

M6 REACT JS CONSUME API

1. TUJUAN

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

Sub-CPMK:

- a. Mahasiswa mampu menggunakan dan membuat front end dengan menggunakan React JS
- b. Mahasiswa mampu menggunakan Rest API yang mengimplementasikan dalam front end React JS

2. DURASI WAKTU

1 pertemuan x 3 jam

3. DASAR TEORI

React JS

4. PERCOBAAN

A. Struktur Projek React JS

1. Download dan Install Node JS: <https://nodejs.org/en/download/>
2. Buatlah sebuah folder baru didalam disk anda dengan nama PRG4_Penjualan_Produk
3. Buka folder tersebut dalam Visual Studio Code
4. Selanjutnya, buka terminal dalam Visual Studio Code, lalu ketikkan perintah: *npm create vite@latest penjualan-fe*

```
PS F:\Instruktur\Pemrograman 4\PRG4_Penjualan_Produk> npm create vite@latest penjualan-fe
  Select a framework: » React
  Select a variant: » JavaScript
```

```
Scaffolding project in F:\Instruktur\Pemrograman 4\PRG4_Penjualan_Produk\penjualan-fe...
```

```
Done. Now run:
```

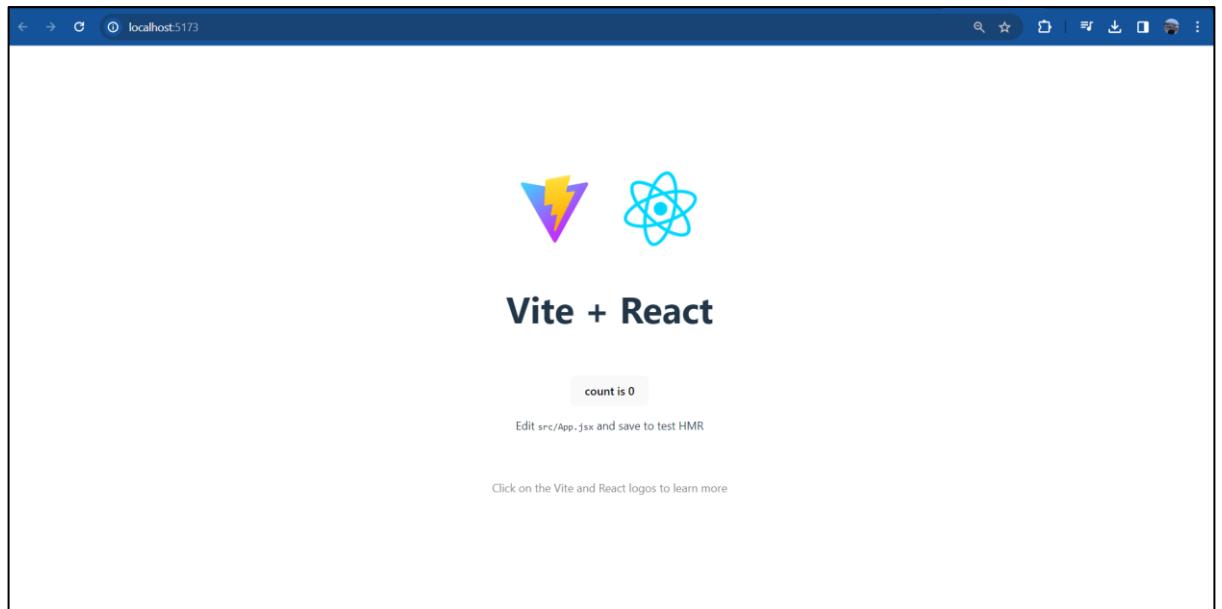
```
cd penjualan-fe
npm install
npm run dev
```

5. Pilihlah framework React dan variant JavaScript
6. Selanjutnya, ketikkan perintah sesuai instruksi tersebut secara berurutan, yaitu:
 - a. cd penjualan-fe
 - b. npm install
 - c. npm run dev
7. Apabila sudah, maka akan tampil di terminal seperti dibawah ini

```
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. Apabila dibutuhkan, kita dapat mengatur port yang digunakan didalam file vite.config.js

