



Bienvenido Desarrollador React



@LeonidasEsteban
leo@platzi.com

**A JavaScript library for building user
interfaces**



Estado del arte del Frontend

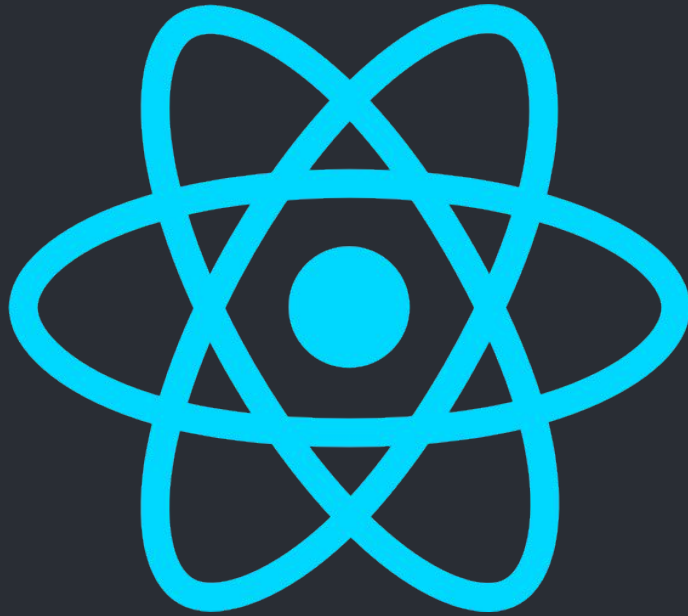






BACKBONE.JS





Declarativo

```
2  
3 <MyWebsite>  
4   . . <Header>  
5     . . . . <Menu . />  
6   . . </Header>  
7   . . <Main>  
8     . . . . <Productos . />  
9   . . </Main>  
10  . . <Footer . />  
11 </MyWebsite>
```

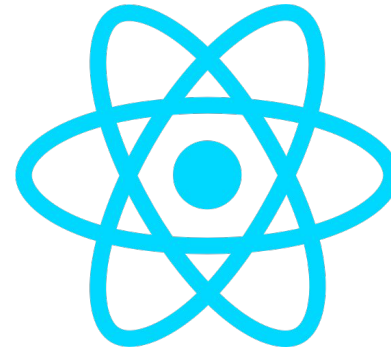

Basado en Componentes

Crear aplicaciones con React es como jugar con legos



Aprende una vez, escribe donde quieras





facebook

Facebook está construido en React.js



¡Empecemos!



Ciclo de vida de un componente



MONTADO



constructor

Método llamado antes de que el componente sea montado (el componente aún no se ve)

1. Podemos iniciar el estado
2. Enlazar (bind) de eventos
3. Es el primer metodo que se llama al instanciar un componente

componentWillMount

Método llamado
inmediatamente antes de que
el componente se vaya a
montar
(el componente aún no se ve)

1. Puedes hacer un `setState()`
2. No hagas llamados a un API o suscripción a eventos

render

Contiene todos los elementos
a renderizar
(estructura del componente)

1. Contiene jsx en el return
2. Puedes calcular
propiedades nCompleto =
name + lastName

componentDidMount

Método llamado luego de montarse el componente (el componente ya está en pantalla)

1. Solo se lanza una vez
2. Enlazar (bind) de eventos
3. Es el primer método que se llama al instanciar un componente

ACTUALIZACIÓN



componentWillReceiveProps

- Método llamado al recibir nuevas propiedades
- Sirve para actualizar el estado con base a las nuevas propiedades



shouldComponentUpdate

- Método que condiciona si el componente se debe volver a renderizar
- Utilizado para optimizar el rendimiento



componentWillUpdate

- Método llamado antes de re-renderizar un componente
- Utilizado para optimizar el rendimiento



render

- re-render



componentDidUpdate

- Método llamado luego del re-render



DESMONTADO



componentWillUnmount

- Método llamado antes de que el componente sea retirado de la aplicación



MANEJO DE ERRORES



componentDidCatch

- Si ocurre algún error al renderizar el componente este método es invocado
- El manejo de errores solo ocurre en componentes hijos



Smart & Dumb Components

...



@dan_abramov

Cómo se ve **VS** Qué hace



Formas de llamarlos

- Smart
- Statefull
- Fat
- Container
- Dumb
- Pure
- Skinny
- Presentational
Component

Presentational

Cómo se ve

1. Puede contener smart components u otros componentes de UI
2. Permiten compocisión con `{props.children}`
3. No dependen del resto de la aplicación
4. No especifica cómo los datos con cargados o mutado
5. Recibe datos y callbacks solo con propiedades
6. Rara vez tienen su propio estado
7. Están escritos como componentes funcionales a menos que necesiten mejoras de performance

Containers

Qué hace

1. Concentrado en el funcionamiento de la aplicación
2. Contienen componentes de UI u otros containers
3. No tienen estilos
4. Proveen de datos a componentes de UI u otros contenedores
5. Proveen de callbacks a la UI
6. Normalmente tienen estado
7. *Llaman acciones*
8. *Generados por higher order components*

¿Por qué?



- Separación de Responsabilidades
- Mejora la capacidad de reutilizar componentes

