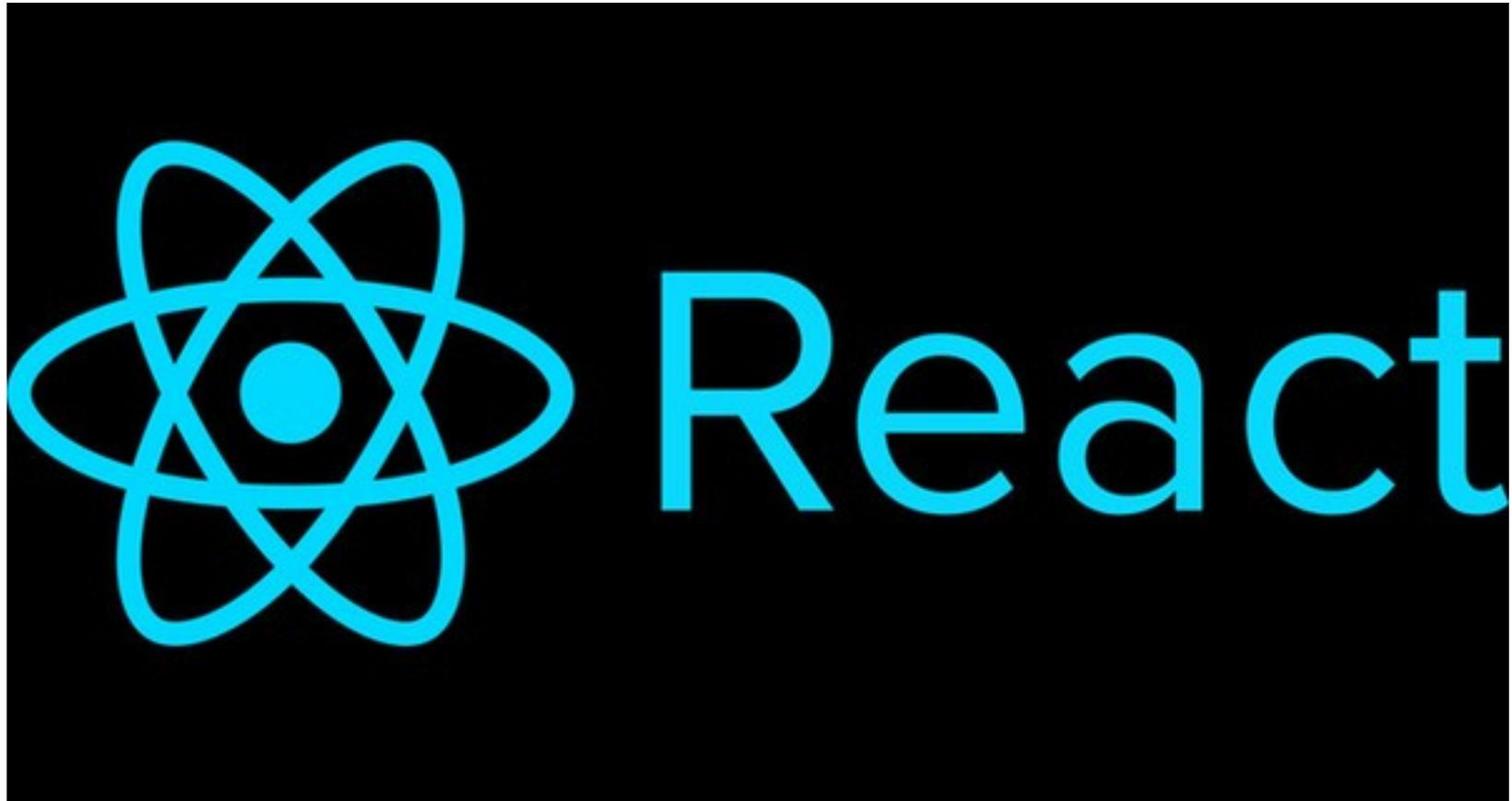


“Navegación entre componentes de React.JS”



Rogelio Ferreira Escutia

Navegar entre
componentes

Componentes

- Se quiere navegar de “App” a:
 - “Catalogo”
 - “Acerca de”

Instalar Biblioteca

Router

- Se requiere instalar “react-router-dom”.
- En la consola de nuestro proyecto (como root):

```
$ npm install --save react-router-dom
```

- Se recomienda también actualizar el “npm”:

```
New patch version of npm available! 6.14.4 → 6.14.5  
Changelog: https://github.com/npm/cli/releases/tag/v6.14.5  
Run npm install -g npm to update!
```