```
vite.config.js X
sertifikasi-fe > vite.config.js > [e] 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

```

- Selanjutnya, kita akan membuat komponen baru untuk menampilkan List Produk secara statis. Buatlah **package baru** dengan nama **components**, didalamnya buatlah **package** baru dengan nama **produk**. Lalu didalamnya buatlah ListProdukComponent.jsx.



- Ketikkan **rfce**, lalu enter maka akan otomatis mengetikkan kode seperti dibawah ini

```

penjualan-fe > src > component > produk > ListProdukComponent.jsx
1 import React from 'react'
2
3 function ListProdukComponent() {
4   return (
5     <div>ListProdukComponent</div>
6   )
7 }
8
9 export default ListProdukComponent

```

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

```

1 import React from 'react';
2
3 function ListProdukComponent() {
4     const dummyData = [
5         {
6             "id": "1",
7             "nama_produk": "Laptop",
8             "jenis_produk": "Infra",
9             "stok": "122",
10            "harga_beli": "20000",
11            "harga_jual": "30000",
12            "status": "Tersedia"
13        },
14        {
15            "id": "2",
16            "nama_produk": "HP",
17            "jenis_produk": "Infra",
18            "stok": "1",
19            "harga_beli": "1300.00",
20            "harga_jual": "15000000.00",
21            "status": "Tersedia"
22        }
23    ];
24
25    return (
26        <div>
27            <br />
28            <h2>List Produk</h2>
29            <br />
30            <table border="1" cellPadding="10" cellSpacing="0">
31                <thead>
32                    <tr>
33                        <th>ID</th>
34                        <th>Nama Produk</th>
35                        <th>Jenis Produk</th>
36                        <th>Stok</th>
37                        <th>Harga Beli</th>
38                        <th>Harga Jual</th>
39                        <th>Status</th>
40                    </tr>
41                </thead>
42                <tbody>
43                    {dummyData.map((item) => (
44                        <tr key={item.id}>
45                            <td>{item.id}</td>
46                            <td>{item.nama_produk}</td>
47                            <td>{item.jenis_produk}</td>
48                            <td>{item.stok}</td>
49                            <td>{item.harga_beli}</td>
50                            <td>{item.harga_jual}</td>
51                            <td>{item.status}</td>
52                        </tr>
53                    )));
54                </tbody>
55            </table>
56        </div>
57    );
58}
59
60 export default ListProdukComponent;

```

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

```

penjualan-fe > src > App.jsx > ...
1  import './App.css'
2  import ListProdukComponent from './component/produk/ListProdukComponent'
3
4  function App() {
5    return (
6      <>
7        | <ListProdukComponent />
8      </>
9    )
10 }
11
12 export default App
13

```

16. Buka halaman web tadi, maka akan berubah seperti dibawah ini

List Produk						
ID	Nama Produk	Jenis Produk	Stok	Harga Beli	Harga Jual	Status
1	Laptop	Infra	122	20000	30000	Tersedia
2	HP	Infra	1	1300.00	15000000.00	Tersedia

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

The screenshot shows a terminal window with the following interface elements:

- Top bar: PROBLEMS (1), OUTPUT, DEBUG CONSOLE, TERMINAL, PORTS.
- Right side: node +, terminal tabs, and a close button.
- Bottom status bar: 11:37:50 [vite] hmr update /src/App.jsx (x2)
- Content area: A single black vertical bar representing the output of the command.

18. Lalu ketikkan perintah untuk masuk ke folder penjualan-fe, lalu perintah install bootstrap

```

● PS F:\Instruktur\Pemrograman 4\PRG4_Penjualan_Produk> cd penjualan-fe
● PS F:\Instruktur\Pemrograman 4\PRG4_Penjualan_Produk\penjualan-fe> npm install bootstrap --save
  added 2 packages, and audited 251 packages in 1s

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

  found 0 vulnerabilities

```

19. 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.
20. 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 memaggil bootstrap.

