

GRADO EN INGENIERÍA INFORMÁTICA

Nombre de la Asignatura:	PROGRAMACIÓN DE SERVIDORES WEB	Código:	G174
Curso:	4	Semestre:	1
Tipo:	Obligatoria	Créditos:	6
Departamento:	Métodos Cuantitativos	Nombre y correo del coordinador:	Andrés Gorostidi andres.gorostidi@cunef.edu

1. DESCRIPCIÓN Y OBJETIVOS DE LA ASIGNATURA

La asignatura "Programación de Servidores Web" se centra en el diseño, desarrollo y mantenimiento de aplicaciones web del lado del servidor. Los estudiantes aprenderán a utilizar tecnologías y frameworks actuales para crear aplicaciones robustas, escalables y seguras. El curso cubre conceptos fundamentales de la programación del lado del servidor, la integración con bases de datos, la seguridad en aplicaciones web y el despliegue en servidores.

2. RESULTADOS DEL APRENDIZAJE

Al finalizar esta asignatura, los estudiantes serán capaces de:

- Comprender los conceptos fundamentales de la programación del lado del servidor.
- Diseñar y desarrollar aplicaciones web utilizando tecnologías modernas.
- Implementar medidas de seguridad para proteger aplicaciones web.
- Integrar bases de datos con aplicaciones web.
- Desplegar aplicaciones web en servidores y servicios en la nube.
- Utilizar JavaScript, AJAX/JQuery y librerías de CSS y JavaScript de abstracción para mejorar la interactividad y usabilidad de las aplicaciones web.

3. PLANIFICACIÓN DOCENTE

3.1. PROGRAMA DE LA ASIGNATURA

Tema 1. Introducción a la Programación de Servidores Web

- Historia y evolución de la web
- Cliente vs. servidor
- HTTP y HTTPS
- Páginas estáticas vs dinámicas
- CMS y Digital Experience
- Redes Sociales y Marca Corporativa / Personal (Linked-in, GitHub Pages, Blogs)
- Diferencia entre front-end y back-end. Roles del equipo de desarrollo.
- Introducción a lenguajes del lado del servidor (Java-Spring Boot, PHP, Node.js, Python)

Tema 2. Configuración del Entorno de Desarrollo

- Instalación de servidores web (Apache, Nginx, XAMPP)
- Herramientas de Diseño/MockUps (Figma, Zeplin)
- Herramientas de desarrollo (IDEs, Visual Studio & Extensiones, Git...)
- Uso de GitHub

Tema 3. Programación del Lado del Cliente

- Fundamentos de HTML y CSS
- Análisis del código mediante la consola de developer
- Principios de diseño responsivo

Tema 4. Tecnologías Web Frontend

- JavaScript básico y avanzado
- Manipulación del DOM
- La evolución de ECMAScript
- Introducción a TypeScript

Tema 5. AJAX y jQuery

- Introducción a AJAX
- Uso de jQuery para simplificar AJAX
- Utilizando Promesas con Fetch API
- AXIOS

Tema 6. Librerías de CSS y JavaScript de Abstracción

- Bootstrap y su uso en el diseño responsivo
- Librerías de JavaScript (React, Vue.js, Angular) y su aplicación en proyectos

Tema 7. Fundamentos de PHP

- Sintaxis básica
- Manejo de formularios
- Gestión de sesiones y cookies

Tema 8. Fundamentos de Node.js

- Introducción a Node.js
- El gestor de versión de node npm
- Módulos y npm
- Creación de servidores y manejo de rutas
- Fundamentos CRUD (crud en estructuras de memoria).
- OpenAPI y Swagger

Tema 9. Integración con Bases de Datos

- SQL y NoSQL
- Conexión a bases de datos relacionales y no relacionales (MySQL, MongoDB)
- CRUD usando Bases de Datos
- Principales mapeadores ORM (Object Relational Mapping)

Tema 10. Seguridad en Aplicaciones Web

- Autenticación y autorización – JWT
- CORs
- Protección contra ataques comunes (XSS, CSRF, SQL Injection)
- HTTPS y cifrado

Tema 11. Despliegue y Escalabilidad

- Despliegue en servidores (Apache Jakarta, VPS, cloud services)
- Contenedores (Docker)
- Escalabilidad horizontal y vertical

Tema 12. Proyecto Final

- Desarrollo de una aplicación web completa
- Presentación y evaluación del proyecto

3.2. EVALUACIÓN

La **asistencia a clase es obligatoria** en primera matrícula. Una asistencia inferior al 80% implica la pérdida del derecho a examen en la convocatoria ordinaria. El estudiante, además, tiene la obligación de asistir a las tutorías a las que sea convocado por su profesor, independientemente de las que solicite de forma voluntaria para resolver cualquier duda. En el caso de los **estudiantes repetidores**, no es obligatoria su asistencia a clase (aunque sí aconsejable, como también la asistencia a tutorías), pero **sí realizar la evaluación continua** (es decir, el sistema de evaluación es el mismo que para los estudiantes que cursan la asignatura por primera vez).

Durante el curso se llevará a cabo un proceso de **evaluación continua**, que implica la realización de trabajo autónomo guiado por el profesor y que puede conllevar la asistencia a conferencias y la realización de presentaciones u otras pruebas en horario de tutorías. Al final del periodo lectivo, y en las fechas establecidas en el calendario académico, se realizará además un **examen final**. En la **calificación final** de la asignatura la evaluación continua tiene un peso del 40% y el examen final del 60%.

