

<p>UNIVERSIDAD AUTÓNOMA DE CHIHUAHUA</p>  <p>UNIDAD ACADÉMICA: FACULTAD DE INGENIERÍA PROGRAMA ANALÍTICO DE LA UNIDAD DE APRENDIZAJE: <u>DESARROLLO BASADO EN PLATAFORMAS.</u></p>	DES:	INGENIERÍA
	Programa académico	Ingeniería en Computación
	Tipo de materia (Obli/Opta):	Obligatoria
	Clave de la materia:	CO405
	Semestre:	Cuarto
	Área en plan de estudios:	Específica
	Total de horas por semana:	5
	<i>Teoría: Presencial o Virtual</i>	0
	<i>Laboratorio o Taller:</i>	4
	<i>Prácticas:</i>	0
	<i>Trabajo extra-clase:</i>	1
	Créditos Totales:	5
	Total de horas semestre (x sem):	80
	Fecha de actualización:	Octubre 2024
<i>Prerrequisito (s):</i>	N/A	

DESCRIPCIÓN:

Este curso proporciona a los estudiantes una comprensión integral del entorno laboral en el desarrollo de software, centrándose en las plataformas web, interfaz de línea de comandos y móviles. A través de una combinación de teoría, práctica y estudio de casos, los participantes adquirirán una visión panorámica de los aspectos fundamentales y específicos del proceso de producción del software en estos entornos.

COMPETENCIAS PARA DESARROLLAR:

B4. Transformación Digital

Transforma la cultura digital en la sociedad, en las organizaciones e instituciones educativas para aprovechar al máximo el potencial de las tecnologías y herramientas digitales; propiciar su uso responsable y ético que estimule la creatividad, innovación, la comunicación efectiva y el trabajo colaborativo e interdisciplinar en la solución de problemas de la sociedad digital; promoviendo la privacidad y la seguridad, así como el respeto a los derechos de autor y la propiedad intelectual.

P2. DESARROLLO DE PROYECTOS DE INGENIERÍA. Desarrolla proyectos de ingeniería complejos en sus etapas de planeación, análisis y diseño, utilizando las tecnologías y los principios de la administración para la optimización de los recursos con base en procesos de calidad, mejora continua y teniendo en cuenta la seguridad, el costo del ciclo de vida, el carbono neto cero y la salud según sea necesario, atendiendo las necesidades de sostenibilidad.

E1. DISEÑO Y DESARROLLO DE SOFTWARE. Utilizar en el diseño y desarrollo de software, integrando algoritmos avanzados y estructuras de datos para crear soluciones de software robustas y de calidad. Implica una comprensión profunda de los principios de programación, un enfoque metódico para la solución de problemas y la capacidad de adaptar y mejorar continuamente las prácticas de desarrollo para satisfacer las cambiantes necesidades tecnológicas y las demandas de los diversos sectores.

