CSC207H5 Project: Phase 2

Phase 2 Due: December 7, 2022 at 8PM Eastern

1 Introduction

Congratulations, you are now ready to make your project a reality with your team. The instructional team is here to help you achieve this goal. In class we talked about how an Agile team works on a software development project: the process, the roles, etc. This is where you will put all this knowledge in practice.

This document outlines how your work will be assessed and what is expected during Phase 2 of your project.

2 The Process Ahead

The first thing you will want to do is to consider any necessary changes to your design document as a team. You can revise your Project Identification, your User Stories, your Design Patterns and, perhaps, your Timeline as you find this to be necessary. With an agreed upon design, you can organize yourself for your first "Sprint". At each Sprint, you will assign user stories to each team member (or team members will continue their work on an existing story that they own). Each team member will be required to implement whatever feature(s) they are assigned. You will not be required to complete all your user stories but we will want you to complete all that are marked as high priority and we will want to see you realize a functional piece of software. It can be an incomplete piece of software; this is ok! But ideally we would like to see you build a strong foundation for future work.

From this point to the completion of the project, there should be time to conduct 3 Sprints at approximately weekly intervals. Our suggestion is that you time these as follows:

1. You can **prepare** for your first Sprint between now and November 18.
2. **Sprint 1** can begin on Nov 18 and end on Nov 25.
3. **Sprint 2** can begin on Nov 25 and end on December 2.
4. **Sprint 3** can begin on December 2 and end on December 7. This spring will likely focus on assembling the work you've managed to complete, and preparing for your presentation and final submission.

At the outset of each sprint you will review and prioritize your user stories with your team; you will then work toward the completion of those stories that you identify as highest priority. At the end of each sprint, you will identify completed user stories and update any incomplete user stories or tasks. The user stories that are targets of your next sprint will form the basis of "product backlog", which your can maintain as a spreadsheet if you like. These product backlogs will guide you and your team through the iterations that follow. At the end of each sprint you will also take a moment to reflect on the sprint, and to identify any development processes that worked well or that are in need of improvement.

We want you to follow Git Flow when you implement your features. Git Flow will require you team to:

1. **Have a 'develop' branch** on your repository;
2. **Create a branch** off of develop for each feature;
3. **Create pull requests** to merge feature branches into develop so that they can be reviewed and approved by group members;
4. Once all the features in a sprint are completed and located in the develop branch (i.e. at the end of Sprint 3), create a pull request from develop to the master branch (i.e. to create a final product or a "release").

Each branch should be named after your user stories using the following convention: "feature/ID-short-feature-description", So, if User Story 1.2 involves updating a radio button, you might name the branch "feature/1.2-update-radio-button". All commits from the team should be pre-pended with a user story number, too. So, for User Story 1.2, each commit should be pre-pended with the words "[DEV-1.2]".

As a part of the pull request process, you will have an oppportunity to do code reviews. We will ask that each team member provide a code review(s) for at least 1 user story owned by one of their peers. Make sure every user story is sufficiently reviewed before you commit anything.

Note that we'll be looking to see that each team member contributes a reasonable number of lines of code to the final product and that some contributions, as well as code reviews, take place during each iteration.

3 Project Demos

At the close of the project, you and your team will be asked to demonstrate your work in front of a small number of CSC207 instructors. You will be marked based on this presentation, your code, on your final report and your tweaks to your design document. During your presentation we will ask you to do the following:

1. Present the running program and show that everything is working as expected.
2. Present your UML diagrams to show your design patterns, talk about how you implemented them and why.
3. Discuss the challenges you faced as a team, how you overcame them, and what you learned from the project as a whole.

Your presentation cannot be more than 10 minutes in total; within this time frame, each team member must detail their own contributions to the final work. Your presentation must be a minimum of 3 minutes long. Make sure to adjust the content of your presentation to match the given time limit.

4 Marking Criteria

The marking criteria for this portion of your work will be as follows:

Project Demo (worth 30% of the mark on this assignment)

- Features must be clearly presented;
- Design Decisions are clearly explained;
- Each teammate must clarify their contributions;
- Each teammate must be present at the presentation to explain what they have done.

Process Artifacts (worth 40% of the mark on this assignment)

- User stories must be identified at each sprint and initiated;
- User stories must match product backlog documentation;
- Pull requests must emanate from owners of user stories;
- Repository branches must be clearly named and related to user stories;
- Commit messages must be clear;
- Commits must be distributed equitably;

- Code must be clearly documented using JavaDoc;
- Code must include basic sanity tests of functionality;
- There must be evidence of incremental work and some consideration of milestones on your timeline.

Phase 1 Revisions and Final Report (worth 20% of the mark on this assignment)

- Process must be clearly documented according to the final project report template.
- Any revisions to UML in the design document should be complete and well related to code.
- Any revisions to Design Patterns complete and related to code.
- Any Revisions to User Stories complete and related to code.

Writing Quality (worth 10% of the mark on this assignment)

- Structure. The organization of your document, and its contents, must be logical and clear.
- Mechanics. Sentences must be complete, grammatically correct, and well written.
- Appropriateness for the Audience.Your writing must be appropriate for consumption by a technical audience of your peers.

What you will turn in

- finalReport.pdf (your final process report)
- designDocument.pdf (your revised design document)
- Sdheduling of the project demos to be announced ASAP.

Have fun and GOOD LUCK!!