

# **JOBSCHEET PEMROGRAMAN BERBASIS FRAMEWORK**

## **React Router**



**Oleh:**

**Nama :Mirza Zarqani Rayhan**  
**NIM : 1841720205**  
**KELAS : TI-3F/17**

**Program Studi Teknik Informatika**  
**Jurusan Teknologi Informasi**  
**Politeknik Negeri Malang**  
**2021**

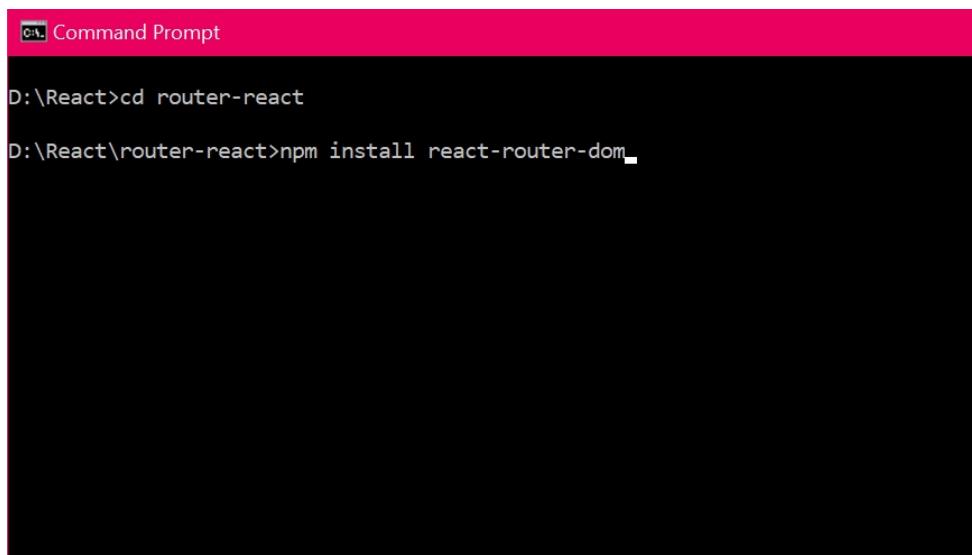
# Praktikum

Dalam membangun web SPA (*Single Page Application*) kita membutuhkan library React Router. Untuk itu langkah pertama yang kita lakukan adalah mengistall library React Router.

## 1.1 Langkah Praktikum

### a. Basic Router

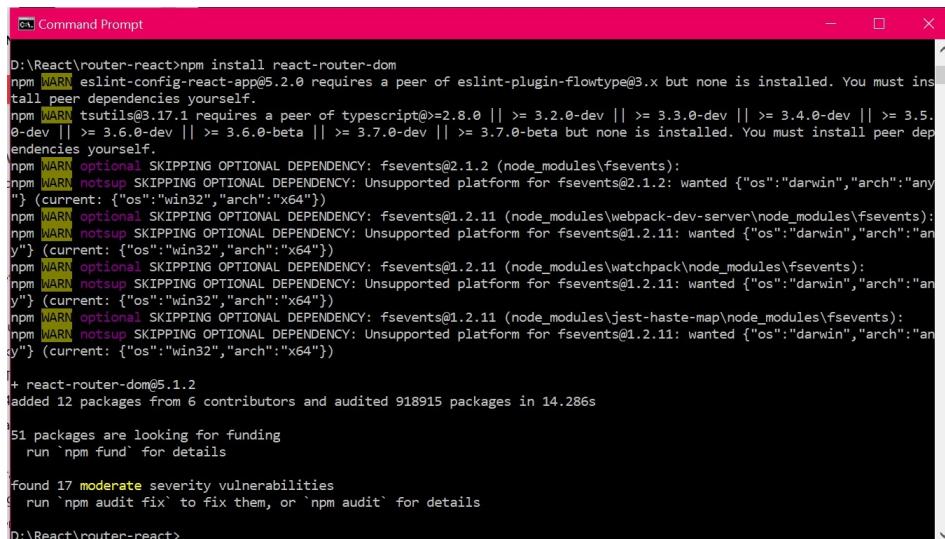
1. Setelah kita sukses melakukan instalasi react (*npm install*), kita perlu menambahkan library react router dengan menambahkan *npm install react-router-dom*.



```
Command Prompt

D:\React>cd router-react

D:\React\router-react>npm install react-router-dom
```



```
Command Prompt

D:\React\router-react>npm install react-router-dom
npm WARN eslint-config-react-app@5.2.0 requires a peer of eslint-plugin-flowtype@3.x but none is installed. You must install peer dependencies yourself.
npm WARN tsutils@3.17.1 requires a peer of typescript@>=2.8.0 || >= 3.2.0-dev || >= 3.3.0-dev || >= 3.4.0-dev || >= 3.5.0-dev || >= 3.6.0-dev || >= 3.6.0-beta || >= 3.7.0-dev || >= 3.7.0-beta but none is installed. You must install peer dependencies yourself.
npm [optional] SKIPPING OPTIONAL DEPENDENCY: fsevents@2.1.2 (node_modules\fsevents):
npm [WARN] notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.1.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm [WARN] optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.11 (node_modules\webpack-dev-server\node_modules\fsevents):
npm [WARN] notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.11: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm [WARN] optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.11 (node_modules\watchpack\node_modules\fsevents):
npm [WARN] notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.11: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm [WARN] optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.11 (node_modules\jest-haste-map\node_modules\fsevents):
npm [WARN] notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.11: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

+ react-router-dom@5.1.2
added 12 packages from 6 contributors and audited 918915 packages in 14.286s

51 packages are looking for funding
  run 'npm fund' for details

found 17 moderate severity vulnerabilities
  run 'npm audit fix' to fix them, or 'npm audit' for details

D:\React\router-react>
```

