

1.- DATOS DE LA ASIGNATURA

Nombre de la asignatura :	Desarrollo e implementación de Sistemas de Información
Carrera :	Ingeniería Informática
Clave de la asignatura :	IFD-1011
SATCA ¹	2-3-5

2.- PRESENTACIÓN

Caracterización de la asignatura.

En esta asignatura se utilizan métodos de Ingeniería de Software orientada a objetos y herramientas CASE para desarrollar sistemas de información. Se deberán de utilizar también los conocimientos previos adquiridos en la asignatura de Análisis y Modelado de Sistemas de Información.

Esta asignatura tiene relación con las siguientes asignaturas:

- Fundamentos de sistemas de información
- Análisis y diseño de sistemas de información
- Fundamentos de base de datos
- Sistema de gestión de base de datos
- Fundamentos de programación
- Estructura de datos

En esta asignatura el alumno deberá tomar un rol activo en cada unidad de aprendizaje que le permita desarrollar capacidades para desarrollar e implementar proyectos de sistemas de información.

Intención didáctica.

Se organiza el plan de estudios de Desarrollo e implementación de Sistemas de Información en 4 unidades temáticas, y cada una de ellas se orienta a la aplicación práctica los conceptos.

La unidad uno llamada “UML y el proceso unificado”, está enfocada en retomar a manera de recordatorio los conocimientos adquiridos sobre UML en la materia Análisis y Modelado de Sistemas de Información.

En la segunda unidad, denominada “Diseño de sistemas”, se abordarán los aspectos teóricos del diseño estructurado de sistemas, los diferentes tipos de diagramas de implementación así como el diseño de las interfaces de usuario considerando tecnologías actuales y análisis de las emergentes; se consideran también en esta unidad métricas de diseño en la implementación de los sistemas de información.

En la unidad tres, “Implementación”, se abordarán los aspectos teórico-metodológicos de la elaboración de un sistema informático se estudian las condiciones del desarrollo de software y las formas en que se puede reutilizar algunas partes de éste y cómo se realiza la

¹ Sistema de Asignación y Transferencia de Créditos Académicos

documentación de los sistemas informáticos.

Por último la unidad cuatro, "Verificación y validación", considera el desarrollo de las diferentes formas de evaluar los sistemas de información así como la comprobación de éstos para un mejor desempeño, dando a conocer las diferentes formas de evaluación de software y su posible mantenimiento para disminuir fallas.

El enfoque sugerido para la materia requiere de actividades prácticas que propicien procesos intelectuales como la inducción, la deducción y análisis, con la intención de generar una actividad intelectual compleja de los temas.

3.- COMPETENCIAS A DESARROLLAR

<p>Competencias específicas:</p> <ul style="list-style-type: none">▪ Adquirir bases y fundamentos sobre metodologías de desarrollo de software e implementación de éstas.▪ Aplicar conocimientos de herramientas CASE para el desarrollo de sistemas de información.▪ Aplicar técnicas de uso de métricas en la elaboración de sistemas de información.▪ Aplicar técnicas actuales orientadas al mantenimiento de sistemas de información.▪ Aplicar los conocimientos adquiridos en la elaboración de un programa de implementación.▪ Conocer y aplicar las diferentes métricas de diseño▪ Conocer y aplicar adecuadamente la simbología UML.▪ Conocer y aplicar las características de mantenimiento de software.	<p>Competencias genéricas:</p> <p><u>Competencias instrumentales</u></p> <ul style="list-style-type: none">• Capacidad de análisis y síntesis.• Capacidad de organizar y planificar.• Comunicación oral y escrita en su propia lengua.• Conocimiento de una segunda lengua.• Habilidades de gestión de información (habilidad para buscar y analizar información proveniente de fuentes diversas).• Toma de decisiones. <p><u>Competencias interpersonales</u></p> <ul style="list-style-type: none">• Capacidad crítica y autocrítica.• Trabajo en equipo.• Habilidades interpersonales.• Capacidad de trabajar en equipo interdisciplinario.• Capacidad de comunicarse con profesionales de otras áreas.• Compromiso ético. <p><u>Competencias sistémicas</u></p> <ul style="list-style-type: none">• Habilidades de investigación.• Capacidad de aprender.• Capacidad de adaptarse a nuevas situaciones.• Capacidad de generar nuevas ideas (creatividad).• Liderazgo.• Conocimiento de culturas y costumbres de otros países.• Habilidad para trabajar en forma autónoma.• Capacidad para diseñar y gestionar proyectos.• Iniciativa y espíritu emprendedor.• Preocupación por la calidad.• Búsqueda del logro.	
--	--	--

