

**PLAN GLOBAL
INGENIERÍA DE SOFTWARE**

I. DATOS DE IDENTIFICACIÓN

■ Nombre de la materia:	Ingeniería de Software
■ Código:	2010020
■ Grupo:	1, 2
■ Carga horaria:	24 hrs
■ Materias con las que se relaciona:	Sistemas de Información II Taller de Ingeniería de Software
■ Docente:	Indira Camacho del Castillo Rosemary Torrico Bascope
■ Teléfono:	4798716
■ Correo Electrónico:	indira@cs.umss.edu.bo rosemarytb@yahoo.com

II. JUSTIFICACIÓN

En los últimos años el avance de la tecnología ha sido muy importante, la informática como una ciencia, ocupa un lugar fundamental en el desarrollo científico, tecnológico, comercial, educativo y ha incursionado en todos los ámbitos de nuestra vida, en la actualidad estamos rodeados de computadoras hasta en el auto la radio digital tiene una pequeña computadora, los semáforos inteligentes, el ascensor, en el supermercado, por donde vayamos nos encontraremos con las tecnologías de la información. La informática se nutre de los avances tecnológicos de la computación y la comunicación.

En la actualidad además de los medios convencionales de mercadeo y marketing, empezó la era de las tiendas virtuales y el marketing por Internet, donde cualquier institución u organización tiene la posibilidad de poner a disposición de quien lo requiera servicios y mercancía de diversa índole. Dada esta realidad, muchas organizaciones que todavía no entraron a la era de la tecnología se han visto obligadas a hacerlo puesto que si no lo hacen quedan desplazadas y hasta fuera del negocio.

Actualmente la demanda de software es cada vez mayor, se requiere producir software para el mundo aplicando las mejores técnicas de desarrollo de software.

Para ello hay que tener presente que el desarrollo de software es una actividad compleja debido a la naturaleza intangible de software y llevar adelante proyectos nuevos de software que se realicen dentro del presupuesto y tiempo planeado es todavía una quimera difícil de lograr. A todo esto hay que añadir que el software para ser útil y de calidad, debe satisfacer las necesidades del usuario y superar sus expectativas, permitir modificaciones futuras (mantenimiento) debido a la naturaleza cambiante del entorno que le rodea, ser fácil de utilizar, etc.

La Ingeniería de Software estudia el proceso de desarrollo de software con el objetivo de producir programas de calidad en un tiempo y presupuesto definidos. Para lograr esto se unen dos ciencias: las ciencias administrativas y las ciencias de la informática, donde una aporta los conocimientos administrativos que permiten planificar, dirigir, organizar y controlar el proceso y las ciencias de la computación los avances tecnológicos, metodologías, métodos y herramientas que permiten construir el producto.

III. OBJETIVOS

- Analizar modelos del proceso: Metodologías tradicionales y Metodologías Ágiles
- Enfrentar al estudiante a experiencias que desarrollen sus habilidades de planificación y organización.
- Identificar diversos modelos de procesos de desarrollo de software y ser capaz de evaluar su utilidad para un proyecto dado.
- Identificar los componentes genéricos de las fases de definición, desarrollo y mantenimiento del software.
- Identificar los productos y subproductos a producir en las diferentes actividades de desarrollo.
- Identificar las principales características de calidad del SW.
- Reflexionar sobre el SQA (el aseguramiento de la calidad del software) en el proceso de desarrollo.
- Elaborar el documento de especificación de requerimientos.
- Definir modelos de estimación de recursos para estimar tiempo, esfuerzo y costo del proceso de desarrollo.
- El estudiante será capaz de reconocer que todo proyecto está sujeto a riesgos, y crear un plan de gestión de riesgos.
- El estudiante reconocerá la importancia de la planificación y gestión de proyectos de software.

- El estudiante será capaz de realizar un plan de proyecto de software para el desarrollo del mismo.
- El estudiante será capaz de desarrollar software como parte de un equipo de desarrollo aplicando técnicas señaladas por la ingeniería de software.

IV. SELECCIÓN Y ORGANIZACIÓN DE CONTENIDOS

UNIDAD 1: INTRODUCCIÓN A LA INGENIERÍA DE SOFTWARE

Objetivo de la Unidad

- Analizar la crisis del Software y por qué la necesidad de la Ingeniería de Software.
- Identificar el proceso de Ingeniería de Software: Actividades involucradas en la producción de software.
- Identificar los productos y subproductos de las diferentes actividades del proceso de desarrollo de software.

Contenido

- 1.1 Historia: por qué surge la necesidad de una Ingeniería de Software
 - La Crisis del Software
 - Estadísticas de Costos, Resultados y Reparaciones
 - Otros aspectos relevantes que dieron lugar a La Ingeniería de Software
 - Condiciones para el cambio
- 1.2 Hacia una Definición
 - Objetivo de la Ingeniería de Software
 - Definición de Ingeniería de Software
 - Aplicaciones de la Ingeniería de Software
- 1.3 El producto y el proceso
 - Análisis del Software
 - Visión Genérica del proceso de desarrollo de Software
 - Definición
 - Desarrollo
 - Mantenimiento
- 1.4 Modelos de procesos
 - Modelos Tradicionales
 - Modelos Agiles

UNIDAD 2: ADMINISTRACIÓN DE PROYECTOS DE SOFTWARE

Objetivo de la Unidad

- Reflexionar sobre la importancia de la administración del proceso de software y su relación con la productividad.
- Enumerar y describir las tareas de la administración de proyectos de software.
- Explicar el proceso de aseguramiento de la calidad.
- Diseñar la estructura organizativa de su grupo de trabajo.
- Utilización de herramientas para el proceso de construcción de software
- Definir estándares de documentación, comunicación y codificación y aplicarlos a un proyecto específico.