2. Kemudian buka app.js pada dan ketikkan code dibawah ini.

```
JS App.js  X
src > JS App.js > App
  1| import React from "react";
  2| import {
  3|   BrowserRouter as Router,
  4|   Switch,
  5|   Route,
  6|   Link
  7| } from "react-router-dom";
  8|
  9| export default function App() {
10|   return (
11|     <Router>
12|       <div>
13|         <nav>
14|           <ul>
15|             <li>
16|               <Link to="/">Home</Link>
17|             </li>
18|             <li>
19|               <Link to="/about">About</Link>
20|             </li>
21|             <li>
22|               <Link to="/users">Users</Link>
23|             </li>
24|           </ul>
25|         </nav>
26|
27|         {/* A <Switch> looks through its children <Route>s and
28|           | renders the first one that matches the current URL. */}
29|         <Switch>
30|           <Route path="/about">
31|             <About />
32|           </Route>
33|           <Route path="/users">
34|             <Users />
35|           </Route>
36|           <Route path="/">
37|             <Home />
38|           </Route>
39|         </Switch>
40|       </div>
41|     </Router>
42|   );
43| }

43| }
44|
45| function Home() {
46|   return <h2>Home</h2>;
47| }
48|
49| function About() {
50|   return <h2>About</h2>;
51| }
52|
53| function Users() {
54|   return <h2>Users</h2>;
55| }
```

3. Lakukan npm start, lalu cek hasilnya akan seperti dibawah ini.

The image contains three screenshots of a browser window showing a navigation menu and content for different pages. The browser address bar shows 'localhost:3000'. The first screenshot shows the 'Home' page with a menu on the left and the text 'Pilih setiap menu lalu amati bagaimana perubahannya.' The second screenshot shows the 'Users' page. The third screenshot shows the 'About' page. Each screenshot has a different background color (light blue, light green, and light orange) and shows a different part of the application's content.

## b. URL Parameters

1. Pada percobaan kali ini kita mencoba membuat link react router dengan menggunakan params, sehingga kita hanya membuat satu template yang dapat berubah kontenya sesuai dengan apa yang kita klik.
2. Ketikkan seperti code dibawah ini.

```

JS App.js x
src > JS App.js > App
1 import React from "react";
2 import {
3   BrowserRouter as Router,
4   Switch,
5   Route,
6   Link,
7   useParams
8 } from "react-router-dom";
9
10 export default function App() {
11   return (
12     <Router>
13       <div>
14         <h2>Accounts</h2>
15         <nav>
16           <ul>
17             <li>
18               <Link to="/netflix">Netflix</Link>
19             </li>
20             <li>
21               <Link to="/gmail">Gmail</Link>
22             </li>
23             <li>
24               <Link to="/yahoo">Yahoo</Link>
25             </li>
26             <li>
27               <Link to="/amazon">Amazon</Link>
28             </li>
29           </ul>
30         </nav>
31
32         {/* A <Switch> looks through its children <Route>s and
33          | renders the first one that matches the current URL. */}
34         <Switch>
35           <Route path="/:id" children={<Child />}>/</Route>
36         </Switch>
37       </div>
38     </Router>
39   );
40 }

