



Jurusan Teknologi Informasi Politeknik Negeri Malang

Tugas Minggu: Modul 4

Mata Kuliah Pemrogramman Berbasis Framework

Pengampu:

Februari 2021

Nama : Muhammad Iqbaluddin Al Huda

Kelas : TI-3E

NIM : 1841720013

Tujuan

- Interaksi dengan API(GET)
- Interaksi dengan API (fake API)
- Interaksi dengan API (DELETE)
- Interaksi dengan API(POST)

Langkah	Keterangan
1	Buka file BlogPost.jsx dan ketikkan kode seperti Gambar 1.3. Pada file index.js, lakukan import component BlogPost seperti Gambar 
2	Modifikasi file index.html pada folder "public" seperti Gambar 1.7. Cermati code program yang ada dalam gambar!.

Screenshot of a code editor showing the file structure and content of a React application named "FIRSTY-APP".

File Explorer:

- public** folder contains:
 - index.html
 - manifest.json
 - robots.txt
 - favicon.ico
 - logo192.png
 - logo512.png
- src** folder contains:
 - component** folder contains:
 - BlogPost** folder contains:
 - Post.jsx
 - HelloComponent.css
 - HelloComponent.jsx
 - container** folder contains:
 - BlogPost.jsx
 - BlogPost.css
 - footer.jsx
 - header.jsx
 - image.jsx
 - list.jsx
 - login.jsx
 - App.css
 - App.jsx

index.html Content:

```

17 <link rel="manifest" href="%PUBLIC_URL%/manifest.json" />
18 <!--
19 Notice the use of %PUBLIC_URL% in the tags above.
20 It will be replaced with the URL of the 'public' folder during the build.
21 Only files inside the 'public' folder can be referenced from the HTML.
22
23 Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC_URL%/favicon.ico" will
24 work correctly both with client-side routing and a non-root public URL.
25 Learn how to configure a non-root public URL by running `npm run build`.
26
27 -->
28 <title>React App</title>
29 </head>
30 <body>
31 <noscript>You need to enable JavaScript to run this app.</noscript>
32
33 <!--
34 This HTML file is a template.
35 If you open it directly in the browser, you will see an empty page.
36 You can add webfonts, meta tags, or analytics to this file.
37 The build step will place the bundled scripts into the <body> tag.
38
39 To begin the development, run `npm start` or `yarn start`.
40 To create a production bundle, use `npm run build` or `yarn build`.
41
42 <div className="container-fluid">
43   <div className="row">
44     <div className="col-2" id="sidebar">Sidebar</div>
45     <div className="col-10" id="content">
46       </div>
47     </div>
48   </div>
49

```

Browser Preview:

The browser shows a blank white page with the title "Blog Artikel" and a sidebar labeled "Sidebar".

3

Buka file index.css dan tambahkan code css seperti Gambar 1.9, untuk menambah sedikit style pada halaman web

The code editor shows the `index.css` file with the following content:

```
margin: 0;
font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen',
'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',
'sans-serif';
-webkit-font-smoothing: antialiased;
-moz-osx-font-smoothing: grayscale;

code {
font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',
'monospace';
}

#sidebar {
background-color: aquamarine;
}
#content {
background-color: lightgray;
}
```

The browser window shows a blank page at `localhost:3000`. The address bar indicates the page is loaded from `localhost:3000`.

4

Kita ingin sebuah website memiliki tampilan seperti pada Gambar 1.1. Dengan minimal ada gambar artikel, judul, dan deskripsi artikel. Maka contoh data dummy yang akan kita pakai bisa menggunakan data dari <http://placeimg.com> contoh wi.

Tahapan edit tampilan post artikel:

Ubah kode program untuk statefull component `BlogPost.jsx` menjadi seperti Gambar 1.11

Screenshot of a code editor (VS Code) showing a file structure and a CSS file content, followed by a screenshot of a web browser displaying a simple article list.

