



Tecnologias Para Desenvolvimento Web

UNIDADE 05

Rotas

Em um site acessamos várias páginas, como por exemplo, página inicial, contatos, sobre, perfil, etc. Para navegar entre as páginas de uma aplicação React, precisamos criar **rotas**, onde **cada rota vai representar uma página**. Para trabalhar com rotas no React vamos utilizar um pacote chamado **React Router Dom** que precisa ser instalado no nosso projeto [1]. É ele que nos auxiliará na criação da navegação das páginas.

Sendo assim, devemos acessar o diretório raiz do projeto pelo terminal e executar o comando demonstrado na Figura 1.

Figura 1. Instalação do React Router Dom.

```
> npm install react-router-dom
```

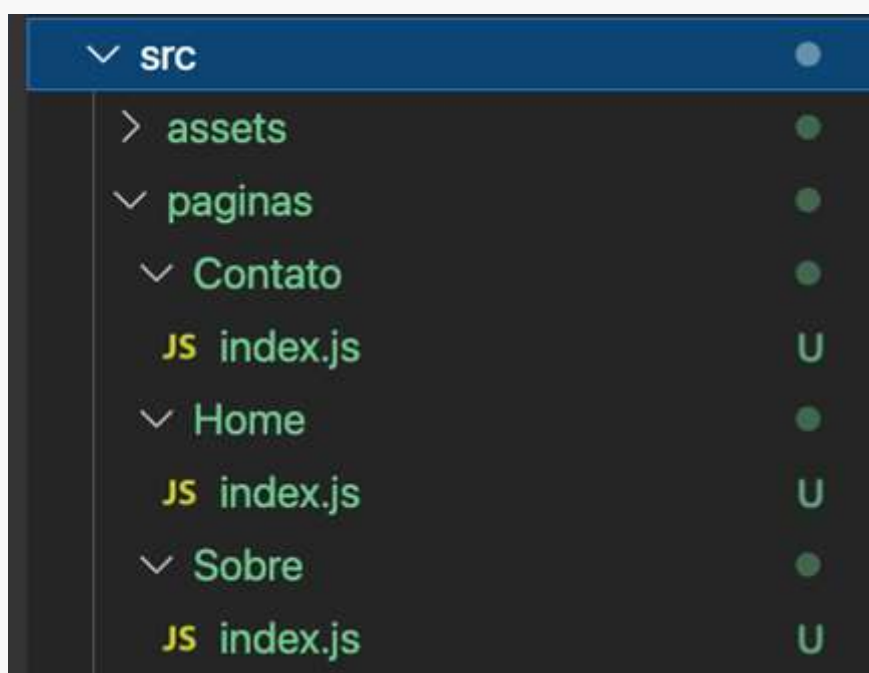
Fonte: o autor, 2021.

Dentro do pacote do **React Router Dom**, temos alguns componentes que vamos precisar aprender para que as rotas funcionem corretamente.

- **BrowserRouter**: este componente é responsável por informar à nossa aplicação que teremos um roteamento de componentes, por conta disso ele ficará em volta dos componentes que são do tipo **Route**.
- **Route**: este componente associará a rota aos componentes. Nele temos três parâmetros:
 - **component**: recebe o componente que precisa ser exibido ao acessar a rota;
 - **path**: caminho na URL que precisa ser acessado para mostrar o componente, definido pelo parâmetro componente;
 - **exact**: determina qual o componente vai ser exibido apenas se a rota for igual ao definido entre aspas.

Quando utilizamos rotas, significa que vamos ter outras páginas e consequentemente, diversos arquivos de componentes de páginas em nosso projeto. Por conta disso, precisamos deixar o diretório de nosso projeto organizado! Dentro da pasta **src** do nosso projeto, é onde fica nossos componentes e assim, vamos criar uma pasta chamada **paginas**. Dentro de **paginas**, teremos pastas com o nome das páginas e dentro delas, os componentes principais de cada página, como pode ser visualizado na Figura 2.

Figura 2. Organização do diretório de componentes.



Fonte: o autor, 2021.