```

```

41
42  function Child() {
43    let { id } = useParams();
44
45    return (
46      <div>
47        |   <h3>ID: {id}</h3>
48      </div>
49    );
50
51 }

```

3. Cek hasilnya maka akan seperti dibawah ini.



### Accounts

- [Netflix](#)
- [Gmail](#)
- [Yahoo](#)
- [Amazon](#)

ID: **amazon**

Secara tampilan memang hasilnya hampir sama dengan praktikum sebelumnya, namun dengan metode yang berbeda.

### c. Use Nesting Router

1. Jika kita ingin melakukan percabangan link di dalam react router, maka kita dapat menggunakan cara ini sebagai solusinya.
2. Ketikkan seperti code dibawah ini.

```

JS App.js      JS Redirect.js      JS Nesting.js      JS Parameters.js      JS Router.js
src > JS Nesting.js > Topics
1  import React from "react";
2  import {
3    BrowserRouter as Router,
4    Switch,
5    Route,
6    Link,
7    useParams,
8    useRouteMatch
9  } from "react-router-dom";
10
11 export default function App() {
12   return (
13     // route merupakan komponen regular dari react
14     // maka dari itu bisa dimasukkan dalam child element
15     // membantu dikarenakan pemisahan kode di app react router yang sama dengan pemecahan kode react lainnya
16     <Router>
17       <div>
18         <nav>
19           <ul>
20             <li>
21               |   <Link to="/">Home</Link>
22             </li>
23             <li>
24               |   <Link to="/topics">Topics</Link>
25             </li>
26           </ul>
27         </nav>
28       </div>

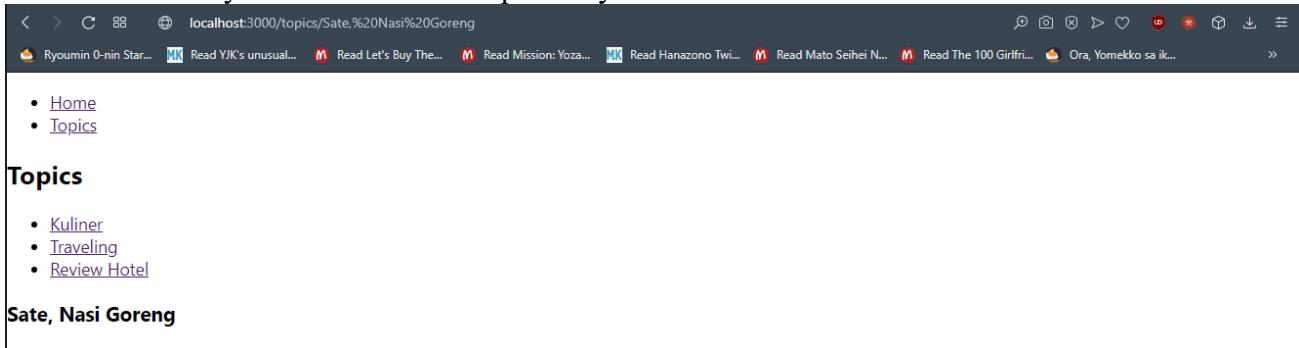
```

```

29     /* A <Switch> looks through its children <Route>s and
30     |   renders the first one that matches the current URL. */
31     <Switch>
32       <Route exact path ="/">
33       |   <Home />
34     </Route>
35       <Route path ="/topics">
36       |   <Topics />
37     </Route>
38   </Switch>
39 </div>
40 </Router>
41 );
42 }
43
44 function Home(){
45   return(
46     <div>
47     |   <h2>Home</h2>
48   </div>
49 );
50 }
51 <function Topics()>
52   // path membuat route terhadap route induk
53   // sedangkan url membuat link
54   let {path, url} = useRouteMatch();
55   return (
56     <div>
57       <h2>Topics</h2>
58       <ul>
59         <li>
60           |   <Link to={`${url}/Sate, Nasi Goreng`}>Kuliner</Link>
61         </li>
62         <li>
63           |   <Link to={`${url}/Wisata Alam, Museum`}>Traveling</Link>
64         </li>
65         <li>
66           |   <Link to={`${url}/Ibis, JW Marriot`}>Review Hotel</Link>
67         </li>
68       </ul>
69
70       <Switch>
71         <Route exact path ={path}>
72           |   <Route />
73         </Route>
74         <Route path ={`${path}/:topicId`}>
75           |   <Topic />
76         </Route>
77       </Switch>
78     </div>
79   );
80 }
81
82 <function Topic()>
83   let { topicId } = useParams();
84
85   return(
86     <div>
87       <h3>{topicId}</h3>
88     </div>
89   );
90 }