```

penjualan-fe > src > App.jsx > ...
1  import ListProdukComponent from './component/produk/ListProdukComponent'
2
3  function App() {
4    return (
5      <>
6        | <ListProdukComponent />
7      </>
8    )
9  }
10
11 export default App

```

```

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

```

21. Selanjutnya, kita akan menambahkan class didalam table

```

30     return (
31         <div className="container mt-4">
32             <h2 className="mb-4">List Produk</h2>
33             <table className='table table-striped'>
34                 <thead>
35                     <tr>

```

List Produk

ID	Nama Produk	Jenis Produk	Stok	Harga Beli	Harga Jual	Status
1	Laptop	Infra	122	20000	30000	Tersedia
2	HP	Infra	1	1300.00	15000000.00	Tersedia

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

```

● PS F:\Instruktur\Pemrograman 4\PRG4_Penjualan_Produk\penjualan-fe> npm install axios --save
added 9 packages, and audited 260 packages in 1s

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

found 0 vulnerabilities

```

23. Selanjutnya, buatlah folder baru dengan nama **services**, didalamnya buatlah file baru dengan nama **ProdukService.js**

```

penjualan-fe > src > services > JS ProdukService.js > ...
1  import axios from "axios"
2
3  const REST_API_BASE_URL = "https://api.roniprsty.com/produk/";
4
5  export const listProduk = () => axios.get(REST_API_BASE_URL + "read.php");

```

24. Selanjutnya, bukalah **ListProdukComponent.jsx**, modifikasi kodenya seperti dibawah ini

```
penjualan-fe > src > component > produk > ListProdukComponent.jsx > ListProdukComponent > useEffect() callback > fetch
  1 import React, { useEffect, useState } from 'react';
  2 import { listProduk } from '../../../../../services/ProdukService';
  3
  4 function ListProdukComponent() {
  5   const [produkData, setProdukData] = useState([]);
  6   const [loading, setLoading] = useState(true);
  7   const [error, setError] = useState(null);
  8
  9   useEffect(() => {
10     const fetchProduk = async () => {
11       try {
12         const response = await listProduk();
13         setProdukData(response.data);
14       } catch (error) {
15         console.error("Error fetching produk data:", error);
16         setError("Gagal mengambil data produk.");
17       } finally {
18         setLoading(false);
19       }
20     };
21     fetchProduk();
22   }, []);
23
24   if (loading) {
25     return <div className="text-center mt-5"><h4>Loading...</h4></div>;
26   }
27
28   return (
29     <div className="container mt-4">
30       <h2 className="mb-4">List Produk</h2>
31       {error && <div className="alert alert-danger">{error}</div>}
32       <table className='table table-striped'>
33         <thead>
34           <tr>
35             <th>ID</th>
36             <th>Nama Produk</th>
37             <th>Jenis Produk</th>
38             <th>Stok</th>
39             <th>Harga Beli</th>
40             <th>Harga Jual</th>
41             <th>Status</th>
42           </tr>
43         </thead>
44         <tbody>
45           {produkData.length > 0 ? (
46             produkData.map((item) => (
47               <tr key={item.id}>
48                 <td>{item.id}</td>
49                 <td>{item.nama_produk}</td>
50                 <td>{item.jenis_produk}</td>
51                 <td>{item.stok}</td>
52                 <td>{item.harga_beli.toLocaleString()} IDR</td>
53                 <td>{item.harga_jual.toLocaleString()} IDR</td>
54                 <td>{item.status}</td>
55               </tr>
56             ))
57           ) : (
58             <tr>
59               <td colspan="7" className="text-center">Tidak ada produk ditemukan.</td>
60             </tr>
61           )
62         </tbody>
63       </table>
64     </div>
65   );
66 }
67
68 export default ListProdukComponent;
```

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

List Produk						
ID	Nama Produk	Jenis Produk	Stok	Harga Beli	Harga Jual	Status
2	Beruang Kutub	Hewan	122	20000.00 IDR	30000.00 IDR	Aktif
38	kutil dika	Laptop	1	1300.00 IDR	15000000.00 IDR	aktif
40	ROG ALLY	Laptop	10222	23000000.00 IDR	24000000.00 IDR	Aktif
46	Lenovo	Laptop	122222	10000000.00 IDR	12000000.00 IDR	Aktif
52	Samsung yuzmi	Laptop	10	10000000.00 IDR	12500000.00 IDR	Aktif
67	icad gaming	tools	5	200000.00 IDR	500000.00 IDR	Aktif
69	Lenovo	Laptop	95	10000000.00 IDR	12000000.00 IDR	Aktif
109	Paracetamol	Obat Pusing	12	10000.00 IDR	120000.00 IDR	Aktif
117	Poco	Smartphone	100	4000000.00 IDR	3000000.00 IDR	Aktif
173	Rohman	Keyboard	1	100000.00 IDR	200000.00 IDR	Aktif
174	HP	Laptop	1	10000000.00 IDR	12000000.00 IDR	Aktif
176	Iphone	SmartPhone	4	5000000.00 IDR	10000000.00 IDR	Aktif
184	Lenovo Ideapad 4	Laptop	11	10000000.00 IDR	12500000.00 IDR	Aktif
189	Laptop Baru	Laptop	8	10000000.00 IDR	12000000.00 IDR	Aktif

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

