

TUGAS INDIVIDU

MODUL 8

Disusun sebagai

MATA KULIAH : PEMROGRAMAN BERBASIS FRAMEWORK

Oleh :

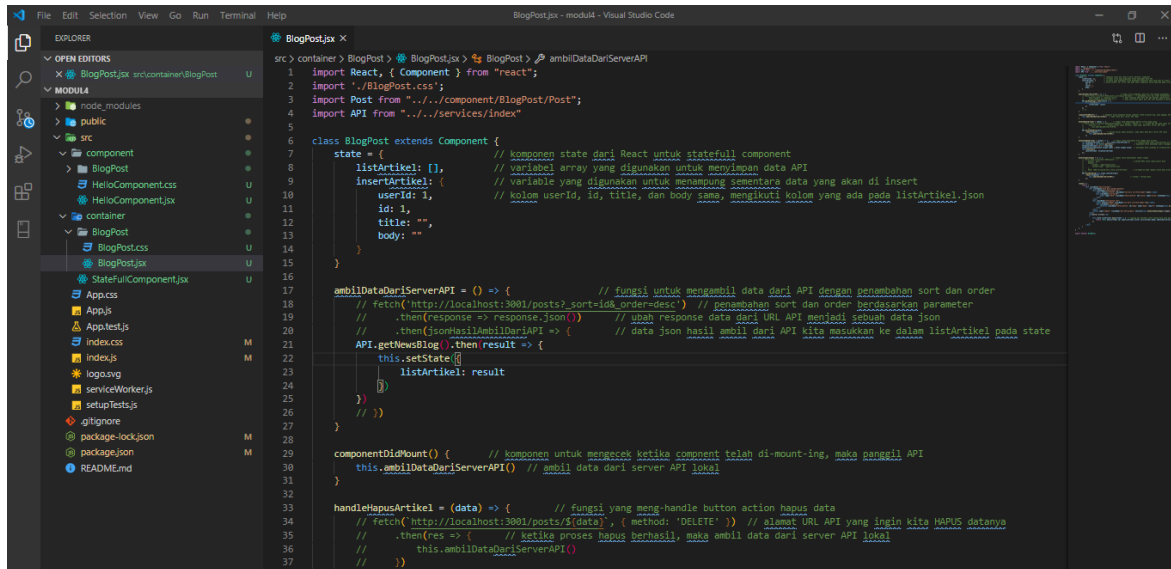
Nor Wahid Hidayad Ulloh / 1641720157 TI 3-A



