

Taller de React

Nombre: Estefany Del Carmen Muñoz Carrascal

1.Ejercicio:

Investiga la historia de React y menciona dos hitos importantes en su desarrollo

R/ React es una biblioteca desarrollada en 2011 por un ingeniero de software en Facebook llamado Jordan Walke para resolver problemas de rendimiento en las aplicaciones web de la compañía. Inicialmente, se usó internamente en Facebook para la sección de noticias, y al notar mejoras en eficiencia y experiencia del usuario, Facebook decidió compartirla con la comunidad de desarrolladores.

En 2013, React se lanzó como un proyecto de código abierto en la conferencia JSConf US. Aunque la recepción inicial fue mixta debido a su enfoque en el Virtual DOM y el uso de JSX (una extensión de sintaxis de JavaScript), pronto ganó popularidad por su enfoque innovador en la creación de interfaces de usuario dinámicas y eficientes.

Dos hitos importantes en el desarrollo de React

Lanzamiento de React 16

Esta versión marcó una gran evolución en la biblioteca. Introdujo el motor de reconciliación "Fiber" y mejoró el rendimiento con un manejo de errores más robusto y nuevas características como fragmentos.

Introducción de los Hooks

Los Hooks permitieron a los desarrolladores usar estado y otras características de React en componentes funcionales, simplificando el código y mejorando la reutilización de lógica.

Escribe una breve explicación sobre por qué Facebook decidió crear React

R/ Facebook decidió crear React para abordar varios desafíos que enfrentaban en el desarrollo de sus interfaces de usuario, especialmente en la gestión de la actualización de la interfaz en aplicaciones web grandes y complejas.

2.Ejercicio:

Menciona tres ventajas de usar React en el desarrollo de aplicaciones web

1. Reutilización de componentes: React permite construir aplicaciones web mediante componentes reutilizables. Cada componente es independiente, lo que permite reutilizar y actualizar partes de la aplicación sin afectar al resto.
2. Virtual DOM: React utiliza un DOM virtual que optimiza el rendimiento. En lugar de actualizar el DOM real directamente, React genera una copia virtual y solo actualiza las partes que han cambiado, lo que mejora la velocidad de renderizado y la eficiencia.
3. Comunidad y ecosistema: React tiene una gran comunidad activa y un extenso ecosistema de bibliotecas y herramientas que facilitan a la integración de nuevas funcionalidades. Además, su amplia adopción hace que sea fácil encontrar recursos, documentación y soporte por casi cualquier tipo de proyecto.

Explica cómo el Virtual DOM mejora el rendimiento de una aplicación

El virtual DOM mejora el rendimiento de una aplicación al minimizar la cantidad de manipulaciones directas en el DOM real, ya que, en el DOM virtual es la copia donde se actualizan las partes cambiadas y eso hace más eficiente la aplicación.

3.Ejercicio:

Define qué es una Single Page Application (SPA)

Una SPA es un tipo de aplicación web que, a diferencia de los sitios web tradicionales que cargan una nueva página completa cada vez que se hace clic en un enlace, carga toda su interfaz de usuario en una sola página HTML inicial. A partir de ahí, la aplicación actualiza dinámicamente el contenido de esa página en respuesta a las acciones del usuario, sin necesidad de recargar la página completa.

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.

React facilita la creación de Single Page Application (SPA) al proporcionar un enfoque basado en componentes, lo que permite construir interfaces de usuarios dinámicas y actualizables sin necesidad de recargar la página.

Ejemplo

```

src > JS ejerciciojs > ...
1 // Importamos React y el hook useState desde la librería de React
2 import React, { useState } from 'react';
3
4 function Contador() {
5   // Definimos el estado inicial del contador en 0
6   const [contador, setContador] = useState(0);
7
8   // Función que incrementa el valor del contador
9   const incrementar = () => {
10     setContador(contador + 1);
11   };
12   // El método return define lo que se mostrará en la interfaz.
13   return (
14     <div>
15       <h1>Contador: {contador}</h1>
16       <button onClick={incrementar}>Incrementar</button>
17     </div>
18   );
19 }
20
21 export default Contador;
22

```

4.Ejercicio:

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

5.Ejercicio:

Explica brevemente el propósito de las carpetas src y public en un proyecto React

La carpeta src contiene el código fuente de la aplicación, es para el desarrollo de la aplicación y public contiene archivos estáticos que no se procesan por el empaquetador, o sea, es para los recursos estáticos accesibles directamente por el navegador.

6.Ejercicio:

Explica cómo JSX se diferencia del HTML tradicional

Se diferencia en que, JSX no es HTML puro, es una extensión de la sintaxis de JavaScript que nos permite escribir estructuras de UI de una manera más intuitiva

y similar al HTML, otra diferencia es que JSX no es entendido directamente por los navegadores antes de ejecutarse, ya que, el código debe ser compilado a JavaScript plano.

7.Ejercicio:

Define los roles principales en un equipo SCRUM

Product Owner: Es quien define y prioriza las tareas del proyecto, asegurando que el equipo desarrolle lo mejor para el cliente.

Scrum Master: Es quien facilita el proceso SCRUM, elimina impedimentos y ayuda al equipo a trabajar de manera eficiente.

Development Team: Son quienes desarrollan el producto, entregan incrementos en cada Sprint y se autoorganizan para cumplir los objetivos.

Explica qué es un sprint y cómo se planifica

Sprint es un ciclo de trabajo en SCRUM de 1 a 4 semanas para completar tareas específicas. Se planifica definiendo un objetivo, seleccionando y desglosando tareas, estimando el esfuerzo y comprometiéndose a completarlas.