Xi'an Jiaotong-Liverpool University
西交利物浦大學

CPT306
Knight Defend

Student Name	:	Xiao Xiao
Student ID	:	1716858

Table of Contents

Overview

The game is a survival game where you play as a hero in a sea of monsters and fight to survive. You can play as a knight to rescue a castle under siege by the Slimes army. A variety of Slime monsters appear from the ground inside the castle and from the land outside, and you have to survive for a certain amount of time, kill the monsters to increase your level, gain better stats to deal with more powerful monsters in subsequent stages, and finally win the game after killing all the monsters.

Story Synopsis

On a fantasy land where humans have lived for hundreds of years and built vast kingdoms. Even though orcs, goblins and other races were stalking the human lands outside the Empire's borders, the kingdom's army was able to keep the people safe and secure. However, the army of the kingdom could not protect every inch of the land. A castle in a remote part of the kingdom border was attacked by the Slimes army. The militia inside the castle could not resist the tide of Slimes monsters, so they had to abandon the castle together with the Lord to protect the people and seek help from the kingdom. At an army post behind the castle, the knight who received the distress message immediately chose to lead the team to rescue and recapture the castle. After receiving the fleeing people and securing a safe place, the knight continued his procession to the castle and attempted to destroy the Slime inside. The knights and army were able to move into the castle without much obstacle, but the knights miscalculated the size of the Slimes' army, which was not a small band led by a single Slimes king, but a large cluster of multiple Slimes Kings combined. So, on the night they entered the castle, the Slimes army suddenly came out of the ground and attacked the army. The soldiers were not able to react, and almost all of them were killed. Only a few of them were able to escape the castle. The knight who remained in the center of the castle was completely surrounded. He had to fight his way through the Slimes and inform the kingdom of what was going on.

Gameplay Design

The game is a top-down, action-survival game where players play as knights against different kinds of Slimes monsters. From the picture, we can see that the main scene of the game takes place in a castle, and the knights are besieged in the center of the castle, surrounded by tower buildings, lamps, campfires, hay carts and other facilities, as well as some boxes of supplies.

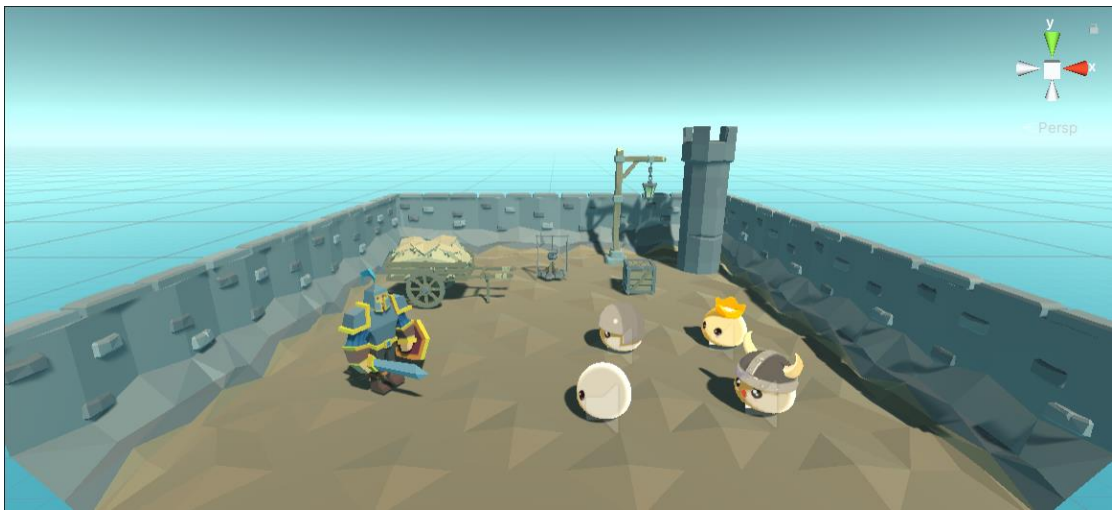


Figure1 Main Menu Scene

The actual scene of the game is the floor inside the castle, surrounded by a circle of towers, and outside the tower is the upwelling earth caused by the continuous flow of Slimes. In the game, the player can control the knight to walk, run, attack and jump. Among them, the keyboard wasd is to manipulate the knight to move up and down, the space bar controls the knight to jump, the shift key controls the knight to run, and the left mouse button controls the knight to attack with a long sword. The Knight has two attack moves, a normal attack and a critical attack. Critical strikes deal more damage than normal attacks, but critical strikes are probabilistic only. A knight's attack on a Slime reduces his health, and he dies when his health drops to zero. Slimes also attack knights, dying when their health drops to zero. The knight has a stamina bar, which is used up when running. When you stop running, the stamina bar is restored over time. When the stamina bar is used up, you cannot run. Each Slimes killed gives the knight a certain amount of experience, and more advanced monsters give the knight more experience. Knights level up after gaining enough experience, which increases their health and attack stats, and fully restores their health.

