

# **LAPORAN PEMROGRAMAN FRAMEWORK**

## **“SESSION 9 GLOBAL API”**



**Oleh:**

**Nama : Bagus Satria Putra**  
**Kelas : 3F**  
**Absen/NIM : 8 / 1841720146**

**PROGRAM STUDI TEKNIK INFORMATIKA**  
**JURUSAN TEKNOLOGI INFORMASI**  
**POLITEKNIK NEGERI MALANG**

**16 APRIL 2021**

Link Github :

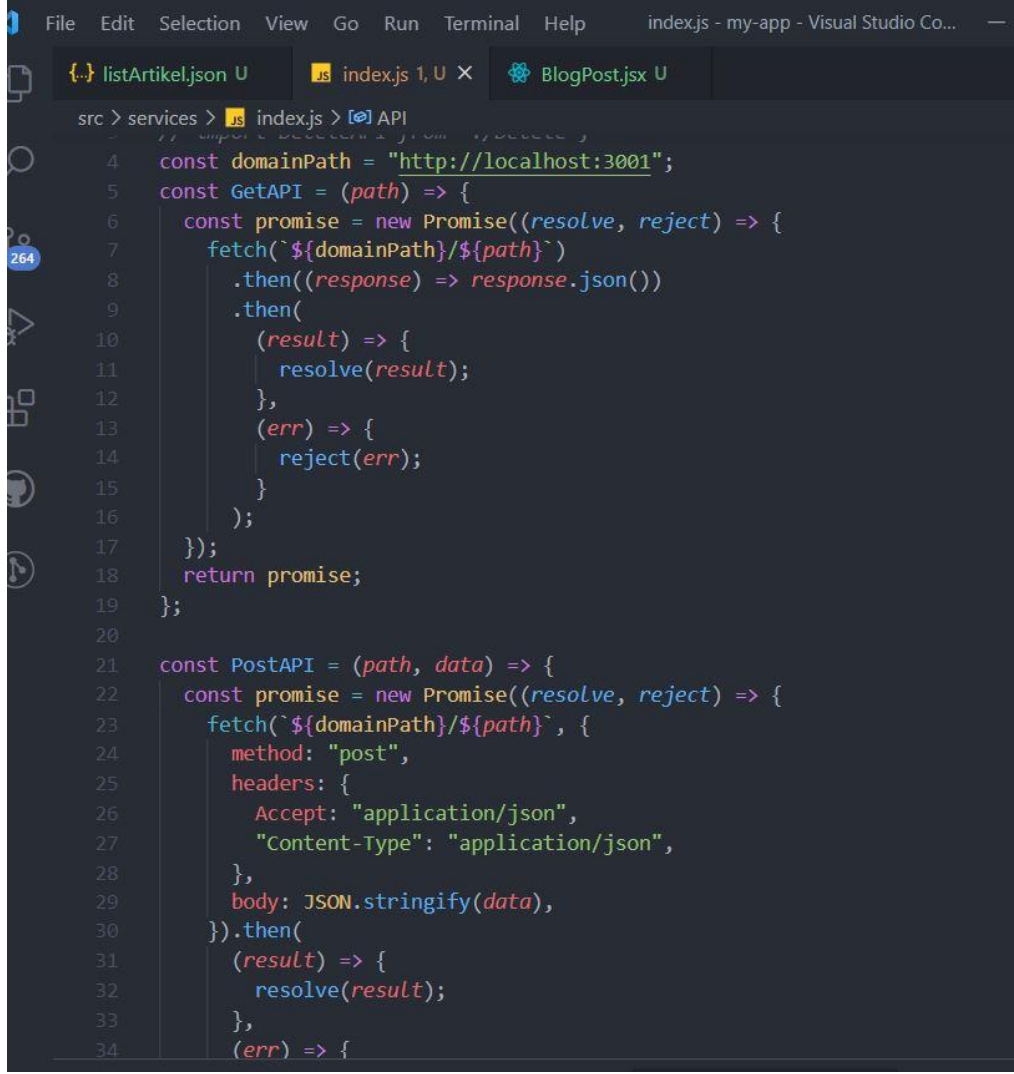
<https://github.com/BagusSatria123/PemrogramanFrameworkReact/tree/master/Pertemuan9>