4.- HISTORIA DEL PROGRAMA

Lugar y fecha de elaboración o revisión	Participantes	Evento
<p>Instituto Tecnológico de Saltillo del 5 al 9 de octubre de 2009.</p>	<p>Representantes de los Institutos Tecnológicos de: Apizaco, Cerro Azul, Chetumal, Ciudad Juárez, Ciudad Madero, Superior de Coahuila de Zaragoza, Colima, Comitancillo, Conkal, Durango, El Llano de Aguascalientes, El Salto, Superior de Fresnillo, Huejutla, Superior de Lerdo, Linares, Los Mochis, Mexicali, Morelia, Oaxaca, Superior del Occidente del Estado de Hidalgo, Ocotlán, Orizaba, Piedras Negras, Pinotepa, Saltillo, San Luis Potosí, Tapachula, Tijuana, Torreón, Tuxtepec, Superior de Valladolid, Valle del Guadiana, Superior de Zacapoaxtla y Zacatecas.</p>	<p>Reunión Nacional de Diseño e Innovación Curricular para el Desarrollo y Formación de Competencias Profesionales de la Carrera de Ingeniería Informática.</p>
<p>Desarrollo de Programas en Competencias Profesionales por los Institutos Tecnológicos del 12 de octubre de 2009 al 19 de febrero de 2010.</p>	<p>Academias de Ingeniería Informática de los Institutos Tecnológicos de: Piedras Negras y San Luis Potosí.</p>	<p>Elaboración del programa de estudio propuesto en la Reunión Nacional de Diseño Curricular de la Carrera de Ingeniería Informática.</p>
<p>Instituto Tecnológico Superior de Poza Rica del 22 al 26 de febrero de 2010.</p>	<p>Representantes de los Institutos Tecnológicos de: Apizaco, Cerro Azul, Chetumal, Ciudad Juárez, Ciudad Madero, Superior de Coahuila de Zaragoza, Colima, Comitancillo, Conkal, Durango, El Llano de Aguascalientes, El Salto, Superior de Fresnillo, Huejutla, Superior de Lerdo, Los Mochis, Mexicali, Morelia, Oaxaca, Superior del Occidente del Estado de Hidalgo, Ocotlán, Orizaba, Piedras Negras, Pinotepa, Saltillo, San Luis Potosí, Tapachula, Tijuana, Torreón, Tuxtepec, Superior de Valladolid, Valle del Guadiana, Superior de Zacapoaxtla y Zacatecas.</p>	<p>Reunión Nacional de Consolidación de los Programas en Competencias Profesionales de la Carrera de Ingeniería Informática.</p>

5.- OBJETIVO GENERAL DEL CURSO

Aplicar métodos de ingeniería de software y herramientas CASE para diseñar, desarrollar e implementar sistemas de información.

6.- COMPETENCIAS PREVIAS

- Conocer el modelo de proceso de software.
- Identificar áreas de oportunidad en una organización, para la propuesta y diseño de sistemas de información
- Analizar diversas alternativas de solución a partir de la identificación y definición de requerimientos especificados por el cliente.
- Establecer una propuesta para el análisis y diseño de un proyecto de software de acuerdo a la alternativa de solución planteada o establecida.
- Planificar y gestionar proyectos de sistemas de información con base en una metodología de desarrollo.
- Aplicar principios de ingeniería del software en las etapas de análisis y diseño de un sistema de información.
- Modelar casos de uso acorde a los requerimientos del proyecto.
- Documentar el proyecto.

7.- TEMARIO

Unidad	Temas	Subtemas
1.	UML y el proceso unificado	<ul style="list-style-type: none">1.1. Conceptualización de UML.<ul style="list-style-type: none">1.1.1. Las primeras metodologías.1.1.2. Surgimiento de UML.1.1.3. Visión general.1.2. Estandarización de UML.<ul style="list-style-type: none">1.2.1. Vistas.1.2.2. Diagramas.1.2.3. Elementos de modelado.1.2.4. Mecanismos.1.2.5. Extensiones a UML.1.3. Herramientas CASE para el desarrollo y modelado de sistemas de información.<ul style="list-style-type: none">1.3.1. Definiciones.1.3.2. Clasificación.1.4. Diagramas.<ul style="list-style-type: none">1.4.1. Actividad.1.4.2. Modelado a distintos niveles.1.4.3. Caso de uso.1.4.4. Relación con los requisitos.1.5. Utilización de diversas herramientas CASE.<ul style="list-style-type: none">1.5.1. Planificación de sistemas de gestión.1.5.2. Gestión de proyectos.1.5.3. Soporte.1.5.4. Análisis y diseño.1.5.5. Programación.1.5.6. Integración y prueba.1.5.7. Creación de prototipos.

