Software Requirements Specification

for

<Multiplayer-FPS>

Version 1.0 approved

Prepared by <PolychronisMystakidisPanagiotis>

<22/1/2021>

Table of Contents

Revision History

1. Introduction

1.1 Purpose

This product has one goal in mind and that is to make a fun and entertaining online first person shooter game.
The this SRS document specifies the requirements of the entire project .

1.2 Document Conventions

This document uses an Entity Relation diagram.

1.3 Intended Audience and Reading Suggestions

This document is intended for project managers , developers , marketing staff and testers .

1.4 Product Scope

This software is a multiplayer FPS game online game . As an open projects its goal is to provide a free but high quality game for everyone to play, inspect the code and get new ideas from this project. Primary purpose is for the game to be fun and entertaining. The game from a marketing standpoint is intended mostly for young people which are the primary users of such software types.

1.5 References

There are no external references.

2. Overall Description

2.1 Product Perspective

The product is a Multiplayer First Person Online game so it belongs to a certain genre but it is a self-contained and brand new project and doesn't connect to any other products .

Other games that belong to this genre are CounterStrike (https://blog.counter-strike.net/), CallOfDuty (https://www.callofduty.com/home), Overwatch (https://playoverwatch.com/en-us/) and you can find those projects in the links so you can get an idea of how it should look like.

2.2 Product Functions

The functions of the product are the above:
- *Creating character model*
- *Creating a map*
- *Moving character inside the map*
- *Shooting weapons and applying damage to other characters if needed*

A simplified version of the classes can be seen in the above ER diagram:
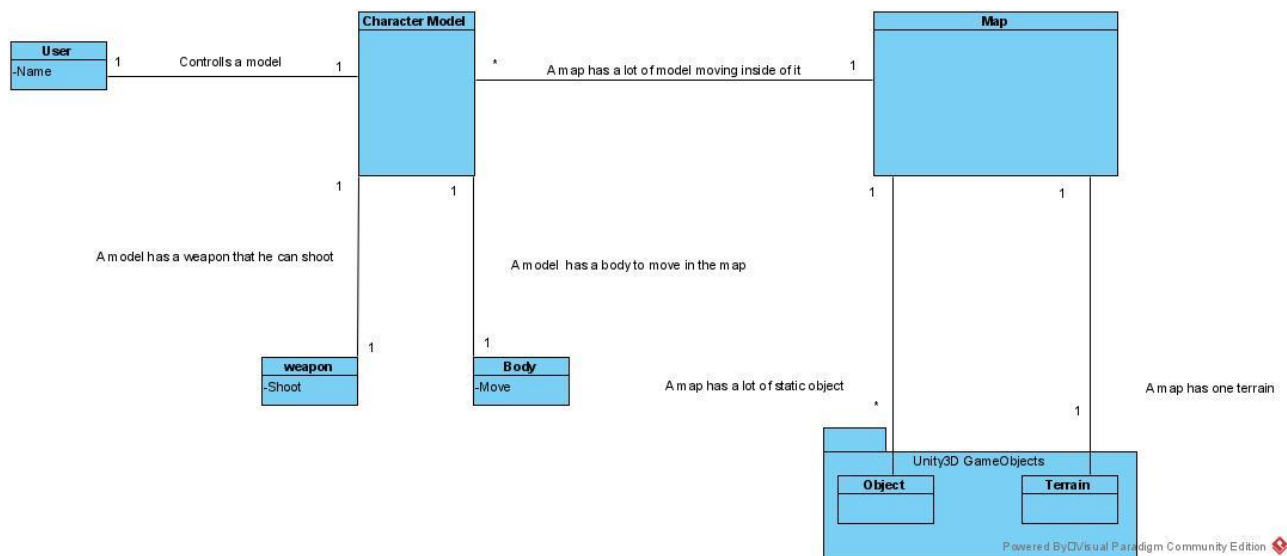
2.3   User Classes and Characteristics

Classes of users are:
- *Experienced users(played the game at least 10 times)*
- *Unexperienced users (played similar game but not this one)*
- *Totally Unexperienced Users (hasn't played this or any similar games)*
- *New user (First time opening the game)*

User must be able to perform the following functions:
- *Getting a name on the server*
- *Joining a map*
- *Moving their character on the map*
- *Shooting the weapon*

2.4 Operating Environment

The game is running in:

- *OS: Windows 7, 8, or Windows 10 64-bit.RAM: 4GB RAM.*

- *Minimum Hardware requirements :*
    - *FREE DISK SPACE: 30GB Free Space.*
    - *VIDEO GRAPHICS CARD: NVIDIA GeForce GTX 460 / ATI Radeon HD 4850 / Intel HD Graphics 4400.*
    - *Screen Resolution: 1024 x 768 minimum.*
    - *PIXEL SHADER: 4.0.*
    - *VERTEX SHADER: 4.0*

2.5 Design and Implementation Constraints

- *Every user must have a good response time from the server (maximum 100ms)*
- *The organization responsible for the maintenance is this organization*
- *The project must be made with Unity3D and use its client – server services and no external libraries that support networking*
- *There is no need for a database with the players*
- *You must use C# programming language*
- *You must use VirtualStudio2019 as the editor*

2.6 User Documentation

There should not be any users manuals and any user help must be provided inside the product.