File Explorer:

- FIRSTY-APP
- robots.txt
- src
- component
- BlogPost
- Post.jsx
- HelloComponent.css
- HelloComponent.jsx
- container/BlogPost
- BlogPost.css
- BlogPost.jsx
- footer
- header
- image
- list
- login
- App.css
- App.js
- App.test.js
- index.css
- index.js
- logo.svg
- reportWebVitals.js
- serviceWorker.js
- setupTests.js
- .gitignore
- db.json
- listArtikel.json

Code Editor (BlogPost.css):

```
1 .artikel {
2     width: 100%;
3     padding: 10px;
4     border: 1px solid blue;
5     border-radius: 4px;
6     margin-bottom: 10px;
7     box-shadow: 0 0 16px rgba(0, 0, 0, 0.5);
8     display: flex;
9 }
10 .gambar-artikel {
11     height: 80px;
12     width: 80px;
13     margin-right: 20px;
14     vertical-align: top;
15 }
16 .gambar-artikel img {
17     width: 100%;
18     height: 100%;
19     object-fit: cover;
20 }
21 .konten-artikel {
22     flex: 1;
23 }
24 .konten-artikel div.judul-artikel {
25     font-size: 20px;
26     margin-right: 20px;
27     margin-bottom: 10px;
28 }
29 .konten-artikel p.isi-artikel
```

Browser Preview:

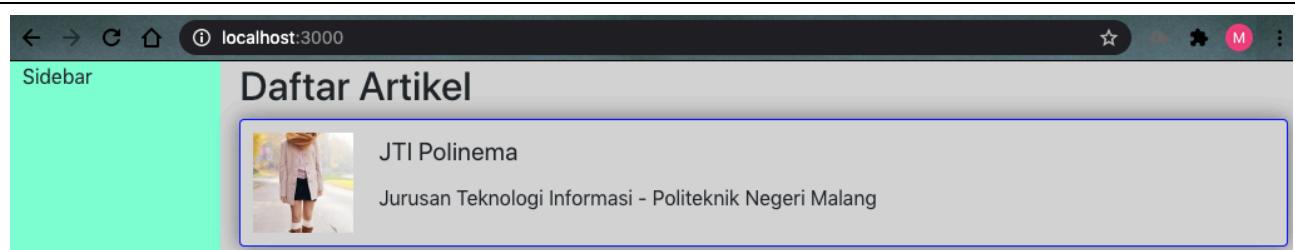
The browser shows a sidebar on the left and a main content area. The main content displays a single article card with a thumbnail image, the title "Judul Artikel", and the content "Isi Artikel".

5

The screenshot shows the VS Code interface with the following details:

- EXPLORER:** Shows the project structure: FIRSTY-APP, src, component, BlogPost, Post.jsx, HelloComponent.css, HelloComponent.jsx, container/BlogPost, BlogPost.css, and BlogPost.jsx.
- EDITOR:** The current file is BlogPost.jsx. The code is as follows:

```
src > container > BlogPost > BlogPost.jsx > default
32     render() {
33         return (
34             <div className="post-artikel">
35                 <h2>Daftar Artikel</h2>
36                 {
37                     this.state.listArtikel.map(artikel => {
38                         return <Post key={artikel.id} judul={artikel.title} i
39                     })
40                 }
41             </div>
42         )
43     }
44
45
46 export default BlogPost;
```



- 6 Edit pada statefull component BlogPost.jsx seperti pada Gambar 1.18 dan perhatikan dengan seksama akan penjelasan dibeberapa baris kode program tersebut.

