



Biblioteca React JS

Javascript de código abierto

× × ×



Introducción

React.js es el marco de JavaScript de front-end más conocido. Los desarrolladores usan JSX, una combinación de HTML y JavaScript, para crear vistas de forma natural.



Definición

React es una biblioteca Javascript de código abierto diseñada para crear interfaces de usuario con el objetivo de facilitar el desarrollo de aplicaciones en una sola página.





Características



TS

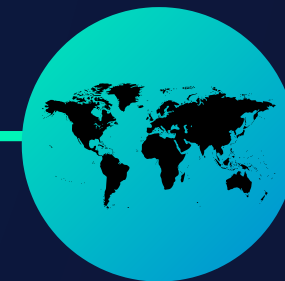
01. Componentes

React está basado en la componetización de la UI. La interfaz se divide en componentes independientes, que contienen su propio estado.



02. Virtual DOM

React usa un DOM virtual para renderizar los componentes. El DOM virtual es una representación en memoria del DOM real.



03. Universal

React se puede ejecutar tanto en el cliente como en el servidor. Además, puedes usar React Native para crear aplicaciones nativas para Android e iOS.





Ventajas de El uso de react

01

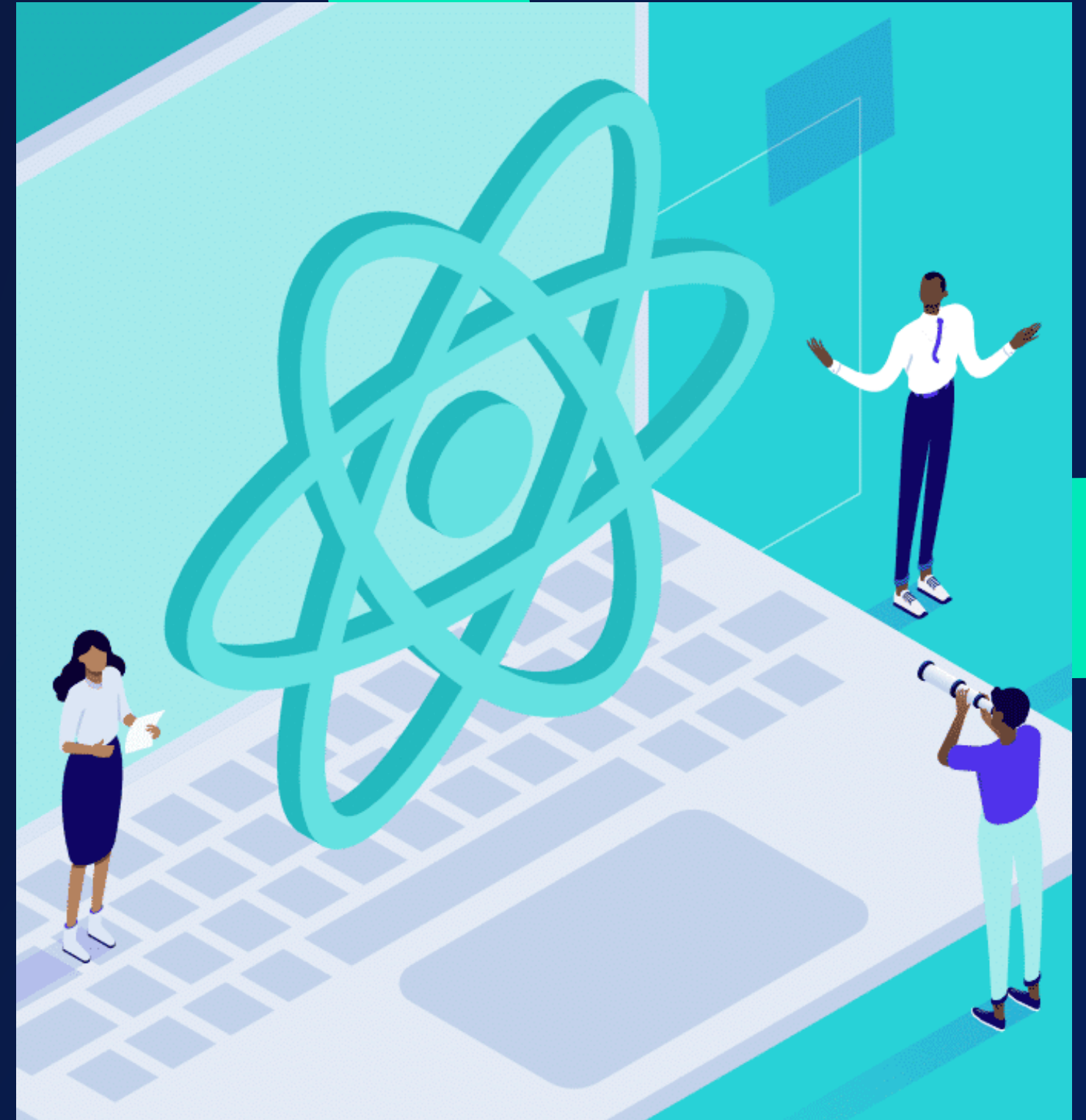
La biblioteca proporciona a los desarrolladores una forma rentable de crear aplicaciones multiplataforma, por ejemplo, en lugar de crear dos aplicaciones diferentes para Android e iOS.

