

M5 REACT JS FUNDAMENTAL

1. TUJUAN

CPMK: Mahasiswa mengetahui dan memahami dasar dari fundamental komponen react js.

Sub-CPMK:

- Mahasiswa mampu menggunakan dan membuat front end dengan menggunakan React JS
- Mahasiswa mampu menggunakan Rest API dari Java Spring dan diimplementasikan dalam front end React JS

2. DURASI WAKTU

1 pertemuan x 3 jam

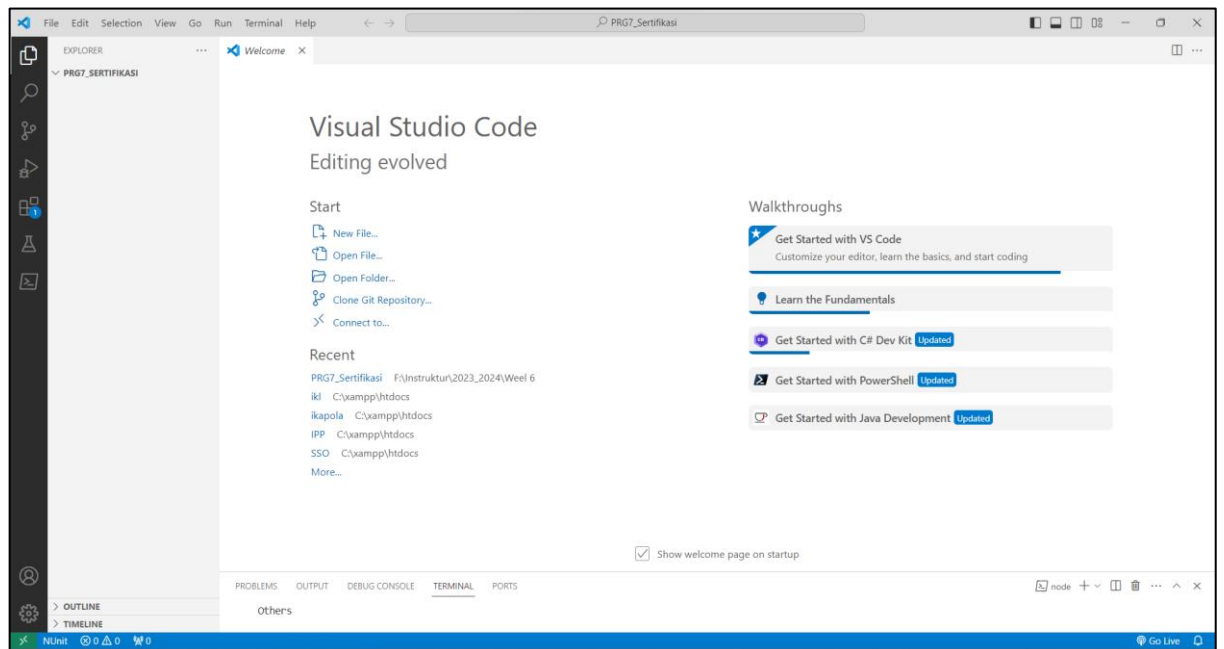
3. DASAR TEORI

React JS Fundamental

4. PERCOBAAN

A. Struktur Proyek React JS

- Download dan Install Node JS: <https://nodejs.org/en/download/>
- Buatlah sebuah folder baru didalam disk anda dengan nama PRG7_Sertifikasi
- Bukalah folder tersebut dalam Visual Studio Code



- Selanjutnya, bukalah terminal dalam Visual Studio Code, lalu ketikkan perintah: `npm create vite@latest sertifikasi-fe`

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS F:\Instruktur\2023_2024\Week 6\PRG7_Sertifikasi> npm create vite@latest sertifikasi-fe
? Select a framework: » - Use arrow-keys. Return to submit.
  Vanilla
  Vue
>  React
  Preact
  Lit
  Svelte
  Solid
  Qwik
  Others
```

5. Pilihlah framework React dan variant JavaScript

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

√ Select a framework: » React
√ Select a variant: » JavaScript

Scaffolding project in F:\Instruktur\2023_2024\Week 6\PRG7_Sertifikasi\sertifikasi-fe...

Done. Now run:

  cd sertifikasi-fe
  npm install
  npm run dev

PS F:\Instruktur\2023_2024\Week 6\PRG7_Sertifikasi> █
```

6. Selanjutnya, ketikkan perintah sesuai instruksi tersebut secara berurutan, yaitu:

- a. cd sertifikasi-fe
- b. npm install
- c. npm run dev

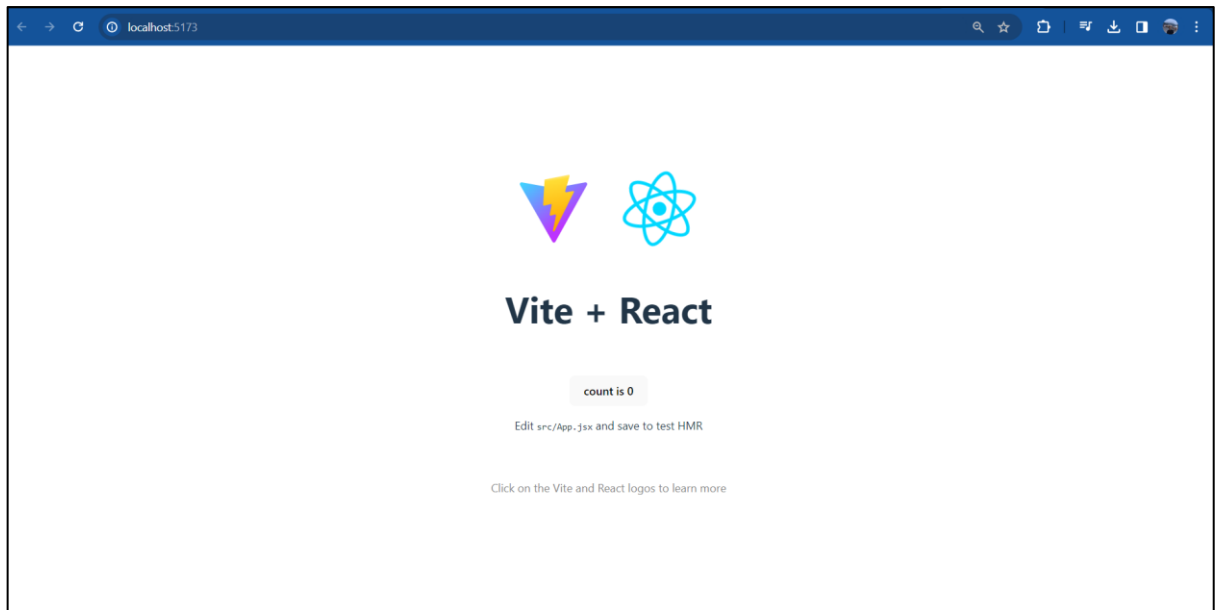
7. Apabila sudah, maka akan tampil di terminal seperti dibawah ini

