

LAPORAN TUGAS MINGGU 10

Disusun sebagai

MATA KULIAH: FRAMEWORK (PBF)

Oleh:

Geraldo Bintang I / 1741720154

TI – 3B / 15



PROGRAM STUDI: D-IV TEKNIK INFORMATIKA

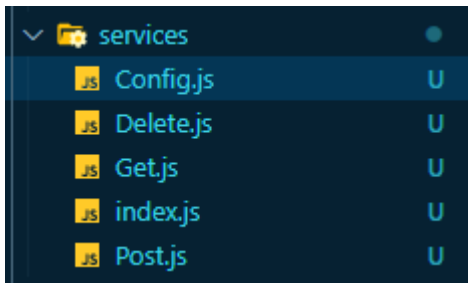
JURUSAN: TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

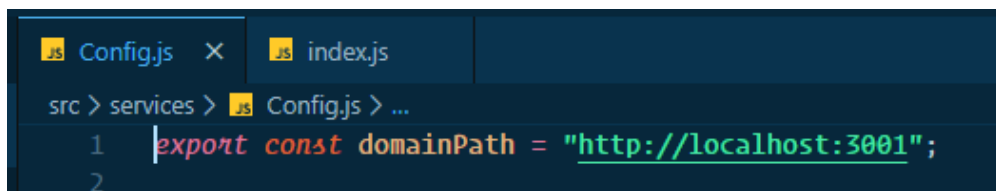
2020

TUGAS

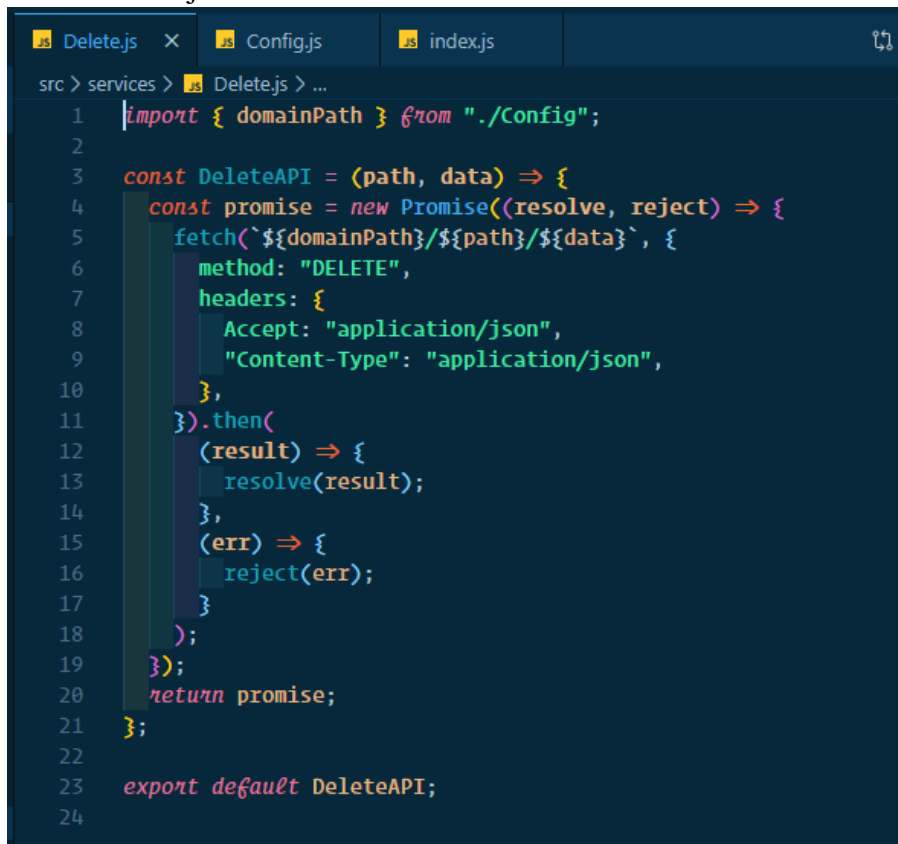
1. Buat folder services di dalam folder src, kemudian buat file Config, Delete, Get, index, Post. Seperti berikut :



