LAPORAN MINGGU 7

Disusun sebagai

MATA KULIAH: PEMGROGRAMAN BERBASI FRAMEWORK

Oleh:

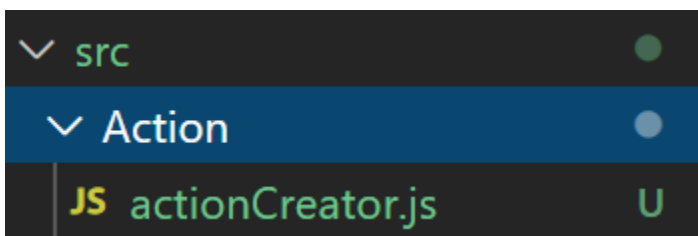Firnanda Kusuma Pratama / 1741720012
TI – 3B / 14

PROGRAM STUDI: D-IV TEKNIK INFORMATIKA
JURUSAN: TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2020

1. Install redux 'npm install react-redux' enter

```
PS D:\Tugas\Semester 6\Framework\Tugas_Firnanda_Kusuma_Pratama_1741720012_PBF\firnanda-minggu7> npm install react-redux
npm WARN eslint-config-react-app@5.2.0 requires a peer of eslint-plugin-flowtype@3.x but none is installed. You must install peer de
pendencies yourself.
npm WARN react-scripts@3.4.0 requires a peer of typescript@^3.2.1 but none is installed. You must install peer dependencies yourself
.
npm WARN sass-loader@8.0.2 requires a peer of node-sass@^4.0.0 but none is installed. You must install peer dependencies yourself.
npm WARN sass-loader@8.0.2 requires a peer of sass@^1.3.0 but none is installed. You must install peer dependencies yourself.
npm WARN sass-loader@8.0.2 requires a peer of fibers@>= 3.1.0 but none is installed. You must install peer dependencies yourself.
npm WARN tsutils@3.17.1 requires a peer of typescript@>=2.8.0 || >= 3.2.0-dev || >= 3.3.0-dev || >= 3.4.0-dev || >= 3.5.0-dev || >=
3.6.0-dev || >= 3.6.0-beta || >= 3.7.0-dev || >= 3.7.0-beta but none is installed. You must install peer dependencies yourself.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.1.2 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.1.2: wanted {"os":"darwin","arch":"any"} (current:
 {"os":"win32","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.11 (node_modules\webpack-dev-server\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.11: wanted {"os":"darwin","arch":"any"} (current
: {"os":"win32","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.11 (node_modules\watchpack\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.11: wanted {"os":"darwin","arch":"any"} (current
: {"os":"win32","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.11 (node_modules\jest-haste-map\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.11: wanted {"os":"darwin","arch":"any"} (current
: {"os":"win32","arch":"x64"})
```
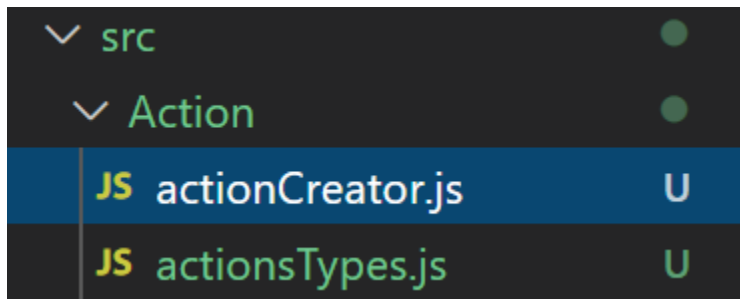
2. Buat folder bernama 'Action' di dalam folder src, kemudian di dalam folder Action, buat file bernama 'actionCreator.js'

