

LAPORAN MATA KULIAH  
PEMROGAMAN BERBASIS FRAMEWORK  
“ MODUL 6 ”



Nama : Ivan Crisnanda

NIM : 1741720024

Kelas : TI-3C

PROGRAM STUDI D-IV TEKNIK INFORMATIKA

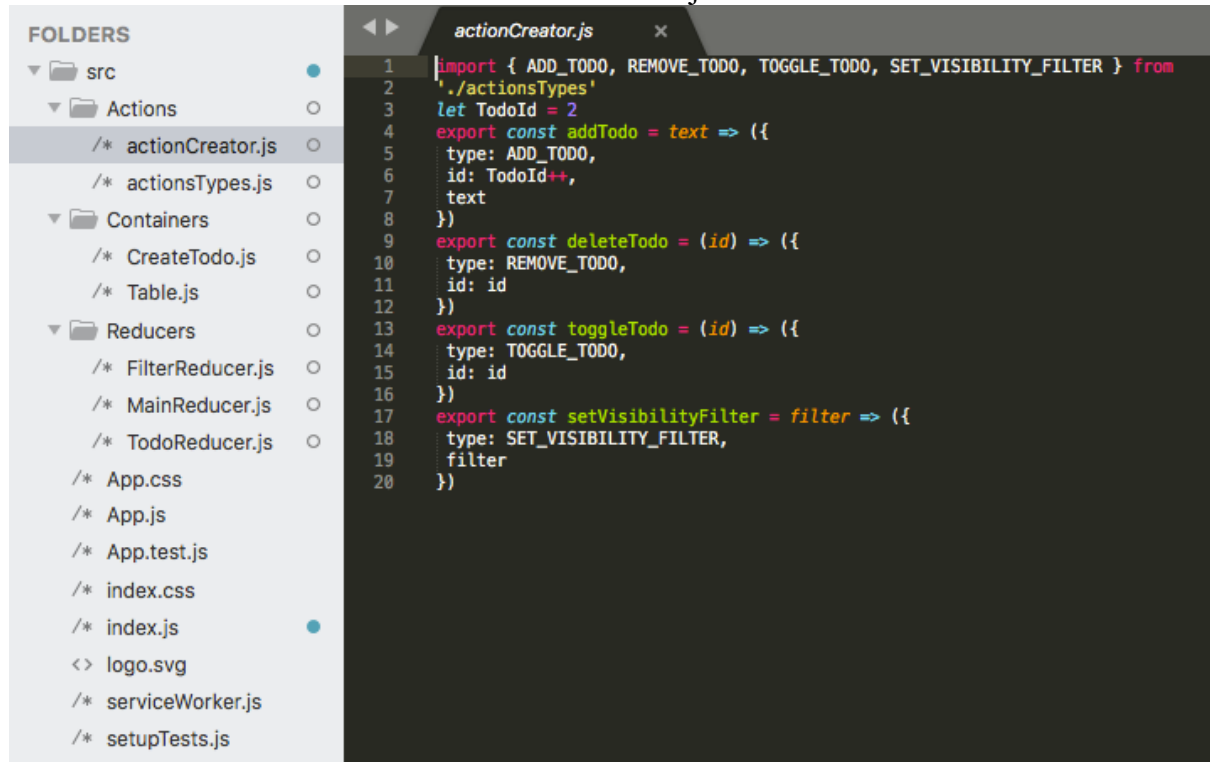
JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

2020

# Praktikum

## actionCreator.js



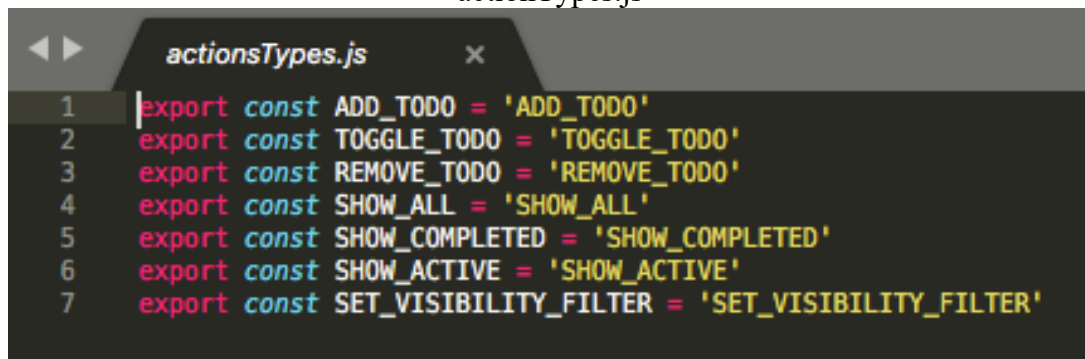
The screenshot shows a code editor with a file explorer on the left and a code editor on the right. The file explorer lists the following files and folders:

