Codeversation:

Sprint 1 Retrospective

CS 307 - Software Engineering - Team 23

Alex Aralis, Patricia Chun, Jeremy Lehman, Zach Perry

What went well during this sprint:

1. Server Setup

The isomorphic server is running, and can serve webpages.

2. Setup remote database.

A mongodb has been created and is being hosted by mLab. This remote database is used by the isomorphic server to store user information.

3. Created compiler micro-service.

An extensible compilation micro-service has been set up. The service can be interfaced with via a REST API and can currently compile ruby to javascript. More languages are planned for the future.

4. User Authentication and Session creation.

User accounts can be saved to mongodb via a REST API, and JSON web tokens can be requested via the sessions route. A middleware will verify a session token provided to the server via the Authentication header, and provide a user object to later server services.

What went poorly this sprint:

1. Communication

We did not do a very good job of letting each other know if we were struggling with certain components of our project, or if we were waiting on someone to finish a task, so they could finish implementing their own. We also didn't truly realize how behind we were until we reached the final week of the sprint, since we were not communicating very well where we were during the spring.

2. Time Management

During this sprint, we waited until the last week to do a majority of our work, which lead to us not finishing a lot of our stories since we ran into blocking issues. Because of

these issues, it made it difficult to finish some of the stories that were reliant on other stories.

3. Learning Curve

Some of the technologies we are using for our project are unfamiliar to members of the team. Due to the learning curve, some stories took longer to implement than originally planned.

4. Blocking Issues

Also during his sprint, we allowed blocking issues to stop us from working on other components. Often times, we would focus on one story, and if we couldn't keep working on it, we would stop, instead of switching to a new story, which resulted in stories we never even got to.

How can the team improve for the next sprint:

1. Meet more frequently during the sprint.

While team members met several times for regular meetings during this sprint, we were not communicating enough and not very productive. To fix this problem, the team will be meeting more frequently and check what has been done and what will be done, so that if there is an impediment, it should be immediately fixed and solved with the help from other team members.

2. Work together.

Finishing this sprint, the team found that it would be better to work together instead of working individually. This will make sure that team members are committing regularly and spending as much time as desired on the sprint's development.

3. More effective time management

This sprint we will implement the github kanban board more effectively, so that we can make sure everyone knows where we are on tasks.

4. Daily standup via Slack

We will have a daily standup where people will let everyone know where they are at currently on their tasks so that we will keep updated on where everyone is at.

5. Have a weekend meeting for working collaboratively.

We will meet collectively at least twice during this sprint to spend time working on places where the current sprint tasks interact with each other, so that we can collaborate more effectively.