2. Isi file Config.js



3. Isi file Delete.js



4. Isi file Get.js

```
Get.js  X  Delete.js  Config.js  index.js

src > services > Get.js > ...

1  import { domainPath } from "../Config";
2
3  const GetAPI = (path) => {
4    const promise = new Promise((resolve, reject) => {
5      fetch(`${domainPath}/${path}`)
6        .then((response) => response.json())
7        .then(
8          (result) => {
9            resolve(result);
10           },
11          (err) => {
12            reject(err);
13          }
14        );
15    });
16    return promise;
17  };
18
19  export default GetAPI;
20
```

5. Isi file index.js

```
index.js  ...services X  Get.js  Delete.js  Config.js  ...

src > services > index.js > ...

1  import GetAPI from "../Get";
2  import PostAPI from "../Post";
3  import DeleteAPI from "../Delete";
4
5  const getNewsBlog = () => GetAPI("posts?_sort=id&_order=desc");
6  const postNewsBlog = (dataYgDiKirim) => PostAPI("posts", dataYgDiKirim);
7  const deleteNewsBlog = (dataYgDiHapus) => DeleteAPI("posts", dataYgDiHapus);
8
9  const API = {
10    getNewsBlog,
11    postNewsBlog,
12    deleteNewsBlog,
13  };
14
15  export default API;
16
```

6. Isi file Post.js

```
Post.js  index.js ...\services  Get.js  Delete.js  ...
src > services > Post.js > ...
1  import { domainPath } from "../Config";
2
3  const PostAPI = (path, data) => {
4    const promise = new Promise((resolve, reject) => {
5      fetch(`${domainPath}/${path}`, {
6        method: "post",
7        headers: {
8          Accept: "application/json",
9          "Content-Type": "application/json",
10       },
11       body: JSON.stringify(data),
12     }).then(
13       (result) => {
14         resolve(result);
15       },
16       (err) => {
17         reject(err);
18       }
19     );
20   });
21   return promise;
22 };
23
24 export default PostAPI;
25
```

7. Isi file Blogpost.jsx

```
BlogPost.jsx X Post.js index.js ...services Get.js Delete.js Config.js ...
src > container > BlogPost > BlogPost.jsx > ...
1 import React, { Component } from "react"; 8.4K (gzipped: 3.4K)
2 import "./BlogPost.css";
3 import Post from "../../component/BlogPost/Post";
4 import API from "../../services/index";
5
6 class BlogPost extends Component {
7   state = {
8     listArtikel: [],
9     insertArtikel: {
10       id: 1,
11       nim: 1,
12       nama: "",
13       alamat: "",
14       hp: "",
15       angkatan: 1,
16       status: "",
17     },
18   };
19
20   ambilDataDariServerAPI = () => {
21     API.getNewsBlog().then((result) => {
22       this.setState({
23         listArtikel: result,
24       });
25     });
26   };
27
28   componentDidMount() {
29     this.ambilDataDariServerAPI();
30   }
31
32   handleTambahArtikel = (event) => {
33     let formInsertArtikel = { ...this.state.insertArtikel };
34     let id = new Date().getTime();
35     formInsertArtikel["id"] = id;
36     formInsertArtikel[event.target.name] = event.target.value;
37     this.setState({
38       insertArtikel: formInsertArtikel,
39     });
40   };
41
42   handleTombolSimpan = () => {
43     API.postNewsBlog(this.state.insertArtikel).then(() => {
44       this.ambilDataDariServerAPI();
45     });
46   };
47 }
```

```

47
48   handleHapusArtikel = (data) => {
49     API.deleteNewsBlog(data).then(() => {
50       this.ambilDataDariServerAPI();
51     });
52   };
53
54   render() {
55     return (
56       <div className="post-artikel">
57         <div className="form pb-2 border-bottom">
58           <div className="form-group row"></div>
59           <label htmlFor="title" className="col-form-label">
60             NIM
61           </label>
62           <div className="col-sm-10">
63             <input
64               type="text"
65               className="form-nim"
66               id="nim"
67               name="nim"
68               onChange={this.handleTambahArtikeL}
69             />
70           </div>
71           <div className="form-group row"></div>
72           <label htmlFor="title" className="col-form-label">
73             Nama
74           </label>
75           <div className="col-sm-10">
76             <input
77               type="text"
78               className="form-nama"
79               id="nama"
80               name="nama"
81               onChange={this.handleTambahArtikeL}
82             />
83           </div>
84           <div className="form-group row"></div>
85           <label htmlFor="title" className="col-form-label">
86             Alamat
87           </label>
88           <div className="col-sm-10">
89             <input
90               type="text"
91               className="form-alamat"
92               id="alamat"
93               name="alamat"