	<p>The screenshot shows the VS Code interface. On the left is the Explorer sidebar with files like robots.txt, src, component, BlogPost, container/BlogPost, footer, header, image, list, login, App.css, App.js, App.test.js, index.css, index.js, logo.svg, reportWebVitals.js, serviceWorker.js, setupTests.js, .gitignore, db.json, and listArtikel.json. The right pane shows the code for BlogPost.jsx:</p> <pre> import React, { Component } from "react"; import Post from "../../component/Post"; import './BlogPost.css'; class BlogPost extends Component { state = { listArtikel: [] } ambilDataDariServerAPI = () => { fetch('http://localhost:3001/posts') .then(response => response.json()) .then(jsonHasilAmbilDariAPI => { this.setState({ listArtikel: jsonHasilAmbilDariAPI }) }) } componentDidMount() { this.ambilDataDariServerAPI() } handleHapusArtikel = () => { fetch('http://localhost:3001/posts/1', {method: 'DELETE'}) .then(res => { this.ambilDataDariServerAPI(); }) } render() { return (<div className="post-artikel"> </pre>
7	<p>Screenshot of a browser window titled 'localhost:3000'. The page displays a list of articles with small images and text snippets. The developer tools console shows the following warnings:</p> <ul style="list-style-type: none"> [HMR] Waiting for update signal from WDS... log.js:24 Warning: Invalid DOM property 'class'. Did you mean 'className'? index.js:1 <ul style="list-style-type: none"> at div at div at Post (http://localhost:3000/static/js/main.chunk.js:376:25) at div at BlogPost (http://localhost:3000/static/js/main.chunk.js:819:5) Warning: Each child in a list should have a unique "key" prop. index.js:1 <ul style="list-style-type: none"> Check the render method of `BlogPost`. See https://reactjs.org/link/warning-keys for more information. at Post (http://localhost:3000/static/js/main.chunk.js:376:25) at BlogPost (http://localhost:3000/main.0be1eb3...hot-update.js:30:5)
7	<p>27. Jika terjadi hal demikian, hal ini terjadi karena dalam react "class" dalam tag html harus ditulis menjadi "className". selain itu, pada statefull component yang dinamis, harus ada "UNIQUE KEY" pada tiap komponen yang diproses sehingga komponen perlu diberi UNIQUE KEY.</p>

```
2     render() {
3         return (
4             <div className="post-artikel">
5                 <h2>Daftar Artikel</h2>
6                 {
7                     this.state.listArtikel.map(artikel => {
8                         return <Post key={artikel.id} judul={artikel.title}>
9                     })
10                }
11            </div>
12        )
13    }
14
15    export default BlogPost;
```

The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left displays the project structure of 'FIRSTY-APP' with files like robots.txt, src, component, and various components like BlogPost and Post. The code editor on the right shows the 'Post.jsx' file with the following content:

```
src > component > BlogPost > Post.jsx > default
1   n 'react';
2
3   props) => {
4
5       <div className="artikel">
6           <div className="gambar-artikel">
7               
9           <div className="konten-artikel">
10              <div className="judul-artikel">
11                  {props.judul}
12              </div>
13              <p className="isi-artikel">{props.isi}</p>
14              <button className="btn btn-sm btn-warning" onClick={() => props.hapus}
15              </div>
16          </div>
17
18
19
20      post;
```

The screenshot shows a web application running at localhost:3000. The main content area displays a list of three articles. Each article has a small thumbnail image on the left and a snippet of Latin text on the right. The developer tools are open, showing the console tab with some messages related to HMR (Hot Module Replacement) and WDS (Webpack Dev Server).

	Article Content
sunt aut facere repellat provident occaecati excepturi optio reprehenderit quia et suscipit suscipit recusandae consequuntur expedita et cum reprehenderit molestiae ut ut quas totam nostrum rerum est autem sunt rem eveniet architecto	
qui est esse est rerum tempore vitae sequi sint nihil reprehenderit dolor beatae ea dolores neque fugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis qui aperiam non debitis possimus qui neque nisi nulla	
ea molestias quasi exercitationem repellat qui ipsa sit aut et iusto sed quo iure voluptatem occaecati omnis eligendi aut ad voluptatem doloribus vel accusantium quis pariatur molestiae porro eius odio et labore et velit aut	

1.1 Pertanyaan Praktikum 1

a. Pada langkah 8, sekarang coba kalian ganti class `container` dengan `container-fluid` atau sebaliknya pada file "[public/index.html](#)" dan lihat apa perbedaannya.

1. Tampilan seperti apa yang kalian temukan setelah mencoba mengganti nama class tersebut?
Jawab : grid berbeda
2. Apa perbedaan dari `container` dan `container-fluid` ?
Jawab : Container sendiri jika diterjemahkan adalah Kontainer, yang berarti tempat untuk menampung, atau menaruh sesuatu.
`container-fluid` memperluas untuk mengisi lebar yang tersedia

```

firsty-app — node /usr/local/bin/json-server --watch listArtikel.json --port 300...
Type s + enter at any time to create a snapshot of the database
Watching...

^C
[Iqbaluddin:firsty-app md1010$ json-server --watch listArtikel.json --port 3001

\{^_^\}/ hi!

Loading listArtikel.json
Done

Resources
http://localhost:3001/posts
http://localhost:3001/comments
http://localhost:3001/profile

Home
http://localhost:3001

Type s + enter at any time to create a snapshot of the database
Watching...

GET /posts 200 11.735 ms - 77

