Hangman Testing

Key

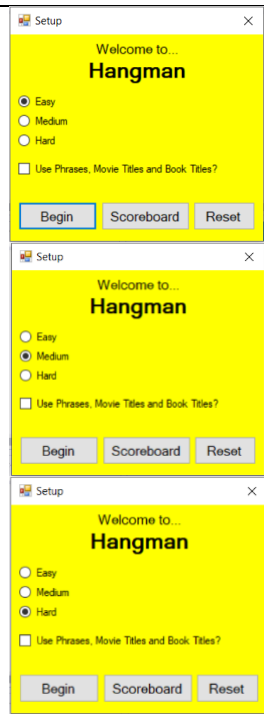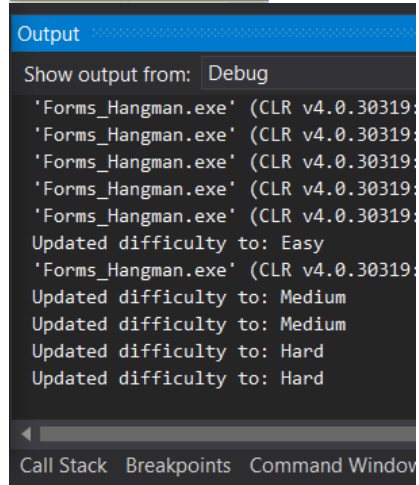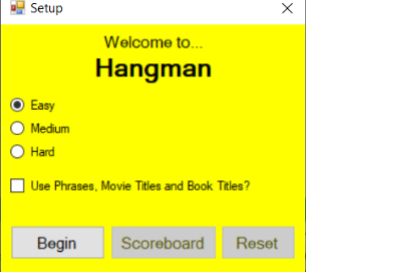**Green** - Typical Interaction
Orange - Erroneous Interaction
Red - Extreme Interaction
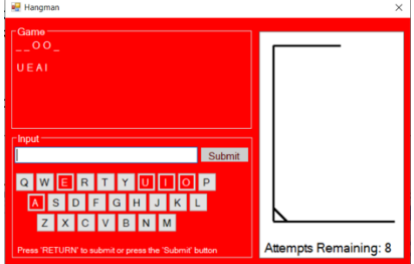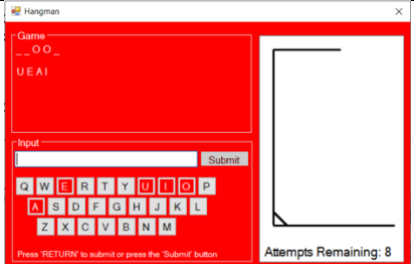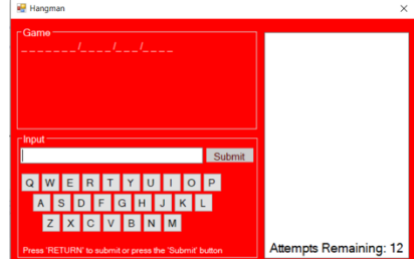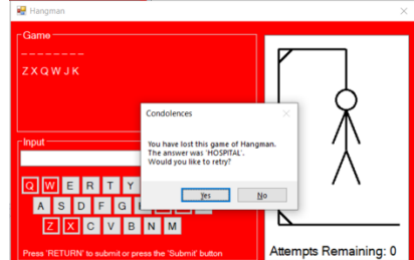
Test Table
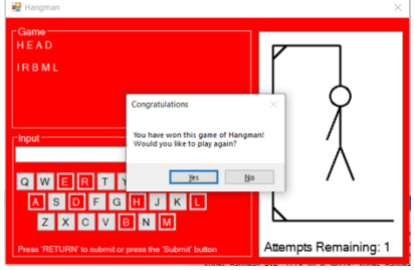
Test No.	Data Type	Expected Outcome	Actual Outcome	Corrective Action/Notes	Evidence
Setup					
1	**Typical**	The difficulty should change respective of the difficulty radio buttons	The difficulty does indeed change correctly	None (00:21)	

