SW Engineering CSC 648/848-04

Team 04

Leiyi Gao: Team Leader
Justin Mao: Backend Leader
Yinyin Wu: Frontend Leader
Michael Han: Git Master
Nicholas Hamada: Scrum Master

Milestone 3

11/08/2022

Project Name: RateMyResume

CSC 648/848 SFSU 2022 Milestone 3

Review of functionality, UI, SW and planning for final product delivery

Includes Checklist for instructors and teams (Appendix I and II)

Objective and Overview

The objectives of Milestone 3 are to:

- **Define exactly what product you are delivering.** We will come to agreement on what the final application is going to look like in terms of functionality, especially which functional items are priority 1 (P1). This will be your commitment to the instructor/client.
- **Ensure software development is on track.** We will verify that the all the SW components are installed and integrated and that most major functions work.
- **Provide feedback on all major UI screens and functionality**. This will be done through a review of the so-called horizontal or UI prototype.
- **Check software architecture** by reviewing your code, the database and overall design at a high level
- **Check all algorithms** like search or machine learning component, whatever major algorithm in your application will be checked.
- **Identify and address all technical risks**
- **Ensure effective teamwork** by verifying that all team members have started implementation of their portion of the project
- **Ensure software development is effective** by verifying your team's collaboration practices based on github branch policy and github review policy.

Milestone 3 will be in the form of two-part review:

1. **Part 1 – each team presents to Prof. Song: Review of functionality, UI and general project status will be done during the meeting of ~15 minutes. – on deadline during the class. In order to be efficient, teams must observe strict schedule and come fully prepared. For this, please prepare Appendix I before the class hour of deadline. And please update Appendix I after the meeting. The Appendix 1 should be submitted to your M3 folder for review (Due : on the next day of class).**
2. **Part 2: SW review (in-emails): Architecture, Code, github usage, database and general SW (including code review) will be done by TA after Part 1 review. For this, please share your web application URL with TA and the instructor.**

M3 will be graded, and the feedback will be given if any. The teams will meet to analyze feedback and revise M3, design and implementation accordingly, as well as to fix on P1 set of features. The revised M3 doc will be resubmitted at M5 and it will

be regraded in the part of M5. After M3, the teams will have "feature freeze" e.g. the teams must focus on intense implementation of only P1 features.

Part 1 Review – Functionality and UI feedback and general project status

IMPORTANT!

Appoint a "scribe" e.g. person to collect meeting feedback and main points and action items (use Appendix I as a template)

What to bring to the meeting

Each team **must** prepare your ***Product prototype*** to the Milestone 3 Part 1 meeting:

- ***Product prototype*** has limited functionality
- You have to demonstrate the 5~6 key P1 functionalities for your product prototype for the meeting. For the key functionalities, you should connect back-end and front-end.
- The product prototype should provide UI implementation of 5~6 key P1 functionality.
    - The UI implementation should follow UX story board in M2.
- The current version of your SW should run on deployment server .

On the part1 meeting, the instructor will let each team to demonstrate major functionalities on real-time using your SW and will give you feedback. **You are requested to appoint a scribe who collects the feedback. Use Appendix I as a template to record feedback.**

After the M3 Part 1 meeting (recommended to do it immediately after the meeting): Team has to meet, analyze meeting feedback and revise M3 doc (Appendix 1), design and implementation as necessary. Team also must finalize P1 set of features. This feedback as well as finalized P1 list MUST be written down using template as in Appendix I. You will submit it with M3 folder.

Functionality:

- **Retrieve all the posts**
- **Search post**
- **Comment on a post**
- **View posts**
- **Like posts**

Part 2 review: SW review – to be done by TA after Part I review, by accessing your github repo. See Appendix II for details

Appendix I – Rubrics and checklist for Part 1 Milestone 3 review: Project Status and UI Review. Use modification of this for M3 summary

Section: 4 Team: 4 Date: 11/7/22
Number of students present: 5

1. *UI and functionality feedback (P1 functions only)*

- **Instructor's comments on functionality for your demo (should be filled after your demo on M3)**
  P1 features are not fully implemented: login, resume upload liking and commenting posts, searching.

- **Instructor's comments on UI (should be filled after your demo on M3)**
  The instructor suggested a lot more work is needed on the frontend; a concrete plan to implement the frontend UI based on M2 mockup is needed; styles across pages needs to look consistent (currently they look incoherent).

2. *List of P1 features committed for delivery– write down the items before the demo and verbally explain it during the meeting if time is allowed*

- User login
- User registration
- Post creation
- Post viewing
- Liking/commenting a post

3. *Project status – write down the items before the demo and verbally explain it during the meeting if time is allowed*

a) *Teamwork*

Work hasn't been evenly divided in the backend due to no fault of any single member. One team member did the bulk of the work enthusiastically without delegating some of the tasks to another member. Another team member is eager to contribute but has been left out of the codebase. Since they are less experienced with the web framework and the database of choice, catching up with the codebase has been difficult.

In the frontend, one team member was busy with assignments and projects from other classes, so we had less time to dedicate to working on the frontend. Also, at the beginning, there was miscommunication that resulted in both frontend members working on the same component and not realizing this until the 2nd week of milestone3.

b) *Risks*

Scheduling & technical risk - frontend team does not know how to read the backend code or how to run the backend server while our backend team does not know how to read the frontend code or how to run the frontend client. The work being done so far has been extremely compartmentalized, resulting in the majority of bugs created during development not getting detected until integration testing.

- **Plan:** better documentation on how to run our application locally; paired-programming sessions where a member from the frontend team will be paired with a member from the backend team to push out features or fix bugs.

Technical risk - The frontend team is fairly new to React and had never developed a functional React app before. The frontend team has been learning React framework almost from the ground up while building the application. This makes getting the intended functionalities for our app and polishing its UI a challenge.

- **Plan:** continue to learn the necessary concepts around React framework to get our application functional and running; frontend team will reach out to some SFSU friends and alumni for some feedback and pointers for the project.

Appendix II– Rubrics and checklist for Part 2 Milestone 3 review: SW Review (to be done off-line by TA after Part 1 review)

Section: **Team:** **Date:**

Instructor/TA to Check and comment below:

- Git/Github organization (e.g. organization of branches)
  - To setup Dev branch and Feature branches are strongly recommended.
  - Grading check points : how many branches are setup and how they are used for.

- Git/Gith, git hub usage: code review practices (to see if the review comments are proper and enough)
  - Grading check points :
    - how many code reviews are being done for the dev branch (or any integration branch)
    - what are check items to review codes

- Frameworks (back end front end) deployed correctly

- Database organization (tables, naming…)

- Efficiency (proper use of image thumbnails, efficient search etc.)

- Other