

EJERCICIO 1.

- ✓ **Investiga la historia de React y menciona dos hitos importantes en su desarrollo.**

Respuesta.

2007-2011: Los Primeros Pasos de React

La historia de React comenzó en Facebook a finales de la década de 2000 en 2007, Jordan Walke, un ingeniero de software en Facebook comenzó a trabajar en una biblioteca llamada **FaxJS** para abordar los desafíos de la actualización eficiente de la interfaz de usuario en la plataforma. Más tarde, en 2011, Jordan presentó **FaxJS** a un pequeño equipo de ingenieros de Facebook, y esto marcó el comienzo de lo que eventualmente se convertiría en React.

Como dato curioso, quiero mencionar que parte de la inspiración de Walke para la creación de esta librería nace del uso de XHP, una librería de componentes para PHP.

2015: Introducción de React Native

Un gran hito en la historia de React fue la introducción de React Native en 2015. React Native permitió a los desarrolladores crear aplicaciones móviles nativas para iOS y Android utilizando la misma base de código de React. Esto revolucionó el desarrollo móvil al proporcionar una forma más eficiente de crear aplicaciones multiplataforma de alto rendimiento.

2018: React 16 y Hooks

En octubre de 2018, React 16 introdujo una característica revolucionaria llamada **Hooks** los Hooks permitieron a los desarrolladores utilizar el estado y otras características de React en componentes funcionales, eliminando la necesidad de clases en gran medida y simplificando el código. Esto cambió la forma en que se escriben los componentes en React y se convirtió en una parte integral de la biblioteca.

<https://www.paradigmadigital.com/dev/hooks-como-utilizarlos-react/>

<https://www.itdo.com/blog/la-era-de-react/>

- ✓ **Escribe una breve explicación sobre por qué Facebook decidió crear React.**

Respuesta.

Ya que anteriormente la construcción de interfaces de usuario interactivas en aplicaciones web eran complicadas y propensas a errores debido a la complejidad en la gestión de las actualizaciones del DOM y el estado de la aplicación.

EJERCICIO 2.

- ✓ **Menciona tres ventajas de usar React en el desarrollo de aplicaciones web.**

Respuesta.

1. **Componentes reutilizables:** React permite crear componentes reutilizables que se pueden combinar fácilmente para crear interfaces de usuario complejas esto facilita el desarrollo y el mantenimiento de aplicaciones.
2. **Representación rápida:** el DOM virtual y los componentes funcionales de React permiten representar rápidamente actualizaciones en la interfaz de usuario.
3. **Sintaxis declarativa:** React utiliza una sintaxis declarativa, lo que facilita que los desarrolladores comprendan y razonen sobre el código.

- ✓ **Explica cómo el Virtual DOM mejora el rendimiento de una aplicación.**

Respuesta.

Mejora el rendimiento en una aplicación al optimizar el proceso de actualización de la interfaz de usuario en lugar de manipular directamente el DOM real, el Virtual DOM crea una representación ligera y en memoria de la estructura del DOM. Cuando se realizan cambios en la aplicación, el Virtual DOM calcula de manera eficiente las diferencias entre el nuevo estado y el anterior mediante un proceso llamado **diffing** luego, aplica solo las actualizaciones necesarias al DOM real minimizando el número de manipulaciones directas y, por ende, mejorando la velocidad y la eficiencia de la interfaz de usuario optimizando las actualizaciones y disminuyendo la carga sobre el navegador, lo que resulta en una experiencia de usuario más rápida y fluida.

EJERCICIO 3.

- ✓ **Define qué es una Single Page Application (SPA).**

Respuesta.

Es una página en la cual todo el contenido está dentro de un solo archivo, es decir, solo se carga un archivo HTML y todo el contenido está dentro de este luego, se actualiza dinámicamente su contenido sin necesidad de cargar páginas adicionales del servidor.

- ✓ **Explica cómo React facilita la creación de una SPA. Proporciona un ejemplo de cómo un componente de React puede actualizar la interfaz sin recargar la página.**

Respuesta.

Facilita la creación ya que al permitir que los desarrolladores construyan interfaces de usuario interactivas y dinámicas en una SPA, toda la aplicación se carga una sola vez en el navegador y, a medida que el usuario interactúa con la aplicación, solo se actualizan las partes necesarias de la interfaz sin recargar la página completa. Esto se logra mediante el uso de componentes reutilizables y el manejo eficiente del estado.

EJERCICIO 4.

- ✓ Crea un nuevo proyecto React utilizando Create React App.
- ✓ Inicia el servidor de desarrollo y comparte una captura de pantalla de tu proyecto en ejecución.

