

LAPORAN MINGGU 10

Disusun sebagai

MATA KULIAH: PEMGROGRAMAN BERBASIS FRAMEWORK

Oleh:

Aditya Panca Putra / 1741720187
TI – 3B / 02



PROGRAM STUDI: D-IV TEKNIK INFORMATIKA
JURUSAN: TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2020

PERTANYAAN

Praktikum 1

- 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. Value yang akan di jadikan acuan yaitu insertArtikel

Praktikum 3

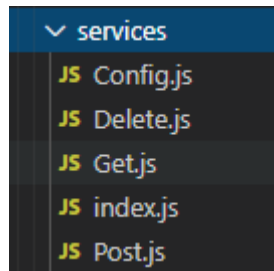
1. Untuk menghapus data yang ada pada file .json
2. Value yang akan di jadikan acuan yaitu insertArtikel

Praktikum 4

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

TUGAS

1. Membuat folder services di dalam folder src, kemudian buat file baru. Seperti berikut



2. Isikan code pada file Config.js

```
tugas > src > services > JS Config.js > ...  
1   export const domainPath = "http://localhost:3001";
```

3. Isikan code pada file Delete.js

```
tugas > src > services > JS Delete.js > [⌘] default  
1   import { domainPath } from "../Config";  
2  
3   const DeleteAPI = (path, data) => {  
4     const promise = new Promise((resolve, reject) => {  
5       fetch(`${domainPath}/${path}/${data}`, {  
6         method: "DELETE",  
7         headers: {  
8           Accept: "application/json",  
9           "Content-Type": "application/json",  
10        },  
11      }).then(  
12        (result) => {  
13          resolve(result);  
14        },  
15        (err) => {  
16          reject(err);  
17        }  
18      );  
19    });  
20    return promise;  
21  };  
22  
23  export default DeleteAPI;
```

4. Isikan code pada file Get.js

```
tugas > src > services > JS Get.js > [🔍] default
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;
```

5. Isikan code pada file index.js

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

6. Isikan code pada file Post.js

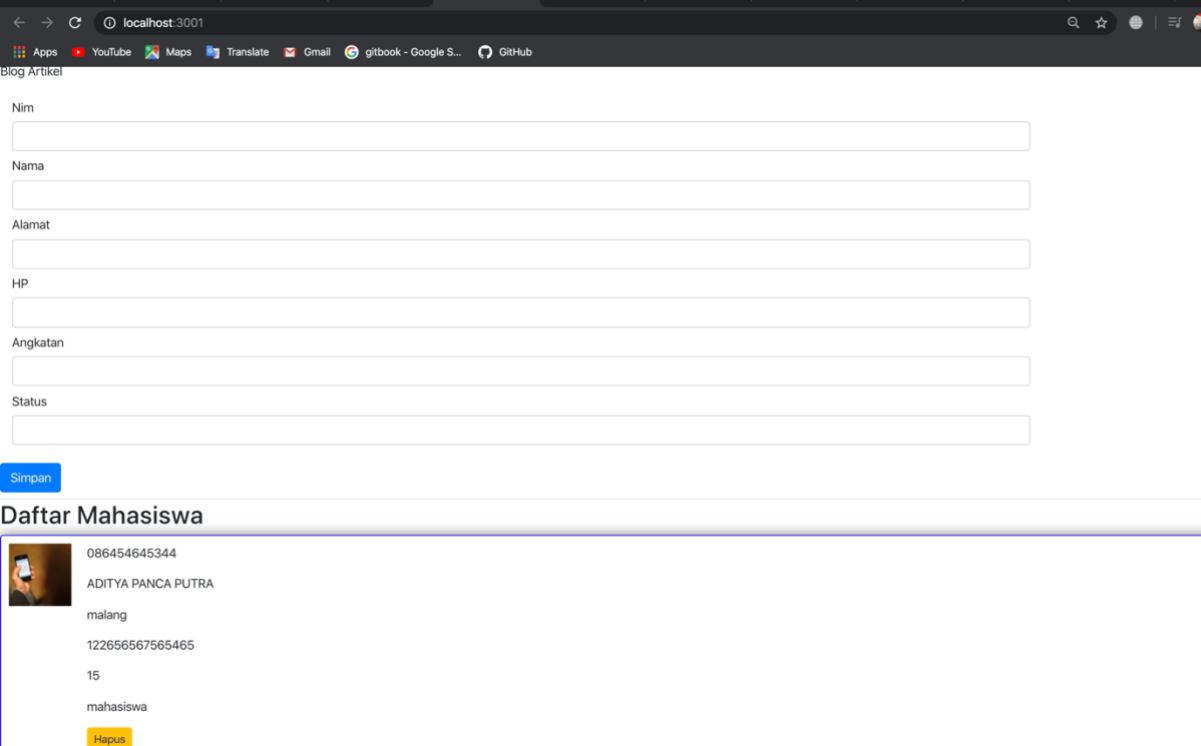
```
tugas > src > services > JS Post.js > [⌕] default
1  import { domainPath } from "../Config";
2
3  const PostAPI = (path, data) => {
4    const promise = new Promise((resolve, reject) => {
5      fetch(`${domainPath}/${path}`, {
6        method: "post",
7        headers: {
8          Accept: "application/json",
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;
```

7. Isikan code pada file Post.js

```
tugas > src > component > BlogPost > Post.jsx > default
1  import React from "react";
2
3  const Post = props => {
4    return (
5      <div className="artikel">
6        <div className="gambar-artikel">
7          </img>
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(props.nim)}>
20            Hapus
21          </button>
22        </div>
23      </div>
24    );
25  };
26
27  export default Post;
```

8. Hasil saat di Run

➤ Tambah




The screenshot shows a web browser at localhost:3001. The page has a header "Blog Artikel" and a navigation bar with links to Apps, YouTube, Maps, Translate, Gmail, gitbook - Google S..., and GitHub. The main content area contains a form for adding a student with the following fields:

- Nim:
- Nama:
- Alamat:
- HP:
- Angkatan:
- Status:

Below the form is a blue button labeled "Simpan".

Below the button is a section titled "Daftar Mahasiswa" with a table of student data:

	086454645344
	ADITYA PANCA PUTRA
	malang
	122656567565465
	15
	mahasiswa
	<button>Hapus</button>

➤ Hapus



Blog Artikel

Nim

Nama

Alamat

HP

Angkatan

Status

[Simpan](#)

Daftar Mahasiswa