3. Ketikan code berikut pada file actionCreator.js

```js
import {
    ADD_TODO,
    REMOVE_TODO,
    TOGGLE_TODO,
    SET_VISIBILITY_FILTER
} from "./actionsTypes";

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
});
```
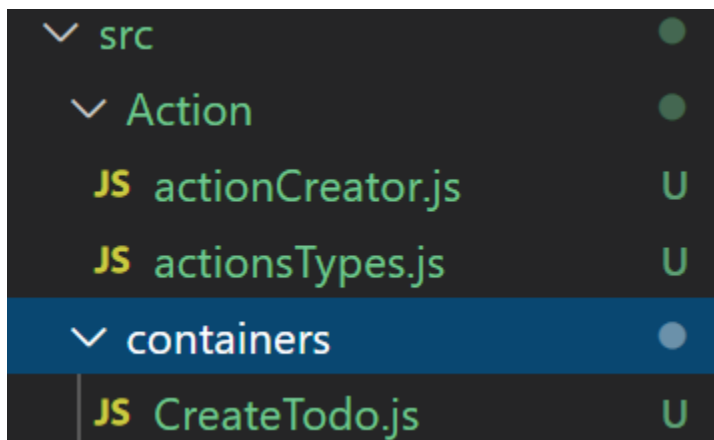
4. Kemudian membuat file 'actionsTypes.js' pada folder Action

5. Ketikkan code berikut pada file actionsTypes.js

```
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";
```

6. Membuat folder 'containers' pada di dalam folder src, kemudian buat file dengan nama 'CreateTodo.js'

7. Ketikkan code berikut pada file 'CreateTodo.js'

```
1    import React, { Component } from "react";
2    import { connect } from "react-redux";
3    import { addTodo } from "../Action/actionCreator";
4    import { bindActionCreators } from "redux";
5
6    class CreateTodo extends Component {
7      constructor(props) {
8        super(props);
9        this.state = {
10          todotext: ""
11        };
12        this.onChangeTodoText = this.onChangeTodoText.bind(this);
13      }
14
15      onChangeTodoText(e) {
16        this.setState({
17          todotext: e.target.value
18        });
19      }
```

```jsx
21    render() {
22      return (
23        <div className="form-group row">
24          <div className="col-sm-10">
25            <input
26              onChange={this.onChangeTodoText}
27              value={this.state.todotext}
28              type="text"
29              className="form-control"
30              id="inputEmail3"
31              placeholder="add todo here"
32            />
33            <button
34              type="button"
35              onClick={() => this.setState({ todotext: "" })}
36              style={{ marginTop: "25px", marginRight: "15px" }}
37              className="btn btn-danger"
38            >
39              Cancel
40            </button>
41            <button
42              type="button"
43              onClick={() => {
44                this.props.addTodo(this.state.todotext);
45                this.setState({ todotext: "" });
46              }}
47              style={{ marginTop: "25px" }}
48              className="btn btn-success"
49            >
50              Add Todo
51            </button>
52          </div>
53        </div>
54      );
55    }
56  }
```

```
58    const mapDispatchToProps = dispatch => {
59      return bindActionCreators(
60        {
61          addTodo
62        },
63        dispatch
64      );
65    };
66
67    export default connect(null, mapDispatchToProps)(CreateTodo);
```

8. Buat file 'Table.js' di dalam folder containers

```
∨ src                          ●
  ∨ Action                     ●
    JS actionCreator.js        U
    JS actionsTypes.js         U
  ∨ containers                 ●
    JS CreateTodo.js           U
    JS Table.js                U
```

