



# Desarrolla una SPA básica usando React

Jefferson A. Peña Torres

*jefferson.amado.pena@correounivalle.edu.co*

## React

- Create React App ✓
- JSX ✓
- Custom Components ✓
- Props →
- Wrapper →

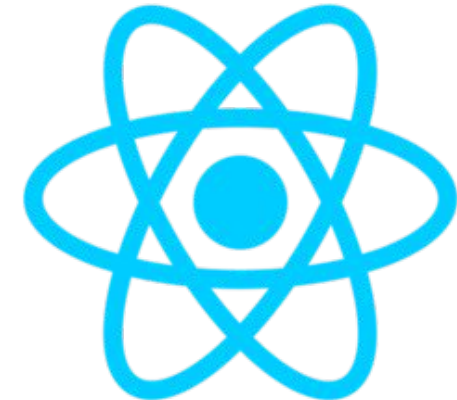
Desarrolla una SPA básica usando React

## Repasemos React: ¿Que és?

React es una biblioteca o **framework** de Javascript que se utiliza para crear **interfaces de usuario**, en especial aplicaciones de una **sola página** (SPA). Permite crear aplicaciones web que cambian los datos, sin recargar la página. React se caracteriza por ser **rápido, escalable y sencillo**.

Antes de empezar es necesario que recuerdes y conozcas algunas conceptos relacionados con:

- Variables y cómo declararlas
- POO: Programación Orientada a Objetos
- funciones flecha (arrow)
- Callbacks
- Promises
- Async/ Await
- Módulos ES



1 Create a App (SPA)

2 JSX

3 Custom Componentes

4 Props & Wrappers



El futuro digital  
es de todos

MinTIC



Frameworks JS: Características de los proyectos SPA

## ¿Qué es create-react-app?

1

React dispone de algunas formas de acelerar el proceso de configuración y creación de las aplicaciones. Una de esas formas es **create-react-app** que realiza la configuración e instalación de los paquetes necesarios para iniciar el desarrollo de una aplicación SPA.

2

3

4



```
npx create-react-app mtic-app
```



El futuro digital  
es de todos

MinTIC



Frameworks JS: Características de los proyectos SPA

## ¿Qué es create-react-app?

1

React dispone de algunas formas de acelerar el proceso de configuración y creación de las aplicaciones. Una de esas formas es **create-react-app** que realiza la configuración e instalación de los paquetes necesarios para iniciar el desarrollo de una aplicación SPA.

2

3

4

```
npx create-react-app mtic-app
```

Si no está instalado npx puede instalarlo así:

```
npm install -g npx
```



El futuro digital  
es de todos

MInTIC



Frameworks JS: Características de los proyectos SPA

## ¿Qué es create-react-app?

1

React dispone de algunas formas de acelerar el proceso de configuración y creación de las aplicaciones. Una de esas formas es **create-react-app** que realiza la configuración e instalación de los paquetes necesarios para iniciar el desarrollo de una aplicación SPA.

2

3

4

```
npx create-react-app mtic-app
```

Puedes reemplazar **mtic-app** con el nombre que desees darle a tu aplicación



El futuro digital  
es de todos

MinTIC



Frameworks JS: Características de los proyectos SPA

## ¿Qué es create-react-app?

1

React dispone de algunas formas de acelerar el proceso de configuración y creación de las aplicaciones. Una de esas formas es **create-react-app** que realiza la configuración e instalación de los paquetes necesarios para iniciar el desarrollo de una aplicación SPA.

2

3

4

```
npx create-react-app mtic-app
```

Puede tomar un par de minutos....  
Una vez haya terminado ya tendremos todo lo necesario.

Frameworks JS: Características de los proyectos SPA

## ¿Qué es create-react-app?

1

React dispone de algunas formas de acelerar el proceso de configuración y creación de las aplicaciones. Una de esas formas es **create-react-app** que realiza la configuración e instalación de los paquetes necesarios para iniciar el desarrollo de una aplicación SPA.