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

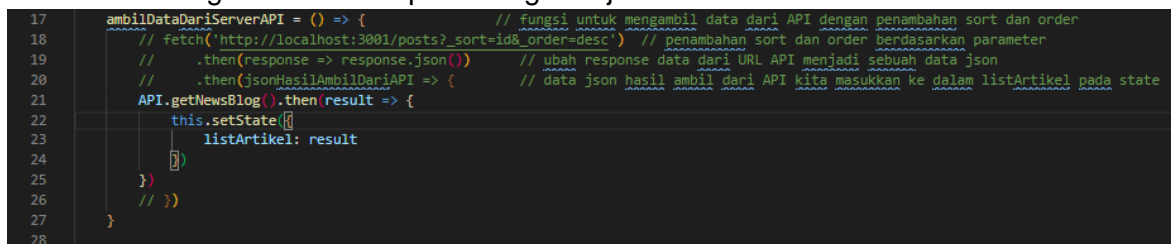
A. Praktikum 1

1. Membuat index.js pada folder services
2. Membuka file BlogPost.jsx



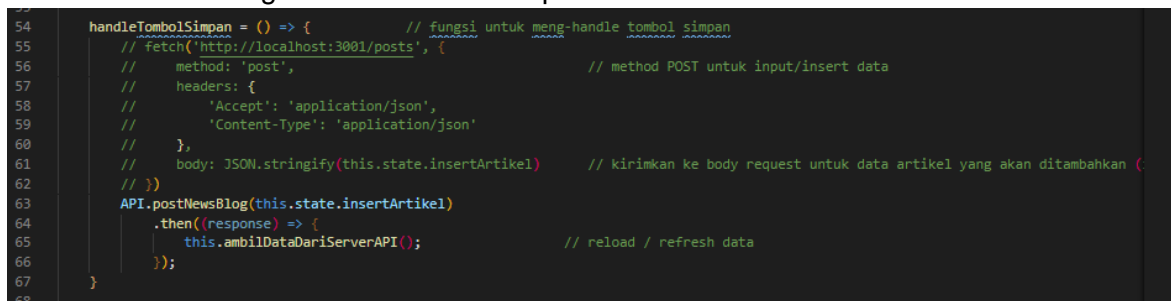
3. Membuat kode pada index.js

4. Mengubah baris 17 pada BlogPost.jsx



B. Praktikum 2

1. Buka fungsi handleTombolSimpan



2. Mengedit fungsi handleTombolSimpan dan membuat post

```

handleTombolSimpan = () => { // fungsi untuk meng-handle tombol simpan
  // fetch('http://localhost:3001/posts', {
  //   method: 'post', // method POST untuk
  //   headers: {
  //     'Accept': 'application/json',
  //     'Content-Type': 'application/json'
  //   },
  //   body: JSON.stringify(this.state.insertArtikel) // kirimkan ke body
  // })
  API.postNewsBlog(this.state.insertArtikel)
    .then((response) => {
      this.ambilDataDariServerAPI(); // reload / refresh data
    });
}

```

C. Praktikum 3

1. Buka fungsi handleHapusArtikel

```

33 handleHapusArtikel = (data) => { // fungsi yang meng-handle button action hapus data
34   // fetch('http://localhost:3001/posts/${data}', { method: 'DELETE' }) // alamat URL API yang ingin kita HAPUS datanya
35   // .then(res => { // ketika proses hapus berhasil, maka ambil data dari server API lokal
36   //   this.ambilDataDariServerAPI()
37   // })
38   API.deleteNewBlog(data)
39   .then(response => { // ketika proses hapus berhasil, maka ambil data dari server API lokal
40     this.ambilDataDariServerAPI()
41   })
42 }
43

```

2. Mengedit isi fungsi handleHapusArtikel

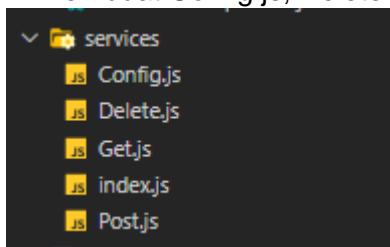
```

handleHapusArtikel = (data) => { // fu
  // fetch('http://localhost:3001/posts/${data}', {
  //   .then(res => { // ketika proses
  //     this.ambilDataDariServerAPI()
  //   })
  API.deleteNewBlog(data)
    .then(response => { // ketika proses
      this.ambilDataDariServerAPI()
    })
}

```

D. Praktikum 4

1. Membuat Config.js, Delete.js, Get.js, Post.js pada folder services



2. Mengisikan kode pada Config.js

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

3. Mengisikan kode pada Delete.js

```
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 |         })  
8 |         .then((result) => {  
9 |             resolve(result)  
10 |         }, err => {  
11 |             reject(err)  
12 |         })  
13 |     })  
14 |     return promise;  
15 | }  
16 |  
17 | export default DeleteAPI;|
```

4. Mengisikan kode pada Get.js

```
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((result) => {  
8 |             resolve(result)  
9 |         }, err => {  
10 |             reject(err)  
11 |         })  
12 |     })  
13 |     return promise;  
14 | }  
15 |  
16 | export default getAPI;|
```

5. Mengisikan kode pada Post.js

```

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 |        })
13 |        .then((result) => {
14 |            resolve(result)
15 |        }, err => {
16 |            reject(err)
17 |        })
18 |    })
19 |    return promise;
20 | }
21 |
22 | export default PostAPI;

```

6. Mengedit index.js

```

src > services > JS index.js > [⌕] default
1 | import React from 'react';
2 | import getAPI from './Get';
3 | import PostAPI from './Post';
4 | import DeleteAPI from './Delete';
5 |
6 | const getNewsBlog = () => getAPI("posts?_sort=id&_order=asc")
7 | const postNewsBlog = (dataYangDikirim) => PostAPI("posts", dataYangDikirim)
8 | const deleteNewBlog = (dataYgDihapus) => DeleteAPI("posts", dataYgDihapus)
9 |
10 | const API = {
11 |     getNewsBlog,
12 |     postNewsBlog,
13 |     deleteNewBlog
14 | }
15 |
16 | export default API;

```

7. Hasil setelah dijalankan

- Setelah ditambahkan