02

ReactJS tiene la capacidad de acelerar el desarrollo de aplicaciones, y los desarrolladores pueden usar múltiples componentes listos para usar para crear la funcionalidad de la aplicación más rápido que nunca.

03

JavaScript es el lenguaje de programación más utilizado en el mundo, lo que hace que ReactJS sea fácil de usar para los desarrolladores de JavaScript porque el marco está escrito en JavaScript.





Hooks

Los hooks en React son una nueva adición a la sintaxis de esta librería JavaScript. Este elemento nos permite reemplazar los métodos de estado y ciclo de vida de componentes en React, una serie de funciones que empiezan con «use» y que nos permiten definir secciones de código a ejecutar en momentos específicos de render.

Los Hooks son funciones que te permiten “enganchar” el estado de React y el ciclo de vida desde componentes de función. Los hooks no funcionan dentro de las clases — te permiten usar React sin clases.



Flexibilidad

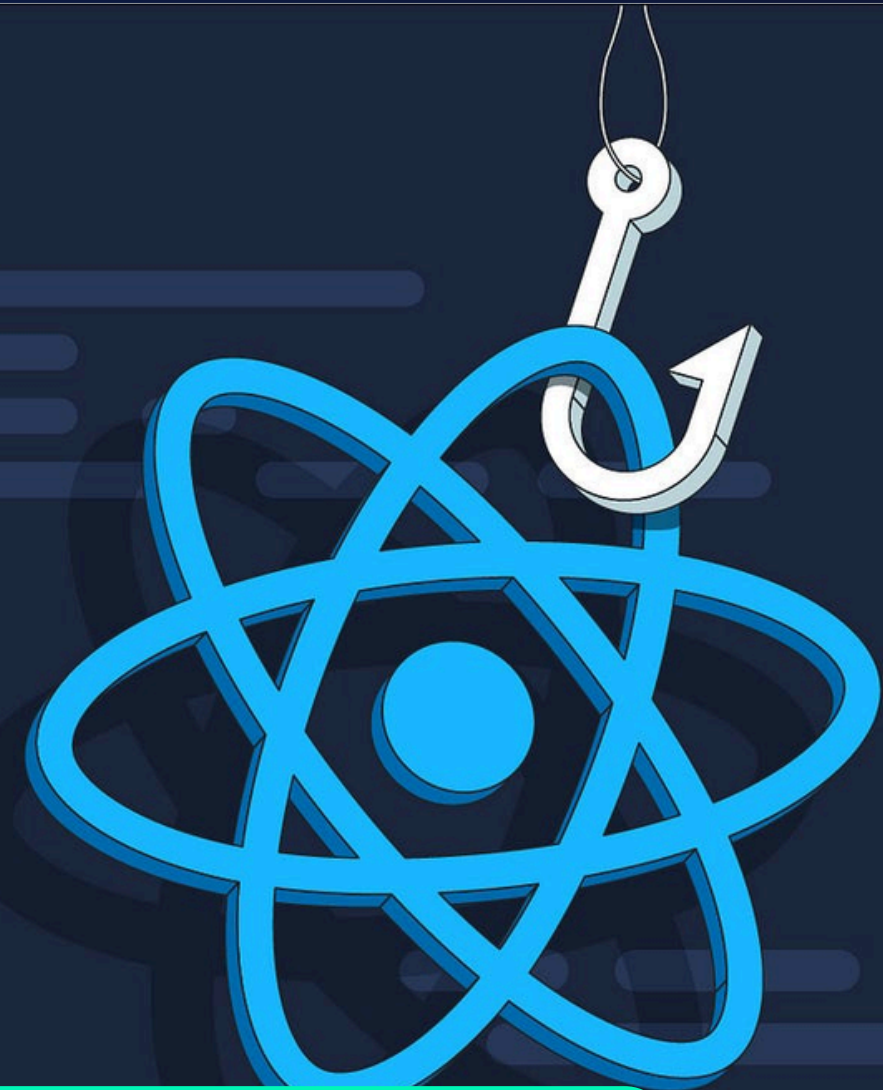


Rendimiento



Curva de
Aprendizaje

ALEGORIA A UN GANCHO



Tipos de Hooks

Tipos de Hooks En React JS

01

useState: Es una forma sencilla de crear y actualizar variables de estado en un componente de React sin tener que escribir una clase completa.