2

3

4

Una vez termine la creación de la carpeta y las dependencia podemos ejecutar y ver en nuestro computador la aplicación con una plantilla vacía.

```
npx create-react-app mtic-app
```

Frameworks JS: Características de los proyectos SPA

## ¿Qué es create-react-app?

1

React dispone de algunas formas de acelerar el proceso de configuración y creación de las aplicaciones. Una de esas formas es **create-react-app** que realiza la configuración e instalación de los paquetes necesarios para iniciar el desarrollo de una aplicación SPA.

2

3

4

Una vez termine la creación de la carpeta y las dependencia podemos ejecutar y ver en nuestro computador la aplicación con una plantilla vacía.

```
npx create-react-app mtic-app
```

```
cd mtic-app
```



Frameworks JS: Características de los proyectos SPA

## ¿Qué es create-react-app?

1

React dispone de algunas formas de acelerar el proceso de configuración y creación de las aplicaciones. Una de esas formas es **create-react-app** que realiza la configuración e instalación de los paquetes necesarios para iniciar el desarrollo de una aplicación SPA.

2

3

4

Una vez termine la creación de la carpeta y las dependencia podemos ejecutar y ver en nuestro computador la aplicación con una plantilla vacía.

```
npx create-react-app mtic-app
```

```
cd mtic-app
```

```
npm start
```



Frameworks JS: Características de los proyectos SPA

## ¿Qué es create-react-app?

1

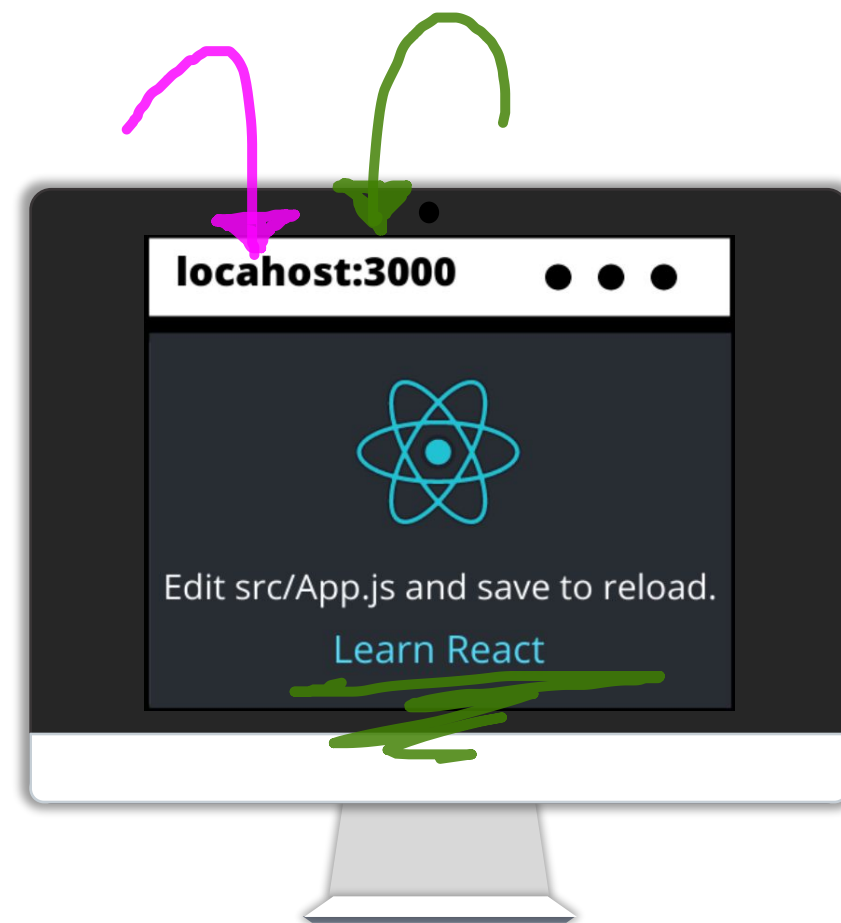
React dispone de algunas formas de acelerar el proceso de configuración y creación de las aplicaciones. Una de esas formas es **create-react-app** que realiza la configuración e instalación de los paquetes necesarios para iniciar el desarrollo de una aplicación SPA.

