

Faculty of Applied Sciences

Bachelor of Science in Computing

**COMP490 Final Year Project**

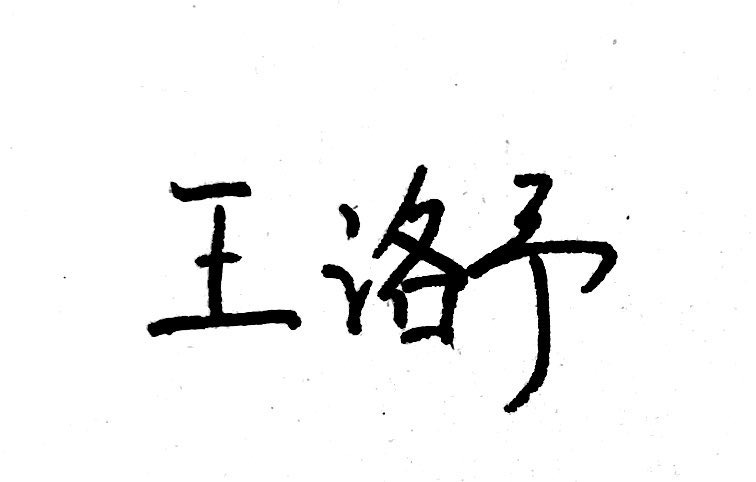
**Progress Report**

Academic Year 2022/23

|  |  |
| --- | --- |
| Indie Game Design and Development | |
|  |  |
| Project number: | Your project number |
| Student ID: | P1908404 |
| Student Name: | Vivia Wang |
|  |  |
| Supervisor: | Liam Lei |
| Assessor: | Your assessor |
|  |  |
| Submission Date: | Submission date |

Declaration of Originality

I, Wang Luoyu, declare that this report and the work reported herein was composed by and originated entirely from me. This report has not been submitted in any form for another degree or diploma at any university or other institute of tertiary education. Information derived from the published and unpublished work of others has been acknowledged in the text and a list of references is given in the bibliography.



2022/11/04

Abstract

This template file provides the Word styles for writing the Final Year Project.

Text highlight in green are instruction or hints. Text highlight in gray are sample text to demonstrate formatting. The following paragraph is an example.

Sample text sample text Sample text sample text Sample text sample text Sample text sample text. Sample text sample text Sample text sample text, Sample text sample text Sample text sample text.

There are also some placeholder highlight in red. Change them to your own information, e.g. your name and project title.

In any submitted report, you must delete or replace all the colour text.

Table of Contents

[1 Introduction 7](#_Toc16205)

[1.1 Objectives 8](#_Toc14694)

[1.2 Risk Assessment 9](#_Toc8804)

[1.3 Summary 10](#_Toc13817)

[2 Background and Related Work 11](#_Toc21048)

[2.1 Background 11](#_Toc28788)

[2.2 Related Work 14](#_Toc5384)

[2.2.1 Comparison of Different Game Engines 15](#_Toc24805)

[2.2.2 Another Subtopic in Domain 2 16](#_Toc27181)

[3 Completed Work 17](#_Toc31398)

[3.1 Project Outcome 17](#_Toc11132)

[3.1.1 Player Movement 17](#_Toc31913)

[3.1.2 Close-range Weapons 18](#_Toc32449)

[3.1.3 Projectile Weapons 19](#_Toc30094)

[3.1.4 Health Point (HP) System 19](#_Toc16412)

[3.1.5 Experience and Level Up 20](#_Toc8450)

[3.1.6 Upgrading 21](#_Toc5194)

[3.1.7 Passive Items 21](#_Toc12662)

[3.1.8 Enemy 21](#_Toc2584)

[3.1.9 Spawn System 21](#_Toc28721)

[3.1.10 Damage Popups 21](#_Toc11414)

[3.1.11 Music 21](#_Toc12103)

[3.1.12 Main Menu, Pause and Losing Game 21](#_Toc13256)

[3.2 Second Topic 21](#_Toc9214)

[4 On-going and Future Work 22](#_Toc12253)

[4.1 First Topic 22](#_Toc13737)

[4.2 Second Topic 22](#_Toc11071)

[5 Conclusion 23](#_Toc27645)

[References 24](#_Toc25075)

Table of Figures

[Figure 1: Probability impact matrix before proposed solution 1](#_Toc454957831)

List of Tables

[Table 1: Table of prioritized risk 1](#_Toc449539554)

# Introduction

Introduction must include the following:

* Related works

Do you play video games? What is your favourite game? Video games have been very popular with teenagers since their occurred. Today, with video games’ development, they involve more and more concepts and aspects about our life, and their themes have gradually become more and more profound and different from the initial "just for fun". Like movies and plays, many people are now calling games "the ninth art." Depends on different types of developers, now there are two main types of games, which are indie games and mainstream game developed by large companies.

