



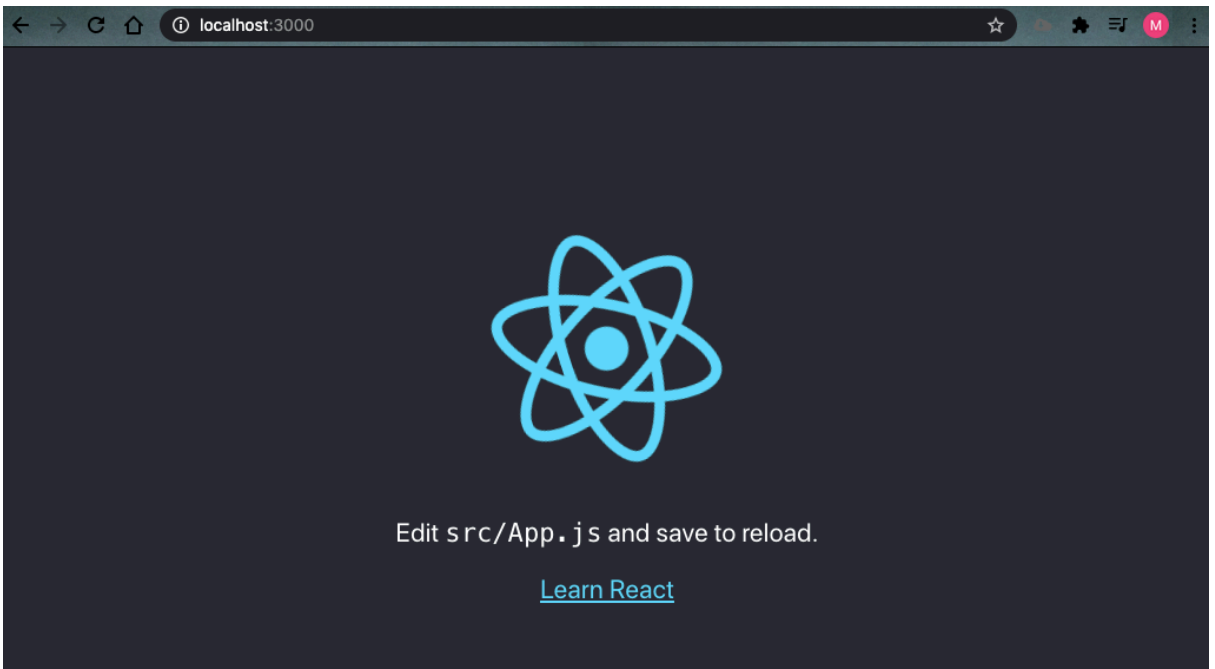
Nama : Muhammad Iqbaluddin Al Huda

Kelas : TI-3E

NIM : 1841720013

Tujuan

1. Mahasiswa dapat mengetahui dan mengimplementasikan kegunaan Stateless component & statefull component
2. Mahasiswa dapat mengetahui dan mengimplementasikan kegunaan Dynamic component menggunakan props
3. Mahasiswa dapat mengetahui dan mengimplementasikan kegunaan Update parent oleh child component
4. Mahasiswa dapat mengetahui dan mengimplementasikan kegunaan Lifecycle component 1&2

| Langkah | Keterangan |
|---------|--|
| 1 | <p>a. Membuat Stateless component & statefull component Project sebelumnya create hello world Bagaimana cara membuat komponen dengan stateless component : 1. Membuka halaman react</p>  |
| 2 | Membuat function baru dengan hello component (stateless component) |