2

3

4

Una vez termine la creación de la carpeta y las dependencia podemos ejecutar y ver en nuestro computador la aplicación con una plantilla vacía.





Frameworks JS: Características de los proyectos SPA

## ¿Qué es create-react-app?

1

React dispone de algunas formas de acelerar el proceso de configuración y creación de las aplicaciones. Una de esas formas es **create-react-app** que realiza la configuración e instalación de los paquetes necesarios para iniciar el desarrollo de una aplicación SPA.

2

3

4

Una vez termine la creación de la carpeta y las dependencia podemos ejecutar y ver en nuestro computador la aplicación con una plantilla vacía.



\*\* Investigar cómo operan los servicios DNS y cómo estos convierten una URL a dirección IP con puerto



Frameworks JS: Características de los proyectos SPA

## Sintaxis JSX



React también soporta JSX una extensión creada por Facebook que agrega XML a Javascript. Esta extensión preprocesa todo que se escriba en XML y aunque puedes utilizar React sin JSX, es mucho más elegante.



```
<?xml version="1.0"
      encoding="iso-8859-1" ?>
<languages>
  <language id="fr">
    <name lang="fr">Français</name>
    <name lang="en">French</name>
    <name lang="es">Frances</name>
    <name lang="de">Französisch</name>
    <name lang="eo">Franca</name>
  </language>
```



## Sintaxis JSX

1

React también soporta JSX una extensión creada por Facebook que agrega XML a Javascript. Esta extensión preprocesa todo que se escriba en XML y aunque puedes utilizar React sin JSX, es mucho más elegante.

2

3

En React en lugar de usar Javascript para la creación de plantillas se utiliza JSX. Simplemente escribe etiquetas para representar **componentes**.

4



```
<?xml version="1.0"
      encoding="iso-8859-1" ?>
<languages>
  <language id="fr">
    <name lang="fr">Français</name>
    <name lang="en">French</name>
    <name lang="es">Frances</name>
    <name lang="de">Französisch</name>
    <name lang="eo">Franca</name>
  </language>
```

Frameworks JS: Características de los proyectos SPA

## Componentes en React

Revisa los archivos y carpetas que se crearon para tu aplicación.



code mtic-app



El futuro digital  
es de todos

MinTIC



Frameworks JS: Características de los proyectos SPA

## Componentes en React

1

Revisa los archivos y carpetas que se crearon para tu aplicación.

2

3

4

+

`code mtic-app`

Si estas utilizando VSCode.



## Componentes en React

1 Revisa los archivos y carpetas que se crearon para tu aplicación. Abre el archivo llamado App.js.

2 \* Un componente es responsable sólo de la capa de visualización. React se ocupa de las vistas de la aplicación (interfaz). La mayor parte del desarrollo será creación de componentes.

Este es el primer **componente** que se crea con React

```
1  import logo from './logo.svg';
2  import './App.css';
3
4  function App() {
5    return (
6      <div className="App">
7        <header className="App-header">
8          <img src={logo} className="App-logo" alt="logo" />
9          <p>
10             Edit <code>src/App.js</code> and save to reload.
11          </p>
12          <a
13             className="App-link"
14             href="https://reactjs.org"
15             target="_blank"
16             rel="noopener noreferrer"
17           >
18             Learn React|
19          </a>
20        </header>
21      </div>
22    );
23  }
24
25  export default App;
```





## Componentes en React

1

Revisa los archivos y carpetas que se crearon para tu aplicación. Abre el archivo llamado App.js.

2

Un componente es responsable sólo de la capa de visualización. React se ocupa de las vistas de la aplicación (interfaz). La mayor parte del desarrollo será creación de componentes.

