



UNIVERSIDAD AUTÓNOMA DE SINALOA
SECRETARÍA ACADÉMICA UNIVERSITARIA
 Coordinación General de Evaluación, Innovación y Calidad Educativa
UNIDAD ACADÉMICA: FACULTAD DE INFORMÁTICA CULIACÁN

PROGRAMA DE ESTUDIOS

1. DATOS DE IDENTIFICACIÓN			
UNIDAD DE APRENDIZAJE	Análisis y diseño de software		
Clave:	4420		
Horas y créditos:	Teóricas: 55	Prácticas: 25	Estudio Independiente: 40
	Total de horas: 120		Créditos: 8
Tipo de unidad de aprendizaje:	Teórico:	Teórico-práctico x	Práctico
Competencia (s) del perfil de egreso que desarrolla o a las que aporta.	<ul style="list-style-type: none"> • Selecciona, analiza, diseña, desarrolla y evalúa aplicaciones y sistemas informáticos, asegurando la satisfacción de todos los requisitos del usuario, su fiabilidad, seguridad y calidad, conforme a principios éticos, estándares de calidad y a la legislación y normativa vigente. • Aplica los conocimientos matemáticos en el área de ingeniería. • Analiza y entiende los sistemas inherentes a este proceso. • Analiza la diversa problemática que se presenta durante el proceso de comunicación. 		
Cursos consecuentes relacionados:	<ul style="list-style-type: none"> • Desarrollo en implementación de sistemas • Sistemas de información 	Cursos subsecuentes relacionados:	<ul style="list-style-type: none"> • Análisis y Diseño Orientado a Objetos • Fundamentos de Base de Datos.
Responsables de elaborar y/o actualizar el programa:	Dr. Roberto Bernal Guadiana, MC. José de Jesús Adrián Uriarte, MC. Zeus del Valle Castillo Najera, Dra. Xiomara Penelope Zaldivar Colado, Dr. Ulises Zaldivar Colado		
Fecha de elaboración:	Junio 2011	Actualización: Junio 2018	
2. PROPÓSITO			
El alumno realizará una propuesta de solución de un problema a través de los fundamentos de administración de proyectos para el análisis y diseño orientado a objetos.			
3. SABERES			
Teóricos:	<ul style="list-style-type: none"> • Identifica el proceso de la planeación para el desarrollo de Software (objetivos, metas, recursos, actividades, tiempos, roles, políticas). Identifica una herramienta de gestión de proyectos. • Identifica las funciones y tareas del equipo de trabajo que interviene en el proceso de desarrollo de desarrollo de Software. 		

	<ul style="list-style-type: none"> • Identifica las acciones pertinentes para orientar de la mejor manera un proyecto de desarrollo de Software. • Describe las diferentes métricas para la evaluación de los proyectos de Software.
Prácticos:	<ul style="list-style-type: none"> • Elabora el plan de trabajo que desarrolle un proyecto de TIC, utilizando herramientas de gestión de proyectos. • Asigna las tareas y funciones necesarias para el desarrollo de software. • Propone las acciones de coordinación para el desarrollo del proyecto de software. • Compara los resultados obtenidos contra las métricas establecidas en el plan de trabajo.
Actitudinales:	Líder, Analítico, responsable, Sistemático, Coherente, Visionario, Capaz de comunicarse claramente, honesto, ético, Crítico, Hábil para trabajar en equipo.

