CPT306 Principles of Computer Games Design

Legends of Animalia

Zirui Zhou
1927924

Table of Contents

1. **Overview** **3**
2. **Story Synopsis** **3**
    1. Background 3
    2. Character 3
        1. Rex 4
    3. Villains 5
        1. Slime 5
        2. Turtle 5
        3. Skeleton 5
3. **Gameplay Design** **6**
    1. Game Genre 6
    2. Gameplay Mechanics and Gameplay 6
        1. Action 6
        2. Mechanisms 7
    3. Game Elements 8
        1. Graphs 8
        2. Animation 8
        3. Artificial Intelligence (AI) 10
        4. Sound 11
    4. Game Progression 12
    5. User Interfaces 12
4. **Implementation** **14**
    1. Game Manager 14
    2. Camera Manager 14
    3. Player Manager 15
    4. Enemy Manager 15
    5. UI Manager 15
    6. Save Manager 15
5. **Ethical and Social Issues** **15**
    1. Animal Welfare and Conservation 15
    2. Cultural Sensitivity 16
    3. Violence and Aggression 16
    4. Inclusivity and Diversity 16
6. **Limitations and Future Work** **16**
    1. Limitations 16
    2. Future Work 16
        1. Equipment and Prop System 16
        2. Skill System 17
        3. Boss System 17

1. Overview

The " Legends of Animalia" is a survival and roguelike game which tells the traditional adventure stories of different animal characters to survive among the attack of the monster waves. The main character Shiba Inu Rex is one of the adventurer members, who has two abilities, sword mastery and shield defense, depended on his small sword and shield. However, countless monsters rampage, occupying many important places and causing many casualties. In one quest, Rex encounters the ambush of slimes, turtles and skeletons in the forest. Faced with the enemy swarming like a tidal wave, he needs to slay as many enemies as possible to raise his level and eventually survive. Player can explore the forest and fighter against the monsters in the third-person top-down perspective. In the role of this adventure knight Rex, your own legends will be written and remembered in this Animalia land.

2. Story Synopsis

1. Background

The " Legends of Animalia" tells the traditional adventure stories of different animal characters to survive among the attack of the monster waves. The world of this game is located in a fantasy land, Animalia, of breathtaking beauty, filled with lush forests, majestic mountains, serene rivers, and vast grasslands. Animals are endowed with the intelligence and emotions similar with humans in the real life and established their own settlements and kingdoms in the thriving fields. However, the peaceful realm is now under threat from an ancient curse, that has cast a shadow of darkness and chaos across the land. Countless monsters rampage, occupying many important places and causing many casualties. As a group of brave animal adventurers and knights, it is your duty to embark on a thrilling quest to restore balance and harmony to Animalia and write your own legends.

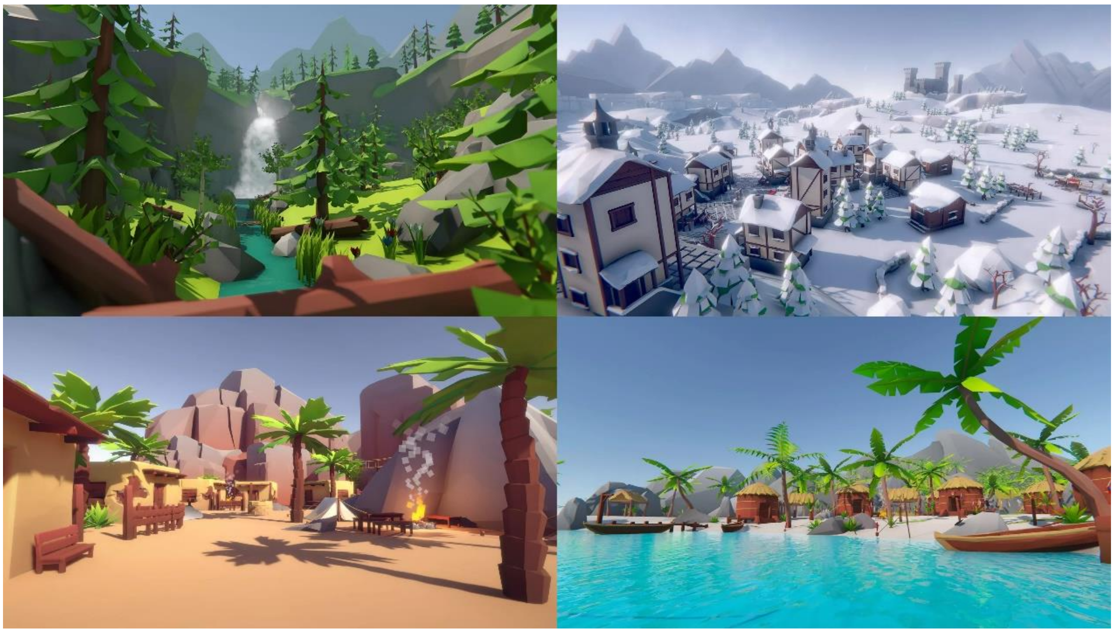


Figure 1: Different terrains and scenes prototype in the Animalia in low-poly style.