```
penjualan-fe > src > component > templates > HeaderComponent.jsx > ...
1  import React from 'react';
2  import 'bootstrap/dist/css/bootstrap.min.css';
3
4  function HeaderComponent() {
5      return (
6          <nav className="navbar navbar-expand-lg navbar-light bg-light">
7              <div className="container">
8                  <a className="navbar-brand" href="/">Toko Informatika</a>
9                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav"
10                     aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">
11                      <span className="navbar-toggler-icon"></span>
12                  </button>
13                  <div className="collapse navbar-collapse" id="navbarNav">
14                      <ul className="navbar-nav">
15                          <li className="nav-item">
16                              <a className="nav-link" href="/">Home</a>
17                          </li>
18                          <li className="nav-item">
19                              <a className="nav-link" href="/list-produk">Produk</a>
20                          </li>
21                          <li className="nav-item">
22                              <a className="nav-link" href="/about">Tentang</a>
23                          </li>
24                          <li className="nav-item">
25                              <a className="nav-link" href="/contact">Kontak</a>
26                          </li>
27                      </ul>
28                  </div>
29              </div>
30          );
31      );
32  }
33  export default HeaderComponent;
```

27. Selanjutnya buat juga FooterComponent.jsx

```

1 import React from 'react';
2 import 'bootstrap/dist/css/bootstrap.min.css'; // Pastikan Bootstrap diimpor
3
4 function FooterComponent() {
5     return (
6         <footer className="bg-light text-center text-lg-start mt-auto">
7             <div className="container p-4">
8                 <div className="text-center">
9                     <h6 className="text-uppercase fw-bold mb-4">Toko Informatika</h6>
10                    <p>© {new Date().getFullYear()} Toko Informatika . All rights reserved.</p>
11                </div>
12                <div className="d-flex justify-content-center">
13                    <a href="/" className="me-4 text-reset">Home</a>
14                    <a href="/list-produk" className="me-4 text-reset">Produk</a>
15                    <a href="/about" className="me-4 text-reset">Tentang</a>
16                    <a href="/contact" className="me-4 text-reset">Kontak</a>
17                </div>
18            </div>
19        </footer>
20    );
21 }
22
23 export default FooterComponent;
24

```

28. Panggil HeaderComponent serta FooterComponent ke dalam App.jsx

```

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

```

29. Buka web browser anda, maka akan tampil seperti dibawah ini

ID	Nama Produk	Jenis Produk	Stok	Harga Beli	Harga Jual	Status
2	Beruang Kutub	Hewan	122	20000.00 IDR	30000.00 IDR	Aktif
38	kutil dika	Laptop	1	1300.00 IDR	1500000.00 IDR	aktif
40	ROG ALLY	Laptop	10222	23000000.00 IDR	24000000.00 IDR	Aktif
46	Lenovo	Laptop	122222	10000000.00 IDR	12000000.00 IDR	Aktif
52	Samsung yuzmi	Laptop	10	10000000.00 IDR	12500000.00 IDR	Aktif
67	icad gaming	tools	5	200000.00 IDR	500000.00 IDR	Aktif
69	Lenovo	Laptop	95	10000000.00 IDR	12000000.00 IDR	Aktif
109	Paracetamol	Obat Pusing	12	10000.00 IDR	120000.00 IDR	Aktif
117	Poco	Smartphone	100	4000000.00 IDR	3000000.00 IDR	Aktif

30. 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
```