4. CONTENIDOS

BLOQUE I:	Aprendizajes Esperados
<p>Conceptos básicos.</p> <p>1.1. Importancia del Análisis y el Diseño.</p> <p style="padding-left: 20px;">1.1.1 Fundamentación.</p> <p style="padding-left: 20px;">1.1.2 Objetivos del análisis.</p> <p style="padding-left: 20px;">1.1.3 Objetivos del diseño.</p> <p>1.2. El análisis y el diseño en el contexto de las metodologías de la Ingeniería de Software.</p> <p style="padding-left: 20px;">1.3.1 El ciclo de vida del desarrollo de software.</p> <p style="padding-left: 20px;">1.3.2 El enfoque en cascada.</p> <p style="padding-left: 20px;">1.3.2 El enfoque iterativo e incremental.</p> <p style="padding-left: 20px;">1.3.3 El enfoque en los métodos ágiles.</p> <p>1.3. Técnicas y herramientas.</p> <p style="padding-left: 20px;">1.4.1 Uso de patrones para análisis y diseño.</p> <p style="padding-left: 20px;">1.4.2 Herramientas CASE.</p>	<p>Refuerza la comprensión de tópicos y conceptos relacionados con el proceso de desarrollo del software (Ciclo de vida del software) haciendo énfasis en lo referente al análisis y diseño del software.</p>
<p>BLOQUE II:</p> <p>El proceso de análisis.</p> <p>2.1 La importancia del Análisis.</p> <p>2.2 El modelo de análisis.</p> <p style="padding-left: 20px;">2.2.1 Modelado del dominio.</p> <p style="padding-left: 40px;">2.2.1.1 Objetivo.</p> <p style="padding-left: 40px;">2.2.1.2. Clases conceptuales, atributos y relaciones.</p> <p style="padding-left: 40px;">2.2.1.3. Diagramas UML.</p> <p style="padding-left: 60px;">2.2.1.3.1. El diagrama de clases durante el análisis.</p>	<p>Conoce y aplica técnicas de análisis OO de software para la construcción del modelo de análisis.</p>

<ul style="list-style-type: none"> 2.2.1.3.2. El diagrama de actividad durante el análisis. 2.2.1.3.3. El diagrama de máquina de estados. 2.2.1.4. Técnicas de modelado. 2.2.2 Glosario de términos. 2.3 La gestión de los requisitos. <ul style="list-style-type: none"> 2.3.1 Requisitos funcionales y no funcionales. 2.3.2 Descubrimiento. 2.3.3 Especificación. <ul style="list-style-type: none"> 2.3.3.1 Casos de uso. <ul style="list-style-type: none"> 2.3.4.1.1 Conceptos. 2.3.4.1.2 Formatos de escritura. 2.3.4.1.3 Diagrama de casos de uso. 2.3.3.2 Las historias de uso (historias de usuario). 2.3.4 Operaciones del sistema. <ul style="list-style-type: none"> 2.3.4.1 Contratos. 2.3.5 Uso de prototipos. 2.3.6 Priorización de los requisitos. 2.4 Prototipo de la interfaz del usuario. 	
<p>BLOQUE III:</p>	
<p>El proceso de diseño.</p> <ul style="list-style-type: none"> 3.1 Objetivos del diseño. 3.2 Transición del análisis al diseño. <ul style="list-style-type: none"> 3.2.1 El diseño dirigido por las responsabilidades. <ul style="list-style-type: none"> 3.2.1 El concepto de responsabilidad. 3.2.2 Diseño de las colaboraciones. 3.3 Perspectivas estáticas y dinámicas del diseño. <ul style="list-style-type: none"> 3.3.1 Modelado estático. <ul style="list-style-type: none"> 3.3.1.1 Clases de diseño y clases de software. <ul style="list-style-type: none"> 3.3.1.2 Agrupamientos de clases. <ul style="list-style-type: none"> 3.3.1.2.1 Modelado de paquetes. <ul style="list-style-type: none"> 3.3.1.2.1.1 El diagrama de paquetes de UML. 3.3.1.2.2 Patrones para la cohesión y dependencia de paquetes. 	<p>Conoce y aplica técnicas básicas de diseño OO de software para la creación de los productos del diseño y expresarlos utilizando el lenguaje de modelado UML.</p>

