



## **Temario Diplomado Full Stack septiembre 2024(4 meses)**

**Duración Total:** 4 meses

**Instructor:** Nazir Rosas

**Clases por Semana:** 1 (lunes 20:00 pm a 22:00 pm hora de la Ciudad de México).

**Duración de Cada Clase:** 2 horas

**Total de Clases:** 16

**Nota:** Las clases que caigan en días festivos se repondrán el siguiente día hábil según lo indique la coordinación académica.

---

### **Mes 1: Fundamentos de Front-end (HTML, CSS, JavaScript)**

#### **Clase 1: Introducción a la Web y HTML Básico**

- **Teoría (20% - 24 min):**
  - Estructura básica de una página web.
  - Introducción a HTML, etiquetas básicas y semánticas.
- **Práctica (80% - 96 min):**
  - Creación de una página HTML básica.

#### **Clase 2: Formularios y Tablas en HTML**

- **Teoría (20%):**
  - Formularios y tablas en HTML.
- **Práctica (80%):**
  - Implementación de formularios y tablas.

#### **Clase 3: Introducción a CSS**

- **Teoría (20%):**
  - CSS: selectores, propiedades y modelo de caja.
- **Práctica (80%):**
  - Estilización de una página con CSS.

#### **Clase 4: Diseño Responsivo y Preprocesadores CSS**

- **Teoría (20%):**
  - Diseño responsivo y preprocesadores CSS (Sass).
- **Práctica (80%):**
  - Creación de una página responsiva usando media queries.

#### **Clase 5: Fundamentos de JavaScript**

- **Teoría (20%):**
  - Sintaxis básica de JavaScript, estructuras de control.
- **Práctica (80%):**
  - Creación de scripts básicos para interactividad.

#### **Clase 6: Manipulación del DOM y Eventos**

- **Teoría (20%):**
  - Introducción al DOM y eventos en JavaScript.
- **Práctica (80%):**
  - Manipulación del DOM y eventos de usuario.

---

#### **Evaluación Mes 1: Avance del Proyecto (HTML, CSS, JavaScript)**

- **Objetivo:** Los estudiantes presentarán la estructura y estilización de su proyecto.
  - **Tiempo de Evaluación:** Se dedicará la última hora de la clase 6 para revisar avances.
-

## Mes 2: Front-end Avanzado (React, Angular)

### Clase 7: Introducción a React

- **Teoría (20%):**
  - Componentes, JSX, y estado en React.
- **Práctica (80%):**
  - Creación de una aplicación básica con componentes.

### Clase 8: Rutas y Navegación en React

- **Teoría (20%):**
  - Enrutamiento en React.
- **Práctica (80%):**
  - Implementación de rutas con React Router.

### Clase 9: Introducción a Angular

- **Teoría (20%):**
  - Fundamentos de Angular: componentes y módulos.
- **Práctica (80%):**
  - Creación de una aplicación básica con Angular.

### Clase 10: Servicios y Consumo de APIs en Angular

- **Teoría (20%):**
  - Servicios y manejo de HTTP en Angular.
- **Práctica (80%):**
  - Implementación de servicios y consumo de una API.

---

### Evaluación Mes 2: Avance del Proyecto (React o Angular)

- **Objetivo:** Los estudiantes agregarán interactividad avanzada y enrutamiento.
- **Tiempo de Evaluación:** Última hora de la clase 10.

## **Mes 3: Fundamentos de Back-end**

### **Clase 11: Introducción al Back-end y Node.js**

- **Teoría (20%):**
  - Conceptos de back-end, Node.js y Express.
- **Práctica (80%):**
  - Creación de un servidor con Express.

### **Clase 12: Bases de Datos SQL**

- **Teoría (20%):**
  - Conceptos de bases de datos relacionales y SQL.
- **Práctica (80%):**
  - Creación de una base de datos y consultas SQL básicas.

### **Clase 13: Bases de Datos NoSQL y MongoDB**

- **Teoría (20%):**
  - Conceptos de NoSQL y MongoDB.
- **Práctica (80%):**
  - Creación de una base de datos MongoDB y operaciones CRUD.

### **Clase 14: APIs RESTful**

- **Teoría (20%):**
  - Diseño de APIs RESTful y métodos HTTP.
- **Práctica (80%):**
  - Creación de una API RESTful con Express o Django.

---

## **Evaluación Mes 3: Avance del Proyecto (Back-end y API)**

- **Objetivo:** Implementar el back-end y la API para su proyecto.

