

React routes

1. Instalar react router dom con el comando `npm install react-router-dom@6`
2. Ir a `src/index.js` y envolver el component `App.js` con `BrowserRouter` (hay que importarlo desde `react-router-dom`)

```
import { BrowserRouter as Router } from "react-router-dom";
ReactDOM.render(
  <React.StrictMode>
    <Router>
      <App />
    </Router>
  </React.StrictMode>,
  document.getElementById('root')
);
```

3. Dentro de `src/components` crear un component para incorporar todas las rutas.
4. Importar `Routes` y `Route` de `react-router-dom`

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

5. Utilizar `Routes` para envolver todas las rutas/pages a utilizar.

```
const ProjectsRoutes = () => {
  return (
    <div>
      <Routes>
        <Route path="/portfolio_react/" element={<Home />} />
        <Route path="/proyectos-api" element={<ApiProjects />} />
        <Route path="/proyectos-react" element={<ReactProjects />} />
        <Route path="*" element={<Error404 />} />
      </Routes>
    </div>
  );
};
```

6. Armado de cada ruta: se pone la ruta a la cual se tiene que acceder y luego el componente al que va a referir.

```
<Route path="/portfolio_react/" element={<Home />} />
```

7. Para navegar entre componentes, utilizar Link de react-router-dom (se debe importar)

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

8. Generar el componente Link: se utilize to={} para establecer a qué hace referencia y luego se le debe poner nombre

```
<Link to={"/login"}>Login</Link>
```

9. Si se necesita redireccionar a una ruta específica, hay que utilizar Navigate

```
let navigate = useNavigate();  
navigate("/")
```

10. Para realizar rutas privadas se debe crear un nuevo componente PrivateRoute e importar Outlet y Navigate. Luego realizar el siguiente código en el componente:

```
import { Outlet, Navigate } from "react-router-dom";  
  
const userToken = JSON.parse(localStorage.getItem("userToken"));  
  
const PrivateRoute = () => {  
  return userToken ? <Outlet /> : <Navigate to="/login" />;  
};
```

11. Para que lo anterior funcione, en el component de las rutas se debe utilizar el PrivateRoute:

```
<Route path="/" element={<PrivateRoute />}>  
  <Route path="/" element={<Homepage />} />  
  <Route path="/buscar-platos" element={<DishSearcher />} />  
</Route>
```

12. Si el path de PrivateRoute es igual al del hijo que contiene, en el hijo se puede poner index en lugar del path:

```
<Route path="/" element={<PrivateRoute />}>
  <Route index element={<Homepage />} />
  <Route path="/buscar-platos" element={<DishSearcher />} />
</Route>
```