

React

Patrones y componentes avanzados

¿Qué voy a aprender?

- Aprende patrones y componentes avanzados que te ayudarán a mejorar el código de tu aplicación desarrollada con React.
- Mejora la experiencia de tus usuarios con React.Suspense y la carga de componentes en segundo plano
- Conoce Redux y cómo utilizar un store global para almacenar el estado de tu aplicación
- Aprende cómo gestionar llamadas a una API desde React
- Conoce los aspectos básicos de accesibilidad y cómo hacer tus aplicaciones más accesibles
- Testea tus componentes y tu aplicación completa con Jest, Enzyme y Puppeteer.
- Optimiza tu código para producción. Carga sólo lo necesario en cada momento y haz que tu código sea compatible con navegadores antiguos
- Por último, convierte tu aplicación en una PWA. Almacena tu código en el cliente y haz que tu aplicación cargue de manera instantánea

Plazas Limitadas

Duración: 5 semanas (50 horas aprox. Incluyendo lecciones en video, apuntes, realización de ejercicios y práctica final.)

Online: Accede cuando y desde donde quieras.

Tutelados: Seguimiento continuo. Foro de dudas y consultas con respuesta en 24 horas laborables. Corrección de práctica.

Bonificable por Fundae.

Precio: 375€ + Iva*

* Consultar promociones en www.trainingit.es

¿Qué conocimientos son necesarios?

- Conocimientos básicos de desarrollo web orientado a componentes
- Uso de la terminal. Navegación entre carpetas, ejecución de comandos, etc.
- Manejo de paquetes con yarn o npm
- Conocimientos básicos de React:
 - ✓ Cómo crear un proyecto
 - ✓ Creación de componentes de clase y funciones
 - ✓ Uso del estado de los componentes
 - ✓ JSX. Condicionales, bucles, etc.
- Webpack. Configuración básica del proyecto y de plugins
- Uso de las herramientas de debug del navegador





¿Qué programas y materiales son necesarios?

- Un editor de código. VSCode es gratuito, bastante rápido y tiene muchos plugins para acelerar nuestro desarrollo con React. Es el que se utilizará durante el curso.
- Una terminal donde poder ejecutar nuestros comandos
- Nodejs y yarn instalados. Yarn es opcional, todos los comandos se pueden extrapolar a npm
- Un navegador con capacidades de debug, como Chrome o Firefox. Durante el curso utilizaremos Chrome

Profesor

Ángel María De Miguel Meana

Inquieto y creativo. Actualmente trabaja como UX Engineer en Bitnami, diseñando y desarrollando las interfaces de sus productos. Estudió Ingeniería de Telecomunicaciones, pero poco a poco ha ido ampliando sus conocimientos hasta centrarse en el diseño y desarrollo de interfaces y la interacción con usuarios.



“Me encanta probar nuevas tecnologías y colaborar en proyectos open source, además de impartir cursos y charlas”.

Acerca de TrainingIT

TrainingIT es una iniciativa en colaboración con **HiringIT** para dar formación bonificable de altísima calidad en temas relacionados con IT: Desarrollo back y front, sistemas, bases de datos, data science, inteligencia artificial, big data...

Nos diferenciamos sobre todo por la calidad de los cursos, que nacen con la mentalidad de un curso presencial: profesor dedicado, ejercicios, tests, prueba final, tutorías, diploma de certificación...

Contáctanos en info@trainingit.es o en el 656.27.65.04.





Temario del curso

1. Componentes y patrones avanzados
 1. Async rendering en React
 2. Fragments
 3. Error boundaries
 4. High Order Components (HOC)
 5. Context API
 6. Referencias (ref)
 7. Portals
 8. Gestión de formularios
 9. Pure components (PureComponent and React.memo)
 10. Introducción a Hooks
2. Suspense y lazy loading
 1. Concurrent rendering
 2. React.Suspense
 3. Lazy loading
3. Routers en React
 1. React-router
 2. Parámetros en rutas
 3. Routing imperativo
 4. Rutas privadas
 5. 404
4. Redux
 1. Flux y redux
 2. Store
 3. Reducers y acciones
 4. Middlewares
 5. ImmutableJS
5. Conectando con una API
 1. Llamadas con Fetch
 2. Acciones asíncronas en Redux
 1. Thunks
 2. Promise
 3. Saga
6. Accesibilidad
 1. WAI-ARIA
7. Testing
 1. Jest
 2. Enzyme
 3. Mocks
 4. Snapshot testing
 5. Tests de integración con Puppeteer
8. En producción
 1. ¿Qué hay que tener en cuenta para producción?
 2. Modos en webpack
 3. Compatibilidad con navegadores (Polyfills)
 4. Code-splitting
 5. Tree-shaking
 6. Source maps
9. Progressive Web Applications (PWA)
 1. PWAs
 2. Manifest
 3. Service workers
 4. Versionando nuestro código
10. Bonus: Introducción a SSR con NextJS
 1. SPAs
 2. NextJS

