**SOFTWARE REQUIREMENT SPECIFICATION AND SYSTEM MODELLING**

**“*OOZRA*”**

**Group Members:**

**Syeda Saman Afaq (SE-012)**

**Sheraz Mirza (SE-025)**

**Khurram Meraj (SE-045)**

**Nayel Hashmi (SE-049)**

**TABLE OF CONTENTS**

1. Introduction

    1.1 Purpose

    1.2 Document Conventions

    1.3 Intended Audience and Reading Suggestions

    1.4 Project Scope

    1.5 References

2. Overall Description

    2.1 Product Perspective

    2.2 Product Features

    2.3 User Classes and Characteristics

    2.4 Operating Environment

    2.5 Design and Implementation Constraints

    2.6 Assumptions and Dependencies

3. Functional Requirements

4. Nonfunctional Requirements

    4.1 Performance Requirements

    4.2 Safety Requirements

    4.3 Security Requirements

    4.4 Software Quality Attributes

5. Interface Requirements

    5.1 User Interfaces

    5.2 Hardware Interfaces

    5.3 Software Interfaces

    5.4 Communications Interfaces

6. System Model Diagrams

6.1 Use-Case Model

6.2 Context Model

6.3 State Diagram

6.4 UML Diagram

1. **Introduction:**

**1.1 Purpose**

This Software Requirement Specification document is written with the intention to give a complete and thorough overview of the game *Oozra*.These requirements relate to the functionality, constraints, performance, attributes, game mechanics and user interface.

The game consists of the character *Furry* who is operated by the player so that it avoids obstacles in order to pass the three levels.

**1.2 Document Conventions**

In order to make the SRS of *Oozra* effective and readable for the game developers and concerned audience, the following document conventions have been used;

i. Font type *Calibri* in size *12*.

ii. Each new section’s heading is bolded and underlined along with bullet points *1*, *2* and so on.

iii. The subsections headings are bolded along with bullet points *1.1*, *1.2* and so on.

iv. Important words in text are emphasized by using *italics*.

**1.3 Intended Audience and Reading Suggestions**

This document is written for the end-users, testers, and developers of the game *Oozra*. Although the document has to be read start-to-finish in order to be comprehended fully, the document has been divided into the following sections for the convenience of the reader;

1. Introduction
2. Overall description
3. Functional requirements
4. Non functional requirements
5. Interface requirements

**1.4 Project Scope**

This game will provide entertainment to its user in their idle time. It is also useful to divert one’s attention and relieve stress or to regain focus. Since the targeted audience is diverse i.e. people of all ages, hence, the game is of great scope.

**1.5 References**

Guidance for making a 2D game on Unity Engine is to be taken from Brackeys; [**http://brackeys.com/**](http://brackeys.com/)

1. **Overall Description:**

**2.1 Product Perspective**

The game is not a sequel of any other product; it is a new idea with the motive of 2D game entertainment. The game essentially provides a light interface and graphics model to the user, easy to play and playable in short breaks in between work. This unity based game is developed to provide short interval entertainment for gamers who want to divert their attention or relieve stress. The game model focuses on avoiding obstacles and reaching towards the point where the level ends and a new level can be played. The game also provides the user a pause option which lacks in many 2D games and is an inconvenient factor for the user.

**2.2 Product Features**

The unity based game is a short interval entertainment basis and isn’t expected to have fancy features that exist in heavy 3D games. It is a 2D platform that requires the protagonist *Furry* to jump over obstacles and enemies. It also includes a grappling hook mechanism, by pressing the “E” key, Furry throws a hook at the vantage points and then the grappling hook pulls the character towards the point where the hook was targeted at by using the mouse position as a reference. The game allows the player to switch between levels once you have completed the whole game. All the levels are unlocked once you have defeated the monster in the final level (last stage). The game offers the user to change controls via an option at the start screen which is also a key feature and lacks in basic 2D games. Default controls are also set when the game is downloaded and a control menu in the settings panel tells the user the default controls as well as double clicking the control allows the user to change the control to what they please. The game also includes background sound, themed music and game sounds that enhance the overall experience.

**2.3 User Classes and Characteristics**

Our controls and user participation is easy and can be understood completely by playing just a couple of games. The distinguishing factor between a good and an average player will be the time and focus they provide to the game. Jumping and dodging is simple game play but can be tricky as levels advance. Hence a good player will require good foresight which will be achieved by playing more frequently.

**2.4 Operating Environment**

The game is a simple unity based project, and is not a consuming software. It will run perfectly on a simple Windows or Mac based system. A very basic system is enough to run the game properly and no safety measures are required.

**2.5 Design and Implementation Constraints**

i. Due to a bug in the menu system, the text does not line up with the input.

ii. The second level had one particular constraint where there was a bug at one particular tile, hence, the character, *Furry* would jump higher than it was designed to, due to some unknown anomaly related with the box collider at that particular tile.

**2.6 Assumptions and Dependencies**

The game makes an assumption that you know how to navigate the title screen menu using the up and down arrow keys found on your keyboard, and using clicks and enter key to select the specific option. The game is dependent on Windows OS to run and I/O devices keyboard and monitor.

