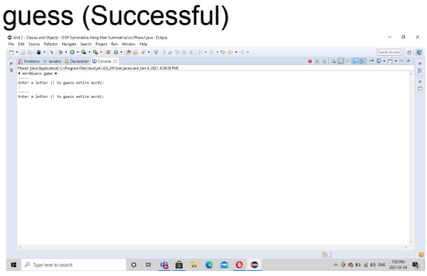
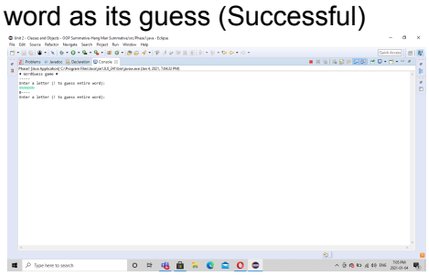
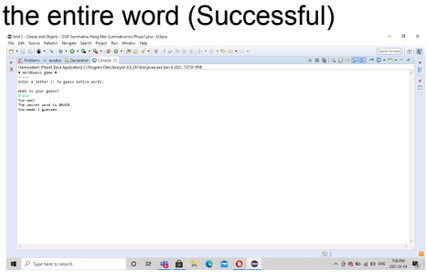
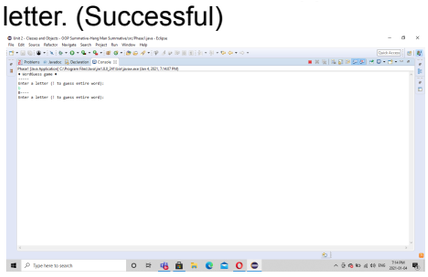
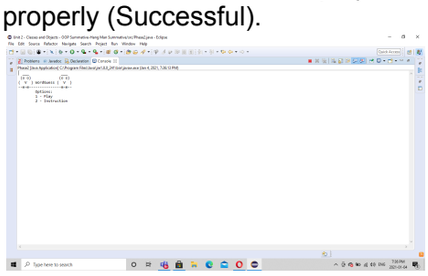
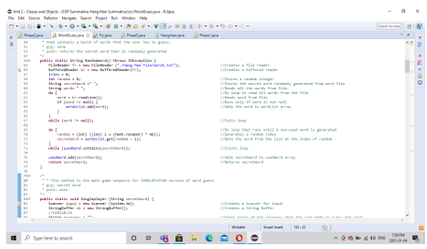
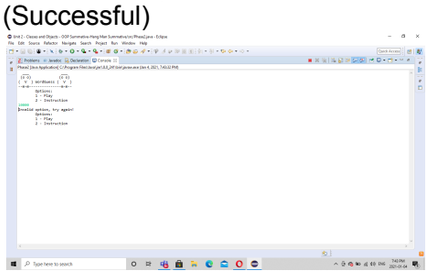
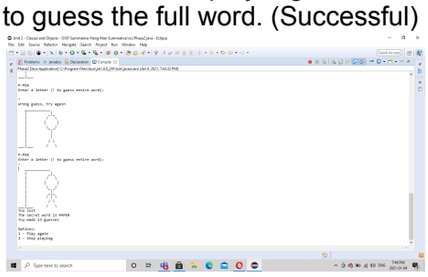
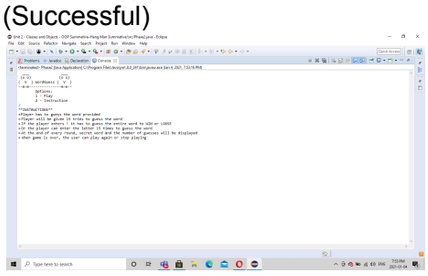
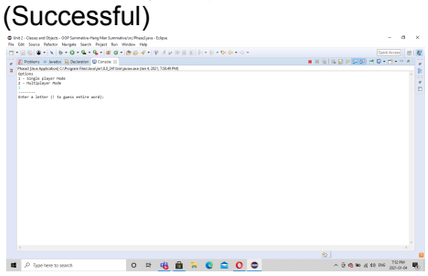
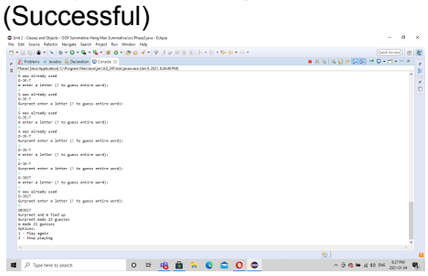
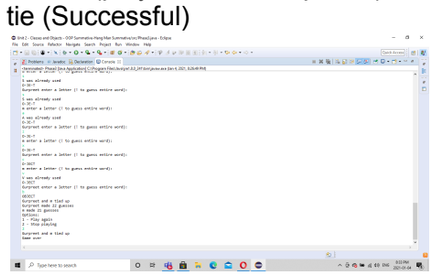
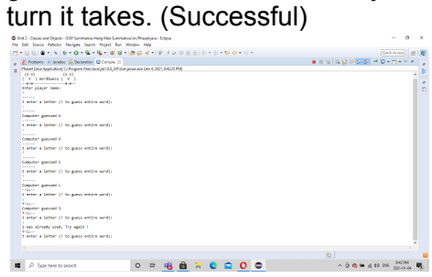
Hang Man - Test Cases