DOMINIOS	OBJETOS DE ESTUDIO	RESULTADOS DE APRENDIZAJE	METODOLOGÍA	EVIDENCIAS
<p>P2. DESARROLLO DE PROYECTOS DE INGENIERÍA.</p> <p>2. Desarrolla proyectos complejos de ingeniería que integra la planeación, análisis, diseño y administración con base en los criterios de sostenibilidad.</p> <p>E1. DISEÑO Y DESARROLLO DE SOFTWARE.</p> <p>4. Proponer soluciones innovadoras en el diseño y desarrollo de software, en diferentes plataformas y dispositivos, aplicando procesos, métodos y mejores prácticas de ingeniería de software, para desarrollar proyectos medibles, repetibles y de calidad.</p>	<p><u>I. INTRODUCCIÓN AL DESARROLLO DE APLICACIONES</u></p> <p>1.1 Introducción a los sistemas Unix.</p> <p>1.1.1 ¿Qué es un sistema Unix?</p> <p>1.2 El sistema de archivos.</p> <p>1.3 Comandos básicos de la terminal.</p> <p>1.4 Enlaces del sistema.</p> <p>1.5 Usuarios y permisos.</p> <p>1.6 Tareas programadas.</p> <p>1.7 Control de procesos por medio de la terminal.</p> <p>1.2 Programación dentro del sistema operativo.</p> <p>1.2.1 Diseño de programas mediante instrucciones de script.</p> <p>1.2.2 Declaraciones y sentencias.</p> <p>1.2.3 Operadores aritméticos, lógicos y comparativos.</p> <p>1.2.4 Estructuras de control condicionales.</p> <p>1.2.5 Estructuras de control selectivas.</p> <p>1.2.6 Estructuras de control repetitivas.</p> <p>1.2.7 Funciones.</p> <p>1.3 Sistemas de control de versiones.</p> <p>1.3.1 Introducción</p> <p>1.3.2 Configurando el entorno.</p> <p>1.3.3 Versionado de archivos.</p> <p>1.3.4 Ignorando archivos.</p> <p>1.3.5 Manejo de llaves públicas y privadas.</p>	<p>Aprende el uso del sistema operativo Unix y la programación dentro del mismo. Además utiliza sistemas de control de versiones y manejo de contenedores, brindando una comprensión integral de las herramientas y prácticas fundamentales en el desarrollo de software moderno.</p>	<ul style="list-style-type: none"> ● Clase interactiva Maestro - Alumno. ● Recursos tecnológicos institucionales. ● Laboratorio de cómputo. 	<ul style="list-style-type: none"> ● Tareas. ● Proyecto. ● Examen.

<p>B4.3 Aplica de forma ética diferentes herramientas digitales que favorezcan el trabajo colaborativo e interprofesional, considerando las principales innovaciones científicas y tecnológicas, relacionadas con la profesión.</p>	<p>1.3.6 Utilizando repositorios remotos. 1.3.7 Reservas de código. 1.3.8 Resolución de conflictos. 1.3.9 Ramas.</p> <p>1.4 Manejo de contenedores. 1.4.1 Introducción 1.4.2 Aspectos básicos de los contenedores. 1.4.3 Ejecutando un contenedor. 1.4.4 Red e información de los contenedores. 1.4.5 Creando un contenedor. 1.4.6 Trabajando con imágenes. 1.4.7 Archivo de configuración. 1.4.8 Redes. 1.4.9 Conexión de contenedores. 1.4.10 Volúmenes. 1.4.11 Etiquetas. 1.4.12 Publicar imágenes.</p>			
	<p><u>II. CONCEPTOS BÁSICOS DE LAS PLATAFORMAS WEB</u></p> <p>2.1 Protocolos de internet. 2.1.1 Protocolos TCP/IP. 2.1.2 HTTP / HTTPS. 2.1.3 SSL.</p> <p>2.2 Introducción a los lenguajes de programación de la web. 2.2.1 Introducción a los lenguajes de scripting. 2.2.2 Programación funcional aplicada a la web. 2.2.3 Programación orientada a objetos aplicada a la web.</p>	<p>Comprende los elementos fundamentales necesarios para el desarrollo web. Aplica los protocolos de Internet mediante lenguajes de programación de scripting usando programación funcional y orientada a objetos. Además, crea sitios web estáticos mediante HTML, CSS y Javascript.</p>	<ul style="list-style-type: none"> ● Clase interactiva Maestro - Alumno. ● Recursos tecnológicos institucionales. ● Laboratorio de cómputo. 	<ul style="list-style-type: none"> ● Tareas. ● Proyecto. ● Examen.