```

### 3. Cek hasilnya dan coba klik setiap link nya.



localhost:3000/topics/Sate.%20Nasi%20Goreng

Ryoumin 0-nin Star... [Read YJK's unusual...](#) [Read Let's Buy The...](#) [Read Mission: Yoza...](#) [Read Hanazono Twi...](#) [Read Mato Sehei N...](#) [Read The 100 Girffri...](#) [Ora, Yomekko sa ik...](#)

• [Home](#)  
• [Topics](#)

**Topics**

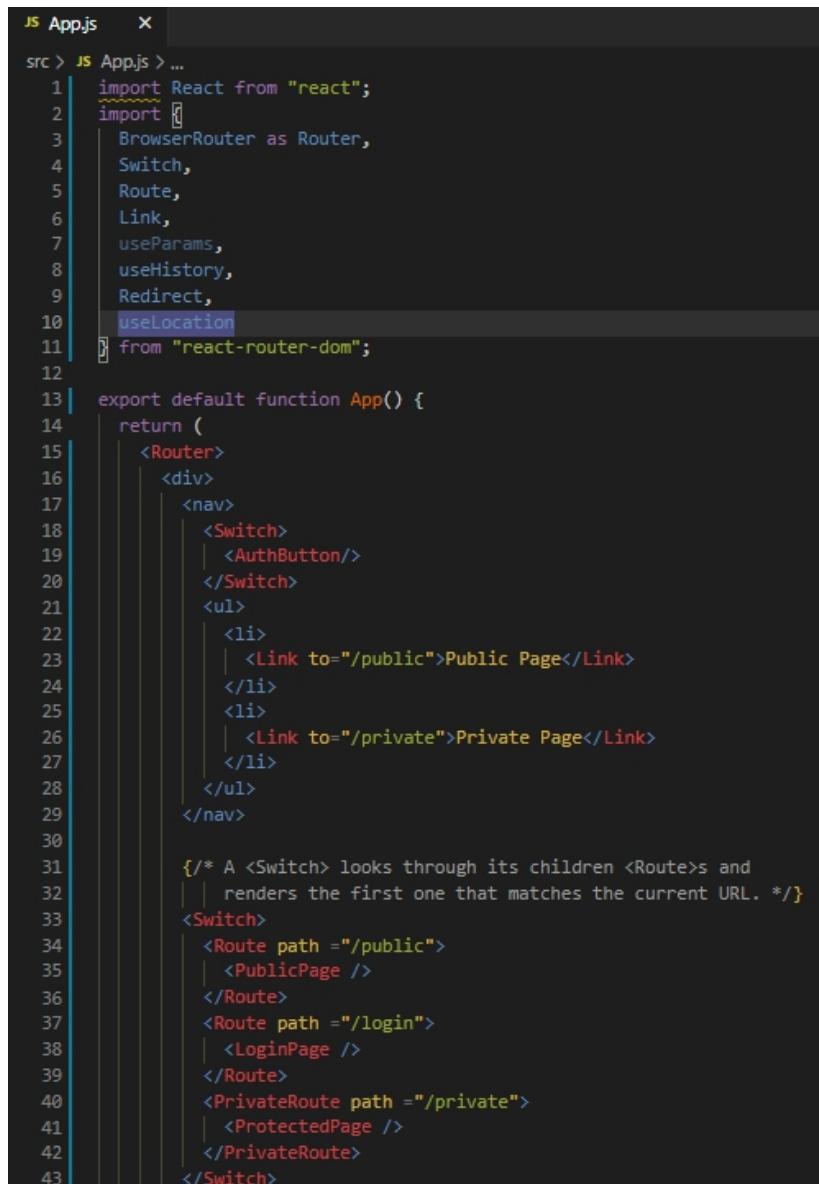
- [Kuliner](#)
- [Traveling](#)
- [Review Hotel](#)

