TUGAS INDIVIDU

REDUX dan HOC

Disusun sebagai

MATA KULIAH : PEMROGRAMAN BERBASIS FRAMEWORK

Oleh:

Arifanny Ramadhan Sukma

1741720217 / TI-3A

04

PROGRAM STUDI D-IV TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

2020

Praktikum :
1. Buat dan install redux

2. Install Form npm install --save redux react-redux redux-form

3. Ubah file index.html pada folder /public/index.html

```html
<!DOCTYPE html> <html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1, shrink-to-fit=no">
    <meta name="theme-color" content="#000000">
    <link rel="manifest" href="%PUBLIC_URL%/manifest.json">
    <link rel="shortcut icon" href="%PUBLIC_URL%/favicon.ico">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/lib
s/bulma/0.6.2/css/bulma.min.css">
    <title>React App</title>
  </head>
  <body>
    <noscript>
      You need to enable JavaScript to run this app.
    </noscript>
    <div id="root"></div>
  </body>
</html>
```

4. Ubah file App.js pada folder /src/App.js

```javascript
import React, { Component } from 'react';
import { reduxForm, Field } from 'redux-form';
import logo from './logo.svg'; import './App.css';

let SignInForm = props => {
  const { handleSubmit } = props;
    return <form onSubmit={handleSubmit} className="form">
```

```jsx
          <div className="field">
            <div className="control">
              <Field name="firstName" component={renderField} type="text"
label="First Name"/>
            </div>
          </div>

          <div className="field">
            <div className="control">
              <Field name="lastName" component={renderField} type="text" la
bel="Last Name"/>
            </div>
          </div>

          <div className="field">
            <div className="control">
              <Field name="email" component={renderField} type="email" lab
el="Email Address"/>
            </div>
          </div>

          <div className="field">
            <div className="control">
              <Field name="age" component={renderField} type="number" labe
l="Age"/>
            </div>
          </div>

          <div className="field">
            <div className="control">
              <label className="label">Proficiency</label>
                <div className="select">
                  <Field className="input" name="proficiency" component="s
elect">
                    <option />
                      <option value="beginner">Beginner Dev</option>

                        <option value="intermediate">Intermediate Dev</opt
ion>

                          <option value="expert">Expert Dev</option>

                  </Field>
                </div>
            </div>
          </div>

          <div className="field">
            <div className="control">
```

```jsx
              <label className="label">Gender</label>
                <label className="radio">
                  <Field name="gender" component="input" type="radio" valu
e="male" />
                  {' '}
                  Male
              </label>
                <label className="radio">
                  <Field name="gender" component="input" type="radio" va
lue="female" />
                  {' '}
                  Female
              </label>
          </div>
        </div>

        <div className="field">
        <div className="control">
        <label className="checkbox">
        <Field name="saveDetails" id="saveDetails" component="input" typ
e="checkbox"/>
        Save Details
        </label>
        </div>
        </div>

        <div className="field">
          <div className="control">
            <label className="label">Message</label>
              <Field className="textarea" name="message" component="text
area" />
          </div>
        </div>

        <div className="field">
          <div className="control">
            <button className="button is-link">Submit</button>
          </div>
        </div>

  </form>;
};

const validate = val => {
  const errors = {};
  if (!val.firstName) {
    console.log('First Name is required');
    errors.firstName = 'Required';   }
```

```jsx
    if (!val.lastName) {
      console.log('Last Name is required');
      errors.lastName = 'Required';
    }
    if (!val.email) {
      console.log('email is required');
      errors.email = 'Required';
    } else if (!/^.+@.+$/i.test(val.email)) {
      console.log('email is invalid');
      errors.email = 'Invalid email address';
    }
    if (!val.age) {
      errors.age = 'Required'
    } else if (isNaN(Number(val.age)))
    {
      errors.age = 'Must be a number'
    } else if (Number(val.age) < 18) {
      errors.age = 'Sorry, you must be at least 18 years old'
    }   return errors;
  };

const renderField = ({ input, label, type, meta: { touched, error, war
ning } }) => (
<div>
  <div className="control">
  <label className="field">{label}</label>
  <input className="input" {...input} placeholder={label} type={type}/
>
  {touched && ((error && <span>{error}</span>) || (warning && <span>{w
arning}</span>))}
  </div>
  </div>
  )
  SignInForm = reduxForm({
    form: 'signIn',
    validate, })(SignInForm);

class App extends Component {

  handleSignIn = values => {
    console.log(values);   };

  render() {
    return (
      <div className="App">
        <header className="App-header">
          <img src={logo} className="App-
logo" alt="logo" />
```

```
                <h1 className="App-title">Welcome to React x redux-
form</h1>
        </header>
        <div className="container">
          <p className="App-intro">
            Contact Form
          </p>
          <SignInForm onSubmit={this.handleSignIn} />
      </div>
    </div>
    );
  }
}

export default App;
```

5. Ubah file /src/index.js

```
import React from 'react';
import ReactDOM from 'react-dom';

import { createStore, combineReducers } from 'redux';
import { Provider } from 'react-redux';
import { reducer as formReducer } from 'redux-form';

import './index.css';
import App from './App';
import registerServiceWorker from './registerServiceWorker.js';

const rootReducer = combineReducers({
  form: formReducer,
});

const store = createStore(rootReducer);

ReactDOM.render(
<Provider store={store}>
<App />
</Provider>,
document.getElementById('root') );

registerServiceWorker();
```

6. Running hasilnya dan buatlah laporan