```

Kita cek *url resource* yang ada pada Fake API server ke browser apakah bisa diakses. Ketik url <http://localhost:3001/posts> pada browser



```
[
  {
    "userId": 1,
    "id": 1,
    "title": "worksop Amazon Web Service(AWS0 untuk para dosen jurusan teknologi informasi politeknik negeri malang",
    "body": "jumat (24/1/2020), jurusan teknologi informasi politeknik negeri malang (JTI Polinema), workshop ini diadakan untuk
menunjang performa dan.."
  },
  {
    "userId": 1,
    "id": 2,
    "title": "Pelaksanaan rapat hasil akhir yudisium semester ganjil 2019/2020",
    "body": "jumat (24/1/2020), jurusan teknologi informasi politeknik negeri malang (JTI Polinema), workshop ini diadakan untuk
menunjang performa dan.."
  }
]
```

9

1. Untuk memastikan lagi, kita edit *statefull component* **BlogPost** (Gambar 1.18) pada baris 11. Kita ganti url API dari <https://jsonplaceholder.typicode.com/posts> menjadi <http://localhost:3001/posts>
2. Simpan perubahan dan amati apa yang terjadi.

10

Sidebar

Daftar Artikel

 worksop Amazon Web Service(AWS0 untuk para dosen jurusan teknologi informasi politeknik negeri malang

jumat (24/1/2020), jurusan teknologi informasi politeknik negeri malang (JTI Polinema), workshop ini diadakan untuk menunjang performa dan..

Hapus

 Pelaksanaan rapat hasil akhir yudisium semester ganjil 2019/2020

jumat (24/1/2020), jurusan teknologi informasi politeknik negeri malang (JTI Polinema), workshop ini diadakan untuk menunjang performa dan..

Hapus

1. Buka stateless component Post. Tambahkan 1 baris kode program pada baris 10 seperti pada Gambar 3.1

```
10   <div className="judul-artikel">
11     |   {props.judul}
12   </div>
13   <p className="isi-artikel">{props.isi}</p>
14   <button className="btn btn-sm btn-warning" onClick={() => props.hapus}>
15     </div>
16   </div>
17
18
19
20 >st;
```

11

2. Kemudian pada *statefull component BlogPost*, modifikasi kode program sebelumnya sesuai dengan Gambar 3.2

EXPLORER

- FIRSTY-APP
 - robots.txt
 - src
 - component
 - BlogPost
 - Post.jsx
 - # HelloComponent.css
 - HelloComponent.jsx
 - container/BlogPost
 - # BlogPost.css
 - BlogPost.jsx
 - footer
 - header
 - image
 - list
 - login
 - # App.css
 - App.js
 - App.test.js
 - # index.css
 - index.js
 - logo.svg
 - reportWebVitals.js
 - serviceWorker.js
 - setupTests.js
 - .gitignore
 - { db.json
 - { listArtikel.json

src > container > BlogPost > BlogPost.jsx > default

```

11  ambilDataDariServerAPI = () => {
12    fetch('http://localhost:3001/posts')
13      .then(response => response.json())
14      .then(jsonHasilAmbilDariAPI => {
15        this.setState({
16          listArtikel: jsonHasilAmbilDariAPI
17        })
18      })
19    }
20
21  componentDidMount() {
22    this.ambilDataDariServerAPI()
23  }
24
25  handleHapusArtikel = () => {
26    fetch('http://localhost:3001/posts/1', {method: 'DELETE'})
27      .then(res => {
28        this.ambilDataDariServerAPI();
29      })
30  }
31
32  render() {
33    return (
34      <div className="post-artikel">
35        <h2>Daftar Artikel</h2>
36        {
37          this.state.listArtikel.map(artikel => {
38            return <Post key={artikel.id} judul={artikel.title} :>
39          })
40        }
41      </div>
42    )
43  }
44

```

3. Klik tombol hapus pada list artikel di browser. Amati apa yang terjadi

Sebelum id data didefinisi

Sidebar

localhost:3000

Daftar Artikel

	<p>worksop Amazon Web Service(AWS0 untuk para dosen jurusan teknologi informasi politeknik negeri malang</p> <p>jumat (24/1/2020), jurusan teknologi informasi politeknik negeri malang (JTI Polinema), workshop ini diadakan untuk menunjang performa dan..</p> <p>Hapus</p>
	<p>Pelaksanaan rapat hasil akhir yudisium semester ganjil 2019/2020</p> <p>jumat (24/1/2020), jurusan teknologi informasi politeknik negeri malang (JTI Polinema), workshop ini diadakan untuk menunjang performa dan..</p> <p>Hapus</p>

Console

Some messages have been moved to the Issues panel.