2. Character

The main characters are most adventurer members of The Adventure Association, which acts as a central gathering point for individuals who seek to embark on daring quests, explore uncharted territories, and confront the numerous challenges that threaten the kingdom.

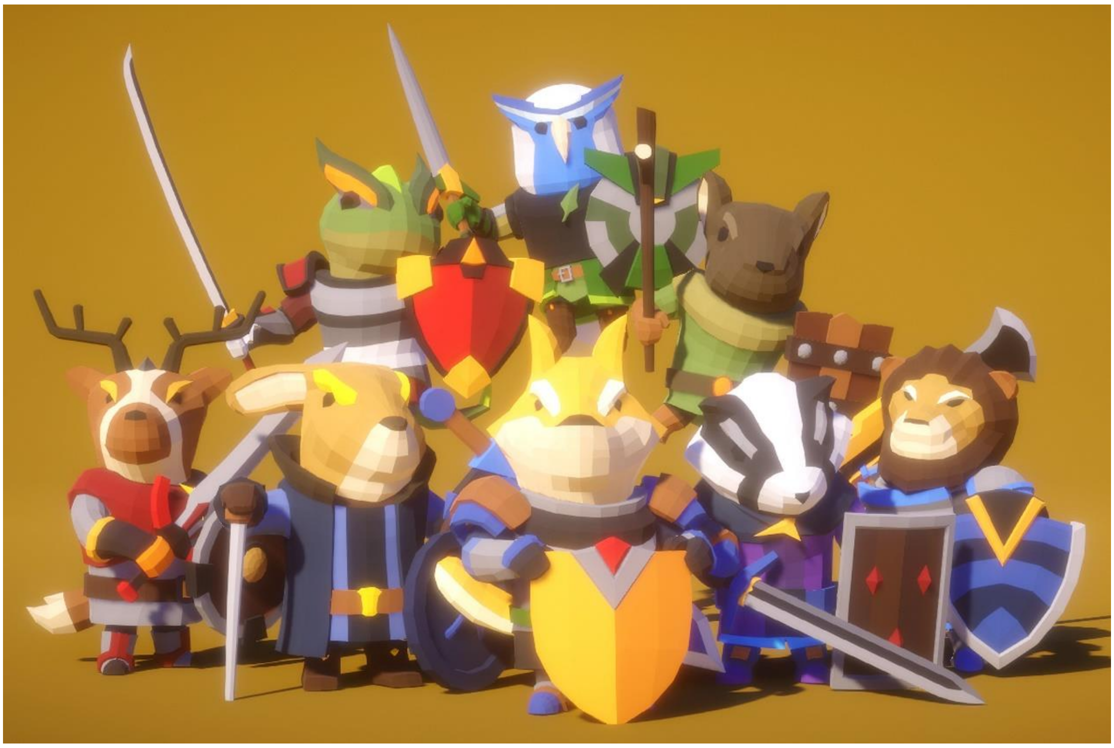


Figure 2: The character prototypes in The Adventure Association.

1. Rex

A brave Shiba Inu with a heart of gold (so do his fur). Rex joins the animal adventurers as a skilled warrior, wielding a small sword and shield with precision and agility. He is not only known for his unwavering loyalty and unwavering determination to protect his friends, but also his optimistic altitude to approach every situation with a lighthearted demeanor, finding joy in the midst of even the direst circumstances. He has two abilities depended on his small sword and shield.

a) Sword Mastery: Rex's training and natural dexterity with his small sword allow him to execute swift and precise strikes, dealing significant damage to enemies.

b) Shield Defense: Equipped with a sturdy shield, Rex can block incoming attacks, mitigating damage to himself and his allies.

