



## IFCD54. Desarrollo web back end

**Sku:** PSIFCD54

**Horas:** 300

**Formato:** HTML

### OBJETIVOS

Confeccionar servicios backend para aplicaciones web con lenguaje php. Modelar, diseñar y crear una base de datos a partir de las especificaciones del usuario y teniendo en cuenta las normas de diseño establecidas.

Desarrollar aplicaciones web utilizando el framework CodeIgniter

Desarrollar una aplicación completa integrando el frontend con el backend e incorporando librerías externas y otras herramientas avanzadas.

### CONTENIDOS

#### CONOCIMIENTOS / CAPACIDADES COGNITIVAS Y PRÁCTICAS

- Introducción al lenguaje php - Variables de texto, numéricas y booleanas - Operadores de concatenación, aritméticos, asignación, relacionales y lógicos - Estructuras de decisión y uso correcto de cada una de ellas: IF, IF ELSE y SWITCH - Uso de operadores lógicos con estructuras de decisión - Estructuras de iteración: FOR, WHILE y DO WHILE - Definición y uso de funciones con o sin parámetros de entrada y salida - Arrays: definición, tipos y métodos para trabajar con ellos - Arrays asociativos y multidimensionales - Funciones para trabajar con cadenas de texto, números y fechas - Control eficiente de errores con la estructura try...catch
- Integración de php en una aplicación web - Recoger información de una aplicación web: o Método POST o Método GET - Uso de variables de sesión - Cookies: o Uso de cookies en la página web o Políticas de aceptación de cookies o Seguridad en el uso de cookies - Uso de ficheros y carpetas: o Leer y escribir en ficheros o Crear y borrar carpetas - Incorporación de multidioma en una página web - Envío de correos electrónicos: o Utilizando la función mail() o Utilizando librerías externas - Anexar imágenes y/o archivos a un correo electrónico
- Recepción y envío de datos entre el servidor y la aplicación web - Recepción de datos desde la aplicación web utilizando AJAX - Envío de datos a la aplicación web utilizando AJAX y el protocolo JSON - Peticiones a servicios externos al dominio utilizando el protocolo CORS
- Programación orientada a objetos con php - El paradigma de la orientación a objetos - Diseño de una aplicación con orientación a objetos: el diagrama

UML - Creación de clases, atributos y métodos - Calificadores de acceso: públicos, protegidos y privados - Instanciación de objetos - Herencia y polimorfismo - Interfaces y traits - Competencias transversales: - Planificación y organización del trabajo - Búsqueda y uso de recursos externos - Resolución de problemas de forma autónoma

## **Tema 2. Diseño y construcción de bases de datos relacionales**

### **CONOCIMIENTOS / CAPACIDADES COGNITIVAS Y PRÁCTICAS**

- Diseño de bases de datos - Introducción a las bases de datos: definición, utilización y tipo - Sistemas gestores de bases de datos relacionales: o Esquema y entidades o Atributos y claves - Claves primarias, únicas, múltiples y foráneas - Relaciones entre tablas: o Uno a muchos o Uno a uno o Muchos a muchos - Modelado de base de datos utilizando draw.io - Creación del diagrama entidad/relación - Integridad referencial y el diagrama entidad/relación - Teoría relacional: las cuatro formas normales del diseño de bases de datos • Construcción de bases de datos: - Instrucciones DDL para construir una base de datos - Construcción de una base de datos directamente con phpMyAdmin - Construcción de una base de datos utilizando la aplicación MySql WorkBench • Instrucciones SQL para acceder a una base de datos - Operaciones CRUD con instrucciones DML: o Insertar datos o Consultar datos o Borrar datos o Modificar datos - Concepto de transacción y punto de sincronía - Accesos avanzados para leer la base de datos utilizando expresiones, funciones y predicados - Acceso a más de una entidad simultáneamente utilizando JOINS - Uso de disparadores o triggers • Integración de la base de datos con el servicio php - Librerías php para acceder a la base de datos: o La librería mysqli estructurada o La librería mysqli orientada a objetos o La librería PDO - Construcción de una plataforma web utilizando el patrón MVC • Competencias transversales: - Realización de diseños de aplicaciones - Abstracción para realizar diseños a partir de especificaciones - Planificación y organización del trabajo - Búsqueda y uso de recursos externos - Resolución de problemas de forma autónoma

## **Tema 3. Frameworks CodeIgniter**

### **CONOCIMIENTOS / CAPACIDADES COGNITIVAS Y PRÁCTICAS**

- Desarrollo de servicios web utilizando el framework CodeIgniter - Instalación de CodeIgniter - Estructura de carpetas y configuración de parámetros - Creación de vistas, controladores y modelos - Desarrollo de una aplicación con llamadas síncronas al servidor - Desarrollo de una aplicación con llamadas asíncronas utilizando AJAX • Competencias transversales: - Búsqueda y uso de recursos externos - Resolución de problemas de forma autónoma

## **Tema 4 .Desarrollo de aplicaciones completas**

### **CONOCIMIENTOS / CAPACIDADES COGNITIVAS Y PRÁCTICAS**

- Desarrollo del frontend: - Frontend para el usuario final - Frontend para mantenimiento de la aplicación • Desarrollo del backend utilizando el patrón MVC - Diseño y construcción de la base de datos - Diseño y construcción de los servicios del controlador - Diseño y construcción de los servicios del modelo • Incorporación de pasarelas de pago Paypal • Incorporación de la librería DOMPDF para generar documentos descargables • Generación de los casos de prueba de la plataforma • Generación de la documentación técnica y de usuario • Despliegue de la plataforma en el servidor CDMON • Competencias transversales: - Búsqueda y uso de recursos externos - Trabajo en equipo - Trabajo por proyectos - Aprender a trabajar con metodologías ágiles