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

VITE v5.2.6 ready in 220 ms

→ Local:  http://localhost:5173/
→ Network: use --host to expose
→ press h + enter to show help
█
```

8. Coba bukalah link yang tampil, maka akan mengakses halaman seperti dibawah ini



9. Kita dapat mengatur port yang digunakan didalam file vite.config.js

```
JS vite.config.js X
sertifikasi-fe > JS vite.config.js > [default]
1  import { defineConfig } from 'vite'
2  import react from '@vitejs/plugin-react'
3
4  // https://vitejs.dev/config/
5  export default defineConfig({
6    plugins: [react()],
7
8    server: {
9      port: 1234
10   }
11 })
12
```

10. Dalam terminal juga akan otomatis terganti menjadi <http://localhost:1234/>.

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
10:54:23 [vite] vite.config.js changed, restarting server...
10:54:23 [vite] server restarted.

→ Local:   http://localhost:1234/
→ Network: use --host to expose
```

11. Kita akan mencoba untuk memahami struktur direktori didalam react js
12. Bukalah index.html, ini merupakan file dasar dalam mengakses halaman utama

```

1  <!doctype html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <link rel="icon" type="image/svg+xml" href="/vite.svg" />
6      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7      <title>Vite + React</title>
8    </head>
9    <body>
10     <div id="root"></div>
11     <script type="module" src="/src/main.jsx"></script>
12   </body>
13 </html>

```

13. Dalam line 10, terdapat tag div dengan id root, dan di line 11, terdapat script yang mengakses main.jsx.

14. Bukalah main.jsx

```

sertifikasi-fe > src > main.jsx
1  import React from 'react'
2  import ReactDOM from 'react-dom/client'
3  import App from './App.jsx'
4  import './index.css'
5
6  ReactDOM.createRoot(document.getElementById('root')).render(
7    <React.StrictMode>
8      <App />
9    </React.StrictMode>,
10  )
11

```

15. Dalam main.jsx, dalam line 6 terdapat fungsi yang mengambil id root dimana akan mengembalikan sebuah komponen dengan nama App. App ini berasal dari file App.jsx.

16. Bukalah file App.jsx

```

sertifikasi-fe > src > App.jsx > App
1  import { useState } from 'react'
2  import reactLogo from './assets/react.svg'
3  import viteLogo from '/vite.svg'
4  import './App.css'
5
6  function App() {
7    const [count, setCount] = useState(0)
8
9    return (
10     <>
11       <div>
12         <a href="https://vitejs.dev" target="_blank">
13           <img src={viteLogo} className="logo" alt="Vite logo" />
14         </a>
15         <a href="https://react.dev" target="_blank">
16           <img src={reactLogo} className="logo react" alt="React logo" />
17         </a>
18       </div>
19       <h1>Vite + React</h1>
20       <div className="card">
21         <button onClick={() => setCount((count) => count + 1)}>
22           count is {count}
23         </button>
24         <p>
25           Edit <code>src/App.jsx</code> and save to test HMR
26         </p>
27       </div>
28       <p className="read-the-docs">
29         Click on the Vite and React logos to learn more
30       </p>
31     </>
32   )
33 }
34
35 export default App

```

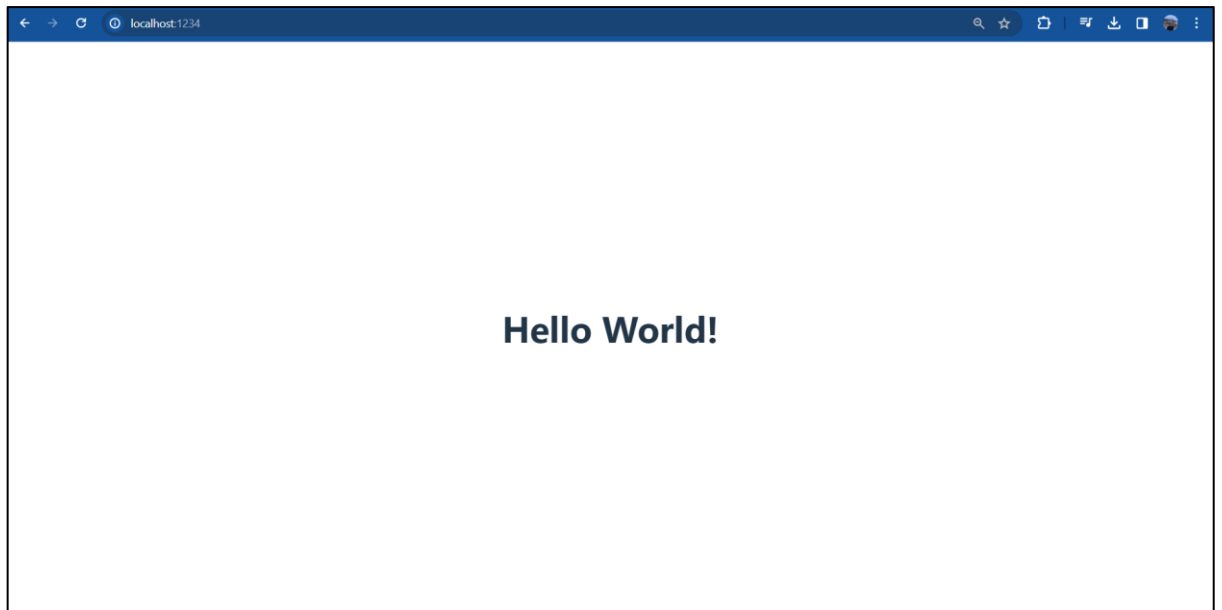
17. File ini merupakan file dasar yang ditampilkan dari halaman index tadi
18. Selanjutnya, kita akan mengubahnya menjadi halaman yang menampilkan Hello World! Bukalah App.jsx, modifikasi hingga menjadi seperti dibawah ini

```

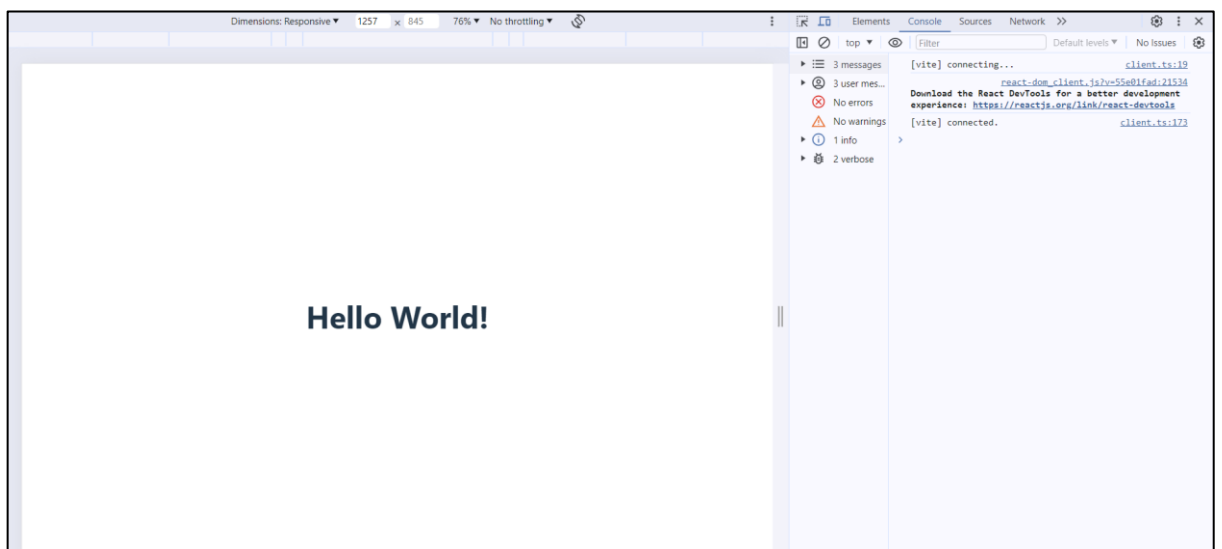
JS vite.config.js  main.jsx  App.jsx  X
sertifikasi-fe > src > App.jsx > ...
1  import './App.css'
2
3  function App() {
4
5    return (
6      <>
7        <h1>Hello World!</h1>
8      </>
9    )
10 }
11
12 export default App
13