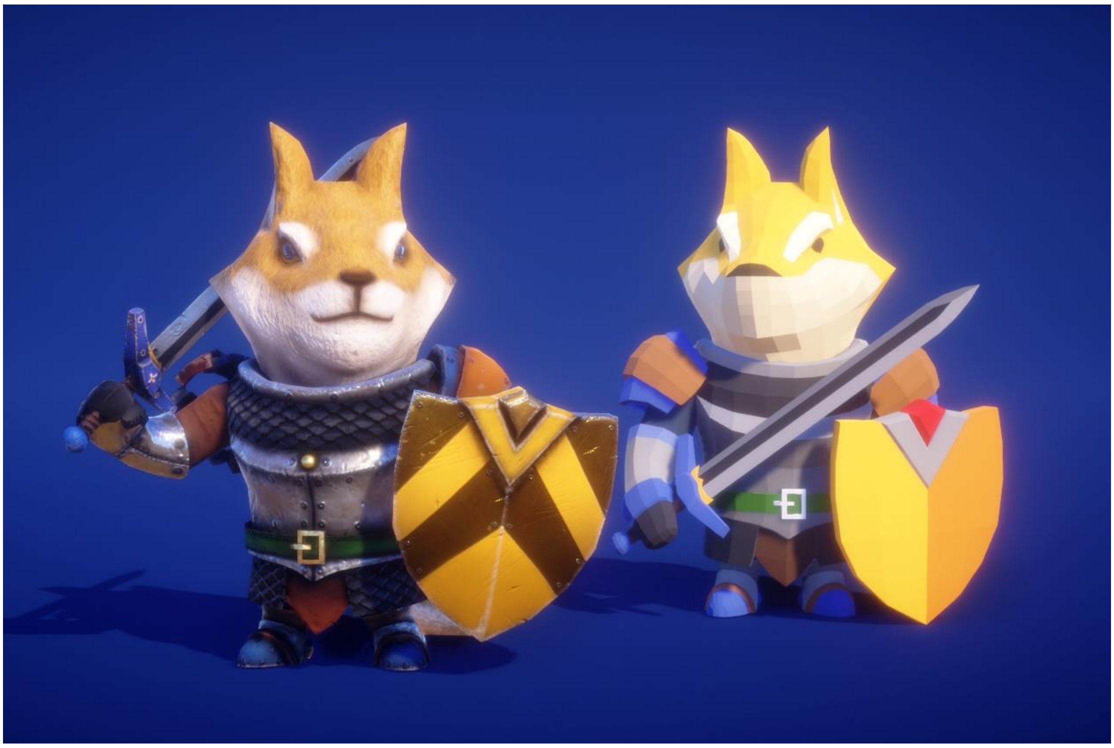


Figure 3: The 3D model of Rex equipped with his sword and shield.

3. Villains

1. Slime

Slimes are gelatinous creatures that inhabit the dark and damp areas. They come in various colors and sizes, ranging from small puddle-like slimes to large, menacing blobs, in different environments. For example, slime populations in forested terrain mostly show a medium red body. Slimes are known for their acidic attacks, which can corrode armor and weapons. However, crashing with their deformable, hardened body is also a means of attack in some cases.

2. Turtle

Turtles in Animalia are not the ordinary slow-moving creatures. These formidable reptiles have developed hardened shells and powerful limbs. They come in different species, each with unique abilities. Some turtle species are equipped with sharp spikes on their shells, enabling them to perform devastating rolling attacks. Others have the ability to withdraw into their shells, rendering them impervious to damage for a short period. Turtles are known for their resilience and endurance, making them formidable adversaries in combat.

3. Skeleton

Rising from the depths of forgotten tombs and ancient battlegrounds, skeletons are undead creatures that haunt the darkest corners of Animalia. These skeletal warriors are remnants of fallen heroes and warriors, animated by dark magic or cursed spirits. Skeletons are clad in tattered armor and wield rusty weapons, their bones rattling as they march forward with an eerie determination.

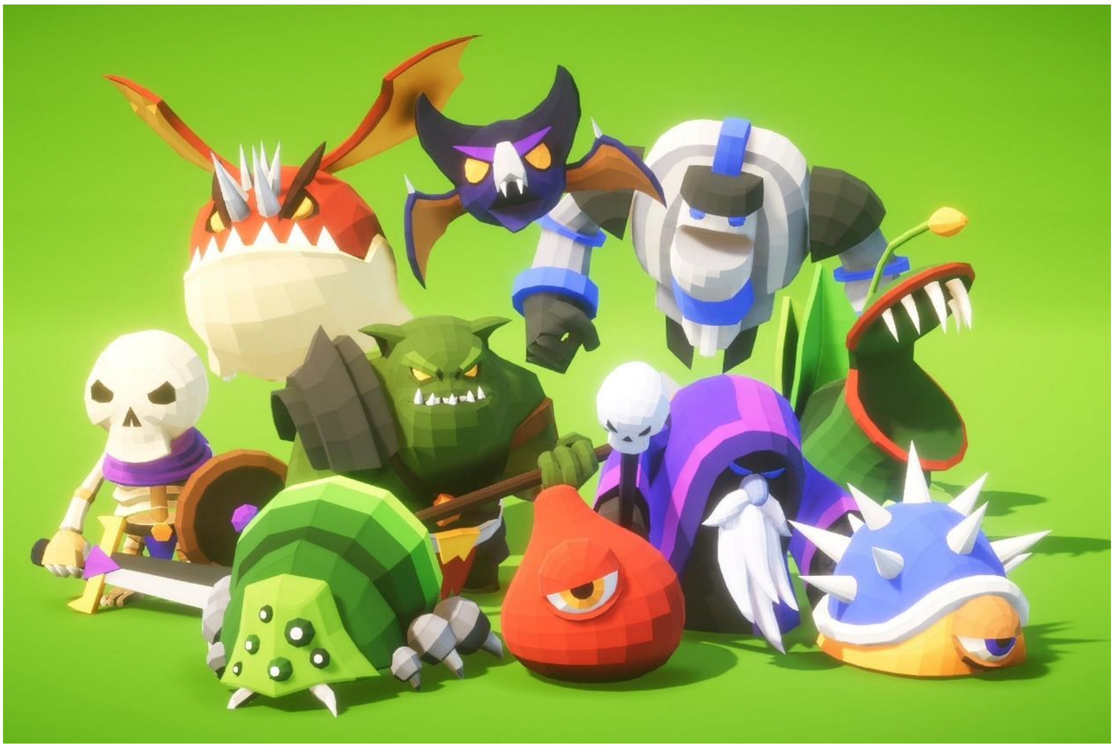


Figure 4: The monster prototypes under an ancient curse.

