Individual Report: Rate My Teammate (ver.CWRU)

Nora Tang

November 2021

1   Project Responsibilities

As a whole team, we together completed the database design, the prototypes of the front-end. As a front-end subteam, we together discussed and broke down the components, and helped each other out whenever there are issues with rendering, or connection.

Individually, I am mainly responsible for the forms and student page UI design of the front-end, marrying the form connection between the front-end and backend, and trigger functions in the database design.

Specifically,

- Created, configured, and wrote instructions of the initial React project.

- Wrote the React.js code in the **/FormElement**, **/RatingElement**, and **/StudentElement** (UI) folders, **AuthForm.js**, **AddNewStudent.js**, **ReviewForm.js**, **StudentPage.js** (UI), **SubmitCorrection.js**, and configured the initial routing for each page.

- In controllers, I wrote all the java code in **ReviewNegativeTagsController.java, ReviewPositiveTagsController.java, NegativeTagController.java, and PositiveTagController.java.**

- In **ReviewController.java**, I implemented the following APIs: **Review/add**, **Review/update**.

- I also debugged for **AccountController.java**, and **ProfileController.java**.

- Other than all the above, I also fixed some minor bugs in the backend.

In addition, thanks to Liyuan Huang's and Haihan Jiang's hard work, they have helped me a lot with front-end design and connection issues.

2   Achieving Objectives

The workload and design of the front-end is far more complicated than my previous projects. During my previous project, I was only required to finish the basic form design and some search result rendering with React.js, Node.js and Express.js, and was mainly responsible for databse design with PostgreSQL. I didn't have enough experience with configuring React with Spring Boot and MySQL. When I realized that my knowledge for the stack development for this project is limited, I immediately set off learning React with Spring Boot as backend from the beginning in order to deliver a front-end above the standards. By going through each section in online courses, I began to be famliar with Javascript, CSS, and Restful APIs with Spring Boot. At the first stage, I followed the instructions of the online course to complete the same application as they did to gain basic knowledge of React. Afterwards, I started our project from scratch, transferring what I learned into my own understanding and writing. Step by step, I was able to deliver the required designs.

   Later, when encountering with the Spring Boot connection, I also followed the same procedure as above. In addition, thanks to Haihan's work, I was able to follow his style and wrote the restful APIs for reviews and tags.

3   Issues

During the implementation, I encountered several issues:

- The position of React components are sometimes hard to adjust. Instead of using *pt* to set the distances, *rem* or percentage are better choices.

- Authorization status of the users is necessary which I didn't implement at the first place. Later on, I set the expiration time of authorization token to be 1 hour and stored the authorization token in local since

some pages require users' login status, otherwise, users should not be allowed to access these pages.

- During the implementation, I realized some fields of tables in the database should be revised to be more logical, such as the name should be separated into firstname and lastname. These issues have been fixed afterwards.

- Establishing connection was time-consuming. Because the backend team and front-end team were working separately in the first half stage, we did not realize that some fields in backend and frontend were not consistent. Such inconsistency took me a lot of time to re-configure the input and output data.

4 Lessons Learned

- **Communication matters.** The most essential one is that font-end and backend should work coherently from the beginning. And we should first specify our requirements of restful APIs, and set up rules for field names, which could save us a lot of time during integration. The efficient communication in our second half stage has speeded up our progress as well.

- **Familiarity of full stack development.** Creating a website with appealing look and efficient backend is a valued experience for me to become familiar with full stack development and sparkled my interest in further pursing software engineering.

- **Boosting both soft skills and technical skills.** From this project, I learned a project cannot be developed without neither cooperation nor technical skills. A great project needs both excellent skills and efficient communication.

5 Insights

Overall, it is a pleasure to work with my teammates to deliver a website within our expectations. Their excellence cannot be overstated. Although we never worked as a team before, and we didn't know each other's work

styles before this project, it is such a surprise that we could quickly be on the same line. With various backgrounds of our team, some having little experience with the front-end, some having little experience with the back-end, we are able to efficiently allocate our workload to accomodate everyone's needs. In addition, developing a simplest forum-like website from scratch requires more non-trivial and essential designs. It is not as simple as we used to assume. Passing data from parents to children components, handling input data type from form component through connection, setting up routing with considerations of users' security, setting up authorisation and so on are all essential parts to ensure users' experience. However, there's still available imporvement for our data visualisation and UI design. In the course of time, I will learn more tools to design a more user-friendly front-end.

I am more than grateful to be in such an outstanding team. It's the effort and dedication we made together that leads to our achievement.