Perceba que na Figura 2, foi criada a pasta **paginas** e dentro dela, foram criadas outras 3 novas pastas: **Home**, **Contato**, e **Sobre**. Dentro de cada pasta das páginas, temos arquivos em JavaScript chamados **index.js**. Este arquivo representará o componente principal de cada página.

Desta forma, nós já temos os diretórios e os arquivos para os componentes. Agora precisaremos criar as rotas. Estas rotas podem ser criadas em um novo arquivo chamado **rotas.js**.

Neste arquivo, devemos importar os componentes **BrowserRouter** e **Route** do pacote **React Router Dom**, como pode ser visualizado na Figura 3.

Figura 3. Importação dos componentes BrowserRouter e Route.

```
import { BrowserRouter, Route } from "react-router-dom";
```

Fonte: o autor, 2021.

Também devemos importar os componentes que farão parte das nossas rotas, ou seja, os componentes das páginas que poderão ser acessadas (Figura 4).

Figura 4. Importação dos componentes para as rotas.

```
import Home from './paginas/Home';  
import Contato from './paginas/Contato';  
import Sobre from './paginas/Sobre';
```

Fonte: o autor, 2021.

Assim, já podemos criar o novo componente com as rotas dos componentes das páginas. Sendo assim, precisamos retornar o **BrowserRouter** e dentro dele o mapeamento dos componentes **Route**, que vão acessar os componentes das páginas importadas. A Figura 5 mostra como esse mapeamento de rotas fica.

Figura 5. Mapeamento das rotas.

```
const Rotas = () => {
  return (
    <BrowserRouter>
      <Route exact={true} path="/" component={Home} />
      <Route exact={true} path="/sobre" component={Sobre} />
      <Route exact={true} path="/contato" component={Contato} />
    </BrowserRouter>
  )
}
```

Fonte: o autor, 2021.

Não se esqueça de no final, exportar o componente **Rotas**.

Desta forma, criamos o nosso mapeamento de rotas para as nossas páginas e isso poderá ser utilizado por todo o nosso projeto. Mas antes, precisamos fazer com que este novo componente **Rotas** seja executado. Para isso, vamos precisar importá-lo em nosso primeiro componente do projeto e em seguida, utilizado no HTML deste primeiro componente. Se você manteve as nomenclaturas dos nomes de arquivos padrões que são utilizados ao criar um novo projeto React, o nome do primeiro componente se chama **App.js**. Neste arquivo, vamos importar o componente Rotas e utilizá-lo. A Figura 6 mostra como deverá ficar o componente **App.js**.

Figura 6. Componente App.js.

```
import Rotas from './rotas';

function App() {
  return (
    <div>
      <Rotas />
    </div>
  );
}

export default App;
```

Fonte: o autor, 2021.

Assim, logo que for executado o projeto, irá criar o componente Rotas. Dentro deste componente temos o mapeamento das rotas dentro do **BrowserRouter**. A rota que possui o endereço raiz do projeto (apenas “/”), deverá ser o primeiro componente a ser executado.

Para trocar de páginas dentro de um componente em tempo de execução, podemos criar o componente **Link**, que é um componente muito parecido com a tag de hiperlink `<a>`. Para isso, precisamos importar o componente Link do pacote **React Router Dom** e utilizá-lo para redirecionar a página. Este componente possui um atributo chamado **to**, que devemos colocar a rota que será redirecionado ao clicar nele, como pode ser visualizado na Figura 7.

Figura 7. Componente Link.

```
<Link to="/sobre"> Sobre </Link>
```

Fonte: o autor, 2021.

Desta forma, já podemos criar uma aplicação em React mais completa, com diversas páginas e componentes.

| CONCLUSÃO

Nesta unidade, foi apresentado o conceito de componentes em classe e estado do componente. Estes dois conceitos são essenciais para o andamento do nosso conteúdo e entendimento da construção de componentes mais avançados.

| REFERÊNCIAS

[1] React. *React – Uma biblioteca JavaScript para criar interfaces de usuário*. Disponível em: <https://pt-br.reactjs.org/>. Acessado em: 12 de outubro de 2021.