**Sate, Nasi Goreng**

Bisa kita lihat bahwa link induk yang diatas akan menentukan hasil yang keluar dibawahnya, dan link yang dibawah dapat memunculkan hasil yg berbeda pula.

#### d. Use Redirects (Auth)

1. Cara ini dapat anda gunakan untuk masuk ke halaman yang membutuhkan authentication. Setelah masuk dan melakukan klik ke menu lain termasuk masuk ke halaman yang memerlukan authentication, posisi user tidak perlu login ulang (karena posisi sudah logged) sampai user malakukan sign out.



```
JS App.js  X
src > JS App.js > ...
1 import React from "react";
2 import {
3   BrowserRouter as Router,
4   Switch,
5   Route,
6   Link,
7   useParams,
8   useHistory,
9   Redirect,
10  useLocation
11 } from "react-router-dom";
12
13 export default function App() {
14   return (
15     <Router>
16       <div>
17         <nav>
18           <Switch>
19             | <AuthButton/>
20           </Switch>
21           <ul>
22             | <li>
23               |   <Link to="/public">Public Page</Link>
24             | </li>
25             | <li>
26               |   <Link to="/private">Private Page</Link>
27             | </li>
28           </ul>
29         </nav>
30
31         {/* A <Switch> looks through its children <Route>s and
32           | renders the first one that matches the current URL. */}
33         <Switch>
34           <Route path="/public">
35             | <PublicPage />
36           </Route>
37           <Route path="/login">
38             | <LoginPage />
39           </Route>
40           <PrivateRoute path="/private">
41             | <ProtectedPage />
42           </PrivateRoute>
43         </Switch>

```

2. Ketikkan seperti code dibawah ini.

```
JS App.js  x
src > JS App.js > ...
43 |         </Switch>
44 |     </div>
45 |     </Router>
46 | );
47 }
48
49 const fakeAuth = {
50     isAuthenticated: false,
51     authenticate(cb) {
52         fakeAuth.isAuthenticated = true;
53         setTimeout(cb, 100);
54     },
55     signout(cb) {
56         fakeAuth.isAuthenticated = false;
57         setTimeout(cb, 100);
58     }
59 };
60
61 function AuthButton(){
62     let history = useHistory();
63
64     return fakeAuth.isAuthenticated ? (
65         <p>
66             Welcome!{" "}
67             <button onClick={() => {
68                 fakeAuth.signout(() => history.push("/"));
69             }}>
70             >
71                 Sign out
72             </button>
73         </p>
74     ) : (
75         <p>You are not logged in.</p>
76     )
77 }
78 }
```