3. Gameplay Design

1. Game Genre

The " Legends of Animalia" combines elements of survival and roguelike shoot 'em up mechanisms to create an immersive and exciting gameplay experience. The game focuses on survival aspects, requiring players to manage their health while also protecting themselves from monsters. In addition to survival mechanics, the roguelike elements allow players to develop their animal character's attributes. They can explore the world and fight against the monsters in their own ways. The modern shoot 'em up like "Brotato" provides players with thrilling moments of combat, requiring precise aiming, dodging enemy attacks, and strategic positioning to overcome challenging encounters.

Figure 5: Some games of similar genres ("Party Animal" and "Brotato").

2. Gameplay Mechanics and Gameplay

1. Action

Player can move around the map to explore the forest and track or hide from the monsters. When facing the monsters, player can attack them with the small sword in hand. Here lists all the actions and relevant operation methods in the Table 1.

Control	Action
Key "W"	Player walks upwards
Key "A"	Player walks towards the left side
Key "S"	Player walks downwards
Key "D"	Player walks towards the right side
Key "Esc"	Pause or resume game and show menu
Mouse Left-Click	Interact with the menu items Player attacks

Table 1: Common actions and relevant control operations.

The main UI interactions are button clicking in the different menus. Here lists all the function of each button in the Table 2.

Button	Function
Start / Restart	Start a new game
Resume	Resume the paused game
Save	Save the current game status.
Load	Load the saved game status and continue the game
Menu	End the game and go back to main menu
Exit	Exit the game and go back to desktop

Table 2: Common buttons and relevant functions.

2. Mechanisms

This game is driven by survival time, where the player should pass all three monster waves in three minutes. For each wave, the monsters will be generated on the five generation points in sequence. The type of monster is determined by the random rates which differ among waves. Generally, the later wave has more fierce monsters with higher HP and attacks. However, there remains the possibility for player to face the threat of powerful monsters in the beginning, which may disturb the player's strategies. Player can also have growth in round to fight against the growing monsters. After slaying a monster, the player will gain EXP based on the type of the monster in amount. When the EXP bar is fulfilled, the player can increase their level to improve their HP, attack, and defense with an extra bonus of full HP. The cost is that the required EXP to level up increases as well. If the HP bar is back to zero, it means player is finally killed by monsters after resistance.

Outcome	Condition
Win	Pass the Wave 3 (3 minutes).
Lose	Player HP is empty.

Table 3: The win/lose conditions.

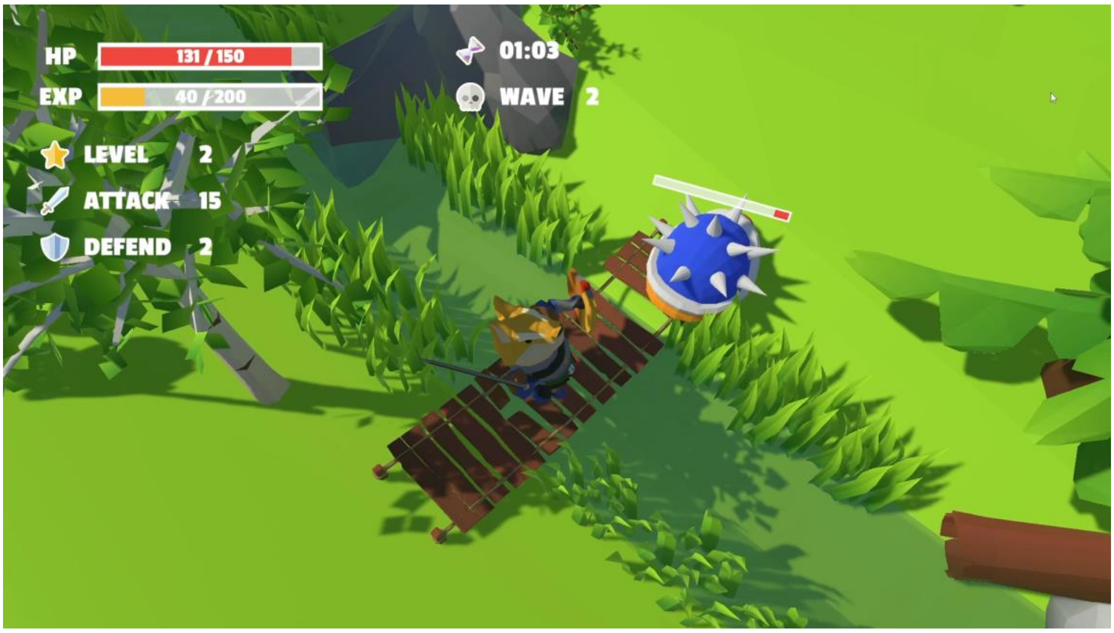


Figure 6: The game screenshot when player is attempting to attack the turtle.