- src
  - Actions
    - /\* actionCreator.js
    - /\* actionTypes.js
  - Containers
    - /\* CreateTodo.js
    - /\* Table.js
  - Reducers
    - /\* FilterReducer.js
    - /\* MainReducer.js
    - /\* TodoReducer.js
  - /\* App.css
  - /\* App.js
  - /\* App.test.js
  - /\* index.css
  - /\* index.js
  - <> logo.svg
  - /\* serviceWorker.js
  - /\* setupTests.js

The code editor on the right shows the content of `actionCreator.js`:

```
1 import { ADD_TODO, REMOVE_TODO, TOGGLE_TODO, SET_VISIBILITY_FILTER } from
2 './actionTypes'
3 let TodoId = 2
4 export const addTodo = text => ({
5   type: ADD_TODO,
6   id: TodoId++,
7   text
8 })
9 export const deleteTodo = (id) => ({
10   type: REMOVE_TODO,
11   id: id
12 })
13 export const toggleTodo = (id) => ({
14   type: TOGGLE_TODO,
15   id: id
16 })
17 export const setVisibilityFilter = filter => ({
18   type: SET_VISIBILITY_FILTER,
19   filter
20 })
```

## actionTypes.js



The screenshot shows a code editor with the `actionTypes.js` file open. The code defines several action types as constants:

```
1 export const ADD_TODO = 'ADD_TODO'
2 export const TOGGLE_TODO = 'TOGGLE_TODO'
3 export const REMOVE_TODO = 'REMOVE_TODO'
4 export const SHOW_ALL = 'SHOW_ALL'
5 export const SHOW_COMPLETED = 'SHOW_COMPLETED'
6 export const SHOW_ACTIVE = 'SHOW_ACTIVE'
7 export const SET_VISIBILITY_FILTER = 'SET_VISIBILITY_FILTER'
```

## CreateTodo.js

FOLDERS

- src
  - Actions
    - /\* actionCreator.js
    - /\* actionTypes.js
  - Containers
    - /\* CreateTodo.js
    - /\* Table.js
  - Reducers
    - /\* FilterReducer.js
    - /\* MainReducer.js
    - /\* TodoReducer.js
  - /\* App.css
  - /\* App.js
  - /\* App.test.js
  - /\* index.css
  - /\* index.js
  - <> logo.svg
  - /\* serviceWorker.js
  - /\* setupTests.js

CreateTodo.js

```

1 import React, {Component} from 'react'
2 import { connect } from 'react-redux'
3 import { addTodo } from '../Actions/actionCreator'
4 import { bindActionCreators } from 'redux'
5 class CreateTodo extends Component {
6   constructor(props){
7     super(props)
8     this.state = {
9       todotext: '',
10     }
11     this.onChangeTodoText = this.onChangeTodoText.bind(this)
12   }
13   onChangeTodoText(e){
14     this.setState({
15       todotext: e.target.value
16     })
17   }
18   render(){
19     return (
20       <div className="form-group row">
21         <div className="col-sm-10">
22           <input onChange={this.onChangeTodoText} value={this.state.todotext}
23             type="text" className="form-control" id="inputEmail3" placeholder="add todo here"/>
24           <button type="button" onClick={ () => this.setState({ todotext: '' }) }
25             style={{marginTop: "25px", marginRight: "15px"}} className="btn btn-danger">Cancel</button>
26           <button type="button" onClick={() => {
27             this.props.addTodo(this.state.todotext); this.setState({ todotext: '' }) } }
28             style={{marginTop: "25px"}} className="btn btn-success">Add Todo</button>
29         </div>
30       </div>
31     );
32   }
33 }
34 const mapDispatchToProps = (dispatch) => {
35   return bindActionCreators({
36     addTodo
37   }, dispatch)
38 }
39 export default connect(null, mapDispatchToProps)(CreateTodo)