```

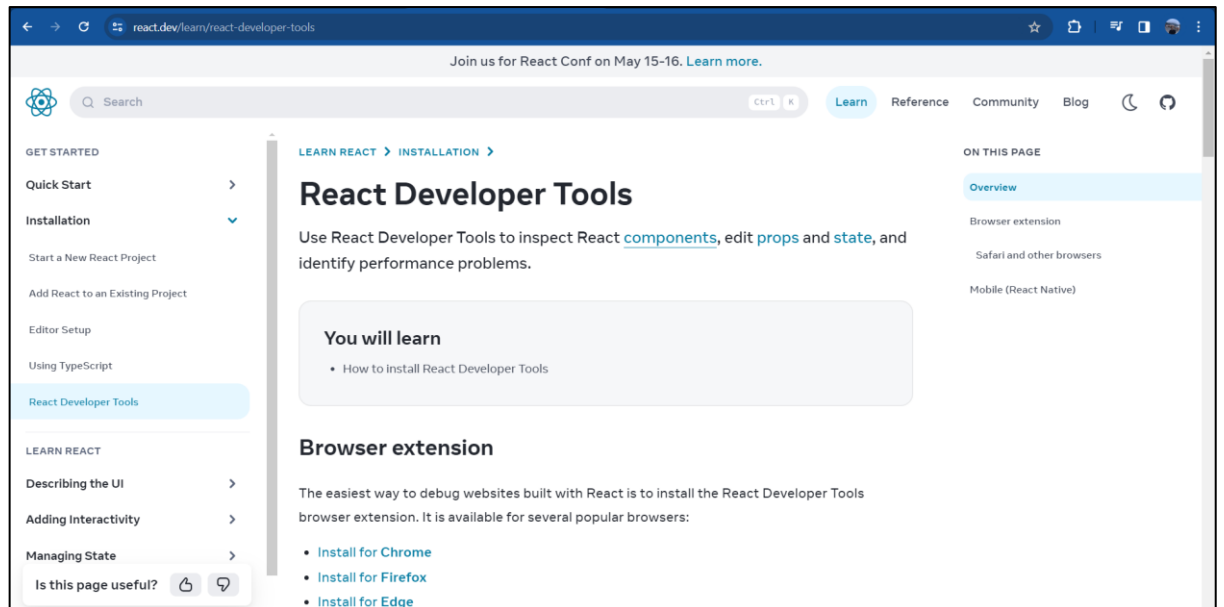
19. Selanjutnya apabila kita buka, maka akan tampil seperti dibawah ini



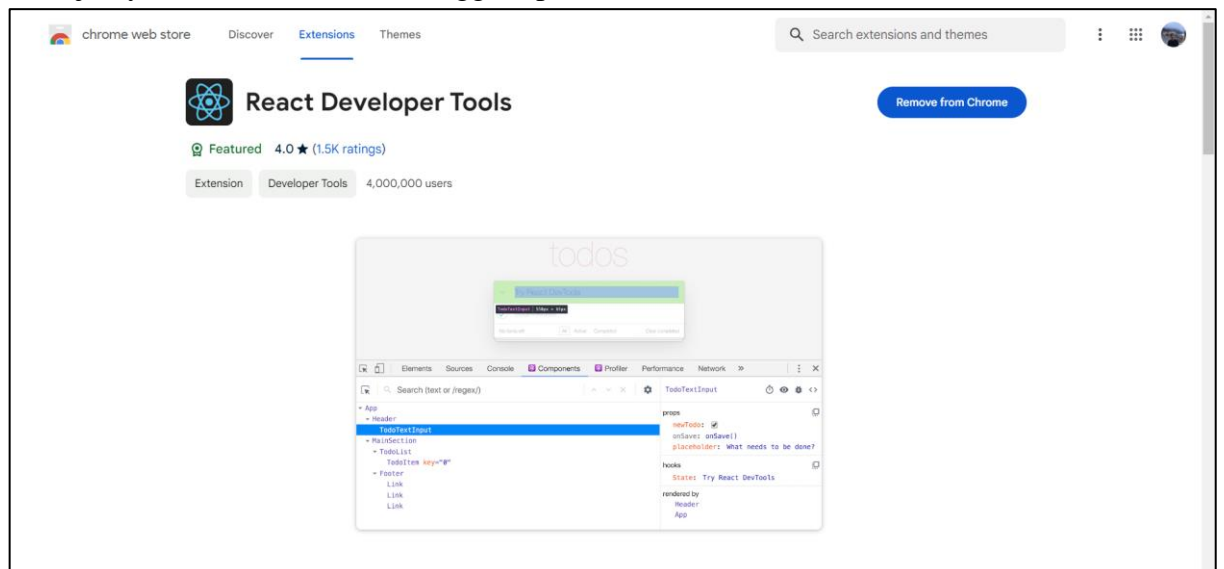
20. Bukalah console di dalam chrome, disini kita akan menemukan pesan seperti dibawah ini