However, for adults, as the pace of society accelerates, they have to devote more time to their work or study, and the time for them to play video games is often fragmented. This makes it difficult for them to play games which require time to practice their gaming skills, or because the gaps are too long, they tend to forget the stories of games with large-scale continuous plots. For these reasons, when many adults want to play games in their free time, they have to face these frustrating facts, so it's hard for them to enjoy gaming in this situation.

Therefore, that is why it is necessary to design some casual mini-games for adults. They often do not have obscure plots, do not require too much learning time cost, and allow players to pass a level within a few minutes to ten minutes, so players can relax and enjoy a sense of accomplishment quickly. Also, since these types of games tend to be played in fragmented times, they are best played on mobile platforms as well, such as mobile phones or game consoles.

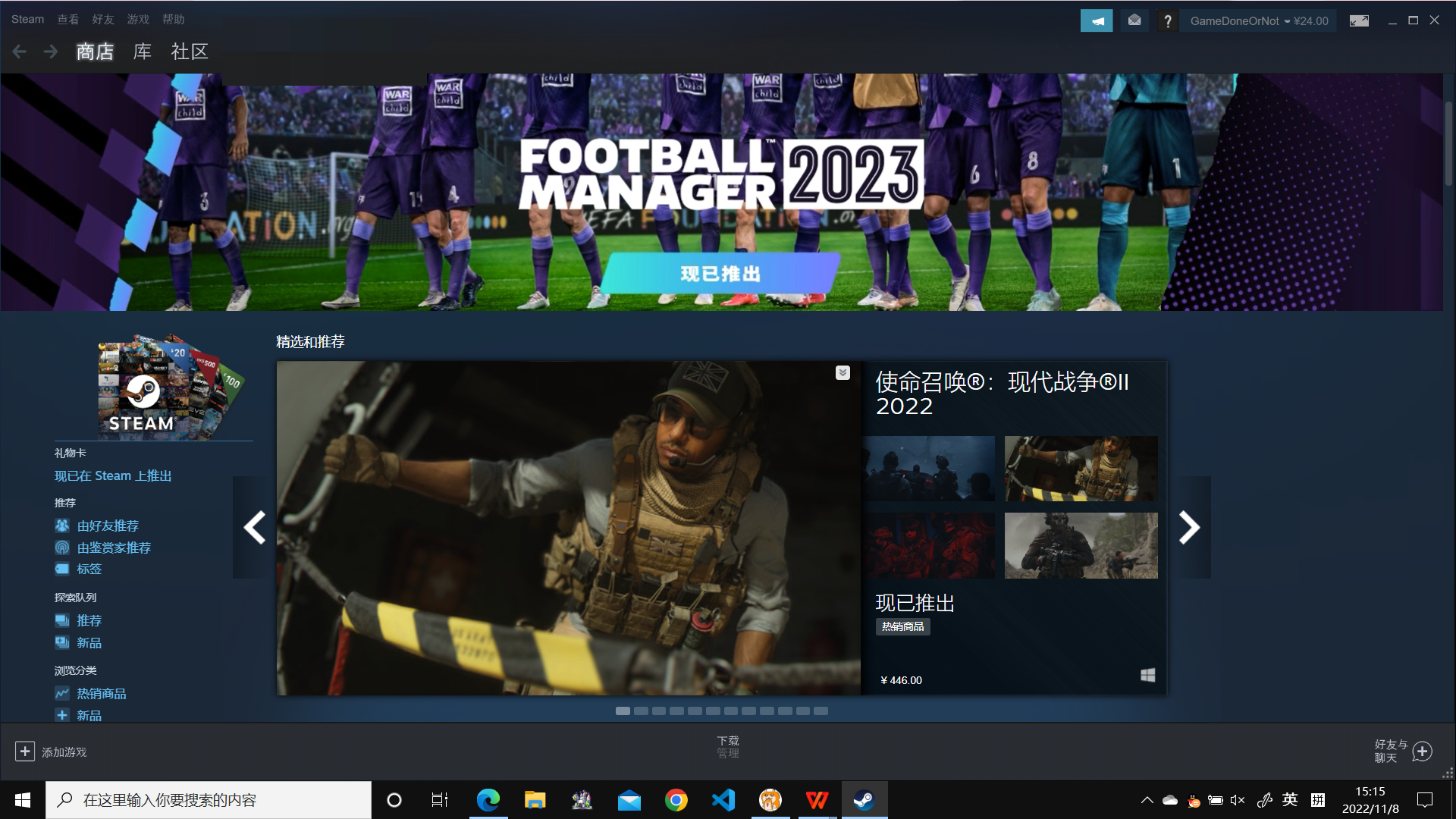


Figure 1.1 Steam——Developed by Valve company, is one of the most popular game platform in the world.