1. **Functional Requirements:**

i. The screen must load and appear every time the game is launched.

ii. If the player quits during any stage of the game, they must be returned to the title screen.

iii. If the player pauses by pressing *P* during any stage of the game, they must be redirected to the user interface.

iv. If the player presses exit at any stage of the game, the game will end and the will return the player to the PC’s regular interface.

v. If the player completes the game, the game will end and return the player to the title screen.

1. **Non-functional Requirements:**

**4.1. Performance Requirements:**

Modern computer hardware should be able to run this game with relative ease since the game uses image based sprites which reduces burden on the CPU since the game is meant to be as accessible to as many people possible, with no major hardware limitations. Although, the game does require a minimum Windows 7 to be able to run properly without any compatibility issues.

**4.2. Safety Requirements**

*Oozra* will not interfere with or damage any other application/program installed on your computer. This game has gone through a series of tests to ensure it does not cause any overheating of your laptop/PC. No memory leaks were observed.

**4.3. Security Requirements**

*Oozra* will not ask any personal information from you without you nor will it collect any data or connect to the internet. It is strictly an offline single-player experience. No player authentication is required to run the game hence anyone can access the game if they have access to your computer.

**4.4. Software Quality Attributes**

For the most fluid experience possible, the response time to a user input within the game has been shaved down to milliseconds so that for example when the player presses the “JUMP” key, the reaction time for *Furry* is almost instantaneous. Another attribute about the game to be noted is the AI programmed within the enemies found in the game; as colliding with them triggers an attack animation comprising of multiple PNGs changing frame by frame to create an animation. If the player comes in contact with the enemy’s weapons the “GAME OVER” layout is triggered prompting the player to restart the level. In case the player wants to pause the game mid level they can press the “P” key which pauses the game and gives them the option to return to the title screen. The game will also slowly introduce mechanics such as the grappling hook mechanic further into the game which are necessary if the player wants to proceed further into the game. The game is all about ease of use and intuitiveness as to not overwhelm the player when they start playing, encouraging them to experiment with the mechanics and a built in editor to create their very own community levels if they desire to.

1. **Interface Requirements:**

**5.1. User Interface**

* 1. Main Menu Screen: The player will see this screen every time upon opening the game. Through the main menu, player can choose to start a new game or exit the game. When this screen loads the user can see two options “Play” and “Exit” and “Instructions”. Music at low volume plays in the background when the user is on this screen. The main menu also has an option of instructions which provide necessary guide to play the game.

* 1. Pause Screen: The player is able to pause anytime during game-play and this screen fulfills that requirement. The pause menu also allows the user to navigate between game-play screen and title screen. This screen is displayed when the user presses “P” during the game-play, buttons on this screen include “Resume”, “Main Menu”, “Exit”.
  2. Last Screen: When the player completes all the levels a credit screen appears which displays a message “Thanks for playing.” At the end of this screen a button of “Main Menu” is given, which again leads the player to Main Menu.

**5.2. Hardware Interface**

*Oozra* is PC gaming application designed for all newer versions of PCs. The game is developed for Windows 10 and all subsequent release. The Windows OS is graphically adaptable with a 2-dimensional graphics library, scaling and other necessary requirements for 2d graphics.

**5.3. Software Interface**

*Oozra* is developed under Windows OS system by the use of two tools

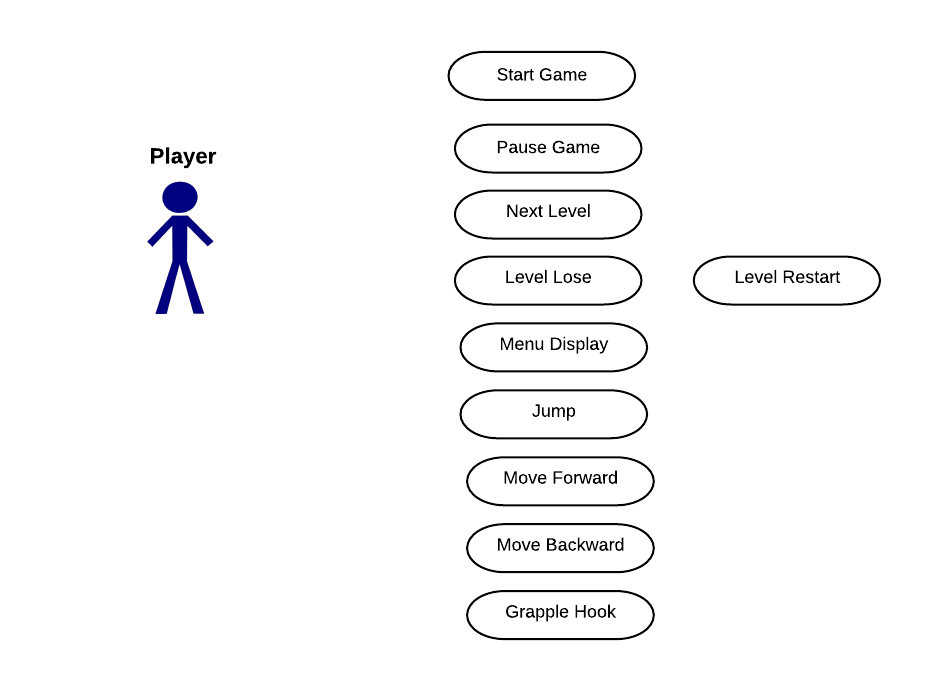
* Unity Engine
* Microsoft Visual Studio 2017