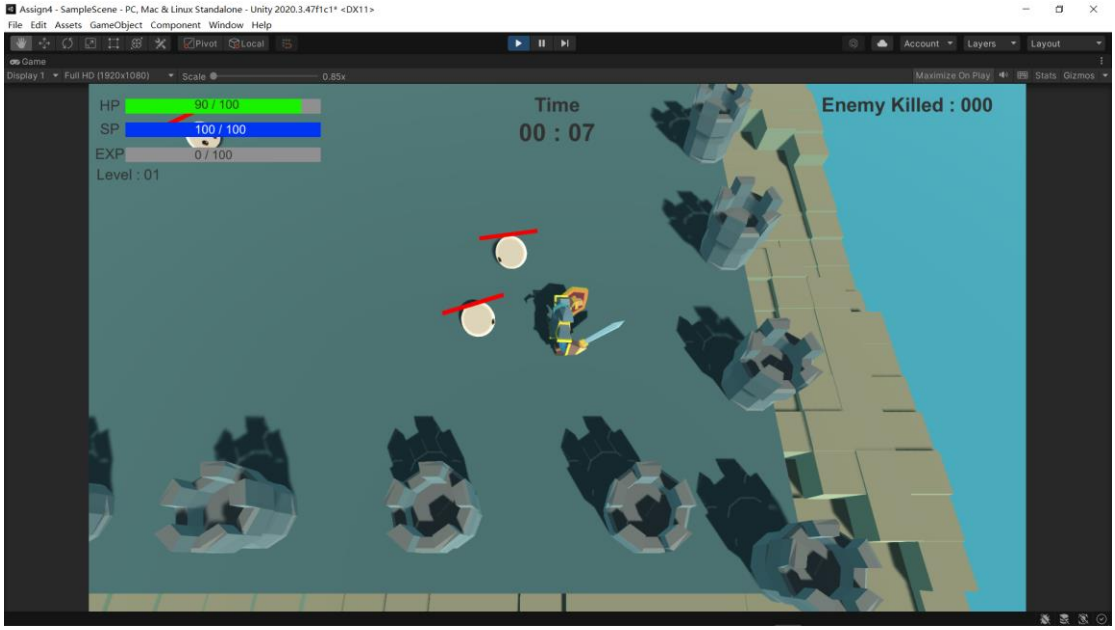


Figure2 Gameplay Scene

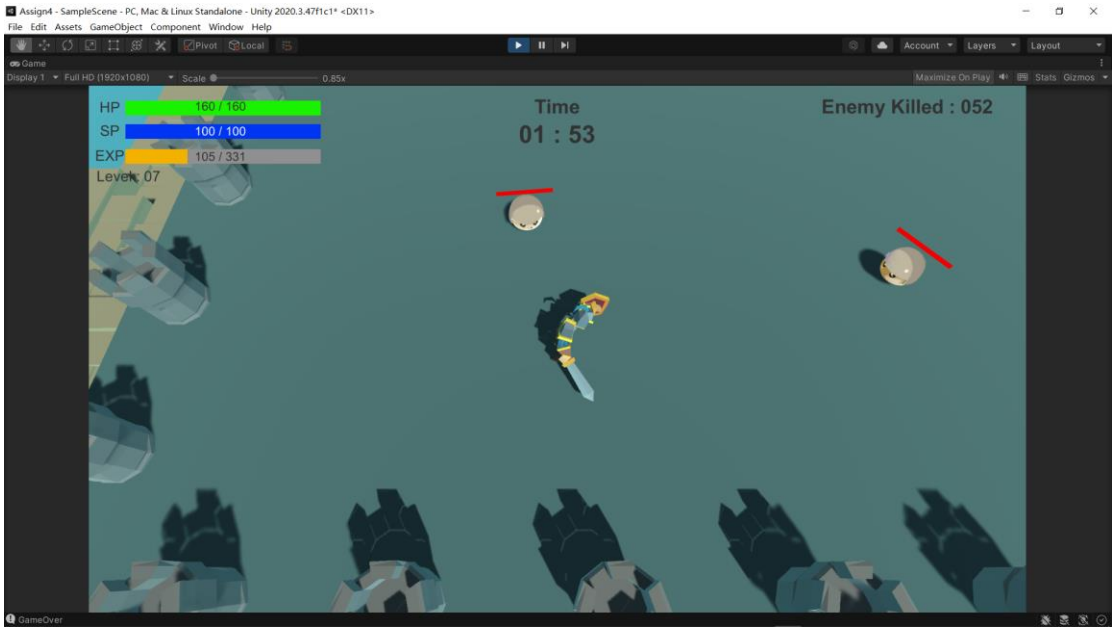


Figure3 Attack

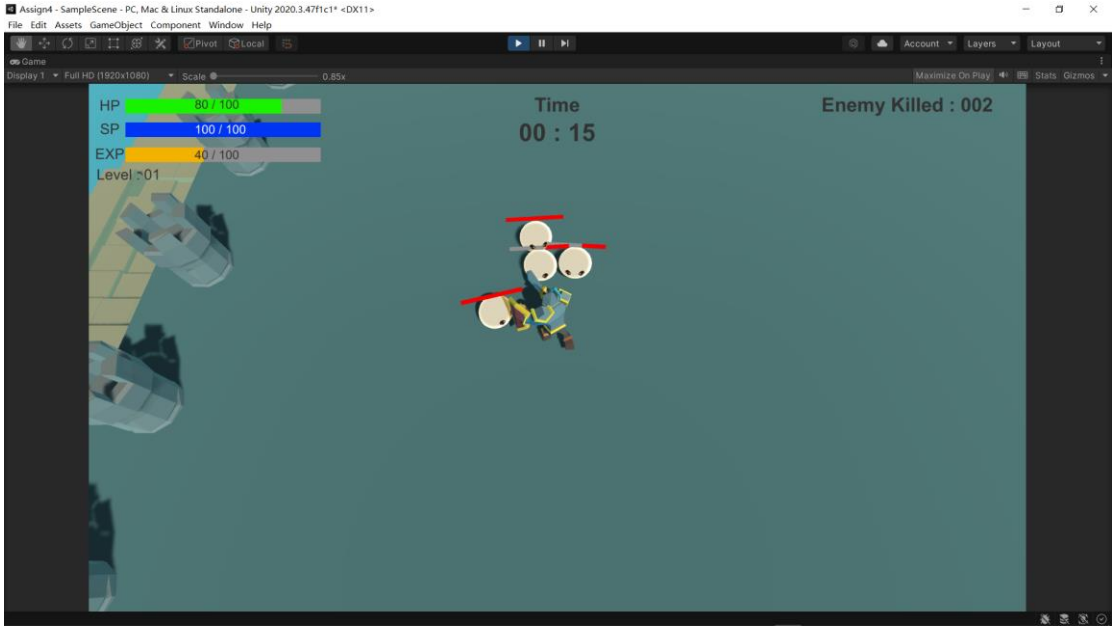


Figure4 Critical Attack

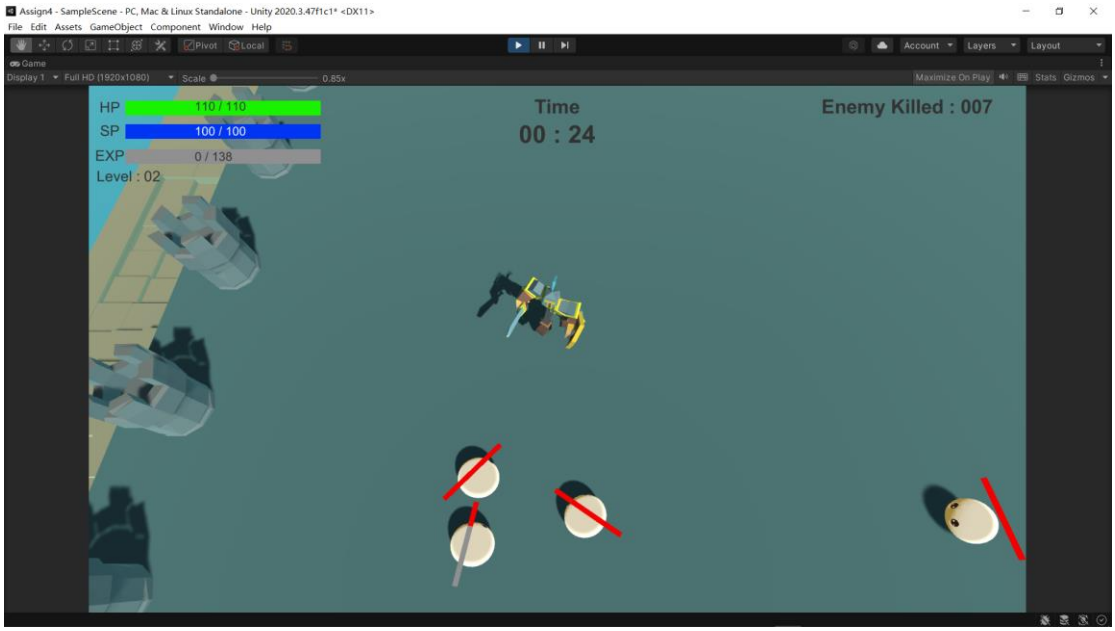


Figure5 Jump

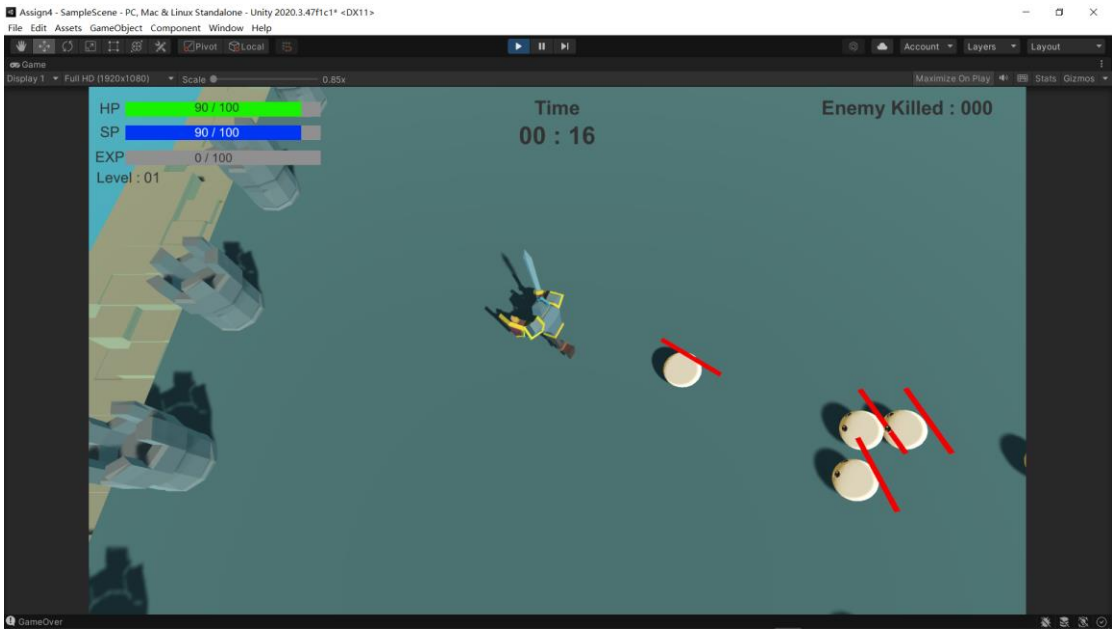


Figure6 Run

The game has a total of four stages, each of which lasts for one minute and automatically enters the next stage after one minute. The corresponding Slime monster will be generated. If the player does not kill the monster immediately after it is generated, the monster will gradually increase its speed according to the passage of time. Players need to move around in each stage, killing enemies to increase their level and gain better stats against Slimes monsters with higher health and attack stats in the next stage. The game is won when the player has survived all four stages and killed all the monsters currently alive in the field after four minutes. However, if the player allows a knight to be killed by a monster at any stage of the game, the game is lost.