## Objectives

The main objective of the project is to develop an indie lightweight casual indie 2D game with multiple levels, which allows players to enjoy the sense of achievement brought by completing levels while attack, dodge and solve enemies, discover different maps and pick up buff items.

SMART objective

This project aims to achieve:

* Design the whole structure of the game
* Create the video and sound effects that are used in the game
* Develop the game interface
* Develop the different game level
* Develop the game interaction system
* Write the scripts which implement specific game functionality

After achieving these, this project also needs to be added some advanced features. Such as running on multi-platforms, or different difficulty levels to make it more interesting and easy to play for everybody.

## Risk Assessment

Table 1: Table of prioritized risk

|  |  |
| --- | --- |
| Priority | Risk Identifier and Description |
| 1 | Risk 1: Games made with Unity have a lower threshold for cracking, and users are more likely to develop plug-ins to cheat. |
| 2 | Risk 2: If the developer changes the Unity versions during development, the project may not continue to be developed normally due to incompatibility between different versions. |
| 3 | Risk 3: If the player use an old device to play, the CPU/GPU performance may not allow the game to play normally. |
| 4 | Risk 4: When the game is played on devices in different sizes, the window of the may not be displayed normally due to the different aspect ratios of the screens, or the images may become blur. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Probability** | High | Risk 4 |  |  |
| Medium |  | Risk 3 |  |
| Low |  | Risk 1 | Risk 2 |
|  |  | Low | Medium | High |
|  |  | **Impact** | | |

Figure 1: Probability impact matrix before proposed solution

## Summary

The summary should be finished like this: This report is organized as follows: Chapter 2 introduces the background of our work. Chapter 3 presents our design approach. Chapter 4 shows the implementation details….

# Background and Related Work

In this chapter, you provide background information for readers to help them understand your project. There may be more than one sections on background domain.

This chapter also provides detail about related work. In chapter 1, you should have mentioned some related works and explain how your project is related to them when you discuss relevancy. If you want to provide detail of related works, include them in this chapter.

Call this chapter “Background” if all related works have been described in Chapter 1.

This chapter contains two sections, providing the description of the background and related work of our project. The prototype for the project’s kind of game’s introduction, and 【】are in the *background* section, and the using developing techniques are discussed in the *related work* section.

## Background

This project aims to be a lightweight indie 2D game. Generally, ‘Indie game’ symbolizes originality and forward-thinking, especially in music and design. For the developers, An indie game’s developer is any business, developer or designer that is not associated with a large corporation, especially a global one[1]. Due to this different quality from games made by large companies, indie games have always been loved and recommended by certain players.

In recent years, more and more people don't want to be constrained by all the restrictions that the big game companies place on creating titles for profit, and want to make their own games. Although they may not necessarily have the same beautiful modelling and scenery as the big game companies’ work, indie games still have a loyal following due to their creative freedom and the variety of types and gameplay. In addition, many indie game creators do not consider developing games as work, but as a hobby of sorts, in their spare time. This allows them to remain inspired and passionate about their work.

