

Instituto tecnológico de Costa Rica
Sede central
Introducción al desarrollo de Aplicaciones Web

INVESTIGACIÓN REACT

Profesora:
Ericka Solano

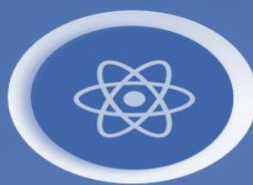
Realizado por:
Jenny Corrales
Josué Guerrero
Alonso Jiménez

TEC | Tecnológico
de Costa Rica

Tabla Contenido

Contenido

Introducción	3
React	4
Funcionalidades	5
Utilización de React	8
Conclusiones	11
Referencias	11

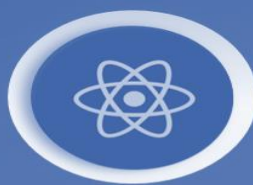


Introducción

El crecimiento y avance en la tecnología ha sido crucial a lo largo del tiempo, en el desarrollo de la vida de las personas. Es por lo anterior que, es de suma importancia cada avance que se logra desde cualquier área tecnológica, consigo mismo las mejoras que se le puedan realizar a este.

Ahora bien, realizando un enfoque desde el área computacional, cada innovación que surja es de primordial para los desarrolladores, porque les permite darle un mejor servicio y experiencia al usuario; de aquí radica la importancia de la capacitación en las distintas herramientas para lograr proyectos exitosos.

En esta investigación se le presentará la herramienta *React* como capa de visualización, sus características, ventajas, desventajas, funcionalidades y la correcta utilización de este para el desarrollo de aplicaciones web.



React

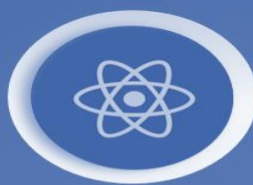
En el 2013 cuando Facebook estaba en auge y comenzaba a presentar cambios en sus componentes para proporcionarle una mejor aplicación al usuario, sucedió que al realizar estos cambios la aplicación se comenzó a tornar un tanto lenta, por lo que se le asignó a un equipo de programadores en donde se encontraba John Walke que mejoraran el rendimiento de esta aplicación. Por tanto, al realizarlo se tuvo como consecuencia el comienzo de lo que conocemos como React.

¿Ahora bien, en sí que es realmente React?

Bueno, React es una librería de JavaScript, para crear interfaces de usuario con el objetivo de facilitar el desarrollo de aplicaciones en una sola página del lado del cliente, por lo que trabaja con componentes, así lo único que se modificaría es el componente.

Características

- Es una librería open source por lo que puede obtener mejoras de su comunidad.
- Facilita mucho el desarrollo porque al tener muchas cosas listas, predeterminadas no debemos invertir tiempo en ello.
- Más ordenada y no necesita la creación de más código que si usáramos JavaScript puro.
- React se encargará de actualizar y renderizar de manera eficiente los componentes correctos cuando los datos cambien.



- HTML y JavaScript van de la mano en un solo paquete.

Funcionalidades

- **Composición de Componentes**

ReactJS puede pasar funciones como parámetros, conocido también como composición funcional. Las aplicaciones se elaboran con la composición de varios componentes y que encapsulan un comportamiento, una vista y un estado. No hay que preocuparse por la complejidad de estos porque el comportamiento reside dentro de cada uno de ellos y al final la aplicación será un conjunto de componentes que trabajan entre sí.

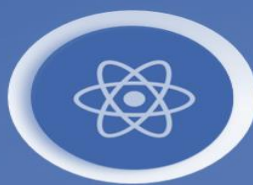
Facilidad de mantenimiento, depuración y escalabilidad son sólo algunas de las ventajas que podemos encontrar en esta forma de trabajar.

- **Desarrollo Declarativo vs Imperativo**

Con otras librerías, como jQuery, o Vanilla Javascript, el tipo de programación que utilizamos es imperativo. Se crean scripts que tienen que informar en el DOM qué debe de realizar o cómo debe de actuar. Esta forma de declarar implica escribir mucho código y a la larga un difícil mantenimiento del mismo.

El que utiliza React es más declarativo, porque contamos con un estado de la aplicación y sus componentes responden ante la variación de ese estado.

Cuando cambiamos las propiedades de los componentes se produce un cambio en su funcionalidad.



- **Flujo de datos unidireccional**

Ese patrón de funcionamiento que posee React hace que los componentes superiores propaguen los datos a los que están en un orden inferior, estos trabajarán con esos datos y cuando cambian su estado envían los eventos hacia los de orden superior para actualizarse.

Aunque el flujo suele ser unidireccional, en ocasiones podemos hacerlo bidireccional por comodidad, pero recuerda que si lo mantenemos unidireccional, el mantenimiento de la aplicación será más sencillo.

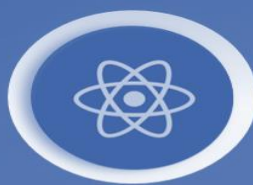
de la aplicación será mucho más sencillo.

- **Rendimiento gracias al virtual DOM**

Las operaciones que realiza React, cuando se ejecutan las aplicaciones, son mucho más rápidas porque se realizan antes sobre el DOM Virtual, que cargado en memoria, permite actualizaciones de hasta 60 frames por segundo, dando como resultado una sensación de fluidez impresionante.

- **Isomorfismo**

También conocido como Javascript Universal, es la capacidad de ejecutar código tanto en el cliente como el servidor, solucionando problemas de posicionamiento tradicionales.



