

React

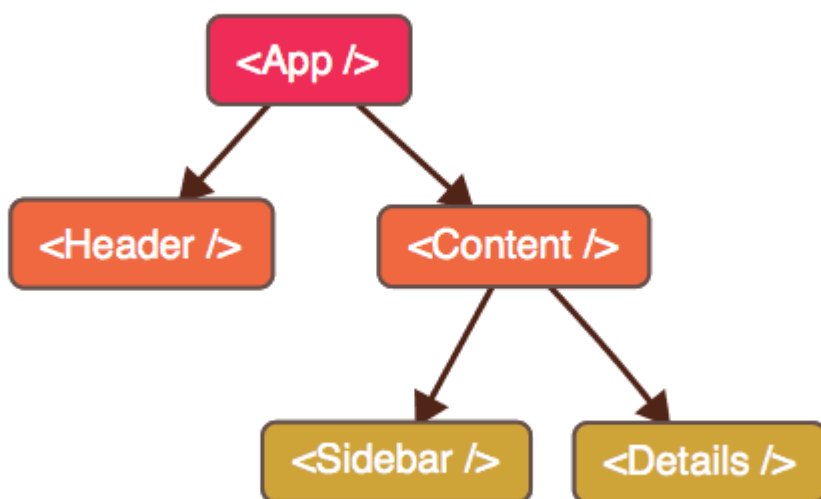
¿Qué es?

Fue desarrollada inicialmente por Facebook (ahora llamada Meta) y la ventaja es que es un software libre. Sus primeros usos fueron en aplicaciones como Facebook e Instagram, otras empresas web de primer nivel, también lo incorporaron, como Walmart, Netflix, Airbnb, Dropbox, Spotify, etc.

Se creó para hacer el desarrollo de los componentes de las interfaces del usuario más fácil y su requisito principal es ofrecer un rendimiento mayor al de las otras alternativas existentes en el mercado.

¿Cómo funciona?

Componentes



Al ser una librería moderna, nos permite crear una aplicación con conceptos novedosos como su concepto de componentes que nos permite dividir una aplicación en pequeñas partes lo que nos lleva a poder crear y administrar una aplicación compleja de una manera mucho más sencilla y entendible, ya que estos componentes de React tienen la ventaja de ser:

- Reusables
- Fácil de crear y usar
- Fácil de comprobar su funcionamiento

Para utilizar esta librería necesitamos tener un par de requisitos básicos en cuenta:

- Manipulación de HTML: sus etiquetas y el DOM.
- Entendimiento de JavaScript.
- Práctica en el uso de ECMA6 como: arrow functions, template, métodos, etc.

Aplicación de página única

Una aplicación de página única (single-page application) es una aplicación que carga una única página HTML y todos los componentes necesarios (tales como JavaScript y CSS) para que se ejecute la aplicación. Cualquier interacción con la página o páginas subsecuentes no requiere hacer solicitudes al servidor lo que significa que la página no es recargada.

Aunque se puede construir una aplicación de página única con React, esto no es un requerimiento. React también puede ser utilizado para mejorar pequeñas partes de sitios web existentes con interactividad adicional.

¿Qué diferencias hay entre crear un sitio común de cero?

Demos un paseo por el espacio de la tecnología antes de 2015, cuando el desarrollo web se trataba de secuencias de comandos y renderizado. La época en que lenguajes como HTML, CSS dominaban el frontend y PHP dominaba el backend.

El desarrollo web era distinto en ese entonces. Todo lo que teníamos que hacer era maquetar páginas HTML estáticas en algunas carpetas y renderizarlas usando PHP. Aunque esa no es una forma única e intuitiva de desarrollar sitios web, aún pudiendo establecer una conexión bidireccional entre el cliente y el servidor.

Todo el mérito es para la representación del lado del servidor (SSR). Hemos estado construyendo aplicaciones web de esta manera durante mucho tiempo, pero lo que no vimos venir es la revolución de los sitios web después de las librerías Javascript como Reactjs.

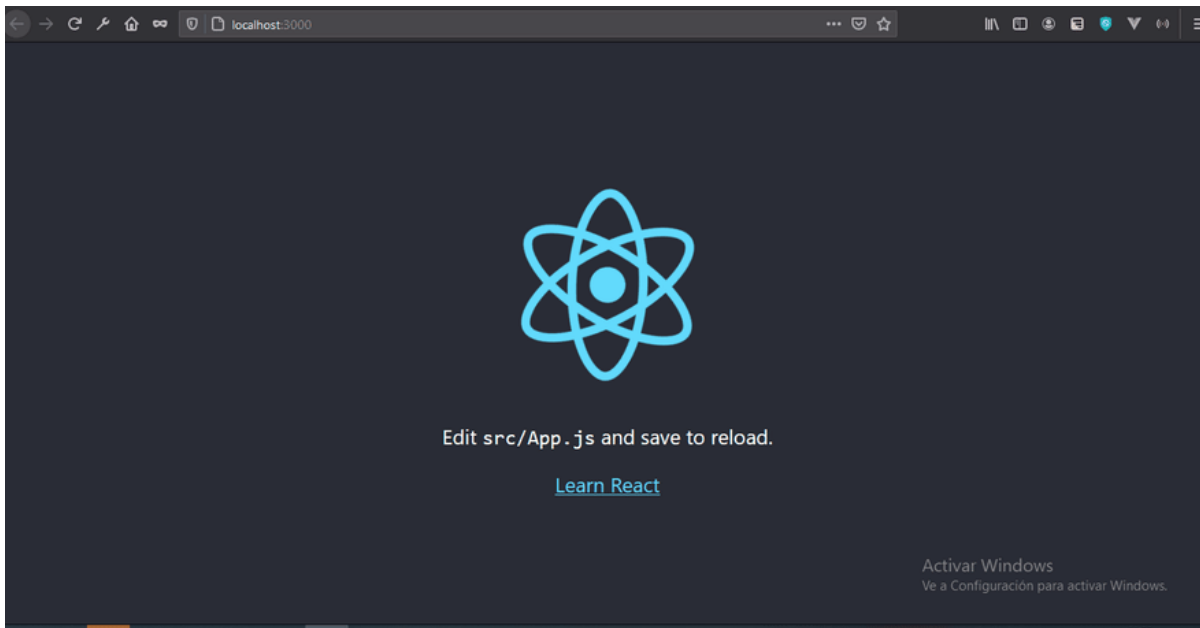
¿Cómo crear un proyecto usando create-react-app?

