

TUGAS MINGGU 4
PEMROGRAMAN APLIKASI MOBILE
REDUX



Oleh :
Tamara Damayanti (118140069)
PAM RB

PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK ELEKTRO, INFORMATIKA DAN SISTEM FISIS
INSTITUT TEKNOLOGI SUMATERA
2021

PEMBAHASAN

Sebelum membuat aplikasi sederhana dengan menggunakan redux, maka lakukan penginstall-an terlebih dahulu, seperti pada gambar di bawah ini:

```
C:\Users\warna>npx create-react-app kalkulator-redux
npx: installed 67 in 55.646s

Creating a new React app in C:\Users\warna\kalkulator-redux.

Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts with cra-template...

[Progress bar] .....] - finalize:@babel/plugin-transform-exponentiation-operator: sill finalize C:\Users\warna\kalkulator-redux\node_modules\@babel\plugin-transform-exp
```

Tunggu dahulu sampai proses instal selesai. Jika sudah selesai, bisa Kembali melakukan proses install selanjutnya seperti gambar di bawah ini:

```
Command Prompt
Found 0 vulnerabilities

Created git commit.

Success! Created kalkulator-redux at C:\Users\warna\kalkulator-redux
Inside that directory, you can run several commands:

  npm start
    Starts the development server.

  npm run build
    Bundles the app into static files for production.

  npm test
    Starts the test runner.

  npm run eject
    Removes this tool and copies build dependencies, configuration files
    and scripts into the app directory. If you do this, you can't go back!

We suggest that you begin by typing:

  cd kalkulator-redux
  npm start

Happy hacking!

C:\Users\warna> npm install --save redux react-redux
npm WARN saveError ENOENT: no such file or directory, open 'C:\Users\warna\package.json'
npm WARN enoent ENOENT: no such file or directory, open 'C:\Users\warna\package.json'
npm WARN react-redux@7.2.2 requires a peer of react@^16.8.3 || ^17 but none is installed. You must install peer dependencies yourself.
npm WARN warna No description
npm WARN warna No repository field.
npm WARN warna No README data
npm WARN warna No license field.

+ redux@4.0.5
+ react-redux@7.2.2
added 7 packages from 5 contributors, updated 1 package and audited 12 packages in 31.273s
Found 0 vulnerabilities
```

Jika sudah muncul tampilan seperti pada gambar di atas, maka langkah selanjutnya adalah masuk ke folder yang sudah dibuat dengan perintah cd <nama folder>.

Jika sudah masuk ke folder tersebut, maka masukkan perintah code . untuk bisa membuat kode program di teks editor.

Pertama, buat folder public yang berisi file index.html, yang mana file ini digunakan sebagai tampilan agar menarik. Berikut adalah isi programnya:

```
public > indexhtml > html > body
1 <html lang="en">
2 <head>
3 <meta charset="utf-8">
4 <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
5 <meta name="theme-color" content="#000000">
6 <!-- Bootstrap CSS -->
7 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCw98/SFnGE8fJT3
8 <!-- font awesome -->
9 <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.3.1/css/all.css" integrity="sha384-mzrmE5qonljUremFsqc01S846JvR057bZ
10 <style>
11 .screen {
12 font-size: 40px;
13 text-align: right;
14 padding: 10px;
15 margin: 10px 0;
16 border: 1px solid gainsboro;
17 border-radius: 3px;
18 width: 100%;
19 display: block;
20 }
21 </style>
22 <title>Kalkulator</title>
23 </head>
24 <body>
25 <noscript>
26 Jalankan aplikasi ini.
27 </noscript>
28 <div class="container" id="root"></div>
29 </body>
30 </html>
```

Kedua, buat folder src dengan 4 file berekstensi .js. folder pertama ada actions dengan file index.js didalamnya. Folder actions ini digunakan untuk melakukan perubahan state. Berikut kode program dari index.js:

