Cancer Team 1

Bi-Weekly Report 5

Friday 16th December 2016

Team 35 Julien Nahum, Ben Hadfield, Sim Zi Jian

Overview

During the last two weeks, we started the experimentation and prototype stage of the experiment. We made incremental experiments, testing out React, Redux and Node.js to create test-components for our application. We also learned how to use mongoDB and created an API for with node.js and mongoDB.

We worked a lot on our project website, doing very important changes. At some point we had decided that we would use Jekyll to power our website, but realized in these two weeks that it wasn't suitable for us. Markdown is a useful and powerful feature, but it was not flexible enough for us. So we decided to move on to a basic HTML website. We started from a multi-pages template, but very quickly we thought that it would be easier to have a one page template, as we wouldn't have to change the content of navbar and footer everywhere. So we ended up creating our very own website, starting from scratch and respecting the PEACH design guidelines, which helped us to practise using them.

Alongside this we held several meetings throughout the course of the two weeks to consolidate on what we were doing and discuss what we had learnt about building the prototypes.

Meetings

Friday 9th of December: Team 35

In this meeting, we worked on the team website. This involved looking back over the course of the project, and at the preceding four bi-weekly reports, and considering our progress. From this we could complete the background and progress sections of the website. We also linked to our prototype repositories.

We also discussed our progress with our TA to get a better understanding of where we were in relation to the global aims of the module and his suggested next steps.

Saturday 10th of December: Team 35

In this meeting, we almost finished our website, and deployed an almost final version of it. We kept working our prototypes.

Friday 16th of December: Team 35

In this meeting, we discussed among ourselves the experiments and results produced from them, and how we can apply them to the construction of the components. Also, we decided to carry out more experiments to improve the current prototype that we have. Besides that, we also discussed about the content of the team video, and how we are going to present our Proof of Concept (PoC) to our client.

Task Completed

- Created prototypes for Electrode (React/Redux.js) and Node.js
- Created a new look for our team's project site
- Added more content to the team's project site

Plan

- Consolidate over the holidays
- Create a team video regarding the Proof of Concept of the component

Individual Section

Ben Hadfield

Over the past two weeks we begin working on our prototypes. I worked with Electrode.io, with the aim to create a job list. This would serve the dual purpose of getting used to the Electrode style of developing applications and building a component of the prototype.

I found working with react and redux together quite challenging, because it enforced quite a different style of programming. However, once I got used to the methodology of redux it did make sense and I could appreciate the benefits it would bring to our clients application.

Julien Nahum

For the past two weeks, we worked a lot on both our prototypes and our website. Personally, I worked a lot on the final version of the website, implementing all the website layout using bootstrap and the PEACH branding guidelines.

Moreover I also worked a lot on node.js and databases. I made some very simple prototypes using the express framework that allowed me to get more familiar with it. I also created the database restful API using both node.js and mongoDB. I used postman to test the API while building it.
Over the winter break, I plan to learn more about react and electrode, which are the project's technologies that I am the less familiar with.

Sim Zi Jian

For the past two weeks, I researched more about the various libraries in Electrode. I also tried out some small experiments with the React draggable elements to see if it is viable for this project. At first, I attempted moving an entry on a test scheduler by clicking on the boxes of the scheduler. After successfully doing that, I tried making the element draggable, and creating a drag-and-drop interface for that test scheduler. These experiments are useful for the drag-and-drop interface that is required for the scheduler in the web application. For the next few weeks, I plan to do more experiments about Electrode, while learning how to implement them into our application.