[HMR] Waiting for update signal from WDS... log.js:24

✖ ➔ DELETE http://localhost:3001/posts/undefined BlogPost.jsx:26 404 (Not Found)

The development server has disconnected. Refresh the page if necessary.

12

Sesudah id data definisi maka data json akan terhapus

Sidebar

Daftar Artikel

Pelaksanaan rapat hasil akhir yudisium semester ganjil 2019/2020

jumat (24/1/2020), jurusan teknologi informasi politeknik negeri malang (JTI Polinema), workshop ini diadakan untuk menunjang performa dan..

Hapus

localhost:3000

Elements Console Sources > 1 Default levels Filter View issues [HMR] Waiting for update signal from WDS... log.js:24 >

13

```
1  {
2   "posts": [
3     {
4       "userId": 1,
5       "id": 2,
6       "title": "Pelaksanaan rapat hasil akhir yudisium semester ganjil 2019/",
7       "body": "jumat (24/1/2020), jurusan teknologi informasi politeknik nego"
8     }
9   ],
10  "comments": [],
11  "profile": {
12    "name": "typicode"
13  }
14 }
```

EXPLORER

- FIRSTY-APP
 - robots.txt
 - src
 - component
 - BlogPost
 - Post.jsx
 - HelloComponent.css
 - HelloComponent.jsx
 - container/BlogPost
 - BlogPost.css
 - BlogPost.jsx
 - footer
 - header
 - image
 - list
 - login
 - App.css
 - App.js
 - App.test.js
 - index.css
 - index.js
 - logo.svg
 - reportWebVitals.js
 - serviceWorker.js
 - setupTests.js
 - .gitignore
 - db.json
 - listArtikel.json
 - OUTLINE

3.1 Pertanyaan Praktikum 3

- a. Apa yang terjadi setelah kalian klik tombol hapus?

Jawab : data json terhapus

- b. Perhatikan file **listArtikel.json**, apa yang terjadi pada file tersebut? Kenapa demikian?

14

Jawab : data yang di klik tombol hapus akan terhapus dalam data listArtikel.json

- c. Fungsi `handleHapusArtikel` itu untuk apa?

Jawab : menghapus data listArtikel.json

- d. Jelaskan perbedaan fungsi `componentDidMount()` pada Gambar 1.18 dengan fungsi `componentDidMount()` pada Gambar 3.2 ?

Jawab : tahap setelah component di render. pada tahap ini proses pemanggilan ajax dan perubahan isi statis setelah proses ajax di panggil.

15

The screenshot shows a code editor with the following details:

- EXPLORER** pane on the left showing project structure:

 - FIRSTY-APP**
 - src** folder containing `BlogPost.jsx`, `Post.jsx`, and `index.css`.
 - component** folder containing `BlogPost` and `Post`.
 - `robots.txt`
 - `index.js`
 - `App.css`
 - `App.js`
 - `reportWebVitals.js`
 - `serviceWorker.js`
 - `setupTests.js`
 - `.gitignore`
 - `db.json`
 - `listArtikel.json`

- BlogPost.jsx** file content:

```
src > container > BlogPost > BlogPost.jsx > render
26   :ch( http://localhost:3001/posts/1 , {method: 'DELETE'})
27   .then(res => {
28     |   this.ambilDataDariServerAPI();
29   })
30
31   () {
32     |   turn (
33       <div className="post-artikel">
34         <div className="form pb-2 border-bottom">
35           <div className="form-group row">
36             <label htmlFor="title" className="col-sm-2 col-form-label">Judul</label>
37             <div className="col-sm-10">
38               |   <input type="text" className="form-control" id="title" name="title" or
39             </div>
40
41             <div className="form-group row">
42               <label htmlFor="body" className="col-sm-2 col-form-label">isi</label>
43               <div className="col-sm-10">
44                 |   <textarea className="form-control" id="body" name="body" rows="3" onCh
45               </div>
46             </div>
47           </div>
48         </div>
49       </div>
50       <h2>Daftar Artikel</h2>
51       {
52         |   this.state.listArtikel.map(artikel => {
53           |       return <Post key={artikel.id} judul={artikel.title} isi={artikel.body} hap
54         })
55       }
56     </div>
57   )
58
59 }
```