```

```

94         onChange={this.handleTambahAntike}
95     />
96 </div>
97 <div className="form-group row"></div>
98 <label htmlFor="title" className="col-form-label">
99     Nomor HP:
100 </label>
101 <div className="col-sm-10">
102     <input
103         type="text"
104         className="form-hp"
105         id="hp"
106         name="hp"
107         onChange={this.handleTambahAntike}
108     />
109 </div>
110 <div className="form-group row"></div>
111 <label htmlFor="title" className="col-form-label">
112     Angkatan
113 </label>
114 <div className="col-sm-10">
115     <input
116         type="text"
117         className="form-angkatan"
118         id="angkatan"
119         name="angkatan"
120         onChange={this.handleTambahAntike}
121     />
122 </div>
123 <div className="form-group row"></div>
124 <label htmlFor="title" className="col-form-label">
125     Status
126 </label>
127 <div className="col-sm-10">
128     <input
129         type="text"
130         className="form-status"
131         id="status"
132         name="status"
133         onChange={this.handleTambahAntike}
134     />
135 </div>
136 </div>
137 <button
138     type="submit"
139     className="btn btn-primary"
140     onClick={this.handleTomboSimpan}

```

8. Isi file Post.jsx

```
Post.jsx | BlogPost.jsx | Postjs | indexjs ...services | Getjs | Deletejs | ...
src > component > BlogPost > Postjs > ...
1  import React from "react"; 8.4K (gzipped: 3.4K)
2
3  const Post = props => {
4    return (
5      <div className="artikel">
6        <div className="gambar-artikel">
7          
11        </div>
12        <div className="konten-artikel">
13          <p className="nim">NIM: {props.nim}</p>
14          <p className="nama">Nama: {props.nama}</p>
15          <p className="alamat">Alamat: {props.alamat}</p>
16          <p className="hp">Nomor HP: {props.hp}</p>
17          <p className="angkatan">Angkatan (Tahun): {props.angkatan}</p>
18          <p className="status">Status: {props.status}</p>
19          <button className="btn btn-sm btn-warning" onClick={() => props.hapusArtikel(
20            props.nim)}>
21            Hapus
22          </button>
23        </div>
24      </div>
25    );
26  };
27  export default Post;
28
```

9. Hasil

Penambahan Mahasiswa:

React App | localhost:3002

NIM

Nama


Alamat

Nomor HP:

Angkatan

Status

Daftar Mahasiswa

	<p>NIM: 1586342419022</p> <p>Nama: Geraldo Bintang I</p> <p>Alamat: Malang , Jatim</p> <p>Nomor HP: -</p> <p>Angkatan (Tahun): 2017</p> <p>Status: Mahasiswa</p> <p><input type="button" value="Hapus"/></p>
---	--

Hapus Mahasiswa

NIM

1741720154

Nama

Geraldo Bintang I

Alamat

Malang , Jatim

Nomor HP:

-

Angkatan

2017

Status

Mahasiswa

Simpan

Daftar Mahasiswa

Status pada json-server

Post = Untuk Penambahan Mahasiswa

Delete = Untuk Penghapusan Mahasiswa

```
POST /posts 201 9.523 ms - 169
GET /posts?_sort=id&_order=desc 200 4.274 ms - 191
DELETE /posts/1586342419022 200 5.972 ms - 2
GET /posts?_sort=id&_order=desc 200 4.240 ms - 2
[]
```

PERTANYAAN

Praktikum 1

a. Fungsi GetAPI: Untuk menjalankan method promise yang dimana domainPath dapat di *resolve* (di *fetch*) atau di *reject*

Fungsi getNewsBlog: Untuk mendeklarasi bagaimana artikel akan di fetch

Praktikum 2

1. Untuk menambah data yang akan di tambahkan pada file .json
2. Adalah value yang akan di jadikan acuan yaitu insertArtikel

Praktikum 3

1. Untuk menghapus data yang ada pada file .json
2. Adalah value yang akan di jadikan acuan yaitu insertArtikel

Praktikum 4

1. Membuat file baru kemudian mengimport-nya pada index.js