

LAPORAN TUGAS MATA KULIAH
PEMROGRAMAN BERBASIS
FRAMEWORK



Nama : Alenovan Wiradhika Putra

Kelas : TI-3C D-IV

T.INFORMATIKA NIM1741720065

No. Absen 02

PROGRAM STUDI D-IV TEKNIK
INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2020

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

```
PS D:\Kuliah\Semester 6\React\pertemuan 5\starter> npm install react-router-dom
npm WARN rollback Rolling back node-pre-gyp@0.14.0 failed (this is probably harmless): EPERM: operation not permitted, rmdir 'D:\Kuliah\Semester 6\React\pertemuan 5\starter\node_modules\webpack-dev-server\node_modules\node-pre-gyp'
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 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\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"})
```

Kemudian buka app.js pada dan ketikkan code dibawah ini

localhost:3000/dashboard localhost:3000/about localhost:3000

- [Home](#)
- [About](#)
- [dashboard](#)

```
import React from 'react';
import logo from './logo.svg';
import './App.css';
import {
  BrowserRouter as Router,
  Switch,
  Route,
  Link
} from "react-router-dom";

export default function BasicExample() {
  return (
    <Router>
```

```
<div>
  <ul>
    <li><Link to="/">Home</Link></li>
    <li><Link to="/about">About</Link></li>
    <li><Link to="/dashboard">dashboard</Link></li>
  </ul>
</div>
<Switch>
  <Route exact path="/"></Route>
  <Route exact path="/about"></Route>
  <Route exact path="/dashboard"></Route>
</Switch>
</Router>
);
}

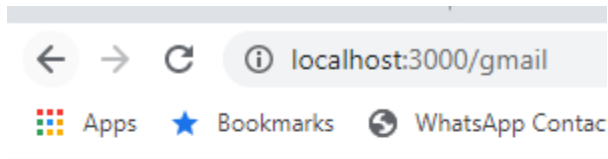
function Home() {
  return (
    <div>
      <h2>Home</h2>
    </div>
  );
}

function About() {
  return (
    <div>
      <h2>About</h2>
    </div>
  );
}

function Dashboard() {
  return (
    <div>
      <h2>Dashboard</h2>
    </div>
  );
}
```

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 kontennya sesuai dengan apa yang kita klik.
2. Ketikkan seperti code dibawah ini.



- [netflix](#)
- [gmail](#)
- [yahoo](#)
- [amazon](#)

ID: gmail

```
import React from 'react';
import logo from './logo.svg';
import './App.css';
import {
  BrowserRouter as Router,
  Switch,
  Route,
  Link,
  useParams
} from "react-router-dom";

export default function BasicExample() {
  return (
    <Router>
      <div>
        <ul>
          <li><Link to="/netflix">netflix</Link></li>
          <li><Link to="/gmail">gmail</Link></li>
          <li><Link to="/yahoo">yahoo</Link></li>
          <li><Link to="/amazon">amazon</Link></li>
        </ul>
      </div>
      <Switch>
        <Route path="/:id" children={<Child />}</Route>
      </Switch>
    </Router>
  );
}
```

```

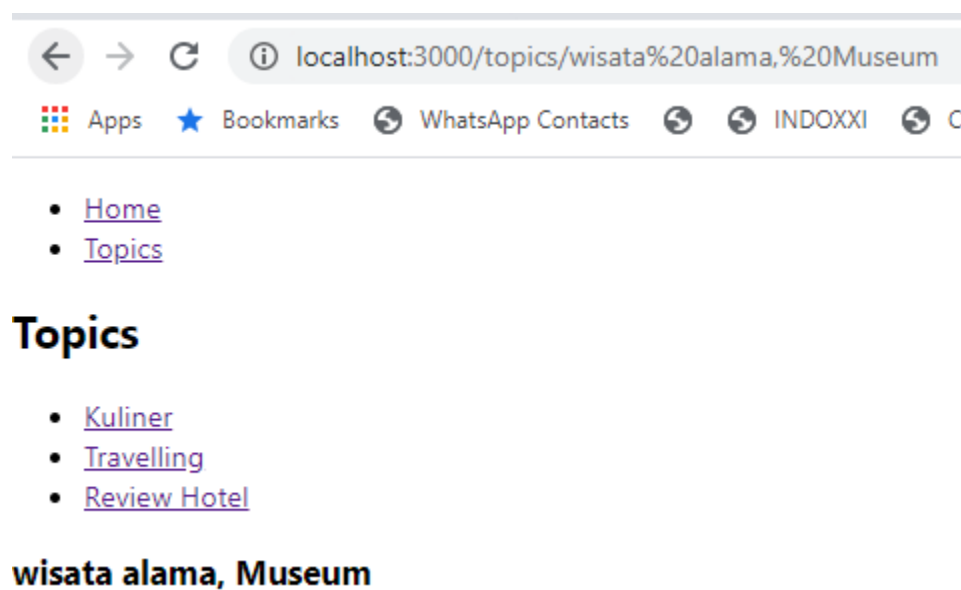
    );
  }

  function Child(){
    let {id} = useParams();
    return(
      <div>
        <h3>ID: {id}</h3>
      </div>
    )
  }
}