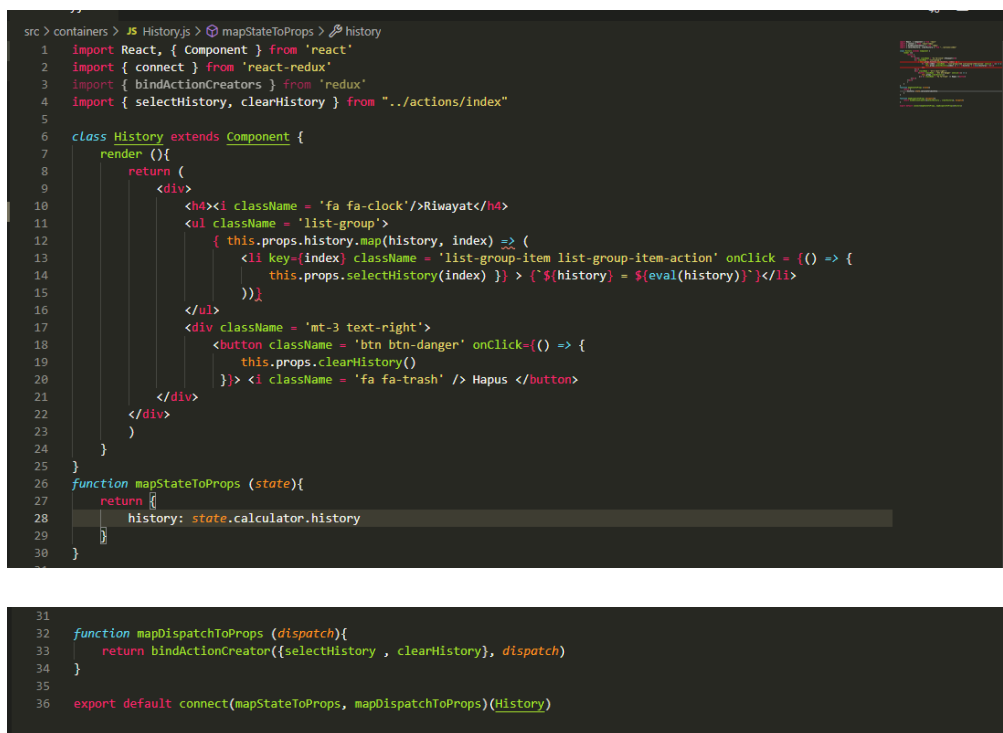
```
src > actions > JS indexjs > clearHistory
1 export const PRESS_BUTTON = 'PRESS_BUTTON'
2 export const PRESS_EQUALS = 'PRESS_EQUALS'
3 export const PRESS_RESET = 'PRESS_RESET'
4 export const SELECT_HISTORY = 'SELECT_HISTORY'
5 export const CLEAR_HISTORY = 'CLEAR_HISTORY'
6
7 export function pressButton (value){
8 console.log('PRESSED BUTTON : ', value)
9 return {
10 type : PRESS_BUTTON,
11 value
12 }
13 }
14
15 export function pressEqual () {
16 console.log('PRESSED BUTTON : equals')
17 return {
18 type : PRESS_EQUALS
19 }
20 }
21
22 export function pressReset () {
23 console.log('PRESSED RESET')
24 return {
25 type : PRESS_RESET
26 }
27 }
28
29 export function selectHistory (id) {
30 console.log('SELECT HISTORY', id)
31 return {
32 type : SELECT_HISTORY,
33 id
34 }
35 }
36
37 export function clearHistory() {
38 console.log('CLEAR_HISTORY')
39 return {
40 type : CLEAR_HISTORY
41 }
42 }
```

Folder kedua ada folder components dengan isi file bernama Apps.js. file Apps.js inilah yang akan memanggil container-container untuk disusun dalam tampilan nantinya. Kode program dapat dilihat pada gambar di bawah ini:



