

React.JS

Introducción

Módulo 01

Pensando en React.JS

Pensando en React.JS

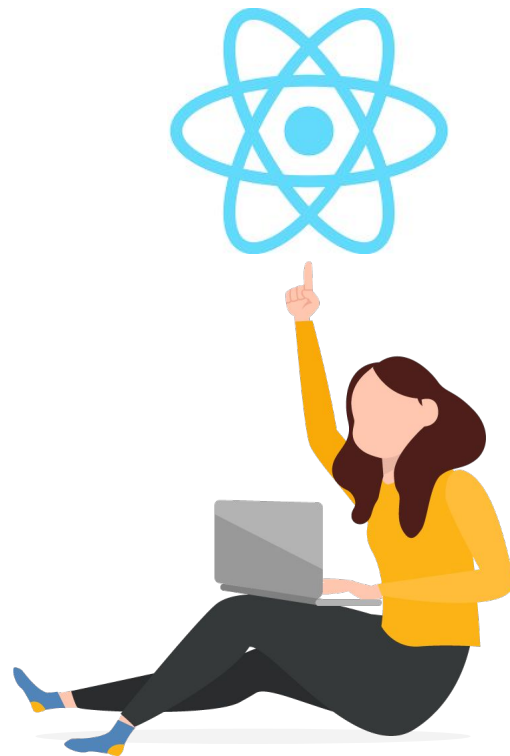
Una de las grandes ventajas de React es cómo te hace pensar acerca de la aplicación mientras la construyes.

Este proceso se puede resumir en tres pasos:

1. Pensar en Componentes.
2. Decide dónde debe vivir la información de la vista (estado).
3. Decide qué cambia cuando dicha información cambia.

Fuente:

<https://es.reactjs.org/docs/thinking-in-react.html>



1. Pensar en componentes

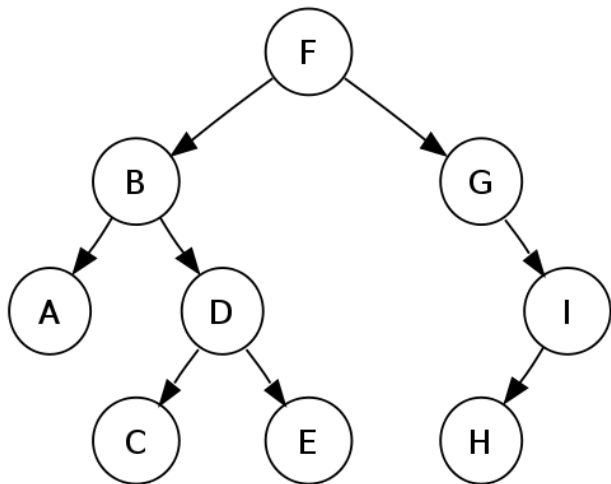
- Podemos empezar con un mock o un diseño estático de la vista que deseamos construir.
- Luego, será importante dividir esa interfaz en pequeñas cajitas y darle nombre. Esas cajitas serán tus componentes.
- ¿Cómo decidir qué es un componente? Puedes usar el Principio de Responsabilidad Única: *"Si tiene más de una función, se debe dividir."*
- Recuerda que los Componentes muestran información, así que deben estar organizados según la información a mostrar.

Name	Price
Sporting Goods	
Football	\$49.99
Baseball	\$9.99
Basketball	\$29.99
Electronics	
iPod Touch	\$99.99
iPhone 5	\$399.99
Nexus 7	\$199.99

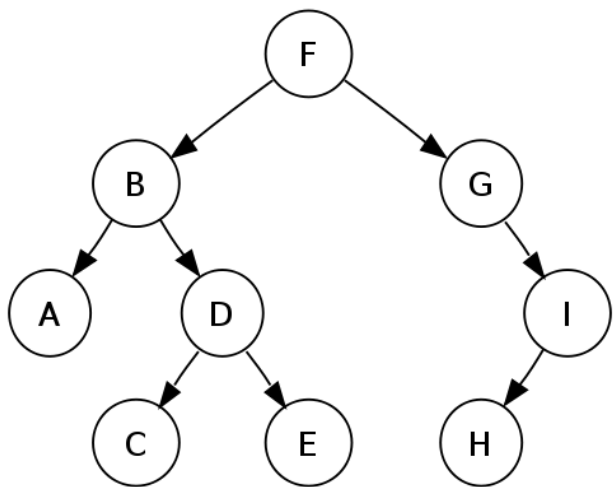
- Una vez que hemos decidido los distintos componentes, el siguiente paso es crear una primera versión estática de dicha aplicación.
- Una versión estática es una versión de prueba en la que no hay interactividad ni conexión entre los componentes.
- Es importante separar estos dos procesos para hacer el desarrollo más mantenible.
- En resumen: luego de desarrollar un mock, desarrollamos una primera versión **sin interactividad**.

Name	Price
Sporting Goods	
Football	\$49.99
Baseball	\$9.99
Basketball	\$29.99
Electronics	
iPod Touch	\$99.99
iPhone 5	\$399.99
Nexus 7	\$199.99

2. Decide dónde debe vivir el estado



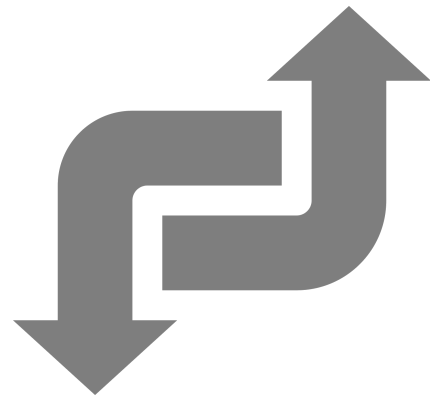
- Para hacer tu interfaz de usuario interactiva vas a necesitar realizar cambios en tu modelo de datos interno. React lo logra gracias a su estado.
- Para descubrir la mínima versión del estado, sería importante recordar el DRY: *Don't Repeat Yourself*. Es decir, no repetir información.
- Para decidir si un dato es parte del estado puedes aplicar los siguientes criterios:
 - No debe venir como atributo del componente.
 - Debe cambiar con el tiempo.
 - No debe ser calculable en base a otro dato.



- Recuerda que, en ReactJS, la información fluye *en un solo: **del componente padre a los componentes hijos***. Tal como se muestra en el gráfico.
- Es importante decidir **dónde** debe vivir esa información.
- Identifica qué componentes muestran algo con base a este estado.
- Busca un componente común a estos más arriba en la jerarquía.
- Este componente o uno más arriba en la jerarquía debería poseer el estado.
- Si no puedes crear un nuevo componente que tenga sentido que posea el estado, crea un nuevo componente simplemente para poseer el estado y agrégalo en la jerarquía sobre los componentes que lo necesitan.

3. Decide qué cambia cuando cambia el estado

- Una aplicación interactiva en ReactJs cambia su comportamiento cuando el estado cambia.
- Decide qué componentes cambian cuando cambia el estado.
- Puedes hacerlo mediante escuchadores de eventos.
- Hay librerías como *Redux* que te permiten manejar el estado de forma centralizada.



¡Muchas gracias!

¡Sigamos trabajando!