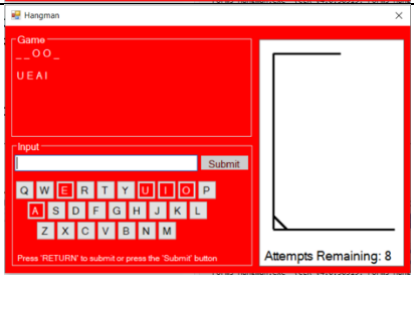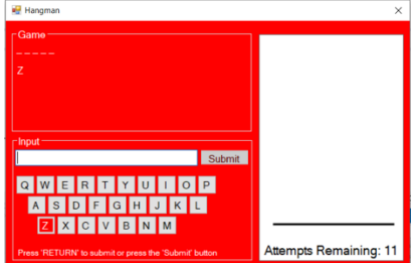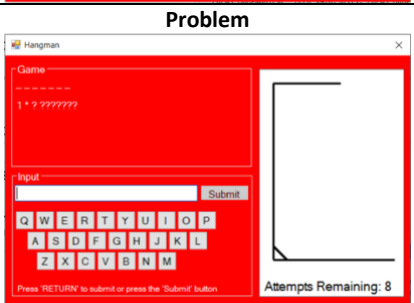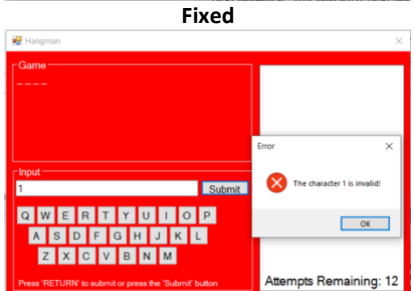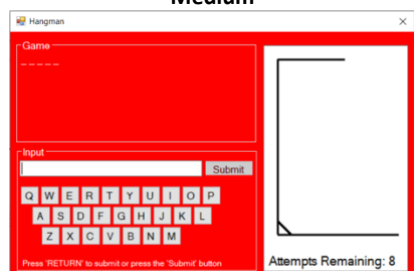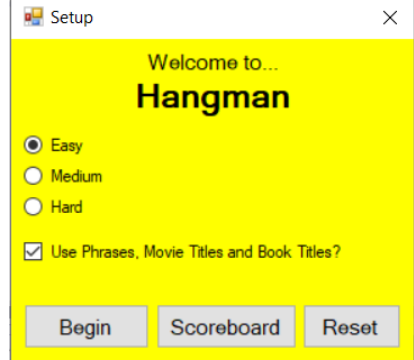| 2	Typical	The 'UsePhrases' Boolean should change respective of its checkbox	The 'UsePhrases' Boolean value is updated appropriately	None (1:04)	

Output window contents:

```
Output
Show output from: Debug
'Forms_Hangman.exe' (CLR v4.0.30319: For
'Forms_Hangman.exe' (CLR v4.0.30319: For
Updated difficulty to: Easy
'Forms_Hangman.exe' (CLR v4.0.30319: For
Updated difficulty to: Medium
Updated difficulty to: Medium
Updated difficulty to: Hard
Updated difficulty to: Hard
Updated UsePhrases to True
Updated UsePhrases to False
Updated UsePhrases to True

Call Stack  Breakpoints  Command Window  Im
```

3	Typical	When there is either no 'scoreboard.txt' file, or the file is empty, both the 'Scoreboard' button and the 'Reset' button should be disabled	When there is no 'scoreboard.txt' file, the 'Scoreboard' and 'Reset' buttons both disable	None (1:46)	

No 'scoreboard.txt' file

Both buttons are now disabled

Main

4	Typical	An onscreen keyboard should be generated at runtime when the window first opens, preferably using 'QWERTY' keyboard layout	An onscreen keyboard is generated in the 'QWERTY' keyboard layout as intended	None (2:52)	

5	Typical	User interactions with the onscreen keyboard should insert the respective characters into the text box	The keyboard successfully types the correct letters into the text box when clicked on	None (4:25)	
6	Typical	After being checked, the contents of the input text box should be cleared	The text box does indeed clear after the submit button is pressed	None (4:54)	
7	Typical	After the submit button is pressed, if a single letter is in the text box, the respective keyboard button should change colour to notify the user that they have already picked that letter	The corresponding buttons do indeed change colour correctly.	None (5:18)	
8	Typical	The submit button should be disabled if no text is in the text box	The submit button is disabled when there is no text in the text box (this does not include whitespace)	None (5:59)	
9	Erroneous	Entering numbers and other unexpected characters into the textbox should show an error popup	The error popup did not show, and the unexpected character was added to the list of incorrect characters	The submitted inputs are now checked character-by-character using the 'Asc' method to check whether the letter is in the alphabet (6:37)	**Problem** **Fixed**

10	**Extreme**	Entering more than one character into the text box should show an error popup, unless the number of characters is the same as the number of characters in the word	The error popup window successfully displays	None (7:45)	
11	**Typical**	If a letter or word is correct, it should replace the corresponding dashes in the game view	The correct letter(s) and words are placed in the correct positions in the label	None (8:55)	
12	**Typical**	If a letter or word is incorrect, it should be inserted in the incorrect inputs label in the game view	Any incorrect inputs are appended to the incorrect inputs label	None (9:39)	
13	**Typical**	The number of attempts should decrease if a valid incorrect input is entered	The number of attempts do decrease when incorrect inputs are entered	None (10:39)	
14	**Typical**	The gallows should update if an incorrect input is entered	The gallows do indeed update when incorrect inputs are entered	None (11:17)	
15	**Typical**	The gallows and number of attempts should start at either twelve, eight or six depending on difficulty	The gallows are pre-drawn or partially drawn when using medium and hard difficulties,	None (12:24)	**Easy**

				which is expected		

Medium

Hard

16	Typical	When the 'UsePhrases' Boolean is true, a phrase, film title or book title should be used to guess instead of a single word	This Boolean value does show either a phrase, film title or book title each time.	None (13:04)	