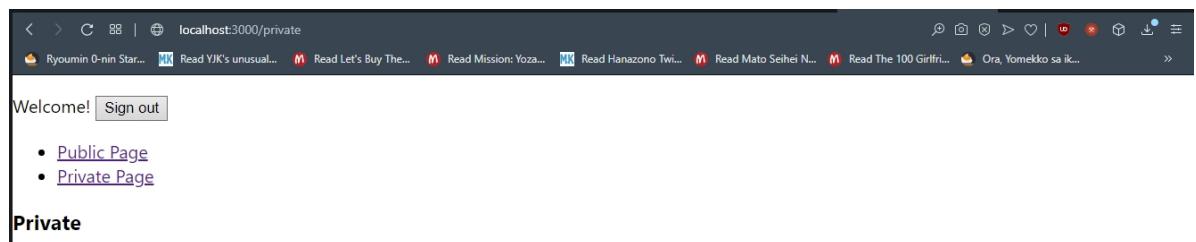
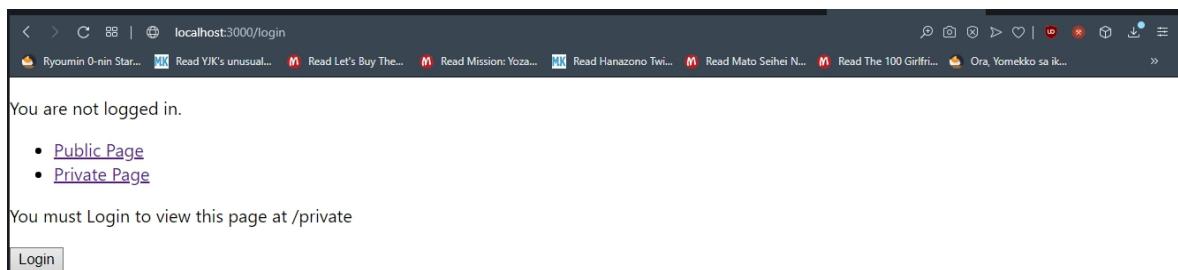
```
79 | function PrivateRoute({children, ...rest }) {
80 |     return(
81 |         <Route
82 |             {... rest}
83 |             render = {({ location }) =>
84 |                 fakeAuth.isAuthenticated ? (
85 |                     children
86 |                 ) : (
87 |                     <Redirect
88 |                         to= {{
89 |                             pathname: "/login",
90 |                             state: {from: location}
91 |                         }}
92 |                     />
93 |                 )
94 |             }
95 |         />
96 |     );
97 | }
98
99 | function PublicPage(){
100 |     return <h3>Public</h3>;
101 | }
102
103 | function ProtectedPage(){
104 |     return <h3>Private</h3>;
105 | }
106 | }
```

```

107 |     function LoginPage(){
108 |       let history = useHistory();
109 |       let location = useLocation();
110 |
111 |       let { from } = location.state || { from: { pathname: "/" } };
112 |       let login = () =>{
113 |         fakeAuth.authenticate(() =>{
114 |           history.replace(from);
115 |         });
116 |       };
117 |
118 |       return (
119 |         <div>
120 |           <p>You must Login to view this page at {from.pathname}</p>
121 |           <button onClick={login}>Login</button>
122 |         </div>
123 |       );
124 |     }

```

3. Cek hasilnya, ketika klik private page maka diharuskan untuk login terlebih dahulu, ketika sudah masuk dan klik private page kembali kita tidak perlu login kembali karena kita berada dalam posisi logged.



## 1.2 Soal Latihan Praktikum

1. Buatlah sebuah situs marketplace(online shop) sederhana, dengan menerapkan konsep SPA (Single Page Application) React Router dimana beberapa menu nya jika di klik terdapat fitur “Nesting” dan “Redirect (Auth)”.

```

src > JS App.js > ...
1 u import React from "react";
2 import 'bootstrap/dist/css/bootstrap.min.css'
3 import {
4   BrowserRouter as Router,
5   Switch,
6   Route,
7   Link,
8   Redirect,
9   useHistory,
10  useLocation,
11  useParams,
12  useRouteMatch
13 } from "react-router-dom";
14
15 v export default function App() {
16   return (
17     <Router>
18       <div>
19         <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
20           <div class="container-fluid">
21             <a class="navbar-brand" href="#">Fujiwara Tofu Shop</a>
22             <div class="collapse navbar-collapse" id="navbarNavAltMarkup">
23               <div class="navbar-nav">
24                 <Link to="/home" class="nav-item nav-link">Home</Link>
25                 <Link to="/barang" class="nav-item nav-link">Barang</Link>
26                 <Link to="/bio" class="nav-item nav-link">Bio</Link>
27                 <Switch class="nav-item nav-link">
28                   <AuthButton />
29                 </Switch>
30               </div>
31             </div>
32           </div>
33         </nav>
34         {/* A <Switch> looks through its children <Route>s and
35          |   renders the first one that matches the current URL. */}
36         <Switch>
37           <Route path="/home">
38             <PublicPage />
39           </Route>
40           <Route path="/login">
41             <LoginPage />
42           </Route>