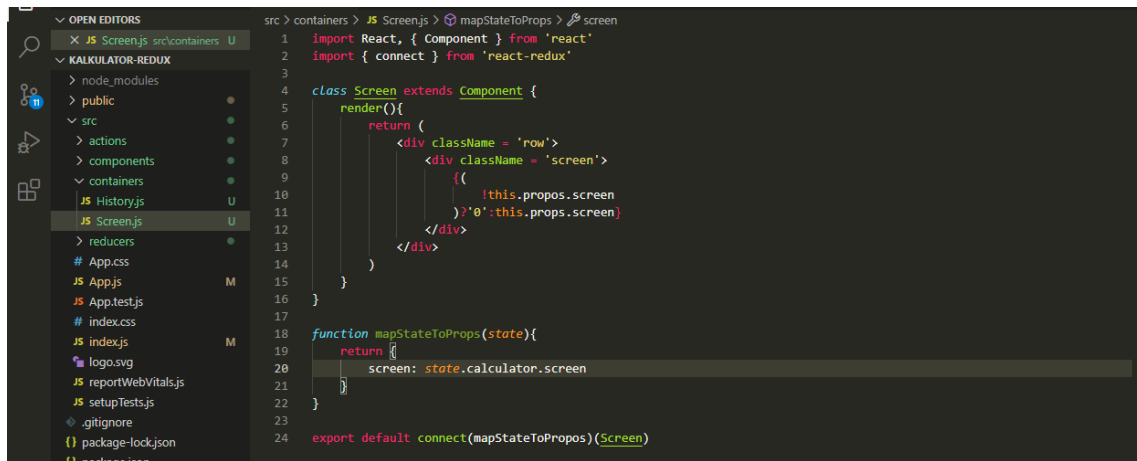
```
src > components > JS Appsjs > Apps > render
1 import React, { Component } from 'react'
2 import Screen from '../containers/Screen'
3 import History from '../containers/History'
4
5 class Apps extends Component {
6   render() {
7     return (
8       <div className = 'row'>
9         <div className = 'col-sm-12'>
10          <h3 className = 'mt-4'><i className = 'fa fa-calculator' /> Kalkulator </h3>
11        </div>
12        <div className = 'col-sm-8'>
13          <Screen />
14        </div>
15        <div className = 'col-sm-8'>
16          <h5>Button</h5>
17          <ButtonList />
18        </div>
19        <div className = 'col-sm-4'>
20          <History />
21        </div>
22      </div>
23    )
24  }
25 }
26
27 export default Apps
```

Selanjutnya ada folder containers dengan 2 file di dalamnya. File pertama bernama History.js, yang mana nantinya akan menampilkan Riwayat perhitungan. Berikut adalah kode programnya:



```
src > containers > JS History.js > mapStateToProps > history
1 import React, { Component } from 'react'
2 import { connect } from 'react-redux'
3 import { bindActionCreators } from 'redux'
4 import { selectHistory, clearHistory } from '../actions/index'
5
6 class History extends Component {
7   render () {
8     return (
9       <div>
10        <h4><i className = 'fa fa-clock' /> Riwayat</h4>
11        <ul className = 'list-group'>
12          { this.props.history.map(history, index) => (
13            <li key={index} className = 'list-group-item list-group-item-action' onClick = {() => {
14              this.props.selectHistory(index) }} > {`${history} = ${eval(history)}`}</li>
15          )}
16        </ul>
17        <div className = 'mt-3 text-right'>
18          <button className = 'btn btn-danger' onClick={() => {
19            this.props.clearHistory()
20          }} > <i className = 'fa fa-trash' /> Hapus </button>
21        </div>
22      </div>
23    )
24  }
25 }
26
27 function mapStateToProps (state){
28   return {
29     history: state.calculator.history
30   }
31 }
32
33 function mapDispatchToProps (dispatch){
34   return bindActionCreators({selectHistory , clearHistory}, dispatch)
35 }
36
37 export default connect(mapStateToProps, mapDispatchToProps)(History)
```

File kedua yakni file Screen.js, yang mana file inilah yang akan menampilkan layer perhitungan. Kode program dapat dilihat pada gambar di bawah ini:



```
1 import React, { Component } from 'react'
2 import { connect } from 'react-redux'
3
4 class Screen extends Component {
5   render(){
6     return (
7       <div className = 'row'>
8         <div className = 'screen'>
9           {
10             !this.props.screen
11             ? '0':this.props.screen
12           }
13         </div>
14       </div>
15     )
16   }
17 }
18
19 function mapStateToProps(state){
20   return {
21     screen: state.calculator.screen
22   }
23 }
24 export default connect(mapStateToProps)(Screen)
```

Kemudian folder terakhir yang ada di dalam folder src yakni reducer. Di dalam reducer itu sendiri terdapat file