- **JSX**

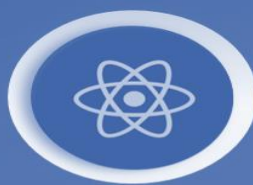
El código que embebemos con ReactJS dentro de Javascript, es JSX, aunque tenga apariencia HTML, es decir, utilizamos una sintaxis propia y el resultado son elementos en memoria y no DOM tradicional. Sin intervención por nuestra parte, los encargados en convertir los elementos devueltos por JSX en DOM Real, son React DOM y la herramienta de diffing.

- **Estado de los componentes**

Los que no tienen estado se llaman stateless y son los componentes que no guardan datos en su memoria, pudiéndose escribir una pequeña función que devuelve el JSX. Los que pueden guardar un estado se les llama statefull. La diferencia respecto a los anteriores es que en su código se escribe por medio de una clase ES6 (Javascript con ECMAScript 2015, como puedes ver en el artículo ¿Qué es ECMAScript 6 (ES6)?) y en la que podemos disponer de atributos y métodos para realizar todo tipo de operaciones.

- **Ciclo de vida de los componentes**

React añade un método que se activa cuando pasan cosas frecuentes en un componente y que permite suscribir acciones cuando se provoca una inicialización, se obtiene la devolución de una promesa, etc.



Tiene 3 clasificaciones:

1. Montaje
2. Actualización
3. Desmontaje

Utilización de React

Agregar React

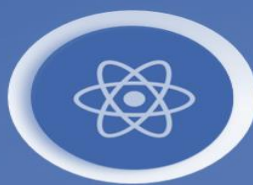
La implementación de react es bastante simple y puede hacerse mediante dos técnicas, dependiendo de cuál es el alcance que se le desea dar a react dentro del proyecto a implementar. Los dos métodos son:

1. Agregar react a una página web.

Los componentes de react pueden ser agregados a una página html individual.

Para utilizar este método no es necesario la instalación de ninguna otra herramienta. Ya que se utilizan tres etiquetas de script las cuales cargan todo lo necesario para utilizar react en la página deseada.

Las primeras dos son enlaces CDN, los cuales cargan react en la página y la tercera es el script donde se crean los componentes que se desean utilizar en la página.



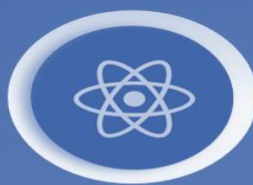
2. Crear una aplicación React

Para obtener una mejor experiencia como usuario y desarrollador se recomienda utilizar una cadena de herramientas integrada. Esta opción es necesaria si se presentan alguno de los siguientes problemas:

- Escalar a múltiples archivos y componentes.
- Usar bibliotecas de terceros desde npm.
- Detección temprana de errores comunes.
- Edición en vivo de CSS y JS en desarrollo.
- Optimización de la salida para producción.

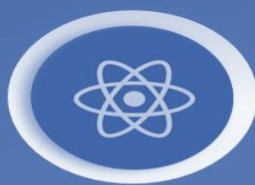
El equipo de React recomienda las siguientes cadenas de herramientas integradas:

- Si estás aprendiendo React o creando una nueva aplicación de página única, usa Create React App.
- Si estás construyendo un sitio web renderizado en servidor con Node.js, prueba Next.js.
- Si estás construyendo un sitio web orientado a contenido estático, prueba Gatsby.
- Si estás construyendo una biblioteca de componentes o integrando una base de código existente, prueba Cadenas de Herramientas más Flexibles.



Características en su utilización

Ya que react está diseñado para que su implementación sea gradual, solo se utiliza lo que realmente se necesita. La idea de React es hacer todos los cambios que se quieran hacer utilizando componentes y luego por medio del DOM se renderizan en la página. Para evitar que este método se vuelva ineficiente y lento, React utiliza un DOM virtual que hace selectivamente sub-árboles de los nodos sobre la base de cambios de estado, desarrollando esto, con la menor cantidad de manipulación DOM posible, con el fin de mantener los componentes actualizados, estructurando sus datos.



Conclusiones

React presenta una opción bastante completa para la creación de aplicaciones de una sola página o móviles. Además, gracias a la utilización de un DOM virtual, los cambios en las páginas se hacen de una manera eficiente y rápida. También gracias al uso de los componentes y su reusabilidad la creación de diferentes páginas puede hacerse bastante fácil.

React es una librería completa, que se puede adecuar a diferentes tipos de proyectos distintos. Nos permite a los desarrolladores un desarrollo ágil, ordenado y con una arquitectura mantenible, focalizada en componentes y que nos ofrece un gran performance.

Referencias

Empezando – React. (2019). Recuperado el 18 Diciembre 2019, de <https://es.reactjs.org/docs/getting-started.html>

García, J. (2019). ¿Cuáles son las principales (y más prácticas) ventajas de ReactJS? - Blog de arsys.es. tomado el 18 Diciembre 2019, de <https://www.arsys.es/blog/programacion/caracteristicas-reactjs/>

W3school.com. (2019). Introduction to react. Recuperado el 18 de diciembre del 2019, de: https://www.w3schools.com/REACT/react_intro.asp

DevCode Tutoriales. (2019). ¿Cómo funciona React.js?. Recuperado el 18 de diciembre del 2019, de: <https://devcode.la/blog/como-funciona-reactjs/>

En.wikipedia.org. (2019). React (web framework). Recuperado el 18 de diciembre del 2019, de: [https://en.wikipedia.org/wiki/React_\(web_framework\)](https://en.wikipedia.org/wiki/React_(web_framework))