21. Klik link tersebut maka akan tampil seperti dibawah ini, lalu install dalam browser anda

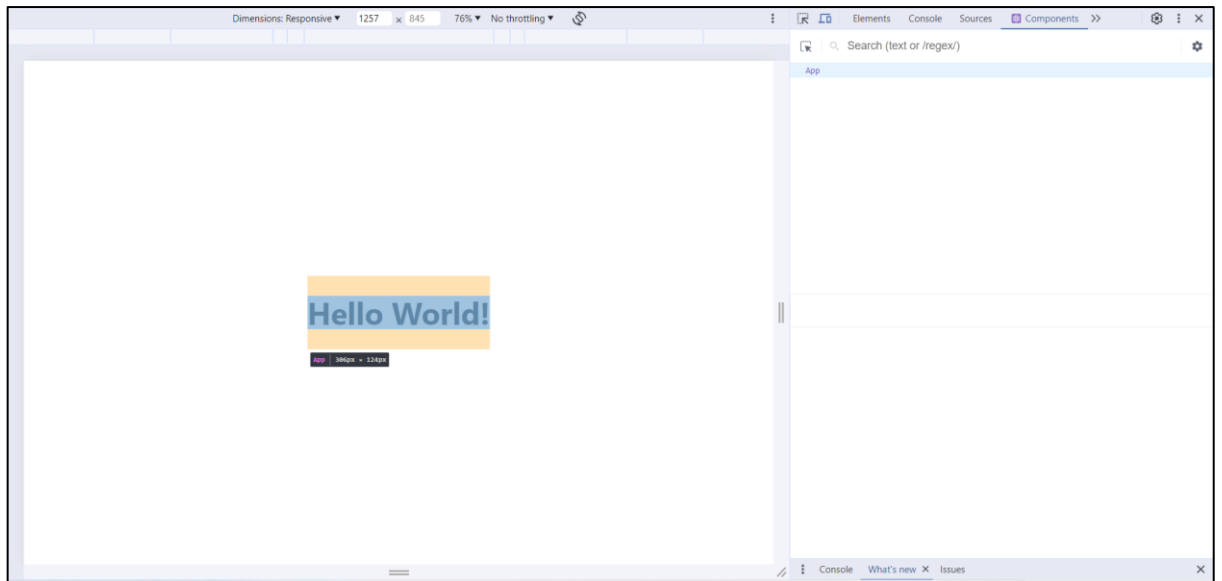


22. Selanjutnya klik add to chrome hingga seperti dibawah ini



23. Apabila kita refresh, maka pesan tadi tidak akan muncul kembali

24. Extension ini berfungsi ketika kita inspect, kita dapat menggunakan tab component, dimana disini apabila kita arahkan ke komponen App, maka akan mengarahkan ke Hello World!



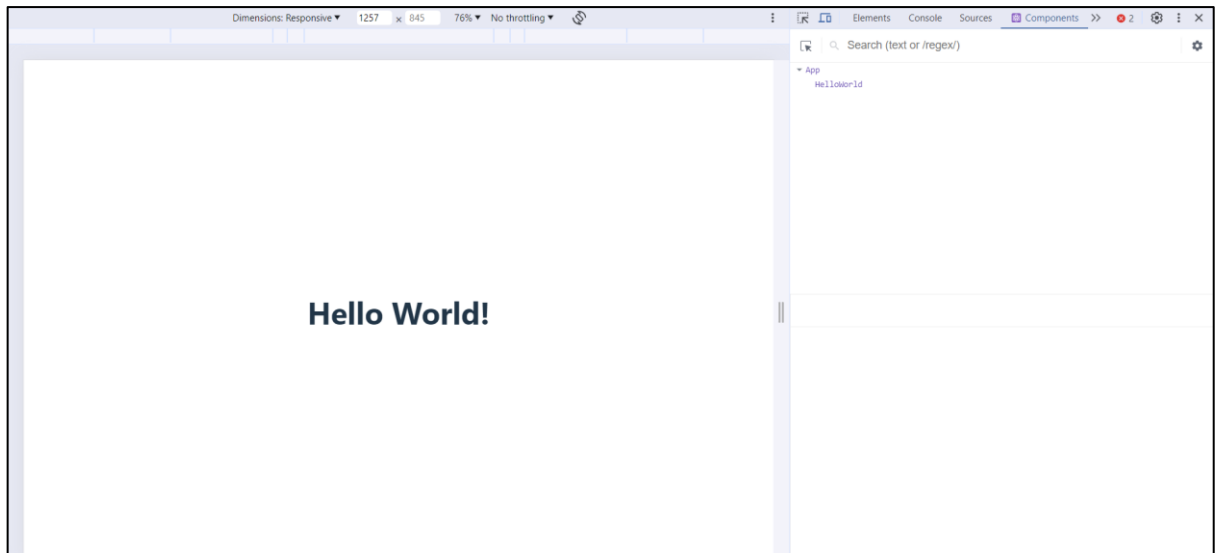
25. Selanjutnya, kita akan membuat komponen baru yang akan dipanggil dalam komponen App.jsx. Buatlah HelloWorld.jsx dalam package src

```
JS vite.config.js  main.jsx  App.jsx  HelloWorld.jsx X
sertifikasi-fe > src > HelloWorld.jsx > ...
1  function HelloWorld() {
2    |    return <h1>Hello World!</h1>
3  }
4
5  export default HelloWorld
```

26. Selanjutnya, bukalah App.jsx, panggilah komponen HelloWorld

```
JS vite.config.js  main.jsx  App.jsx  X  HelloWorld.jsx
sertifikasi-fe > src > App.jsx > App
1  import './App.css'
2  import HelloWorld from './HelloWorld'
3
4  function App() {
5
6    return (
7      <>
8      |   <HelloWorld />
9      </>
10   )
11 }
12
13 export default App
14
```