31. Tambahkan Button Add Produk dalam ListProdukComponent.jsx

```
penjualan-fe > src > component > produk > ListProdukComponent.jsx > ListProdukComponent
  1 import React, { useEffect, useState } from 'react';
  2 import { listProduk } from '../../services/ProdukService';
  3 import { Link } from 'react-router-dom';
  4
  5
  31 <div className="container mt-4">
  32   <h2 className="mb-4">List Produk</h2>
  33   <Link to="/tambah-produk" className="btn btn-primary mb-3">Add Produk</Link>
  34   {error && <div className="alert alert-danger">{error}</div>}
  35   <table className='table table-striped'>
  36     <thead>
```

32. Buatlah AddProdukComponent.jsx, lalu ketikkan kode seperti dibawah ini

```
penjualan-fe > src > component > produk > AddProdukComponent.jsx > ...
1 import React, { useState } from 'react';
2 import { addProduk } from '../../services/ProdukService';
3 import { Link } from 'react-router-dom';
4
5 function AddProdukComponent() {
6   const [namaProduk, setNamaProduk] = useState('');
7   const [jenisProduk, setJenisProduk] = useState('');
8   const [stok, setStok] = useState('');
9   const [hargaBeli, setHargaBeli] = useState('');
10  const [hargaJual, setHargaJual] = useState('');
11  const [status, setStatus] = useState('Tersedia'); // Default status
12  const [error, setError] = useState(null);
13  const [successMessage, setSuccessMessage] = useState(null);
14
15  const handleSubmit = async (e) => {
16    e.preventDefault();
17    setError(null);
18    setSuccessMessage(null);
19
20    try {
21      // Mengirim data produk ke API
22      const newProduct = {
23        nama_produk: namaProduk,
24        jenis_produk: jenisProduk,
25        stok,
26        harga_beli: hargaBeli,
27        harga_jual: hargaJual,
28        status: status,
29      };
30      await addProduk(newProduct); // Menggunakan fungsi yang akan Anda buat di ProdukService
31      setSuccessMessage('Produk berhasil ditambahkan!');
32      // Reset form setelah berhasil
33      setNamaProduk('');
34      setJenisProduk('');
35      setStok('');
36      setHargaBeli('');
37      setHargaJual('');

```

```

38     setStatus('Tersedia');
39   } catch (error) {
40     console.error("Error adding product:", error);
41     setError("Gagal menambahkan produk. Silakan coba lagi.");
42   }
43 }
44
45 return (
46   <div className="container mt-4">
47     <h2>Tambah Produk</h2>
48     {error && <div className="alert alert-danger">{error}</div>}
49     {successMessage && <div className="alert alert-success">{successMessage}</div>}
50     <form onSubmit={handleSubmit}>
51       <div className="mb-3">
52         <label htmlFor="namaProduk" className="form-label">Nama Produk</label>
53         <input
54           type="text"
55           className="form-control"
56           id="namaProduk"
57           value={namaProduk}
58           onChange={(e) => setNamaProduk(e.target.value)}
59           required
60         />
61       </div>
62       <div className="mb-3">
63         <label htmlFor="jenisProduk" className="form-label">Jenis Produk</label>
64         <input
65           type="text"
66           className="form-control"
67           id="jenisProduk"
68           value={jenisProduk}
69           onChange={(e) => setJenisProduk(e.target.value)}
70           required
71         />
72     </div>

