

**CENTRO UNIVERSITARIO DE TECNOLOGÍA Y ARTE DIGITAL**



# **PLANIFICACIÓN DE LA DOCENCIA UNIVERSITARIA**

## **GUÍA DOCENTE**

### **Programación web II Servidor**

# 1. DATOS DE IDENTIFICACIÓN DE LA ASIGNATURA.

Título:	Grado en Ingeniería del Software
Facultad:	Centro Universitario de Tecnología y Arte Digital (U-TAD)
Materia:	Programación web
Denominación de la asignatura:	Programación web II Servidor
Curso:	3º
Cuatrimestre:	2
Carácter:	Obligatoria
Créditos ECTS:	6
Modalidad/es de enseñanza:	Híbrido Presencial
Idioma:	Castellano
Profesor/a:	Manoel Fernando Alonso Gadi
E-mail:	Manoel.alonso@u-tad.com
Teléfono:	

## 2. DESCRIPCIÓN DE LA ASIGNATURA.

### 2.1 Descripción de la materia

Esta materia dota de los conocimientos y competencias necesarios por parte de un ingeniero del software para el desarrollo de un proyecto web en sus componentes de servidor y cliente y su potencial exportación como aplicaciones nativas o híbridas.

### 2.2 Descripción de la asignatura

Los navegadores web se comunican con los servidores web mediante el Protocolo de transferencia de hipertexto (HTTP). Cuando hace clic en un enlace en una página web, envía un formulario o ejecuta una búsqueda, se envía una solicitud HTTP desde su navegador al servidor de destino. El código del lado del servidor se puede escribir en cualquier número de lenguajes de programación; ejemplos de lenguajes web populares del lado del servidor incluyen PHP, Python, Ruby, C # y NodeJS (JavaScript). El código del lado del servidor tiene acceso completo al sistema operativo y la base de datos del servidor. La programación del lado del servidor es muy útil porque nos permite entregar de manera eficiente información personalizada para usuarios individuales y, por lo tanto, crear una experiencia de usuario mucho mejor.

El objetivo de este curso es entrar al fondo en el mundo de Programación Web Servidor, entender y practicar temas tan diversos como el almacenamiento y entrega de información de manera eficiente, experiencia de usuario personalizada, acceso controlado al contenido, control información de sesión y estado, notificaciones y comunicación.

### 3. COMPETENCIAS

#### Competencias Básicas y Generales

CB1: Que los estudiantes hayan demostrado poseer y comprender conocimientos en un área de estudio que parte de la base de la educación secundaria general, y se suele encontrar a un nivel que, si bien se apoya en libros de texto avanzados, incluye también algunos aspectos que implican conocimientos procedentes de la vanguardia de su campo de estudio.

CB2: Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio.

CB4: Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado.

CB5: Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía

CG1 - Capacidad para entender, planificar y resolver problemas a través del desarrollo de soluciones informáticas.

CG2 - Desarrollo de soluciones informáticas respetuosas con el medio ambiente, los deberes sociales y los recursos naturales, además de cumplir con la legislación y la ética

CG9 - Capacidad para aprender, modificar y producir nuevas tecnologías informáticas

CG10 - Uso de técnicas creativas para la realización de proyectos informáticos

#### Competencias Específicas

CE5 - Capacidad para diseñar e implementar aplicaciones web tanto en lado del cliente como del servidor con tecnologías estándar escalables

CE6 - Conocimiento del uso de mecanismos de comunicación asíncrona entre cliente-servidor y empaquetado de estas aplicaciones web para plataformas móviles para el desarrollo de aplicaciones web dinámicas

CE10 - Capacidad para manejar un gestor de versiones de código y generar la documentación de una aplicación de forma automática.

## 4. CONTENIDOS

### 4.1. Temario de la asignatura

**Tema 1.- Introducción a Spark y Python**

Introducción y instalación de Spark. Introducción a las estructuras de datos en Python: listas, diccionarios y data frames. Métodos de manipulación de estructuras: lista comprendida, funciones anónimas/lambda y vía map/reduce/filter, apply y np.vectorize.

**Tema 1.- Introducción a Programación Servidor**

Experiencia de usuario personalizada, acceso controlado al contenido, control información de sesión y estado, notificaciones y comunicación

**Tema 2.- Estructura de Proyectos Web**

Estructura de pequeños y grandes Proyectos

**Tema 3.- Frameworks Web**

Explorar los principales Frameworks

**Tema 4.- Base de Datos**

Object Relationship Mapping

**Tema 5.- Email**

Integración de comunicación vía email

**Tema 5.- Construcción de un Proyecto**

Practicar sobre un gran proyecto a largo de todo el curso

### 4.2. Desarrollo temporal

UNIDADES DIDÁCTICAS / TEMAS	PERÍODO TEMPORAL
Introducción a la asignatura y a Flask	Semana 1
Estructura de un Proyecto Web	Semanas 2 a 3
Frameworks Web (Presentación Proyecto 1)	Semanas 4 a 5
Base de Datos	Semanas 6 a 7
Integración Email	Semanas 8 a 9
Construcción de un Proyecto	Semanas 10 a 13
Tutoría (Presentación Proyecto 2 Final)	Semanas 14 a 15