9. Ketikkan code berikut pada file 'Table.js'

```jsx
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>
```

```jsx
45        </nav>
46        {this.props.todos.length !== 0 ? (
47          <table
48            style={{ marginTop: "60px" }}
49            className="table table-hover table-dark"
50          >
51            <thead>
52              <tr>
53                <th scope="col">Todos</th>
54                <th scope="col">Actions</th>
55              </tr>
56            </thead>
57            <tbody>
58              {this.props.todos.map(todo => (
59                <tr key={todo.id}>
60                  <td
61                    style={{
62                      textDecoration: todo.completed ? "line-through" : "none"
63                    }}
64                  >
65                    {todo.text} {todo.completed === true ? "(completed)" : ""}
66                  </td>
67                  <td>
68                    <span
69                      className="fas fa-minus-circle"
70                      onClick={() => this.props.deleteTodo(todo.id)}
71                      style={{
72                        color: "white",
73                        fontSize: "20pt",
74                        marginRight: "20px"
75                      }}
76                    />
77                    <span
78                      className="fas fa-check-circle"
79                      onClick={() => this.props.toggleTodo(todo.id)}
80                      style={{ color: "white", fontSize: "20pt" }}
81                    />
82                  </td>
83                </tr>
84              ))}
85            </tbody>
86          </table>
87        ) : (
88          <div
```

```
89            style={{ marginTop: "50px" }}
90            className="col-lg-10 col-md-10 col-xs-12 col-sm-12 offset-lg-1"
91          >
92            <div className="alert alert-danger" role="alert">
93              Todo List is empty or Filter results show no results
94            </div>
95          </div>
96        )}{" "}
97      </div>
98    );
99  }
100 }
101
102 const getVisibleTodos = (todos, filter) => {
103   switch (filter) {
104     case SHOW_ALL:
105       return todos;
106     case SHOW_COMPLETED:
107       return todos.filter(t => t.completed);
108     case SHOW_ACTIVE:
109       return todos.filter(t => !t.completed);
110     default:
```

```
111          throw new Error("Unknown filter: " + filter);
112      }
113    };
114
115    const mapStateToProps = state => {
116      return {
117        todos: getVisibleTodos(state.todos, state.visibilityFilter),
118        visibilityFilter: state.visibilityFilter
119      };
120    };
121
122    const mapDispatchToProps = dispatch => {
123      return bindActionCreators(
124        {
125          deleteTodo,
126          toggleTodo,
127          setVisibilityFilter
128        },
129        dispatch
130      );
131    };
132
133    export default connect(mapStateToProps, mapDispatchToProps)(Table);
```
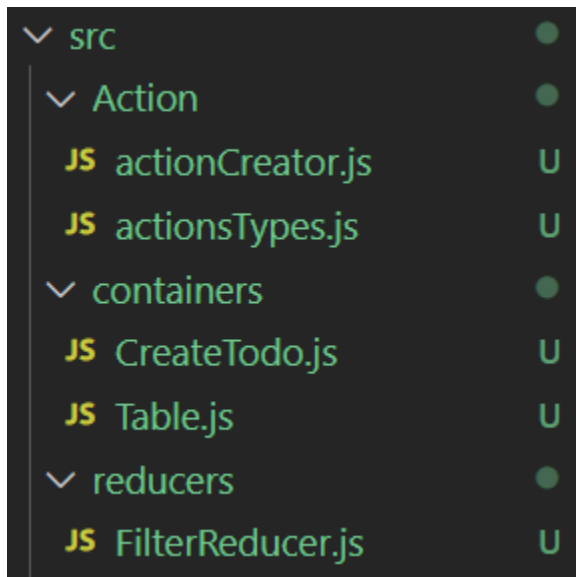
10. Edit file 'index.js' yang ada di dalam folder src seperti berikut

```
1    import React from 'react';
2    import ReactDOM from 'react-dom';
3    import { Provider } from 'react-redux'
4    import { createStore, applyMiddleware, compose } from 'redux'
5    import MainReducer from './reducers/MainReducer'
6    import './index.css';
7    import App from './App';
8    import * as serviceWorker from './serviceWorker';
9    // import registerServiceWorker from './serviceWorker';
10
11   const store = compose(window.devToolsExtension ? window.devToolsExtension() : f => f)
12   (createStore)(MainReducer)
13
14   ReactDOM.render(<Provider store={store}>
15   <App />
16   </Provider>
17   , document.getElementById('root'));
18   serviceWorker.unregister();
```
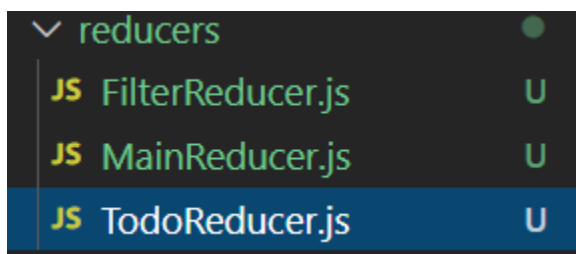
11. Buat folder 'reducers' pada didalam folder src, kemudian buat file 'FilterReducers.js'