16

localhost:3000

Sidebar

Judul iqbal

isi contoh

Simpan

Daftar Artikel

 worksop Amazon Web Service(AWS0 untuk para dosen jurusan teknologi informasi politeknik negeri malang
jumat (24/1/2020), jurusan teknologi informasi politeknik negeri malang (JTI Polinema), workshop ini diadakan untuk menunjang performa dan..
[Hapus](#)

 Pelaksanaan rapat hasil akhir yudisium semester ganjil 2019/2020
jumat (24/1/2020), jurusan teknologi informasi politeknik negeri malang (JTI Polinema), workshop ini diadakan untuk menunjang performa dan..
[Hapus](#)

localhost:3000

Simpan

Daftar Artikel

 worksop Amazon Web Service(AWS0 untuk para dosen jurusan teknologi informasi politeknik negeri malang
jumat (24/1/2020), jurusan teknologi informasi politeknik negeri malang (JTI Polinema), workshop ini diadakan untuk menunjang performa dan..
[Hapus](#)

 Pelaksanaan rapat hasil akhir yudisium semester ganjil 2019/2020
jumat (24/1/2020), jurusan teknologi informasi politeknik negeri malang (JTI Polinema), workshop ini diadakan untuk menunjang performa dan..
[Hapus](#)

 iqbal
contoh
[Hapus](#)

17

The screenshot shows a code editor interface with a dark theme. On the left is a file tree for a project named 'FIRSTY-APP'. The 'listArtikel.json' file is open in the main editor area. The code is a JSON object with several nested arrays and objects. It includes posts with titles like 'worksop Amazon Web Service(AWS0 untuk para dosen jurusan teknologi informasi' and 'Pelaksanaan rapat hasil akhir yudisium semester ganjil 2019/2020', and a comment section.

```
1    "posts": [
2      {
3        "userId": 1,
4        "id": 1,
5        "title": "worksop Amazon Web Service(AWS0 untuk para dosen jurusan teknologi informasi",
6        "body": "jumat (24/1/2020), jurusan teknologi informasi politeknik negeri malang (",
7      },
8      {
9        "userId": 1,
10       "id": 2,
11       "title": "Pelaksanaan rapat hasil akhir yudisium semester ganjil 2019/2020",
12       "body": "jumat (24/1/2020), jurusan teknologi informasi politeknik negeri malang (",
13     },
14   ],
15   {
16     "userId": 1,
17     "id": 1614572405670,
18     "title": "iqbal",
19     "body": "contoh"
20   },
21 ],
22   "comments": [
23     {
24       "id": 1,
25       "body": "some comment",
26       "postId": 1
27     },
28   ],
29   "profile": {
30     "name": "typicode"
31   }
32 }
```

Ln 13, Col 19 Spaces: 2 UTF-8 LF JSON ✓ ESLint Prettier ⌂ ⌂

18

Tugas

Buatlah program menggunakan Fake API (JSON Server) tentang pendataan Mahasiswa aktif/cuti/lulus di Jurusan Teknologi Informasi. Atribut-atribut yang ada dari mahasiswa adalah NIM, nama, alamat, no hp, tahun Angkatan, dan status. Buatlah aplikasi yang menggunakan API dengan method GET, DELETE, dan POST.

The screenshot displays two tabs of a React application in a code editor:

Top Tab: src > container > BlogPost > BlogPost.jsx > render

```
1 import React, { Component } from "react";
2 import Post from "../../component/BlogPost/Post";
3 import './BlogPost.css';
4 import 'bootstrap/dist/css/bootstrap.min.css';
5
6 class BlogPost extends Component {
7     state = {
8         listArtikel: [],
9         insertArtikel: {
10             userId: 1,
11             id: 1,
12             nim: '',
13             nama: '',
14             alamat: '',
15             telepon:'',
16             angkatan:'',
17             status:''
18         }
19     }
20
21     ambilDataDariServerAPI = () => {
22         fetch('http://localhost:3001/mahasiswa')
23             .then(response => response.json())
24             .then(jsonHasilAmbilDariAPI => {
25                 this.setState({
26                     listArtikel: jsonHasilAmbilDariAPI
27                 })
28             })
29     }
30
31     componentDidMount() {
32         this.ambilDataDariServerAPI()
33     }
34
35     handleHapusArtikel = () => {
36         const artikelId = event.target.value;
37         const artikelIndex = this.state.listArtikel.findIndex(item => item.id === artikelId);
38
39         if (artikelIndex !== -1) {
40             const artikelList = [...this.state.listArtikel];
41             artikelList.splice(artikelIndex, 1);
42             this.setState({ listArtikel: artikelList });
43         }
44     }
45
46     handleTambahArtikel = (event) => {
47         let formInsertArtikel = {...this.state.insertArtikel};
48         let timestamp = new Date().getTime();
49         formInsertArtikel['id'] = timestamp;
50         formInsertArtikel[event.target.name] = event.target.value;
51         this.setState({
52             insertArtikel: formInsertArtikel
53         });
54     }
55
56     handleTombolSimpan = () => {
57         fetch('http://localhost:3001/mahasiswa/', {
58             method: 'post',
59             headers: {
60                 'Accept': 'application/json',
61                 'Content-Type': 'application/json'
62             },
63             body: JSON.stringify(this.state.insertArtikel)
64         })
65             .then((Response) => {
66                 this.ambilDataDariServerAPI();
67             })
68     }
69 }
```

Bottom Tab: src > container > BlogPost > BlogPost.jsx > render

```
31     componentDidMount() {
32         this.ambilDataDariServerAPI()
33     }
34
35     handleHapusArtikel = () => {
36         fetch('http://localhost:3001/mahasiswa/1', {method: 'DELETE'})
37             .then(res => {
38                 this.ambilDataDariServerAPI();
39             })
40     }
41
42     handleTambahArtikel = (event) => {
43         let formInsertArtikel = {...this.state.insertArtikel};
44         let timestamp = new Date().getTime();
45         formInsertArtikel['id'] = timestamp;
46         formInsertArtikel[event.target.name] = event.target.value;
47         this.setState({
48             insertArtikel: formInsertArtikel
49         });
50     }
51
52     handleTombolSimpan = () => {
53         fetch('http://localhost:3001/mahasiswa/', {
54             method: 'post',
55             headers: {
56                 'Accept': 'application/json',
57                 'Content-Type': 'application/json'
58             },
59             body: JSON.stringify(this.state.insertArtikel)
60         })
61             .then((Response) => {
62                 this.ambilDataDariServerAPI();
63             })
64     }
65 }
```

REACT-APP-MODUL4-TUGAS

```

src > container > BlogPost > BlogPost.jsx > render
  <div className="form-group row">
    <label htmlFor="alamat" className="col-sm-2 col-form-label">Alamat</label>
    <div className="col-sm-10">
      <textarea className="form-control" id="alamat" name="alamat" rows="3" onChange={this.handleChange}></textarea>
    </div>
  </div>
  <div className="form-group row">
    <label htmlFor="telfon" className="col-sm-2 col-form-label">telfon</label>
    <div className="col-sm-10">
      <input type="text" className="form-control" id="telfon" name="telfon" onChange={this.handleChange}>
    </div>
  </div>
  <div className="form-group row">
    <label htmlFor="angkatan" className="col-sm-2 col-form-label">angkatan</label>
    <div className="col-sm-10">
      <input type="text" className="form-control" id="angkatan" name="angkatan" onChange={this.handleChange}>
    </div>
  </div>
  <div className="form-group row">
    <label htmlFor="status" className="col-sm-2 col-form-label">status</label>
    <div className="col-sm-10">
      <input type="text" className="form-control" id="status" name="status" onChange={this.handleChange}>
    </div>
  </div>
  <button type="submit" className="btn btn-primary" onClick={(this.handleTombolSimpan)}>Simpan</button>
</div>
<h2>Daftar Mahasiswa</h2>
<ul>
  {
    this.state.listArtikel.map(artikel => {
      return <Post key={artikel.id} nim={artikel.nim} nama={artikel.nama} alamat={artikel.alamat}>
    })
  }
</ul>

```

localhost:3000

NIM

Nama

Alamat

Telfon

Angkatan

Status

Simpan

Daftar Mahasiswa

Mahasiswa 1

Jl. Soekarno Hatta No. 9 Malang

2018

19

20

localhost:3000

status

Simpan

Daftar Mahasiswa

	Mahasiswa 1 Jl. Soekarno Hatta No. 9 Malang 2018 aktif Hapus
	Mahasiswa X Jl. Menur No. 9 Surabaya 2014 lulus Hapus

localhost:3000

Sidebar

NIM	1841720013
Nama	Muhammad Iqbaluddin
Alamat	Mojokerto
telfon	09837642
angkatan	2018
status	aktif

Simpan

Daftar Mahasiswa

	Mahasiswa 1 Jl. Soekarno Hatta No. 9 Malang 2018
---	--

The screenshot shows a web application running at `localhost:3000`. It displays two student profiles. Each profile card includes a small thumbnail image, the student's name, address, year of graduation, status, and a yellow "Hapus" button.

