

Just Okay RPG

23.06.2023

**─**

Gavin Lampe

# 

# Changelog

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Changes** |
| 1.0.0 | 15.06.2023 | Initial Setup |
| 2.0.0 | 23.06.2023 | Final Version |
|  |  |  |
|  |  |  |

# Contents

[Changelog 2](#_Toc137112658)

[Contents 3](#_Toc137112659)

[Introduction 4](#_Toc137112660)

[Game System Brief 4](#_Toc137112661)

[Rationale 4](#_Toc137112662)

[Background 4](#_Toc137112663)

[Terminology 4](#_Toc137112664)

[Non-Goals 4](#_Toc137112665)

[Proposed Design 4](#_Toc137112666)

[Engine and Tools 4](#_Toc137112667)

[Legal 4](#_Toc137112668)

[Target Platform 5](#_Toc137112669)

[Software and Hardware Requirements 5](#_Toc137112670)

[Hardware Minimum and Maximum Requirements 5](#_Toc137112671)

[Controls 6](#_Toc137112672)

[Mappings 6](#_Toc137112673)

[Style 6](#_Toc137112674)

[Colour Pallet 6](#_Toc137112675)

[Moodboards 6](#_Toc137112676)

[User Interface Design 6](#_Toc137112677)

[Debugging 6](#_Toc137112678)

[Testing Report 6](#_Toc137112679)

[Analysis and conclusions 7](#_Toc137112680)

[Evaluation 7](#_Toc137112681)

[Self-Reflection 7](#_Toc137112682)

[Sign Off 8](#_Toc137112683)

[Software approval 8](#_Toc137112684)

# Introduction

## Game System Brief

As part of the Game System class we wrote a simple RPG game. Unfortunately we do not have a strong UI or a load / save system. This document outlines the design and plan for main menu and pause menu options and navigation, options menu, and saving and loading both settings and game data such as the player’s position.

## Rationale

The goal is to make UI that includes game settings and a load / save system for both settings and game data for an RPG game.

## Background

RPGs, Role-Playing Games, in the context of video games are long-form adventure games in which the player embarks on a journey usually with a strong narrative and multiple adventures that can be played out in almost any order as the player roams the land in which the game is set. While an RPG can be set in any time period or universe, there are certain tropes and stereotypes in player expectation, UI design, and controller and keybinding choices, among others.

## Terminology

**RPG** – Role-Playing Game, traditionally played around the table using books and imagination, it has become the generic term for almost any video adventure game due to table-top RPGs being the dominant method to play out fantasy adventures before video game RPGs became mainstream.

**Canvas Scaler** – A Unity component that allows UI elements to be positioned and aligned for different screen resolutions and orientations.

**IDE** – Integrated Development Environment, an application with an assortment of tools for programming and debugging software.

## Non-Goals

Due to the tight time-line the code is not as optimised as it should be and a non-goal would be to refactor it to allow for easier expansion and re-use.

Another non-goal would be to use the Canvas Scaler component to set the UI up to work with multiple devices and resolutions without problems as currently it is designed only for 16:9 screens.

## Proposed Design

The setup for the project will be multiple scene files for the main menu and the game itself with static singleton classes to maintain data between scene changes.

The code for the project will utilise:

- A Startup script behind an “Any Key” screen that initialises settings and game data

- A Menu Handler class for menu navigation

- Static singleton classes for storing Game Data and Settings Data

- A Data Manager class for handling the stored data and a File Handler class for file I/O

- An Options Menu class and a Keybind class will handle settings and keybinding

- A Loading Level class will handle transitions between scene files.

## Engine and Tools

The project will be developed using the Unity game engine for the core game development, Visual Studio and JetBrains Rider IDEs for coding, and Clip Studio Paint for image editing of UI elements

## Legal

The game is developed using the Unity game engine under Unity’s Personal Licence. This allows for commercial release of the game without cost as long as the revenue is under $100,000 per year and the Unity logo is shown at the start. As this game is not going to be release commercially this is not a problem.

All art assets from the game are either free for public use without attribution or created exclusively for this project.

Clip Studio Paint has a standard licence, and JetBrains Rider and Visual Studio are both used under their respective student licences.

## Target Platform

The game is aimed at the Windows platform as it is the most ubiquitous and most available for testing and development purposes.

