



UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

**Centro de
e-Learning**

Desarrollo en React JS

Centro de e-Learning SCEU UTN - BA.

Medrano 951 2do piso (1179) // Tel. +54 11 4867 7589 / Fax +54 11 4032 0148

www.sceu.frba.utn.edu.ar/e-learning



UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

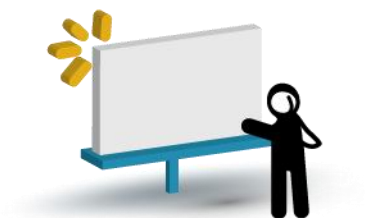
**Centro de
e-Learning**

p. 2

Módulo II

Componentes y Virtual DOM

Unidad 3: Manejo de rutas



Centro de e-Learning SCEU UTN - BA.

Medrano 951 2do piso (1179) // Tel. +54 11 4867 7589 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning



Presentación:

En esta unidad aprenderemos como hacer que nuestra aplicación sea navegable, como configurar nuestro router y hacer los links entre páginas.

Veremos un ejemplo de cómo refactorizar nuestra app para que su estructura sea más legible y escalable.



Objetivos:

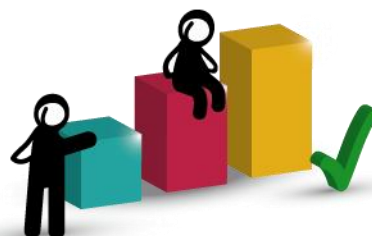
Que los participantes*:

- Comprendan que es el router
- Aprendan cómo desarrollar el router en react
- Configuren su aplicación con una estructura más escalable y legible.



Bloques temáticos*:

- React router
- Configurar nuestra aplicación
- Recibir parametros por URL
- Enlaces entre páginas
- Refactorizando nuestra app



Consignas para el aprendizaje colaborativo

En esta Unidad los participantes se encontrarán con diferentes tipos de actividades que, en el marco de los fundamentos del MEC*, los referenciarán a tres comunidades de aprendizaje, que pondremos en funcionamiento en esta instancia de formación, a los efectos de aprovecharlas pedagógicamente:

- Los foros proactivos asociados a cada una de las unidades.
- La Web 2.0.
- Los contextos de desempeño de los participantes.

Es importante que todos los participantes realicen algunas de las actividades sugeridas y compartan en los foros los resultados obtenidos.

Además, también se propondrán reflexiones, notas especiales y vinculaciones a bibliografía y sitios web.

El carácter constructivista y colaborativo del MEC nos exige que todas las actividades realizadas por los participantes sean compartidas en los foros.



Tomen nota*

Las actividades son opcionales y pueden realizarse en forma individual, pero siempre es deseable que se las realice en equipo, con la finalidad de estimular y favorecer el trabajo colaborativo y el aprendizaje entre pares. Tenga en cuenta que, si bien las actividades son opcionales, su realización es de vital importancia para el logro de los objetivos de aprendizaje de esta instancia de formación. Si su tiempo no le permite realizar todas las actividades, por lo menos realice alguna, es fundamental que lo haga. Si cada uno de los participantes realiza alguna, el foro, que es una instancia clave en este tipo de cursos, tendrá una actividad muy enriquecedora.

Asimismo, también tengan en cuenta cuando trabajen en la Web, que en ella hay de todo, cosas excelentes, muy buenas, buenas, regulares, malas y muy malas. Por eso, es necesario aplicar filtros críticos para que las investigaciones y búsquedas se encaminen a la excelencia. Si tienen dudas con alguno de los datos recolectados, no dejen de consultar al profesor-tutor. También aprovechen en el foro proactivo las opiniones de sus compañeros de curso y colegas.



React router

El componente router no es módulo oficial de react. Este nos permitirá hacer una web navegables, es decir poder redirigir entre distintas páginas.

