

TUTORIAL DE REACT ROUTER DOM VERSION 6

Requisitos

- Una aplicación de *React*
- Un buen entendimiento de qué son los componentes en *React*.
- Tener instalado *Node.js*.

Cuando pulsas un enlace en el navegador, en aplicaciones con más de una página, es enviada una petición al servidor antes de que la página HTML se muestre.

En *React*, el contenido de la página es creado a partir de nuestros componentes. Así que lo que hace *React Router* es interceptar la petición que se envía al servidor y luego inyectar el contenido dinámicamente desde los componentes que hemos creado.

Esta es la idea general detrás de las *SPA*, que permiten que el contenido se muestre más rápido sin que la página sea actualizada.

Cuando creas un nuevo proyecto, siempre verás un archivo `index.html` en la carpeta *public* ('pública'). Todo el código que escribas en tu componente `App`, que actúa como componente raíz, se renderiza en este archivo HTML. Esto significa que sólo hay un archivo HTML en el cual será renderizado tu código.

¿Qué ocurre cuando prefieres renderizar en una página distinta otro componente? ¿Creas un nuevo archivo HTML? La respuesta es no. *React Router* – como su nombre en inglés indica ('enrutador de *React*')– te ayudará a dirigirte a -o navegar- a un componente concreto, y te ayudará a renderizar tu nuevo componente en el archivo `index.html`.

Así que, en una aplicación de página única, cuando se navega a un nuevo componente utilizando *React Router*, el archivo `index.html` será reescrito en función de la lógica del componente.

TUTORIAL DE REACT ROUTER DOM VERSION 6

Cómo instalar React Router

Todo lo que tienes que hacer para instalar *React Router* 6 es ejecutar `npm install react-router-dom@6` en la terminal de tu proyecto y esperar a que se complete la instalación.

Cómo configurar React Router

Lo primero que hay que hacer, una vez completada la instalación, es conseguir que *React Router* esté disponible en cualquier parte de tu aplicación.

Para hacer esto, abre el archivo `index.js` de la carpeta `src` (de *source*, 'fuente' en inglés) e importa (*import*) `BrowserRouter` desde `react-router-dom`, y luego, rodea el componente raíz (el componente `App`) con él.

Este es el aspecto inicial del archivo `index.js`:

```
import React from 'react';
import ReactDOM from 'react-dom';
import App from './App';
```

```
ReactDOM.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>,
  document.getElementById('root')
);
```

Archivo `index.js`

TUTORIAL DE REACT ROUTER DOM VERSION 6

Después de hacer los cambios con *React Router*, deberías tener lo siguiente:

```
import React from 'react';
import ReactDOM from 'react-dom';
import App from './App';
import { BrowserRouter } from "react-router-dom";

ReactDOM.render(
  <BrowserRouter>
    <App />
  </BrowserRouter>,
  document.getElementById("root")
);
```

Todo lo que hemos hecho es reemplazar `React.StrictMode` por `BrowserRouter`; que hemos importado primero de `react-router-dom`. Ahora las características del enrutador son accesibles desde cualquier parte de tu aplicación.

Cómo redirigir hacia otros componentes

Por fin hemos terminado de configurar las cosas, así que ahora veremos el enrutamiento y la renderización de los diferentes componentes.

Paso 1 - Crea varios componentes

Crearemos de la manera siguiente los componentes:

Inicio, SobreNosotros y Contacto:

```
function Inicio() {
  return (
    <div>
      <h1>Esta es la página de inicio</h1>
    </div>
  );
}
export default Inicio;
```

TUTORIAL DE REACT ROUTER DOM VERSION 6

```
import React from 'react'
function SobreNosotros() {
  return (
    <div>
      <h1>Esta es la página sobre nosotros</h1>
    </div>
  )
}
export default SobreNosotros;
```

```
import React from 'react'
function Contacto() {
  return (
    <div>
      <h1>Esta es la página de contacto</h1>
    </div>
  )
}
export default Contacto;
```

Paso 2 - Define las rutas

Dado que el componente `App` actúa como el componente raíz, en dónde se renderiza al principio nuestro código *React*, vamos a crear todas nuestras rutas en él.

```
import { Routes, Route } from "react-router-dom"
import Inicio from "./Inicio"
import SobreNosotros from "./SobreNosotros"
import Contacto from "./Contacto"

function App() {
  return (
    <div className="container">
      <Routes>
```

TUTORIAL DE REACT ROUTER DOM VERSION 6

```
        <Route path="/" element={ <Inicio /> } />
        <Route path="sobre-nosotros" element={
<SobreNosotros /> } />
        <Route path="contacto" element={ <Contacto /> }
        />
    </Routes>
</div>
)
}
export default App;
```

Primero importaremos las funcionalidades que emplearemos: `Routes` y `Route`. A continuación, importaremos todos los componentes a los que necesitamos proporcionar una ruta. Ahora desglosaremos el proceso.

- `Routes` actúa como contenedor padre de todas las rutas individuales que se crearán en nuestra aplicación.
- `Route` se utiliza para crear una única ruta. Recibe dos atributos:
 - `path`, que especifica la ruta *URL* del componente deseado. Puedes llamar a esta ruta de la manera que quieras. Arriba, habrás notado que el primer nombre de ruta es una barra diagonal (/). Cualquier componente cuyo nombre de ruta sea una barra diagonal se mostrará primero, siempre que la aplicación se cargue por primera vez. Esto implica que el componente `Inicio` será el primero en mostrarse.
 - `element`, que especifica el componente que debe renderizar el enrutador.

Todo lo que hemos hecho hasta ahora es definir nuestras rutas, y sus direcciones, y ligarlas a sus respectivos componentes.

TUTORIAL DE REACT ROUTER DOM VERSION 6

Paso 3 - Usa `Link` para navegar a las rutas

Si has estado codificando hasta este punto sin ningún error, tu navegador debería estar mostrando el componente `Inicio`.

Ahora emplearemos una funcionalidad distinta de *React Router* para navegar a otras páginas con base en esas rutas y nombres de ruta que hemos creado en el componente `App`. Esto es:

```
import { Link } from "react-router-dom";

function Inicio() {
  return (
    <div>
      <h1>Esta es la página de inicio</h1>
      <div className ="btn btn-group">
        <Link to="sobre-nosotros"
          className ="btn btn-primary">Haz clic para
ver la página sobre nosotros</Link>
        <Link to="contacto" className ="btn btn-success">
Haz clic para ver la página de contacto</Link>
      </div>
    </div>
  );
}

export default Inicio;
```

El componente `Link` es similar al elemento *anchor* (`<a>`) en HTML. Su atributo `to` especifica a qué ruta nos lleva el enlace.

Recuerda que hemos creado los nombres de ruta listados en el componente `App` para que cuando hagas clic en el enlace, revise tus rutas y renderice el componente que corresponda a dicho nombre. Recuerda que siempre debes importar `Link` desde `react-router-dom` antes de poder usarlo.

Documento original:

<https://www.freecodecamp.org/espanol/news/tutorial-de-react-router-version-6-como-navegar-a-otros-componentes-y-configurar-un-enrutador/>