- **Tiempo de Evaluación:** Última hora de la clase 14.
- 

## **Mes 4: Librerías y Frameworks de Back-end**

### **Clase 15: Autenticación y Seguridad**

- **Teoría (20%):**
  - Autenticación, autorización y buenas prácticas de seguridad.
- **Práctica (80%):**
  - Implementación de autenticación de usuarios en la API.

### **Clase 16: Deployment y Presentación del Proyecto Final**

- **Teoría (20%):**
    - Conceptos de deployment y CI/CD.
  - **Práctica (80%):**
    - Despliegue del proyecto en producción y presentación final.
- 

### **Evaluación Final del Proyecto**

- **Objetivo:** Presentación del proyecto final completo, incluyendo front-end, back-end y despliegue.
  - **Tiempo de Evaluación:** Clase 16 completa para la exposición y evaluación final.
- 

Este temario incluye la reposición de clases en caso de días festivos, para asegurar que el programa completo se cubra en las 16 sesiones previstas.

### **Sugerencias para Proyectos Finales**

## 1. Tienda en Línea (E-commerce)

- **Descripción:** Desarrollar una tienda en línea que permita a los usuarios navegar por productos, añadirlos al carrito de compras, realizar pedidos, y gestionar el proceso de pago (simulado).
- **Características:**
  - Front-end: Interfaz responsiva con React o Angular.
  - Back-end: API RESTful para gestionar productos, usuarios y pedidos.
  - Base de Datos: MongoDB o SQL para almacenar productos y pedidos.
  - Autenticación de usuarios para la creación de cuentas y proceso de compra.
  - Despliegue en un servicio cloud como Heroku o Vercel.

## 2. Aplicación de Gestión de Tareas (To-Do App)

- **Descripción:** Crear una aplicación de gestión de tareas que permita a los usuarios crear, editar, eliminar y organizar tareas por categorías y fechas de vencimiento.
- **Características:**
  - Front-end: Aplicación interactiva con React o Angular para gestionar las tareas.
  - Back-end: API RESTful para el manejo de tareas y usuarios.
  - Base de Datos: Almacenar las tareas en una base de datos SQL o NoSQL.
  - Autenticación de usuarios para que cada uno tenga su propio listado de tareas.
  - Opciones para marcar tareas como completadas o en progreso.

### 3. Red Social Básica

- **Descripción:** Crear una red social simple donde los usuarios puedan registrarse, crear publicaciones, comentar en las publicaciones de otros, y seguir a otros usuarios.
- **Características:**
  - Front-end: Aplicación responsiva con diseño de feed de publicaciones en React o Angular.
  - Back-end: API para manejar usuarios, publicaciones y comentarios.
  - Base de Datos: MongoDB o SQL para almacenar las relaciones de usuarios, publicaciones y comentarios.
  - Autenticación con login y registro de usuarios.
  - Sistema de "seguir" a otros usuarios para ver sus publicaciones en un feed personalizado.

### 4. Blog con Panel de Administración

- **Descripción:** Crear un sistema de blog donde los administradores puedan escribir y publicar artículos, y los usuarios puedan leer, comentar y compartir artículos.
- **Características:**
  - Front-end: Página web responsiva donde se pueda navegar por los artículos del blog.
  - Back-end: API RESTful para gestionar los artículos y los comentarios.
  - Base de Datos: Almacenamiento de artículos y comentarios en SQL o NoSQL.
  - Panel de administración para que los administradores puedan crear y gestionar artículos.
  - Autenticación de usuarios con roles (administrador y usuario común).

### 5. Aplicación de Reservas de Eventos

- **Descripción:** Desarrollar una aplicación donde los usuarios puedan ver y reservar entradas para eventos como conciertos, conferencias o espectáculos.
- **Características:**
  - Front-end: Interfaz moderna y responsiva para ver eventos disponibles y hacer reservas.
  - Back-end: API para gestionar eventos y reservas.

### Tiempo Estimado para Realización de los Proyectos

- **Desarrollo del Front-end:** 2-3 semanas.
- **Desarrollo del Back-end y API:** 2-3 semanas.
- **Integración y Despliegue:** 1-2 semanas.

### Requisitos para el Proyecto Final

1. **Front-end:** Implementación de una interfaz responsiva e interactiva utilizando React o Angular.
2. **Back-end:** Desarrollo de una API RESTful utilizando Node.js con Express o Django.
3. **Base de Datos:** Conexión a una base de datos SQL o NoSQL para almacenar los datos relevantes.
4. **Autenticación:** Sistema básico de autenticación para usuarios registrados.
5. **Despliegue:** Proyecto desplegado en un entorno accesible públicamente (Heroku, Vercel, etc.).