

Mohammad Raffi Akbar  
1606917973  
APAP – C  
write-up tutorial10

### Jawaban Pertanyaan

1. Pada setiap fungsi di Appointment.js, parameter pertama pasti memiliki prefix “\${cors}” yang merefer pada variable cors. Mengapa hal tersebut perlu dilakukan? Apa yang terjadi jika prefix tersebut dihilangkan?

Jawab:

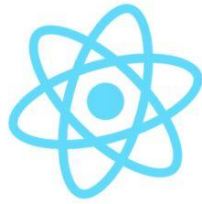
Dengan menambah prefix “\${cors}”, dimana cors itu sendiri merupakan variable berisikan url, didepan url yang ingin di-*fetch* (datanya ingin diambil), memungkinkan terjadinya *cross-origin requests* antar host yang berada di heroku. (dokumentasi <https://cors-anywhere.herokuapp.com/>)

Tanpa penambahan prefix tersebut, maka kita tidak dapat mengambil data (mendapat *response*) dari host lain yang berada di heroku.

Bila prefix tersebut dihilangkan (seperti pada gambar di bawah ini):

```
export const Appointment = {
  getAllPasien() {
    return fetch(`${baseUrl}/1/getAllPasien`, {
      method: 'GET',
    })
    .then(response => {
      return response.json()
    })
    .then(jsonResponse => {
      return jsonResponse
    })
  },
}
```

Maka tidak akan mendapat response apapun:



Fetching Data...

2. Pada fungsi `handleFormSubmit(e)` di `UpdatePasien.jsx` terdapat klausa:

```
let last = name.pop()
name.reduce((prev, next) => {
  return prev[next] = prev[next] || {};
}, dataJson)[last] = val
```

Apa yang dilakukan pada code tersebut? Jika klausa tersebut diganti dengan `dataJson[key] = val`, apa yang terjadi dengan isi variable `dataJson`?

Jawab:

Code tersebut memetakan

### Implementasi Soal Latihan

1. Tambahkan menu pada navbar untuk menampilkan daftar seluruh staf Farmasi, dan implementasikan paginya hingga menampilkan data seperti pada page daftar pasien

Jawab:

- Menambahkan menu Daftar Staff Farmasi pada Navbar

```

<nav className="navbar navbar-expand-sm navbar-dark bg-dark">
  <NavLink to="#" className="navbar-brand">
    
  </NavLink>
  <div className="collapse navbar-collapse">
    <ul className="navbar-nav mr-auto">
      <li className='nav-item'>
        <NavLink to="/" exact className="nav-link" activeClassName="active" >Home</NavLink>
      </li>
      <li className='nav-item'>
        <NavLink to="/all-pasien" exact className="nav-link" activeClassName="active">Daftar Pasien</NavLink>
      </li>
      <li className='nav-item'>
        <NavLink to="/all-staff-farmasi" exact className="nav-link" activeClassName="active">Daftar Staff Farmasi</NavLink>
      </li>
    </ul>
  </div>
</nav>

```

- Menambahkan route untuk url “/all-staff-farmasi” untuk mengakses component DaftarStaffFarmasi

```

<div className="container">
  <Switch>
    <Route path="/" exact component={Home} />
    <Route path="/all-pasien" exact component={DaftarPasien} />
    <Route path="/update-pasien/:id" exact component={UpdatePasien} />
    <Route path="/not-found" exact component={NotFound} />
    <Route path="/all-staff-farmasi" exact component={DaftarStaffFarmasi} />
    <Route path="/add-lab-result/:id" exact component={AddLabResult} />
    <Route component={NotFound} />
  </Switch>
</div>

