H446 OCR Programming Project

*Eshan Fadi*

*Upton Court Grammar School*

*Centre Number: 51427*

*Candidate Number: 8052*

[Project Name to be decided]

Table of Contents

[Project Ideas 3](#_Toc170658950)

[Chosen Idea and Justification 4](#_Toc170658951)

[Problem Identification 5](#_Toc170658952)

[Stakeholders 5](#_Toc170658953)

[Research 5](#_Toc170658954)

[Titanfall 2 5](#_Toc170658955)

[Doom (2016) and Doom Eternal 5](#_Toc170658956)

[Grapple 5](#_Toc170658957)

[Half Life (and other Source Engine games) 5](#_Toc170658958)

[Essential Features 5](#_Toc170658959)

[Movement 5](#_Toc170658960)

[Combat 6](#_Toc170658961)

[Sound Effects 6](#_Toc170658962)

[Multiple Levels 6](#_Toc170658963)

[Scoring System 6](#_Toc170658964)

[Death Screens 6](#_Toc170658965)

[Graphical User Interface (GUI) 6](#_Toc170658966)

[Limitations 6](#_Toc170658967)

[Solution Requirements 6](#_Toc170658968)

[Success Criteria 6](#_Toc170658969)

[Design 7](#_Toc170658970)

[Decomposition 7](#_Toc170658971)

[Structure Definition 7](#_Toc170658972)

[Algorithms 7](#_Toc170658973)

[Usability 7](#_Toc170658974)

[Variables 7](#_Toc170658975)

[Data Structures 7](#_Toc170658976)

[Classes 7](#_Toc170658977)

[Validation 7](#_Toc170658978)

[Iterative Tests 7](#_Toc170658979)

[Post Tests 7](#_Toc170658980)

[Development 7](#_Toc170658981)

[Evidence 7](#_Toc170658982)

[Prototyping 7](#_Toc170658983)

[Modules 8](#_Toc170658984)

[Comments 8](#_Toc170658985)

[Naming 8](#_Toc170658986)

[Validation 8](#_Toc170658987)

[Reviews 8](#_Toc170658988)

[Development Testing 8](#_Toc170658989)

[Evidence 8](#_Toc170658990)

[Fails 8](#_Toc170658991)

[Evaluative Testing 8](#_Toc170658992)

[Evidence 8](#_Toc170658993)

[Usability Tests 8](#_Toc170658994)

[Evaluation 8](#_Toc170658995)

[Success Criteria 8](#_Toc170658996)

[Usability Features 8](#_Toc170658997)

[Limitations 9](#_Toc170658998)

[Maintenance 9](#_Toc170658999)

[Bibliography 9](#_Toc170659000)

# Project Ideas

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Idea | Short Explanation | Complexity | Language | Computational Methods |
| Movement first-person-shooter game | A game where the player must simultaneously manage their momentum and there aim | Medium - ample use of math and GUI. Movement must be fine-tuned along with movement based bug fixes. Player physics, and bullet physics, enemy AI, and modular interconnected components, OOP, libraries | C#, C++, GDScript, Python, easier with engines | OOP, database, modular design, abstraction for the GUI, reusability for functions |
| Game engine | An application designed to help and aid someone in the creation of game or other app | Very - lots of math, OOP, GUI, implementing a physics engine, along with functionality for any component, must include compilers and built in libraries to assist in building the game | C#, C++ | OOP, decomposition of functions |
| Programming Language | Writing a language and run it either being compiled, or to interpreted | Easy - filled with mainly trees and if statements, and exception raising, must have built in functions, handle memory and add basic data structures | C#, C++, CPython | Selection, trees, decomposition and modular design for different parts of the compiler |

## Chosen Idea and Justification

The game engine is far too complex and time consuming for the constraint of time, while also requiring more learning than either of the other options. The programming language on the other hand, is more simple and would not allow me to easily incorporate as many computational methods. The movement first-person-shooter game is a great medium between the complexity of the two other options, while also allowing me to incorporate a great deal of creativity and interesting implementations of the course content. Analysis

## Problem Identification

An up and coming group of the gaming community is a collection of speedrunners, craving for fast movement and fluidity to beat the games faster and faster. To satiate this need, I have decide to construct a fast-paced movement first-person-shooter game, demanding both an understanding of momentum and precise aim, to hit a selection of enemies and reach the end of the level as fast as possible. The game will also have a secondary genre of a role-playing game, allowing players to progress and unlock more movement abilities and upgrade already existing ones to navigate the game at faster rates.

## Stakeholders

Amarveer Flora

Henry Masters

## Research

### Titanfall 2

Released on 28th October, 2016 by Respawn Entertainment and published by Electronic Arts, Titanfall 2 is movement shooter, where the player swaps between moving around as a human character and playing within a Titan, where the style of play completely changes.

### Doom (2016) and Doom Eternal

### Grapple

### Half Life (and other Source Engine games)

## Essential Features

### Movement

Walking

Sprinting

Jumping

Crouching (both toggle and hold)

Sliding

Wallrunning

Grappling

Double Jump

Boosting

### Combat

Fire

Zoom

Swap weapon

Quick melee

Pickup weapon

Reload

### Sound Effects

### Multiple Levels

### Scoring System

### Death Screens

### Graphical User Interface (GUI)

## Limitations

## Solution Requirements

## Success Criteria

# Design

## Decomposition

## Structure Definition

## Algorithms

## Usability

## Variables

## Data Structures

## Classes

## Validation

## Iterative Tests

## Post Tests

# Development

## Evidence

## Prototyping

## Modules

## Comments

## Naming

## Validation

## Reviews

# Development Testing

## Evidence

## Fails

# Evaluative Testing

## Evidence

## Usability Tests

# Evaluation

## Success Criteria

## Usability Features

## Limitations

## Maintenance

# Bibliography

Anon., 2024. [Online]   
Available at: https://tf2sr.github.io/TF2SR-Wiki/