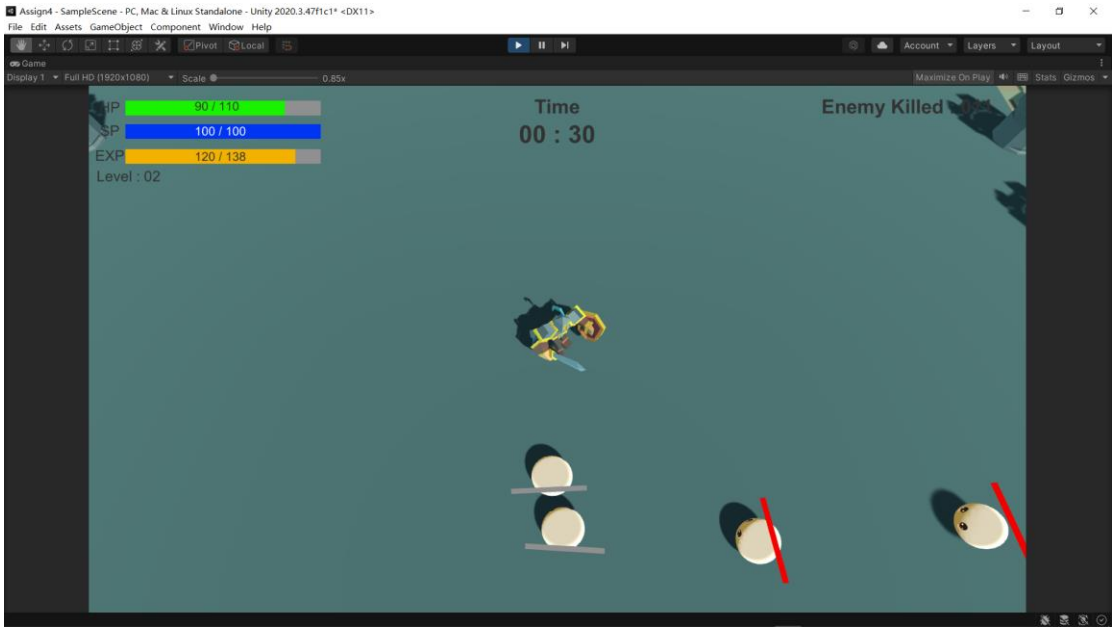


Figure7 Stage 1

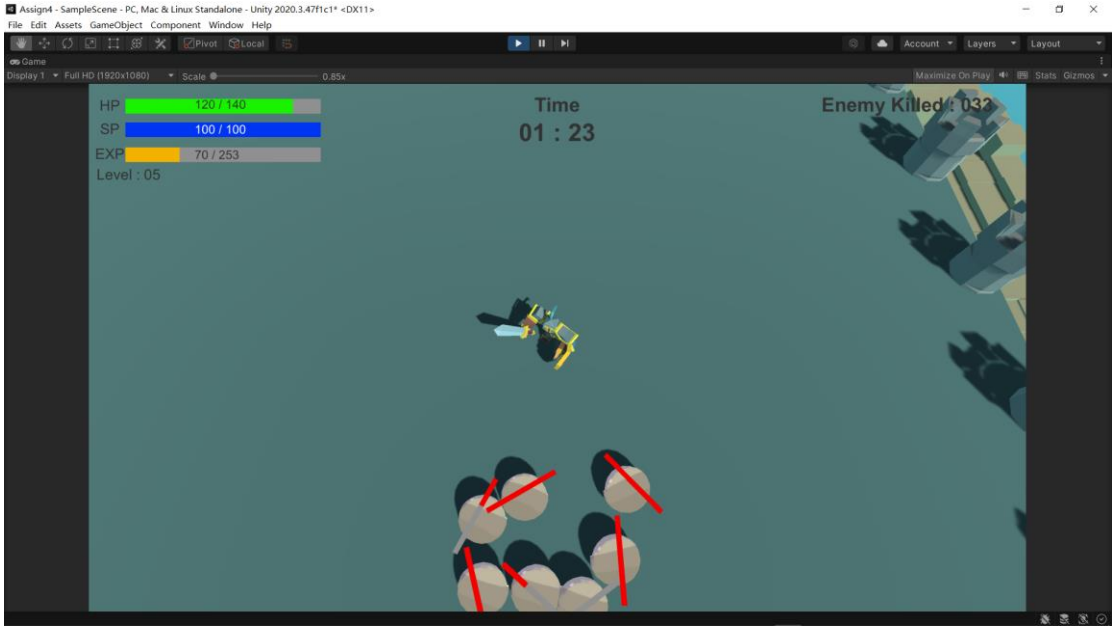


Figure8 Stage 2

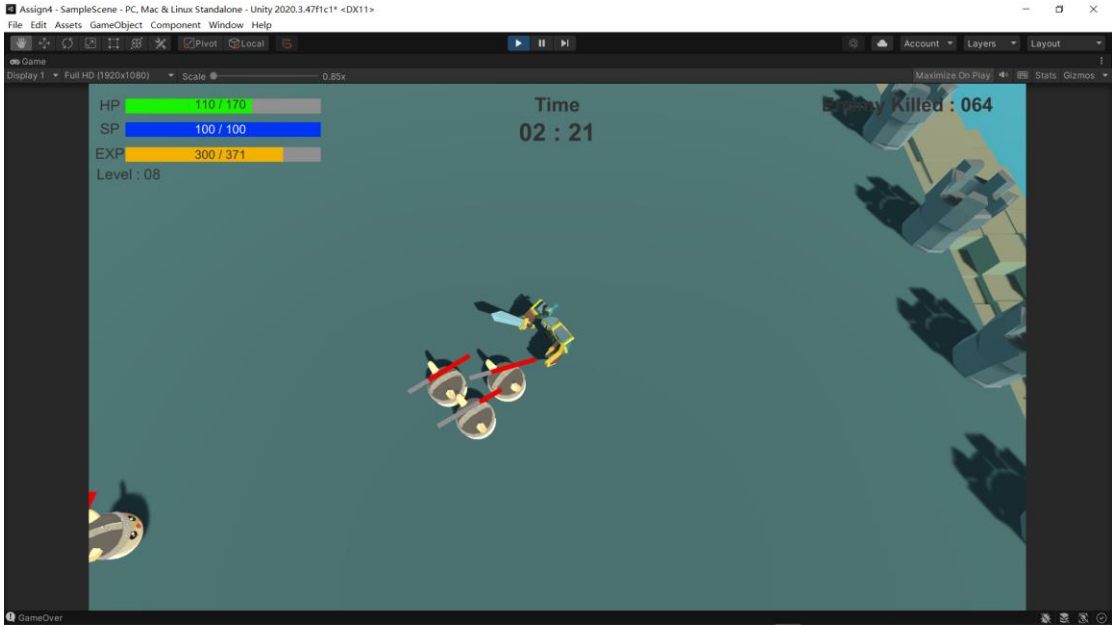


Figure9 Stage 3

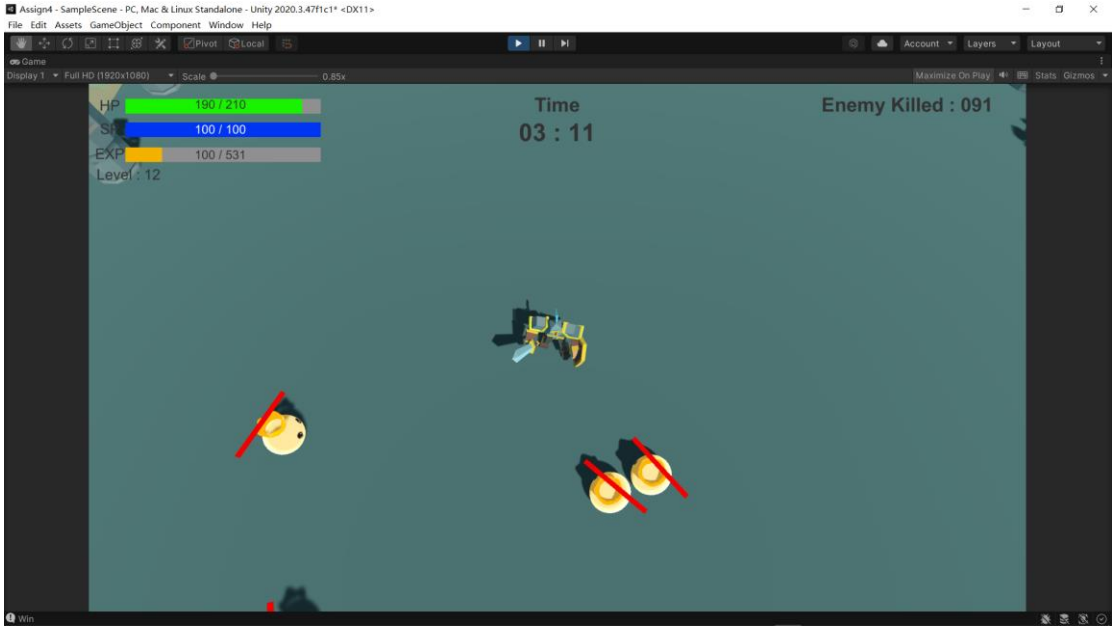


Figure10 Stage 4

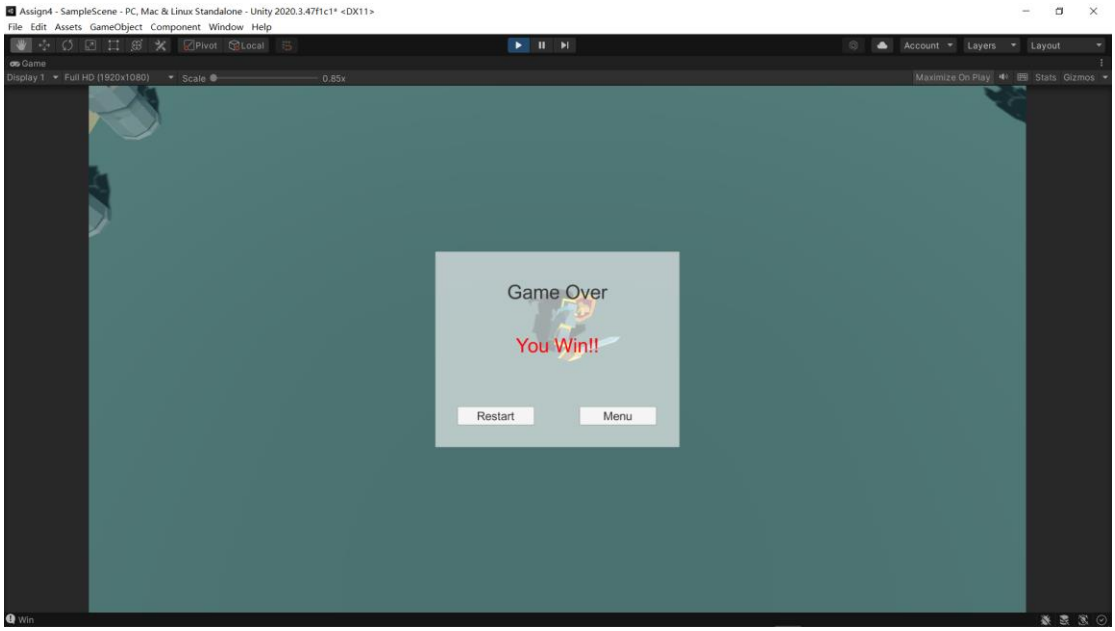


Figure11 Win Scene

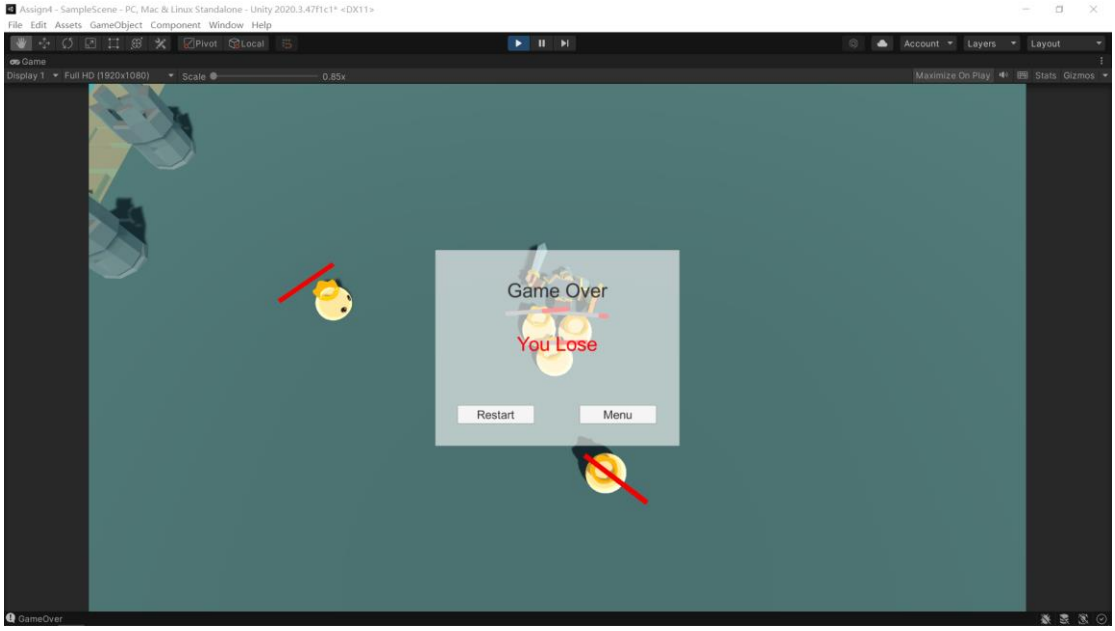


Figure12 Lose Scene

During the course of the game, monsters will spawn from various places and automatically find their way to the player's location and attack the player. Monsters generated in the off-field land will climb from the land to the castle floor and then automatically find their way to the player. Different background music is provided in the main menu interface of the game and at different stages of the game. At the same time, clicking the button in each menu will also have the sound effect of clicking the mouse, and the player controls the knight to swing the sword and hit the monster will have the corresponding attack and hit the sound effect.

