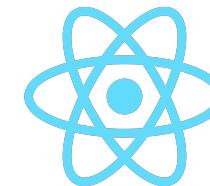


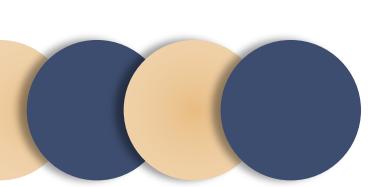


# Pemrograman Sisi Cleint

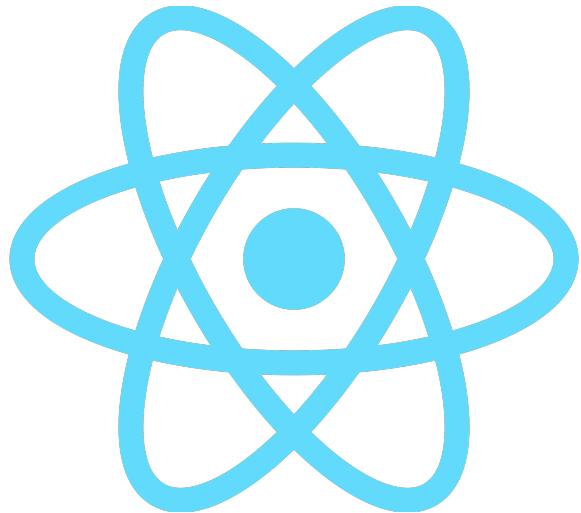
## ReactJS Route

Muhammad Syaifur Rohman, S.Kom, M.CS





# Definisi



## FYI

React Router adalah pustaka standar untuk mengelola navigasi dan routing dalam aplikasi React. Ini memungkinkan pengembang untuk membuat aplikasi single-page dengan navigasi multipage yang terasa seperti aplikasi multi-page tradisional. React Router mendukung berbagai fitur seperti route nesting, URL parameters, dan route guards.

# Instalasi

atau

`npm install react-router-dom`

`yarn add react-router-dom`

1. Instalasi
2. Persiapkan File tiap halaman
3. Buat Route
4. Eksekusi !

# Persiapkan File

```
// Pages/Home.jsx
import React from 'react';

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

export default Home;

// Pages/About.jsx
import React from 'react';

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

export default About;
```

```
// Pages/Contact.jsx
import React from 'react';

function Contact() {
  return <h2>Contact Page</h2>;
}

export default Contact;

// Pages/Notfound.jsx
import React from 'react';

function Notfound() {
  return <h2>Contact Page</h2>;
}

export default Notfound;
```

TRY...  


# Buat Route

**BrowserRouter:** Membungkus seluruh aplikasi dan mengaktifkan penggunaan routing di dalamnya.

**Routes:** Membungkus semua route

**Route:** Menentukan komponen mana yang akan dirender berdasarkan URL path.

**Link:** Digunakan untuk navigasi tanpa memuat ulang halaman.

```
// App.js
import React from 'react';
import { BrowserRouter as Router, Routes, Route, Link } from 'react-router-dom';
import Homepage from './pages/Homepage';
import About from './pages/About';
import Contact from './pages/Contact';
import Notfound from './pages/Notfound';

function App() {
  return (
    <Router>
      <div>
        <nav>
          <ul>
            <li><Link to="/">Home</Link></li>
            <li><Link to="/about">About</Link></li>
            <li><Link to="/contact">Contact</Link></li>
          </ul>
        </nav>
        <Routes>
          <Route path="/" element={<Homepage />} />
          <Route path="/about" element={<About />} />
          <Route path="/contact" element={<Contact />} />
          <Route path="*" element={<Notfound />} />
        </Routes>
      </div>
    </Router>
  );
}

export default App;
```

TRY...

# Hasil



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

Halaman About

# Nested Routing

**React Router memungkinkan nesting routes, yang berarti Anda bisa membuat struktur routing yang lebih kompleks di dalam komponen.**

**useRouteMatch:** Hook yang membantu mendapatkan informasi tentang path dan URL saat ini untuk membuat routing dinamis lebih mudah.

**Nested Route:** Menggunakan path yang dinamis (`path`/`team`) untuk membuat sub-route di dalam halaman About.

```
// App.js
import React from 'react';
import { BrowserRouter as Router, Route, Routes, Link } from 'react-router-dom';
import Homepage from './pages/Homepage';
import About from './pages/About';
import Contact from './pages/Contact';
import Notfound from './pages/Notfound';
import Team from './pages/Team';

function App() {
  return (
    <Router>
      <div>
        <nav>
          <ul>
            <li><Link to="/">Home</Link></li>
            <li><Link to="/about">About</Link></li>
            <li><Link to="/contact">Contact</Link></li>
          </ul>
        </nav>

        <Routes>
          <Route path="/" element={<Homepage />} />
          <Route path="/about" element={<About />}>
            <Route path="team" element={<Team />} />
          </Route>
          <Route path="/contact" element={<Contact />} />
          <Route path="*" element={<Notfound />} />
        </Routes>
      </div>
    </Router>
  );
}

export default App;
```

 TRY...