```

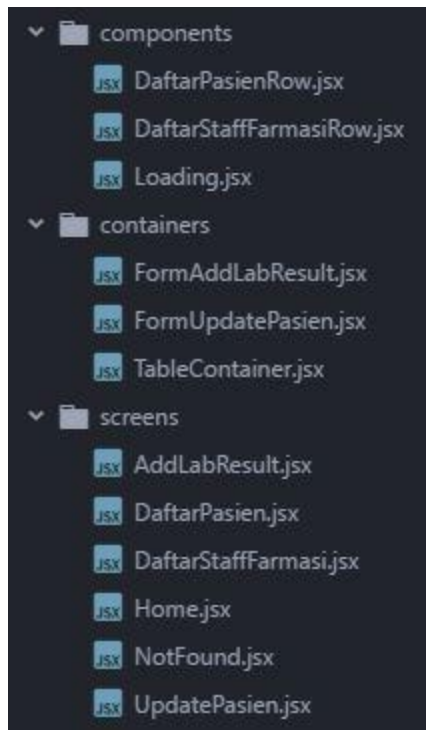
- Import component DaftarStaffFarmasi

```

import React from 'react';
import { BrowserRouter as Router, NavLink, Route, Switch } from "react-router-dom";
import './App.css';
import { DaftarPasien } from './screens/DaftarPasien';
import { Home } from './screens/Home';
import { NotFound } from './screens/NotFound';
import { UpdatePasien } from './screens/UpdatePasien';
import { DaftarStaffFarmasi } from './screens/DaftarStaffFarmasi';
import { AddLabResult } from './screens/AddLabResult';

```

- Menambahkan file DaftarStaffFarmasiRow.jsx dan DaftarStaffFarmasi.jsx



- Membuat method getAllStaffFarmasi pada Appointment.js

```
getAllStaffFarmasi() {  
  return fetch(`${cors}${baseUrl}/1/getAllStaffFarmasi`, {  
    method: 'GET',  
  })  
  .then(response => {  
    return response.json()  
  })  
  .then(jsonResponse => {  
    return jsonResponse  
  })  
},
```

- Melengkapi isi file DaftarStaffFarmasi.jsx

```
DaftarStaffFarmasi.jsx
1 import React from 'react';
2 import { DaftarPasienRow } from '../components/DaftarPasienRow';
3 import { Loading } from '../components/Loading';
4 import { TableContainer } from '../containers/TableContainer';
5 import { Appointment } from '../utils/Appointment';
6 import { DaftarStaffFarmasiRow } from '../components/DaftarStaffFarmasiRow';
7
8 export class DaftarStaffFarmasi extends React.Component {
9
10   constructor(props) {
11     super(props)
12     this.state = {
13       loading: true,
14       listStaffFarmasi: []
15     }
16
17     Appointment.getAllStaffFarmasi().then(response => {
18       this.setState({
19         loading: false,
20         listStaffFarmasi: response.result
21       })
22     })
23   }
24
25   render() {
26     if (this.state.loading) {
27       return (
28         <Loading msg="Fetching Data..." />
29       )
30     } else {
31       return (
32         <TableContainer title="Daftar Staff Farmasi" header={['Nama Staff Farmasi', 'Jenis']}>
33           <DaftarStaffFarmasiRow listStaffFarmasi={this.state.listStaffFarmasi}/>
34         </TableContainer>
35       )
36     }
37   }
38 }
```

- Melengkapi isi file DaftarStaffFarmasiRow.jsx

```

1  import React from 'react';
2  import { Link } from "react-router-dom";
3
4  export const DaftarStaffFarmasiRow = (props) => {
5    return (
6      <tbody>
7        {props.listStaffFarmasi.map(staffFarmasi => {
8          return (
9            <tr key={staffFarmasi.id}>
10              <td>{staffFarmasi.nama}</td>
11              <td>{staffFarmasi.jenis}</td>
12            </tr>
13          )
14        })}
15      </tbody>
16    )
17  }

```

2. Pada daftar pasien, tambahkan button untuk menambahkan hasil lab pasien, yang jika diklik maka akan menampilkan form untuk menginput hasil lab pasien. Implementasikan form hingga berhasil melakukan request POST ke SI-Appointment.

