

ReactJS Debugging

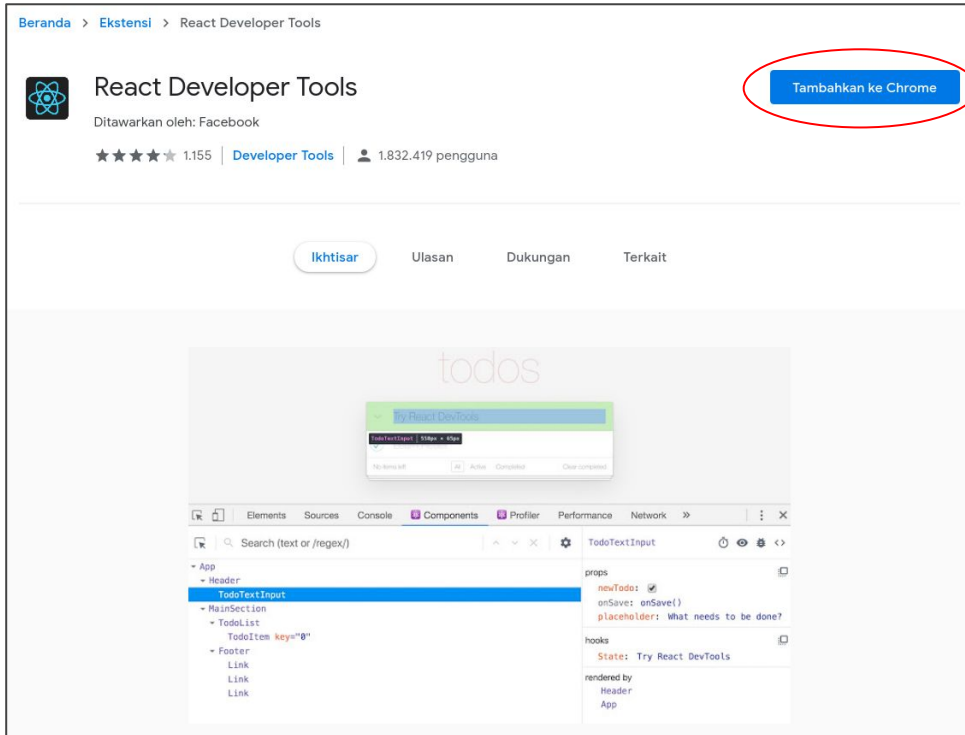
Isumi
Batam, Jan 2020

GOALS

1. React Development Tools
2. VS Code Debugger



1. React Dev Tools



A chrome plugin that provides handy ways to inspect and debug your React components.



ToDoList

Tambahkan aktivitas kamu disini, sayang... SUBMIT

<input checked="" type="checkbox"/> Hello World, Binar!	✎	🗑
<input checked="" type="checkbox"/> Induction	✎	🗑
<input checked="" type="checkbox"/> Basic class	✎	🗑
<input type="checkbox"/> Focus class	✎	🗑
<input type="checkbox"/> Party class Part 1	✎	🗑
<input type="checkbox"/> Develop class	✎	🗑
<input type="checkbox"/> Showcase	✎	🗑
<input type="checkbox"/> Graduation Batch 4 GA Batam	✎	🗑

5d626c1a21413d0017f129bd todo is complete **true** App.js:59

App.js:60

```
{isComplete: false, _id: "5d626c1a21413d0017f129bd", title: "Basic class", __v: 0}
```

5d626c1a21413d0017f129bd todo is complete **false** App.js:59

App.js:60

```
{isComplete: true, _id: "5d626c1a21413d0017f129bd", title: "Basic class", __v: 0}
```

deleting App.js:27

```
{success: true, message: "Todo Graduation Batch 4 GA Batam deleted"}
```

/* First: Don't forget to use console.log ;) */



The screenshot displays a web browser at localhost:3000 showing a 'TodoList' application. The application has a header with the title 'TodoList', an input field with the placeholder text 'Tambahkan aktivitas kamu disini, sayang...', and a 'SUBMIT' button. Below the input, there is a list of todos, each with a checkbox, the text, an edit icon, and a delete icon. The todos are: 'Hello World, Binar!', 'Induction', 'Basic class', 'Focus class', 'Party class Part 1', 'Develop class', 'Showcase', and 'Graduation Batch 4'. A status bar at the bottom left of the list shows 'Todos | 467px x 421px'.

On the right side, the React DevTools component inspector is open, showing the 'Components' tab. The component tree is expanded to 'App', which contains 'Header', 'AddTodo', and 'Todos'. The 'Todos' component is selected, showing its props and context. The props are: `delTodo: fn()`, `markComplete: fn()`, `todos: Array`, and `updateData: fn()`. The context is an empty object. The rendered component is 'App'.

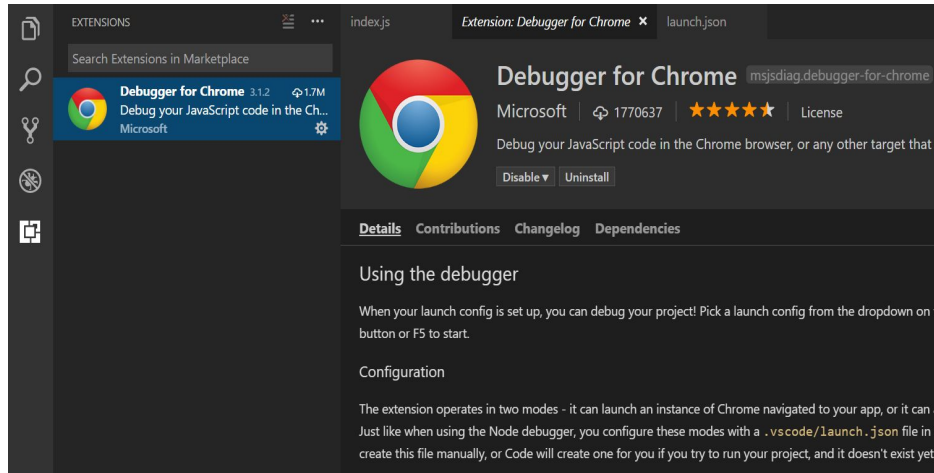


Like the regular JavaScript profiler found in your dev tools of choice, you'll have to "record" a set of interactions.

Once you've done that, you'll see a visualization of each re-render your application goes through, with accurate timing information and color-coded flame graphs.



2. VS Code Debugger



To debug the client side React code, we'll need to install the **Debugger for Chrome** extension.

Open the Extensions view (**Ctrl+Shift+X**) and type 'chrome' in the search box. You'll see several extensions which reference Chrome.

Exercise

Create Todo Web App with ReactJs and try to debugging your code. ;)

Used this endpoint:

<https://btm-rn.herokuapp.com/api/v1/todo/>