

## Taller React

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

R/

- React es una de las bibliotecas más usadas el día de hoy para desarrollar aplicaciones del lado del cliente, es decir aplicaciones web. Esta biblioteca combina JavaScript con HTML a través de JSX, lo cual la hace fácil e interactiva de usar, permitiendo así crear componentes reutilizables, eficientes y más escalables.
- En el 2015, Facebook lanzó React Native, una versión de React para desarrollar aplicaciones móviles nativas usando JavaScript. Esto permitió a los desarrolladores utilizar la misma base de conocimientos para crear aplicaciones tanto web como móviles.

### 2. Escribe una breve explicación sobre por qué Facebook decidió crear React.

R/ Facebook decidió crear React debido a los grandes desafíos que enfrentaba en cuanto al rendimiento y escalabilidad de su aplicación web. A medida que la plataforma crecía y millones de usuarios interactuaban en tiempo real, se volvió necesario hacer la interfaz de usuario más eficiente para manejar actualizaciones rápidas sin comprometer la experiencia de los usuarios. El renderizado tradicional de páginas completas hacía que la aplicación fuera lenta y menos escalable. Por ello, Facebook decidió mejorar y optimizar los procesos de actualización de la interfaz, desarrollando una solución que permitiera actualizar solo el contenido necesario de manera más eficiente.

### 3. Menciona tres ventajas de usar React en el desarrollo de aplicaciones web.

R/

- **Componentes reutilizables:** React permite dividir la interfaz de usuario en componentes independientes y reutilizables, lo que facilita el mantenimiento y la escalabilidad de la aplicación. Estos componentes se pueden reutilizar en diferentes partes de la aplicación, lo que ayuda a los desarrolladores a reducir el tiempo de desarrollo.
- **Facilidad de integración:** React se integra fácilmente con otros frameworks o bibliotecas, como Next.js, Angular o Node.js, lo que lo convierte en una herramienta versátil para diversos entornos de desarrollo.
- **JSX:** React implementa JSX, lo que permite escribir código HTML dentro de JavaScript de una manera más legible. Esto facilita la creación y el mantenimiento de interfaces de usuario de manera más eficiente.

### 4. Explica cómo el Virtual DOM mejora el rendimiento de una aplicación.

R/ El DOM Real contiene toda la estructura y datos de una aplicación web. El Virtual DOM crea una copia de ese DOM Real para poder realizar actualizaciones de manera eficiente. Primero, se hace una comparación entre el DOM real y el Virtual DOM para identificar los cambios o actualizaciones. Luego, en lugar de redibujar toda la página desde cero, React solo aplica los cambios necesarios en el DOM real.

Esto mejora el rendimiento porque las actualizaciones directas en el DOM real son lentas. Al usar el Virtual DOM, React reduce la cantidad de modificaciones, enfocándose solo en las partes que realmente lo necesitan, lo que hace que la aplicación sea más rápida y eficiente.

## **5. Define qué es una Single Page Application (SPA).**

**R/** Una Single Page Application (SPA) es una aplicación web en la que no es necesario recargar toda la página cada vez que el usuario interactúa con ella. En lugar de eso, la SPA actualiza y muestra el contenido dinámicamente según las acciones del usuario.

Esto mejora la experiencia del usuario porque hace que la navegación sea más rápida y fluida, ya que no se requiere volver a cargar toda la página cada vez que se realiza una acción.

## **6. Explica cómo React facilita la creación de una SPA.**

**R/** React facilita la creación de una SPA (Single Page Aplicación) porque se basa en componentes reutilizables. Estos componentes pueden recibir datos a través de props (propiedades) y manejar su propio estado interno.

Cuando el usuario interactúa con la aplicación, los componentes actualizan su estado y, en consecuencia, solo las partes necesarias de la interfaz se actualizan sin recargar la página completa. Los componentes pueden pasar datos a otros componentes mediante props, lo que permite una comunicación fluida entre diferentes partes de la aplicación.

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

**R/** src: Esta es la carpeta principal donde se desarrolla la aplicación. Aquí se crean los componentes de React, se definen las rutas, se implementa la lógica de negocio y se gestionan los estilos CSS. Es el lugar donde se escribe el código que dará vida a la aplicación.

public: Esta carpeta contiene archivos estáticos que se sirven directamente al navegador. El archivo index.html es la página principal de la aplicación, y aquí también se pueden agregar carpetas para imágenes, fuentes, iconos y logotipos que se necesiten para el diseño y la apariencia de la aplicación.

## **8. Explica cómo JSX se diferencia del HTML tradicional.**

**R/** La diferencia es que JSX combina JavaScript con HTML.

JSX permite usar lógica, funciones y variables dentro de la estructura HTML. Esto significa que se puede integrar directamente lógica en la estructura de la interfaz. Mientras que HTML solo define la estructura estática de la página web. Para agregar interactividad o lógica, se tendrían que crear archivos JavaScript separados y luego manipular el DOM para añadir funcionalidades.

## **9. Define los roles principales en un equipo SCRUM. Explica qué es un sprint y cómo se planifica.**

R/

Roles de SCRUM:

- **Product Owner:** Es la persona que decide qué se va a hacer en el proyecto y en qué orden. Se asegura de que el equipo esté trabajando en las tareas más importantes para el negocio o para el cliente.
- **Scrum Máster:** Es como el líder del equipo, se encarga de que todo el equipo siga las reglas de Scrum. Ayuda a eliminar cualquier problema que impida que el equipo avance y asegura que todos puedan concentrarse en su trabajo.
- **Equipo de Desarrollo:** Es el grupo de personas que se encarga de construir el producto. Son los responsables de entregar una parte funcional del proyecto al final de cada sprint.

Un sprint es un período corto de trabajo (normalmente entre 1 y 4 semanas) en el que el equipo se enfoca en completar tareas específicas para crear una parte funcional del proyecto. Al final del sprint, el equipo debería tener algo que esté listo para ser mostrado o usado.

### ¿Cómo se planifica?

**Reunión de planificación del sprint:** El equipo se reúne para decidir qué tareas del Product Backlog se van a trabajar durante el sprint. El Product Owner muestra las tareas más importantes, y el equipo elige las que pueden terminar durante ese sprint.

**Definir la meta del sprint:** Basado en la complejidad de las tareas, se define la meta del sprint, que es el objetivo que se espera cumplir al final del período. Esto incluye entregar funcionalidades que estén listas y que funcionen.

**Dividir y asignar tareas:** Las tareas seleccionadas se dividen en subtareas más pequeñas. Luego, cada miembro del equipo recibe tareas específicas. Para mantener todo organizado, las tareas se detallan en un tablero (como Trello o Jira), donde se establecen tiempos de entrega y se puede hacer seguimiento al progreso.