02

useCallback: El hook useCallback es una función proporcionada por React que se utiliza para optimizar el rendimiento al trabajar con componentes funcionales.

03

useMemo: Es un Hook que se utiliza para memorizar el resultado de una operación costosa de cálculo entre re-renderizaciones de un componente funcional.

04

useEffect: El Hook useEffect nos permite realizar efectos secundarios y proporciona un mecanismo para limpiar esos efectos secundarios.

05

useContext: Permite a los componentes funcionales acceder al valor de un contexto proporcionado por el componente Proveedor más cercano en la jerarquía.

06

useReducer: El useReducer es un Hook que nos permite manipular el state de nuestros componentes funcionales, esto se logra a través de: Una función reducer.



¿Que es NPX?

NPX es una herramienta que viene incluida con Node.js y npm (Node Package Manager). Su nombre es un acrónimo de Node Package execute. NPX se utiliza para ejecutar paquetes de Node.js directamente desde el repositorio npm, sin necesidad de instalarlos de forma global o local en tu sistema.

Esto puede ser especialmente útil cuando deseas ejecutar comandos o paquetes de manera ocasional, sin necesidad de mantenerlos en tu sistema. Es una extensión de npm para ejecutar una dependencia, generalmente global o sin una instalación como tal de npm . npx es un ejecutador de paquetes binarios de npm.



```
~/Desktop/react native projects
npx react-native init my-app --verbose
error Failed to load configuration of your project.
info Run CLI with --verbose flag for more details.
```

```
~/Desktop/react native projects
npx --verbose
npm verb cli /Users/jatinsharma/.nvm/versions/node/v16.15.1/bin/node /Users/jatinsharma/.nvm/versions/node/v16.15.1/bin/npm
npm info using npm@8.11.0
npm info using node@v16.15.1
npm timing npm:load:whichnode Completed in 0ms
npm timing config:load:defaults Completed in 1ms
npm timing config:load:file:/Users/jatinsharma/.nvm/versions/node/v16.15.1/lib/node_modules/npm/config/user-config.json Completed in 1ms
npm timing config:load:builtin Completed in 1ms
npm timing config:load:cli Completed in 2ms
npm timing config:load:env Completed in 0ms
npm timing config:load:file:/Users/jatinsharma/Desktop/.npmrc Completed in 1ms
npm timing config:load:project Completed in 3ms
npm timing config:load:file:/Users/jatinsharma/.npmrc Completed in 0ms
npm timing config:load:user Completed in 1ms
npm timing config:load:file:/Users/jatinsharma/.nvm/versions/node/v16.15.1/lib/node_modules/npm/config/user-config.json Completed in 1ms
npm timing config:load:global Completed in 0ms
npm timing config:load:validate Completed in 0ms
npm timing config:load:credentials Completed in 0ms
npm timing config:load:setEnvs Completed in 1ms
npm timing config:load Completed in 9ms
npm timing npm:load:configload Completed in 9ms
npm timing npm:load:mkdirpcache Completed in 1ms
npm timing npm:load:mkdirplogs Completed in 0ms
npm verb title npm exec
npm verb argv "exec" "--loglevel" "verbose"
npm timing npm:load:setTitle Completed in 14ms
npm timing config:load:flatten Completed in 2ms
npm timing npm:load:display Completed in 5ms
npm verb logfile logs-max:10 dir:/Users/jatinsharma/.npm/_logs
npm verb logfile /Users/jatinsharma/.npm/_logs/2022-08-18T14_16_24_29Z-npm-exec.log
npm timing npm:load:logFile Completed in 3ms
npm timing npm:load:timers Completed in 0ms
npm timing npm:load:configScope Completed in 0ms
npm timing npm:load Completed in 34ms
```

```
Entering npm script environment at location:
/Users/jatinsharma/Desktop/react native projects
Type 'exit' or ^D when finished
```



Axios

Axios es una librería de JavaScript que se utiliza para hacer solicitudes HTTP desde el navegador o desde un servidor usando Node.js. Está basada en la simplicidad y una de sus características principales es la facilidad de uso mientras ofrece funciones avanzadas para el manejo de solicitudes y respuestas en la API.

Esta potente librería JavaScript simplifica las solicitudes HTTP y ofrece una integración perfecta con **React.js**. Además de tener un par de ventajas sobre **Fetch API**, tiene un excelente soporte para trabajar con APIs RESTful y facilita las pruebas y la manipulación de datos.