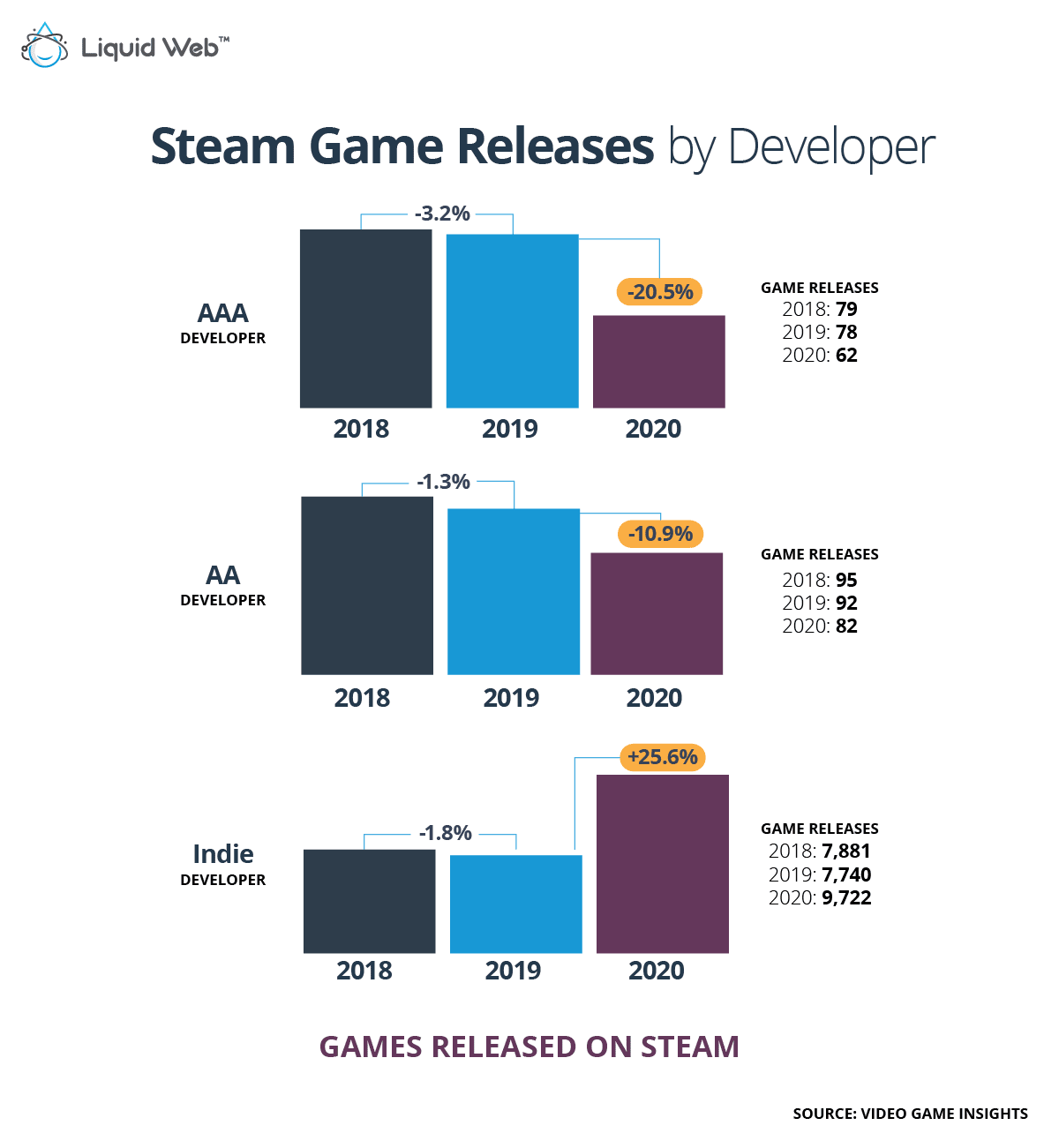


Figure 2.1.1 Indie developers can escape the adverse effects caused by the pandemic[2].

As a prototype for the project, *Vampire Survivors* is a time survival game with minimalistic gameplay and roguelite elements[3]. In this type of game, almost all attacks are performed automatically by the system after a specific time interval. The player does not need to do anything to trigger an attack. In fact, there are only two actions that the player needs to perform in a game of limited duration: to navigate through the thousands of monsters moving towards the player to avoid being hit, and to decide which weapons to increase or upgrade when picking up the drops used to increase experience and upgrade to become more powerful in order to deal with more powerful monsters.



Figure 2.1.2 A screenshot for Vampire Survivors.

The gameplay of this type of game can be summarised as follows:

* Player can move freely in a very large game map.
* There will be a constant flow of enemies moving in the direction of the player. Once touching an enemy, the player's character will lose HP.
* Players start the game with at least one weapon, which has a fixed attack frequency, and they need to control the weapon by moving their character so that it can hit the enemy.
* After kill an enemy, the enemy drops pick-up items. Players can pick them up by approaching them. Picking up such drops will increase the player's experience value and when it reaches a certain value, the player will automatically upgrade, which can make the player stronger.
* As players move around the map, they will find treasure chests. A attack of any type will open the treasure chest. The treasure chests contain items that help the player, such as restoring HP or gaining shields.
* A timer will be displayed in the game screen to allow the player to see how long a game has been played. When the timer reaches the required length of time and the player is still alive, the game is considered passed.

## Related Work

On the technical aspect, this project will be built with the Unity game engine. Unity is a cross-platform game engine initially released by Unity Technologies, in 2005, using C# to do the script. The focus of Unity lies in the development of both 2D and 3D games and interactive content[4]. In addition, Unity has very convenient packaging built-in exporting tools, which can easily deploy the completed game project on a variety of operating systems. The games made with the Unity engine make up for 50% of all mobile games. Every day, there are 15,000 new projects made within the platform. Unity engine has become a development environment for such extremely popular games as *Pokemon Go*, *Rust*, *Pillars Of Eternity*, and *Escape from Tarkov*[5].