Jawab:

- Menambahkan route untuk url “/add-lab-result/:id” untuk mengakses component AddLabResult

```

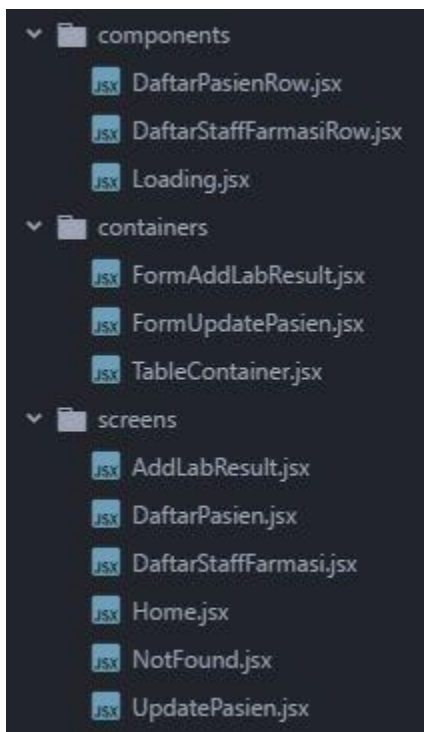
<div className="container">
  <Switch>
    <Route path="/" exact component={Home} />
    <Route path="/all-pasien" exact component={DaftarPasien} />
    <Route path="/update-pasien/:id" exact component={UpdatePasien} />
    <Route path="/not-found" exact component={NotFound} />
    <Route path="/all-staff-farmasi" exact component={DaftarStaffFarmasi} />
    <Route path="/add-lab-result/:id" exact component={AddLabResult} />
    <Route component={NotFound} />
  </Switch>
</div>

```

- Import component AddLabResult

```
import React from 'react';
import { BrowserRouter as Router, NavLink, Route, Switch } from "react-router-dom";
import './App.css';
import { DaftarPasien } from './screens/DaftarPasien';
import { Home } from './screens/Home';
import { NotFound } from './screens/NotFound';
import { UpdatePasien } from './screens/UpdatePasien';
import { DaftarStaffFarmasi } from './screens/DaftarStaffFarmasi';
import { AddLabResult } from './screens/AddLabResult';
```

- Menambahkan file FormAddLabResult.jsx dan AddLabResult.jsx



- Membuat method addLabResult pada Appointment.js

```
addLabResult(requestBody) {
  return fetch(`${cors}${baseUrl}/1/addLabResult`, {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json'
    },
    body: JSON.stringify(requestBody)
  })
  .then(response => {
    return response.json()
  })
  .then(jsonResponse => {
    return jsonResponse
  })
},
```



- Melengkapi isi file AddLabResult.jsx

```

1  import React from 'react';
2  import { Loading } from '../components/Loading';
3  import { FormAddLabResult } from '../containers/FormAddLabResult';
4  import { Appointment } from '../utils/Appointment';
5
6  export class AddLabResult extends React.Component {
7
8      constructor(props) {
9          super(props)
10         this.state = {
11             loading: true,
12             hasilLab: {
13                 "jenis" : "",
14                 "hasil" : "",
15                 "tanggalPengajuan" : "",
16                 "pasien" : {
17                     "id" : 0
18                 }
19             },
20         },
21
22         this.handleFormSubmit = this.handleFormSubmit.bind(this)
23
24         Appointment.getDetailPasien(this.props.match.params.id).then(response => {
25             if (response.status === 200) {
26                 this.setState({
27                     loading: false,
28                     hasilLab: {
29                         "jenis" : "",
30                         "hasil" : "",
31                         "tanggalPengajuan" : "",
32                         "pasien" : {
33                             "id" : response.result.id
34                         }
35                     },
36                 })
37             } else {
38                 alert('Data tidak ditemukan')
39                 this.props.history.push('/all-pasien')
40             }
41         })
42     }