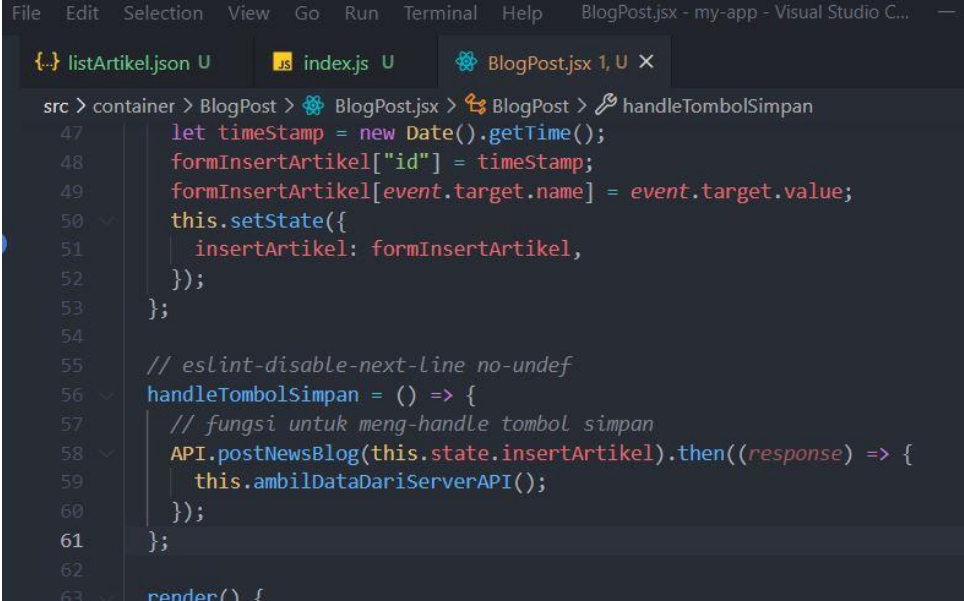
Link Youtube : <https://youtu.be/fWIkol4ePlo>

No.	Keterangan
<b>PRATIUM 1 GET API</b>	
	<p>Buat folder baru bernama “Services” dalam folder src, kemudian buat file index.js seperti pada Gambar 1.1</p> <p>Buka file BlogPost.jsx pada folder container (statefull component) dan akan tampil kode program seperti Gambar 1.2</p> <p><b>Jawaban :</b></p>  <pre>src &gt; container &gt; BlogPost &gt; BlogPost.jsx &gt; BlogPost &gt; render 1 import './BlogPost.css'; 2 3 import Post from "../../component/BlogPost/Post"; 4 import './BlogPost.css'; 5 import API from "../../services/index"; 6 7 class BlogPost extends Component { 8   // eslint-disable-next-line no-undef 9   state = { 10     listArtikel: [], 11     insertArtikel: { 12       NIM: 1, 13       id: 1, 14       nama: "", 15       alamat: "", 16       hp: "", 17       angkatan: "", 18       status: "", 19     }, 20   }; 21 22   // eslint-disable-next-line no-undef 23   ambilDataDariServerAPI = () =&gt; { 24     API.getNewsBlog().then((result) =&gt; { 25       this.setState({ 26         listArtikel: result, 27       }); 28     }); 29   }; 30 31   componentDidMount() { 32     this.ambilDataDariServerAPI();</pre>
	<p>Pada fungsi ambilDataDariServerAPI (baris16) terdapat pemanggilan API GET untuk merequest data artikel. Proses dari fungsi inilah yang akan kita manage kedalam satu tempat yaitu index.js.</p> <p>Buka file index.js pada folder service, yang telah kita buat tadi dan tuliskan baris kode seperti Gambar 1.3 berikut</p> <p><b>Jawaban :</b></p>