<p>3.3.2 Modelado dinámico.</p> <p>3.3.2.1 Diseño de colaboraciones.</p> <p>3.3.2.1.1 Asignación de responsabilidades.</p> <p>3.3.2.1.2 Patrones GRASP.</p> <p>3.3.2.1.3 Las tarjetas CRC.</p> <p>3.3.2.1.4 Definición de las relaciones entre clases.</p> <p>3.3.2.2 UML para el diseño.</p> <p>3.3.2.2.1 Diagramas de secuencia y colaboración.</p> <p>3.3.2.2.2 Diagrama de máquina de estados.</p> <p>3.3.2.2.3 Diagrama de actividad.</p> <p>3.4.2.3 Principios SOLID de diseño .</p> <p>3.4.2.4 Patrones de diseño (Gang of 4).</p> <p>3.4 Modelado de la arquitectura del software.</p> <p>3.4.1 Conceptos sobre arquitectura del software.</p> <p>3.4.2 Modelos y estilos arquitectónicos.</p> <p>3.4.3 UML para el modelado de la arquitectura.</p> <p>3.4.3.1 El diagrama de paquetes.</p> <p>3.4.3.2 El Diagrama de componentes.</p>	
<p>BLOQUE IV:</p>	
<p>Diseño de la fuente de datos.</p> <p>4.1 Clases, relaciones y persistencia.</p> <p>4.2 UML para el diseño de la fuente de datos.</p> <p>4.3 Generación de scripts para la creación de la fuente de datos.</p> <p>4.4 Refactorización de la fuente de datos.</p>	<p>Aplica la práctica de desarrollo iterativo e incremental al modelo relacional de la fuente de datos.</p>
<p>BLOQUE V:</p>	
<p>Proyecto final.</p> <p>5.1 Asignación del proyecto.</p> <p>5.2 Programación de revisiones de avances.</p>	<p>Muestra el dominio de las técnicas adquiridas durante el curso en un proyecto diseñado con dicho objetivo.</p>
<p>5. ACTIVIDADES PARA DESARROLLAR LAS COMPETENCIAS</p>	

Actividades del maestro.

- Actividades de inicio: técnica expositiva, conferencia, la pregunta.
 - Actividad de desarrollo: panel, simposio, investigación bibliográfica, estudio supervisado, diálogo, mesa redonda con moderador y/o relator,
- Actividad de evaluación: informe de investigación documental, ensayo, mapa conceptual, cuadro sinóptico, cuadro comparativo, portafolio de evidencias, rúbrica.

Actividades del estudiante.

- Actividades de inicio: Diario, fichas de trabajo, memoria, lluvia de ideas.
 - Actividades de desarrollo: concordar y discordar,
- Actividades finales: Informe de investigación documental o de campo, ensayo, mapa conceptual, cuadro sinóptico, cuadro comparativo, portafolio de evidencias, rúbrica.

6. EVALUACIÓN DE LAS COMPETENCIAS

6.1. Evidencias de aprendizaje	6.2. Criterios de desempeño	6.3. Calificación y acreditación
Evidencias: Exámenes escritos, Trabajos de investigación, Trabajo-prácticas parciales y trabajo final integrador	Al final de cada tema, maestro y alumnos, revisarán el avance del curso y la comprensión lograda hasta el momento.	Se obtendrá de los puntos acumulados en el curso o del porcentaje asignado a los exámenes aplicados y/o trabajos de investigación y/o a la práctica final, sin perder de vista la competencia a lograr. Sobresaliente (10), Competente (9), Suficiente (8), Básico (7-6), No Competente (5 o menor)

7. FUENTES DE INFORMACIÓN**Bibliografía:**

- 1 Applying UML and Patterns An Introduction to O O Analysis and Design and the Unified Process 2nd Ed - Larman - PH PTR 2002.
- 2 Software Requirements (3rd Edition) Karl Weigers, Joy Beatty. Microsoft Press.
- 3 Mastering the Requirements Process 2nd Edition - Robertson,Robertson - Addison Wesley(2006)
- 4 Extreme Programming Explained: Embrace Change, Second Edition, Kent Beck, Cynthia Andres, Addison Wesley Professional (2004)
- 5 Análisis y Diseño OO de Sistemas usando UML, Bennet, McRobb, Farmer, 3^a. Edición, McGraw Hill.

8. PERFIL DEL PROFESOR:

- Profesionista en el área de la Informática o posgrado afín
- Facilidad de comunicación o expresión tanto oral como escrita
- Capacidad de generar y transmitir el conocimiento
- Interés por los temas relacionados con su especialidad

- Capacidad para desarrollar e implementar actividades educativas que faciliten el aprendizaje del alumnado.
- Dinamismo e interés por los temas desarrollados y capacidad de transmitirlo a sus estudiantes.
- Capacidad para motivar y propiciar el trabajo en grupos.
- Puntualidad y Responsabilidad