		1.5.8. Mantenimiento.
2.	Diseño de sistemas	<ul style="list-style-type: none"> 2.1. Diseño estructurado de sistemas. <ul style="list-style-type: none"> 2.1.1. Conceptos básicos. 2.1.2. Diagramas de flujo de datos. 2.1.3. Ampliaciones para sistemas de tiempo real 2.2. Diagramas de interacción de objetos. <ul style="list-style-type: none"> 2.2.1. De secuencia. 2.2.2. De colaboración. 2.3. Modelos de clases. <ul style="list-style-type: none"> 2.3.1. Clases. <ul style="list-style-type: none"> 2.3.1.1. Dispositivo. 2.3.1.2. Propiedad. 2.3.1.3. Interacción. 2.3.2. Características. 2.3.3. Estructuras y jerarquías. 2.3.4. Subsistemas. 2.4. Diagramas de implementación. <ul style="list-style-type: none"> 2.4.1. Definición. 2.4.2. Objetivo. 2.4.3. Tipos. <ul style="list-style-type: none"> 2.4.3.1. De componentes. 2.4.3.2. De ejecución 2.4.4. Aplicaciones. 2.4.5. Adaptación de UML. 2.5. Diseño de la interfaz de usuario. <ul style="list-style-type: none"> 2.5.1. Interacción hombre máquina. 2.5.2. Diseño de interfaz hombre/máquina. 2.5.3. Directrices para el diseño de interfaces. 2.5.4. Estándares de interfaz. 2.6. Diseño de la base de datos. <ul style="list-style-type: none"> 2.6.1. Objetivos. 2.6.2. Almacén de datos. 2.7. Métricas del diseño. <ul style="list-style-type: none"> 2.7.1. Factores que afectan. 2.7.2. Productividad. 2.7.3. Medidas relacionadas. <ul style="list-style-type: none"> 2.7.3.1. Tamaño. 2.7.3.2. Función. 2.7.3.3. Puntos de objeto. 2.7.4. Métricas de diseño arquitectónico. 2.7.5. Métricas a nivel de componentes. 2.7.6. Métricas de diseño de interfaz.
3.	Implementación	<ul style="list-style-type: none"> 3.1. Elaboración de un programa de implementación. <ul style="list-style-type: none"> 3.1.1. Objetivo. 3.2. Desarrollo del software basado en procesos ágiles. <ul style="list-style-type: none"> 3.2.1. Definición de procesos ágiles. 3.2.2. Modelos ágiles de procesos.

		<ul style="list-style-type: none"> 3.3. Reutilización del software. <ul style="list-style-type: none"> 3.3.1. Usos de reutilización. 3.3.2. Patrones de diseño. 3.3.3. Basada en generadores. 3.3.4. Marcos de trabajo. 3.3.5. Sistemas de aplicaciones. 3.4. Documentación. <ul style="list-style-type: none"> 3.4.1. Objetivo e importancia. 3.4.2. Tipos.
4.	Verificación y validación	<ul style="list-style-type: none"> 4.1. Pruebas. <ul style="list-style-type: none"> 4.1.1. Objetivo. 4.1.2. Justificación. 4.2. Tipos de Pruebas. <ul style="list-style-type: none"> 4.2.1. Integración. <ul style="list-style-type: none"> 4.2.1.1. Descendente. 4.2.1.2. Ascendente. 4.2.1.3. Regresión 4.2.2. Validación. <ul style="list-style-type: none"> 4.2.2.1. Alfa. 4.2.2.2. Beta. 4.2.3. Sistema. <ul style="list-style-type: none"> 4.2.3.1. Recuperación. 4.2.3.2. Seguridad. 4.2.3.3. Resistencia. 4.2.3.4. Rendimiento. 4.3. Mantenimiento. <ul style="list-style-type: none"> 4.3.1. Concepto. 4.3.2. Objetivo. 4.4. Características del mantenimiento. <ul style="list-style-type: none"> 4.4.1. Costos. 4.4.2. Efectos. 4.4.3. Tipos. <ul style="list-style-type: none"> 4.4.3.1. Correctivo. 4.4.3.2. Preventivo/perfectivo. 4.4.3.3. Adaptativo. 4.4.4. Actividades.