17	Typical	When the number of attempts remaining reaches zero, a 'Game Over' popup should appear	A 'Game Over' popup window appears when there are no more attempts remaining	None (13:54)	

18	Typical	If all the letters are the same as the word or phrase to guess, a 'You Win' popup should appear	A 'You Win' popup window appears as expected when the word or phrase is guessed	None (14:49)	
19	Typical	When a game is won or lost, if the 'scoreboard.txt' file is empty, it should be updated to include the number of games won and lost, and should also include the word and how many attempts it took to guess	The 'scoreboard.txt' updates correctly and shows the relevant information	None (15:21)	**Before winning** **After winning**
20	Typical	When a game is either won or lost and the 'scoreboard.txt' is not empty, it should be appended and edited to show the newly won or lost words	The 'scoreboard.txt' files updates correctly and overwrites the necessary numeric values at the top	None (16:18)	**Winning a second time** **Losing for the first time**
21	Extreme	If the player re-enters a letter or word, they should receive an error popup	An error popup successfully appears when the same letter or word is re-entered	None (17:20)	
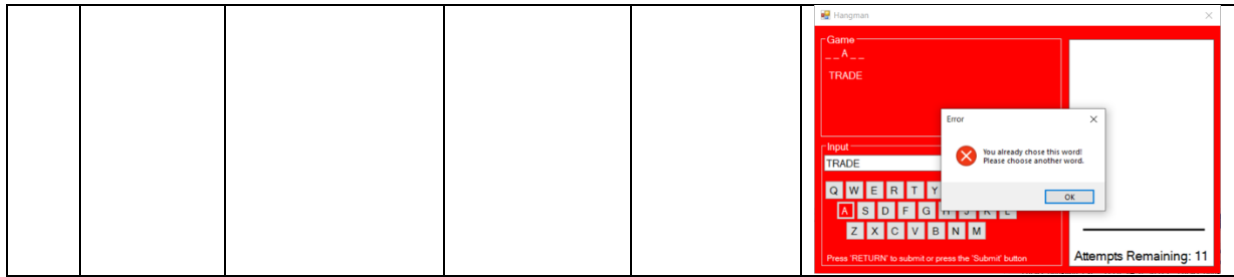
A testing video can be found at: https://www.youtube.com/watch?v=umRBDpijxOQ