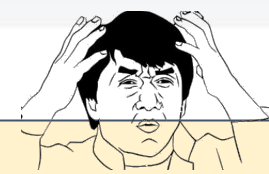
3

4

+

```
1  import logo from './logo.svg';
2  import './App.css';
3
4  function App() {
5    return (
6      <div className="App">
7        <header className="App-header">
8          <img src={logo} className="App-logo" alt="logo" />
9          <p>
10             Edit <code>src/App.js</code> and save to reload.
11          </p>
12          <a
13             className="App-link"
14             href="https://reactjs.org"
15             target="_blank"
16             rel="noopener noreferrer"
17          >
18             Learn React|
19          </a>
20        </header>
21      </div>
22    );
23  }
24
25
```

XML



Parece HTML, se utilizan etiquetas similares/iguales a las de HTML pero no es HTML



## Componentes en React

1

Revisa los archivos y carpetas que se crearon para tu aplicación. Abre el archivo llamado App.js.

2

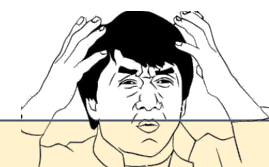
Un componente es responsable sólo de la capa de visualización. React se ocupa de las vistas de la aplicación (interfaz). La mayor parte del desarrollo será creación de componentes.

3

4

+

```
1  import logo from './logo.svg';
2  import './App.css';
3
4  function App() {
5    return (
6      <div className="App">
7        <header className="App-header">
8          <img src={logo} className="App-logo" alt="logo" />
9          <p>
10             Edit <code>src/App.js</code> and save to reload.
11          </p>
12          <a
13             className="App-link"
14             href="https://reactjs.org"
15             target="_blank"
16             rel="noopener noreferrer"
17          >
18             Learn React|
19          </a>
20        </header>
21      </div>
22    );
23
24
25
```



Es JSX, que será procesado e interpretado.



El futuro digital  
es de todos

MinTIC



Frameworks JS: Características de los proyectos SPA

## Componentes personalizados (Custom)

1

Un componente puede tener su propio estado (*state*) y puede recibir propiedades de otros componentes (*props*).

2

3

4



Frameworks JS: Características de los proyectos SPA

## Componentes personalizados (Custom)

1


Un componente puede tener su propio estado (*state*) y puede recibir propiedades de otros componentes (*props*).

2

Los componentes pueden ser considerados funciones simples que toman información (accesorios). Algunas de estas son funciones puras (paradigma funcional)

3

4



```
function Welcome() {  
  return <p>Aprendiendo MTic UPB</p>  
};
```



```
const FirstComponent = props => (  
  {props.content}   
);
```



## Componentes personalizados (Custom)

1



Un componente puede tener su propio estado (*state*) y puede recibir propiedades de otros componentes (*props*).

2

Los componentes pueden ser considerados funciones simples que toman información (accesorios). Algunas de estas son funciones puras (paradigma funcional)

3

4



```
const AComponent = React.createClass({  
  render: function() {  
    return (  
      <p>Aprendiendo MTic UPB</p>  
    )  
  }  
});
```

Frameworks JS: Características de los proyectos SPA

## Componentes personalizados (Custom)

1

Un componente puede tener su propio estado (*state*) y puede recibir propiedades de otros componentes (*props*).


2

Los componentes pueden ser considerados funciones simples que toman información (accesorios). Algunas de estas son funciones puras (paradigma funcional)

3

4

```
const AComponent = React.createClass({  
  render: function() {  
    return (  
      <div>{this.props.content}</div>  
    )  
  }  
});
```





## Componentes personalizados (Custom)

1

Un componente puede tener su propio estado (*state*) y puede recibir propiedades de otros componentes (*props*).

2

Los componentes pueden ser considerados funciones simples que toman información (accesorios). Algunas de estas son funciones puras (paradigma funcional)

3

4

```
const AComponent = React.createClass({  
  render: function() {  
    return (  
      <div>{this.props.content}</div>  
    )  
  }  
});
```

```
const OComponent extends React.createClass({  
  render: function() {  
    return (  
      <div>{this.props.content}</div>  
    )  
  }  
});
```



## Componentes personalizados (Custom)

1

Un componente puede tener su propio estado (*state*) y puede recibir propiedades de otros componentes (*props*).

