

1. Datos Generales de la asignatura

Nombre de la asignatura:	Tópicos Selectos de Programación WEB
Clave de la asignatura:	CGB-2403
SATCA¹:	1 - 4 - 5
Carrera:	Ingeniería en Sistemas Computacionales

2. Presentación

Caracterización de la asignatura
<p>Esta asignatura aporta al perfil del egresado la capacidad para desarrollar y administrar software que apoye la productividad y competitividad de las organizaciones cumpliendo con estándares de calidad, mediante el desarrollo de aplicaciones web utilizando lenguajes de marcas, de presentación, del lado del cliente, del servidor a través de la implementación de bibliotecas y/o frameworks de desarrollo para mejorar la eficiencia y eficacia en la construcción de dichas aplicaciones.</p> <p>La importancia de esta asignatura radica en la generación de mejores prácticas de diseño y desarrollo de software a través de la implementación de patrones de diseño y una correcta arquitectura de un proyecto, que dichos frameworks proveen.</p> <p>La asignatura consiste en el diseño y desarrollo de aplicaciones web desde el lado del cliente y del servidor, permitiendo la interconectividad entre sistemas y la persistencia de datos.</p> <p>Esta asignatura está relacionada con otras enfocadas en la elaboración y administración de software como lo es gestión de proyectos de software.</p>
Intención didáctica

¹ Sistema de Asignación y Transferencia de Créditos Académicos

La asignatura está organizada en cuatro temas:

El primer tema, se centra en la creación, control y manipulación de objetos utilizados por el cliente de las aplicaciones web, así como la integración de diversos Frameworks disponibles.

En el segundo tema, se aborda la programación del lado del servidor en donde se logre la manipulación de objetos y el acceso a datos para la presentación de procesos dinámicos y sus resultados en una aplicación web, utilizando bibliotecas y/o frameworks para la codificación.

En el tercer tema, se trata lo relacionado a la persistencia de datos y su almacenamiento utilizando diversas tecnológicas.

A través del cuarto tema, se implementa una arquitectura basada en microservicios para mejorar la modularidad, escalabilidad y mantenimiento de aplicaciones web desde la perspectiva del servidor.

El docente deberá realizar prácticas guiadas para la demostración de la implementación a manera de ejemplos, para posteriormente dejar al alumno una serie de ejercicios que le permitan reforzar lo aprendido en clase. Además, se sugiere indicar un proyecto de materia para aplicar los conocimientos en aplicaciones que resuelvan problemas de la vida real.

3. Participantes en el diseño y seguimiento curricular del programa

Lugar y fecha de elaboración o revisión	Participantes	Observaciones
Jerez de García Salinas, Zacatecas. abril del 2024	Tecnológico Nacional de México Campus Jerez MTI Salvador Acevedo Sandoval	Reunión para el diseño y desarrollo de la especialidad para el periodo 2024 - 2026

4. Competencia(s) a desarrollar

Competencia(s) específica(s) de la asignatura
Desarrolla aplicaciones web dinámicas del lado cliente y del servidor utilizando diferentes tecnologías y mediante la implementación de bibliotecas y frameworks, enfocadas a una arquitectura de microservicios.

5. Competencias previas

<ul style="list-style-type: none"> • Desarrolla aplicaciones web dinámicas del lado cliente y del servidor, considerando la conectividad a orígenes de datos, la interconectividad entre aplicaciones y cómputo en la nube. • Aplica métodos y herramientas de la ingeniería del software en el desarrollo de software aplicando estándares de calidad y productividad. • Aplica un lenguaje orientado a objetos para la solución de problemas. • Crea y aplica esquemas de bases de datos para garantizar la confiabilidad de los datos en aplicaciones para el tratamiento de información.

6. Temario

No.	Temas	Subtemas
1	Desarrollo del lado del Cliente	1.1 Introducción a las bibliotecas y frameworks 1.2 Vue 1.3 React 1.4 Angular
2	Desarrollo del lado del Servidor	2.1 Introducción a las bibliotecas y frameworks 2.2 JavaScript (NodeJS: Express JS y NestJS) 2.3 PHP - Laravel 2.4 JAVA - SpringBoot 2.5 PYTHON - Django/Flask
3	Persistencia	3.1 Almacenamiento local 3.1.1 Cookies 3.1.2 Sesiones 3.1.3 LocalStorage 3.1.4 SessionStorage 3.1.5 IndexedDB 3.1.6 Service Worker 3.2 Almacenamiento remoto 3.2.1 Gestores relacionales 3.2.2 Gestores No relacionales (MongoDB, Redis, Cassandra) 3.2.3 DBaaS 3.2.3.1 Firebase 3.2.3.2 AWS/Azure/GCP
4	Arquitectura de microservicios	4.1 Arquitectura Monolítica vs Microservicios 4.2 API de comunicación 4.3 Persistencia

