

Lisney Alexandra Sánchez Vega

## TALLER DE REACT:

1. *Ejercicio:*
2. *Investiga la historia de React y menciona dos hitos importantes en su desarrollo.*

### **HISTORIA:**

Fue lanzado en marzo de 2013 y creado por Jordan Walke Ingeniero de Software en Facebook, la primera implementación fue en el timeline de Facebook, buscaban una manera óptima y rápida de cargar los contenidos en su red social.

Al siguiente año paso a ser implementado en Instagram y fue liberado como código abierto para que los desarrolladores Frontend, algunas de las aplicaciones que usan React JS son Netflix, Airbnb, Pinterest, Instagram, Twitter entre otras.

Empezaron a mejorar poco a poco la librería hasta la actualidad la última versión estable de React es **React 19**. Esta versión ha sido lanzada el 25 de abril de 2024 y trae consigo una serie de mejoras y nuevas características que mejoran el rendimiento, la accesibilidad y la facilidad de uso del framework.

### **HITOS:**

- **2015:** React es estable. Flipboard lanza React Canvas, a Netflix le gusta React y Airbnb utiliza React. Además de que Facebook lanzó la primera versión de React Native para React.js Conf 2015 durante una charla técnica y presentan GraphQL en React.js Conf. También se anunció la disponibilidad de React Native para iOS y Android como código abierto y disponible en GitHub. Redux fue lanzado por Dan Abramov y Andrew Clark.

- **2019:** React 16.8 es lanzada en febrero del mismo año, introduce los Hooks, que permite manejar y usar los estados y otros componentes de React sin necesidad de clases.

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

Facebook creó React debido a su gran cantidad de usuarios interactuando constantemente comentando, subiendo contenido, compartiendo, etc. Todo al mismo tiempo hacía que la interfaz se volviera lenta y se bloquearía.

**2. Ejercicio:**

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

1. **DOM Virtual:** La principal ventaja de React es poder generar el DOM de manera dinámica, permitiendo visualizar los cambios sin necesidad de renderizar toda la página de nuevo, sino solamente el componente que se haya actualizado. Mejorando mucho la navegación para el usuario, la rapidez en la carga de las páginas y la facilidad de mantenimiento.
2. **Componentes reutilizables:** React permite crear componentes reutilizables. Esto permite reutilizar componentes en diferentes partes de la aplicación, facilitando que la aplicación sea escalable y fácil de mantener.
3. **Flexibilidad:** Es muy flexible y se puede integrarse con otras bibliotecas o frameworks, consiguiendo mayor funcionalidad y facilitando el desarrollo. Un gran ejemplo es integración con Redux haciendo una combinación

poderosa para manejar el estado de aplicaciones complejas de manera eficiente y predecible.

*5. Explica cómo el Virtual DOM mejora el rendimiento de una aplicación.* [OBJ]

Cuando el estado de la aplicación cambia, React crea un nuevo Virtual DOM y lo compara con la versión anterior. Después de identificar las diferencias, React aplica solo actualiza esas partes específicas del DOM real, haciendo los ajustes necesarios sin recargar toda la página. Esto hace que las actualizaciones sean mucho más rápidas y eficientes.

**3. Ejercicio:**

*6. Define que es una Single Page Application (SPA).*

Es un tipo de aplicación web que carga una única página inicial desde el servidor y luego actualiza dinámicamente su contenido, mediante las interacciones del usuario, sin necesidad de cargar páginas adicionales del servidor.

*7. Explica como React facilita la creación de una SPA.*

React facilita la creación de una SPA desde su configuración inicial usando Create-React-App que genera una estructura básica de un proyecto en React, incluyendo la configuración inicial necesaria para empezar a desarrollar una SPA, continuamente crear componentes reutilizables y combinándolos para formar componentes más complejos.

**5. Ejercicio:**

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

**Carpeta public:** Almacena archivos estáticos como imágenes y el archivo html.

**Carpeta src:** Contiene todo el código fuente de la aplicación, ahí es donde se desarrolla la lógica y la estructura de la misma.

**6. Ejercicio:**

*11. Explica cómo JSX se diferencia del HTML tradicional.*

Se diferencia en que JSX permite escribir HTML en JavaScript, incluir expresiones JavaScript directamente dentro del marcado con llaves { } y necesita compilarse a JavaScript puro antes de que el navegador lo interprete.

**7. Ejercicio:**

*12. Define los roles principales en un equipo SCRUM.*

- Product Owner: Responsable de definir la visión del producto y gestionar el Product Backlog.
- Scrum Master: 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.

*13. Explica qué es un sprint y cómo se planifica.*

Es un ciclo de desarrollo de duracion fija generalmente de 2-4 semanas, durante este tiempo el equipo trabaja para completar una cantidad determinada de trabajo del Product Backlog.

La planificación de un Sprint se realiza en una reunión llamada “Sprint Planning” donde se decide que tareas se van a abordar durante el sprint, el equipo planifica como abordará estas tareas día a día mediante las “Daily Stand-ups”. Luego de finalizar el Sprint se realiza la reunión “Sprint Review” para demostrar lo que se ha completado y recibir retroalimentación y finalmente la reunión “Sprint Retrospective” en donde se reflexiona sobre lo que salió bien, mal y como mejorar en próximos sprints.

## **BIBLIOGRAFIA**

[Que es React JS, Historia y otros detalles \(nubecolectiva.com\)](#)

[La era de React \(itdo.com\)](#)

[7 razones para utilizar React | tiThink Technology](#)

[Virtual DOM, el concepto detrás de RIA \[2024\] \(keepcoding.io\)](#)

[Qué es una single page application, cómo funciona y ejemplo \(hubspot.es\)](#)

[Desarrolla una aplicación SPA con React: pasos iniciales \(keepcoding.io\)](#)