27. Bukalah browser anda, lalu buka tab components maka akan terdapat 2 komponen yaitu App dan HelloWorld



28. Selanjutnya, kita akan membuat komponen baru untuk menampilkan List User secara statis. Buatlah package baru dengan nama components, didalamnya buatlah package baru dengan nama users.
29. Ketikkan rfce, lalu enter maka akan otomatis mengetikkan kode seperti dibawah ini

```
JS vite.config.js  main.jsx  App.jsx  ListUserComponent.jsx 1 X  HelloWo
sertifikasi-fe > src > components > users > ListUserComponent.jsx > ...
1  import React from 'react'
2
3  function ListUserComponent() {
4    return (
5      <div>ListUserComponent</div>
6    )
7  }
8
9  export default ListUserComponent
```

30. Ini merupakan cara cepat yang disediakan oleh ekstensi ES7+ React/Redux/React-Native snippets (Download jika belum ada)
31. Selanjutnya, modifikasi kode tersebut dengan mendeklasikan sebuah array dan menampilkannya

```

1  import React from 'react'
2
3  function ListUserComponent() {
4      const dummyData = [
5          {
6              "id": 1,
7              "nama": "Roni Prasetyo",
8              "username": "roni.prasetyo",
9              "email": "roniprsty@gmail.com",
10             "role": "Admin"
11         },
12         {
13             "id": 2,
14             "nama": "Elora Manuella",
15             "username": "elora.m",
16             "email": "eloramanuella@gmail.com",
17             "role": "Admin"
18         }
19     ]
20
21     return (
22         <div>
23             <h2>List of User</h2>
24             <table>
25                 <thead>
26                     <tr>
27                         <th>Name</th>
28                         <th>Username</th>
29                         <th>Email</th>
30                         <th>Role</th>
31                     </tr>
32                 </thead>
33                 <tbody>
34                     {
35                         dummyData.map(user =>
36                             <tr key={user.id}>
37                                 <td>{user.nama}</td>
38                                 <td>{user.username}</td>
39                                 <td>{user.email}</td>
40                                 <td>{user.role}</td>
41                             </tr>
42                         )
43                     }
44                 </tbody>
45             </table>
46         </div>
47     )
48 }
49 export default ListUserComponent

```

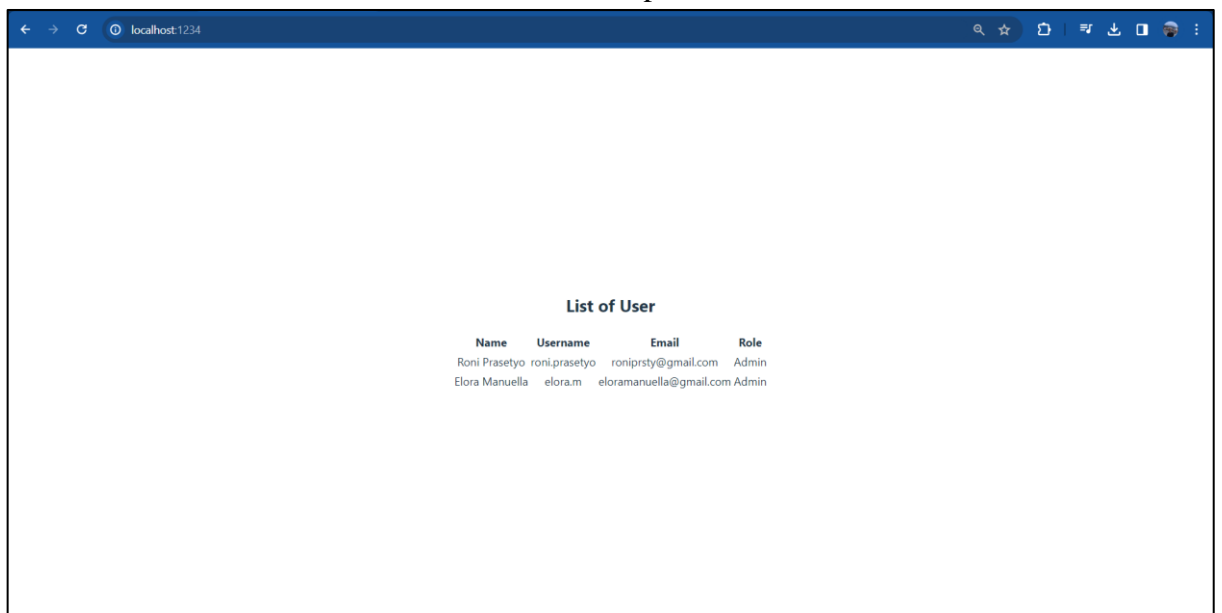
32. Selanjutnya, panggilah component ini didalam komponen App.jsx

```

sertifikasi-fe > src > App.jsx > ...
1  import './App.css'
2  import ListUserComponent from './components/users/ListUserComponent'
3
4  function App() {
5
6      return (
7          <>
8              <ListUserComponent />
9          </>
10     )
11 }
12
13 export default App
14

```

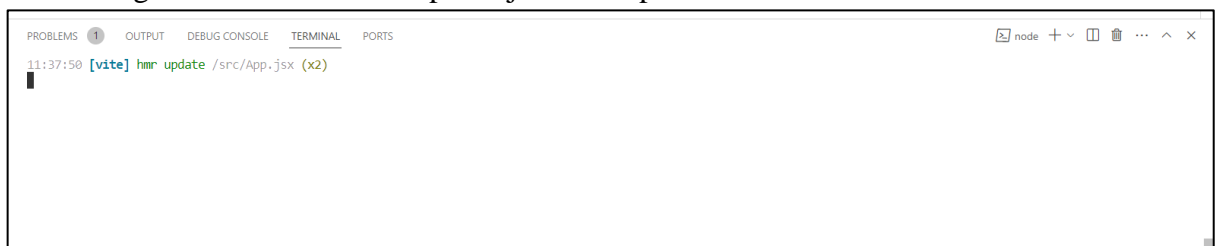
33. Bukalah halaman web tadi, maka akan berubah seperti dibawah ini



The screenshot shows a web browser window with the address bar set to localhost:1234. The main content area displays a table titled "List of User". The table has four columns: Name, Username, Email, and Role. There are two rows of data.

Name	Username	Email	Role
Roni Prasetyo	roni.prasetyo	roniprsty@gmail.com	Admin
Elora Manuella	elora.m	eloramanuella@gmail.com	Admin

34. Selanjutnya, kita akan menginstall bootstrap dengan menggunakan npm, klik icon new terminal agar terminal utama tetap menjalankan aplikasi



The screenshot shows a VS Code terminal window with the following output:

```

11:37:50 [vite] hmr update /src/App.jsx (x2)

```

35. Lalu ketikkan perintah untuk masuk ke folder sertifikasi-fe, lalu perintah install bootstrap

```

PS F:\Instruktur\2023_2024\Week 6\PRG7_Sertifikasi> cd sertifikasi-fe
PS F:\Instruktur\2023_2024\Week 6\PRG7_Sertifikasi\sertifikasi-fe> npm install bootstrap --save

```

36. Bukalah folder node_modules, disini kita akan menemukan folder bootstrap. Begitu juga ketika kita membuka package-lock.json, kita akan menemukan libray bootstrap yang terinstall.

