





13 - Ciclo de vida

2022 © Wawiwa Tech | Confidential



Introducción

Los métodos del ciclo de vida son funciones personalizadas que corren durante varias fases de un componente. Las fases de un componente de React son:

1. **Renderización o montaje inicial** - insertar elementos en el DOM
2. **Actualización** - actualización de los componentes en el DOM (Cuando los estados usados en los componentes o los props agregados a los componentes cambian)
3. **Desmontaje** - quitar un componente del DOM

Cada fase tiene sus propios métodos que facilita el rendimiento o operaciones típicas en los componentes.



1. Renderización o montaje inicial

En esta variante de `useEffect`, el mensaje va a ser impreso solamente cuando el componente es **montado** y cada vez que el **estado del componente o sus props sean actualizados**.

```
useEffect(() => {  
  console.log("This is mounted or updated.");  
});
```




2. Actualización

En esta variante, el mensaje va a ser impreso **solamente una vez** en el ciclo de vida del componente, después de que el componente haya sido **montado**:

```
useEffect(() => {  
  console.log("This is mounted only, not updated.");  
}, []);
```

En esta variante, el mensaje va a ser impreso cuando el componente es **montado y cada vez que el estado “counter” sea cambiado**:

```
useEffect(() => {  
  console.log("This is mounted or counter state updated.");  
}, [counter]);
```



3. Desmontaje

Si queremos hacer un cambio cuando el componente sea desmontado (removido), podemos usar la variante siguiente.

```
useEffect(() => {  
  return () => {  
    console.log("This is unmounted.");  
  };  
}, []);
```



Controlar y prevenir re-renderizaciones

Es posible controlar la re-renderización de un componente y una manera es usando **useMemo**.

useMemo - un hook React que retorna un valor memorizado.

React.memo usa memorización. Compara **superficialmente** los props **previos** con los props **actuales** para determinar si los props han sido cambiados. Si ese es el caso, el componente se re-renderiza.

También podemos proveer un comparador personalizado para usar useMemo para **personalizar la comparación**.