Respuesta.

```
factura html y css C:\Users\frede\OneDrive\Escritorio
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\frede> cd OneDrive
PS C:\Users\frede\OneDrive> npx create-react-app sofiar-app

Creating a new React app in C:\Users\frede\OneDrive\sofiar-app.

Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts with cra-template...

added 1480 packages in 4m

261 packages are looking for funding
  run `npm fund` for details
```

```
factura html y css C:\Users\frede\OneDrive\Escritorio
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

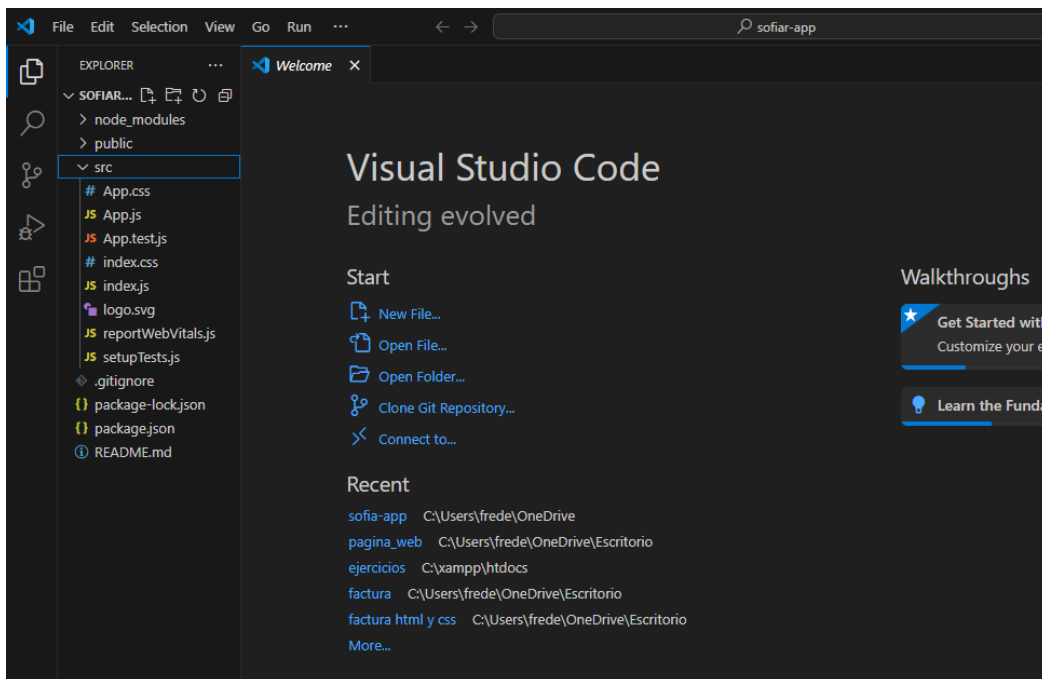
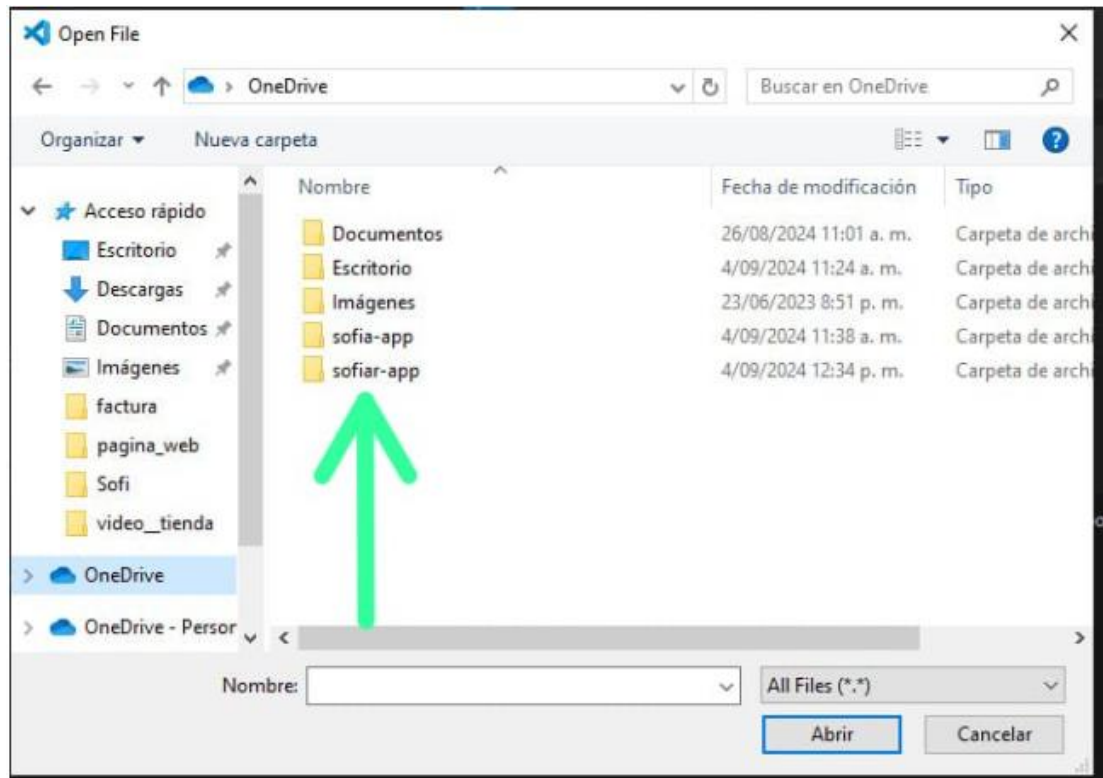
Starts the test runner.

npm run eject
  Removes this tool and copies build dependencies, configuration files
  and scripts into the app directory. If you do this, you can't go back!

We suggest that you begin by typing:

  cd sofiar-app
  npm start

Happy hacking!
PS C:\Users\frede\OneDrive> |
```



```
ejercicios C:\xampp\htdocs
factura C:\Users\frede\OneDrive\Escritorio
factura html y css C:\Users\frede\OneDrive\Escritorio

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

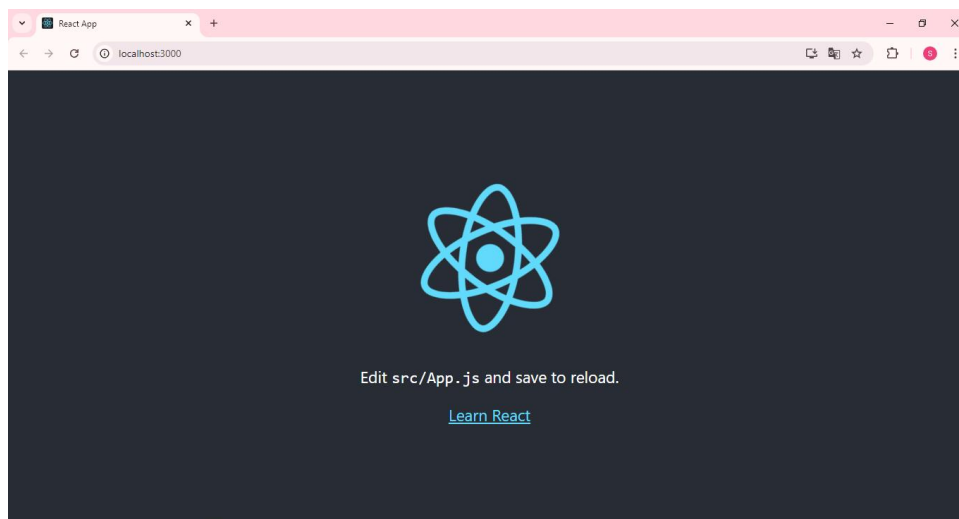
Compiled successfully!

You can now view sofiar-app in the browser.

Local: http://localhost:3000
On Your Network: http://192.168.1.37:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```



EJERCICIO 5.

- ✓ **Explica brevemente el propósito de las carpetas src y public en un proyecto React.**

Respuesta.

Src: Contiene el código fuente de la aplicación React (componentes, vistas, recursos, funciones útiles y más). que contiene toda la aplicación en React.