Test Cases	**Scenario**	**Result** (Was it successful? How do you know?) You can include a screenshot if you would like.
Phase 1	**Word guess for "BRAIN"** **Case 1:** What happens if I enter an invalid letter ? **Case2:** What happens if the user types more than 1 letter per guess ? **Case 3:** What happens the user enters "!" ? **Case 4:** What happens if user guesses a lower case letter	**Case 1:** Treats it as an invalid guess (Successful)  **Case 2:** Takes the first letter of the word as its guess (Successful)  **Case 3:** Allows the user to guess the entire word (Successful)  **Case 4:** Lowercase letter gets treated the same as an uppercase letter. (Successful) 
Phase 2	**Single Player** **Case 1:** Does the title display in a proper format **Case 2:** Do the same words get generated twice ? **Case 3:** What happens if the user enters an invalid option ? **Case 4:** Is there a defined way to win or lose ? **Case 5:** Does the instruction option work?	**Case 1:** Yes the title is displayed properly (Successful).  **Case 2:** No, the used words get stored in an array that compares each new word to make sure it does not get repeated again. (Successful)

Hang Man - Test Cases

		 Case 3: The game outputs an error and prompts the user for option until a valid option is entered (Successful)  **Case 4:** Yes, the player gets 7 tries to guess the full word. (Successful)  **Case 5:** Yes, the instruction option gets displayed when picked. (Successful) 
Phase 3	**Multiplayer** **Case 1:** Does the option for single player and multiplayer work ? **Case 2:** Does the game keep track of both players' guesses ? **Case 3:** Is there a defined way for each player to win ? **Case 4:** Does the game keep track of user points ?	**Case 1:** Yes, both single player and multiplayer option word (Successful)  **Case 2:** Yes, both player guesses are kept track by individual variables for both players and displayed when the game ends. (Successful) 

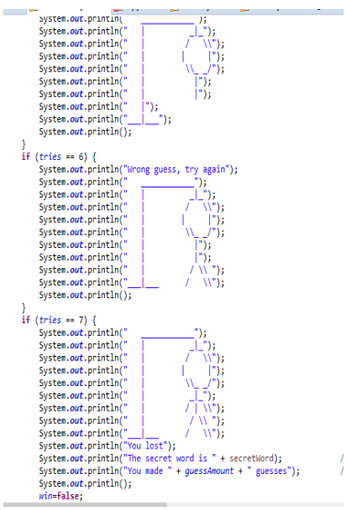
Hang Man - Test Cases

		Case 3: In order to win the player must guess more correct letters than the other player. (Successful) **Case 4:** Yes, the game keeps track of the user's final score and outputs the winner (player with most points) or tie (Successful) 
Phase 4	**Computer** **Case 1:** Does the computer guess a letter that is already guessed/used ? **Case 2:** Does the game declare the winner and run basic game tasks ?	**Case 1:** No, the computer does not guess any used letter and it generates a new letter for every turn it takes. (Successful)  **Case 2:** Yes, the game class declares a clear winner and runs basic game tasks such as generating random words, displaying options etc.