# Nested Routing 2

```
// pages/About.jsx
import React from 'react';
import { Link, Outlet } from 'react-router-dom';

function About() {
  return (
    <div>
      <h2>About Page</h2>
      <ul>
        <li><Link to="team">Our Team</Link></li>
      </ul>
      <Outlet />
    </div>
  );
}

export default About;
```

```
// pages/Team.jsx
import React from 'react';

function Team() {
  return <h3>Meet our team!</h3>;
}

export default Team;
```

TRY...  
→

The image displays two side-by-side screenshots of a web browser interface, likely from a mobile device, showing the 'About' page and its 'Team' section.

**Screenshot 1: About Page**

- Address bar: localhost:3000/about
- Content:
  - [Home](#)
  - [About](#)
  - [Contact](#)
- Section header: **About Page**
- List item: • [Our Team](#)

**Screenshot 2: Team Section of the About Page**

- Address bar: localhost:3000/about/team
- Content:
  - [Home](#)
  - [About](#)
  - [Contact](#)
- Section header: **About Page**
- List item: • [Our Team](#)
- Text: **Meet our team!**

# Routing URL Paramters

**React Router mendukung parameter dinamis dalam URL, yang sangat berguna untuk membuat aplikasi yang lebih dinamis, seperti halaman detail.**

**useParams:** Hook yang memungkinkan akses ke parameter dinamis dalam URL.

**Dynamic Route:** Menggunakan path dinamis /product/:id untuk menangani rute yang berbeda berdasarkan parameter URL.

```
// App.jsx
import React from 'react';
import { BrowserRouter as Router, Route, Routes, Link } from 'react-router-dom';
import Homepage from './pages/Homepage';
import About from './pages/About';
import Contact from './pages/Contact';
import Product from './pages/Product';
import Notfound from './pages/Notfound';
import Team from './pages/Team';

function App() {
  return (
    <Router>
      <div>
        <nav>
          <ul>
            <li><Link to="/">Homepage</Link></li>
            <li><Link to="/about">About</Link></li>
            <li><Link to="/contact">Contact</Link></li>
            <li><Link to="/product/Beras">Product Beras</Link></li>
            <li><Link to="/product/Gula">Product Gula</Link></li>
          </ul>
        </nav>
        <Routes>
          <Route path="/" element={<Homepage />} />
          <Route path="/about" element={<About />} />
          <Route path="team" element={<Team />} />
        </Route>
        <Route path="/contact" element={<Contact />} />
        <Route path="/product/:id" element={<Product />} />
        <Route path="*" element={<Notfound />} />
      </Routes>
    </div>
  </Router>
);
}

export default App;
```

TRY...  
→

# Routing URL Parameters 2

- □ ×

```
// pages/Product.jsx
import React from 'react';
import { useParams } from 'react-router-dom';

function Product() {
  let { id } = useParams();
  return <h2>Product: {id}</h2>;
}

export default Product;
```

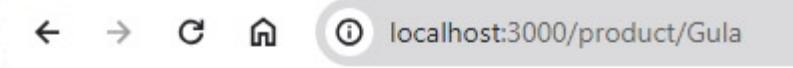
TRY...  
→

# Hasil



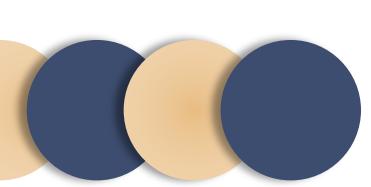
- [Homepage](#)
- [About](#)
- [Contact](#)
- [Product Beras](#)
- [Product Gula](#)

**Product: Beras**



- [Homepage](#)
- [About](#)
- [Contact](#)
- [Product Beras](#)
- [Product Gula](#)

**Product: Gula**



# Penutup



**Dengan memahami dan menggunakan React Router, Anda dapat membuat aplikasi React yang kompleks dengan navigasi dan routing yang mudah dikelola. Hal ini tidak hanya meningkatkan pengalaman pengguna tetapi juga membuat aplikasi lebih modular dan terstruktur dengan baik.**

---



# Penutup

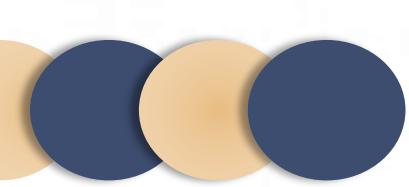
**BrowserRouter:** Gunakan BrowserRouter untuk membungkus aplikasi Anda dan mengaktifkan routing.

**Route Management:** Gunakan Route, Switch, dan Link untuk mengelola routing aplikasi.

**Nested Routing:** Manfaatkan nested routing untuk struktur routing yang lebih kompleks.

**Dynamic URL:** Gunakan parameter dinamis untuk membuat rute yang lebih fleksibel.

**Konsistensi:** Pertahankan struktur dan konvensi penamaan yang konsisten untuk path URL dan komponen.



# Referensi

1. Eran Krinsbruner, **A Frontend Web Developer's Guide to Testing**, 2022
2. Adam Boduch , Roy Derks , Mikhail Sakhniuk, **React and React Native - Fourth Edition**, 2022
3. Maya Shavin , Raymond Camden, **Frontend Development Projects with Vue.js 3 - Second Edition**, 2023
4. Waweru Mwaura, **End-to-End Web Testing with Cypress**, 2021
5. <https://roadmap.sh/frontend>
6. <https://jsonplaceholder.typicode.com/posts>



# Thanks

Muhammad Syaifur Rohman, S.Kom, M.CS

