

Name: Tan Michael Olsen

UCID: 30226343

## **Introduction**

The project that I have been working on with my group mates throughout the Winter 2024 semester for the CPSC 233 course, is a grade tracker program that allows instructors to store the grades of their students. It is similar to how D2L works for instructors, where they can store their student marks, which is the goal of our project. When I was starting this project, I was quite challenged to design the flow of the program, as the use of database (a crucial element for a program like D2L) are not allowed in this project. However, my group mates and I were able to figure out an alternative in designing the backend of the program without the use of database.

## **Group Collaboration**

Collaborating and communicating with other group peers is the key to success in this project. The more often we collaborate the better result we have. My group would often hold a group meeting (usually every day, until we as a group feel that we are done with the stages of our project, what I mean by stages is like demo1, demo2, or demo3).

During our group meetings, we would discuss what each of us must do, and make sure that every single one of us understands our “part to contribute” to the project. We would also explain our code to our peers in the case that our peers “part to contribute” is dependent on the codes that we have made. In addition to that, during every group meeting, we also share the accomplishments that we have made towards the overall project. This allows us to always be in track in the development of our project. We mainly rely on discord as a platform for our communication.

In terms of collaboration, in my perspective, I often try to make my peers understand my ideas in developing the project, which is one of my greatest challenges in this project (Some of my ideas are hard to understand for some peers). It is not rare for me to draw out the flow diagram (or data structures) of the program on a piece of paper and share the diagram with my peers. It is easier for them to understand in visuals rather than I’d be explaining to them by words. I would usually draw them on a piece of paper. The part where I greatly contribute in this project is the designing of the backend system of this program, which includes on how the data is being stored and processed when the program is running. I also contributed substantially contributed on the designing of the GUI of this project, and how the GUI works.

There were quite a few other challenges on doing this project. During the initial phase of this project, we are quite confused on how the merge works in GitLab. But after several trial and errors, we are able to figure out the solution the merging issues with had. Another challenge in this project, there could be at times where my idea clashes with other peers. When this happened, we would compare and see who's idea would work better for the project, or we could also at times combine the two ideas together for the project.

## **Technical Reflection**

This project has 3 phase, phase 1 being the procedural version of the program, phase 2 the implementation of OOP principles into the program, and the last phase, phase 3 the implementation of GUI to take user input. Each phase has its own challenges, and new lessons to be learned.

I would say that the first phase is one of the toughest phase throughout the project. As our program is to make a D2L program for instructors to store grades of each student, there are quite a couple of basic features that the program needs to have. For my part these features includes a feature that allows an instructor to select a course to teach (as well as remove the courses selected), adding a student to a course (as well as removing a student from a course), modifying and adding a specific student grades. These features are not easy to make and require quite an extensive thinking on the algorithm. As the algorithm is quite complex, it also has quite a long line of code for each feature, which may be hard to debug if an error occurs. With this experience I got from procedural programming, I learned that procedural programming concept may not be an efficient way to code for a huge complex program.

The Second phase of the project, being the implementation of OOP principles, were quite confusing at first. My group mate and I were quite confused on how to transform the procedural concept of our program into an OOP concept. So instead of directly starting phase 2 right after demo 1, we as a group decided to start with the second assignment to grasp the OOP concept required. Things went well as according to plan, after finishing the second assignment, we were able to implement our program into OOP principles. My group decided to make a master data class (which has a similar role to galactic map class in assignment 2) to store all the students and courses, and as well as different classes (Like the battle ships in assignment 2) for each course. The different class for each course represents a different grading system for the course itself. We have 3 different classes for the courses available to select. At the end, we are able to

implement the OOP concept into our project. By completing this phase, it made me realize that OOP is a way more convenient way to code, and as well as more efficient compared to procedural. OOP allows us to break codes in different classes, which may simplify the program as an overall.

The last phase being, phase 3 is to implement GUI as the source of data input from the user. We didn't really implement the design we had in assignment 3, but we use some similar elements that we had in assignment (adding a listener to a button, as well as text entries). As a group, we design the GUI so to have a main page, as well as each separate pages for all the basic features we have (such as adding a student, removing a student, modifying and adding grades to a specific student, etc.). So when the user press the add a student button (as an example), it will lead to the add a student page. As a result of this, we have multiple fxml and controller files. The challenge at this phase is to be able to link all the fxml files together, so that if a button is pressed another page is loaded, and that it will still be able to go back to the main page. This challenge was overcome by understanding how the fxml loader function works and as well as the fxml initialize function. We managed to be able to link all the fxml files together. There are a lot of lessons learned at this phase, which there is one that I like the most. That is, working on the GUI stage has envisioned me that validating user input in GUI form is much easier than in command line interface, as in GUI you have the option to disable elements. Another lesson that could be pointed out, is that in designing the GUI is better to not have a lot of empty space. This can improve the user interface experience.

