**Documentation**

By Thuy

**Technologies Used**

**Software:** Sublime 3, Apache2

**Languages:** HTML, CSS, JavaScript

**Libraries/Frameworks:** d3.js, bootstrap, jQuery

**Design**

* Have arrows on the side of the page with the screenshots displayed in the middle.
* Use bootstrap carousel for a more fluid UI.
* Color choices of a mix of grey colors. Got the color scheme from my most used software called Discord.
* Below it will be a dashboard that will dynamically appear (and adjust) to each screenshot.
  + Each performance data will only need to display the average (in text).
  + Each performance data value will have their respective test run (repeated for confidence interval) results displayed above as a bar chart.
  + Out of all the charts (bar, pie, line), I believe a bar chart best represents the data sets given as they are the easiest to understand and known to be the best for comparisons.
  + Make sure each performance data has separate colors to easily distinguish from one another.
  + Research average app page load time to color code the “Launch Time” data sets. This will help users understand how fast or slow their launch times are. This idea came from various games that display your pings.
* Using bootstrap will allow the page to be responsive across multiple devices.

**Process**

1. Examined challenge requirements
   1. Challenge: Given a json file, dynamically pull the data from the file and display it onto an HTML page. Arrange and display data that’s visually intuitive to the user.
2. Wrote up an outline before coding to avoid wasting too much time
3. Determined tools to use
   1. D3.js was the first thing to come to mind, as it is known to be a great tool to use for data visualization. I also find it extremely pretty & cool.
   2. I believe bootstrap to be a very useful tool for organization and responsiveness.
4. Setup HTML page with appropriate imports from the tools chosen for the challenge (d3.js, bootstrap, jQuery)
5. First step was to grab all the keys and values from the JSON file.
6. Next, pull specific data that should be displayed onto the page: test steps name, screenshot file name, launch times, memory, and cpu.
7. Cleaned up UI/UX.
8. Cleaned up code and organized it better.

**Final // Comments**

If you have any questions/comments, feel free to ask me as I will be happy to answer. Any tips and constructive criticism are also welcomed! Thank you 😊