CS4471B: Software Design and Architecture

Course Project Handout – Fall 2022

Due Nov 30 @ 5:00pm

Introduction

A significant component of this course is a software development project. The aim here is to create a software product for a client who will use it daily.

Thus, you will work together through the complete development cycle, from understanding the requirements to delivering a working product. Your customer can be an imaginary person or organization other than yourself.

Project Goal

The project will provide you with an understanding of the software engineering process and the ability to apply it. A software system isn't the goal in itself; instead, it's a vehicle for applying the knowledge you gained in this course and improving your current skills.

It'll also teach you how to operate in a group (team skills), which is essential for large-scale software development. As you have already experienced from other courses, the collaboration between team members has both positive and negative sides and understanding both can help you prepare for your future job. Teamwork, building on each other's strengths and compensating for weaknesses, balancing different working styles, preferences, motivation levels, and so on are all challenges to excellent coordination.

Teamwork and Project Management

Team members need to split the project work into practical **activities/tasks** and ensure each member has a meaningful and essential role. Thus, all team members must participate in all project activities, and no single student should be responsible for them. Although it is acceptable to request assistance from colleagues who are more trained in technological work, each team member must be accountable for all parts of development required for the features they own (e.g., database integration, user interface design, etc.).

Project Management is primarily concerned with the organizational aspects of effective teamwork. The team members can select a "team leader" to lead project coordination efforts or split up coordination roles. The following are some of the parts of the process:

- Organizing group meetings and keeping track of deadlines
- Managing shared resources (such as website, database, software repository, etc.)
- Integrating individual contributions coherently and resolving ambiguities and conflicting information, including
    - collating different sections into a project report representing the entire team
    - proofreading the collated report to ensure consistent layout, font styles, section numbering, and language style

    - overseeing the integration efforts for different parts of the program into one software system
    - supervising the running integration tests and ensuring that the whole system works as intended
- Anything else that affects everyone in the team

You have already experienced how challenging it is to work in a team. Here are some recommendations for teamwork (none of which are required!):

- Select the team leader as the focal point. Individual skills and capabilities should be discussed (algorithm design, coding, report writing, etc.)
- Establish the meeting's agenda and time constraints. After the task responsibilities have been established, have each student report on their progress on their assignment since the previous meeting. It would help if you were at the start of the meeting. Everyone should hand out versions of their past or present slides of their diagrams or screenshots if you're working on a project report. Team members should be encouraged to communicate frequently and collaborate to solve the problem.
- Decide how the team will communicate via email, Slack (like IBM), a blog, or instant messenger. Send copies to others as you work on your project and request feedback and suggestions. Please do not wait until everything is polished and perfect; instead, inform them that this is a draft and when the next version will be available.
- Make every effort to ensure that everyone on the team is comfortable with each other's contributions; be open about any complaints. Avoiding problems by being friendly and professional in your discussions is a good concept. However, it would help if you were honest about any issues that pop up because they will affect your final grade.
- If you are unable to attend team meetings in the days leading up to a project deadline (due to travel, illness, etc.), contact all team members as soon as possible and strive to participate to the best of your abilities.
- Prepare for disaster scenarios: team members usually depend on one another's efforts. Set specific internal deadlines for providing individual components (far ahead of the real deadline) so that if one team member fails to produce, the others will have enough time to step in and finish the job.
- In general, if you're unsure, communicate! It's OK to have redundancy. Every time you send an email relevant to the project, make a copy for everyone on the team. Please don't presume they're uninterested or don't understand what you're talking about. Recognize any electronic communication that is primarily directed at you.
- Meetings with your team should be planned to include:
    - Report on the status of the project since the last meeting.
    - Come up with specific action items after discussing the tasks that need to be completed.
    - Find a time to allocate duties, prepare the portions that each person is accountable for, and ask questions to ensure that everyone is on the same page.
- Tools to assist in the organizing of teamwork include:
    - **Doodle** is a time-management and meeting-scheduling application.

    - **Slack** is a messaging application that allows you to communicate with groups of people using your computer or mobile phone.
    - ***ProjectLibre*** is a planning and tracking software.
    - **Google Drive** is a cloud-based storage service.
- It is inappropriate to claim that "**no one requested me to do this or that**" or that "**I accomplished everything that was asked of me**." Each team member should be proactive rather than passively waiting for assignments to be handed. Instead of asking others what should be done, take the initiative and suggest what should be done to ensure the success of your project.
- Take advantage of every opportunity to redistribute and/or rotate tasks, and make your suggestions known! It's sometimes more challenging to define the problem and determine what needs to be done than execute it. As a result, all team members must participate rather than rely solely on the team leader to define the problem and assign tasks.