Componente principal

App.js

- Arrancamos con nuestro componente principal vacío:

```
src > JS App.js > [🔍] default
 1  import React from 'react';
 2
 3  function App() {
 4      return (
 5          <div>
 6
 7          </div>
 8      )
 9  }
10
11  export default App;
```

Crear componentes

Catalogo.js

- Creamos un nuevo archivo “Catalogo.js” dentro del directorio “src” de nuestro proyecto, el cual será nuestro componente “Catalogo”:

```
src > JS Catalogo.js > ...
 1  import React from 'react';
 2
 3  function Catalogo() {
 4      return (
 5          <div>
 6              <h1>Catálogo</h1>
 7          </div>
 8      )
 9  }
10
11  export default Catalogo;
```

Acerca.js

- Creamos un nuevo archivo “Acerca.js” dentro del directorio “src” de nuestro proyecto, el cual será nuestro componente “Acerca”:

```
src > JS Acerca.js > [e] default
 1  import React from 'react';
 2
 3  function Acerca() {
 4      return (
 5          <div>
 6              <h1>Acerca De</h1>
 7          </div>
 8      )
 9  }
10
11  export default Acerca;
```



Diseño

App.css

- Borramos nuestro archivo “App.css” y le agregamos diseño a la barra de navegación (nav) y a los enlaces que vamos a colocar (enlaces):

```
src > # App.css > nav
1  nav {
2      display: flex;
3      justify-content: space-around;
4      align-items: center;
5      min-height: 10vh;
6      background: gray;
7      color: white;
8  }
9
10 .enlaces {
11     width: 50%;
12     display: flex;
13     justify-content: space-around;
14     align-items: center;
15     list-style: none;
16 }
```



Barra de Navegación

Navegacion.js

- Creamos un nuevo archivo “Navegacion.js” dentro del directorio “src” de nuestro proyecto, el cual será nuestro componente “Navegacion” y servirá para controlar los enlaces de navegación:

```
src > JS Navegacion.js > [🔗] default
1   import React from 'react';
2   import './App.css';
3   import {Link} from 'react-router-dom';
4
5   function Navegacion() {
6     return (
7
8     )
9   }
10
11  export default Navegacion;
```



Navegacion.js

- Creamos los enlaces:

```
src > JS Navegacion.js > [🔗] default
 1  import React from 'react';
 2  import './App.css';
 3  import {Link} from 'react-router-dom';
 4
 5  function Navegacion() {
 6      const estilo = {
 7          color: 'white'
 8      };
 9      return (
10          <nav>
11              <ul className="enlaces">
12                  <Link style={estilo} to='/'>
13                      <li>Página Principal</li>
14                  </Link>
15                  <Link style={estilo} to='/catalogo'>
16                      <li>Catálogo</li>
17                  </Link>
18                  <Link style={estilo} to='/acerca'>
19                      <li>Acerca de</li>
20                  </Link>
21              </ul>
22          </nav>
23      )
24  }
25
26  export default Navegacion;
```



Llamar a los
componentes

App.js

- En “App.js” mandamos importar las bibliotecas y los componentes que acabamos de crear:

```
import React from 'react';  
import {BrowserRouter as Router, Switch, Route} from 'react-router-dom';  
import Catalogo from './Catalogo';  
import Acerca from './Acerca';  
import Navegacion from './Navegacion';
```

App.js

- Modificamos “App.js” para llamar a “<Navegacion />” y para agregar las rutas de los enlaces (Router):

```
function App() {  
  return (  
    <Router>  
      <div>  
        <Navegacion />  
        <Switch>  
          <Route path="/" exact component={index} />  
          <Route path="/catalogo" exact component={Catalogo} />  
          <Route path="/acerca" exact component={Acerca} />  
        </Switch>  
      </div>  
    </Router>  
  )  
}
```

App.js

- A nuestro archivo “App.js” le agregamos una constante (index) para indicarle que debe de renderizar cuando se mande al enlace “index”, que sería nuestra “Pagina Principal”:

```
const index = () => (  
  <div>  
    <h1>Página Principal</h1>  
  </div>  
);
```

App.js (completo)