2.7 Assumptions and Dependencies

- *This is a project that uses the Unity3D game engine*

3. External Interface Requirements

3.1 User Interfaces

The GUI in the starting screen must include:
- *A input box with the players name*
- *A list with rooms which the player can enter*
- *A button to enter the room*

The GUI while playing the game must include:
- *The players life bar*
- *A message box for communication between the players*

3.2 Hardware Interfaces

The hardware from the user incudes:
- *A mouse (controls the head and weapon of the player)*
- *A keyboard (must have ASDW buttons to move the players)*
- *A monitor*
- *A audio device (with stereo audio if possible)*
- *A microphone (not necessary but must be supported)*

3.3 Software Interfaces

This software has no connection to external software except the Unity3D game engine.

3.4 Communications Interfaces

The networking service is the client – server connection that is provided by the Unity3D game engine.

4. System Features

4.1 Connection to a Room

4.1.1   Description and Priority

The feature provides multiple rooms with a map where players can enter.

Priority: 9

4.1.2   Stimulus/Response Sequences

The system must provide the user with a UI to allow him to join a map.
Once he has selected his map a new character model must be created in that map which the player can control (as show by the diagram in 2.2).
First the player must enter his name and then he can select one of the rooms.
There should be an option for the player to create a room of his own and of course enter it. If he doesn't enter a name he will not be able to enter a room.

4.1.3   Functional Requirements

- *There must be a UI for the player to choose a room*
- *Player must be able to create a new room for others to join*
- *Player must be able to leave a room and go to the selection screen again*
- *Only the creator of the room can delete it*
- *If a player leaves the room his character model is deleted from the room*
- *The a user enters a room a character model is created which the player can control*

4.2 Move player character

4.2.1   Description and Priority

The feature provides the player with the option to move his character inside the map he has entered.

Priority: 9

4.2.2   Stimulus/Response Sequences

If the player wants to move in front he must press W.
If the player wants to move right he must press D.
If the player wants to move left he must press A.
If the player wants to move back he must press S.
If the player wants to jump he must press SPACE.

4.2.3   Functional Requirements

- *The player must be able to jump only if he is touching the terrain*
- *The player can't move right, left, front, back while he is not touching the terrain.*
- *There must terrain which the character model can move as shown in (2.2)*
- *There must be gravity affecting the player model so if he jumps he will eventually fall to the terrain again.*
- *The character model must not be able to leave the area of the map.*

4.3 Shoot

4.3.1   Description and Priority

The feature provides the player must be able to use the shoot function in his weapon to shoot and deal damage to enemy players.

Priority: 9

4.3.2   Stimulus/Response Sequences

Player is shooting with the Mouse1 button.
If the bullet hits an enemy it deals damage.
There should be a crosshair in the GUI which indicates where the bullet is going to go.

4.3.3   Functional Requirements

- *Character model must have a life point system so when the bullet hits it can deal damage.*
- *Damage is defined by the weapon.*
- *When damage is dealt to a model it should decrease its life bar by the amount of damage the weapon has.*
- *The crosshair must be in the center of the screen and the bullet must land where the crosshair is indicating.*
- *Bullets must not have any travel time and should immediately reach their destination.*

4.4 Weapon Selection

4.4.1   Description and Priority

Player should be provided with a wide choice of weapons he can use all different to each other.

Priority: 7

4.4.2   Stimulus/Response Sequences

There must be a UI which opens with B button that shows the available weapons.
From there the user can click on a button to select weapon.

4.4.3   Functional Requirements

- *The UI must be able to open and close with B button*
- *Every weapon must be associated with a unique button*
- *A character can only hold 1 weapon at the same time*
- *Every weapon must have an integer Damage which is the damage value when it hits an enemy.*
- *The weapons must belong to categories.*
- *Weapon categories must be more than 3.*

4.5 In Game Chat Room

4.4.1   Description and Priority

A chat room where the players inside the map can communicate

Priority: 5

4.4.2   Stimulus/Response Sequences

There must be a UI which in which the previous messages are visible.
There must be an input box where the user can type what he want to send as a message.

4.4.3   Functional Requirements

- *The owner of every message must be visible to all other users on the map.*
- *The messages must take a small portion of the screen so it won't block any important information from the players while playing the game.*
- *The messages must be able to hide from the screen and appear again.*
- *When writing a message the player won't be able to take another action inside the game (for example pressing W won't move the player but it will only write W on the message input box)*

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- *The game shall be as 'real time ' as possible so there shouldn't be any network lag issues (ping must be <=100ms)*
- *The game must run at 60 fps refresh rate minimum.*

5.2 Safety Requirements

- *The program does not have any important information that can be leaked since there is no personalization for the players.*
- *The information of the state of the system must be protected so other software's must not use this information to help the players.*

5.3 Security Requirements

- *There is no need for authentication.*
- *There must be no use of external software accessing data of the game to help the players (help in shooting mechanics or decision making) .*

5.4 Software Quality Attributes

The product must have:
- *Usability (it should be easy for new user to start learning the game)*
- *Reliability (no big network issues of FPS drops while playing the game)*
- *Maintainability (due to the nature of the project new feature should be easy to be added)*

5.5 Business Rules

- *As an open project anyone can propose new ideas and import code*
- *But only the manager has authentication to choose which changes will end up in the end product*

6. Other Requirements

Glossary

- *Multiplayer game : a game with multiple players playing at the same time*
- *Online game : a game that is played from multiple computer (usually connected to an online server)*
- *First Person game : a game in which the camera is set in front of the character model which gives the impression that the players is inside the game*
- *Shooter game : a game in which you have a weapon you can use to shoot other player models and gain kills*
- *Map : the environment where you can move your character model*