	<p>2.3 Plataformas para el uso de los protocolos de la web. 2.3.1 Introducción. 2.3.2 Manejo de los diferentes protocolos de internet por medio de plataformas.</p> <p>2.4. HTML. 2.4.1 Introducción a HTML. 2.4.2 Etiquetas. 2.4.3 Elementos. 2.4.4 Atributos 2.4.5 Anatomía de un documento 2.4.6 Encabezados y párrafos. 2.4.7 Listas. 2.4.8 Cursivas, negritas y subrayados. 2.4.9 Hipervínculos. 2.4.10 Tablas. 2.4.11 Formularios. 2.4.12 D.O.M.</p> <p>2.5. CSS 2.5.1 Introducción a CSS 2.5.2 Agregando CSS al HTML 2.5.3 Reglas. 2.5.4 Selectores. 2.5.5 Modelo de caja. 2.5.6 Variables. 2.5.7 Niveles de reglas. 2.5.8 Pseudoclasas. 2.5.9 Texto. 2.5.10 Color.</p>			
	<p><u>III. INTRODUCCIÓN AL DESARROLLO EN PLATAFORMAS MÓVILES</u></p> <p>3.1 Tipos de desarrollo de aplicaciones móviles. 3.1.1 ¿Qué son las aplicaciones móviles? 3.1.2 Historia de las aplicaciones móviles.</p>	<p>Identifica los tipos de desarrollo de aplicaciones móviles y su aplicación como estrategias de desarrollo para generar aplicaciones móviles efectivas y versátiles.</p>	<ul style="list-style-type: none"> ● Clase interactiva Maestro - Alumno. ● Recursos tecnológicos institucionales. ● Laboratorio de cómputo. 	<ul style="list-style-type: none"> ● Tareas. ● Proyecto. ● Examen.

	<p>3.2. Desarrollo nativo de aplicaciones móviles.</p> <p>3.2.1 Configuración de un ambiente nativo.</p> <p>3.2.2 Primera aplicación nativa.</p> <p>3.2.3 Navegación simple.</p> <p>3.2.4 Modelo de distribución.</p> <p>3.3 Desarrollo híbrido de aplicaciones móviles.</p> <p>3.3.1 Configuración de un ambiente híbrido.</p> <p>2.3.2 Primera aplicación híbrida.</p> <p>3.3.3 Navegación simple.</p>			
--	--	--	--	--

FUENTES DE INFORMACIÓN (Bibliografía, direcciones electrónicas)	EVALUACIÓN DE LOS APRENDIZAJES
<p>W3C. (2024). W3Schools Online Web Tutorials. W3 Schools. Recuperado de https://www.w3schools.com/</p> <p>Cuesta, P. (2023). Desarrollo de aplicaciones multiplataforma con Flutter: Guía completa para crear aplicaciones móviles nativas con Dart. Anaya Multimedia.</p> <p>Giroux, M.-A., & Gallaire, F. (2023). React Native: desarrollo de aplicaciones móviles multiplataforma con JavaScript. Ediciones ENI. ISBN: 978-2-409-04604-9</p> <p>Simpson, K. (2015). You Don't Know Js: Es6 & Beyond. O'Reilly Media. ISBN: 9781491904244</p>	<p>Parcial 1 (30%)</p> <ul style="list-style-type: none"> ● Tareas. (20%) ● Proyecto. (50%) ● Examen. (30%) <p>Parcial 2 (30%)</p> <ul style="list-style-type: none"> ● Tareas. (20%) ● Proyecto. (50%) ● Examen. (30%) <p>Parcial 3 (40%)</p> <ul style="list-style-type: none"> ● Tareas. (20%) ● Proyecto. (50%) ● Examen. (30%) <p>La calificación mínima es 7.0.</p> <p>Se usará rúbrica para la entrega de actividades o tareas a realizar.</p>

CRONOGRAMA

Objetos de estudio	Semanas															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
I. INTRODUCCIÓN AL DESARROLLO DE APLICACIONES	■	■	■	■	■	■										
II. CONCEPTOS BÁSICOS DE LAS PLATAFORMAS WEB							■	■	■	■	■	■				
III. INTRODUCCIÓN AL DESARROLLO EN PLATAFORMAS MÓVILES													■	■	■	■