El **examen final** es de cátedra, comprensivo (está diseñado para que el estudiante sea evaluado de todos los contenidos incluidos en el programa) y escrito. Las pruebas que se incluyan en la evaluación continua en ningún caso eximen al alumno de ser evaluado en el examen final de la totalidad de los contenidos de la asignatura (no hay exámenes “liberatorios”).

Para aprobar la asignatura es necesario tener una nota **superior o igual a 5 en el examen final**. Si se suspende el examen final, la nota final en el acta de la convocatoria ordinaria será la nota del examen final sobre 10. Si el estudiante no se presenta al examen final, o no puede presentarse por no llegar al porcentaje de asistencia requerido, su calificación en la nota del examen final es cero. Este criterio se aplica tanto a la convocatoria ordinaria como extraordinaria.

En la **convocatoria extraordinaria**, la **nota de la evaluación continua** será la nota **más favorable** para el estudiante de las dos siguientes: (1) la nota de la evaluación continua de la convocatoria ordinaria; o (2) la nota que resulta de calcular la media ponderada de la evaluación continua (40%) y del examen final (60%) obtenidas por el estudiante en la convocatoria ordinaria. En todos los casos la nota se calcula sobre 10.

ACTIVIDADES DE EVALUACIÓN CONTINUA

ACTIVIDAD DE EVALUACIÓN 1		ACTIVIDAD DE EVALUACIÓN 2	
Ponderación:	15%	Ponderación:	30 %
Descripción:	Examen teórico sobre los temas 1 y 2. Este examen incluirá preguntas sobre la historia y evolución de la web, diferencias entre cliente y servidor, protocolos HTTP y HTTPS, así como la configuración del entorno de desarrollo, instalación de servidores web, uso de GitHub y herramientas de desarrollo.	Descripción:	Tarea de desarrollo frontend y uso de tecnologías web frontend. Los estudiantes deberán desarrollar una "marca personal" vinculando una presencia en Linked-in, con una página personal realizada en GitHub Pages, y con acceso a un portfolio de proyectos. Deberán mostrar sus conocimientos de HTML y CSS, aplicando principios de diseño responsivo en el diseño de la pagina, y el portfolio referenciado (o la propia página) deberá contener ejemplos de manipulaciones del DOM utilizando JavaScript básico y avanzado, así como implementar AJAX y jQuery. También deberán emplear Bootstrap y alguna librería de JavaScript como React, Vue.js o Angular
ACTIVIDAD DE EVALUACIÓN 3		ACTIVIDAD DE EVALUACIÓN 4	
Ponderación:	15%	Ponderación:	40%
Descripción:	Examen práctico sobre los temas 7 y 8, en el cual los estudiantes deberán crear una pequeña aplicación web utilizando PHP y Node.js. Se evaluará la sintaxis básica, manejo de formularios, gestión de sesiones y cookies en PHP, así como la creación de servidores y manejo de rutas en Node.js.	Descripción:	Proyecto Final. Desarrollo de una aplicación web completa que integre todos los temas tratados durante el curso. Los estudiantes deberán desplegar la aplicación en servidores (Apache Jakarta, VPS, cloud services), utilizando contenedores como Docker, aplicando conceptos de escalabilidad horizontal y vertical. La presentación y defensa del proyecto demostrarán la funcionalidad y seguridad de la aplicación.

EXAMEN FINAL	
Calificación Mínima:	5 puntos de 10 puntos en total.
Ponderación:	60%
Descripción:	El examen final consistirá en preguntas teóricas y prácticas que abarcan todos los contenidos de la asignatura. El examen será escrito y tendrá una duración de 3 horas.

4. RECURSOS

4.1. BIBLIOGRAFÍA

- K. Tatroe, P. MacIntyre y R. Lerdorf: Programming PHP. O'Reilly Media.
- Mead: Learning Node.js Development. Packt Publishing.

- E. Brown: Web Development with Node and Express. O'Reilly Media.
- D. Crockford: JavaScript: The Good Parts. O'Reilly Media.
- M. Haverbeke: Eloquent JavaScript. No Starch Press.
- J. Duckett: HTML and CSS: Design and Build Websites. Wiley.
- Bauer y G. King: Java Persistence with Hibernate. Manning Publications.
- Walls: Spring in Action. Manning Publications.
- M. Young: Java Web Services: Up and Running. O'Reilly Media.
- R. Johnson, J. Hoeller, A. Arendsen, T. Risberg, D. Kopylenko: Professional Java Development with the Spring Framework. Wrox.

4.2. SOFTWARE

DENOMINACIÓN:	XAMPP
DESCRIPCIÓN DE SU USO:	Entorno de desarrollo PHP y MySQL
DISPONIBILIDAD:	https://www.apachefriends.org/es/index.html
DENOMINACIÓN:	Visual Studio Code
DESCRIPCIÓN DE SU USO:	Editor de código
DISPONIBILIDAD:	https://code.visualstudio.com/
DENOMINACIÓN:	Spring Tools 4
DESCRIPCIÓN DE SU USO:	Editor de código Java (Spring Boot)
DISPONIBILIDAD:	https://spring.io/tools
DENOMINACIÓN:	Git
DESCRIPCIÓN DE SU USO:	Sistema de control de versiones
DISPONIBILIDAD:	https://git-scm.com/