	 <pre> File Edit Selection View Go Run Terminal Help index.js - my-app - Visual Studio Co...  {...} listArtikel.json U JS index.js 1, U X BlogPost.jsx U  src &gt; services &gt; JS index.js &gt; ... 1 // import GetAPI from "../Get"; 2 // import PostAPI from "../Post"; 3 // import DeleteAPI from "../Delete"; 4 const domainPath = "http://localhost:3001"; 5 const GetAPI = (path) =&gt; { 6   const promise = new Promise((resolve, reject) =&gt; { 7     fetch(`\${domainPath}/\${path}`) 8       .then((response) =&gt; response.json()) 9       .then( 10         (result) =&gt; { 11           resolve(result); 12         }, 13         (err) =&gt; { 14           reject(err); 15         } 16       ); 17   }); 18   return promise; 19 }; 20 21 // const PostAPI = (path, data) =&gt; { </pre>
	<p>Kembali ke file BlogPost.js. Ganti baris 17 sampai baris 23 menjadi seperti Gambar 1.4</p> <p><b>Jawaban :</b></p>  <pre> {...} listArtikel.json U JS index.js U BlogPost.jsx 1, U X  src &gt; container &gt; BlogPost &gt; BlogPost.jsx &gt; BlogPost &gt; render 18 status. , 19 }, 20 }; 21 22 // eslint-disable-next-line no-undef 23 ambilDataDariServerAPI = () =&gt; { 24   API.getNewsBlog().then((result) =&gt; { 25     this.setState({ 26       listArtikel: result, 27     }); 28   }); </pre>
	<p>Simpan file BlogPost.js dan index.js tersebut, dan amati apa yang terjadi pada browser kalian.</p> <p><b>Jawaban :</b></p>

	<h2 style="text-align: center;">Daftar Artikel</h2> <div style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;">  <h3>Mengenal Covid-19</h3> <p>Covid-19 merupakan sebuah virus mematikan yang berasal dari china.</p> <p style="text-align: center;"><a href="#">Hapus</a></p> </div> <div style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;">  <h3>explicabo et eos deleniti nostrum ab id repellendus</h3> <p>animi esse sit aut sit nesciunt assumenda eum voluptas quia voluptatibus provident quia necessitatibus ea rerum repudiandae quia voluptatem delectus fugit aut id quia ratione optio eos iusto veniam iure</p> <p style="text-align: center;"><a href="#">Hapus</a></p> </div> <div style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;">  <h3>provident vel ut sit ratione est</h3> <p>debitis et eaque non officia sed nesciunt pariatur vel voluptatem iste vero et ea numquam aut expedita ipsum nulla in voluptates omnis consequatur aut enim officiis in quam qui</p> <p style="text-align: center;"><a href="#">Hapus</a></p> </div> <div style="border: 1px solid #ccc; padding: 10px;">  <h3>fuga nam accusamus voluptas reiciendis itaque</h3> <p>ad mollitia et omnis minus architecto odit voluptas doloremque maxime aut non ipsa qui alias veniam blanditiis culpa aut quia nihil cumque facere et occaecati qui aspernatur quia eaque ut aperiam</p> </div>
	<p><b>Pertanyaan Praktikum 1</b></p> <p>a. Perhatikan file index.js, apa tujuan dibuatnya fungsi GetAPI pada baris 2 dan fungsi getNewsBlog pada baris 16?</p> <p><b>Jawaban :</b></p> <ul style="list-style-type: none"> <li>- GetAPI berfungsi untuk membaca data/resource dari REST server.</li> <li>- getNewsBlog mengambil membaca data json.</li> </ul>
<b>PRATIUM 2 POST API</b>	
	<ol style="list-style-type: none"> <li>1. Kita perhatikan pada fungsi handleTombolSimpan pada file BlogPost.js. Bagian inilah yang selanjutnya kita kelola agar bisa di-manage.</li> <li>2. Buatlah fungsi untuk menampung action POST dari file BlogPost.js pada file services/index.js dan inialisasi fungsi tersebut seperti pada Gambar 2.2</li> </ol> <p><b>Jawaban :</b></p>