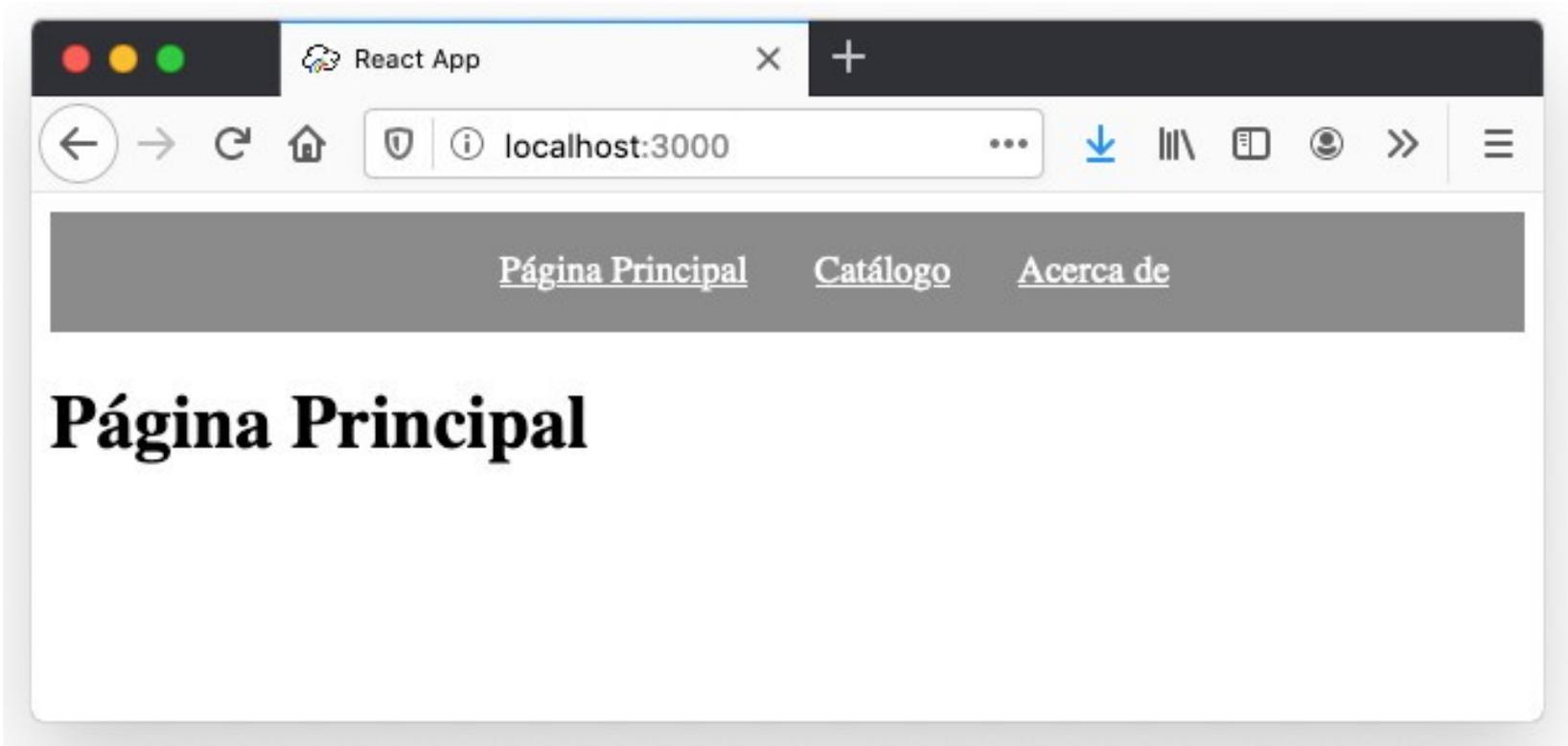
```
src > JS App.js > [🔍] default
 1  import React from 'react';
 2  import {BrowserRouter as Router, Switch, Route} from 'react-router-dom';
 3  import Catalogo from './Catalogo';
 4  import Acerca from './Acerca';
 5  import Navegacion from './Navegacion';
 6  function App() {
 7    return (
 8      <Router>
 9        <div>
10          <Navegacion />
11          <Switch>
12            <Route path="/" exact component={index} />
13            <Route path="/catalogo" exact component={Catalogo} />
14            <Route path="/acerca" exact component={Acerca} />
15          </Switch>
16        </div>
17      </Router>
18    )
19  }
20  const index = () => (
21    <div>
22      <h1>Página Principal</h1>
23    </div>
24  );
25  export default App;
```



Aplicación en ejecución
en el Navegador

Salida

- Así se observa la aplicación final con su barra de navegación y sus enlaces. Al hacer click se ejecuta el componente que se manda llamar y se mantiene la barra de navegación:





Rogelio Ferreira Escutia

***Instituto Tecnológico de Morelia
Departamento de Sistemas y Computación***

***Correo: rogelio@itmorelia.edu.mx
 rogeplus@gmail.com***

***Página Web: http://sagitario.itmorelia.edu.mx/~rogelio/
 http://www.xumarhu.net/***

Twitter: http://twitter.com/rogeplus

Facebook: http://www.facebook.com/groups/xumarhu.net/