# 5. ACTIVIDADES FORMATIVAS Y MODALIDADES DE ENSEÑANZAS

## 5.1 Modalidades de enseñanza

La asignatura se desarrollará a través de los siguientes métodos y técnicas generales, que se aplicarán diferencialmente según las características propias de la asignatura:

- **Método expositivo/Lección magistral:** el profesor desarrollará, mediante clases magistrales y dinámicas los contenidos recogidos en el temario.
- **Estudio de casos:** análisis de casos reales relacionados con la asignatura.
- **Resolución de ejercicios y problemas:** los estudiantes desarrollarán las soluciones adecuadas aplicando procedimientos de transformación de la información disponible y la interpretación de los resultados.
- **Aprendizaje basado en problemas:** utilización de problemas como punto de partida para la adquisición de conocimientos nuevos.
- **Aprendizaje orientado a proyectos:** se pide a los alumnos que, en pequeños grupos, planifiquen, creen y evalúen un proyecto que responda a las necesidades planteadas en una determinada situación.
- **Aprendizaje cooperativo:** Los estudiantes trabajan en grupo para realizar las tareas de manera colectiva.

## 5.2 Actividades formativas

Actividad Formativa	Horas	Presencialidad
AF1 Clases teóricas / Expositivas	45	100%
AF2 Clases Prácticas	36	100%
AF3 Tutorías	6	50%
AF4 Estudio independiente y trabajo autónomo del alumno	57,5	0%
AF5 Elaboración de trabajos (en grupo o individuales)	29	0%
AF6: Actividades de Evaluación	9	100%

## 6. SISTEMA DE EVALUACIÓN

ACTIVIDAD DE EVALUACIÓN	VALORACIÓN MÍNIMA RESPECTO A LA CALIFICACIÓN FINAL (%)	VALORACIÓN MÁXIMA RESPECTO A LA CALIFICACIÓN FINAL (%)
SE1 Evaluación de la participación en clase, en prácticas o en proyectos de la asignatura	10%	30%
SE2 Evaluación de trabajos, proyectos, informes, memorias	30%	60%
SE3 Prueba Objetiva	30%	60%

### 6.1. Criterios de calificación

ACTIVIDAD DE EVALUACIÓN	VALORACIÓN MÁXIMA RESPECTO A LA CALIFICACIÓN FINAL (%)
Nota de Participación (asistencia en clase y participación activa en clase y en el proyecto)	10%
Evaluación de trabajos, proyectos, informes, memorias	40%
Exámenes parciales (promedio simple)	50%

- Para aprobar la asignatura en la convocatoria ordinaria, es imprescindible que la nota de las actividades/proyecto sea superior a 5.0 (promedio simple presentación 1 y 2) y del examen ordinario sea como mínimo de 5.0 (sobre 10).
- En caso de no conseguir el aprobado en la convocatoria ordinaria, el alumno podrá presentarse a la convocatoria extraordinaria de julio. En dicho caso, este examen engloba todo el contenido de la asignatura y la nota obtenida en convocatoria extraordinaria representará el 100% de la calificación del alumno en la asignatura.
- No está permitido el uso de teléfonos móviles en el aula durante el período de evaluación continua, excepto indicación expresa en sentido contrario del profesor. Los ordenadores portátiles podrán utilizarse únicamente para actividades relacionadas con la asignatura. El profesor podrá retirar el derecho al uso del ordenador a aquellos alumnos que lo utilicen para actividades que no estén relacionadas con la asignatura (consulta de correos, noticias o redes sociales, consulta o elaboración de actividades de otras asignaturas, etc.).
- No está permitido consumir bebidas ni comidas en el aula. Tampoco está permitida la presencia de cualquier tipo de bebida en las mesas, incluso en envases cerrados.

- Se demandará del alumno una participación activa, necesaria para el desarrollo de las clases.
- Se exigirá al alumno un buen comportamiento en todo momento durante el desarrollo de las clases. El mal comportamiento que impida el normal desarrollo de la clase puede conllevar la expulsión del aula por un tiempo a determinar por el profesor.

## 7. BIBLIOGRAFÍA / WEBGRAFÍA

### Bibliografía Básica:

<https://flask.palletsprojects.com/en/1.1.x/>

[https://coddyschool.com/upload/Flask\\_Web\\_Development\\_Developing.pdf](https://coddyschool.com/upload/Flask_Web_Development_Developing.pdf)

### Bibliografía Recomendada:

<https://www.djangoproject.com/>

<http://www.web2py.com/>

<https://bottlepy.org/docs/dev/>

[https://programacion-php.readthedocs.io/es/latest/Tutorial1\\_Conceptos.md.html](https://programacion-php.readthedocs.io/es/latest/Tutorial1_Conceptos.md.html)

<https://rubyonrails.org/>

<https://revel.github.io/>

<https://dotnet.microsoft.com/apps/aspnet>

## 8. MATERIAL, SOFTWARE Y HERRAMIENTAS NECESARIAS

### TIPOLOGÍA DEL AULA:

Sala virtual del curso en Blackboard

### MATERIALES DEL ALUMNO:

Ordenador personal, webcam y micrófono

### SOFTWARE:

- Notepad++
- Anaconda Python