3. Game Elements

1. Graphs

"Legends of Animalia" introduces a low-poly art style which is characterized by its use of simple geometric shapes and flat shading to represent objects and characters within the game world. The game also employs a rich color palette, featuring bold and vibrant hues. The colors are often flat and solid, creating a visually striking and eye-catching environment. In line with the low-poly style, details are kept minimalistic and straightforward, instead of intricate textures or fine details. This makes the game visually captivating and delightful for players of all ages.

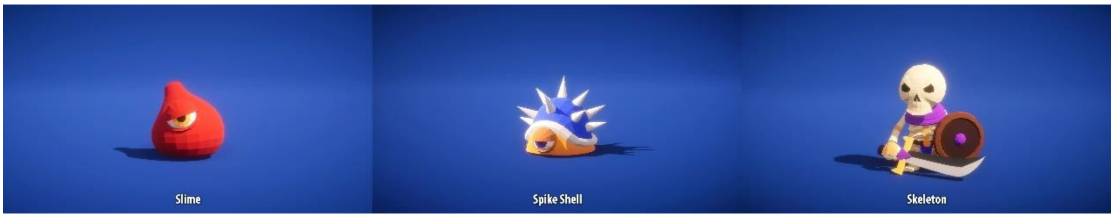


Figure 7: The models of the monsters (Slime, Turtle, and Skeleton).

2. Animation

The main animations focus on idle animation, locomotion animation, and combat animation which are the main actions in the game process. The character animations serve to bring the animal protagonists to life, enhancing their believability, personality, and immersion within the game world. The attention to detail in the animations adds an extra layer of depth and enjoyment, making the gameplay experience even more engaging and captivating for players. Furthermore, the combat animation influences the hit checks by placing colliders in the weapons and triggers in the frames.

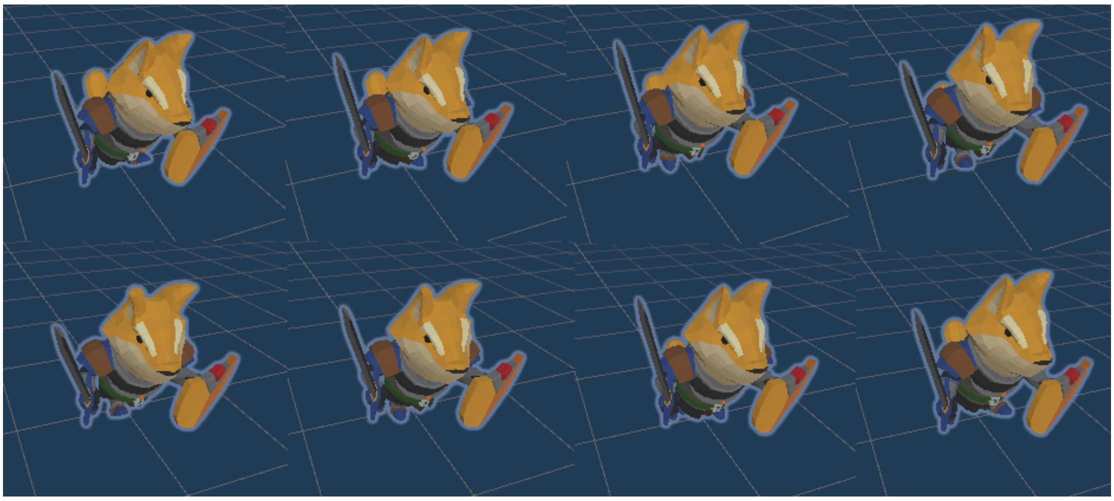


Figure 8: The animation clips "RunForwardBattle" of Dog Knight.