12. Kemudian ketikkan code berikut pada file 'FilterReducers.js'

```js
import { SHOW_ALL, SET_VISIBILITY_FILTER } from "../Action/actionsTypes";

const visibilityFilter = (state = SHOW_ALL, action) => {
  switch (action.type) {
    case SET_VISIBILITY_FILTER:
      return action.filter;
    default:
      return state;
  }
};

export default visibilityFilter;
```
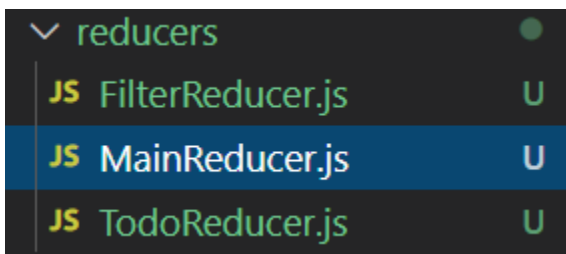
13. Buat file 'TodoReducer.js' di dalam folder reducers

14. Ketikkan code berikut pada file 'TodoReducer.js'

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

15. Buat file 'MainReducer.js' pada folder src

```
v reducers                    ●
  JS  FilterReducer.js        U
  JS  MainReducer.js          U
  JS  TodoReducer.js          U
```
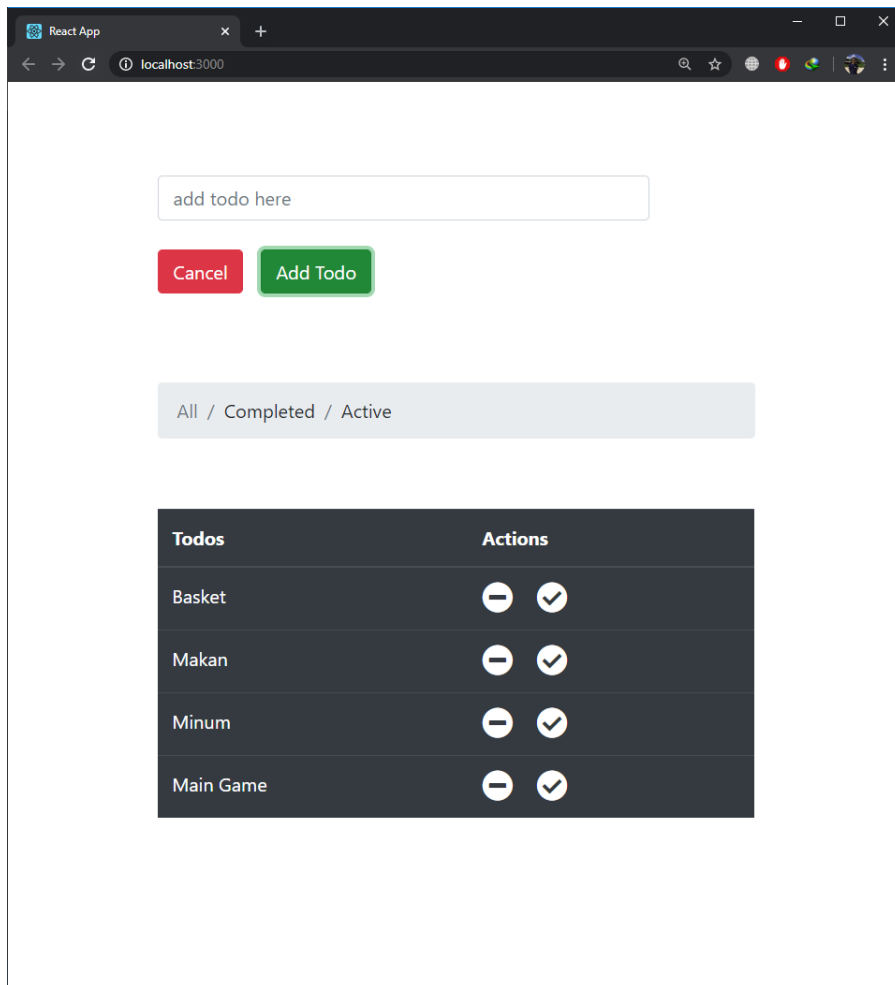
16. Ketikkan code berikut pada file 'MainReducer.js'

```
1    import { combineReducers } from "redux";
2    import todos from "./TodoReducer";
3    import visibilityFilter from "./FilterReducer";
4
5    export default combineReducers({ todos, visibilityFilter });
```
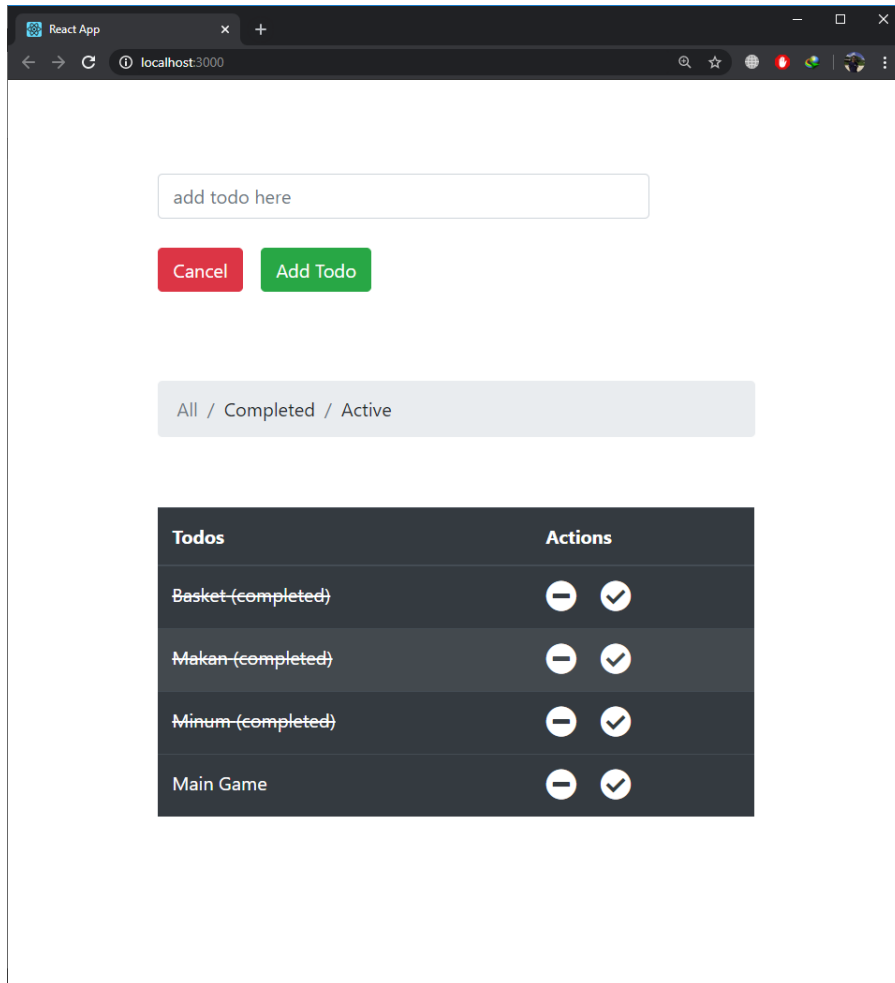
17. Edit file app.js menjadi seperti berikut

```
1    import React, { Component } from 'react'
2    import CreateTodo from './containers/CreateTodo'
3    import Table from './containers/Table'
4    import 'bootstrap/dist/css/bootstrap.css';
5    class App extends Component {
6      render() {