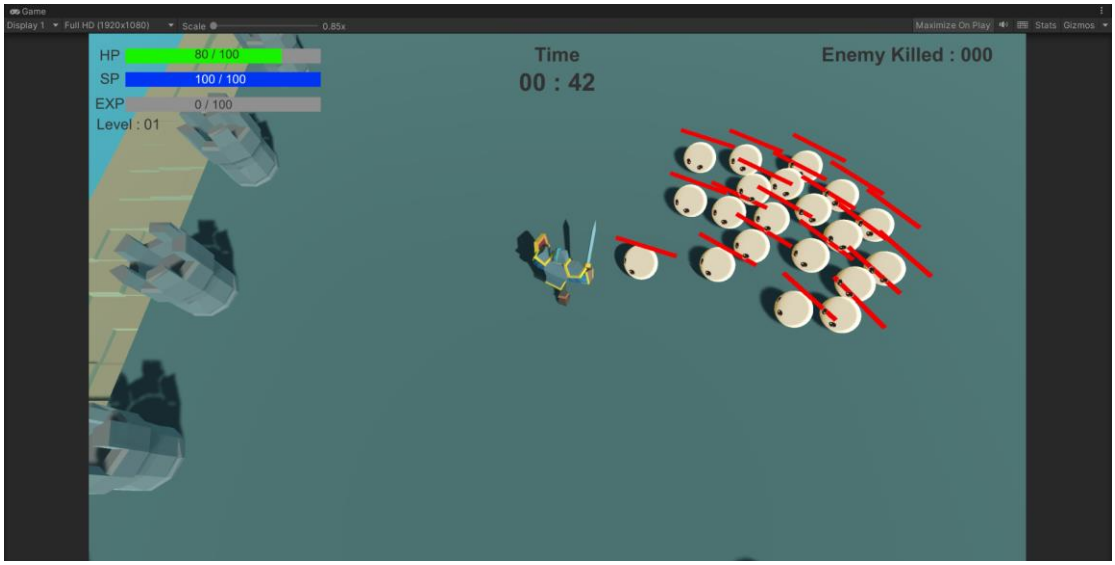


Figure13 Auto Routing

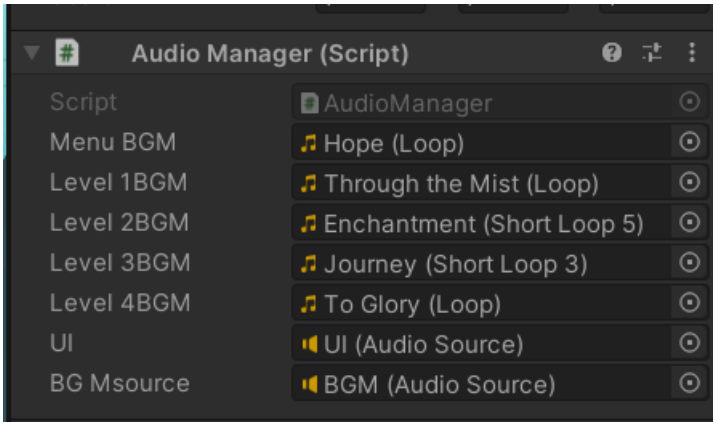


Figure14 Back Ground Music

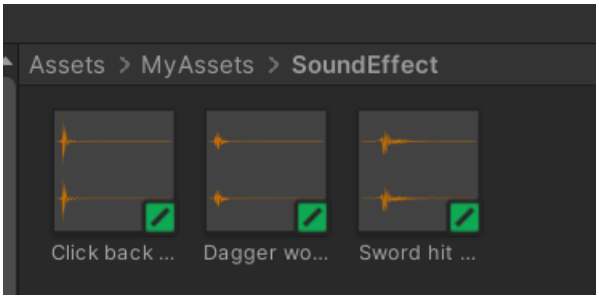


Figure15 Click, Attack, Hit Sound

The user interface of the game includes the main menu interface, the pause interface, the game victory interface and the game end interface. The game victory interface and the game end interface have been shown in the previous article. In the main menu interface, there are three options: new game, loading game and quitting game. Click the new game to start a new game, and load the game progress saved before can be loaded. In the pause interface, there are three options: save the game, resume the game and return to the main menu. Save the game can save the existing progress of the game and continue to play through loading the game in the main menu interface; resume the game can release the pause and continue the game; return to the main menu is to quit playing and return to the main menu without saving the game.

In addition to the menu interface, the game interface also has a corresponding user interface design. In the upper left corner of the game screen, the player character's health, health and experience bars are arranged, and the player's current level is displayed below. On the top of the game interface in the middle is the display of the current game time, to remind the player of the game progress and the time until the next stage. Finally, the top right of the game screen shows the player's current monster kills.

