

TUGAS PERORANGAN

LAPORAN MINGGU 6 ROUTE

Disusun sebagai

MATA KULIAH: PEMROGRAMAN BERBASIS FRAMEWORK

Oleh:

Rois Dwi Admaja/1741720193

TI – 3B / 23



PROGRAM STUDI: D-IV TEKNIK INFORMATIKA

JURUSAN: TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

2020

a. Basic Router

1. Instalasi react-router

```
PS E:\KULIAH\SEMESTER 6\PEMROGRAMAN FRAMEWORK\tugas_Rois\rois\modul5> 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 || >= 3.7.0-dev || >= 3.7.0-beta but none is installed, you must install peer dependencies yourself.
npm WARN 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\webpack\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
```

2. Melakukan import modul route

```
1 import React from 'react';
2 import logo from './logo.svg';
3 import './App.css';
4 import {BrowserRouter as Router,
5       Switch,
6       Route,
7       Link} from "react-router-dom";
8
```

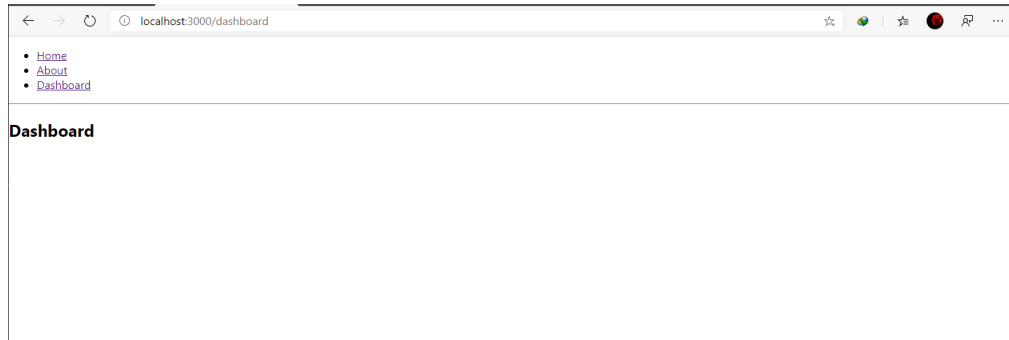
3. Membuat route

```
9 function App() {
10   return (
11     <Router>
12       <div>
13         <ul>
14           <li><Link to="/">Home</Link></li>
15           <li><Link to="/about">About</Link></li>
16           <li><Link to="/dashboard">Dashboard</Link></li>
17         </ul>
18         <hr />
19         <Switch>
20           <Route exact path="/">
21             <Home/>
22           </Route>
23           <Route path="/about">
24             <About/>
25           </Route>
26           <Route path="/dashboard">
27             <Dashboard/>
28           </Route>
29         </Switch>
30       </div>
31     </Router>
32   );
33 }
```

4. Membuat fungsi yang menampung halaman route

```
rois-modul5 > src > App.js > App
35 function Home() {
36   return (
37     <div>
38       <h2>Home</h2>
39     </div>
40   );
41 }
42
43 function About() {
44   return(
45     <div>
46       <h2>About</h2>
47     </div>
48   )
49 }
50
51 function Dashboard() {
52   return(
53     <div>
54       <h2>Dashboard</h2>
55     </div>
56   )
57 }
58
59 export default App;
```

5. Hasil dari program diatas



b. Url Parameter

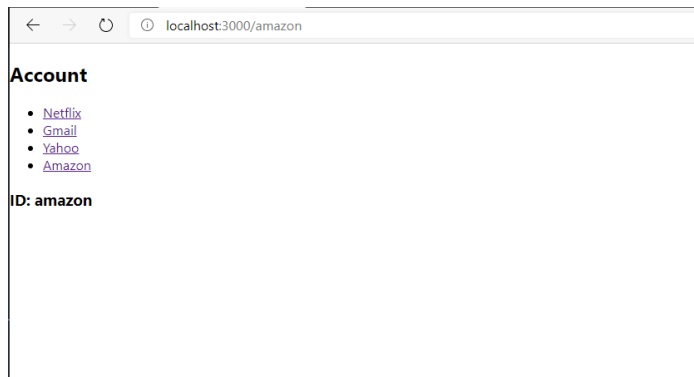
1. Membuat router

```
rois-modul5 > src > App.js > Child
1 import React from 'react';
2 import logo from './logo.svg';
3 // import './App.css';
4 import {BrowserRouter as Router,
5       Switch,
6       Route,
7       Link, useParams} from "react-router-dom";
8
9 export default function ParamsExample() {
10   return (
11     <Router>
12       <div>
13         <h2>Account</h2>
14         <ul>
15           <li><Link to="/netflix">Netflix</Link></li>
16           <li><Link to="/gmail">Gmail</Link></li>
17           <li><Link to="/yahoo">Yahoo</Link></li>
18           <li><Link to="/amazon">Amazon</Link></li>
19         </ul>
20         <Switch>
21           <Route path("/:id" children={<Child />} /> />
22         </Switch>
23       </div>
24     </Router>
25   );
26 }
```

Pada function child digunakan untuk mengambil nilai dari link

```
26 }
27
28 function Child() {
29   let {id} = useParams();
30
31   return(
32     <div>
33       <h3>ID: {id}</h3>
34     </div>
35   )
36 }
```

