

Cartas Descriptivas

Datos de identificación				
Unidad Académica	Facultad de Contaduría y Administración (Tijuana). Facultad de Ciencias Administrativas y Sociales (Ensenada). Facultad de Ciencias Administrativas (Mexicali).			
Programa	Maestría en Gestión de Tecnologías de la Información y la Comunicación			
Nombre de la asignatura	Ingeniería de Software			
Tipo de Asignatura	Optativa			
Clave (Posgrado e Investigación)	1040			
Horas teoría	2	Horas laboratorio	0	Créditos Totales
Horas taller	2	Horas prácticas de campo	0	6
Perfil de egreso del programa				
El egresado de la Maestría en Gestión de Tecnologías de la Información y la Comunicación tendrá la capacidad de planear, organizar, desarrollar, dirigir, controlar, proponer e implementar proyectos tecnológicos innovadores, en las organizaciones, generando soluciones que contribuyan a la competitividad de las mismas, mediante la aplicación de metodologías y técnicas vanguardistas en el ámbito de las TIC dentro de un marco social ético, responsable y sostenible.				
Definiciones generales de la asignatura				
Aportación de esta materia al perfil de egreso del estudiante.	La capacidad de dirigir el proceso de desarrollo y documentación de software, mediante el uso de metodologías y técnicas adecuadas para lograr un producto de calidad con el estudio, comprensión y evaluación de técnicas y herramientas de modelado de ingeniería de software.			
Descripción de la orientación de la asignatura en coherencia con el perfil de egreso.	La asignatura pertenece al área de énfasis de desarrollo de software y tiene como propósito aplicar las estrategias para diseñar el software con base en el análisis y el uso de técnicas para el diseño en la construcción de un producto de software de calidad			
Cobertura de la asignatura.	En esta asignatura se considera el análisis de los conceptos básicos de la ingeniería de software, del proceso de desarrollo de software y modelos; determinar los requerimientos de software; la implementación de estándares de procesos aplicables al desarrollo del software; la aplicación de las estrategias para diseñar y la construir un producto de software de calidad y la validación de software.			
Profundidad de la asignatura.	Desarrollo de aplicaciones de sistemas de información utilizando técnicas y herramientas de ingeniería de software de calidad, que satisfagan necesidades de información específicas de una organización.			

Temario			
Unidad	Objetivo	Tema	Producto a evaluar (evidencia de aprendizaje)
Unidad I. Introducción a la Ingeniería de Software.	El alumno analizará los conceptos básicos de la ingeniería de software para comprender el desarrollo de un proyecto de software.	1.1 La naturaleza del software. 1.2 Conceptos de ingeniería de software. 1.3 Ética en la ingeniería de software. 1.4 Formación de equipos de trabajo. 1.5 Paradigma orientado a objetos. 1.6 Modelos de ciclo de vida del software	Cuadro sinóptico de los conceptos de ingeniería de software.
Unidad II. Procesos del Software.	El alumno analizará el proceso de software y comparar los diferentes modelos de proceso de software integrando buenas prácticas de ingeniería de software para crear procesos de software adaptables.	2.1 Modelos del proceso. 2.1.1 Modelo general de procesos. 2.1.2 Evaluación y mejora de procesos. 2.1.3 Modelos de proceso prescriptivo. 2.1.4 Modelos de proceso especializado. 2.1.5 El proceso unificado. 2.1.6 Modelos del proceso personal y del equipo. 2.2 Desarrollo Ágil. 2.2.1 La agilidad y el costo del cambio. 2.2.2 Métodos ágiles. 2.2.3 Programación extrema. 2.2.4 Otros modelos ágiles de procesos. 2.2.4.1 Scrum. 2.2.4.2 MDSD. 2.2.4.3 DAS. 2.2.4.4 PUA.	Cuadro comparativo de los diferentes modelos de proceso de software.

<p>Unidad III. Requerimientos y modelado.</p>	<p>El alumno podrá determinar los requerimientos de software a través de la indagación de los mismos para el modelado de casos de uso.</p>	<p>3.1 Requerimientos. 3.1.1 Establecer las bases de los requerimientos. 3.1.2 Indagación de requerimientos. 3.1.3 Desarrollo de casos de uso. 3.1.4 Elaboración del modelo de los requerimientos. 3.1.5 Validación de los requerimientos. 3.2 Modelado de los requerimientos. 3.2.1 Análisis de los requerimientos 3.2.2 Modelado basado en escenarios. 3.2.3 Modelos UML que proporcionan el caso de uso. 3.2.4 Conceptos de modelado de datos. 3.2.5 Modelado masado en clases. 3.2.5 Modelado orientado al flujo. 3.2.6 Creación de un modelo de comportamiento. 3.2.7 Patrones para el modelado de requerimientos. 3.2.8 Modelado de requerimientos para web apps.</p>	<p>Análisis comparativo de los diferentes modelos de sistema. Informe del análisis del sistema de información.</p>
<p>Unidad IV. Gestión de la calidad.</p>	<p>El alumno controlará el proceso de desarrollo de software a través de la aplicación de métricas de aseguramiento de calidad de software, así como de la implementación de estándares de procesos aplicables al desarrollo del software, con la finalidad de lograr la construcción de un sistema que cumpla con los requerimientos definidos por el usuario</p>	<p>4.1 Introducción al concepto de gestión de calidad. 4.2 Garantía de calidad y estándares. 4.4 Planificación de la calidad. 4.4 Control de la calidad del software. 4.4.1 Mediciones y métricas de software. 4.4.2 Análisis y modelado de procesos. 4.4.3 Mediciones del proceso. 4.4.4 Calidad del proceso. 4.4.5 El modelo de madurez del proceso (CMM). 4.4.6 El modelo MOPROSOFT. 4.5 Evaluación de la calidad 4.5.1 Evaluación de la madurez de los procesos en el software. 4.5.2 Pruebas de software. 4.5.3 Control de la configuración del software.</p>	<p>Informe donde aplique el Modelo de Procesos de Software (MOPROSOFT) al desarrollo de su proyecto de software. Diferencie los niveles del modelo y adecúelos correctamente.</p> <ol style="list-style-type: none"> 1. Alta dirección 2. Gestión 3. Operación

