Hi! My name is Damon Murdoch, and today I’m going to be doing a basic overview of my minesweeper game.

The game runs through three source files – ms.py, which handles all the classes and functions for standard and hexagon based minesweeper and cs.py, which handles all the classes and functions for colour based minesweeper and main.py, which handles all the interface objects and references the functions and classes in main.py and cs.py to run the game. An additional file ‘test.py’ is utilised to perform unit testing on some simple functions in the program. The hexagon and standard minesweeper games have been fully implemented, however there are some issues related to the colour based minesweeper implementation.

Before I start the game, I’ll do a basic demonstration of test.py so you can see what it tests exactly. It basically operates on simple functions related to the square and colour\_square classes, and tests the solution function in ms.py.

\*run test.py\*

Okay, all tests passed! Now that’s done, I will start the game. You will see that there are four different menus in the application – Standard, Hexagon, Colours and High Scores. Each menu is used to either play a different version of the minesweeper game, or to view the high scores stored in the database scores.db.

We will first look at the standard minesweeper game, which I will play on the default settings of a 9x9 grid and 8 bombs. I’m not very good at playing minesweeper, so I will fail this one and then set the grid size to 2x1 so I can prove that the game winning code functions.

\*play game and lose\*

Ok, so as you can see when the game is lost, a popup window appears on the screen and all the grid is revealed. We can click start again to restart the game, but I’ll pick an easier grid size this time. The random number generator always puts a bomb in the top left-hand corner for 2x2 minesweeper, so if we flag that space we will win instantly.

\*flag space\*

Excellent! As you can see, it displays the time taken for us to flag that space, and that score will now be inserted into the database. I’ll be visiting the high scores menu momentarily, but first I will show you the hexagon minesweeper game.

\*play game and lose\*

Hexagon minesweeper is almost functionally identical to standard minesweeper, and the algorithm is somewhat less complicated than the original. I will again play the game on the standard grid size, which I will probably lose and then demonstrate the win condition on the 2x2 grid.

\*flag space\*

Cool. As you can see it displays the same popup it gave us for standard minesweeper, and it will insert the score into the database. I’ll now demonstrate the implementation for the colours minesweeper game. This game is very difficult, so I will not be able to complete it without cheating a little like the others and choosing a small grid size. First off, I’ll demonstrate the game on a regular size board.

\*play game and lose\*

Same window as displayed on the other gamemodes pops up here too. I’ll just change the parameters now and redemonstrate the game.

\*flag space and win\*

Alright great, now I’ve demonstrated all the gamemodes. Now I‘ll demonstrate the high scores table. By default, it selects the 9x9 with 8 bombs however, I don’t have any scores in the database for that so we’ll look at the 2x2 with 1 bomb table. Now, several scores come up – All for that set game mode. I’ll now search for scores from each of the other gamemodes.

\*search for scores\*

Done! Now, I’ve demonstrated all the features for my minesweeper game. In the report, I mentioned that I would discuss the principle of psychological acceptability as it was difficult to pin down to a single section in the report. The general user interface of this application is reasonably user friendly, and there is no reason for the user to try and find shortcuts in the interface. The only interface where I believe this could be questioned is the high score table, where the user is required to enter game types manually. This is an issue which would be improved in future versions of the application.

Now, I hope I have provided a satisfactory explanation of all the features contained within the game, and any additional documentation can be found within the accompanying report. Thanks for watching, and I hope you enjoyed the presentation.