	<div data-bbox="293 193 1300 1260"><pre>src &gt; services &gt; index.js &gt; [API] 1 // import axios from 'axios'; 2 3 4 const domainPath = "http://localhost:3001"; 5 const GetAPI = (path) =&gt; { 6   const promise = new Promise((resolve, reject) =&gt; { 7     fetch(`\${domainPath}/\${path}`) 8       .then((response) =&gt; response.json()) 9       .then( 10         (result) =&gt; { 11           resolve(result); 12         }, 13         (err) =&gt; { 14           reject(err); 15         } 16       ); 17   }); 18   return promise; 19 }; 20 21 const PostAPI = (path, data) =&gt; { 22   const promise = new Promise((resolve, reject) =&gt; { 23     fetch(`\${domainPath}/\${path}`, { 24       method: "post", 25       headers: { 26         Accept: "application/json", 27         "Content-Type": "application/json", 28       }, 29       body: JSON.stringify(data), 30     }).then( 31       (result) =&gt; { 32         resolve(result); 33       }, 34       (err) =&gt; {</pre></div> <p data-bbox="293 1264 1421 1333">3. Selanjutnya pindah ke file BlogPost.js dan ganti isi dari fungsi handleTombolSimpan menjadi seperti Gambar 2.3</p> <p data-bbox="293 1339 418 1369"><b>Jawaban :</b></p>
--	--

	 <pre> File Edit Selection View Go Run Terminal Help BlogPost.jsx - my-app - Visual Studio C... listArtikel.json U index.js U BlogPost.jsx 1, U X src &gt; container &gt; BlogPost &gt; BlogPost.jsx &gt; BlogPost &gt; handleTombolSimpan 47 let timeStamp = new Date().getTime(); 48 formInsertArtikel["id"] = timeStamp; 49 formInsertArtikel[event.target.name] = event.target.value; 50 this.setState({ 51   insertArtikel: formInsertArtikel, 52 }); 53 }; 54 55 // eslint-disable-next-line no-undef 56 handleTombolSimpan = () =&gt; { 57   // fungsi untuk meng-handle tombol simpan 58   API.postNewsBlog(this.state.insertArtikel).then((response) =&gt; { 59     this.ambilDataDariServerAPI(); 60   }); 61 }; 62 63 render() { </pre>
	<p>4. Selesai. Kode program pada BlogPost.js sedikit lebih simple dari seelumnya.</p> <p>5. Silahkan kalian jalankan proses input artikel melalui browser dan amati apa yang terjadi</p> <p><b>Jawaban :</b></p>

	
	<p><b>Pertanyaan Praktikum 2</b></p> <ul style="list-style-type: none"> <li>- Perhatikan file index.js, apa tujuan dibuatnya fungsi PostAPI dan fungsi postNewsBlog?</li> <li>- Pada fungsi postNewsBlog, terdapat variable dataYangDiKirim. Apa tujuan dari dibuatnya variable tersebut?</li> </ul> <p><b>Jawaban :</b></p> <ul style="list-style-type: none"> <li>- Fungsi PostAPI Menggunaannya method untuk kelompok HTTP verb ini adalah untuk membuat (create) item/resource baru. Kelompok method ini biasanya tidak mengembalikan keluaran/output yang kadang disebut <b>procedure</b>. Fungsi postNewsblog mengambil membaca data json.</li> <li>- Agar variabel NewsBlog di post tidak bertabrakan dengan vaiable NewsBlog lainnya.</li> </ul>
	<p><b>Pratikum 3 Delete API</b></p> <p>Kita perhatikan pada fungsi handleHapusArtikel pada file BlogPost.jsx. Bagian inilah yang selanjutnya kita kelola agar bisa di-manage.</p> <p><b>Jawaban :</b></p>

	 <pre>File Edit Selection View Go Run Terminal Help BlogPost.jsx - my-app - Visual Studio C... {..} listArtikel.json U JS index.js U BlogPost.jsx 1, U X src &gt; container &gt; BlogPost &gt; BlogPost.jsx &gt; BlogPost &gt; ambilDataDariServerAPI 27     }); 28   }); 29   }; 30   31   componentDidMount() { 32     this.ambilDataDariServerAPI(); 33   } 34   35   // eslint-disable-next-line no-undef 36   handleHapusArtikel = (data) =&gt; { 37     API.deleteNewsBlog(data).then((response) =&gt; { 38       this.ambilDataDariServerAPI(); 39     }); 40   }; 41   42   // eslint-disable-next-line no-undef</pre>
	<p>Buatlah fungsi untuk menampung action DELETE dari file BlogPost.jsx pada file services/index.js dan inisialisasi fungsi tersebut seperti pada Gambar 3.2</p> <p><b>Jawaban :</b></p>



```
File Edit Selection View Go Run Terminal Help index.js - my-app - Visual Studio Co...
listArtikel.json U index.js 1, U X BlogPost.jsx U
src > services > index.js > API > deleteNewsBlog
39   return promise;
40 };
41
42 const DeleteAPI = (path, data) => {
43   const promise = new Promise((resolve, reject) => {
44     fetch(`${domainPath}/${path}/${data}`, { method: "DELETE" }).then(
45       (result) => {
46         resolve(result);
47       },
48       (err) => {
49         reject(err);
50       }
51     );
52   });
53   return promise;
54 };
55
56 const getNewsBlog = () => GetAPI("posts?_sort=id& order=desc");
57 const postNewsBlog = (dataYgDikirim) => PostAPI("posts", dataYgDikirim);
58 const deleteNewsBlog = (dataYgDiHapus) => DeleteAPI("posts", dataYgDiHapus);
59
60 const API = {
61   getNewsBlog,
62   postNewsBlog,
63   deleteNewsBlog,
64 };
65
66 export default API;
67
```