Unity uses C# for users to write scripts. As a statically typed language, C# is easy to read and understand, making it easier to find errors in code and also understand the code. Another advantage for C# is that it is an Object-Oriented Programming language. This makes it highly efficient, flexible, scalable, and easy to maintain[6].

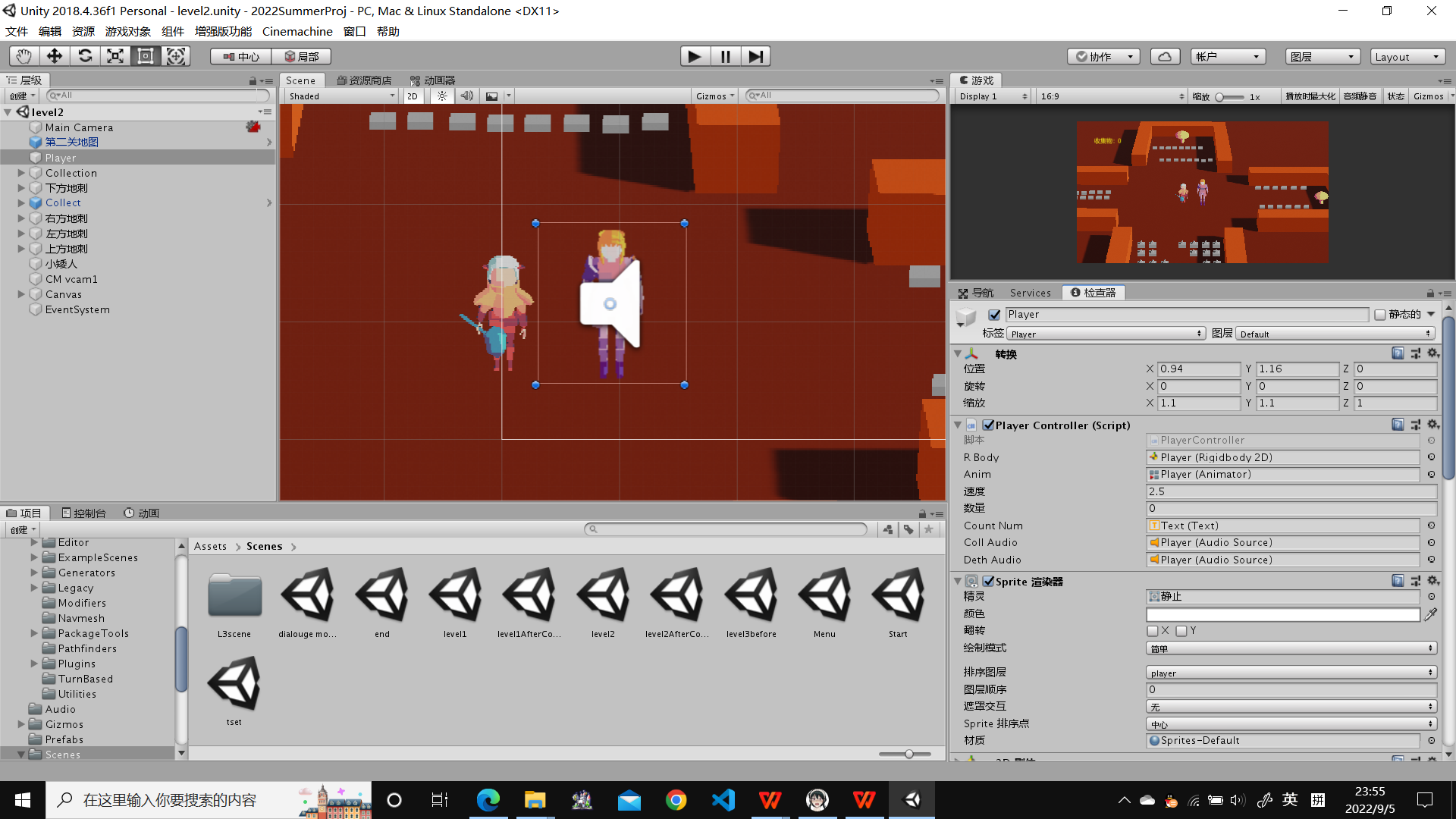


Figure 2.2.1 The development interface of Unity

【表】

### **Comparison of Different Game Engines**

Also, there are other different engines for game developing. We will do the comparison in the next paragraphs.