Unidad V. Diseño e implementación	El alumno aplicará las estrategias para diseñar el software con base en el análisis y el uso de técnicas para el diseño orientado por objetos en la construcción de un producto de software de calidad	5.1 El proceso del diseño. 5.2 Conceptos del diseño. 5.3 El modelo del diseño. 5.4 Diseño de la arquitectura. 5.5 Diseño en el nivel de componentes. 5.6 Diseño de la interfaz de usuario. 5.7 Diseño orientado a objetos. 5.8 Diseño basado en patrones. 5.9 Elección de lenguaje de programación. 5.10 Conflictos de implementación 5.11 Desarrollo de código abierto.	Reporte que contenga el diseño e implementación del sistema de información elaborado.
UNIDAD VI. Pruebas y mantenimiento	El alumno seleccionará el tipo de prueba adecuado para la validación de software, mediante los procesos necesarios para la utilización de las mismas	6.1 Fundamentos de las pruebas de software. 6.2 Prueba de ruta básica. 6.3 Prueba basada en modelo. 6.3 Pruebas de versión. 6.4 Pruebas de usuario. 6.5 Pruebas de aplicaciones orientadas a objetos. 6.6 Pruebas de aplicaciones web. 6.7 Validación y verificación. 6.8 Mantenimiento.	Informe acerca de las diferentes pruebas de software.
Unidad VII. Ingeniería de Software Avanzada	El alumno demostrara las ventajas y desventajas de cada una de las ingenierías de software avanzadas, a través de la un estudio comparativo para seleccionar la mejor, de acuerdo a la necesidad del cliente	7.1 Reutilización de software. 7.2 Ingeniería de software basada en componentes. 7.3 Ingeniería de software distribuido. 7.4 Arquitectura orientada a servicios.	Investigación y cuadro comparativo de las ingenierías de software avanzadas.

Estrategias de aprendizaje utilizadas:

- Exposición y orientación del profesor.
- Lecturas complementarias.
- Investigación de temas relacionados al curso.
- Evaluación de caso práctico.

Métodos y estrategias de evaluación:

- Participación en clase: 10%
- Trabajos de Investigación: 30%
- Proyecto final: 60%

Bibliografía:

- Sian Sommerville., Ingeniería de Software. Pearson. Novena edición. Año: 2011. ISBN: 978-607-32-0603-7
- Roger S. Pressman. Ingeniería de Software. Un enfoque práctico. McGrawHill. Séptima edición. 2010. ISBN: 978-607-15-0314-5.
- Stephen R. Scach. Ingeniería de Software clásica y orientada a objetos. McGrawHill. Sexta edición. 2006. ISBN: 0-07-286551-2
- Mario G. Piattini Velthuis, Félix Oscar García Rubio, Javier Garzás Parra, Marcela Fabiana Genero Bocco. Medición y estimación del software. Técnicas y métodos para mejorar la calidad y la productividad. Alfaomega. 2008. ISBN: 978-970-15-1413-9

Nombre y firma de quién diseñó carta descriptiva:

- Dra. Maricela Sevilla Caro¹.
- Dra. María del Consuelo Salgado Soto¹
- Dra. Hilda Beatriz Ramírez Moreno¹.

¹ Cuerpo Académico Sistemas de Información y Gestión empresarial

Nombre y firma de quién autorizó carta descriptiva:

- Dr. Sergio Octavio Vázquez Núñez
Director de la Facultad de Contaduría y Administración – Tijuana
- Dra. Mónica Lacavex Berumen
Director de la Facultad de Ciencias Administrativas y Sociales – Ensenada
- Dr. Raúl González Núñez
Director de la Facultad de Ciencias Administrativas – Mexicali

Nombre(s) y firma(s) de quién(es) evaluó/revisó(evaluaron/ revisaron) la carta descriptiva:

- Dra. Margarita Ramírez Ramírez
Coordinadora de Posgrado de la Facultad de Contaduría y Administración-Tijuana
- Dr. Ariel Moctezuma Hernández
Coordinador de Posgrado de la Facultad de Ciencias Administrativas y Sociales-Ensenada
- Dr. Manuel Alejandro Ibarra Cisneros
Coordinador de Posgrado de la Facultad de Ciencias Administrativas-Mexicali