Profile 1:

- Mahasiswa X
- Jl. Menur No. 9 Surabaya
- 2014
- lulus
- Hapus

Profile 2:

- 1841720013
- Muhammad Iqbaluddin
- Mojokerto
- 09837642
- 2018
- aktif
- Hapus

Below the profiles, the browser's developer tools are open, specifically the "Sources" tab. It shows the file structure of the React application and the contents of the `datamahasiswa.json` file. The JSON data corresponds to the two profiles shown above.

```
REACT-APP-MODUL4-TUGAS
├── BlogPost
│   ├── Post.jsx
│   └── HelloComponent.css
└── container/BlogPost
    ├── BlogPost.css
    └── BlogPost.jsx
> footer
> header
> image
> list
> login
# App.css
JS App.js
JS App.test.js
# index.css
JS index.js
logo.svg
JS reportWebVitals.js
JS serviceWorker.js
JS setupTests.js
{ datamahasiswa.json
{ db.json
{ listArtikel.json
{ package-lock.json
{ package.json
 README.md
> OUTLINE
> TIMELINE
40
41
42
43
44 }
```

```
{
  "mahasiswa": [
    {
      "status": "lulus",
      "angkatan": 2014,
      "nama": "Mahasiswa X",
      "alamat": "Jl. Menur No. 9 Surabaya",
      "hp": "081802023523",
      "id": 1,
      "userId": 1,
      "NIM": "140823453",
      "nim": "1841720013",
      "nama": "Muhammad Iqbaluddin",
      "alamat": "Mojokerto",
      "telepon": "09837642",
      "angkatan": 2018,
      "status": "aktif"
    },
    {
      "status": "aktif",
      "angkatan": 2018,
      "nama": "Mahasiswa X",
      "alamat": "Jl. Menur No. 9 Surabaya",
      "hp": "081802023523",
      "id": 2,
      "userId": 1,
      "NIM": "140823453",
      "nim": "1841720013",
      "nama": "Muhammad Iqbaluddin",
      "alamat": "Mojokerto",
      "telepon": "09837642",
      "angkatan": 2018,
      "status": "aktif"
    }
  ],
  "comments": [
    {
      "postId": 1,
      "body": "some comment",
      "id": 1
    }
  ],
  "profile": {
    "name": "typicode"
  }
}
```

localhost:3000

status

Simpan

Daftar Mahasiswa

	Mahasiswa 1
thumbnail	Jl. Soekarno Hatta No. 9 Malang
artikel	2018
	aktif
Hapus	
	Mahasiswa X
thumbnail	Jl. Menur No. 9 Surabaya
artikel	2014
	lulus
Hapus	

```

REACT-APP-MODUL4-TUGAS
├── status
│   ├── Sir
│   └── Da
├── datamahasiswa.json
└── index.html

```

```

datamahasiswa.json
{
  "mahasiswa": [
    {
      "userId": 1,
      "id": 1,
      "NIM": "180823453",
      "nama": "Mahasiswa 1",
      "alamat": "Jl. Soekarno Hatta No. 9 Malang",
      "hp": "081802023249",
      "angkatan": 2018,
      "status": "aktif"
    },
    {
      "userId": 1,
      "id": 2,
      "NIM": "140823453",
      "nama": "Mahasiswa X",
      "alamat": "Jl. Menur No. 9 Surabaya",
      "hp": "081802023523",
      "angkatan": 2014,
      "status": "lulus"
    }
  ],
  "comments": [
    {
      "id": 1,
      "body": "some comment",
      "postId": 1
    }
  ],
  "profile": {
    "name": "typicode"
  }
}

```

<https://youtu.be/giLSqIDYV0E>

<https://github.com/iqbali922/Pemrogramman-Berbasis-Framework/tree/master/Minggu%20-Interaction%20with%20api>

-- Selamat Mengerjakan --