Hangman Evaluation
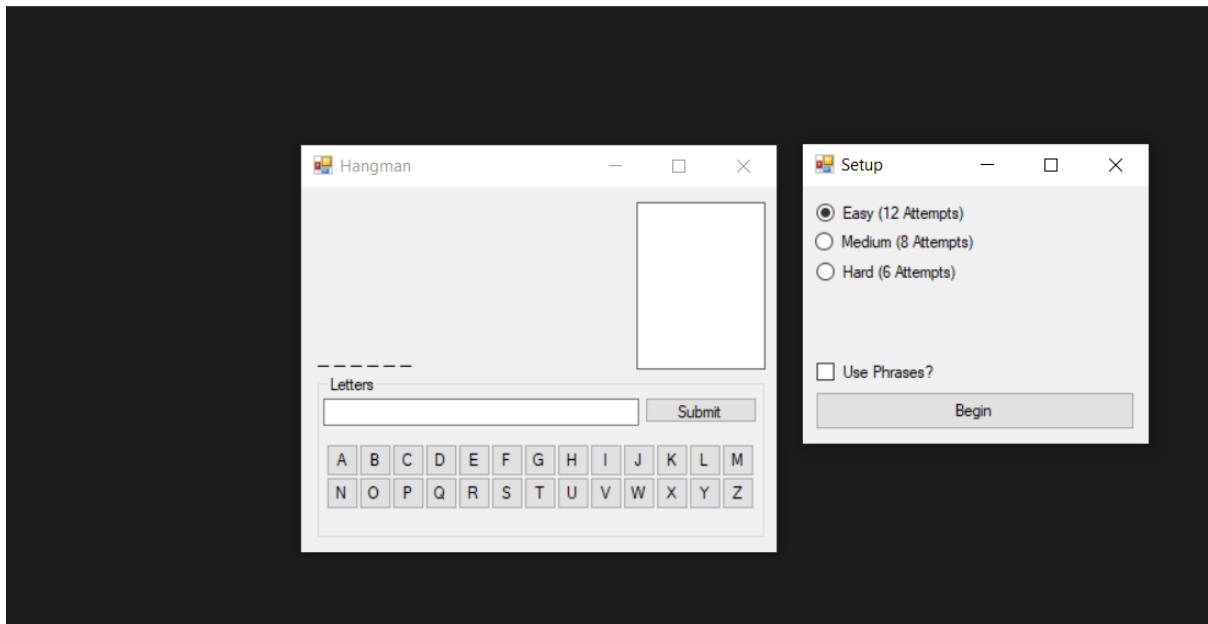
My original design

My original objectives were to have the gallows be displayed in a picture box, in order to give full control over drawing lines and other primitive shapes, to display letters-to-be-guessed as underscores in a label, and to display the already guessed letters/words above the incomplete word. My original testing plan was to test: the number of attempts depending on the difficulty selected, that erroneous inputs are not accepted and do not cause errors, and that a random word/phrase is picked each time with the gallows being drawn correctly.

I believe I have almost completely met the objectives I set out in my design document, albeit with some necessary design changes to improve the usability of the application. I have displayed the gallows in a picture box and used the Graphics class to draw lines and other primitive shapes to represent the gallows and the hanging man. I used underscores to represent letters-to-be-guessed and replaced the already guessed letters appropriately.

However, I did make a few changes to the User Interface that differ from my original plan. The previous guesses are displayed in a label, but the label is below the main word label. I made this change because if the user enters multiple whole word/phrase guesses, the label truncates and some of the guesses become obscured by the limits of the 'Game' group control. To fix this, I moved the previous guesses label and allowed for a lot of space below the label so that all guesses can be clearly seen. I also changed a number of other User Interface elements. I changed the background colours and the size of all text in the windows in order to abide by Jill's request in her interview: *"As long as it's not that boring window grey and reads clearly..."*, so that the window looks more visually pleasing and is easier for people to read.

For testing, I did stay true to my original tests, but I decided to add some more tests, in order to ensure the application is completely bug-free. I believe that my original tests and the newer tests are all relevant and have helped me to successfully debug the application.
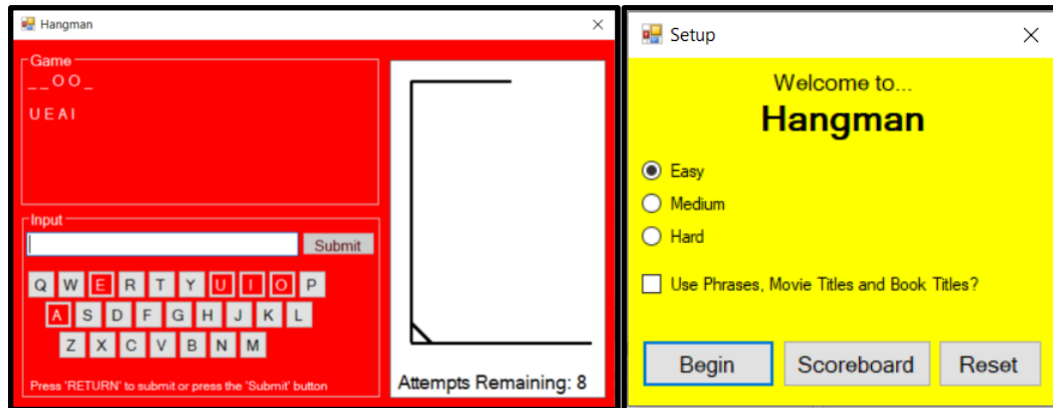
My new design

To improve my Hangman game, I feel I could have added some form of background image to all the forms, to make them more visually appealing than a solid colour. I also think that adding other buttons to the onscreen keyboard, such as backspace, return or a spacebar, might have made typing, for Jill, a lot easier than having to constantly alternate between the onscreen keyboard and the physical keyboard. At the moment, the user can simply enter the 'scoreboard.txt' file and alter it how they please. To make this more secure, I should have made a custom class for storing user's scores and then used the 'System.Runtime.Serialization.Formatters.Binary' namespace to serialize the class into a custom binary file, thus somewhat encrypting the file.

To personalise the scoreboard, I think that giving players the ability to give themselves a name would make the game more practical for sharing scores with others. To that effect, having the contents of the file be displayed in a message box simply is simply not enough. Therefore, the creation of a new form in which players can view their scores and give themselves a nickname would be necessary. Another interesting feature might be online multiplayer, where a small "room" of players can take turns to guess a word or phrase together. This and the ability to store players' scores in a database would undoubtedly make the game more interesting.

After sending the game to one of my friends and asking him to test it, he sent this feedback: *"works well, no problems, if your looking for stylistic advice id change the background colour depending on the difficulty. it gets the job done, welldone"*. I think that this is a valid suggestion since, at first glance, changing the difficulty does not appear to have any effect on the game – apart from parts of the gallows being drawn early.