The title of my project was Hangman Game Project. Throughout the development of my project I used NetBeans IDE 12.6 and JDK 11. This allowed me to create a robust and frictionless program. To ensure that my project could not be lost or broken, I backed my files up on One Cloud. This was especially helpful when I changed something and the code broke. I was able to go back to a previous version and reinstate it. My dependencies were Java Standard library which I utilized for game logic, file handling, and user input.

The file structure of my program includes the overall src file which is named hangman project, HangmanGame.java, Player.java, Wordlist.java, Filemanager.java, Game.java, Statistics.java, HangmanDrawing.java, Hints.java, Name.java and words.txt. HangmanGame.java included the main method, the game logic and all of the user interaction. For example, the HangmanGame class manages the game loop, game reset functionality, prints all questions, and stores and input. Player.java manages player states like guessed letters and attempts left. WordList.java handles word selection from the external file - words.txt. To make the program more robust I also added error handling incase of an absent word file, etc. FileManager.java is responsible for doing all file- related operations. It most importantly manages reading and writing operations of word files. An important method in FileManager.java is readLines(String filePath) method. It loads the list of words for the Hangman game and returns them as List <String>. Game.java is implemented to demonstrate inheritance in my program, which was a requirement of the submission, I did not have inheritance elsewhere in my project. Statistics.java tracks and displays the game statistics (e.g number of wins and losses). HangmanDrawing.java shows a hangman character being drawn in the console throughout the game. Hints.java manages the hint system, if you need help you can type “hint” and a letter will be shown. Name.java handles the input of the users nickname so it can be displayed. Words.txt is a text file of words for guessing in my hangman game.

Throughout the duration of the first assessment, I conducted testing for different game scenarios correct/incorrect guesses, game win/loss conditions. This testing was done manually. This testing helped me debug issues related to file paths and user input, as well as improving robustness. I also added comments throughout so markers can understand each part of the code in depth.

To execute my game you must run the HangmanGame.java class as it homes the main method. When executed my game reads words from words.txt, tracks player guesses, and displays player results.

Contribution: 100% of the project, including design, implementation, testing, and documentation.