		4.4 Seguridad 4.5 Despliegue
--	--	---------------------------------

7. Actividades de aprendizaje de los temas

1. Desarrollo del lado del Cliente	
Competencias	Actividades de aprendizaje
<p>Específica(s): Comprende y aplica bibliotecas y/o frameworks en el lado del cliente de una aplicación web.</p> <p>Genéricas:</p> <ul style="list-style-type: none"> • Capacidad de análisis y síntesis. • Comunicación oral y escrita. • Habilidad para buscar, analizar, clasificar y sintetizar información proveniente de fuentes diversas. • Solución de problemas. • Capacidad de trabajar en equipo. • Capacidad de aplicar los conocimientos en la práctica. • Capacidad de aprender. 	<ol style="list-style-type: none"> 1. Investigar los conceptos importantes acerca de las bibliotecas y/o frameworks del lado del cliente. Elaborar un mapa conceptual. 2. Desarrollar ejercicios con interfaces graficas de usuario utilizando bibliotecas y/o frameworks de JavaScript. 3. Realizar un cuadro comparativo entre el desarrollo con y sin bibliotecas y/o frameworks de JavaScript. Elaborar cuadro comparativo.
2. Desarrollo del lado del Servidor	
Competencias	Actividades de aprendizaje
<p>Específica(s): Comprende y aplica bibliotecas y/o frameworks en el lado del servidor de una aplicación web.</p> <p>Genéricas:</p> <ul style="list-style-type: none"> • Capacidad de análisis y síntesis. • Comunicación oral y escrita. • Habilidad para buscar, analizar, clasificar y sintetizar información proveniente de fuentes diversas. • Solución de problemas. • Capacidad de trabajar en equipo. • Capacidad de aplicar los conocimientos en la práctica. • Capacidad de aprender. 	<ol style="list-style-type: none"> 1. Investigar los conceptos importantes acerca de las frameworks del lado del servidor. Elaborar un mapa conceptual. 2. Desarrollar ejercicios de comunicación entre el cliente y el servidor a través de frameworks del lado del servidor 3. Realizar un cuadro comparativo entre los distintos frameworks del lado del servidor. Elaborar cuadro comparativo.
3. Persistencia	
Competencias	Actividades de aprendizaje
<p>Específica(s): Implementa diferentes opciones para la persistencia de datos tanto del lado del cliente como del lado del servidor, incluyendo servicios web de almacenamiento.</p> <p>Genéricas:</p>	<ol style="list-style-type: none"> 1. Investigar los conceptos importantes acerca de las formas de persistencia tanto del lado del cliente, como del lado del servidor. Elaborar un mapa conceptual. 2. Desarrollar ejercicios que permitan obtener

<ul style="list-style-type: none"> • Capacidad de análisis y síntesis. • Comunicación oral y escrita. • Habilidad para buscar, analizar, clasificar y sintetizar información proveniente de fuentes diversas. • Solución de problemas. • Capacidad de trabajar en equipo. • Capacidad de aplicar los conocimientos en la práctica. <p>Capacidad de aprender.</p>	<p>información a través de diversas tecnologías que proveen persistencia para aplicaciones web.</p>
4. Arquitectura de microservicios	
Competencias	Actividades de aprendizaje
<p>Específica(s): Comprende y aplica la arquitectura de microservicios en una aplicación web.</p> <p>Genéricas:</p> <ul style="list-style-type: none"> • Capacidad de análisis y síntesis. • Comunicación oral y escrita. • Habilidad para buscar, analizar, clasificar y sintetizar información proveniente de fuentes diversas. • Solución de problemas. • Capacidad de trabajar en equipo. • Capacidad de aplicar los conocimientos en la práctica. <p>Capacidad de aprender.</p>	<ol style="list-style-type: none"> 1. Investigar los conceptos importantes acerca de la arquitectura basada en microservicios. Elaborar un mapa conceptual. 2. Desarrollar ejercicios que permitan la implementación de una aplicación web con arquitectura de microservicios. 3. Realizar un cuadro comparativo entre la arquitectura monolítica y la de microservicios. Elaborar cuadro comparativo.

