

# **Sprint 2 Report**

## **OnCareer**

### **Team OnCareer**

2/21/2018

- **Actions to stop doing:**

- The team should stop being late to meetings or communicate if they are unable to make it. This is important as being late or not letting others know they are unable to come wastes time as others are expecting everyone to attend.

- **Actions to start doing:**

- The team should schedule more group meetings, so we can keep each other updated and get more work done. This can be done by having occasional Google Hangouts.
- The team should try and be more accurate at estimating work tasks. Some tasks ideal and actual hours were a little off. This will help us better estimate our workload in the next sprints.
- The team should try to document their code more. The team tries to make the code easy and readable, but there can definitely be more documentation. This will help everyone understand the code and how to use it.
- The team should communicate even more in Slack. The team does communicate well, but we could always communicate more. If we can communicate more, allocating tasks and updating work is much easier.
- The team should schedule code days. The team has done a lot of coding without group code days and just allocating tasks to be done individually by tasks, however some people have trouble coding and are still learning. Code days could help other people understand the technologies more and get work done at the same time.
- The team should start asking more questions. There is a lot of code and a lot of code that people might not be fully clear on how it works exactly. Asking more questions could help everyone learn about the code, product, and technology.

- **Actions to keep doing:**

- Communication via messaging is working well, so the team should keep communicating via Slack and other messaging tools. This is essential to keeping everyone updated in a group-wide forum.

- Code review and communication is working well, so the team should continue using GitHub branches before merging branch with master. This helps the entire team review code for correctness before pushing it to the main branch.
- Learning about and teaching others about technologies are working well, so the team members should continue to help each other to catch up with technologies. Not everyone is caught up with each other in terms of experience with the technologies and tools, but all the members help each other which allows more growth for the team so everyone can learn the technologies.
- **Work completed/not completed:**

#### **Completed:**

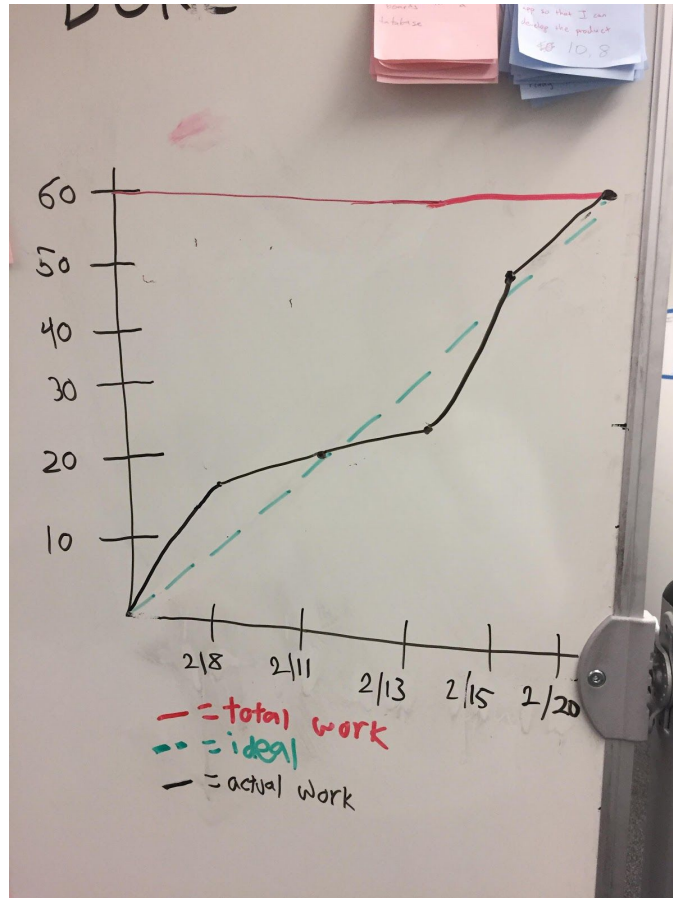
- User Story 1 - As a user, I need a dashboard so that I can see all of my application categories.
  - Build the front-end of dashboard using React
  - Build the back-end and database for boards
- User Story 2 - As a user I want to be able to navigate between pages for simplicity
  - Work with React routing
- User Story 3 - As a user I want to be able to represent my applications on a calendar timeline.
  - Build the front-end of Calendar
  - Connect Google Calendar API
  - Write functions with events using Google Calendar API

#### **Not Completed:**

All planned work completed during sprint.

#### ● **Work completion rate:**

- Number of User Stories in sprint: 3
- Total number of estimated ideal work hours: 20 hours per member, 60 hours total
- Total number of Days in sprint: 20
- User Stories per day: 0.15 Stories per day
- Ideal work hours per day: 3 hours per day among all members total, 0.5 hours per day per member
- Number of User Stories across all sprints: 6
- Total number of estimated ideal work hours across all sprints: 40 hours per member, 120 hours total
- Total number of days across all sprints: 32
- User Stories per day across all sprints: 0.1875 Stories per day
- Ideal Work Hours per day across all sprints: 3.75 hours per day among all members, 0.625 hours per day per member



Sprint 2 Burnup Chart