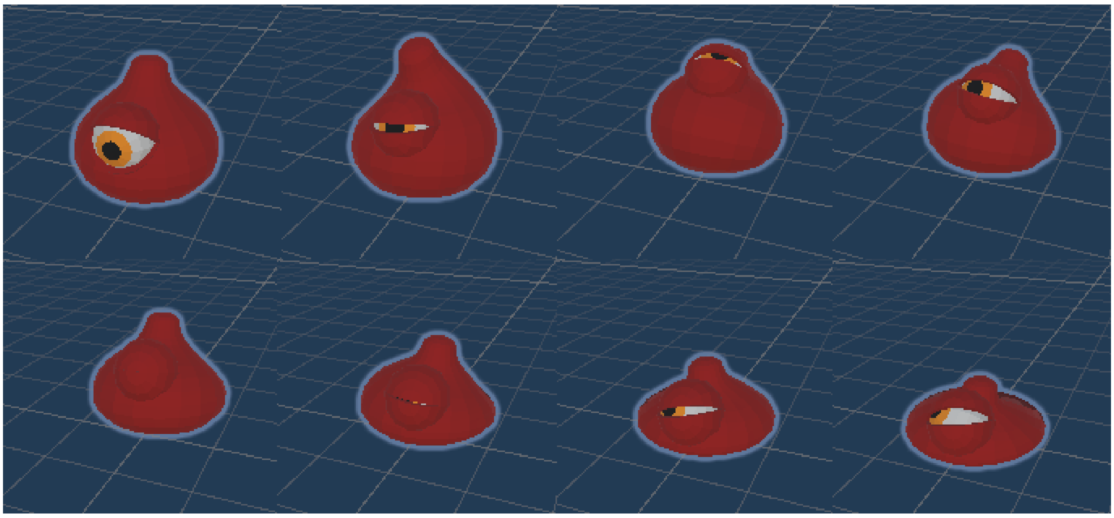


Figure 9: The animation clips "Die" of Slime.

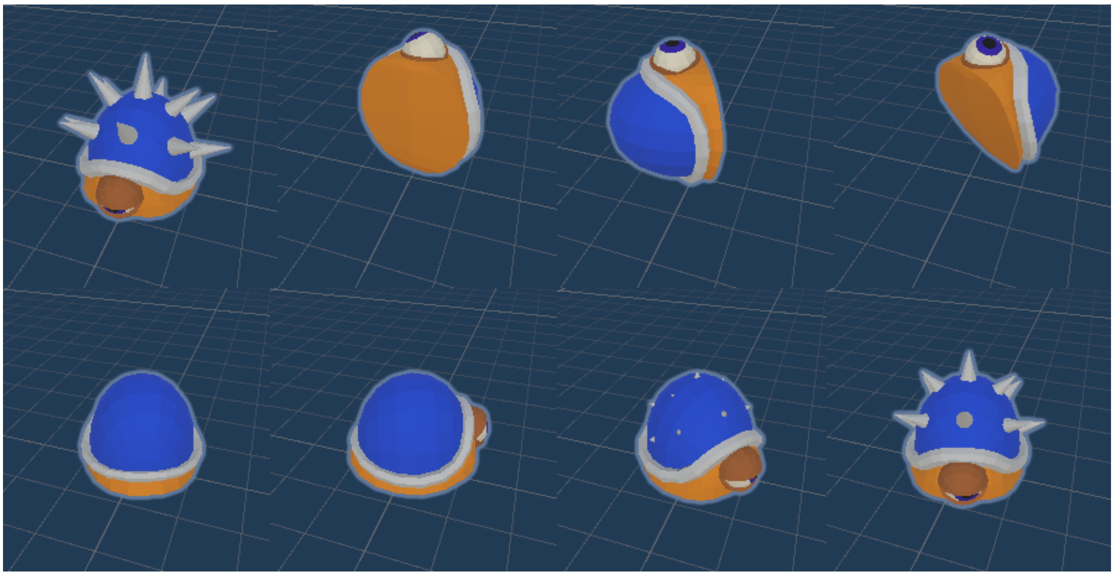


Figure 10: The animation clips "Victory" of Turtle.

