

**PEMROGRAMAN FRAMEWORK**

**MODUL 6: Redux**

Oleh:

**MUHAMMAD IQBAL KHARIS FIRISMANDA / 1741720121**

**TI-3A/21**



**PROGRAM STUDI D4 TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

**2020**

## PRAKTIKUM

1. Buat Folder dalam src yaitu Action
2. Beri nama actionCreator

```
import { ADD_TODO, REMOVE_TODO, TOGGLE_TODO, SET_VISIBILITY_FILTER } from './actionTypes'
let TodoId = 2
export const addTodo = text => ({
  type: ADD_TODO,
  id: TodoId++,
  text
})
export const deleteTodo = (id) => ({
  type: REMOVE_TODO,
  id: id
})
export const toggleTodo = (id) => ({
  type: TOGGLE_TODO,
  id: id
})
export const setVisibilityFilter = filter => ({
  type: SET_VISIBILITY_FILTER,
  filter
})
```

3. Masih dalam Folder Action buatlah file actionTypes.js

```
export const ADD_TODO = 'ADD_TODO'
export const TOGGLE_TODO = 'TOGGLE_TODO'
export const REMOVE_TODO = 'REMOVE_TODO'
export const SHOW_ALL = 'SHOW_ALL'
export const SHOW_COMPLETED = 'SHOW_COMPLETED'
export const SHOW_ACTIVE = 'SHOW_ACTIVE'
export const SET_VISIBILITY_FILTER = 'SET_VISIBILITY_FILTER'
```

4. Kemudian dalam folder src buatlah folder dengan nama containers dan buatlah file dengan judul CreateTodo

```
import React, { Component } from 'react'
import { connect } from 'react-redux'
import { addTodo } from '../Action/actionCreator'
import { bindActionCreators } from 'redux'

class CreateTodo extends Component {
  constructor(props) {
    super(props)
    this.state = {
      todotext: ''
    }
    this.onChangeTodoText = this.onChangeTodoText.bind(this)
  }
  onChangeTodoText(e) {
    this.setState({
      todotext: e.target.value
    })
  }
  render() {
    return (
      <div className="form-group row">
        <div className="col-sm-10">
          <input onChange={this.onChangeTodoText} value={this.state.todotext} type="text" className="form-control"
            id="inputEmail3" placeholder="add todo here" />
          <button type="button" onClick={() => this.setState({ todotext: '' })} style={{ marginTop: "25px",
            marginRight: "15px" }} className="btn btn-danger">Cancel</button>
          <button type="button" onClick={() => { this.props.addTodo(this.state.todotext); this.setState({ todotext: '' })
            }} style={{ marginTop: "25px" }} className="btn btn-success">Add Todo</button>
        </div>
      </div>
    );
  }
}
```

5. Masih dalam folder containers buatlah file dengan judul Table.js

```
import React, { Component } from "react";
import { connect } from "react-redux";
import {
  deleteTodo,
  toggleTodo,
  setVisibilityFilter
} from "../Action/actionCreator";
import { SHOW_ALL, SHOW_COMPLETED, SHOW_ACTIVE } from "../Action/actionsTypes";
import { bindActionCreators } from "redux";

class Table extends Component {
  render() {
    return (
      <div className="col-lg-10 offset-lg-1 col-md-10 col-sm-12 col-xs-12">
        <nav style={{ marginTop: "60px" }}>
          <ol className="breadcrumb">
            <li
              className={"breadcrumb-item " + (this.props.visibilityFilter === SHOW_ALL ? 'active' : '')}
              onClick={() => this.props.setVisibilityFilter(SHOW_ALL)}
            >
              All
            </li>
            <li
              className={"breadcrumb-item " + (this.props.visibilityFilter === SHOW_COMPLETED ? 'active' : '')}
              onClick={() => this.props.setVisibilityFilter(SHOW_COMPLETED)}
            >
              Completed
            </li>
            <li
              className={"breadcrumb-item " + (this.props.visibilityFilter === SHOW_ACTIVE ? 'active' : '')}
              onClick={() => this.props.setVisibilityFilter(SHOW_ACTIVE)}
            >
              Active
            </li>
          </ol>
        </nav>
      </div>
    );
  }
}
```

6. Edit index.js dengan source code berikut ini

```
import React from 'react';
import ReactDOM from 'react-dom';
import { Provider } from 'react-redux'
import { createStore, applyMiddleware, compose } from 'redux'
import MainReducer from './reducers/MainReducer'
import './index.css';
import App from './App';
import * as serviceWorker from './serviceWorker';

const store = compose(window.devToolsExtension ? window.devToolsExtension() : f => f)(createStore)(MainReducer)

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

// If you want your app to work offline and load faster, you can change
// unregister() to register() below. Note this comes with some pitfalls.
// Learn more about service workers: https://bit.ly/CRA-PWA
serviceWorker.register();
```

7. Buat folder dalam src dengan nama reducers dan beri nama file FilterReducer.js dan ketikkan program dibawah ini:

```
ny-app-redux > src > reducers > JS FilterReducer.js > [⌘] default
1  import {
2    |   SHOW_ALL,
3    |   SET_VISIBILITY_FILTER
4  } from "../Action/actionTypes";
5  const visibilityFilter = (state = SHOW_ALL, action) => {
6    |   switch (action.type) {
7    |     |   case SET_VISIBILITY_FILTER:
8    |     |     |   return action.filter;
9    |     |   default:
10    |     |     |   return state;
11    |   }
12  };
13  export default visibilityFilter;
```

8. Masih dalam folder reducers buatlah file dengan nama TodoReducer.js

```
import { ADD_TODO, REMOVE_TODO, TOGGLE_TODO } from '../Action/actionTypes'
const INITIAL_DATA = []
const TodoReducer = (state = INITIAL_DATA, action) => {
  switch (action.type) {
    case ADD_TODO:
      return [
        ...state, {
          id: action.id,
          text: action.text,
          completed: false,
        }
      ]
    case TOGGLE_TODO:
      return state.map(todo =>
        (todo.id === action.id)
          ? { ...todo, completed: !todo.completed }
          : todo
      )
    case REMOVE_TODO:
      const numIndex = parseInt(action.id)
      return state.filter(todo => todo.id !== numIndex);
    default:
      return state
  }
}
export default TodoReducer
```

9. Buat file dengan nama MainReducer.js dan simpan

```
y-app-redux > src > reducers > MainReducer.js > default
1 import { combineReducers } from 'redux'
2 import todos from './TodoReducer'
3 import visibilityFilter from './FilterReducer'
4 export default combineReducers({
5   todos,
6   visibilityFilter
7 })
```

10. Running semua program dengan npm start dan liat hasilnya dan laporkan didalam laporan