```

## Table.js

Table.js

```

1 import React, { Component } from "react";
2 import { connect } from "react-redux";
3 import {
4   deleteTodo,
5   toggleTodo,
6   setVisibilityFilter
7 } from "../actions/actionCreator";
8 import { SHOW_ALL, SHOW_COMPLETED, SHOW_ACTIVE } from "../actions/actionTypes";
9 import { bindActionCreators } from "redux";
10 class Table extends Component {
11   render() {
12     return (
13       <div className="col-lg-10 offset-lg-1 col-md-10 col-sm-12 col-xs-12">
14         <nav style={{ marginTop: "60px" }}>
15           <ol className="breadcrumb">
16             <li
17               className={"breadcrumb-item "+ (this.props.visibilityFilter === SHOW_ALL ?
18                 'active' : '') }
19               onClick={() => this.props.setVisibilityFilter(SHOW_ALL)}
20             >
21               All
22             </li>
23             <li
24               className={"breadcrumb-item "+ (this.props.visibilityFilter ===
25                 SHOW_COMPLETED ? 'active' : '') }
26               onClick={() => this.props.setVisibilityFilter(SHOW_COMPLETED)}
27             >
28               Completed
29             </li>
30             <li
31               className={"breadcrumb-item "+ (this.props.visibilityFilter === SHOW_ACTIVE ?
32                 'active' : '') }
33               onClick={() => this.props.setVisibilityFilter(SHOW_ACTIVE)}
34             >
35               Active
36             </li>
37           </ol>
38         </nav>
39         {this.props.todos.length !== 0 ? (
40           <table
41             style={{ marginTop: "60px" }}
42             className="table table-hover table-dark"
43           >
44             <thead>
45               <tr>
46                 <th scope="col">Todos</th>
47                 <th scope="col">Actions</th>

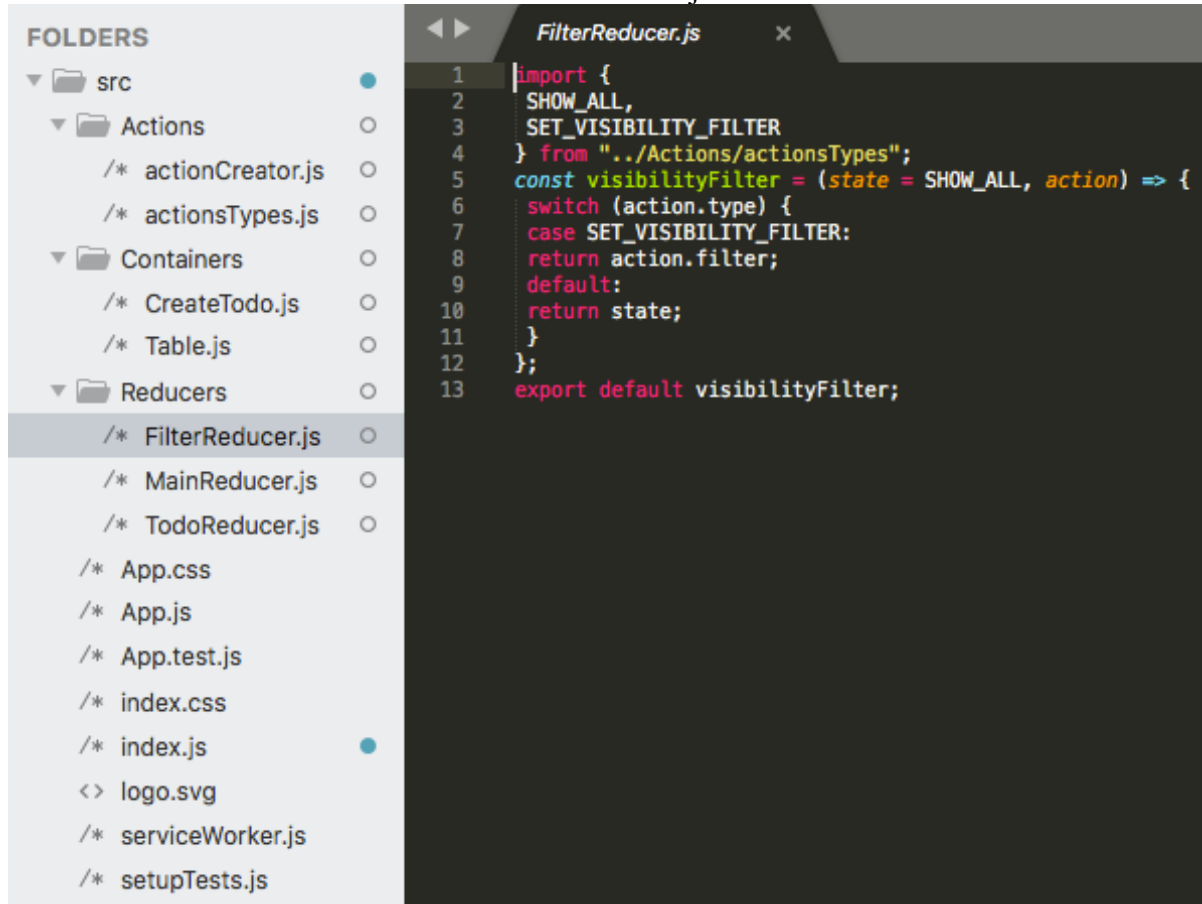
```

```

48     </tr>
49   </thead>
50   <tbody>
51     {this.props.todos.map(todo => (
52       <tr key={todo.id}>
53         <td
54           style={{
55             textDecoration: todo.completed ? "line-through" : "none"
56           }}
57         >
58           {todo.text} {todo.completed === true ? "(completed)" : ""}
59         </td>
60         <td>
61           <span
62             className="fas fa-minus-circle"
63             onClick={() => this.props.deleteTodo(todo.id)}
64             style={{
65               color: "white",
66               fontSize: "20pt",
67               marginRight: "20px"
68             }}
69           />
70           <span
71             className="fas fa-check-circle"
72             onClick={() => this.props.toggleTodo(todo.id)}
73             style={{ color: "white", fontSize: "20pt" }}
74           />
75         </td>
76       </tr>
77     )]}
78   </tbody>
79 </table>
80 ) : (
81   <div
82     style={{ marginTop: "50px" }}
83     className="col-lg-10 col-md-10 col-xs-12 col-sm-12 offset-lg-1"
84   >
85     <div className="alert alert-danger" role="alert">
86       Todo List is empty or Filter results show no results
87     </div>
88   </div>
89   ){ " "}
90 </div>
91 );
92 }
93 }
94 const getVisibleTodos = (todos, filter) => {
95   switch (filter) {
96     case SHOW_ALL:
97       return todos;
98     case SHOW_COMPLETED:
99       return todos.filter(t => t.completed);
100    case SHOW_ACTIVE:
101      return todos.filter(t => !t.completed);
102    default:
103      throw new Error("Unknown filter: " + filter);
104    }
105  };
106  const mapStateToProps = state => {
107    return { todos: getVisibleTodos(state.todos, state.visibilityFilter),
108      visibilityFilter: state.visibilityFilter
109    };
110  };
111  const mapDispatchToProps = dispatch => {
112    return bindActionCreators(
113      {
114        deleteTodo,
115        toggleTodo,
116        setVisibilityFilter
117      },
118      dispatch
119    );
120  };
121  export default connect(mapStateToProps, mapDispatchToProps)(Table);

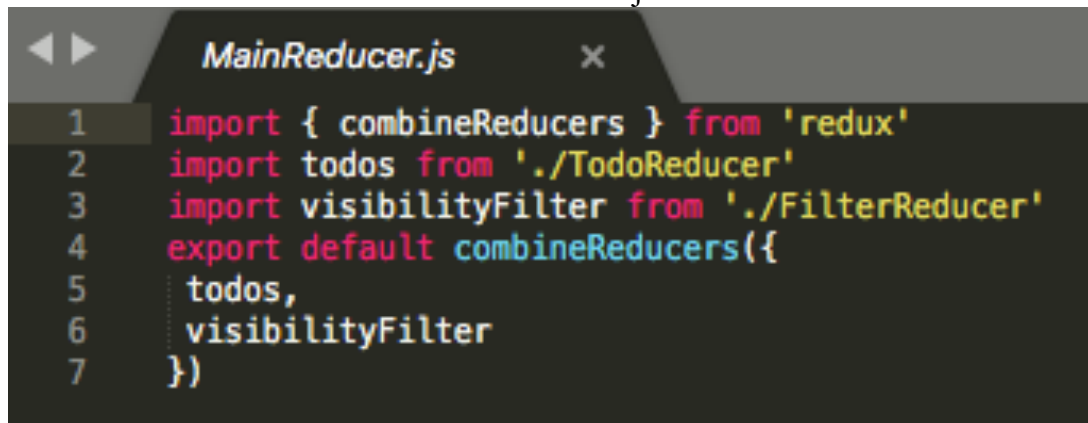
```