3

Buat class component / statefull component dengan memanggil react .component
Memanggil StateFullComponent didalam react

```
index.js x JS Test.js JS List.js JS Footer.js JS App.js
c > JS index.js > [e] HelloComponent
1 import React from 'react';
2 import ReactDOM from 'react-dom';
3 import './index.css';
4 import App from './App';
5 //import reportWebVitals from './reportWebVitals';
6 import * as serviceWorker from './serviceWorker';
7
8
9 const HelloComponent = () => {
10   return HelloComponent
11
12   class StateFullComponent extends React.Component{
13     render() {
14       return <p>StateFullComponent</p>
15     }
16   }
17
18   ReactDOM.render(<StateFullComponent />, document.getElementById('root'));
19
20
21 // If you want to start measuring performance in your app, pass a function
22 // to log results (for example: reportWebVitals(console.log))
23 // or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
24 serviceWorker.unregister();
```



StateFullComponent

4

Props atau properties digunakan untuk menampung nilai yang nantinya dikirimkan ke suatu Component.

```

index.js  JS Test.js  JS List.js  JS Footer.js  JS App.js  x
c > JS App.js > ...
1 // import logo from './logo.svg';
2 import './App.css';
3 import React, { Component } from 'react';
4 import Header from './header/Header';
5 import Footer from './footer/Footer';
6 import List from './list/List';
7
8 class App extends Component {
9
10   render() {
11     return (
12       <div>
13         <Header/>
14         <h1>Component dari Class App</h1>
15         <List/>
16         <Footer judul='Halaman Footer' nama='Aufa' />
17       </div>
18     );
19   }
20 }
21 export default App;
22 // function App() {
23 //   return (
24 //     <div className="App">
25 //       <header className="App-header">
26 //         <img src={logo} className="App-logo" alt="logo" />
27 //       <p>

```

Buatlah Program berikut in

5

```

index.js  JS Test.js  JS List.js  JS Footer.js  x  JS App.js
c > footer > JS Footer.js > [e] Footer
1 import React from 'react'; // Component menggunakan Functionconst
2
3 const Footer = (props) => {
4   return (
5     <div>
6       <h3>Halaman Footer</h3>
7       <h3>Component ini dibuat menggunakan Function bukan Class</h3>
8       <p>Nilai ini ditampilkan dari props: { props.judul } </p>
9       <p>Nama Saya: { props.nama } </p>
10     </div>
11   );
12 }
13
14 export default Footer;
15

```

Component dari Class App



1. Satu



3. Tiga



4. Empat

Halaman Footer

Component ini dibuat menggunakan Function bukan Class

Nilai ini ditampilkan dari props: Halaman Footer

Nama Saya: Aufa

6

Pada kode di atas, kirim mengirimkan sebuah parameter props ke fungsi (anonymous function), lalu parameter tersebut akan kita gunakan untuk mengakses variabel yang dikirimkan dari file App.js.

Props pada Class Component

Contoh Kasus:

Kita ingin menampilkan banyak gambar di dalam component list, di mana component list akan menggunakan component image.

Component List

```

index.js  JS Test.js  JS List.js  JS Footer.js  JS App.js
c > list > JS List.js > ...
1  import React, { Component } from 'react';
2  import Image from '../image/Image';
3
4  class List extends Component {
5    render() {
6      return (
7        <div>
8          <ol>
9            <li>
10             Satu
11             <Image linkGambar='https://s3-ap-southeast-1.amazonaws.com/niomic/img/sample/food' />
12            </li>
13            <li>
14             Dua
15             <Image linkGambar='https://s3-ap-southeast-1.amazonaws.com/niomic/img/sample/nasi' />
16            </li>
17            <li>
18             Tiga
19             <Image linkGambar='https://s3-ap-southeast-1.amazonaws.com/niomic/img/sample/sate' />
20            </li>
21            <li>
22             Empat
23             <Image linkGambar='https://s3-ap-southeast-1.amazonaws.com/niomic/img/sample/soto' />
24            </li>
25          </ol>
26        </div>
27      );
28    }
29  }

