



Binh Tran

Follow

48 Followers

About

Private Route, Public Route, and Restricted Route with React Router



Binh Tran Dec 1, 2018 · 2 min read

React Router is one of the most important components of React ecosystem. In this tutorial, we will learn how to build different types of routes including private, public, and restricted routes. For a quick demonstration, we will build a simple application with following components:

1. Home: a public route that everyone can access to.
2. Dashboard: a private route that only authenticated user can access to.
3. Sign-in: a restricted route that unauthenticated user can see. To explain more, we don't want to show the sign-in page after logged in to the site. If the authorised user goes to the sign-in page, we will redirect to dashboard page.

Note: Please check the [Github repository](#) for the full source code

Private Route

PrivateRoute component is the blueprint for all private routes in the application. If the user is logged in, go on and display the component in question; otherwise, redirect the user to sign-in page. Additionally, we can define the logic of `isLogin` utility function separately in `utils` folder.

```
1 import React from 'react';
2 import { Route, Redirect } from 'react-router-dom';
3 import { isLogin } from '../utils';
4
5 const PrivateRoute = ({component: Component, ...rest}) => {
6   return (
```

```

6     return (
7
8         // Show the component only when the user is logged in
9         // Otherwise, redirect the user to /signin page
10        <Route {...rest} render={props => (
11            isLoggedIn() ?
12                <Component {...props} />
13            : <Redirect to="/signin" />
14        )} />
15    );
16 };
17
18 export default PrivateRoute;

```

PrivateRoute.js hosted with ♥ by GitHub

[view raw](#)

Public Route

On the other hand, PublicRoute component is the model for all public and restricted routes. The difference between two types is defined if `restricted` props. `restricted = false` meaning that route is public; or else, it is restricted.

```

1  import React from 'react';
2  import { Route, Redirect } from 'react-router-dom';
3  import { isLoggedIn } from '../utils';
4
5  const PublicRoute = ({component: Component, restricted, ...rest}) => {
6      return (
7          // restricted = false meaning public route
8          // restricted = true meaning restricted route
9          <Route {...rest} render={props => (
10              isLoggedIn() && restricted ?
11                  <Redirect to="/dashboard" />
12              : <Component {...props} />
13          )} />
14      );
15  };
16
17  export default PublicRoute;

```

PublicRoute.js hosted with ♥ by GitHub

[view raw](#)

Put all together in the <App>

Finally come to the `<App>`, we can conveniently declare all the routes we want without a sweat.

```

1  import React, { Component } from 'react';
2  import './App.css';

```

```

3  import { BrowserRouter, Switch } from 'react-router-dom';
4  import Home from './components/Home';
5  import Dashboard from './components/Dashboard';
6  import SignIn from './components/SignIn';
7  import PrivateRoute from './components/PrivateRoute';
8  import PublicRoute from './components/PublicRoute';
9
10 class App extends Component {
11
12   render() {
13     return (
14       <BrowserRouter>
15         <Switch>
16           <PublicRoute restricted={false} component={Home} path="/" exact />
17           <PublicRoute restricted={true} component={SignIn} path="/signin" exact />
18           <PrivateRoute component={Dashboard} path="/dashboard" exact />
19         </Switch>
20       </BrowserRouter>
21     );
22   }
23 }
24
25 export default App;

```

App.js hosted with ♥ by GitHub

[view raw](#)

Conclusion

Hopefully this tutorial can give you more insight about different types of routes and how to use them.

Please check this [GitHub repository](#) for the full source code and see the working example.

. . .

If you find the post useful, give me some **claps** 🙌 😊

Also check:

React State Management

With Hooks and Context API

medium.com



Dynamic form with React Hooks — Building a cost table

Hooks are new features of React 16.8. They allow us to write class component related features such as use state in a...

medium.com

Cost Table Example

Items	
Item	Price
Apple	40
Banana	20
Total	
60	

JavaScript

React

React Router



[About](#) [Help](#) [Legal](#)

Get the Medium app

