[CS304] Team Project - Milestone 2 (15 points)

> Presentation: week 9 lab session (no rearrangement)
>
> Report submission: 23:59pm, April 14, 2023. 40% penalty for late submission.

Team Report (12 points)

At this point, your team should already be clear about the requirements and designs of the system. Your team should also have executed certain software processes for a while using corresponding tools for project management. Finally, your team should have certain deliverables ready by now. In this team report, you'll be reporting all of the above information, as detailed below.

Requirements Modeling & Design

1. Use Case Diagram (2 points)

Use case diagrams can be used to provide the *scenario view* (from the 4+1 view model) of software systems.

Your team should provide **one or more** use case diagrams to describe the functionalities that system provide to end-users. If your system has complex subsystems, you might want to provide multiple use case diagrams.

2. Swimlane Diagram (for one of the use cases) (1 point)

From the use case diagram above, **pick one of the use case** that lacks details and could benefit from a swimlane diagram. Provide a swimlane diagram for this particular use case.

3. Natural Language Description (for one of the use cases) (1 point)

From the use case diagram above, **pick one of the use case** that lacks details and could benefit from a natural language description. Provide a natural language description for this particular use case.

Note that the description should follow the template we introduced in lectures and labs; it should include detailed description on meta data (e.g., trigger, entry conditions etc.) and flow of events (e.g., basic/alternate/exception flow).

4. Diagrams for *Logical view* (2 points)

Class diagrams or *component diagrams* can be used to provide the *logical view* (from the 4+1 view model) of software systems.

Your team should provide **one or more** class diagrams and/or component diagrams to describe the components, relations, and structure of the system, to provide a logical view of the system. If your system has complex subsystems, you might want to provide multiple such diagrams.

5. Data Design (2 points)

Describe the type of data to be handled by your system (e.g., user information, domain data like book information for a library management system)

Please also explain the data design in detail. For example, if you plan to use a database, you could provide a detailed *database schema* to describe the tables (e.g., table names, fields, data types) and relations between tables.

6. UI Design (2 points)

Before you start implementing UI, it's better to have a concrete UI design to guide implementation and facilitate communication.

Please provide the UI design for **primary user interfaces** for your system, e.g., the main/welcome page for your website, the homework submission UI for a OJ system, the book metadata page for a library system, etc.

You DON'T have to provide UI design for common interfaces like user login page. You should only provide UI design for the **primary user interfaces** that are specific to your own team project.

You should provide images as the UI design output. You could use any tools for this purpose (you could even draw sketches by hand). Yet, we recommend you to explore **wireframe tools** that are widely used by professional UI/UX designers (see page 46 of lecture note 5 for an example of wireframes).

Collaborations (1 point)

Please demonstrate that your team has used git for collaboration.

First, provide a snapshot of the commit distributions from all team members. On your GitHub team repo page, click `Insights`, then select `Contributors`, and you could see the commit stats for all team members.

Second, provide a snapshot of the commit history graph (e.g., rendered by `gitk` as we introduced in lab 3), which possibly contains multiple branches and merges.

Deliverables (1 point)

Please briefly describe the functionalities that you've implemented so far, as well as the corresponding deliverables (e.g., code, documentation, etc.). Provide a snapshot of your repo's directory structure.

Individual Progress Report (1 point)

Each team member should write and submit a progress report **individually**. In this progress report, you should briefly describe:

- How is the team collaboration so far?
- What is your responsibility in the team project for this milestone?
- What have you contributed to the project?
- What difficulties have you encountered so far?

Team Presentation & Demo (2 points)

- Each team will give a 10-minute presentation **in the lab session of week 9**. This time, your team should give a demo of the running system.
- Every team member needs to show up during the presentation.
- Every team member needs to be the presenter at least once throughout the semester.

Submissions

For this milestone, you should have made some coding progress, and **we'll check your team repo**.

You should also submit a team report and an individual progress report for this milestone. Both submissions should be made in GitHub Classroom Assignment.

Team report

- Submitted by each team to `GitHub Classroom` -> `Assignments` -> `Team project`.
- Accept this group assignment from this invitation URL
- Join groups (see lab 1 notes).
- Upload the report as `milestone2-teamID.md` to the assignment repo. You can find your teamID here.

Individual progress report

- Submitted by everyone to `GitHub Classroom` -> `Assignments` -> `Milestone2-Individual-Report`
- Accept this assignment from this invitation URL.
- Upload the report as `milestone2-studentID.md` to the assignment repo.

The file format of report should be `.md` markdown files. Other file format such as `.pdf`, `.docx`, `.txt` will NOT be accepted.