Uso de Node.js



XML HttpRequests



Peticion y
respuesta

PETICIONES HTTP



AXIOS



Fetch

La función `fetch()` arrojará automáticamente un error por errores de red, pero no por errores HTTP como respuestas 4xx o 5xx. Para errores HTTP, podemos verificar la propiedad `response.ok` para ver si la solicitud falló y rechazar la promesa nosotros mismos llamando a `return Promise`.

Fetch es más que una app de “subir recibos”. Es una app de ofertas repleta de oportunidades para acumular un montón de puntos al instante. Aquí tienes un desglose de cada tipo de oferta que puedes encontrar en Fetch. Estas ofertas son las más comunes en Fetch.



Metodo Global



Cuenta con facil
Manejo



Asincronia con la
Red

PETICIONES HTTP



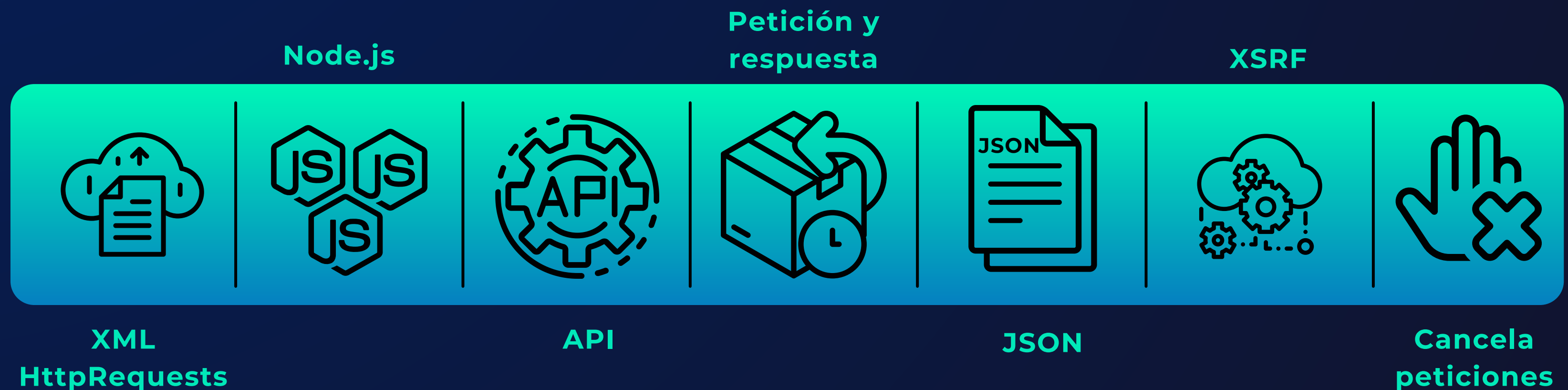
Comparacion Fecth y Axios

CARACTERISTICA	AXIOS	FETCH
SINTAXIS	Preciso y conciso con una API de desencadenamiento.	Requiere progreso adicional para json.
NAVEGADOR	Es compatible con todos los navegadores.	Compatible solamente con los navegadores modernos.
INTERCEPTORES	Soporte integrado para interceptores de solicitud/respuesta.	No es compatible directamente, pero se puede lograr con middleware.
FEXIBILIDAD	Ofrece mas características listas para usar.	Mas ligero, dismponible para implementaciones personalizadas.

¡Axios es una mejor opción!

La realidad es que Axios es mucho más completa que Fetch, pues es una librería entera y no un solo método. Esto quiere decir que nos brinda más posibilidades, como el uso de interceptors en una petición HTTP (puedes conocer este concepto en nuestro post sobre interceptors para un cliente React).

También vale la pena tomar en cuenta la potencia de las herramientas a la hora de decidir si usar Fetch o Axios. La realidad es que Axios es mucho más completa que Fetch, pues es una librería entera y no un solo método. Esto quiere decir que nos brinda más posibilidades, como el uso de interceptors en una petición HTTP.



Conclusion

ReactJS es una librería robusta de JavaScript utilizada en el desarrollo de aplicaciones web dinámicas. Hace que la codificación de JavaScript sea más sencilla, mejora el rendimiento y el SEO de tu aplicación, entre otras ventajas.

ReactJS ayuda a agilizar el proceso de depuración y reduce el riesgo de errores mediante la vinculación de datos en un solo sentido.

A continuación, un resumen de las ventajas de usar React JS:

- Es fácil de usar y aprender, con muchas lecciones de codificación disponibles en línea.
- Admite componentes reutilizables, lo que reduce el tiempo de desarrollo.
- JSX facilita la codificación y la renderización de elementos.
- El DOM virtual elimina el re-renderizado excesivo, asegurando el alto rendimiento de tu aplicación web.
- React ayuda a los motores de búsqueda a rastrear tu aplicación web, impulsando tu SEO.





Muchas Gracias

Página de Recursos

