

Objetivos del Taller

El objetivo es proporcionar a los desarrolladores las habilidades y conocimientos necesarios para dominar React, permitiéndoles construir interfaces de usuario dinámicas, interactivas y de alto rendimiento de manera modular y escalable.

Durante el Taller, los participantes aprenderán a crear aplicaciones modernas utilizando las mejores prácticas del desarrollo frontend.

El enfoque estará en enseñar cómo utilizar React para construir Single Page Applications (SPA), implementando conceptos fundamentales como componentes reutilizables, Hooks y gestión de estado con herramientas como Redux. Además, se abordará la integración con librerías externas, el diseño de interfaces responsivas con Material UI y la autenticación con servicios como Firebase.

A lo largo del Taller, se cubrirán temas avanzados, incluyendo la protección de rutas, el desarrollo de Custom Hooks, pruebas unitarias con Vitest y la optimización de aplicaciones listas para producción. Los participantes trabajarán en ejercicios prácticos progresivos y en un proyecto final integrador que aplicará todos los conceptos aprendidos.

Con un enfoque altamente práctico y orientado a proyectos reales, los alumnos adquirirán la capacidad de crear aplicaciones eficientes, mantenibles y escalables. Al finalizar el Taller, los participantes estarán preparados para enfrentar los desafíos del desarrollo moderno con React, aplicando las mejores prácticas del ecosistema y entregando soluciones de calidad en cualquier entorno profesional.

Contenidos

Módulo 1: Primeros pasos en React

- Introducción a React: ¿Qué es y por qué usarlo?
- Configuración del entorno: Node.js, npm/yarn y herramientas necesarias.
- Creación de un proyecto con **Vite** o **Create React App**.
- Estructura básica de un componente React.
- **JSX**: Sintaxis básica y buenas prácticas.

Módulo 2: Pruebas Unitarias y de Integración

- Introducción a las pruebas en React.
- Herramientas: **Vitest**, Jest y Testing Library.
- Creación de pruebas unitarias en componentes simples.
- Pruebas de integración: Validación de flujos y eventos.

Módulo 3: Generando el Build de Producción y Despliegue

- Optimización del build para producción.
- Configuración de **vite.config** o webpack.
- Despliegue en plataformas como **Netlify** y GitHub Pages.
- Buenas prácticas de optimización de rendimiento.

Módulo 4: Hooks en Profundidad

Parte 1: Hooks Generales

- Introducción a los **Hooks**: useState, useEffect y useRef.
- Gestión del ciclo de vida del componente con useEffect.
- Ejercicios prácticos de manejo de estado y efectos secundarios.

Parte 2: Hooks Avanzados

- useReducer: Gestión avanzada del estado local.
- useContext: Context API para compartir datos entre componentes.
- Creación de **Custom Hooks** reutilizables.

Módulo 5: Single Page Applications (SPA)

- Conceptos de **Single Page Application**.
- Enrutamiento con **React Router**:
- Configuración básica y rutas anidadas.
- Parámetros de rutas dinámicas.
- Ejemplo práctico: Creación de una SPA completa.

Módulo 6: Protección de Rutas

- Introducción a la autenticación y autorización.
- Protección de rutas en React usando **React Router**.
- Ejercicio práctico: Creación de un flujo de autenticación básico.

Módulo 7: Integración de Diseño y Estilización

- Estilización de componentes:
- **CSS Modules** y **styled-components**.
- Uso de librerías de componentes como **Material UI**.
- Creación de interfaces atractivas y responsivas.
- Ejemplo práctico: Diseño de una aplicación con **Material UI**.

Módulo 8: Manejo del Estado Global con Redux

Parte 1: Introducción a Redux

- ¿Qué es Redux y cuándo usarlo?
- Principios básicos: store, actions y reducers.
- Configuración de **Redux Toolkit** en un proyecto React.

Parte 2: Integración Práctica con Firebase

- Configuración de **Firebase** para autenticación.
- Creación de un flujo de autenticación en React usando Redux.
- Ejemplo práctico: Persistencia de datos con Firestore.

Parte 3: CRUD en Firestore y Manejo de Archivos

- Creación de un **CRUD** básico con Firestore.
- Subida y gestión de archivos en Firebase Storage.
- Ejercicio práctico: Desarrollo de una funcionalidad completa.

Metodología

- **70% práctica y 30% teoría.**
- Desarrollo progresivo de un proyecto real a medida que se cubren los módulos.
- Resolución de problemas y optimización de código en cada etapa.
- Retroalimentación constante y revisión del código.

Estrategia Metodológica

- Desarrollo Modular y Gradual
- Enfrentando Problemas Reales
- Prácticas Colaborativas
- Optimización y Revisión de Código
- Implementación en Ambientes Reales
- Retroalimentación Personalizada

Dirigido a

Desarrolladores de aplicaciones

Requisitos participantes

Conocimientos en JavaScript y TypeScript o en algún lenguaje de programación.

Generalidades

- Duración 36 horas cronológicas
- Taller cerrado.