Instalación

Debemos ubicarnos dentro del directorio en el cual está nuestra aplicación y ejecutar:

```
G:\sites\react\utn>npm install --save react-router
```

```
G:\sites\react\utn>npm install --save react-router-dom
```




Configurar nuestra aplicación

En el archivo **index.js** debemos realizar lo siguiente:

```
import React from 'react';
import ReactDOM from 'react-dom';
import './index.css';
import registerServiceWorker from './registerServiceWorker';
import {Link, Route, Router} from 'react-router';
import { BrowserRouter } from 'react-router-dom';
import App from './App';
import Home from './Home';

ReactDOM.render(<BrowserRouter>
  <div>
    <Route path="/" component={App} />
    <Route path="/home" component={Home} />
  </div>
</BrowserRouter>, document.getElementById('root'));

registerServiceWorker();
```

Como vemos debemos incluir los modulos **Link**, **Route**, **Router** y **BrowserRouter**

Luego debemos modificar el primer parametro del metodo **Render** en el cual incluiremos el elemento **BrowserRouter**

Si ejecutamos nuestra aplicación ingresara al contenido del componente **App** y si en la barra de direcciones colocamos **/home** ingresara al contenido del componente **Home**

Vemos que el elemento route recibe 2 propiedades **path** (indica la url por la que se ingresara a ese componente) y **component** (indica el contenido del componente a mostrar)



Recibir parametros por URL

En caso de querer recibir parametros por URL vamos a indicar en el componente route lo siguiente:

```
ReactDOM.render(<BrowserRouter>
  <div>
    <Route path="/" component={App} />
    <Route path="/home/:number" component={Home} />
  </div>
</BrowserRouter>, document.getElementById('root'));
```

Este elemento **:number** se enviará al componente **Home** como una propiedad, de la siguiente manera:

```
constructor(props) {
  super(props)
  console.log(props.match.params.number)
  this.state = {
    error: null,
    isLoading: false,
    perfiles: []
  };
}
```



Enlaces entre páginas

Para crear los enlaces entre páginas debemos importar el módulo **Link** en la página que queremos realizar el enlace. Por ejemplo vamos a realizar un enlace en la página **Home** hacia la página **Productos**:

```
import React, { Component } from 'react';  
import ReactDOM from 'react-dom';  
import Perfil from './Perfil';  
import {Link} from 'react-router-dom';
```

```
class Home extends Component {  
  
  constructor(props) {  
    super(props)  
    console.log(props.match.params.number)  
    this.state = {  
      nombre: null  
    }  
  }  
}
```

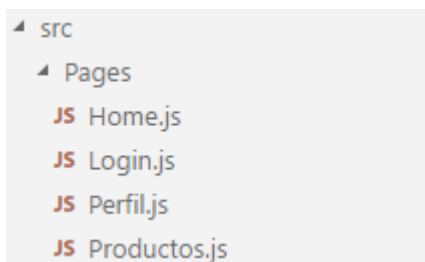
Luego utilizamos el elemento **<Link/>** en el cual colocaremos la URL a la cual queremos dirigirnos:

```
<div>  
<ul>  
  <li><Link to={'/productos/'}>Productos</Link></li>  
</ul>  
</div>
```



Refactorizando nuestra app

Para que quede un código más prolijo vamos a crear una nueva carpeta llamada **pages**, en la cual colocaremos los componentes que son páginas de nuestro código:



Cada página nueva que creemos en nuestra aplicación la agregaremos en la carpeta **Pages**

Debemos modificar algunos archivos, en los cuales debemos modificar el path que incluye ambos componentes:

Index.js

```
import Productos from './Pages/Productos';  
import Login from './Pages/Login';
```

App.js

```
import Productos from './PRODUCTOS';  
import Home from './Pages/Home';  
import logo from './logo.svg';
```

De esta manera nuestra aplicación quedará mucho más legible y con una estructura más escalable.



Bibliografía y Webgrafía utilizada y sugerida

Fedosejev, A. (2015). React.js Essentials (1 ed.). EEUU, Packt.

Amler, . (2016). ReactJS by Example (1 ed.). EEUU, Packt.

Stein, J. (2016). ReactJS Cookbook (1 ed.). EEUU, Packt.

<https://medium.com/@pshrmn/a-simple-react-router-v4-tutorial-7f23ff27adf>



Lo que vimos:

En esta unidad aprendimos como hacer que nuestra aplicación sea navegable, como configurar nuestro router y hacer los links entre páginas.



Lo que viene:

En la próxima unidad aprenderemos cómo manejar eventos y vistas con react js