```

```

43           <PrivateRoute path="/barang">
44             <ProtectedPage />
45           </PrivateRoute>
46           <PrivateRoute path="/bio">
47             <AboutPage />
48           </PrivateRoute>
49         </Switch>
50       </div>
51     </Router>
52   );
53 }
54 // authentication
55 const fakeAuth = {
56   isAuthenticated: false,
57   authenticate(cb) {
58     fakeAuth.isAuthenticated = true;
59     setTimeout(cb, 100);
60   },
61   signout(cb) {
62     fakeAuth.isAuthenticated = false;
63     setTimeout(cb, 100);
64   }
65 };
66 // kondisi login/ blom
67 function AuthButton(){
68   let history = useHistory();
69
70   return fakeAuth.isAuthenticated ? (
71     <a class="nav-item nav-link">
72       Welcome! " "
73       <button onClick={() => {
74         fakeAuth.signout(() => history.push("/"));
75       }}
76         class="btn btn-danger">
77           Sign out
78         </button>
79       </a>
80   ) : (
81     <a class="nav-item nav-link disabled" >You are not logged in.</a>
82   )
83 }

```



```

250  }
251 // tampilan about yang menggunakan nesting
252 function AboutPage(){
253   let {path, url} = useRouteMatch();
254   return (
255     <div>
256       <h2>Biodata</h2>
257       <ul>
258         <li>
259           | <Link to={`${url}/Mirza Zarqani Rayhan`}>Nama</Link>
260         </li>
261         <li>
262           | <Link to={`${url}/1841720205`}>NIM</Link>
263         </li>
264         <li>
265           | <Link to={`${url}/TI-3F, 17`}>Kelas, NoAbsen</Link>
266         </li>
267       </ul>
268
269       <Switch>
270         <Route exact path ={path}>
271           | <Route />
272         </Route>
273         <Route path =` ${path} /:bioId`>
274           | <Bio />
275         </Route>
276       </Switch>
277     </div>
278   );
279 }
280
281 // fungsi untuk menampilkan bioId
282 function Bio(){
283   let { bioId } = useParams();
284
285   return(
286     <div>
287       | <h3>{bioId}</h3>
288     </div>
289   );
290 }

```

```

291 // tampilan laman login
292 function LoginPage(){
293   let history = useHistory();
294   let location = useLocation();
295
296   let { from } = location.state || { from: { pathname: "/" } };
297   let login = () =>{
298     fakeAuth.authenticate(() =>{
299       history.replace(from);
300     });
301   };
302
303   return (
304     <div>
305       | <p>You must Login to view this page at {from.pathname}</p>
306       | <button onClick={login}>Login</button>
307     </div>
308   );
309 }
310

```

## Hasil

Selamat datang di Fujiwara Tofu Shop!  
Toko Tahu dengan pengiriman tercepat di ALAM SEMESTA

Mulai Belanja!

Fujiwara Tofu Shop Home Barang Bio You are not logged in.

You must Login to view this page at /barang

Login

Tahu Yang Tersedia

Tahu Kuning

Tahu Putih

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis

Tahu Pona

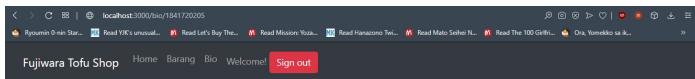
Fujiwara Tofu Shop Home Barang Bio Welcome! Sign out

Mirza Zarqan Rayhan

### Biodata

- Nama
- NIM
- Kelas, NoAbsen

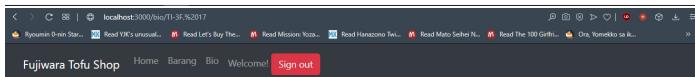
Mirza Zarqan Rayhan



### Biodata

- Nama
- NIM
- Kelas, NoAbsen

1841720205



### Biodata

- Nama
- NIM
- Kelas, NoAbsen

TI-3F, 17

---

Link Github : <https://github.com/mirzarayhan/react-router>

Link Youtube :

<https://www.youtube.com/playlist?list=PLwlHmkeMYrlUWvBZtw3DSNdcDXieLtguJ> (Playlist)

<https://youtu.be/Ho8obscKAf4> (Video)