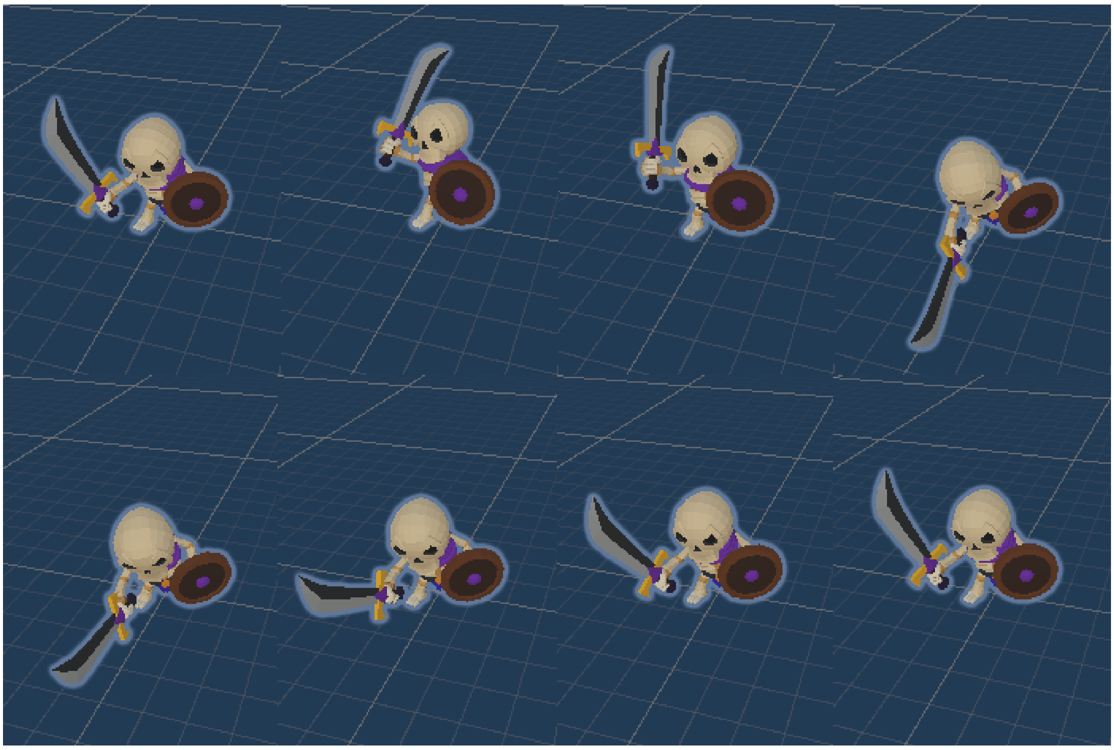


Figure 11: The animation clips "Attack01" of Skeleton.

3. Artificial Intelligence (AI)

This project applies navigation mesh (Nav Mesh) of Unity AI module to perform monsters' pathfinding and avoidance in complicated terrains. The underlying logic of monster is to track and follow the player to launch a melee attack, which is simple and useful to bring challenges to players. The implementation can be divided into two parts, Nav Mesh and Nav Mesh Agent. Nav Mesh is a simplified, three-dimensional representation of the game world's geometry, where obstacles and non-walkable areas are excluded. The Nav Mesh is generated and baked during the game's level design.

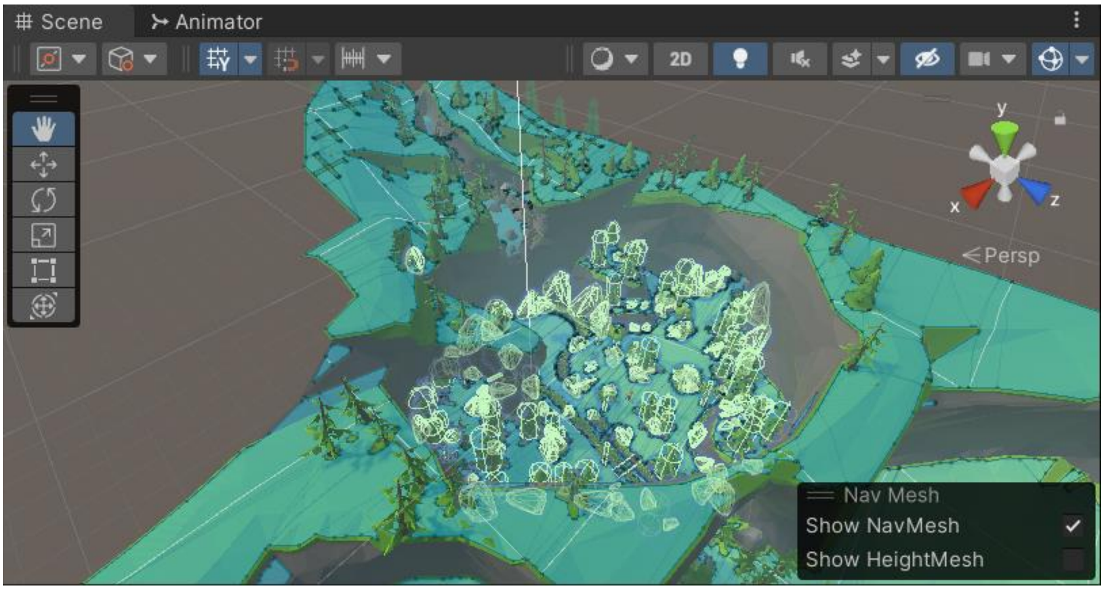
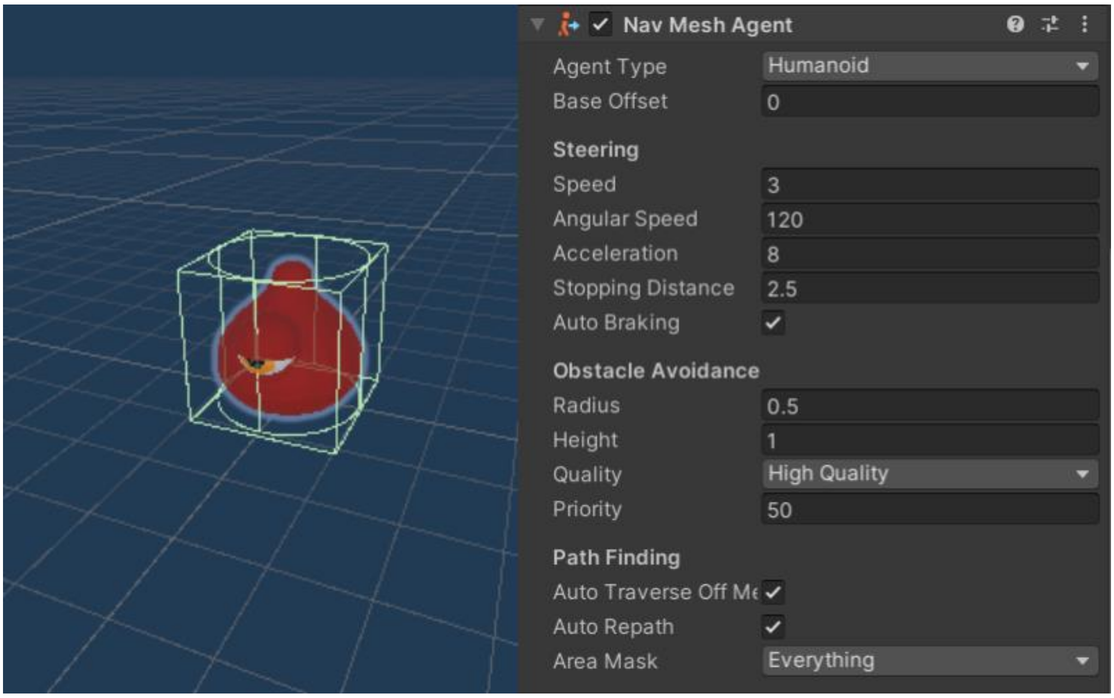