What if your team is not functioning well?

If you find that your team isn't working well together or that it's having a negative impact on your project's performance (and your final grade), you should attempt to talk to the other team members about it. If the "problematic" team member(s) (including the team leader) refuse to comply, you should speak with the course lecturer about your concerns as soon as possible. To know more, you can review appendix A.

Project Proposal

- Each team must submit a written proposal for their project using the following Google form: https://forms.gle/wu3hcfFpU9H6MHML6
- The due date for submitting your proposal is **Oct 15th @ 5:00 pm**.

P.S. Only projects accepted by the course instructed will be permitted to proceed.

Note: It is understood by all parties concerned that the course instructor assumes no responsibility for project failure or an impasse resulting from the selected project idea and that the risk is entirely that of the proposing team. In the worst case, this could result in a mediocre project mark.

Development infrastructure, libraries, and programming languages

- The project can be developed with whatever programming language/technologies the team choose to employ to build its software product. This agreement must be recorded in your team meeting's meeting memo.
- The project requires using **GitHub**, a repository hosting service with source code management and revision control that facilitates collaboration within a project team. Each group member needs to ensure that s/he has access to GitHub, but only one designated member needs to ensure that s/he hosts a private repository for the rest of the group.
- Each group is required to use intra-group communications and collaboration tools.
- There are no constraints on the development libraries that can be used.

- There are no constraints on support infrastructures (e.g., databases), third-party tools, or off-the-shelf (or open source) software used in the project.
- The use of open-source tools and systems is encouraged.

Group member roles in the project

The project involves many different types of tasks (e.g., coding, designing, planning, testing, inspecting, documenting, and others). Thus, there are different roles (corresponding to such tasks) that group members need to assume at different times in the project.

Note that a group member can play more than one role in the project, concurrently or otherwise. Likewise, more than one group member can play the same role (e.g., multiple coders and testers, etc.). The distribution of roles will need to be adapted to the team size.

Project Binder (Deliverables)

The team members are to collaborate to create and submit a project report, the software product's source code, test cases, a PowerPoint slides, and present the proposed solution in a video presentation. All such deliverables will be sorted in a folder/binder as illustrated in the following figure:

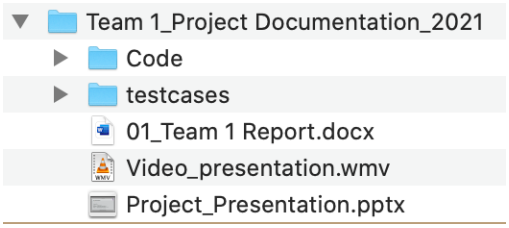

Figure 1: the project folder structure before it's compressed into a single Zip file.

The project is to be documented digitally where the files should be readable by the following applications:

- MS WORD, PowerPoint, and/or Excel.
- Acrobat Reader (PDF files)
- QuickTime or VLC player
- Eclipse or IntelliJ IDEA CE

Note:

- Those artifacts that are not readable by the above-listed applications will **not be graded** and, consequently, will receive a zero mark for that portion. Please get this right first.
- The project folder should be named: Team<number>_Project_2022<Team_Lead_ID>. The project folder should be zipped. However, it's your responsibility that the Zip file of your project is not empty or damaged. If that happens, you will receive a zero mark for the project.

Project Report Structure

The team will compile a report with the following structure:

- Cover sheet {Project **title**, Group members' names and **SID**."
- Problem Definition
- Project Objective
- Stakeholders List
- Success /Acceptance Criteria for each Stakeholder
- Use case diagram(s)
- Selected Use case Descriptions {only two descriptions}
- Sequence diagram(s) {for the selected use case for descriptions}
- System Architecture
- Detailed Class diagram(s)
- State-machine diagram {for the whole system, if possible}
- ER – Diagram (Data modelling)
- GitHub link to your project source code
- Conclusion (lesson learned)
- Reference (if any)
- Project WBS (as an appendix of your report)
- Task Assignment Matrix (as an appendix of your report)
- Sample of commits on the selected version control system (as an appendix of your report)

Note:

- Your report should be well-organized, with single-spaced paragraphs and a Caliber font size of 11 points. The **figures must be numbered & titled**, and each one must have a description of at least a couple of statements. Do not lose grades for being unorganized.