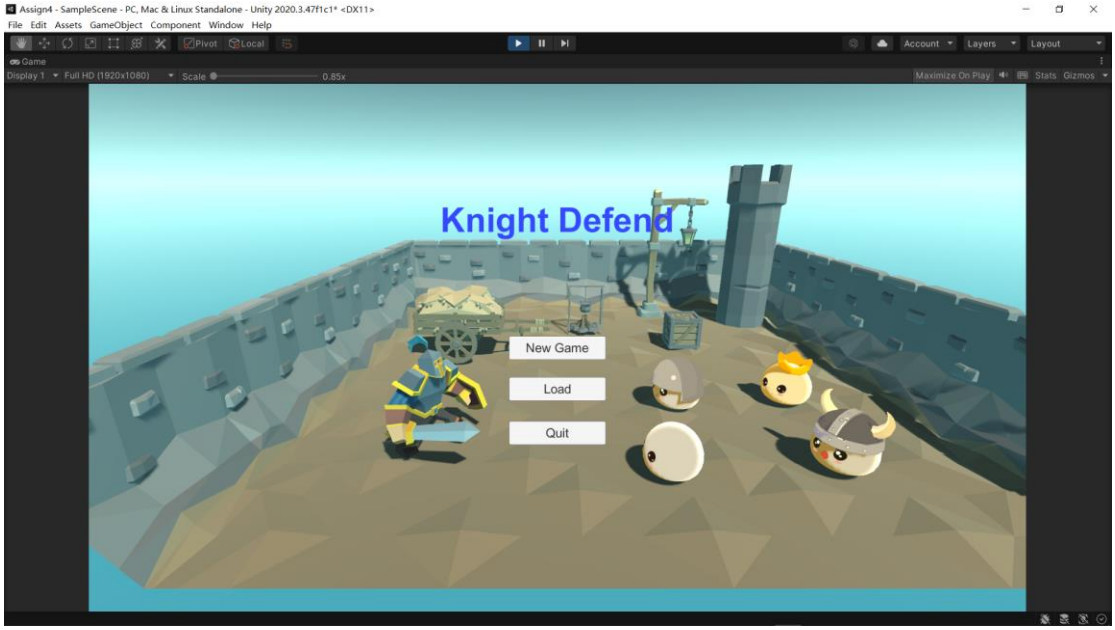


Figure16 Main Menu

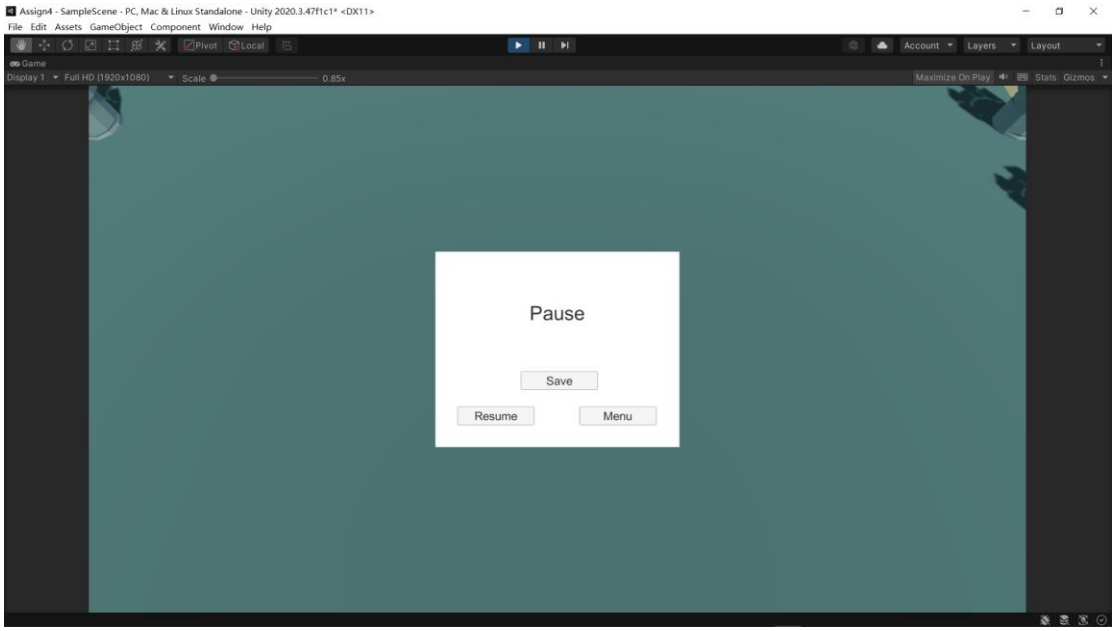


Figure17 Pause Menu

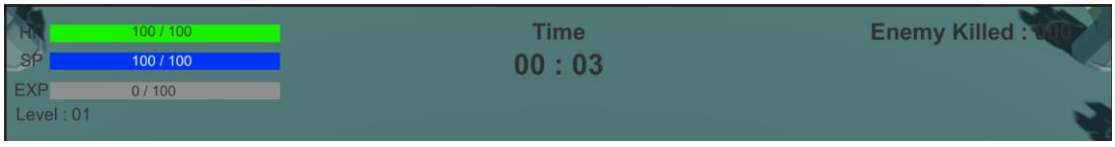
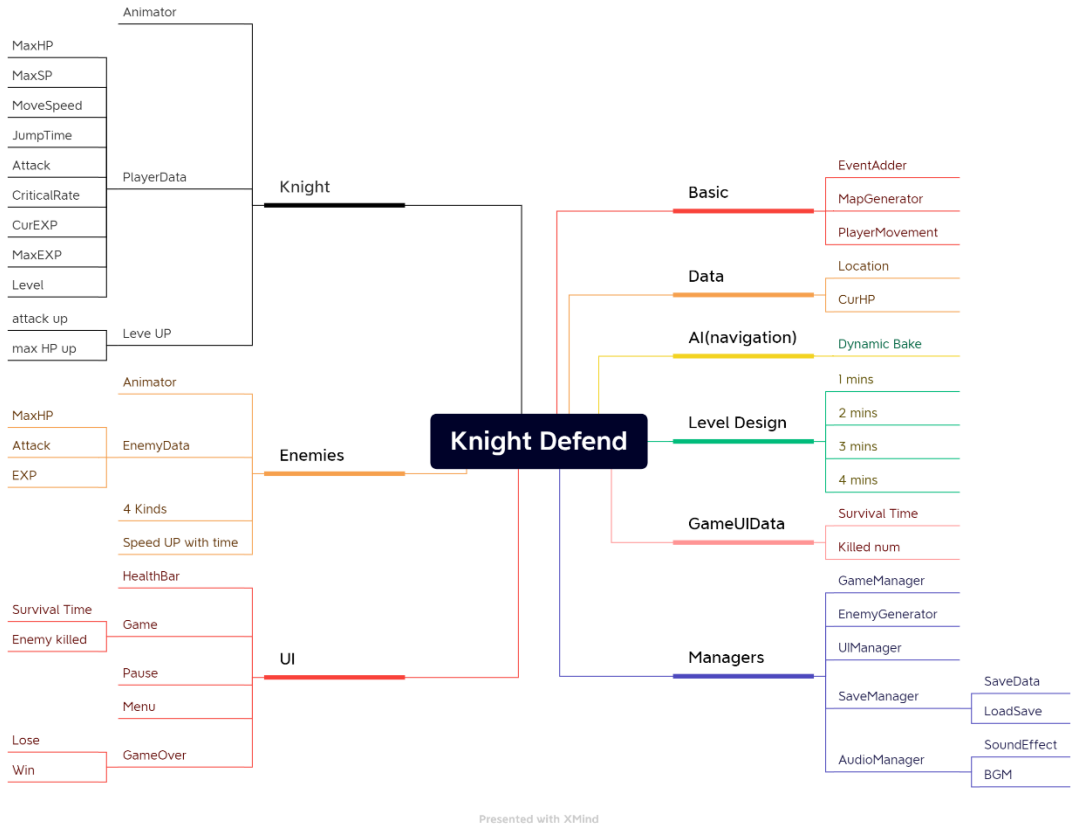