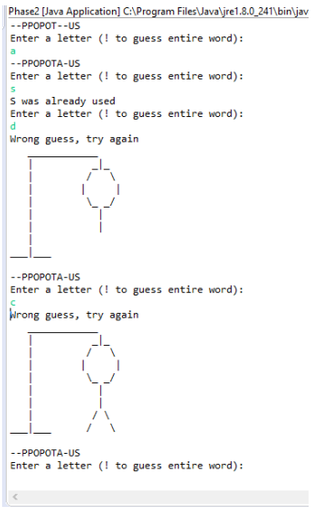
What is your additional feature? How does it work?

In order to improve my game and make it more enjoyable for the user to play I have added some additional functionality to my game. To start with, have added a visual representation of hangman for phase 2. The way this works is that I have created a method in my WordGuess class called Hangman(). This method receives a counter variable called tries which tells the method which image to output based on the user's tries. To add on, this method has 7 different representations of hangman images and everytime the user guesses a wrong letter, the counter variable increments by one and the method outputs the image based on the counter variable. Finally this method will run until tries=7 unless the user guesses the word correctly and wins within these 7 tries. Otherwise the user loses the round and gets displayed the full image of hangman and the game ends. In addition, to make my game more enjoyable, I have appended different sound effects in the game. The way this works is that I have made 4 separate sound files which all contain different sound effects and have different meanings. In order to access these sounds I have created four different methods for each sound file and they all run when they are called from the main game sequence. For instance, if the user enters a wrong letter, a sound effect will be played indicating that the user has entered a wrong letter. Therefore, not only does this make the game more gratifying to play, but it also gives the user more information about the guesses, wins, and losses in terms of sound. Overall, I hope that these additional features make the game a lot more satisfying and enjoyable to play for the users and it also provides some sort of information about their progress in the game.

Hang Man - Test Cases



```
        System.out.println("    _________ );
        System.out.println("   |         _|_");
        System.out.println("   |        /   \\");
        System.out.println("   |        |    |");
        System.out.println("   |        \\_  _/");
        System.out.println("   |          |");
        System.out.println("   |          |");
        System.out.println("   |");
        System.out.println("___|___");
        System.out.println();
    }
    if (tries == 6) {
        System.out.println("Wrong guess, try again");
        System.out.println("    _________ ");
        System.out.println("   |         _|_");
        System.out.println("   |        /   \\");
        System.out.println("   |        |    |");
        System.out.println("   |        \\_  _/");
        System.out.println("   |          |");
        System.out.println("   |          |");
        System.out.println("   |         / \\ ");
        System.out.println("___|___     /   \\");
        System.out.println();
    }
    if (tries == 7) {
        System.out.println("    _________ ");
        System.out.println("   |         _|_");
        System.out.println("   |        /   \\");
        System.out.println("   |        |    |");
        System.out.println("   |        \\_  _/");
        System.out.println("   |          |");
        System.out.println("   |         / | \\");
        System.out.println("   |        /  |  \\");
        System.out.println("___|___     /   \\");
        System.out.println("You lost");
        System.out.println("The secret word is " + secretWord);
        System.out.println("You made " + guessAmount + " guesses");
        System.out.println();
        win=false;
```



```
Phase2 [Java Application] C:\Program Files\Java\jre1.8.0_241\bin\jav
--PPOPOT--US
Enter a letter (! to guess entire word):
a
--PPOPOTA-US
Enter a letter (! to guess entire word):
s
S was already used
Enter a letter (! to guess entire word):
d
Wrong guess, try again
    _________
   |         _|_
   |        /   \
   |        |    |
   |        \_  _/
   |          |
   |
___|___

--PPOPOTA-US
Enter a letter (! to guess entire word):
c
Wrong guess, try again
    _________
   |         _|_
   |        /   \
   |        |    |
   |        \_  _/
   |          |
   |          |
   |         / \
___|___     /   \

--PPOPOTA-US
Enter a letter (! to guess entire word):
```

Reflection

How successful is your program overall? Explain the strengths and weaknesses of your code in the box below. What could be improved upon?