PROJECT REPORT

PROJECT DETAILS	
Project Name	MineSweeper
Project Manager	Vincent Zhang
Date Submitted	August 23, 2023
Project Description	Project MineSweeper is a recreation of the classic logic game "Minesweeper", first developed and released by Microsoft in the 1990s. The object of the game is to uncover all tiles on the playing field while avoiding the "mines" that are randomly distributed throughout the field. The player is able to deduce the locations of mines based on the numbered labels on uncovered tiles, which indicate the number of mines adjacent to that tile.

TEST PHASE SUMMARY	
Date of Test Phase	August 22, 2023
Tester Name	Kai Li
Tester Occupation	Asset Manager
Test Phase Findings	During the testing phase, all major functions of the application were tested to the extent of the usage behaviour of a normal user, albeit with some degree of rough play. Beyond the expected usage of the program, some deliberate actions were conducted with the intention of breaking the application. For instance, at one point during the testing phase, the buttons of the graphical user interface were repeatedly, rapidly, and randomly depressed by various mouse buttons, including the left and right click buttons, the scroll button, and the side buttons, which, much to our confusion initially, revealed the presence of a minor graphical bug which was solved by the addition of 6 lines of code. The issue occurred when the recursive method used to uncover tiles encountered a flagged tile that did not, in reality, conceal a mine, whereupon the uncovered tile displayed some irregular and incoherent red dots instead of the expected number. Besides this bug, there were no other major issues discovered during the testing phase. There were, however, complaints that the

	application window could be bigger, since the current version of the application supports neither fullscreen nor window resizing. There were also some complaints about the clarity of the labels on certain menus. For example, the function of the “Save and Quit” option in the drop-down settings menu was mistaken for an option to save the current score to the score record file. Nevertheless, it was later acknowledged that the overall thematic design of the application could be considered more appropriate and intuitive for younger generations of users. The tester furthermore acknowledged that they appreciated and enjoyed logic and puzzle games such as the project they were testing, but that they would rather use the one available through the Google search engine, which employed better animations, and which was (subtly) the main inspiration for this project. Overall, the tester recommended that a function should be implemented to increase the size of the game display. Excepting the minor graphical glitch explained above, the tester received the project very well and was unable to “break” the application using conventional methods. For the next version of the MineSweeper project, it has been recommended that the game implement more animations for better audience engagement. Due to the capabilities and limitations of the Java Swing package, however, it is difficult to implement animations into the game. Future project development teams may consider using different Java classes, or even entirely different programming languages, that are more suitable for game animation. Future development teams may also consider implementing functions to increase the size of the display, as it has been recommended.
PROJECT MANAGEMENT REVIEW	
Project Review	Of the five main subtasks outlined in the Statement of Work (SOP), three were completed successfully—namely, a) that the application implement a feature to save the current game data to an external file, b) that the application implement a timer, and c) that the application include a hidden menu to access various features. The two subtasks that were not completed successfully were: a) that the application include options to increase or decrease the difficulty of the game, and b) that the application implement a function to toggle fullscreen. These were not, during the development phase, determined to be necessary features for the project release and were therefore removed. As for back-end

	algorithm design, there were virtually no difficulties in their implementation, which project management was greatly pleased to see.
Project Successes	As briefly mentioned above, all back-end algorithm design and implementation proceeded without any major difficulties. Project management was especially pleased to oversee the development of the recursive algorithm used to uncover tiles, which worked exceptionally well.
Project Difficulties	Of the five main subtasks outlined in the Statement of Work (SOW), two were not successfully completed. The first was the ability to change the difficulty of the game, while the second was the function to resize the application to fullscreen. These, as mentioned above, were not considered necessary functions for the release of the project and were therefore excluded to save, above all, time, which was better spent on improving the efficiency and robustness of the back-end algorithms. Another difficulty was the implementation of animations, since the Java Swing package is not exceptionally convenient for animations.
Future Project Recommendations	Perhaps the most important lesson the project development team received was the significance of planning in the world of Object Orientation (OO). Initially, project management planned for only three Java class files, whereas the current release uses six. There is much room for improvement as it relates to encapsulation and communication between class files.