```



```
41     })
42
43   }
44
45   handleFormSubmit(e) {
46     this.setState({
47       loading: true
48     })
49
50     const data = new FormData(e.target)
51     const dataJson = {}
52
53     data.forEach((val, key) => {
54       if (val !== "") {
55         let name = key.split('.');
56         if (name.length > 1) {
57           let last = name.pop()
58           name.reduce((prev, next) => {
59             return prev[next] = prev[next] || {};
60           }, dataJson)[last] = val
61         } else {
62           dataJson[key] = val
63         }
64       }
65     })
66
67     Appointment.addLabResult(dataJson).then(response => {
68       if (response.status === 200) {
69         this.setState({
70           loading: false,
71           hasilLab: response.result
72         })
73         alert(`Sukses menambah hasil lab untuk pasien dengan id ${this.state.hasilLab.pasien.id}`)
74       } else {
75         this.setState({
76           loading: false
77         })
78         alert(`Gagal menambah hasil lab untuk pasien dengan id ${this.state.hasilLab.pasien.id}`)
79       }
80     })
81   }
82 }
```

```

55 let name = key.split('.');
56 if (name.length > 1) {
57   let last = name.pop()
58   name.reduce((prev, next) => {
59     return prev[next] = prev[next] || {};
60   }, dataJson)[last] = val
61 } else {
62   dataJson[key] = val
63 }
64 }
65 })
66
67 Appointment.addLabResult(dataJson).then(response => {
68   if (response.status === 200) {
69     this.setState({
70       loading: false,
71       hasilLab: response.result
72     })
73     alert(`Sukses menambah hasil lab untuk pasien dengan id ${this.state.hasilLab.pasien.id}`)
74   } else {
75     this.setState({
76       loading: false
77     })
78     alert(`Gagal menambah hasil lab untuk pasien dengan id ${this.state.hasilLab.pasien.id}`)
79   }
80 })
81 }
82
83 render() {
84   if (this.state.loading) {
85     return (
86       <Loading msg="Fetching Data..." />
87     )
88   } else {
89     return (
90       <FormAddLabResult hasilLab={this.state.hasilLab} onSubmit={this.handleFormSubmit} />
91     )
92   }
93 }
94 }

```

- Melengkapi isi file FormAddLabResult.jsx

```

1  import React from 'react';
2
3  export const FormAddLabResult = (props) => {
4    return (
5      <form onSubmit={props.onSubmit}>
6        <h2>Tambah Hasil Lab Pasien</h2>
7
8        <input type="hidden" name="id" value={props.hasilLab.id} />
9
10       <div className="form-group">
11         <label>Jenis<span style={{ color: 'red' }}*>/span></label>
12         <input type="text" className="form-control" name="jenis" required />
13       </div>
14
15       <div className="form-group">
16         <label>Hasil<span style={{ color: 'red' }}*>/span></label>
17         <input type="text" className="form-control" name="hasil" required />
18       </div>
19
20       <div className="form-group">
21         <label>Tanggal Pengajuan<span style={{ color: 'red' }}*>/span></label>
22         <input type="date" className="form-control" name="tanggalPengajuan" required />
23       </div>
24
25       <input type="hidden" name="pasien.id" value={props.hasilLab.pasien.id} />
26
27       <button className="btn btn-success" value="submit">Tambah</button>
28     </form>
29   )
30 }

```

- Menambahkan table header 'Tambah Hasil Lab' pada file DaftarPasien.jsx

```

render() {
  if (this.state.loading) {
    return (
      <Loading msg="Fetching Data..." />
    )
  } else {
    return (
      <TableContainer title="Daftar Pasien" header={['Nama Pasien', 'Status Pasien', 'Aksi', 'Tambah Hasil Lab']}>
        <DaftarPasienRow listPasien={this.state.listPasien} />
      </TableContainer>
    )
  }
}

```