Selanjutnya pindah ke file BlogPost.jsx dan ganti isi dari fungsi handleHapusArtikel menjadi seperti Gambar 3.3

**Jawaban :**

```
File Edit Selection View Go Run Terminal Help BlogPost.jsx - my-app - Visual Studio C...
listArtikel.json U index.js U BlogPost.jsx 1, U X
src > container > BlogPost > BlogPost.jsx > BlogPost > handleHapusArtikel
34
35 // eslint-disable-next-line no-undef
36 handleHapusArtikel = (data) => {
37   API.deleteNewsBlog(data).then((response) => {
38     this.ambilDataDariServerAPI();
39   });
40 };
41
42 // eslint-disable-next-line no-undef
43 handleTambahArtikel = (event) => {
```

Silahkan kalian jalankan proses hapus artikel melalui browser dan amati apa yang terjadi.

**Jawaban :**

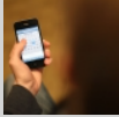
**Sebelum**

→ ↺ ⓘ localhost:3000

alamat

Simpan


Daftar Artikel



**Mengenal Covid-19**

Covid-19 merupakan sebuah virus mematikan yang berasal dari china.


Hapus



**explicabo et eos deleniti nostrum ab id repellendus**

animi esse sit aut sit nesciunt assumenda eum voluptas quia voluptatibus provident quia necessitatibus ea rerum repudiandae quia voluptatem delectus fugit aut id quia ratione optio eos iusto veniam iure

Hapus



**provident vel ut sit ratione est**

debitis et eaque non officia sed nesciunt pariatur vel voluptatem iste vero et ea numquam aut expedita ipsum nulla in voluptates omnis consequatur aut enim officiis in quam qui

Hapus

Sesudah baris nomor 2 dihapus

	
	<b>Pertanyaan Praktikum 3</b>
	<ol style="list-style-type: none"> <li>1. Perhatikan file index.js, apa tujuan dibuatnya fungsi DeleteAPI dan fungsi deleteNewsBlog?</li> <li>2. Pada fungsi deleteNewsBlog, terdapat variable dataYangDiHapus. Apa tujuan dari dibuatnya variable tersebut?</li> </ol> <p><b>Jawaban :</b></p> <ol style="list-style-type: none"> <li>1. Fungsi DeleteAPI melakukan delete json sehingga setiap id Ketika di klik terhapus. Fungsi DeleteNewsBlog delete di json Ketika id dituju.</li> <li>2. Agar variable deleteNewsBlog tidak bertabrakan dengan variable lainnya</li> </ol>
	<b>Pratikum 4 Manage Global API Server</b>
	<p>Buatlah file Config.js, Get.js, Post.js, dan Delete.js dalam folder services seperti pada Gambar 4.1</p> <p><b>Jawaban :</b></p>