```

```

73       <div className="mb-3">
74         <label htmlFor="stok" className="form-label">Stok</label>
75         <input
76           type="number"
77           className="form-control"
78           id="stok"
79           value={stok}
80           onChange={(e) => setStok(e.target.value)}
81           required
82         />
83       </div>
84       <div className="mb-3">
85         <label htmlFor="hargaBeli" className="form-label">Harga Beli</label>
86         <input
87           type="number"
88           className="form-control"
89           id="hargaBeli"
90           value={hargaBeli}
91           onChange={(e) => setHargaBeli(e.target.value)}
92           required
93         />
94       </div>
95       <div className="mb-3">
96         <label htmlFor="hargaJual" className="form-label">Harga Jual</label>
97         <input
98           type="number"
99           className="form-control"
100          id="hargaJual"
101          value={hargaJual}
102          onChange={(e) => setHargaJual(e.target.value)}
103          required
104        />
105      </div>
106      <div className="mb-3">
107        <label htmlFor="status" className="form-label">Status</label>

```

```

107     <label htmlFor="status" className="form-label">Status</label>
108     <select
109       className="form-select"
110       id="status"
111       value={status}
112       onChange={(e) => setStatus(e.target.value)}
113     >
114       <option value="Tersedia">Tersedia</option>
115       <option value="Tidak Tersedia">Tidak Tersedia</option>
116     </select>
117   </div>
118   <button type="submit" className="btn btn-primary">Tambah Produk</button>
119   <Link to="/list-produk" className="btn btn-secondary ms-2">Kembali ke List Produk</Link>
120 </form>
121 </div>
122 );
123 }
124
125 export default AddProdukComponent;
126

```

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

```

penjualan-fe > src > App.jsx > App
1 import { BrowserRouter as Router, Route, Routes } from 'react-router-dom';
2 import HeaderComponent from './component/templates/HeaderComponent';
3 import FooterComponent from './component/templates/FooterComponent';
4 import AddProdukComponent from './component/produk/AddProdukComponent';
5 import ListProdukComponent from './component/produk/ListProdukComponent';
6
7 function App() {
8   return (
9     <Router>
10       <div className="d-flex flex-column min-vh-100">
11         <HeaderComponent />
12         <div className="flex-grow-1">
13           <Routes>
14             <Route path="/tambah-produk" element={<AddProdukComponent />} />
15             <Route path="/list-produk" element={<ListProdukComponent />} />
16             {/* Rute lainnya bisa ditambahkan di sini */}
17           </Routes>
18         </div>
19         <FooterComponent />
20       </div>
21     </Router>
22   );
23 }
24
25 export default App;
26

```

34. Selanjutnya, kita coba akses di halaman web, lalu klik tambah produk, maka halaman akan berpindah ke tambah-produk
35. Bukalah kembali ProdukService.js, tambahkan pemanggilan API untuk create

```

penjualan-fe > src > services > JS ProdukService.js > ...
1 import axios from "axios"
2
3 const REST_API_BASE_URL = "https://api.roniprsty.com/produk/";
4
5 export const listProduk = () => axios.get(REST_API_BASE_URL + "read.php");
6
7 export const addProduk = (newProduct) => {
8   return axios.post(REST_API_BASE_URL + "create.php", newProduct);
9 }

```

36. Cobalah untuk menjalankannya, tambahkan 1 data

Toko Informatika Home Produk Tentang Kontak

Tambah Produk

Nama Produk

Jenis Produk

Stok

Harga Beli

Harga Jual

Status

37. Selanjutnya, klik Submit maka data akan tersimpan

281	SekSekSek	Human	12	9000000.00 IDR	12000000.00 IDR	Aktif
287	Es Kuwut	minuman	30	5000.00 IDR	7000.00 IDR	Tidak Tersedia
288	RJ45	Laptop	20	2000.00 IDR	2000.00 IDR	Tersedia

TOKO INFORMATIKA

© 2024 Toko Informatika . All rights reserved.

Home Produk Tentang Kontak

5. LATIHAN

1. Rapikan setiap kolom didalam tabelnya dengan standart yaitu angka rata kanan, text rata kiri, harga menggunakan separator dan sebagainya
2. Modifikasi tampilan agar lebih menarik
3. Berikan validasi lengkap untuk setiap inputan
4. Tambahkan halaman dan aksi untuk update dan delete Produk menggunakan API di modul JavaScript