Contenido:

- 2.1 Qué hace un administrador de proyectos de software?
 - Dificultades en la administración de un proyecto de software:
 - Actividades de la Administración
 - Factores Humanos en la Ingeniería de Software
 - Documentación
- 2.2 Las métricas y los modelos de calidad en la producción de software
 - Métricas de calidad
- 2.3 Aseguramiento de la Calidad del Software
 - El proceso de aseguramiento de la calidad
 - Modelos CMMI
 - Modelo ISO 9000

UNIDAD 3: FASE DE DEFINICIÓN

Objetivo de la Unidad

- Definir el Modelo del Dominio
- Elaborar el Documento de Especificación de Requerimientos de un proyecto de software.
- Describir en qué consiste la etapa de Planificación del Proyecto de Software.
- Analizar la aplicación de los diferentes modelos de desarrollo de software.
- Estimar esfuerzo, costos y tiempo de un proyecto de desarrollo de software.
- Realizar la gestión de riesgos.
- Elaborar un plan para el desarrollo de software.

Contenido

- 3.1 Análisis del sistema
 - Actividades estructurales del Análisis del sistema
 - Modelado del sistema
 - Modelo del Dominio
 - El documento de especificación de requerimientos
 - Requerimientos funcionales y no funcionales
 - Atributos de los Requerimientos
 - La necesidad de realizar gestión de requerimientos
- 3.2 Planificación del Proyecto de Software
 - Análisis de factibilidad
 - El documento de plan de proyecto
 - Planeación del proceso de desarrollo
 - Calendarización
 - Estimación de Recursos: Esfuerzo, tiempo y costo
- 3.3 Análisis de Riesgos

UNIDAD 4: FASE DE DESARROLLO

Objetivo de la Unidad

- Describir las cualidades esenciales de un “buen diseño”
- Identificar las actividades estructurales del diseño de sistemas.
- Describir en qué consiste la etapa de codificación
- Describir las cualidades esenciales de un buen código
- Describir en qué consiste la etapa de prueba de software
- Explicar los aspectos que hay que tomar en cuenta para conseguir programas confiables
- Explicar los aspectos que hay que tomar en cuenta para conseguir código reusable
- Explicar cuando un proceso de pruebas es exitoso.
- Aplicar técnicas para descubrir defectos en un programa
- Planificar conjunto estrategias para el proceso de verificación y pruebas de software.

Contenido:

4.1 Diseño del Software

- Actividades estructurales del diseño
- Diseño de la interfaz de usuario

4.2 Codificación

- Actividades estructurales de la codificación.
- Aspectos de calidad del código
- Recomendaciones para realizar un código de calidad: Un programa confiable.
- Refactorización del código

4.3 Prueba del Software

- Verificación y validación
- Tipos de pruebas
- El proceso de pruebas de software
- Planificación de las pruebas
- Estrategias de pruebas
- ¿Cómo descubrir defectos en un programa?

UNIDAD 5: FASE DE MANTENIMIENTO

Objetivo de la Unidad

- Identificar el papel del mantenimiento dentro del proceso de desarrollo de software.
- Describir la diferencia entre corrección, adaptación y mejora.
- Identificar las actividades del Mantenimiento de Software: planificación e implementación de los cambios del sistema.
- Realizar la gestión de configuración de un proyecto de software.

Contenido:

5.1 Actividades del proceso de mantenimiento

5.2 Tipos de mantenimiento

- Correctivo
- Adaptativo
- Perfectivo
- Preventivo (Re-ingeniería)

5.3 Costos de Mantenimiento

5.4 Gestión de la configuración del software

V. METODOLOGIAS

La metodología de enseñanza que se consideró apropiada para lograr el máximo aprovechamiento del estudiante dentro del proceso enseñanza-aprendizaje será:

- Clase dialogada.
- Investigación sobre algún aspecto de ingeniería de software.
- Dinámicas grupales incorporando elementos lúdicos para la comprensión de las unidades temáticas a través del uso de metáforas, analogías, etc.

VI. CRONOGRAMA O DURACIÓN EN PERIODOS ACADÉMICOS POR UNIDAD

UNIDAD	DURACIÓN (HORAS ACADÉMICAS)	DURACIÓN EN SEMANA
Introducción a la Ingeniería de Software	24	4
Administración de Proyectos de Software	24	4
Fase de Definición	30	5
Fase de Desarrollo	24	4
Fase de Mantenimiento	18	3

VII. CRITERIOS DE EVALUACIÓN

El nivel de conocimientos de los estudiantes se medirá y calificará mediante pruebas escritas, participación activa en actividades desarrolladas en las clases y trabajo prácticos. Para motivar la participación activa de los estudiantes se calificará los aportes que realice en clase.

Porcentaje de distribución de la nota:

Actividades en clases:	20%
Trabajos Prácticos:	10%
<u>Exámenes:</u>	<u>70%</u>
Total:	100%

VIII. BIBLIOGRAFÍA

Texto base:

- SOMMERVILLE, IAN. Ingeniería de software. 6ta. Edición en castellano, 2002.
- PRESSMAN, ROGER S. Ingeniería de Software, 5ta Edición en castellano, 2006.
- SEBASTIAN PRIOLO, Métodos ágiles, 1ra. Edición, 2009.
- Textos de referencia y consulta:
- BERND BRUEGGE & ALLEN H. DUTOIT, Ingeniería de Software Orientada a Objetos, Prentice Hall, 1ra. Edición, 2002.
- JORGE BORJA, Ingeniería fde Software, Editorial Kapelusz S.A., 1987.
- <http://www.andrew.cmu.edu/user/conzalez/Teaching/ISWM/SyllabusISMP97.html>