Such as Unreal, which is a famous and handy engine in development, too. But eventually, the Unity engine is chosen. Why? Firstly, Unity was built with mobile apps in mind, so development for these devices is very streamlined. On the other hand, Unreal is developed for AAA titles and those geared toward high-end devices. As an indie game with limited processing requirements and developer, it is clear that using Unreal is too cumbersome. What’s more, for individual developers and small enterprises, Unity has a special charge plan for them. They are free to use Unity to develop their projects, then publish them on open platforms with no extra fees. However, although the Unreal engine can be used freely, once the developer decides to publish the game as a premium one, then the 5% of earnings for Unreal’s company is necessary. In the long run, the Unity engine with streamlined functions and lower individual cost seems to be a better choice[7].

In addition, PyGame is also a relatively common game engine. PyGame is developed using the python language, which is an encapsulation of SDL, a popular game development library many years ago. However, the embarrassing point is compared with Unity, Python does not have an graphics lib or physics engine that can be used for game development. This leads to a problem, using PyGame will consume more time, energy and technical learning costs in the case of making the same style of game which made with Unity. Considering the time and cost of developing the project, Unity is more likely to be chosen.

What about Godot? Godot is an open-source game engine especially famous among beginners. It was released in 2014, and it is now the 4th most popular game engine[5]. Although Godot now has a growing number of users and good development prospects, as a game engine released in 2014, the content of Godot's access store is not very complete. For the same reason, its community is still under construction, which means that during the development of the project, developers may face some questions that have never been solved in the community. These factors will drag the progress of development. The developers will not meet these problems while using Unity, as its access store and community are under running for ten years longer by now.

# Completed Work

The first paragraph is the introduction paragraph. This paragraph usually gives a brief overview of each section in the chapter. The logic flow behinds the section arrangement should also be described.

This chapter describes ‘Completed works’. These works typically include system analysis, data modelling, system architecture, experiment design, etc. However, this chapter should only include original, creative works. Text that mostly describes others’ works should be moved to Chap 1 related works or Chap 2 background.

Also highlight difficulties encountered, alternatives evaluated and solutions adopted.

Content of this chapter will be distributed to Chap 3, 4 and 5 of the Final Report.

## Project Outcome

【功能流程图】

### **Player Movement**

In the game interface, when the player presses the arrow keys, or "WASD", the character will move in the direction corresponding to the player's keys.

In the code implementation, the method responsible for controlling the player's movement is set to first get the value (-1, 0 or 1) entered by the user in the horizontal or vertical direction by pressing the arrow keys. This is then used to make the character move in the corresponding direction. This method runs continuously within the game as the game refreshes (in this case once every 0.04 seconds), thus ensuring that the character is constantly moving. At the same time, the character has different animations when it is walking and when it is idling.



Part of the script for player movement.



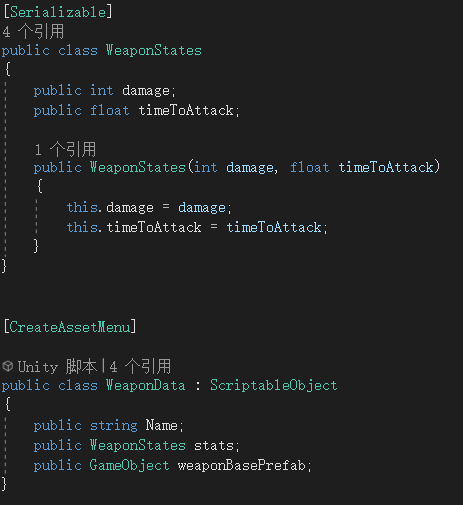
Different animation when the character idling and moving.

### **Weapons**

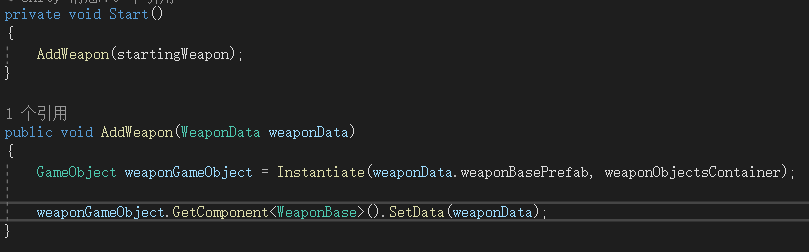
Weapons are used to fight and kill the enemy. In this project, a weapon can be seen as a game object with collision boxes and animation effects. When a weapon's collision box comes into contact with an enemy, it is treated as if an attack has been made on the enemy.

* Weapon Manager

To facilitate the management of the various weapon types, to better reuse the code and to set different values for the weapons, it is necessary to create a weapon manager. First, we set a uniform standard for the basic data of all weapons. Set the type of parameters in this script file.