2

Los componentes pueden ser considerados funciones simples que toman información (accesorios). Algunas de estas son funciones puras (paradigma funcional)

3

4

```
const ParentComponent = function(props) {  
  const name = "Jefferson"  
  return (  
    <Wellcome/>  
    <FirstComponent/>  
    <AComponent content={name}/>  
  )  
}  
};
```





## Propiedades de un componente

Las **propiedades** son valores pasados a componente desde un **componente padre**.

```
const ParentComponent = function(props) {  
  const name = "Jefferson"  
  return(  
    <Wellcome/>  
    <FirstComponent/>  
    <AComponent content={name}/>  
  )  
}  
});
```



## Propiedades de un componente

1

Las propiedades son valores pasados a componente desde un componente padre.

2

3

4

```
function Wellcome() {  
  return <p>Aprendiendo MTic UPB</p>  
};
```

```
const ParentComponent = function(props) {  
  const name = "Jefferson"  
  return(  
    <Wellcome/>  
  )  
}  
});
```

Este componente no  
envía ninguna propiedad



## Propiedades de un componente

1

Las propiedades son valores pasados a componente desde un componente padre.

2

3

4

```
function Wellcome() {  
  return <p>Aprendiendo  
};
```

Este componente no  
recibe ninguna propiedad

```
const ParentComponent = function(props) {  
  const name = "Jefferson"  
  return(  
    <Wellcome/>  
  )  
}  
};
```



Frameworks JS: Características de los proyectos SPA

## Propiedades de un componente

Las propiedades son valores pasados a componente desde un componente padre.

```
function Wellcome({name}) {  
  return <p>Aprendiendo con {name}</p>  
};  
  
const ParentComponent = function() {  
  const name = "Jefferson"  
  return(  
    <Wellcome/>  
  )  
}  
};
```

Este componente  
tiene *props*



## Propiedades de un componente

1

Las propiedades son valores pasados a componente desde un componente padre.

2

3

4

```
function Wellcome({name}) {  
  return <p>Aprendiendo con {name}</p>  
};  
  
const ParentComponent = function(props) {  
  const name = "Jefferson"  
  return(  
    <Wellcome name = {name}/>  
  )  
}  
};
```

Este componente  
envía la propiedad



## Componentes *Wrapper*

Son componentes que envuelven (*wrapper*) otros componentes personalizados y proporcionar una estructura o visualización común.

```
function Panel({children, title}) {  
  return(  
    <div className="panel">  
      <div className="panel-header">  
        <h2>{title}</h2>  
      </div>  
      {children}  
    </div>  
  )  
};
```

Frameworks JS: Características de los proyectos SPA

## Componentes *Wrapper*

1

Son componentes que envuelven (*wrapper*) otros componentes personalizados y proporcionar una estructura o visualización común.

2

3

El componente `Panel` crea un estilo para cualquier componente `Panel` nuevo.

4

```
function Panel({children, title}) {  
  return(  
    <div className="panel">  
      <div className="panel-header">  
        <h2>{title}</h2>  
      </div>  
      {children}  
    </div>  
  )  
};
```

Desarrolla una SPA básica usando React

## Resumen

1 Create a App (SPA)

2 JSX

3 Custom Componentes

4 Props & Wrappers

Una de las formas para realizar la configuración e instalación de los paquetes necesarios para iniciar el desarrollo de una aplicación SPA con React es utilizando create-react-app. Este paquete crea todo lo necesario e incluso la configuración para que ejecutando npm start en la terminal se pueda visualizar una App plantilla.

Con la extensión JSX, React puede crear componentes con etiquetas similares a las de HTML. Los componentes pueden ser considerados funciones simples que toman información (accesorios) y algunas de estas son funciones puras (paradigma funcional).

Las propiedades (props) son los valores que se pasan a los componentes, teniendo en cuenta su relación padre-hijo y los componentes envoltorios (wrappers) sirven para crear un estilo estándar para cualquier componente nuevo.





[www.upb.edu.co/es/mision-tic](http://www.upb.edu.co/es/mision-tic)  
#MisiónTICSomosTodos