37. Untuk menggunakannya, hapus terlebih dahulu line yang memanggil css dari index.css maupun app.css didalam file main.jsx dan App.jsx. Didalam main.jsx, tambahkan kode untuk memanggil bootstrap

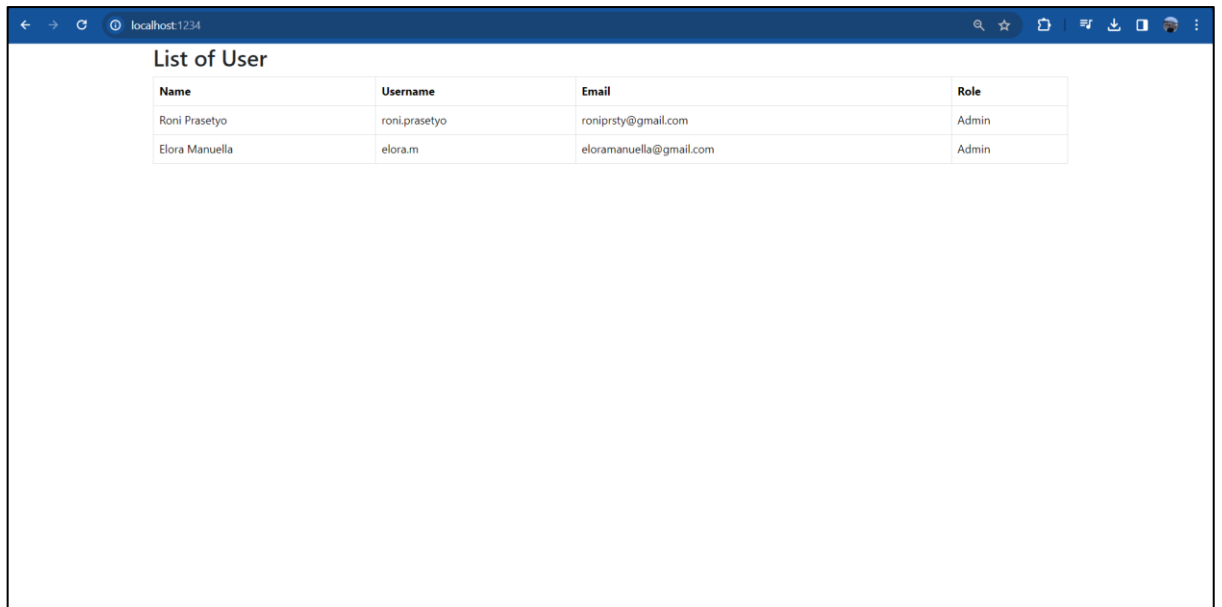
```
sertifikasi-fe > src > App.jsx > ...
1  import ListUserComponent from './components/users/ListUserComponent'
2  |
3  function App() {
4  |
5  |   return (
6  |     <>
7  |       <ListUserComponent />
8  |     </>
9  |   )
10 | }
11
12 export default App
13
```

```
sertifikasi-fe > src > main.jsx
1  import React from 'react'
2  import ReactDOM from 'react-dom/client'
3  import App from './App.jsx'
4  import 'bootstrap/dist/css/bootstrap.min.css'
5
6  ReactDOM.createRoot(document.getElementById('root')).render(
7    <React.StrictMode>
8    |   <App />
9    | </React.StrictMode>,
10 | )
11
```

38. Selanjutnya, buka ListUserComponent, tambahkan atribut className untuk memanggil bootstrap

```
21  return (
22  |    <div className='container'>
23  |      <h2>List of User</h2>
24  |      <table className='table table-bordered'>
25  |        <thead>
```

39. Bukalah web browser anda, maka akan tampil seperti dibawah ini



The screenshot shows a web browser window with the address bar set to localhost:1234. The page title is 'List of User'. Below the title is a table with four columns: Name, Username, Email, and Role. The table contains two rows of data.

Name	Username	Email	Role
Roni Prasetyo	roni.prasetyo	roniprsty@gmail.com	Admin
Elora Manuella	elora.m	eloramanuella@gmail.com	Admin

40. Selanjutnya, kita akan mengambil data dari Rest API yang telah kita buat. Ketikkan kode dibawah ini didalam terminal

```
PS F:\Instruktur\2023_2024\Week 6\PRG7_Sertifikasi\sertifikasi-fe> npm install axios --save
added 9 packages, and audited 291 packages in 2s

106 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
PS F:\Instruktur\2023_2024\Week 6\PRG7_Sertifikasi\sertifikasi-fe> █
```

41. Selanjutnya, buatlah folder baru dengan nama service, didalamnya buatlah file baru dengan nama UserService.js

```
sertifikasi-fe > src > services > JS UserService.js > ...
1  import axios from "axios";
2
3  const REST_API_BASE_URL = 'http://localhost:8080/users';
4
5  export const listUsers = () => axios.get(REST_API_BASE_URL + '/getUsers');
6
```

42. Selanjutnya, bukalah ListUserComponent.jsx, modifikasi kodenya seperti dibawah ini

```

sertifikasi-fe > src > components > users > ListUserComponent.jsx > ListUserComponent > useEffect() callback > catch() callback
1  import React, { useEffect, useState } from 'react';
2  import { listUsers } from '../services/UserService';
3
4  function ListUserComponent() {
5      const [users, setUsers] = useState([])
6
7      useEffect(() => {
8          listUsers()
9              .then(response => {
10                 setUsers(response.data.data);
11             })
12             .catch(error => {
13                 console.error(error);
14             });
15      }, []);
16
17      return (
18          <div className='container'>
19              <h2>List of Users</h2>
20              <table className='table table-bordered'>
21                  <thead>
22                      <tr>
23                          <th>Name</th>
24                          <th>Username</th>
25                          <th>Email</th>
26                          <th>Role</th>
27                      </tr>
28                  </thead>
29                  <tbody>
30                      {
31                          users.map(user =>
32                              <tr key={user.id}>
33                                  <td>{user.nama}</td>
34                                  <td>{user.username}</td>
35                                  <td>{user.email}</td>
36                                  <td>{user.role}</td>
37                              </tr>
38                          )
39                      }
40                  </tbody>
41              </table>
42          </div>
43      );
44  }
45
46  export default ListUserComponent;