```

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



```

import React from 'react';
import './App.css';
import {
  BrowserRouter as Router,
  Switch,
  Route,
  Link,
  useParams,
  useRouteMatch
} from 'react-router-dom';

```

```

} from "react-router-dom";

export default function BasicExample() {
  return (
    <Router>
      <div>
        <ul>
          <li><Link to="/home">Home</Link></li>
          <li><Link to="/topics">Topics</Link></li>
        </ul>
      </div>
      <Switch>
        <Route exact path="/"><Home /></Route>
        <Route path="/topics"><Topics /></Route>
      </Switch>
    </Router>
  );
}

```

```

function Home() {
  return (
    <div>
      <h2>Home</h2>
    </div>
  );
}

```

```

function Topics() {
  let { path, url } = useRouteMatch();
  return (
    <div>
      <h2>Topics</h2>
      <ul>
        <li><Link to={` ${url}/Sate, Nasi Goreng`} >Kuliner</Link></li>
        <li><Link to={` ${url}/wisata alama, Museum`} >Travelling</Link></li>
        <li><Link to={` ${url}/ibis, jw marriot`} >Review Hotel</Link></li>
      </ul>
      <Switch>
        <Route exact path={path}>
          <h3>Please select a topic</h3>
        </Route>
        <Route path={` ${path}/:topicId`} >

```

```

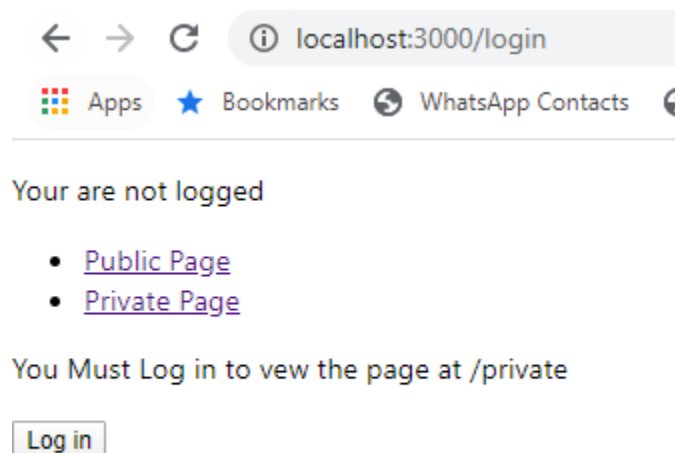
    <Topic />
  </Route>
</Switch>
</div>
)
}

function Topic() {
  let { topicId } = useParams();
  return (
    <div>
      <h3>{topicId}</h3>
    </div>
  )
}

```

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 melakukan sign out.
2. Ketikkan seperti code dibawah ini



```

import React from 'react';
import './App.css';
import {
  BrowserRouter as Router,
  Switch,
  Route,

```

```

Link,
Redirect,
useHistory,
useLocation
} from "react-router-dom";

export default function AuthExample() {
  return (
    <Router>
      <div>
        <AuthButton />
        <ul>
          <li><Link to="/public">Public Page</Link></li>
          <li><Link to="/private">Private Page</Link></li>
        </ul>

        <Switch>
          <Route path="/public"><PublicPage /></Route>
          <Route path="/login"><LoginPage /></Route>
          <PrivateRoute path="/private"><ProtectedPage /></PrivateRoute>
        </Switch>
      </div>
    </Router>
  );
}

const fakeAuth = {
  isAuthenticated: false,
  authenticate(cb) {
    fakeAuth.isAuthenticated = true;
    setTimeout(cb, 100);
  },
  signout(cb) {
    fakeAuth.isAuthenticated = false;
    setTimeout(cb, 100)
  }
}

function AuthButton() {
  let history = useHistory();
  return fakeAuth.isAuthenticated ? (
    <p>
      Welcome !{" "}
      <button onClick={() => { fakeAuth.signout(() => history.push("/")); }} />
    </p>
  ) : (
    <p>
      Not logged in
    </p>
  );
}

```



```

    </p>
  ): (
    <p>Your are not logged</p>
  )
}

function PrivateRoute({ children, ...rest }) {
  return (
    <Route
      {...rest}
      render={({ location }) =>
        fakeAuth.isAuthenticated ? (
          children
        ) : (
          <Redirect to={{
            pathname: "/login",
            state: { from: location }
          }} />
        )
      }
    />
  )
}

function LoginPage() {
  let history = useHistory();
  let location = useLocation();

  let { from } = location.state || { from: { pathname: "/" } }
  let login = () => {
    fakeAuth.authenticate(() => {
      history.replace(from)
    })
  }

  return (
    <div>
      <p>You Must Log in to vew the page at {from.pathname}</p>
      <button onClick={login}>Log in</button>
    </div>
  )
}

function PublicPage() {
  return <h3>Public</h3>
}

```

```
}  
  
function ProtectedPage() {  
  return <h3>Private</h3>  
}
```