```

Component Image

Buatlah Program berikut ini

```

index.js  JS Test.js  JS List.js  JS Image.js  JS Footer.js  JS App.js
c > image > JS Image.js > Image > render
1  import React, { Component } from 'react';
2
3  class Image extends Component {
4    render() {
5      return (
6        <img src={ this.props.linkGambar } alt="Food" width='500' />
7      );
8    }
9  }
10  (alias) class Image
11  export default Image;

```

7

8

Buatlah Program dibawah ini dan deteksilah bagian dari lifecycle Component

```

JS Test.js > Test > render
import React from 'react';
import ReactDOM from 'react-dom';

class Test extends React.Component {
  constructor(props) {
    {
      super(props);
      this.state = { hello : "World!" };
    }

    componentWillMount()
    {
      console.log("componentWillMount()");
    }

    componentDidMount()
    {
      console.log("componentDidMount()");
    }

    changeState()
    {
      this.setState({ hello : "Geek!" });
    }

    render()
    {
      return (
        <div>
          <h1>GeeksForGeeks.org, Hello{ this.state.hello }</h1>
          <h2>
            <a onClick={this.changeState.bind(this)}>Press Here!</a>
          </h2>
        </div>);
    }

    shouldComponentUpdate(nextProps, nextState)
    {
      console.log("shouldComponentUpdate()");
      return true;
    }
  }
}

