**Final Project Statement of Work Template**

|  |
| --- |
| 5 |
| Rami Toma |

**Group Number:**

**Group Leader:**

|  |
| --- |
| Hussein Alanbari, Tajwar Rahman |

**Names of Group Members:**

**Summary of what your program will do (be very specific):**

The purpose of this program is to make an Hangman type of game that has two difficulties.

The base class will take the json data with loads of words and input it inside a vector. It will also make a linked list from a template class. It will contain << and == overloaded operators to output via a function which takes in an ostream& and == compares the guessed word with the word selected. The base class only outputs to the console how many times won or lost.

The inherited class derived from the base class has an overloaded output operation which will change the output by adding the percentage of attempts of guessing word with the overall win rate (times won/attempts)

Aggregation class introduced a more difficulty game, that only uses words longer than six characters, pts earned are doubled, 3 chances instead of 6.

The template class will create a linked list. It will contain a struct, delete, push, return amount of times won, return amount of times lost, output to an ostream& (dynamically allocated template class and this will usually happen when program is quitting i.e. outputting to an external file and to console)

Search function/operation that is user defined by a single character or by word, that removed all words that begin with that letter or that specific word. If char is not a letter plus if the word is not all letters, it loops back around clears the cin and asks again

BubbleSort function/operation will ask the user to sort based on ascending or descending order. This will iterate through the vector via iteration. It will be based on two char characters.

Their will be an additional input regarding if you want to continue to play or quit.

When we output when the user quits, it writes to a file. Their will be an output to the console indicating this.

**Decomposition with Assignments:**

Use the space to list the decomposition or outline of your program. Next to each section, provide the name of the student who will be working on that portion.

|  |  |
| --- | --- |
| **Decomposition** | **Responsibility** |
| Base Class, Template Class | Rami |
| Derived Class, Searching Operation | Hussein |
| Aggregation Class, Sorting Operation | Tajwar |
| Main Setup, Entering Input | Everyone |