



useEffect 🤖

React hook - useEffect

El efecto y/o **lo que sabe hacer react, es mostrar HTML en nuestro DOM**. Procesando nuestros Componentes escritos en JSX y mostrando el resultado en HTML.



```
const elemento = <h1>Hola, mundo!</h1>;  
ReactDOM.render(element, document.getElementById('root'));
```

Tenemos un problema.....

Una aplicación web hace mucho más que mostrar HTML:

- Llamadas HTTP
- Interacción con APIs de los navegadores
- Y mucho más

¿Cómo hacemos para que React haga más que mostrar HTML?

useEffect

En react tenemos **dos hooks fundamentales**:

- useState
- useEffect

¿Por qué se llama **useEffect**?

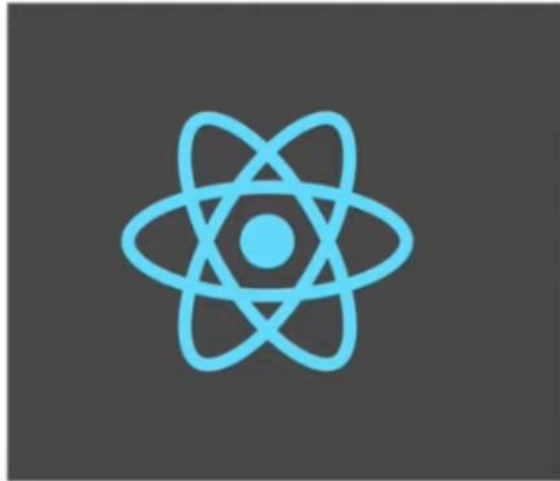
Vimos que el **efecto primario** o lo que sabe hacer react es mostrar HTML en el DOM.

Se llama useEffect porque estamos haciendo un **efecto secundario**. Queremos que react haga algo distinto de mostrar HTML.

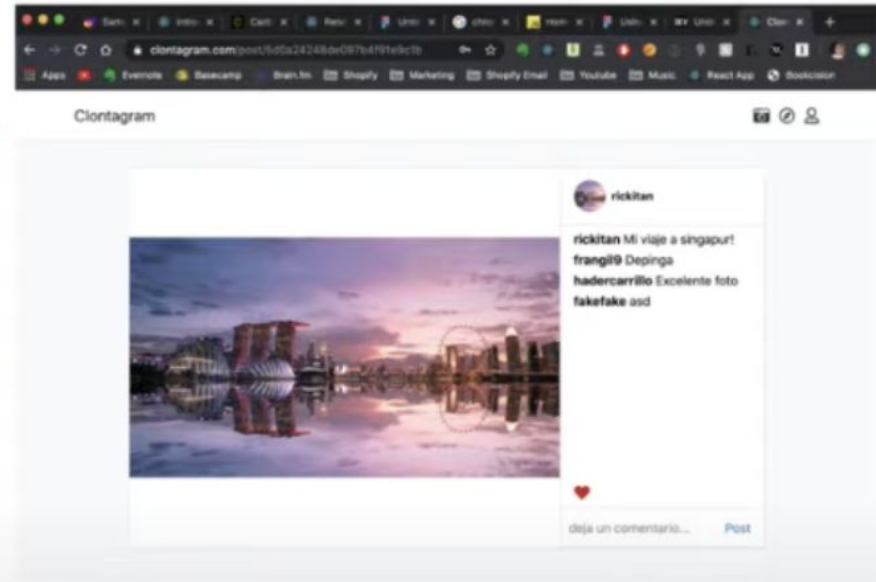
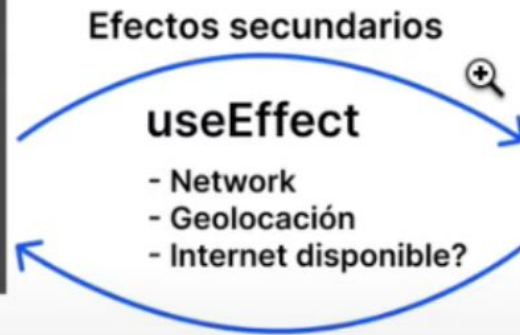
Se podría decir que **useEffect es un puente** entre React y el mundo exterior y el mundo exterior y React.

React hook - useEffect

App



Efecto primario (render)



Ciclo de vida del componente 🤖

React hook - Ciclo de vida del componente

El ciclo de vida se puede dividir en 3 fases, el montado, actualización y desmontado del componente.

- **Montado:** La primera fase ocurre solo una vez por componente cuando este se crea y monta en la UI.
- **Actualización:** Esa fase puede ocurrir múltiples veces (o incluso ninguna), sucede cuando algún dato del componente (ya sea una propiedad, un estado o el contexto) se modifica y por lo tanto requiere que la UI se vuelva a generar para representar ese cambio de datos.
- **Desmontado:** Esta última fase consiste en cuando el componente se elimina del DOM.

LINKS



LINKS

- **React**
 - <https://es.reactjs.org/docs/hooks-effect.html>



[gustavgueez](#)



[gustavgueez](#)



[gustavgueez](#)

GUSTAVO RODRIGUEZ

FULL STACK DEVELOPER
SOLCRE