```

9

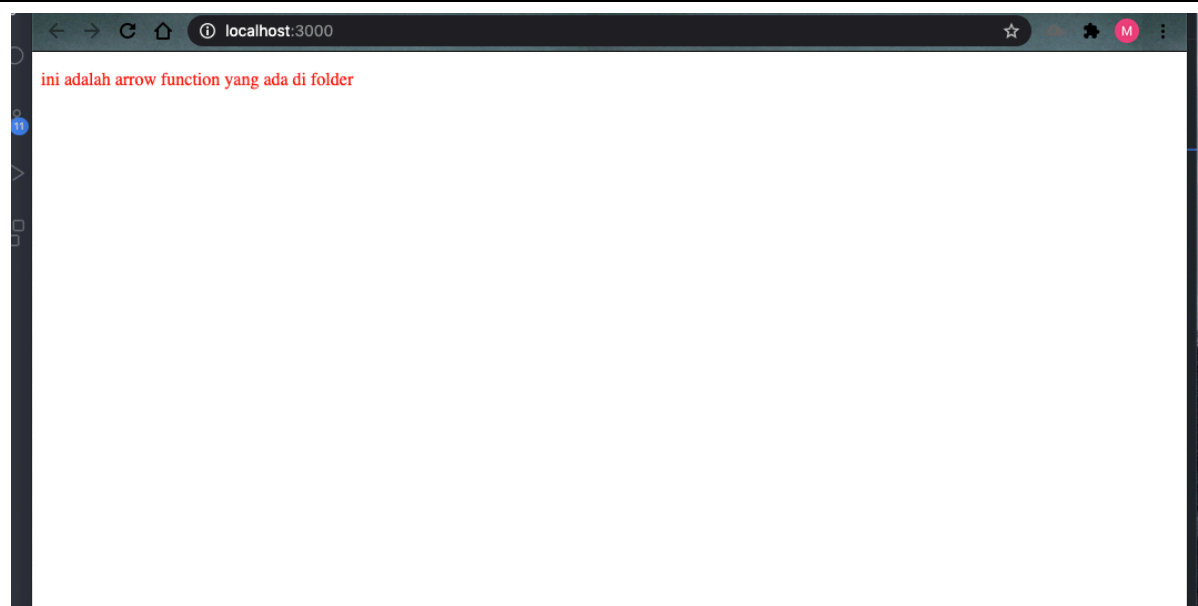
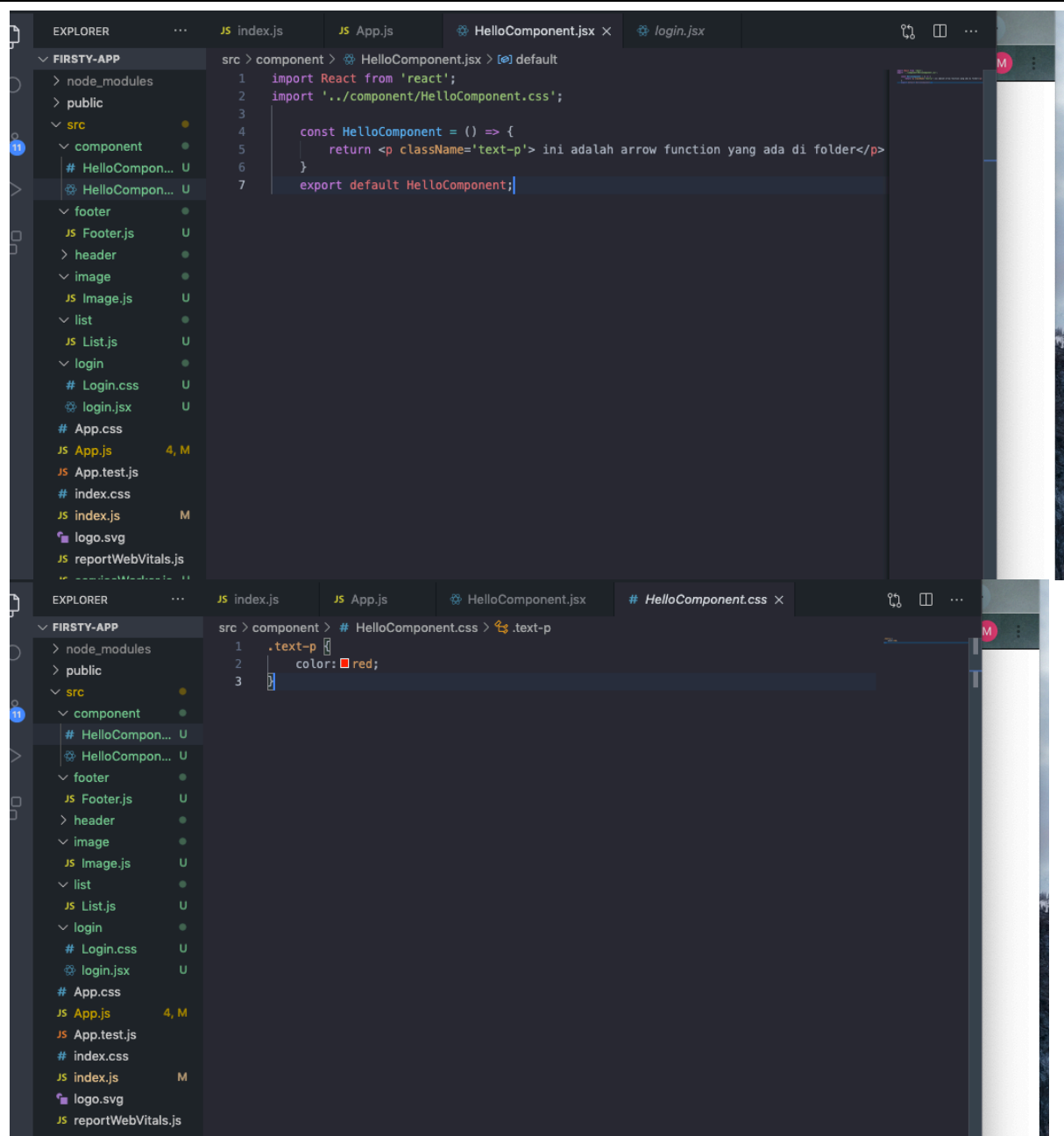
Pada setiap fase, components akan mengeksekusi fungsi atau method yang berbeda-beda yang di kenal sebagai lifecycle methods.

