COMP 2522
Object Oriented Programming 1
Term Project

Yue Wang
yue_wang@bcit.ca

February - April 2023

1 Introduction

For your COMP 2522 term project I would like you to make me a game. You must create a fun and unique desktop single-player game using Java and a solid OOP design, with a riveting UI/UX implemented using JavaFX, and some sort of database persistence that lets me save my game and revisit it after a reboot. Read on, intrepid developers!

2 Submission requirements

Your term project must be completed on or before 23:59:59 on Sunday April 9th, 2023.

There are milestones that must be met during each of the following six weeks of class.

Code and documentation will be submitted via D2L.

3 Schedule

Feb 27 - Mar 05: Brainstorming and ideation

Mar 06 - Mar 12: Design and Coding

Mar 13 - Mar 19: Coding

Mar 20 - Mar 26: Coding

Mar 27 - Apr 02: Coding

Apr 03 - Apr 09: Coding and submission.

4 Setup

Please set up your term project in the following manner:

1. **Choose a partner and join the same group with your partner on D2L**.
2. **Create a new JavaFX project in IntelliJ**. One of you must create a new Java FX project. There are reasonable instructions here: `https://www.jetbrains.com/help/idea/javafx.html#create-project`:
   (a) The name of the project must be 2522-202310-Term-Project-XXX where XXX is some unique and interesting identifier you choose

(b) The Language must be Java

(c) The Build System must be Maven

(d) The Test Framework must be JUnit

(e) The Group must be ca.bcit.comp2522.termproject

(f) The Artifact must match the XXX unique identifier you chose for the final token in the name of the project

(g) Use Java 18 for the SDK

5 Steps

I want you to make me a game. It must be unique. You may not copy or clone any existing games. It must be a product of your imagination. This will go into your portfolio, so this must be your own work:

1. **During week 1**, I would like you to complete your ideation. Brainstorm ideas with your partner. Do research. Play some games. Identify what's fun. Identify which games out there are written in Java. Using `https://coggle.it`, create a no holds barred **mindmap** during a brainstorming session. Start with a single word or phrase that captures a fun main idea, and then explore everything else that comes to mind. Don't censor yourselves! This is the time to go off on wild long tangents. Go down every rabbit hole. **You will submit your brainstorming document and some additional project pitch documentation at the end of week 1**.

2. **During week 2**, I would like you to create a **UML diagram** that contains the main structure of your Java game. Consider all of the actors, behaviours, relationships. Think about classes, inheritance, and interfaces. Now is also the time to consider your entity relationship diagram making skills and what sort of database information you will need to store. Design your game. **You will submit your design and any supporting documentation at the end of week 2**.

3. **During week 2, you will also conduct your first planning meeting**. You and your partner must create a **Trello board** and populate it with Epics (big user stories), Features (smaller user stories that make Epics real), and Tasks (30-minute to 2-hour activities that directly implement Features). Invite me to your board and get ready for me to examine it thoroughly. You will need to identify a few user stories and the tasks to implement them for each week. At the end of each week, you must deliver a working piece of code where those milestones have been met.

4. **Weeks 3, 4, 5, and 6 are for completing your work**. At the end of each week we will compare your progress to your plan. You will try to meet your milestones.

5. **You and your partner will give a presentation (up to 5 mins) about your project during the lab period at the end of the term**.

6 Grading scheme

Your term project will be marked out of 15.

1. (1 point) Week 1 brainstorming documentation
2. (1 point) Week 2 design documentation
3. (2 points) Weeks 3 - 6 Agile project management
4. (9 points) The quality of your software.
   (a) (1 point) Team developed an app that was reasonably complex
   (b) (1 point) The UI/UX is intuitive and appealing
   (c) (1 point) Team accomplished the task that they set out to do
   (d) (1 point) Team showcased elements of creativity and originality

(e) (1 point) The methods are tested using JUnit

(f) (1 point) Team provided correct and complete Javadocs

(g) (3 points) The code and resources are tidy and structured well, including meeting non-functional Java OOP design requirements such as encapsulation, information hiding, minimizing mutability, maximizing cohesion and minimizing coupling, making good choices about associations and thinking about Demeter, etc.

5. (2 points) Presentation