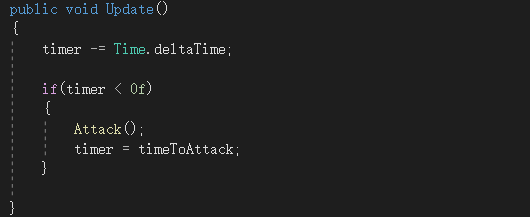


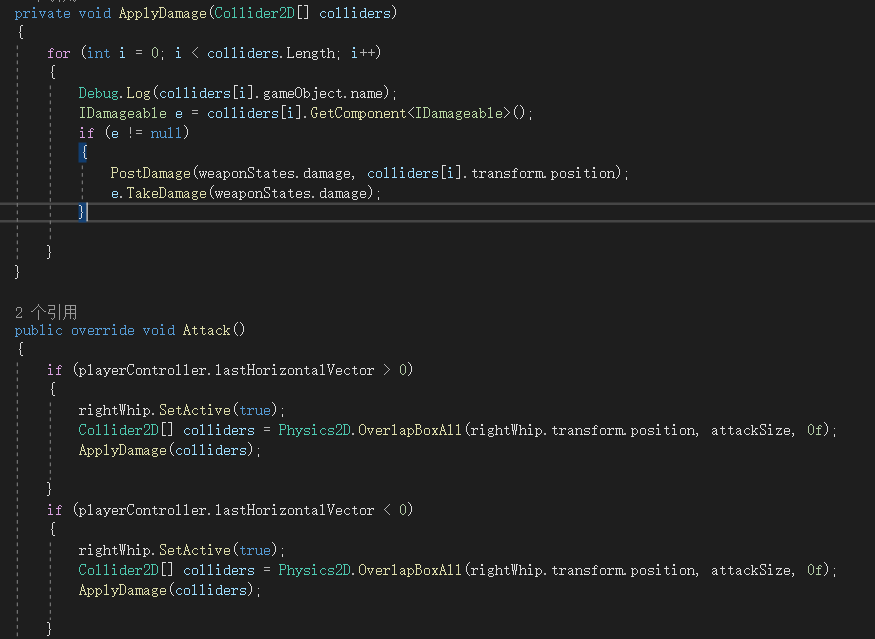
Then, in the Weapon Manager script file, set the starting weapon for the character, which will be set as the first weapon for the character as the game start. The “AddWeapon” function is used to instantiate a weapon when it is needed. Then the new weapon will be added to the character. The specific data of each type is set in the Unity front-end interface, which is convenient for modulation.



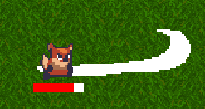
* Close-range Weapons

For close-range weapons, a weapon object bound to the player character is first created in the Unity front-end, which is then controlled by a timer to appear or disappear (if the object appears at the moment its collider overlaps with the enemy’s, it is considered to have attacked the enemy).





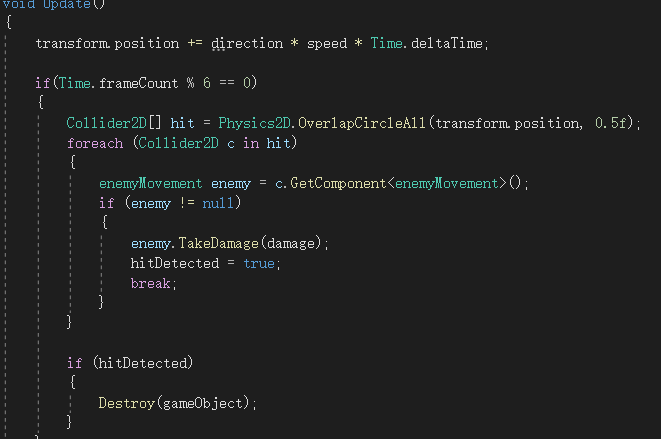
These two parts of the code set a timer to ensure that the system will activate the weapon object after a certain time. Meanwhile the code in the second image controls the weapon object to deal damage to enemies if they collide together, as well as changing the scale of the activated weapon object depending on the player's face scale.

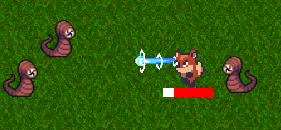
 

* Projectile Weapons

Unlike close-range weapons, while projectile weapons also need to be generated in a direction and position determined by the player's direction and position, they have their own trajectory after generation and therefore require different processing logic.

As the code below, a projectile weapon is generated at fixed intervals. After it has been generated, it will move in a straight line until it hits an enemy, otherwise it will move until it has reached the maximum distance and then disappear (will be introduced in 3.1.10). When it hits an enemy, the projectile weapon deals damage to the enemy it hit and destroys itself.