Buatlah Program dibawah ini dan deteksilah bagian dari lifecycle Component



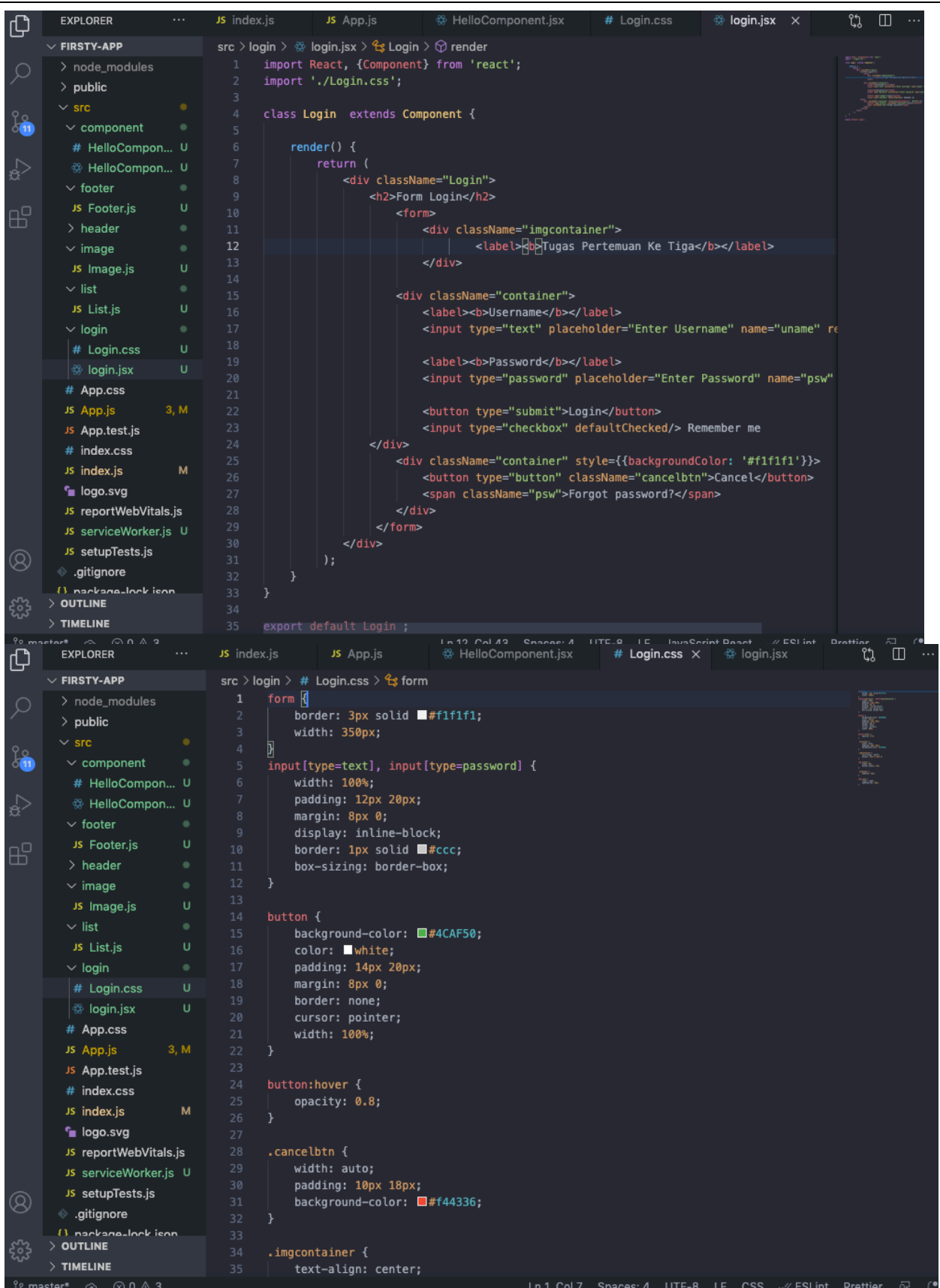
10

Selain kita dapat membuat file stateless component pada folder component, kita juga dapat menambahkan file css seperti pada gambar berikut (sebaiknya nama file css disamakan dengan nama file js nya dan tambahkan code seperti berikut)