	
	<p>Buka Config.js dan isikan code seperti Gambar 4.2</p> <p><b>Jawaban :</b></p> 
	<p>Buka Get.js dan isikan code seperti Gambar 4.3</p> <p><b>Jawaban :</b></p> 
	<p>Buka Post.js dan isikan code seperti Gambar 4.4</p> <p><b>Jawaban :</b></p>

```
File Edit Selection View Go Run Terminal Help Post.js - my-app - Visual Studio Co...
listArtikel.json U index.js U BlogPost.jsx U Post.js 1, U X

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

Buka Delete.js dan isikan kode seperti Gambar 4.5

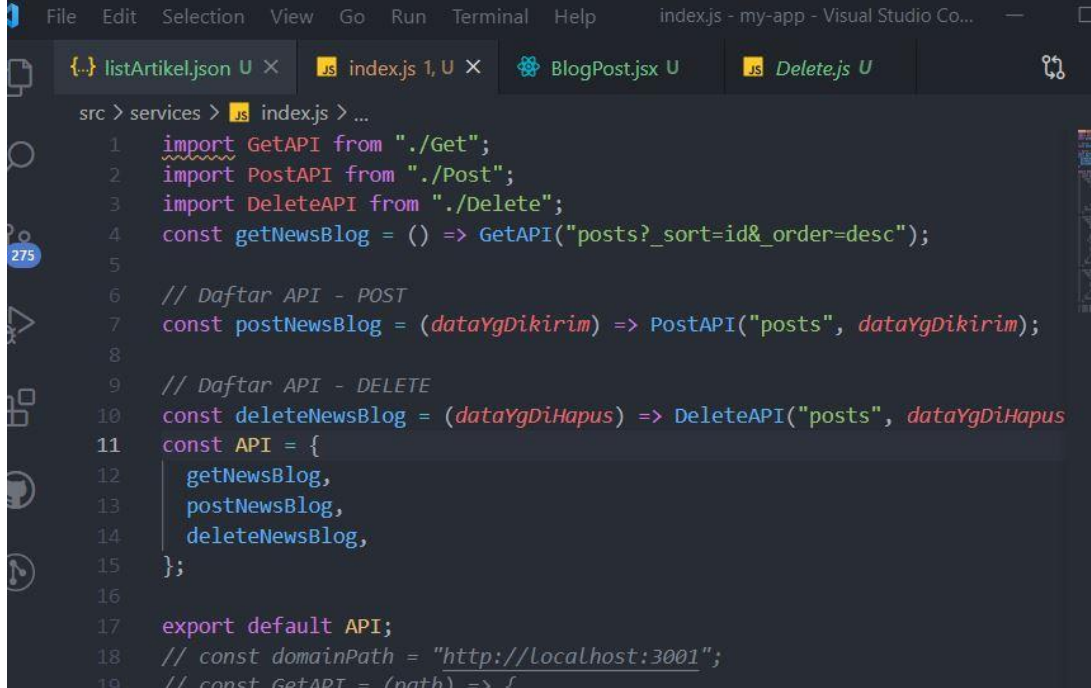
**Jawaban :**

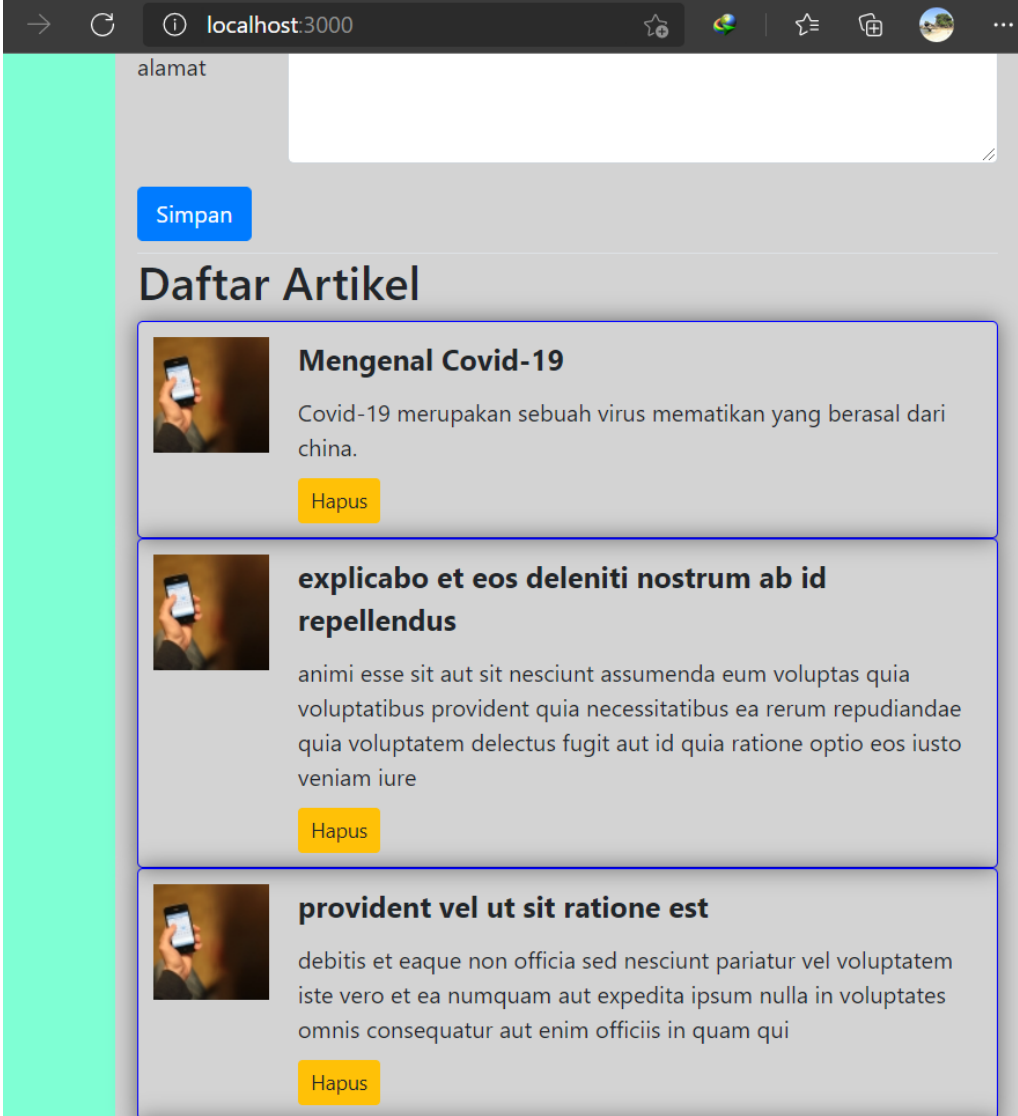
```
File Edit Selection View Go Run Terminal Help Delete.js - my-app - Visual Studio Co...
listArtikel.json U index.js U BlogPost.jsx U Delete.js 1, U X

