

CS 407

Spring 2018

Sprint 3 Retrospective

April 30, 2018

MakeItHire

Team 23

Avi Rakesh
Ben Weis
Nik Suprunov
Rajat Srivastava
Zack Fernandez

1 What went well?

With Sprint 1 and Sprint 2, we focused on we allowing different types of users to sign-up for our application, make accounts, add their information, and securely login and log-out of the application, along with giving special privileges to the admin. Sprint 3 was designed to implement the core purpose of our application which was to let students easily apply for jobs, let recruiters view their applications, and chat with the students who have applied. With all the user stories completed, we were able to develop a full working model of the application, with all the features implemented as promised. The major accomplishments for this sprint were the completed of the Application system, the Chat system, and a supplementary iOS Application for students which could be used to view applications and chat with the recruiters.

The following is what went well in terms of features and user stories:

- Students were able to look at the jobs which had required supplementary questions, and quickly answer them to apply for that particular job. Recruiters were allowed to add supplementary questions to any job they posted.
- Students were able to look at the job listing page, and were able to apply for the job at the click of a button.
- Recruiters were able to easily message interesting students with a click of a button, and then both students and recruiters could chat with each other on the chat platform.
- Admins were given the privilege to modify and delete company profiles, which would in turn trigger the presence of the company in the rest of the database, and delete any information regarding that company.
- Recruiters were able to add tags on company profiles and job postings.
- Students were able to search and filter the job applications based on company tags, company names, job name, job tags, etc.
- Students were also able to use the supplementary iOS view and apply for the job postings, and chat with the recruiters.

2 What did not go well?

With any software project on a deadline, there will always be a few tasks whose specifications need to be compromised or that need to be delayed to the benefit of the entire software as a whole. Our User Story 6 stated: 'As a user of the application, I should only be authenticated to my relevant routes, and should not be able to use routes that were not designed for me.' This was fully implemented alongside a new database schema and better authentication tooling. However, we severely underestimated the hours necessary to complete what was essentially an entire rewrite of MakeItHire's back-end.

We did not account for the amount of technical debt that would be accrued in attempting to integrate the existing front-end's API calls to the new back-end. To remedy this, we decided to demonstrate the improved back-end in our sprint demo via postman, rather than implemented into our product which was still operating on the old implementation. This underestimation of hours,

was by far the biggest obstacle toward completing sprint 3 fully. However, the back-end was still operational and contained the following features:

- Simplified language and use of industry standard CRUD operations.
- Properly authenticated users based on role, rather than on the obfuscation of a users type in their JWT Authentication token.
- Was implemented using modern ES6 practices allowing for more succinct, concise, and readable code.
- Allowed for minimization of backend database queries and a better schema. If fully implemented into our product it would reduce total technical debt in the long run.

3 How should you improve?

Though CS 407 has taught us a lot about how a real world scrum team functions, there are some places we see potential for self-improvement:

- Moving forward, we feel that we could better implement source control than how we did during the course of this class including stricter enforcement of PR styling and testing practices.
- Focus on transforming into more well rounded full-stack developers, rather than only specializing in either front-end or back-end development.
- While we had a very good grasp at estimating how much work was achievable in a given time-frame, we needed to have a better understanding of how much work needs to be done to fully complete a user story. Rather than just the focus of the story, estimate time for integration, testing, and bug fixes.
- Don't accidentally checkout untracked files from source control without committing and losing over 5 hours of work that then needed to be rewritten. :(