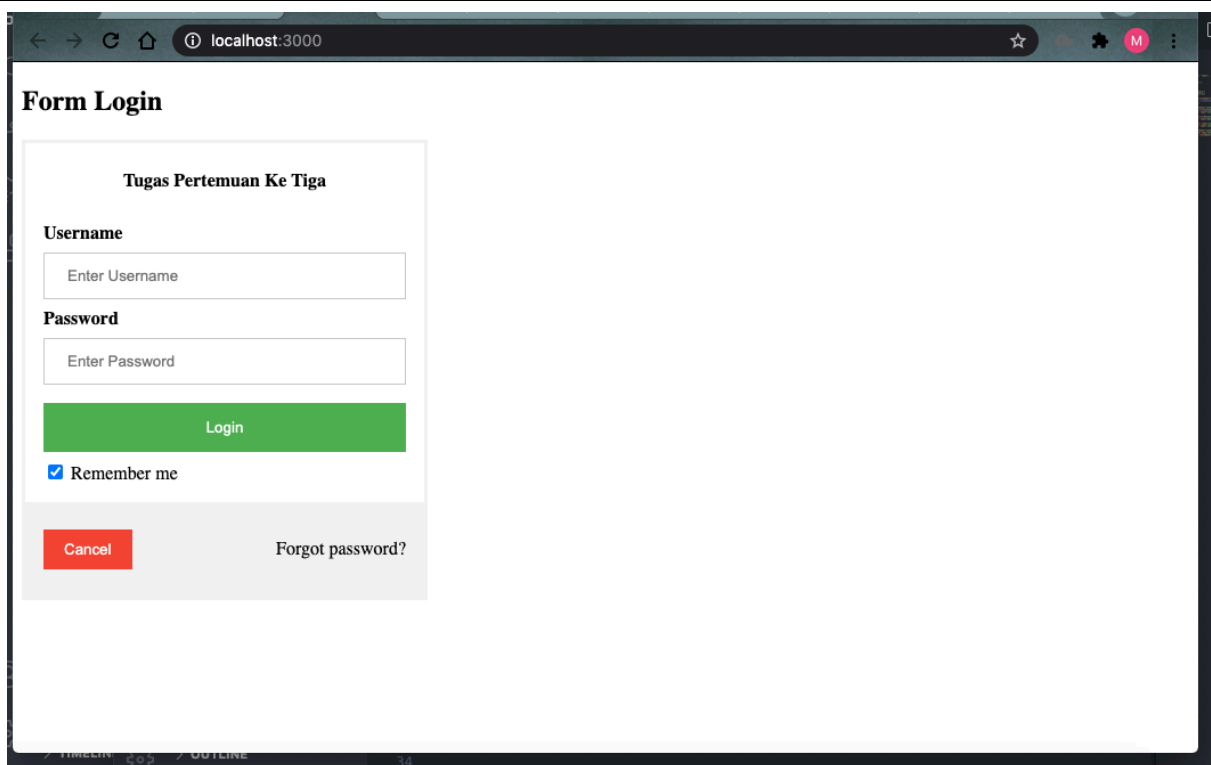


11

Buatlah menu login seperti pada gambar menggunakan CSS



```
src > JS App.js > App > render
1 // import logo from './logo.svg';
2 import './App.css';
3 import React, { Component } from 'react';
4 import './component/HelloComponent';
5 import Header from './header/Header';
6 import Footer from './footer/Footer';
7 import List from './list/List';
8 import Login from './login/login';
9
10 class App extends Component {
11
12   render() {
13     return (
14       <div>
15         <Login />
16       </div>
17     );
18   }
19 }
20 export default App;
21
```



-- Selamat Mengerjakan --