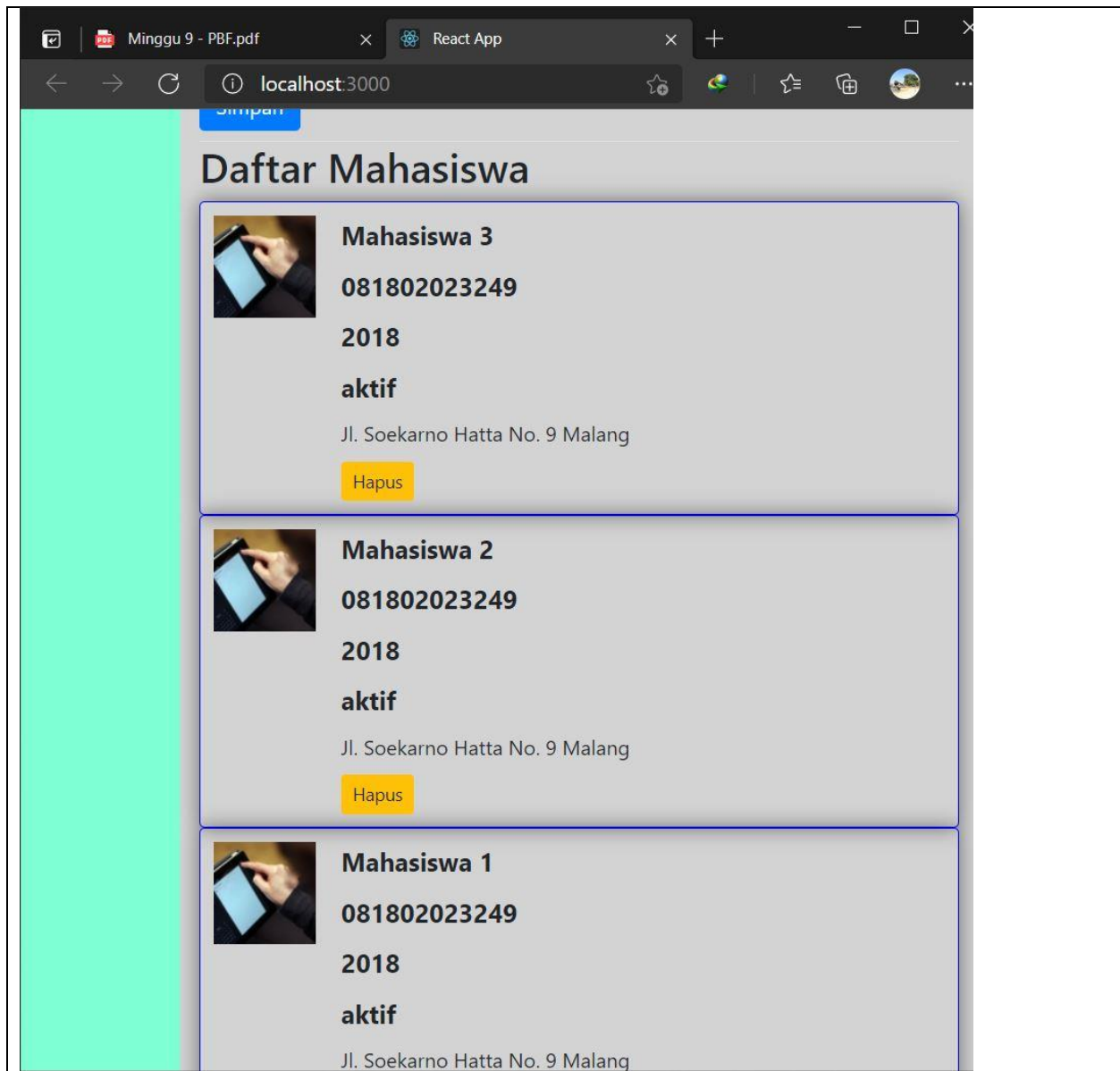
src > services > JS Delete.js > ...
1 import {domainPath} from "../Config";
2
3 const DeleteAPI = (path, data) => {
4   const promise = new Promise((resolve, reject) => {
5     fetch(`${domainPath}/${path}/${data}`, {method: 'DELETE'})
6     .then((result) => {
7       resolve(result);
8     }, (err) => {
9       reject(err);
10    })
11  })
12  return promise;
13 }
14 export default DeleteAPI;
```

Pada index.js kita edit menjadi seperti Gambar 4.6

**Jawaban :**

	 <pre>src &gt; services &gt; index.js &gt; ... 1  import GetAPI from "./Get"; 2  import PostAPI from "./Post"; 3  import DeleteAPI from "./Delete"; 4  const getNewsBlog = () =&gt; GetAPI("posts?_sort=id&amp;_order=desc"); 5 6  // Daftar API - POST 7  const postNewsBlog = (dataYgDikirim) =&gt; PostAPI("posts", dataYgDikirim); 8 9  // Daftar API - DELETE 10 const deleteNewsBlog = (dataYgDiHapus) =&gt; DeleteAPI("posts", dataYgDiHapus); 11 const API = { 12   getNewsBlog, 13   postNewsBlog, 14   deleteNewsBlog, 15 }; 16 17 export default API; 18 // const domainPath = "http://localhost:3001"; 19 // const GetAPI = (path) =&gt; {</pre>
	<p>Amati yang terjadi</p> <p><b>Jawaban :</b></p>

	
<p><b>TUGAS PRAKTIKUM</b></p> <p>Ubahlah Tugas pada Modul 4 yang sudah kalian buat dengan memanfaatkan Global API seperti praktikum yang kita lakukan pada Modul 8 ini.</p>	





← → ↻ ⓘ localhost:3000 ☆ 🌐 ☆ 🗂 🖼 ⋮

Sidebar

nama

hp


status

angkatan

alamat

Simpan

Daftar Mahasiswa



Mahasiswa 3


081802023249

2018

aktif

Jl. Soekarno Hatta No. 9 Malang

Hapus



Mahasiswa 2

--Alhamdulillah--