PowerPoint slides

The team will create a set of slides demonstrating the software development process they follow, the structure and behavior of various system components, and lastly, how the software product functions. The presentation should not exceed eight slides, including the first/introductory slide, and will be created with the following specifications:

- Presentation Content
  - Make sure to show the **logical partitioning** of the work among the team members: who championed which part? You can label your portion with initials or select an icon to represent yourself.
  - The slide set must be in **PPT** or **PDF** format. The first slide will include ONLY the project title and the names of your group members. You cannot have more than **eight** slides.

        - Also, your video presentation must respect the time constraint, which is firmly **five** mins (**at most**).
        - Include citations in the slides, if needed. You can add a single extra slide to list your references.
    - The presentation format is important (style, use of fonts, images)
        - Do *not* include fancy, glittering slide designs – keep them organized but straightforward, please. Black text on a transparent (white) background. Multi-coloured text is OK, e.g., to highlight specific terms.
    - Presentation (suggested) structure:
        - Describe the problem that your application is attempting to solve.
        - Use the system use case to demonstrate the capabilities of your application.
        - Discuss the application architecture.
        - Show the application's structure and the various states it can take.
        - Demonstrate your application's data model.
        - Demonstrate how your application works (Snapshots are not accepted).
        - Conclude what you learned during the development of your app.

Video Presentation

The team will also create a video presentation that utilizes the slides specified above. All the team members must participate in the video presentation. A presentation slides will be made with the following specifications:

- Logical partitioning/structuring of the video presentation matters.
- Equitable presentation-workload sharing matters, as well.
- The presentation must be in **MP4** format. Any other video format will not be accepted. The presentation must not exceed **five** minutes long (DO NOT GO OVER **5 minutes**).
- If you or any group members haven't explicitly stated that you don't want your project video presentation to be broadcast on YouTube, I'll assume that you and your entire group have agreed to submit your video presentation to the course's YouTube channel.

Project review session

- During lectures, there will be a project review session for every project. Each team should be able to report on their progress during a project review session. It's important for each team to give an update on some or all of the project's key

elements, like the work breakdown structure, the tasks they've done, the process they're using or plan to use, requirements gathering and management, diagrams they've made or plan to create, and the reporting or implementation progress. More detail will be given in class during the project orientation.

- Each team will have at most 5 mins for their session.
- The session can be **free-form** or **structured** using PPT slides.

Deadline and submission

- Due date: Nov 30 @ 5:00 pm (EST)
- Your submitted files must be consistent with OWL submission specifications.
- Submission on the dedicated dropbox on OWL. Zero marks for this course component if any of the slides, source code, report and video are missing or not submitted, even if the presentation is made or submitted corrupted.
- You need to submit ONE zipped file that includes all deliverables according to the instructions on the project handout.
- Your responsibility is to ensure that the zipped file of your submitted work and any other files contained within it are not empty or damaged and submitted on time. You can submit your file many times, and if you do, I'll presume that the most recent submission is the one you want me to grade.
- If you forget to submit your work, this is your responsibility.
- Please keep in mind that **just one submission for your course project is required**; numerous versions of the same project are not recommended.
- Remember that the grading is done on our machines and not yours.

Evaluation Scheme (100 points)

- (35%) on the project report.
- (15%) on fully commented and executable source code.
- (10%) on the PowerPoint Slides.
- (40%) project presentation: 10% project review session, 30% video presentation

Appendix A

Course Project Conflict Resolution Procedure

Group Work, Responsibility, and Peer Reviews

This project is teamwork. Individuals in the team collaboratively create different parts of the system into an integrated whole. All the group members are thus expected to contribute earnestly according to the plan.

If a team member has a complaint such that in their interpretation:

- There is a significant complaint from the rest of the team (not only one person);
- The complaint is made to the instructor in writing – by the rest of the team -- at the time of the occurrence of the negative situation.
- There will be a peer review form distributed among the team members; and
- After a preliminary inquiry by the instructor, the instructor is "convinced" that the situation warrants a review. The instructor's decision is final and may not be appealed.

The project mark given to the individuals in the team is the base project mark adjusted by peer-review feedback received from the group.

- The peer-review penalty is as follows -- % grade of the overall project mark reduced:
    - **Minor infraction**: 10%
    - **Significant infraction** but not considered major: 30%
    - **Major infraction**: 70%

Note:

- All interpretations will be made by the instructor and may not be appealed.
- A second-time infraction for the same person will automatically lead to a "major infraction" if the previous case was assessed at "minor or significant" levels.
- If the previous case was assessed as a "major infraction," this will automatically result in a zero mark for the project for the person concerned.
- The best policy is to collaborate for the team's success.