8.- SUGERENCIAS DIDÁCTICAS

El docente debe:

- Propiciar actividades de búsqueda, selección y análisis de información en distintas fuentes.
- Propiciar el uso de las nuevas tecnologías en el desarrollo de los contenidos de la asignatura.
- Desarrollar un sistema de información.
- Fomentar actividades grupales que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración y la colaboración de y entre los estudiantes.
- Propiciar, en el estudiante, el desarrollo de actividades intelectuales de inducción-deducción y análisis-síntesis, las cuales lo encaminan hacia la investigación, la aplicación de conocimientos y la solución de problemas.
- Desarrollar actividades de aprendizaje que propicien la aplicación de los conceptos, modelos y metodologías que se van aprendiendo en el desarrollo de la asignatura.
- Propiciar el uso adecuado de conceptos, y de terminología científico-tecnológica
- Proponer problemas que permitan al estudiante la integración de contenidos de la asignatura y entre distintas asignaturas, para su análisis y solución.
- Relacionar los contenidos de la asignatura con el cuidado del medio ambiente; así como con las prácticas de una ingeniería con enfoque sustentable.
- Observar y analizar fenómenos y problemáticas propias del campo ocupacional.
- Relacionar los contenidos de esta asignatura con las demás del plan de estudios para desarrollar una visión interdisciplinaria en el estudiante.

9.- SUGERENCIAS DE EVALUACIÓN

La evaluación de la asignatura debe de ser continua y formativa, por lo que debe de considerarse el desempeño de cada una de las actividades de aprendizaje, haciendo especial énfasis en:

- Reportes escritos derivados de la unidad temática desarrollada en ese momento.
- Exámenes escritos de teoría manejada en el curso.
- Exposiciones de las investigaciones realizadas acorde a los temas del curso.
- Realizar el reporte del seguimiento del proyecto.
- Participar en foros académicos en donde se expongan los proyectos a la comunidad estudiantil.
- Entregar el sistema de información y documentación correspondiente (Manual técnico, usuario).

10.- UNIDADES DE APRENDIZAJE

Unidad 1: UML y el proceso unificado

<i>Competencia específica a desarrollar</i>	<i>Actividades de Aprendizaje</i>
Adquirir bases y fundamentos sobre metodologías de desarrollo de software e implementación de éstas. Aplicar conocimientos de herramientas CASE para el desarrollo de sistemas de información. Conocer y aplicar adecuadamente la simbología UML.	<ul style="list-style-type: none">• Analizar y discutir los principales conceptos de UML.• Investigar la estandarización de UML y sus aplicaciones en el desarrollo de sistemas de información.• Realizar diferentes reportes de investigación documental.

Unidad 2: Diseño de sistemas

<i>Competencia específica a desarrollar</i>	<i>Actividades de Aprendizaje</i>
Adquirir bases y fundamentos sobre metodologías de desarrollo de software e implementación de éstas. Conocer y aplicar adecuadamente la simbología UML.	<ul style="list-style-type: none">• Investigar en qué consiste el diseño estructurado de sistemas.• Analizar y discutir la metodología de diseño de interface de usuario, interacción hombre-máquina y dispositivos de acceso a sistemas de información.• Realizar prácticas que permitan complementar los ejercicios vistos en clase.• Como alternativa se puede realizar una invitación a un profesional del área de informática que comparta sus experiencias y recomendaciones sobre el diseño de sistemas de información.

Unidad 3: Implementación

<i>Competencia específica a desarrollar</i>	<i>Actividades de Aprendizaje</i>
Aplicar técnicas de uso de métricas en la elaboración de sistemas de	<ul style="list-style-type: none">• Investigar en fuentes bibliográficas diferentes métodos y técnicas para el

<p>información. Aplicar los conocimientos adquiridos en la elaboración de un programa de implementación. Conocer y aplicar adecuadamente la simbología UML.</p>	<p>desarrollo del software basado en procesos ágiles, y con ellos realizar un análisis comparativo que se expondrá frente al grupo y generar un foro de discusión.</p> <ul style="list-style-type: none"> • Analizar e identificar, las ventajas y problemas que se pueden presentar en la reutilización de software. • Analizar e identificar toda aquella información que es necesario documentar para un mejor entendimiento del sistema en el futuro. • El análisis y comprensión de las lecturas, deberá plasmarse en el desarrollo y documentación del sistema de información que se desarrollará como proyecto integrador en la materia.
---	--