## **Git Usage**

Git and GitLab were essential tools in our project's success, providing efficient version control and enabling seamless collaboration within our team. Git facilitated easy file sharing and tracking of edits, ensuring everyone had access to the latest project version, streamlining our workflow and allowing concurrent work on different aspects.

A crucial feature was Git's ability to revert to previous commits, which proved invaluable when errors were introduced. By quickly reverting to stable commits, we avoided major setbacks. Initially, we faced challenges with merging changes from different branches due to our limited experience, resulting in conflicts and code inconsistencies. To overcome this, we dedicated time to learning Git's merging capabilities, experimenting until finding effective methods. This process enhanced our understanding of merging mechanisms and streamlined our workflow.

Maintaining a single branch in our Git repository proved beneficial, as each team member primarily worked on individual Java classes. Multiple branches seemed unnecessary and could complicate version control. Instead, we prioritized well-documented commits and effective communication within the team.

In conclusion, Git played a pivotal role in facilitating collaboration and version control. Despite initial challenges, our perseverance and learning mindset enabled us to leverage Git effectively, ensuring project success.

## **JUnit Testing**

JUnit Testing came handy in the development of the project. JUnit Testing enabled my group to create multiple independent tests, which this allows us to catch any flaws that our algorithm (or code) have. In phase 2, my group made around 5 JUnit Testing for each main core function of our program, as we want to ensure that the program works as what we expected. A time when JUnit Testing became very useful was when I wanted to test for a function that calculates the total grade of a student from the student grade list. This function is known to be crucial to calculate the top 5 performing students. I tried making several students and inputting several sample grades for each student to calculate their total grade. This test that I made gave me an insight into how my function works as what I expected it to be.

## **Project Outcome**

The project went smooth and successful from my point of view. At first, we do have a lot of ideas on what to develop for this project. We initially thought to be having 2 sides of the program, one being the student side, where they can view their grades, and the other being the instructor side, where the instructor can input and modify the grades of the student. However, due to time constraints, my group decided to only work on the instructor side for the project. The development and functionality of the instructor side went along as what we expect the program to be like, indicating a great success.

If given more time (and the fact of having a less hectic schedule) my group would have also done the student side and created a sort of log-in system for each different student and instructor. I would also implement the use of a database if there were no restrictions in the library that we are allowed to use for this project. Regardless, the project is still a great success!

## **Skill Development**

This project has given me the opportunity to assess and improve my quality of work, working in a team environment. Prior to this project, I have always been developing coding projects individually. At first, I didn't seem to enjoy working in a team environment that much, as I couldn't be as independent as I were to be working alone. But as time goes by, it seems to be that I am enjoying it. Working on a team environment allows me to facilitate the group ideas, and as well as facilitating the needs of other peers. What I meant by this is, as an example, I was mainly responsible for handling the data of the program. One of my peers needed to do a function to calculate the overall grade of a student. To facilitate my peers, I created a function to calculate the overall grade of a student. This is very impactful because by having this function, he is now able to develop codes that calculate the course insight (top performing students, course average, course median, etc). In addition to this, there were times where my peers would rely on the use of my code, some of them couldn't really understand what my code does. In the spirit of making this project successful, I gave them a thorough explanation of what my code does. By this experiences, it has allowed me to improve my sense of team work working in a team environment.

## **Personal Growth**

This project has enhanced my skills and understanding of the programming principles learnt in class. It has also given me the opportunity to implement what is being taught in class, rather than just memorizing, and understanding how the theory works. By completing this project, it has broadened my vision in the implementation of different programming concepts. I was truly engaged in developing the program in the implementation of OOP. I was amazed how things could be done more effective and efficient when using the OOP approach instead of procedural. Prior to this course, I have always been programming in a procedural way. After knowing that OOP is much more efficient, I would probably start my individual projects by implementing OOP.

## **Final Thoughts**

The idea of having a project on this course is excellent. It gives an opportunity for students to work in a team environment, which eventually what computer scientist would do (working in the computer industry). From my perspective, this course is dynamic. This course doesn't give students a heavy academic pressure, however it broadens students' experience on programming through its assignments and projects. I'm looking forward to more courses of this type of teaching methodology.