Para poder crear un nuevo proyecto de React.js usando `create-react-app` necesitamos tener instalado en nuestro ordenador tanto `node.js` como `npm`.

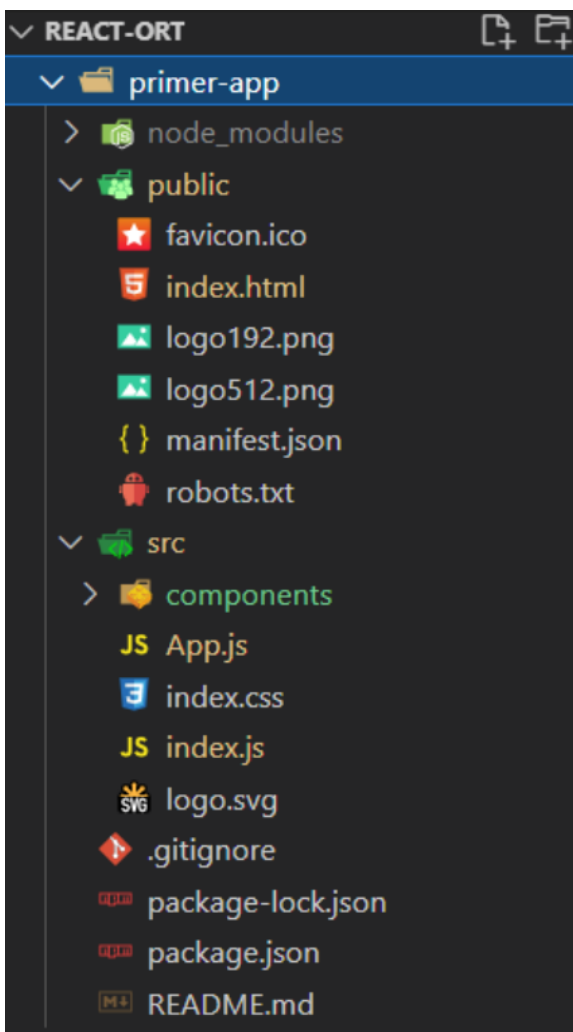
Una vez que cuenten con `node` y `npm` bastará con seguir los siguientes pasos:

1. En una terminal (y en el directorio de su preferencia) ejecutar: `npm install -g create-react-app`. Este comando instalará de manera global el paquete `create-react-app`.
2. Para verificar que el paquete ha sido instalado con éxito, ejecutar en la consola: `create-react-app` y esperar un mensaje de instalación satisfactoria.
3. Ya para finalizar, podemos crear un nuevo proyecto con la siguiente sintaxis de comando: `create-react-app nombre-del-proyecto`

Una vez finalizada la creación del proyecto se puede ejecutar `npm start` que inicializará un servidor de desarrollo (por lo general en el puerto 3000)



Sistema de archivos de React



1 - Nombre de mi proyecto: el nombre que le dimos al instalar vite.

- **2 - node_modules:** es un directorio que se crea en la carpeta raíz de nuestro proyecto cuando instalamos paquetes o dependencias mediante npm.
- **3 - src:** directorio en donde se encontrarán todos los archivos referidos a React
- **4 - Index.css:** archivo de CSS que usa el archivo App.js
- **5- favicon.ico:** si queremos que tenga un ícono como imagen la url al navegar por nuestro sitio.
- **6 - index.css:** archivo de CSS global que usa el sitio.
- **7 - logo.svg:** imagen que se encuentra en App.jsx
- **8 - .gitignore:** Si vamos a usar GIT, ya tendremos éste archivo preparado para no subir archivos a github que no sean necesarios.
- **9 - index.html:** Donde se inicializará nuestro proyecto, y podremos incluir en el "head" llamadas a otros archivos, tales como CSS, JS, etc.
- **10 - package-lock.json:** se genera automáticamente para cualquier operación en la que npm modifique el árbol node_modules o package.json. Describe una representación única de un árbol de dependencias de modo que se garantice que los

compañeros de equipo, las implementaciones y la integración continua instalen exactamente las mismas dependencias.

- **13 - package.json:** contiene todos los metadatos acerca del proyecto tal como descripción, licencia, dependencias y scripts.