7        return (
8          <div className="App">
9            <div className="container" style={{ marginTop: "80px"}} >
10             <div className="row">
11               <div className="col-lg-10 offset-lg-2 col-md-10 col-sm-12 col-xs-12">
12                 <CreateTodo />
13               </div>
14               <Table />
15             </div>
16           </div>
17         </div>
18       );
19     }
20   }
21
22   export default App;
```
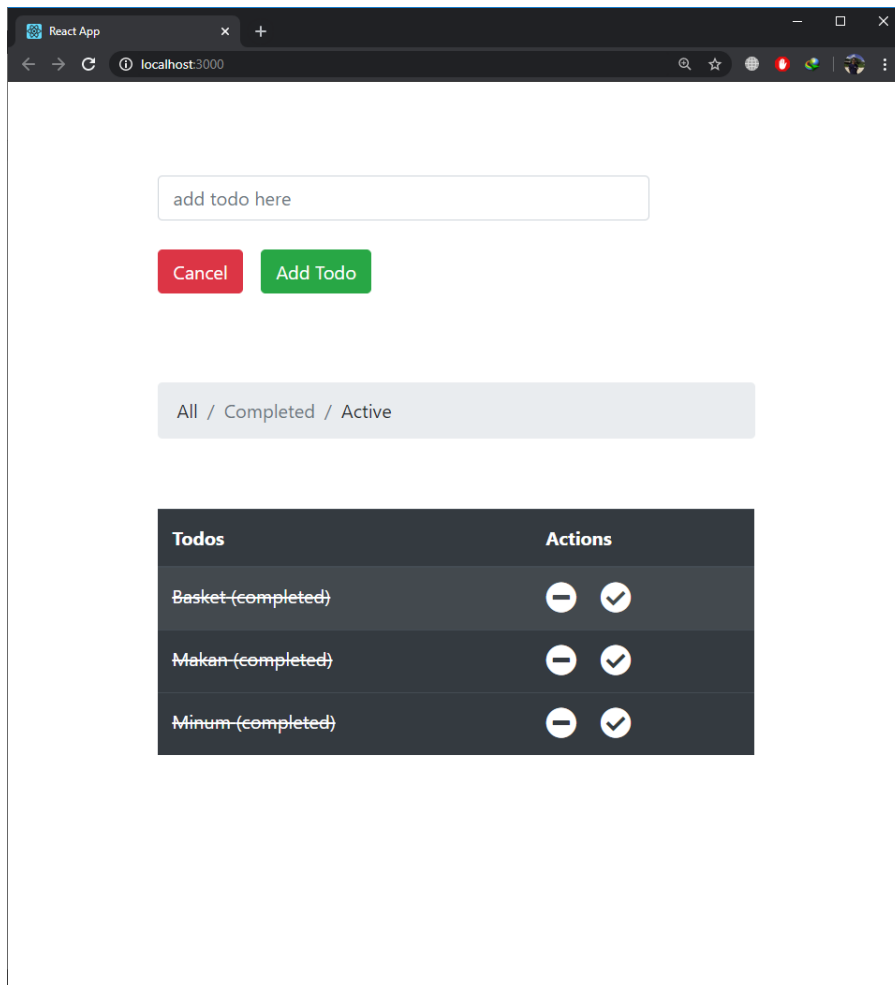
18. Hasil Akhir

Ketika kita mengklik centang pada hal yang sudah di lakukan , maka akan tercoret seperti di bawah

Pada tab complete, akan muncul yang sudah di centang

Hal yang belum di centang akan muncul di tab Active