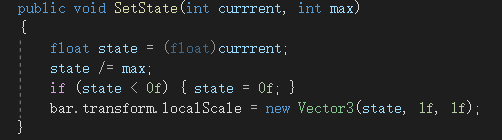


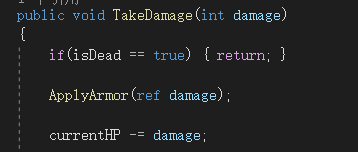


### **Health Point (HP) System**

HP is a value used to keep track of the player's life value.

In this project, the HP bar is a red bar that follows the player's feet. When the player takes damage, the HP is gradually reduced. This behaviour is represented on screen as the red part of the bar gradually being replaced by white. When the player picks up an item that restores life, HP is restored a certain value.



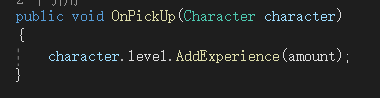


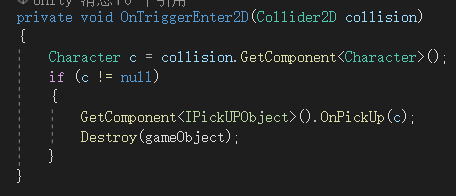


### **Experience and Level Up**

When players defeat enemies, and when they pick up gems (drops from monsters after they die), they increase their experience value. When the experience value reaches a certain value, the player will automatically upgrade.

The visual progress bar of experience value is displayed at the top of the entire game screen, similar to the principle of the HP bar. he experience value for each action increase is set in Unity's front-end interface. The current formula for the amount of experience needed to upgrade is set to current level \* 1000.





JCC]U(M$@CJG$TCH9O1%5YB

OB[X%I@THMWRQ2~4]J%~A1M

### **Upgrading**

### **Passive Items**

### **Enemy**

### **Spawn System**

### **Damage Popups**

### **Destroy Objects**

### **Music**

### **Main Menu, Pause and Losing Game**

## Second Topic

Sample text sample text Sample text sample text Sample text sample text Sample text sample text. Sample text sample text Sample text sample text, Sample text sample text Sample text sample text.

# On-going and Future Work

Write an introduction paragraph to delineate the content and logic flow of this chapter.

* Describe partially done works
* Include a Gantt chart as evidence of effective project planning for the 2nd semester
* Show Clear idea of what to do to complete the project

## First Topic

Sample text sample text Sample text sample text Sample text sample text Sample text sample text. Sample text sample text Sample text sample text, Sample text sample text Sample text sample text.

## Second Topic

Sample text sample text Sample text sample text Sample text sample text Sample text sample text. Sample text sample text Sample text sample text, Sample text sample text Sample text sample text.

# Conclusion

Reflect on the progress of the project. Can use first person pronoun to write.

Content may be moved to the Reflection appendix in the Final Report.

Sample text sample text Sample text sample text Sample text sample text Sample text sample text. Sample text sample text Sample text sample text, Sample text sample text Sample text sample text.

References

1. Kovanto, A. (2013). The Improvements for Indie Game Development.
2. “A world at play: The current State of the video game industry,” Liquid Web. [Online]. Available: https://www.liquidweb.com/insights/video-game-statistics/.
3. “Vampire Survivors on steam,” Steampowered.com. [Online]. Available: https://store.steampowered.com/app/1794680/Vampire\_Survivors/. .
4. Unity - Introduction. (n.d.). Tutorialspoint.com. Retrieved September 3, 2022, from https://www.tutorialspoint.com/unity/unity\_introduction.html
5. Godot vs Unity: Detailed comparison in 2022. (2022, July 15). Whimsy Games. https://whimsygames.co/blog/godot-vs-unity-all-you-need-to-know/
6. Payne, J. (2021, May 5). Benefits of C#. CodeGuru. https://www.codeguru.com/csharp/benefits-of-c/
7. Omelchenko, S. (2021, April 13). Unity vs. Unreal: Choosing the best engine for 2022. Program-Ace. https://program-ace.com/blog/unity-vs-unreal/

[6] Hector J. Levesque. Foundations of a functional approach to knowledge representation. Artificial Intelligence, 23(2):155–212, July 1984.

[7] Hector J. Levesque. A logic of implicit and explicit belief. In Proceedings of the Fourth National Conference on Artificial Intelligence, pages 198–202, Austin, Texas, August 1984. American Association for Artificial Intelligence.

[8] Bernhard Nebel. On the compilability and expressive power of propositional planning formalisms. Journal of Artificial Intelligence Research, 12:271–315, 2000.

[9] Ivan Marsic. A short guide for writing a thesis. http://www.ece.rutgers.edu/~marsic/thesis-guide.html, 2004 [Mar. 6, 2014].

[10] Matlab documentation. http://www.mathworks.com/help/?s\_tid=hp\_ff\_s\_doc [Mar. 6, 2014].