2. Hasil dari percobaan diatas



c. Use Nesting router

1. Membuat program nesting router

```
rois-modul5 > src > App.js > Topic
1  import React from 'react';
2  import {BrowserRouter as Router,
3      Switch,
4      Route,
5      Link, useParams, useRouteMatch} from "react-router-dom";
6
7  export default function NestingExample() {
8      return (
9          <Router>
10             <div>
11                 <h2>Account</h2>
12                 <ul>
13                     <li><Link to="/">Home</Link></li>
14                     <li><Link to="/Topics">Topics</Link></li>
15                 </ul>
16                 <hr/>
17                 <Switch>
18                     <Route exact path="/"><Home /></Route>
19                     <Route path="/Topics"><Topics /></Route>
20                 </Switch>
21             </div>
22         </Router>
23     );
24 }
25
26 function Home() {
27     return(
28         <div>
29             <h2>Home</h2>
30         </div>
31     )
32 }
33
34 function Topics() {
35     let {path, url} = useRouteMatch();
36     return(
37         <div>
38             <h2>Topics</h2>
39             <ul>
40                 <li>
41                     <Link to={`${url}/Sate, Nasi goreng`}>Kuliner</Link>
42                 </li>
43                 <li>
44                     <Link to={`${url}/Wisata alam, Museum`}>Traveling</Link>
45                 </li>
46                 <li>
47                     <Link to={`${url}/Ibis, JW Marriot`}>Review Hotel</Link>
48                 </li>
49             </ul>

```

```

50
51     <Switch>
52       <Route exact path={path}>
53         <h3>Please select a topic.</h3>
54       </Route>
55       <Route path={` ${path}/${topicId}`}>
56         <Topic/>
57       </Route>
58     </Switch>
59   </div>
60 )
61 }
62
63 function Topic() {
64   let{topicId} = useParams();
65
66   return(
67     <div>
68       <h3>{topicId}</h3>
69     </div>
70   )
71 }

```

Hasil dari program diatas

← → ↺ 🌐 localhost:3000/Topics/Sate,%20Nasi%20goreng

Account

- [Home](#)
- [Topics](#)

Topics

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

Sate, Nasi goreng

d. Use Redirects (Auth)

1. Membuat link route

```
1  import React from 'react';
2  import {BrowserRouter as Router,
3      Switch,
4      Route,
5      Link, Redirect, useHistory, useLocation} from "react-router-dom";
6
7  export default function AuthExample() {
8      return (
9          <Router>
10             <div>
11                 <AuthButton />
12                 <ul>
13                     <li><Link to="/public">Public Page</Link></li>
14                     <li><Link to="/private">Private Page</Link></li>
15                 </ul>
16                 <hr/>
17                 <Switch>
18                     <Route path="/public"><PublicPage /></Route>
19                     <Route path="/login"><LoginPage /></Route>
20                     <PrivateRoute path="/private"><ProtectedPage /></PrivateRoute>
21                 </Switch>
22             </div>
23         </Router>
24     );
25 }
```

2. Membuat fakeauth digunakan untuk melakukan autentikasi apakah sudah login atau belum, lalu ada auth button digunakan untuk melakukan logout

```
26
27
28  const fakeAuth = {
29      isAuthenticated: false,
30      authenticate(cb) {
31          fakeAuth.isAuthenticated = true;
32          setTimeout(cb, 100);
33      },
34      signOut(cb) {
35          fakeAuth.isAuthenticated = false;
36          setTimeout(cb, 100);
37      }
38  };
39
40  function AuthButton() {
41      let history = useHistory();
42
43      return fakeAuth.isAuthenticated ? (
44          <p>
45              Welcome!{" "}
46              <button onClick={()=>{
47                  fakeAuth.signOut(()=>history.push("/"));
48              }}>Sign Out</button>
49          </p>
50      ) : (
```

3. Private route adalah sebuah route yang hanya bisa diakses jika sudah diautentikasi sudah login atau belum

```
51 |     <p>You're not logged in</p>
52 |   );
53 | }
54 | function PrivateRoute({children, ...rest}) {
55 |   return(
56 |     <Route
57 |       {...rest}
58 |       render={({ location })=>
59 |         fakeAuth.isAuthenticated ? (
60 |           children
61 |         ):(
62 |           <Redirect
63 |             to={{
64 |               pathname: "/login",
65 |               state: {from: location}
66 |             }}
67 |           />
68 |         )
69 |       )
70 |     />
71 |   );
72 | }
73 |
```

4. Jika sudah login maka user akan diberikan akses untuk halaman tersebut

```
74 | function PublicPage() {
75 |   return <h3>Public</h3>
76 | }
77 |
78 | function ProtectedPage() {
79 |   return <h3>Private</h3>;
80 | }
81 |
82 | function LoginPage() {
83 |   let history = useHistory();
84 |   let location = useLocation();
85 |
86 |   let {from} = location.state || {from: {pathname: "/"}};
87 |   let login = () => {
88 |     fakeAuth.authenticate(()=>{
89 |       history.replace(from);
90 |     });
91 |   };
92 |   return (
93 |     <div>
94 |       <p>you must log in to view the page at {from.pathname}</p>
95 |       <button onClick={login}>Log in</button>
96 |     </div>
97 |   )
98 | }
```