```

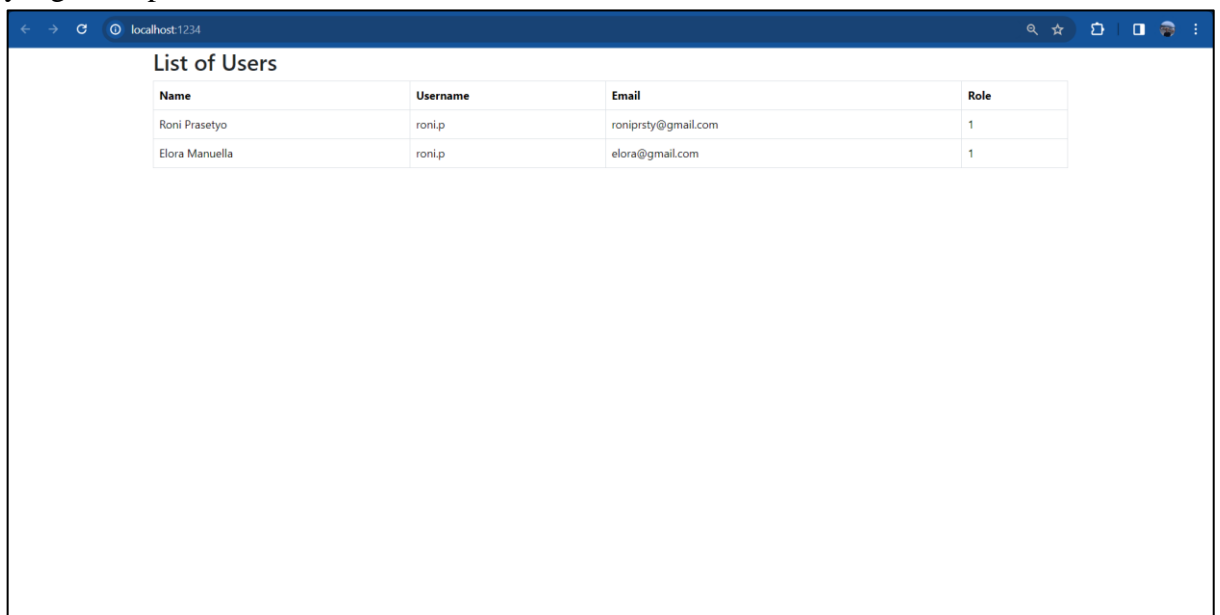
43. Bukalah IntelliJ IDEA anda, lalu tambahkan @CrossOrigin dalam setiap Rest nya sebagai perizinan digunakan di React JS nantinya.

```

1 package id.co.prg7_sertifikasi.rest;
2
3 import id.co.prg7_sertifikasi.model.User;
4 import id.co.prg7_sertifikasi.response.DtoResponse;
5 import id.co.prg7_sertifikasi.service.UserService;
6 import org.springframework.beans.factory.annotation.Autowired;
7 import org.springframework.web.bind.annotation.*;
8
9 @CrossOrigin
10 @RestController
11 @RequestMapping("/users")
12 public class UserRest {

```

44. Selanjutnya, jalankan projek anda, lalu buka web front end maka akan tampil data yang dilempar oleh rest kita



The screenshot shows a web browser at localhost:1234 displaying a table titled 'List of Users'. The table has four columns: Name, Username, Email, and Role. It contains two rows of data.

Name	Username	Email	Role
Roni Prasetyo	roni.p	roniprsty@gmail.com	1
Elora Manuella	roni.p	elora@gmail.com	1

45. Selanjutnya, kita akan membuat header dari aplikasi ini, buatlah sebuah package dengan nama templates di folder components, lalu buatlah file HeaderComponent.jsx.

```

sertifikasi-fe > src > components > templates > HeaderComponent.jsx > HeaderComponent
1 import React from 'react'
2
3 const HeaderComponent = () => {
4   return (
5     <div>
6       <header>
7         <nav className='navbar navbar-dark bg-dark'>
8           <a className='navbar-brand'>ASTRatech Certification</a>
9         </nav>
10      </header>
11    </div>
12  )
13 }
14
15 export default HeaderComponent

```

46. Selanjutnya buat juga FooterComponent.jsx

```
sertifikasi-fe > src > components > templates > FooterComponent.jsx > ...
1  import React from 'react'
2
3  const FooterComponent = () => {
4    return (
5      <div>
6        <footer className='footer'>
7          <span>All rights reserved 2024 by Tim Pengajar PRG7</span>
8        </footer>
9      </div>
10   )
11 }
12
13 export default FooterComponent
```

47. Dalam hal ini, diperlukan css dengan class footer, oleh karena itu, bukalah App.css, hapus semua css didalamnya, tambahkan css untuk selector class footer

```
sertifikasi-fe > src > # App.css > .footer
1  .footer {
2    position: absolute;
3    bottom: 0;
4    width: 100%;
5    height: 50px;
6    background-color: black;
7    text-align: center;
8    color: white;
9  }
```

48. Import App.css kedalam App.jsx, dan juga panggil HeaderComponent serta FooterComponent


```
sertifikasi-fe > src > App.jsx > App
1  import FooterComponent from './components/templates/FooterComponent'
2  import HeaderComponent from './components/templates/HeaderComponent'
3  import ListUserComponent from './components/users/ListUserComponent'
4  import './App.css'
5
6  function App() {
7
8      return (
9          <>
10             <HeaderComponent />
11             <ListUserComponent />
12             <FooterComponent />
13          </>
14      )
15  }
16
17  export default App
18
```

49. Bukalah web browser anda, maka akan tampil seperti dibawah ini

ASTRAtech Certification

List of Users

Name	Username	Email	Role
Roni Prasetyo	roni.p	roniprsty@gmail.com	1
Elora Manuella	roni.p	elora@gmail.com	1

All rights reserved 2024 by Tim Pengajar PRG7

50. Selanjutnya, kita akan mencoba terkait routing di React, tambahkan library baru dengan menggunakan npm

```
PS F:\Instruktur\2023_2024\Week 6\PRG7_Sertifikasi\sertifikasi-fe> npm install react-router-dom --save
[.....] \ idealTree:sertifikasi-f
```

51. Tambahkan UserComponent.jsx dalam folder employees

```

sertifikasi-fe > src > components > users > UserComponent.jsx > default
1  import React from 'react'
2
3  const UserComponent = () => {
4    return (
5      <div>UserComponent</div>
6    )
7  }
8
9  export default UserComponent