Public: Es para archivos estáticos (Webpack en el caso de create react app). Estos archivos son servidos directamente al navegador y se mantienen sin cambios.

EJERCICIO 6.

- ✓ **Explica cómo JSX se diferencia del HTML tradicional.**

Respuesta.

JSX es una extensión de JavaScript utilizada en React que permite escribir código similar a HTML dentro de JavaScript, mientras que HTML es un lenguaje de marcado estático que define la estructura de la página y necesita JavaScript separado.

EJERCICIO 7.

- ✓ **Define los roles principales en un equipo SCRUM.**

Respuesta.

Product Owner: Responsable de definir la visión del producto y gestionar el product backlog.

Scrum Máster: Facilita el proceso scrum, elimina obstáculos y asegura que el equipo siga las practicas scrum.

Equipo de desarrollo: Se encarga de construir y entregar incrementos del producto en cada sprint.

- ✓ **Explica qué es un sprint y cómo se planifica.**

Respuesta.

Un sprint es un ciclo de desarrollo de duración fija (generalmente de 2-4 semanas) durante el cual el equipo trabaja para completar una cantidad determinada de trabajo del product backlog.

Daily Stand-ups: Reuniones diarias breve en las que cada miembro del equipo comparte lo que hizo ayer, lo que planea hacer hoy, y si tiene algún impedimento.

Sprint Planning: Reuniones al inicio de cada sprint para decir que tareas se van a abordar durante el sprint.

Sprint Review: Reunión al final de cada sprint para demostrar lo que se ha completado y recibir retroalimentación.

Sprint Retrospective: Reunión para reflexionar sobre lo que salió bien, y lo que no, y como mejorar en el próximo sprint.

ANA SOFIA HURTADO RAIGOSA