIA

1. Clases magistrales que ayuden al estudiante en la incorporación del conocimiento relativo a conceptos y métodos de la ingeniería de software, gestión de riesgos y estimación de proyectos de software y el Proceso Unificado de Desarrollo.
2. Talleres prácticos dirigidos, basados en un caso de estudio único e integral que proporcionen el espacio temporal y de infraestructura tecnológica tal que permitan al estudiante la aplicación directa y visible de los conocimientos teóricos adquiridos durante las clases magistrales.
3. Trabajos de investigación que fortalezcan en el estudiante la capacidad de interpretación, síntesis, confrontación y presentación de la información relacionada con la investigación.
4. Lecturas Orientadas. El docente elabora un cuestionario con preguntas que orientarán al estudiante en la identificación del conocimiento relevante que debe adquirir hacia el final de la lectura.

CONTENIDO

1. INTRODUCCION (4 HORAS).

- 1.1 Definición de la Ingeniería de Software.
- 1.2 Sistemas de Información.
- 1.3 Modelos de Procesos de Desarrollo de Software.

2. ESTIMACION DEL SOFTWARE (8 HORAS).

- 2.1 Conceptos básicos sobre Planificación de Proyectos
- 2.2 Definición y selección de un conjunto de tareas para el proyecto de software.
- 2.3 Desarrollo de una Estructura de División del Trabajo (WBS).
- 2.4 Definición e importancia de la estimación.
- 2.5 Objetivos de la estimación.
- 2.6 Técnicas de descomposición.
- 2.7 Estimación basada en el problema.
- 2.8 Estimación basada en el proceso.
- 2.9 Estimación con modelos empíricos.

3. INGENIERÍA DE REQUERIMIENTOS (12 HORAS)

- 3.1 Introducción.
- 3.2 Características y Actividades.
- 3.3 Técnicas Utilizadas.

4. ADOO Y PROCESO DE DESARROLLO DE SOFTWARE (24 horas)

- 4.1 Introducción.
- 4.2 Características Fundamentales del Proceso Unificado.
- 4.3 Disciplina de Análisis en el Proceso Unificado.
- 4.4 Disciplina de Diseño en el Proceso Unificado.
- 4.5 Metodologías Alternas y sus características fundamentales.
- 4.6 Disciplina de Análisis en Metodologías Alternas.
- 4.7 Disciplina de Diseño en Metodologías Alternas.
- 4.8 Valoración de las diferentes metodologías y criterios básicos para determinar su utilización.

5. GESTION DE RIESGOS (8 HORAS).

- 5.1 Riesgos del software.
- 5.2 Tipos de Riesgos.
- 5.3 Grupo de Procesos del Área de Gestión de Riesgos
 - 5.3.1 Planeación de la Gestión de Riesgos
 - 5.3.2 Identificación de Riesgos
 - 5.3.3 Análisis Cualitativo de Riesgos
 - 5.3.4 Análisis Cuantitativo de Riesgos
 - 5.3.5 Planeación de Respuestas a los Riesgos
 - 5.3.6 Monitoreo y Control de Riesgos

6. FUNDAMENTOS DE INTERACCION HUMANO – COMPUTADOR (4 horas)

- 6.1 Motivación: Importancia del usuario en el diseño de interfaz.
- 6.2 Principios de HCI.
- 6.3 Desarrollo y Evolución de las interfaces de usuario.
- 6.4 Principios de Diseño de Interfaces de Usuario.

PLANEACIÓN CON RESPECTO AL LABORATORIO

Semana	Tema Asignatura Teórica	Práctica Laboratorio
1	Capitulo 1	Práctica 1
2	Capitulo 2	Práctica 2
3	Capitulo 2	Práctica 3
4	Capitulo 3	Practica 4
5	Capitulo 3	Practica 5
6	Capitulo 3	Practica 6
7	Capitulo 4	Practica 7
8	Capitulo 4	Practica 8
9	Capitulo 4	Practica 9
10	Capitulo 4	Practica 10
11	Capitulo 4	Practica 11
12	Capitulo 4	Practica 12
13	Capitulo 5	Practica 13
14	Capitulo 5	Practica 14
15	Capitulo 6	Practica 15

EVALUACIONES.

Se realizarán tres (3) evaluaciones de la siguiente forma:

CORTE	%	COMPONENTES	
Primero	35	Parcial Escrito	80%
		Talleres, Quices	20%
Segundo	35	Parcial Escrito	80%
		Talleres, Quices	20%
Tercero	30	Parcial Escrito	80%
		Talleres, Quices	20%

Los ejercicios de aplicación y talleres en grupo serán evaluados individualmente y deben estar debidamente documentados. Todo ejercicio NO sustentado pierde validez. Las sustentaciones serán programadas con anterioridad definiendo fecha y hora para cada alumno.

BIBLIOGRAFÍA.

- LARMAN Craig. UML Y PATRONES: Una introducción al análisis y diseño orientado a objetos y al Proceso Unificado. PEARSON - Prentice Hall. Segunda Edición, 2003.
- JACOBSON Ivar, BOOCH Grady, RUMBAUGH James. The Unified Software Development Process. Rational Software Corporation. Addison Wesley, 1999. ISBN: 0-201-57169-2. 463 págs.
- PRESSMAN, Roger. Ingeniería del Software, un enfoque práctico. McGraw-Hill, Sexta Edición, 2006.
- MEYER Bertrand. Construcción de Software Orientado a Objetos. Prentice-Hall, 1999.
- Information Technology Project Management. Schwalbe Kathy. Fifth Edition. Thomson Learning (cengage). 2007.
- A guide to the Project Management Body of Knowledge. Fifth Edition. (PMBOK GUIDE). Project Management Institute. ANSI/PMI 99-001-2004.
- McConnell, S. (1997): Desarrollo y gestión de proyectos informáticos. McGrawHill/Interamericana de España, 691 p.