### FilterReducer.js



```
1 import {
2   SHOW_ALL,
3   SET_VISIBILITY_FILTER
4 } from "../Actions/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;
```

### MainReducer.js



```
1 import { combineReducers } from 'redux'
2 import todos from './TodoReducer'
3 import visibilityFilter from './FilterReducer'
4 export default combineReducers({
5   todos,
6   visibilityFilter
7 })
```

## TodoReducer.js

```
1 import { ADD_TODO, REMOVE_TODO, TOGGLE_TODO } from '../Actions/actionsTypes'
2 const INITIAL_DATA = []
3 const TodoReducer = (state=INITIAL_DATA, action) => {
4   switch (action.type){
5     case ADD_TODO:
6       return [
7         ...state,{
8           id: action.id,
9           text: action.text,
10          completed: false,
11        }
12      ]
13     case TOGGLE_TODO:
14       return state.map(todo =>
15         (todo.id === action.id)
16         ? {...todo, completed: !todo.completed}
17         : todo
18       )
19     case REMOVE_TODO:
20       const numIndex = parseInt(action.id)
21       return state.filter(todo => todo.id !== numIndex);
22     default:
23       return state
24   }
25 }
26 export default TodoReducer
```

Hasilnya :

Cancel Add Todo

All / Completed / Active

Todos	Actions	
asd	Hapus	Completed
compte (completed)	Hapus	Completed
hapus	Hapus	Completed