**You. Will. Die. – Project Document**

Innes Garmany

Analysis of the Problem

Description of Problem

Describe the problem. Your description should include:

1. An outline of the problem, identifying Advanced Higher concepts and integration.
2. The scope and boundaries of the problem, and any constraints you identify.

I will develop a Software Design and Development project which will be integrated with a database.  The game will have platforms, usernames, scores, a death count, high scores, a leaderboard, and a player character - each of which will be created using object-oriented programming. It will be linked to a database to store the user’s score, their username, as well as storing the best ever scores for each user with the ability to display those scores on a leaderboard.

Gameplay

My game will be a rage platformer; a rage platformer is a 2D platforming game that is difficult, has unwieldy controls, and makes you fall back to earlier in the level when you mess up - meaning you lose progress and time. The platformer will consist of platforms above the player witch they must climb going higher and higher in the level, eventually reaching the top platform at which point they will be greeted with message congratulating them on beating the game.

The project meets the requirements of Advanced Higher Computing because:

* The game is linked to a database which will store player’s names and scores.
* It will be an object-oriented program.
* A bubble sort will be used to find the top scores so they can be displayed on the leaderboard.

Scope

The scope of my project will include:

1. A completed design, including a UML case diagram as well as pseudo-code
2. A high-fidelity prototype
3. Documented testing
4. The final program will be completed by April 2020

Boundaries

My solution will contain the following boundaries:

1. The score can never be below zero and will always be an integer
2. There will be validation on usernames
3. The program will only save the user’s top score for each level.

Requirements Specification

A requirements specification should list:

* End-user Requirements
* Functional Requirements
* Input Validation

End-user Requirements

1. The user must be able to control their character using the mouse.
2. They must also be able to enter a username with the keyboard.
3. The user should be able to turn off subtitles.

Functional Requirements

1. The program must receive and store usernames in a database.
2. It must keep track of how long the user has been on a level and must store their score and time when they complete the level.
3. The game must offer a selection of different levels.

Input Validation:

1. The user’s username can only be between 5 and 20 characters long.
2. Scores cannot be lower than zero