8. Práctica(s)

<ol style="list-style-type: none"> 1. Diseño y desarrollo de interfaces gráficas como Landing Page, Login, Dashboard y pantallas para operaciones CRUD (Create, Read, Update & Delete) , utilizando diferentes frameworks. 2. Agregar interacción dinámica a las páginas anteriores a través del uso de frameworks de JavaScript. 3. Agregar contenido dinámico a las páginas anteriores a través del uso de frameworks del lado del servidor. 4. Crear BD de prueba para conexión a través de API's del lado del servidor, permitiendo el intercambio de información en BD SQL y No SQL.

9. Proyecto de asignatura

<p>El objetivo del proyecto que planteé el docente que imparta esta asignatura, es demostrar el desarrollo y alcance de la(s) competencia(s) de la asignatura, considerando las siguientes fases:</p> <ul style="list-style-type: none"> • Fundamentación: marco referencial (teórico, conceptual, contextual, legal) en el cual se fundamenta el proyecto de acuerdo con un diagnóstico realizado, mismo que permite a los estudiantes lograr la comprensión de la realidad o situación objeto de estudio para definir un proceso de intervención o hacer el diseño de un modelo. • Planeación: con base en el diagnóstico en esta fase se realiza el diseño del proyecto por parte de los estudiantes con asesoría del docente; implica planificar un proceso: de intervención empresarial, social o
--

comunitario, el diseño de un modelo, entre otros, según el tipo de proyecto, las actividades a realizar los recursos requeridos y el cronograma de trabajo.

- **Ejecución:** consiste en el desarrollo de la planeación del proyecto realizada por parte de los estudiantes con asesoría del docente, es decir en la intervención (social, empresarial), o construcción del modelo propuesto según el tipo de proyecto, es la fase de mayor duración que implica el desempeño de las competencias genéricas y específicas a desarrollar.
- **Evaluación:** es la fase final que aplica un juicio de valor en el contexto laboral-profesión, social e investigativo, ésta se debe realizar a través del reconocimiento de logros y aspectos a mejorar se estará promoviendo el concepto de “evaluación para la mejora continua”, la metacognición, el desarrollo del pensamiento crítico y reflexivo en los estudiantes.

10. Evaluación por competencias

Para evaluar las actividades de aprendizaje se recomienda solicitar: mapas conceptuales o mentales, reporte de investigación, cuadros comparativos, reportes de prácticas, códigos de programas, estudio de casos, exposiciones en clase, portafolio de evidencias, entre otros.

Para verificar el nivel del logro de las competencias del estudiante se recomienda utilizar: listas de cotejo, listas de verificación, matrices de valoración, guías de observación, rúbricas, entre otros.

11. Fuentes de información

1. Página oficial World Wide Web Consortium. <https://www.w3.org/>
2. Página oficial EcmaScript. <https://www.ecma-international.org/>
3. Ribeiro, H. (2020). Vue.js 3 Cookbook: Discover actionable solutions for building modern web apps with the latest Vue features and TypeScript
4. Santana, C. (2021). React 17 Design Patterns and Best Practices: Design, build, and deploy production-ready web applications using industry-standard practices, 3rd Edition (English Edition).
5. Uluca, D. (2020). Angular for Enterprise-Ready Web Applications: Build and deliver production-grade and cloud-scale evergreen web apps with Angular 9 and beyond, 2nd Edition. Editorial PACKT
6. Morales, G. (2020). Creando API con Node.js, express y MongoDB. LeanPub
7. Lopez, J. (2020). Laravel. Aprende A Crear Aplicaciones Web Desde Cero. Editorial RC Libros.
8. Hinkula, J. (2019). Hands-On Full Stack Development with Spring Boot 2 and React: Build modern and scalable full stack applications using Spring Framework 5 and React with Hooks, 2nd Edition. Editorial Packt.
9. Ben Shaw, Saurabh Badhwar, Andrew Bird, Bharath Chandra K S, Chris Guest. (2021). Web Development with Django: Learn to build modern web applications with a Python-based framework. Editorial Packt Publishing.
10. Vazquez, J., Martinez, D. (2020). Ruby On Rails. Aprende A Crear Aplicaciones Web Desde Cero. Editorial RC Libros.
11. Richardson, C. (2019). Microservices Patterns: With Examples in Java. Editorial Manning
12. Blokdyk, G. (2020). Database As A Service A Complete Guide.