



UNIVERSIDAD AUTÓNOMA DE SINALOA

FACULTAD DE INGENIERÍA MOCHIS LICENCIATURA EN INGENIERÍA EN SOFTWARE

PROGRAMA DE ESTUDIO



1. DATOS DE IDENTIFICACIÓN			
UNIDAD DE APRENDIZAJE O MÓDULO:	METODOLOGÍAS TRADICIONALES DE DESARROLLO DE SOFTWARE		
Clave:			
Ubicación:	Semestre IV	Área: Profesionalizante	
Horas y créditos:	Teóricas: 40	Prácticas: 40	Estudio Independiente: 80
	Total de horas: 160		Créditos: 10
Competencia(s) del perfil de egreso al que aporta:	<p>CG3. Ejerce su conocimiento ponderando los valores éticos para brindar mayores beneficios a la comunidad, con respeto a la ley y los códigos que dirigen su desempeño.</p> <p>CG7. Cultiva el compañerismo, el trabajo en equipo y la coordinación de esfuerzos bajo la aspiración de mejorar las tareas académicas, los entornos laborales y la convivencia social en beneficio para la consecución de metas que impactan en las formas de entablar y mantener relaciones humanas positivas.</p> <p>CE12. Seleccionar estándares de desarrollo de las etapas de diseño, codificación, pruebas y mantenimiento para resolver un caso de estudio, analizando sus características.</p> <p>CE14. Analizar los diferentes métodos de desarrollo de software para comprender cómo se relacionan con los elementos del ciclo de vida del software, identificando las características de cada método en un caso de estudio y aplica métricas para la estimación del software (tamaño, costo, esfuerzo, personal, tiempo, productividad, calidad y documentación) conforme a los modelos de ciclos de vida de los sistemas.</p> <p>CE15. Aplica teoría, técnicas y metodologías para la construcción de una amplia gama de soluciones de información, imprescindibles para el adecuado funcionamiento de todo tipo de organizaciones.</p> <p>CE20. Desarrolla software para diferentes tipos de aplicaciones utilizando técnicas, metodologías y paradigmas de programación en el contexto de los ciclos de vida del software y técnicas de gestión de proyectos cuidando conservar los atributos de calidad requeridos.</p> <p>CE21. Diseña el comportamiento, arquitectura e interfaz de soluciones de software, a partir del reconocimiento del contexto, necesidades e involucrados en un sistema empleando técnicas para identificar, obtener, analizar, priorizar, documentar, verificar y validar los requisitos en el contexto de los ciclos de vida y procesos del desarrollo de software.</p>		
Unidades de aprendizaje relacionadas:	Introducción a la Ingeniería de Software, Ingeniería de Software, Metodologías Ágiles de Desarrollo de Software, Administración de Proyectos de Software		
Responsable(s) de elaborar el programa:	Dr. Juan Francisco Figueroa Pérez		Fecha: 31/01/2025
Responsable(s) de actualizar el programa:			Fecha:



PROGRAMA DE ESTUDIO

2. PROPÓSITO	
Conoce y aplica correctamente los conceptos teóricos y prácticos relacionados con diversas metodologías tradicionales de desarrollo de software.	
3. SABERES	
Teóricos:	Conoce e identifica los aspectos y elementos más relevantes del marco conceptual de las metodologías tradicionales de desarrollo de software.
Prácticos:	Utiliza las metodologías lineal o secuencial y el proceso unificado de desarrollo de software en el desarrollo de un proyecto de software, según sean las características y necesidades del mismo.
Actitudinales:	Comunicación oral y escrita en la propia lengua. Habilidades de gestión de la información (habilidad para buscar y analizar información proveniente de diversas fuentes). Capacidad de crítica y autocrítica. Trabajo en equipo. Compromiso ético. Capacidad para aprender a trabajar de forma autónoma.
4. CONTENIDOS	
<p>1.- Introducción a las metodologías de desarrollo de software tradicionales</p> <p>2.- Modelos clásicos de desarrollo de software</p> <p> 2.1 Cascada</p> <p> 2.2 Prototipado</p> <p> 2.3 Espiral</p> <p> 2.4 Incremental</p> <p> 2.5 Diseño rápido de aplicaciones</p> <p>3.- Metodología lineal o secuencial</p> <p> 3.1 Requisitos</p> <p> 3.2 Análisis</p> <p> 3.3 Diseño</p> <p> 3.4 Programación</p> <p> 3.5 Pruebas</p> <p> 3.6 Práctica integradora</p> <p>4.- Proceso unificado de desarrollo de software</p> <p> 4.1 Fases</p> <p> 4.2 Disciplinas</p> <p> 4.3 El lenguaje unificado de modelado</p> <p> 4.4 Práctica integradora</p>	
5. ACTIVIDADES PARA DESARROLLAR LAS COMPETENCIAS	
<p><i>Actividades del docente:</i></p> <ul style="list-style-type: none">● Selecciona un tema y guía la conducción de las preguntas que los jóvenes abordarán en la clase: ¿Qué? ¿Quién? ¿Cómo? ¿Cuándo? ¿Por qué? ¿Para qué? Sobre los diferentes elementos de las metodologías tradicionales de desarrollo de software● Selecciona un tema y guía la conducción de las preguntas que los jóvenes abordarán en la clase: ¿Qué? ¿Quién? ¿Cómo? ¿Cuándo? ¿Por qué? ¿Para qué? Sobre los diferentes elementos de la metodología lineal o secuencial	