Figure18 Gameplay Scene UI

Implementation



From the UML diagram, we can see that the game is roughly divided into nine components. The first part of development is the Basic section, which contains the game's EventAdder, MapGenerator, and PlayerMovement. EventAdder is used to add an event to the target object, which is used to detect whether the event in the game is triggered normally, while MapGenerator is used to generate the main menu scene and game scene of the game, and finally PlayerMovement is used to write the movement mode and other actions of the player to control the character. Such as attacking, running and jumping.

In the first part we get a basic game scene and how to control the characters, and then in the second part we build the player's characters and monster characters first. After importing the player character model from Unity's resource site, you first need to adjust the various components of the model, adding rigid body, collision boxes to the model, and attaching the character control code to the model components. The player then sets various attributes for the player's character, including health, stamina, movement speed, attack power, level, etc., and sets the conditions and consequences of leveling up accordingly. Finally, add the appropriate action animations for the player's character model, including animations for running, attacks for normal and critical hits, and so on. After setting up the player character, the monster model needs to do the same. First, import the first monster model from the resource website, also add various components to the monster model, and write the movement code for the monster. Then set the corresponding health and attack power for different monsters, as well as the attributes

that give experience. Finally, add motion animations for monsters to move and attack.

The third part is to set the birth location and health for the player character and monsters, and to set up automatic pathfinding AI for the monsters. For the automatic pathfinding AI setup of monsters, we use the NavMesh component, which can easily add AI to monsters.

The fourth part is the UI design of each menu interface and the game interface. The menu interface includes the design of the main menu interface, the pause interface, and the game victory and failure interface. The game interface contains the health bar on top of the monster, the player character's health, stamina, experience, and level UI.

The fifth part is to continue to import the remaining three monster models and set and add model components, model actions and attributes, set different stages for the course of the game, set different monsters in different stages, and set the start and end time for each stage.

The sixth part is to improve the UI of the game interface, adding the timing of the game time and the statistics of the number of roles killed, adding the corresponding UI in the game interface, and adding a GameUIData script file to achieve this function.

The seventh section starts with the GameManager managing and scripting all of the player character setup and generation, map generation, and game running. The monster related setup and generation in EnemyGenerator for unified management and script. Then, all kinds of UI Settings in the scene are unified into a UIManager for scripting, and save and load functions are added to the game, so that it can be normally associated and realized. Finally, background music was added to in-game menus and scenes, as well as sound effects for mouse clicks and player character attacks and hits.

Ethical and Social Issues

Knight Defend is aimed primarily at action-adventure game enthusiasts who prefer an immersive gaming experience. The game aims to appeal to a global audience, transcending geographical boundaries. The game's world and narrative elements strive to provide an experience with universal appeal, while respecting cultural sensitivities and avoiding stereotypes. The game is designed to appeal to players who enjoy intense combat, strategic decisions and character development. It's aimed at people who enjoy the sense of accomplishment that comes from overcoming challenges. The game is aimed at players over the age of 12, but is primarily aimed at adults. Its content may include moderate violence and require a degree of critical thinking and flexibility. With the target demographic in mind, the game is designed to be a fun and engaging experience for young people who enjoy action-oriented gameplay.

While the core of the game is to manipulate the character to defeat monsters, there are elements of violence and aggression in the game, the presentation of violence is very concise, ensuring that the content is not too bloody and exciting, there is nothing that promotes violence, and the backstory is a passive defense in order to defend the home. The game also balances the intensity between the player and the monster, keeping within moral boundaries.

At the same time, the completion time of the game is short -- four minutes is enough to complete the game, so the game is less addictive, allowing players more freedom to manage their time without holding them hostage to the game. Great for helping players maintain a balanced life.

Limitations and Future Work

The Knight Defend has a number of shortcomings, such as the possibility to deal multiple damage with one attack in the player's attack decision against monsters. This is because the player's weapon model collider is not the weapon itself, but is set up as a circular collider, which makes it possible for the weapon to do multiple damage when it overlaps with the monster's collider for a long time. This was intended to fix the problem that if the weapon's collider was the weapon itself, it might not be able to hit the enemy due to the limitations of the player's attack animation. Therefore, in order to correct this problem, I think it is possible to limit the collision detection between a weapon and an individual monster to one, that is, a monster can only be judged once per attack. In addition, there is a problem in the game that when the jump button is pressed, there is a chance that a new player character will regenerate. This is because the decision of player character generation is whether a character touches the ground. Although there are restrictions on player character generation after the first time, there are still bugs. I think this can be fixed by changing the settings of the code or other game interface components.

For future work on the game, I think it's possible to add more things to the player character, such as skills and equipment, as well as more attack modes for monsters, such as ranged attacks, and a boss-type monster in the monster category to balance the increase in skill and

equipment for the player character. In addition, levels and maps can be designed to continue adding to the story of the game.