```

52. Selanjutnya, bukalah ListUserComponent, modifikasi kodenya tambahkan useNavigate, const navigator, fungsi addNewUser serta button Add User

```

1  import React, { useEffect, useState } from 'react';
2  import { listUsers } from '../services/UserService';
3  import { useNavigate } from 'react-router-dom';
4
5  function ListUserComponent() {
6    const [users, setUsers] = useState([])
7    const navigator = useNavigate();
8
9    useEffect(() => {
10      listUsers()
11        .then(response => {
12          setUsers(response.data.data);
13        })
14        .catch(error => {
15          console.error(error);
16        });
17    }, []);
18
19    function addNewUser() {
20      navigator('/add-user')
21    }
22
23    return (
24      <div className='container'>
25        <h2 className='text-center'>List of Users</h2>
26        <button className='btn btn-primary mb-2' onClick={addNewUser}>Add User</button>
27        <table className='table table-bordered'>
28          <thead>
29            <tr>
30              <th>Name</th>
31              <th>Username</th>

```

53. Selanjutnya, kita akan memberikan routing dalam App.jsx, modifikasi kodenya hingga menjadi seperti dibawah ini

```

1  import FooterComponent from './components/templates/FooterComponent'
2  import HeaderComponent from './components/templates/HeaderComponent'
3  import ListUserComponent from './components/users/ListUserComponent'
4  import './App.css'
5  import { Routes, Route, BrowserRouter } from 'react-router-dom'
6  import UserComponent from './components/users/UserComponent'
7
8  function App() {
9
10     return (
11         <>
12         <BrowserRouter>
13             <HeaderComponent />
14             <Routes>
15                 <Route path="/" element= { <ListUserComponent />}></Route>
16                 <Route path="/users" element = { <ListUserComponent />}></Route>
17                 <Route path="/add-user" element = { <UserComponent />}></Route>
18             </Routes>
19             <FooterComponent />
20         </BrowserRouter>
21
22         </>
23     )
24 }
25
26 export default App

```

54. Selanjutnya, kita coba akses di halaman web, lalu klik add user, maka halaman akan berpindah ke add-user
55. Bukalah kembali UserService.js, tambahkan pemanggilan API untuk create

```

sertifikasi-fe > src > services > JS UserService.js > ...
1  import axios from "axios";
2
3  const REST_API_BASE_URL = 'http://localhost:8080/users';
4
5  export const listUsers = () => axios.get(REST_API_BASE_URL + '/getUsers');
6
7  export const createUser = (user) => axios.post(REST_API_BASE_URL + '/saveUser', user);

```

56. Kita akan membuat halaman create, bukalah UserComponent, lalu buatlah kode seperti dibawah ini

sertifikasi-fe > src > components > users > UserComponent.jsx > UserComponent

```
1  import React, { useState } from 'react';
2  import { createUser } from '../services/UserService';
3  import { useNavigate } from 'react-router-dom';
4
5  const UserComponent = () => {
6    const [usr_nama, setUsr_nama] = useState('');
7    const [usr_email, setUsr_email] = useState('');
8    const [usr_username, setUsr_username] = useState('');
9    const [usr_password, setUsr_password] = useState('');
10   const [usr_role, setUsr_role] = useState('');
11   const [pro_id, setPro_id] = useState('');
12
13   const navigate = useNavigate();
14
15   function saveUser(e) {
16     e.preventDefault();
17     const user = { usr_nama, usr_email, usr_username, usr_password, usr_role, pro_id };
18
19     createUser(user).then((response) => {
20       navigate('/users');
21     });
22   }
23
24   return (
25     <div className='container'>
26       <br />
27       <div className='row'>
28         <div className='card'>
29           <h2 className='text-center'>Add User</h2>
30           <div className='card-body'>
31             <form>
32               <div className='form-group mb-2'>
33                 <label className='form-label'>Nama :</label>
34                 <input type='text' name='usr_nama' value={usr_nama} className='form-control'
35                   onChange={(e) => setUsr_nama(e.target.value)} />
36                 <label className='form-label'>Email :</label>
37                 <input type='email' name='usr_email' value={usr_email} className='form-control'
38                   onChange={(e) => setUsr_email(e.target.value)} />
39                 <label className='form-label'>Username :</label>
40                 <input type='text' name='usr_username' value={usr_username} className='form-control'
41                   onChange={(e) => setUsr_username(e.target.value)} />
42                 <label className='form-label'>Password :</label>
43                 <input type='password' name='usr_password' value={usr_password} className='form-control'
44                   onChange={(e) => setUsr_password(e.target.value)} />
45                 <label className='form-label'>Role :</label>
46                 <input type='text' name='usr_role' value={usr_role} className='form-control'
47                   onChange={(e) => setUsr_role(e.target.value)} />
48                 <label className='form-label'>Prodi :</label>
49                 <input type='text' name='pro_id' value={pro_id} className='form-control'
50                   onChange={(e) => setPro_id(e.target.value)} />
51               </div>
52               <button className='btn btn-success' onClick={saveUser}>Submit</button>
53             </form>
54           </div>
55         </div>
56       </div>
57     </div>
58   );
59 };
60
61 export default UserComponent;
62
```

57. Cobalah untuk menjalankan aplikasinya, tambahkan 1 data dengan pro_id yang masih menggunakan angka

ASTRAtech Certification

Add User

Nama :
Sisia Dika

Email :
sisia@gmail.com

Username :
sisia.dika

Password :

Role :
1

Prodi :
1

AI rights reversed 2024 by Tim Pengajar PRG7

58. Selanjutnya, klik Submit maka data akan tersimpan

ASTRAtech Certification

List of Users

Name	Username	Email	Role
Roni Prasetyo	roni.p	roniprsty@gmail.com	1
Elora Manuella	roni.p	elora@gmail.com	1
Sisia Dika	sisia.dika	sisia@gmail.com	1

AI rights reversed 2024 by Tim Pengajar PRG7

5. LATIHAN

1. Pada saat menambahkan data, usr_status akan terisi Null, otomatisasikan agar usr_status otomatis 1, usr_creaby otomatis 0, dan usr_creadate otomatis tanggal saat ini
2. Dalam menambahkan data, role masih menggunakan input textbox, ubahlah agar menggunakan combobox statis, dimana berisi Admin dan Participant
3. Dalam menambahkan data, prodi masih menggunakan input textbox, ubahlah menjadi menggunakan combobox dinamis, yang mengambil data dari database