PROGRAMA DE ESTUDIO

- Selecciona un tema y guía la conducción de las preguntas que los jóvenes abordarán en la clase: ¿Qué? ¿Quién? ¿Cómo? ¿Cuándo? ¿Por qué? ¿Para qué? Sobre los diferentes elementos de la metodología del proceso unificado de desarrollo de software
- Explica los diferentes elementos de la metodología lineal o secuencial que serán usados en la práctica integradora del tema
- Explica los diferentes elementos de la metodología de proceso unificado de desarrollo de software que serán usados en la práctica integradora del tema

Actividades del estudiante:

- ❖ Dan respuesta a las preguntas haciendo referencias a datos, ideas y detalles expresados en las lecturas.
- ❖ Entrega avances de documentación de técnica relacionada con el uso de las metodologías elaborados durante el semestre
- ❖ Elabora documentación técnica, entregas intermedias y entrega final de prototipo de proyecto.

6. EVALUACIÓN DE LAS COMPETENCIAS

6.1. Criterios de desempeño	6.2 Portafolio de evidencias
Lecturas obligadas y estudio individual, resolución de prácticas, problemas o ejercicios propuestos por profesor, preparación de trabajos individuales, exposiciones de trabajos en clase, portafolio de evidencia y heteroevaluación.	Reportes de lectura, documentación del proyecto con metodología utilizada, avances de documentación con prototipo de proyecto y entrega final de documentación con prototipo de proyecto

6.3. Calificación y acreditación:

Parcial: Evaluaciones parciales y avances de documentación con prototipo de proyecto	Final: Entrega final de documentación con prototipo de proyecto
---	--

7. RECURSOS DIDÁCTICOS

Computadora, pintarrón, cañon, diapositivas, exposiciones y plantillas sobre los planes subsidiarios y plan de gestión del proyecto

8. FUENTES DE INFORMACIÓN

Bibliografía básica

Autor(es)	Título	Editorial	Año	URL o biblioteca digital donde está disponible
Pressman, Roger	Ingeniería de Software: Un Enfoque Practico	Mc Graw Hill	2005	
Somerville, Ian	Ingeniería del Software	Pearson	2005	
By Per Kroll , Philippe Kruchten	Rational Unified Process Made Easy: A Practitioner's Guide to the RUP, The	Addison Wesley	2003	



UNIVERSIDAD AUTÓNOMA DE SINALOA

FACULTAD DE INGENIERÍA MOCHIS LICENCIATURA EN INGENIERÍA EN SOFTWARE



PROGRAMA DE ESTUDIO

Gerardus Blokdyk	Waterfall Development A Complete Guide - 2020 Edition	5STARCook	2020	
Larman, Craig	UML y patrones	Pearson	2003	
<i>Bibliografía complementaria</i>				
Autor(es)	Título	Editorial	Año	URL o biblioteca digital donde está disponible
9. PERFIL DEL DOCENTE				
Licenciado en informática, licenciado en sistemas computacionales, licenciado en ingeniería de software, ingeniero en sistemas computacionales, ingeniero en computación				