Unidad 4: Verificación y validación

<i>Competencia específica a desarrollar</i>	<i>Actividades de Aprendizaje</i>
<p>Aplicar conocimientos de herramientas CASE para el desarrollo de sistemas de información. Aplicar técnicas de uso de métricas en la elaboración de sistemas de información. Aplicar técnicas actuales orientadas al mantenimiento de sistemas de información. Aplicar los conocimientos adquiridos en la elaboración de un programa de implementación. Conocer y aplicar las diferentes métricas de diseño. Conocer y aplicar adecuadamente la simbología UML. Conocer y aplicar las características de mantenimiento de software</p>	<ul style="list-style-type: none"> • Analizar lecturas de artículos, y libros acerca de los diversos tipos de pruebas para el software. • Seleccionar de acuerdo a los resultados de los análisis considerados en el punto anterior, el(los) tipo(s) de prueba(s) para el sistema de información que se desarrollará como proyecto final de la asignatura. • Que el alumno investigue en portales especializados casos de éxito en la aplicación de metodologías de pruebas y si es posible, realizar ejercicios de análisis colaborativos de las lecturas utilizando herramientas Web (Ej. Blogs o Wikis). • Desarrollar pruebas al sistema de información que se estudio y/o desarrollo para detectar y corregir fallas.

11.- FUENTES DE INFORMACIÓN

1. Bernd Bruegge, Allen H. Dutoit. *Ingeniería de Software Orientado a Objetos*. Prentice Hall.
2. Ian Sommerville; *Ingeniería de Software*, Edit. Addison Wesley; 2005.
3. James Rumbaugh, Ivar Jacobson, Grady Booch. *El Lenguaje Unificado de Modelado Manual de Referencia*. Addison Wesley.
4. Kenneth C. Lawden, Jane P. Lawden. *Administración de Los Sistemas de Información, Organización y Técnicas*.
5. Laudon, K.; Laudon, J.; *Sistemas de Información Gerencial*. Administración de la Empresa Digital; 10ª Edición; Edit. Pearson Prentice Hall. 2008.
6. Roger S. Pressman; *Ingeniería de software un Enfoque practico*; Edit. Mc. Graw Hill; 2007.
7. Senn A. James. *Analisis y Diseño de Sistemas de Información*. Addison Wesley.
8. Shari Lawrence Pfleeger. *Ingeniería de Software Teoría y Práctica*. Prentice Hall.
9. Alfredo Weitzenfeld. *Ingeniería de Software Orientada a Objetos con UML, Java e Internet*. Edit. Thomson. 2007.
10. IEEE computer Society Software Engineering: <http://www.computer.org/portal/web/seonline/home>
11. Pagina de autor libro de Ingeniería de software Ian Sommerville: <http://www.cs.st-andrews.ac.uk/~ifs/>
12. Advances in software Engineering: <http://www.hindawi.com/journals/ase/>

12.- PRÁCTICAS PROPUESTAS

Las Prácticas estarán basadas en un análisis de los casos de éxito documentados, investigaciones, utilización de las herramientas apropiadas para cada uno de los ejemplos.

- Utilizar alguna herramienta de software:
 - para administración de proyectos y así realizar el estudio de factibilidad de un caso práctico.
 - recolección de requerimientos para identificar y clasificar requerimientos de sistemas.
 - análisis de requerimientos y complementar el análisis y documentación del sistema.
 - realizar el prototipo del sistema.
- Investigar las métricas de evaluación y los estándares que las rigen.
- Medir los requerimientos del proyecto de acuerdo con las diferentes métricas de evaluación del análisis.
- Recabar la documentación correspondiente a la etapa de diseño.
- Que los equipos expongan los avances del proyecto en el grupo para que el profesor asesore y marque los ajustes necesarios. De esta manera se comparten ideas y experiencias fomentando que los proyectos al final del curso.
- Desarrollar sistema modelado anteriormente.
- Realizar las pruebas para verificar el funcionamiento del sistema de información aplicando métricas de evaluación y estándares.
- Recabar la documentación correspondiente a la etapa de construcción.
- Crear un equipo de trabajo que evalúe sistemas desarrollados por otros equipos.

Al final del curso se organice un foro en donde se expongan los proyectos a toda la comunidad estudiantil, fomentando en los estudiantes habilidades para hablar en público y convencer a los clientes de la factibilidad de sus productos de software.