**5.3. Communication Interface**

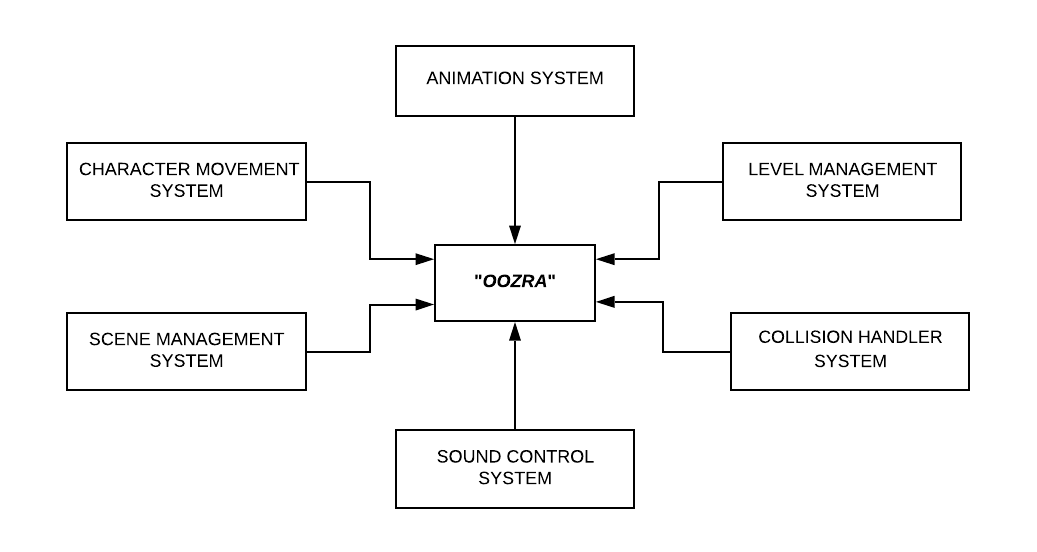
The communication channel between the user and the game character is an easy interface. The character moves left and right upon pressing the left/right keys on keyboard. To jump, the user has to press the spacebar. The output of these keys is displayed on the monitor screen. A new feature awaits for the user as the character reaches 2nd level, this is the feature of grappling hook for which player will use mouse to point at the position where he wants to hook and then press “E” key to throw the hook at that position after which the player can hang on that hook.

1. **System Model Diagrams:**

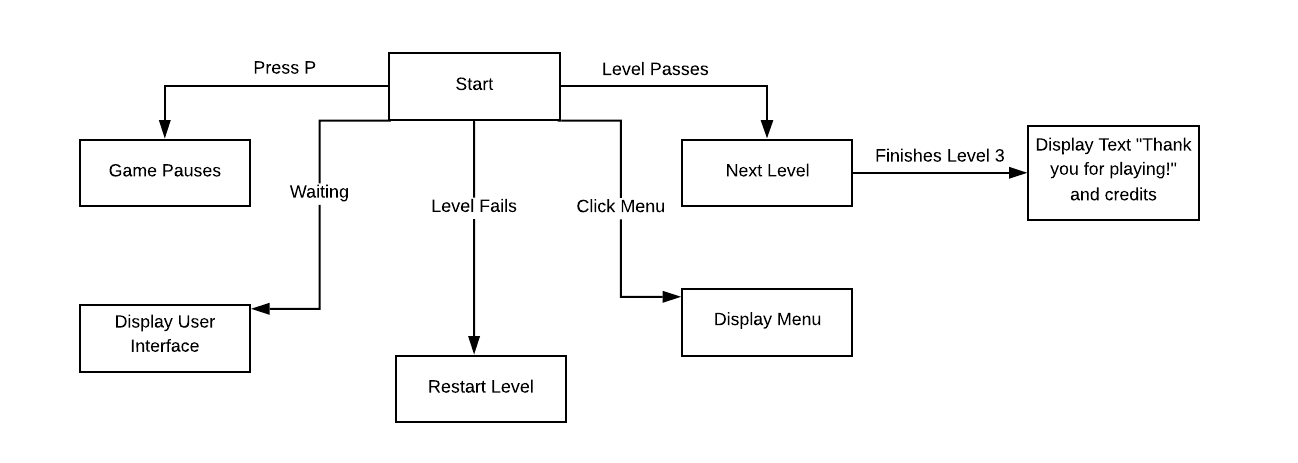
**6.1 Use-Case Model**



**6.2 Context Model**

****

**6.3 State Diagram**



**6.4 UML Diagram**