## Software and Hardware Requirements

**Software Used:**

|  |  |  |
| --- | --- | --- |
| **Software/Hardware** | **Version** | **Pricing** |
| **Unity Game Engine** | 2021.3.23f | Free under licence |
| **Microsoft Visual Studio 2022** | 2022 | Free under student licence |
| **JetBrains Rider** | 2022.3.2 | Free under student licence |
| **Clip Studio Paint Pro** | 1.9.4 | US$50 |

## Hardware Minimum and Maximum Requirements

**Targeted Minimum Hardware:**

|  |  |
| --- | --- |
| **Targeted Hardware** | |
| **Processor:** | Single Core 1.2Ghz |
| **Memory:** | 1GB RAM |
| **Graphics:** | 512MB |
| **DirectX:** | DirectX 9 |
| **Storage:** | 250MB |
| **Operating System:** | Windows Vista |

**Targeted Maximum Hardware:**

|  |  |
| --- | --- |
| **Targeted Hardware** | |
| **Processor:** | Dual Core 2.4Ghz |
| **Memory:** | 2GB RAM |
| **Graphics:** | 1GB |
| **DirectX:** | DirectX 11 |
| **Storage:** | 500MB |
| **Operating System:** | Windows 10 |

# Controls

/ Brief description of how the controls will work (are they mappable or are they static?) /

### Mappings

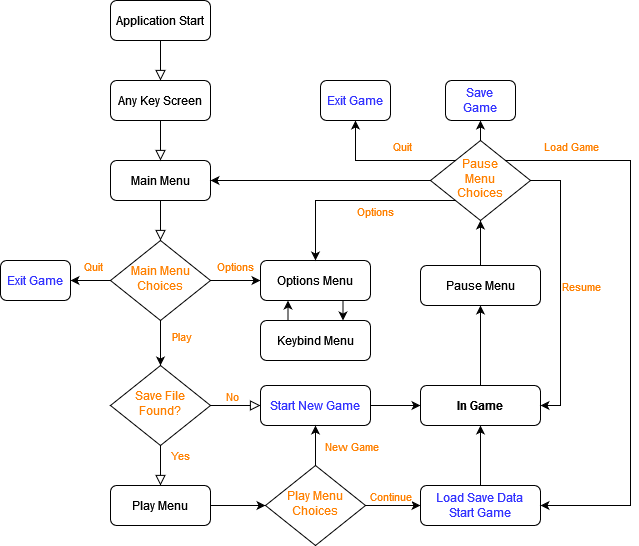
|  |  |  |  |
| --- | --- | --- | --- |
| **Control** | **Function** | **Device** | **Mappable** |
| Movement | Move the player around the scene | Keyboard | Yes |
| Actions | Jump, interact, access inventory | Keyboard | Yes |
| Rotate | Look up and down and rotate the player | Mouse | No (Static) |
|  |  |  |  |
|  |  |  |  |

# Style

## Colour Pallet

## Moodboard

## User Interface Design



# Debugging

## Testing Report

| Test ID | Error | Screenshots of testing/ relevant comments | How did you fix it |
| --- | --- | --- | --- |
| 1 | Player position reverts from saved position to original transform position on level start. |  | Pass  Character Controller was resetting position. Found a setting to auto-update transforms. |
| 2 | Vector3 data type could not be serialised despite being listed as a serialisable type in the Unity manual |  | Pass  It’s not serialisable, so I separated it into 3 floats and saved those. |
| 3 | Jump when changed to a different key only triggers jump every 10-20 key presses |  | Fail  Unable to get the jump button to work consistently after successful keybinding. Disabled jump key change as shown in screenshot |
| 4 | When changing resolutions screen would go to 4:3 even though only 16:9 options available. Also game image would go blurry and not resolve even when changing resolution back. | Did not get a screenshot because I was in a VC for 4 hours with another student trying to debug this together.  Error can be replicated by rolling back to a previous github commit. | Pass  After many hours, instead of just forcing 16:9 resolution choices in the options menu. I also forced 16:9 aspect ratio in the project settings. This fixed the problem. |

# Evaluation

## Self-Reflection

Like the other assessments, I started strong making some concept art for the UI and choosing a colour palette I liked, but then because so many assessments were being done at the same time, I lost track of where I was and what I needed to do. A rookie mistake! As someone who has done project management, I’m ashamed I let things spiral the way I did. I’m happy with my load/save system, and I’m particularly proud of my code that checks available screen resolutions and only shows those that match 16:9.

I was very annoyed with a few Unity quirks that literally took multiple days to fix two issues.

# Sign Off

## Software approval

Software is approved for installation by:

Name

[Name of appropriate person approving the software]

Role

[Click/tap to select role]

Signature

Date

[Click/tap to select date]