Figure 12: The Nav Mesh baking result of the whole terrain and objects inside it.

Nav Mesh Agent is a component that utilizes the Nav Mesh to control the movement of characters within the game world. This component also determines some kinematic properties of the rigid body, such as speed and acceleration. By adjustment of stopping distance, the monster would keep a distance between player and itself to leave enough space for further attacks. When staying at suitable range, the monsters would attempt to attack the player with short pre-swing.



Nav Mesh Agent

Agent Type: Humanoid
Base Offset: 0

Steering
Speed: 3
Angular Speed: 120
Acceleration: 8
Stopping Distance: 2.5
Auto Braking: ✓

Obstacle Avoidance
Radius: 0.5
Height: 1
Quality: High Quality
Priority: 50

Path Finding
Auto Traverse Off Me: ✓
Auto Repath: ✓
Area Mask: Everything

Figure 13: The "Nav Mesh Agent" configuration of Slime.

4. Sound

This game plays the background music "Jungle" during the whole playing process. This music is in a classical video game music style which fits the game art representation. It depicts a scene of walking in a forest or jungle with possible threaten underlying, which creates vibrant but tense battle atmosphere, during the playing.

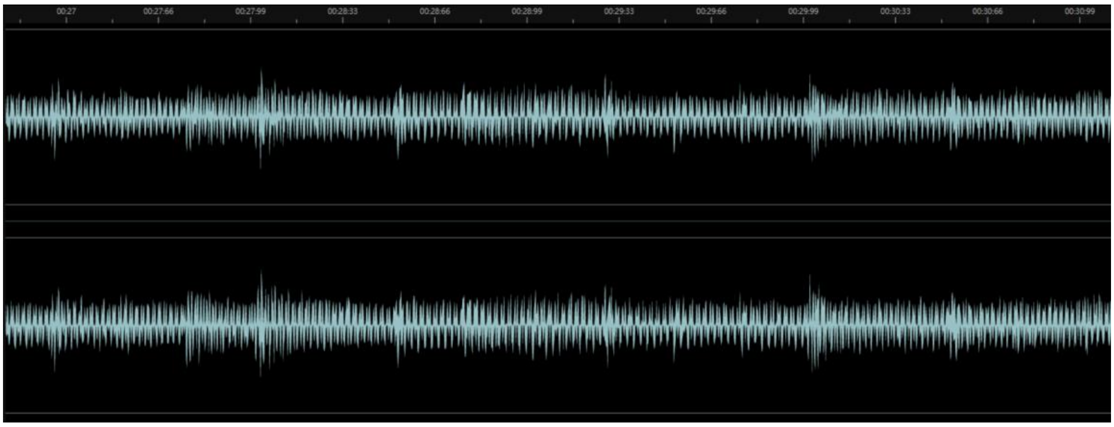


Figure 14: The frequency chart of a clip in the background music "Jungle".

4. Game Progression

At the beginning, the player is located at the center of the whole map with a full 100 HP and 10 attack points. The monsters will be generated randomly in five points to track and attack the player between 5 seconds intervals. By defeating them, the player would gain extra EXP for further leveling up to improve the attack and defense and be healed to full health. As time progresses, the monster wave should be more danger from slimes to skeleton soldiers. Player who can survive from the Wave 3 monsters be the winner of forest ambush.

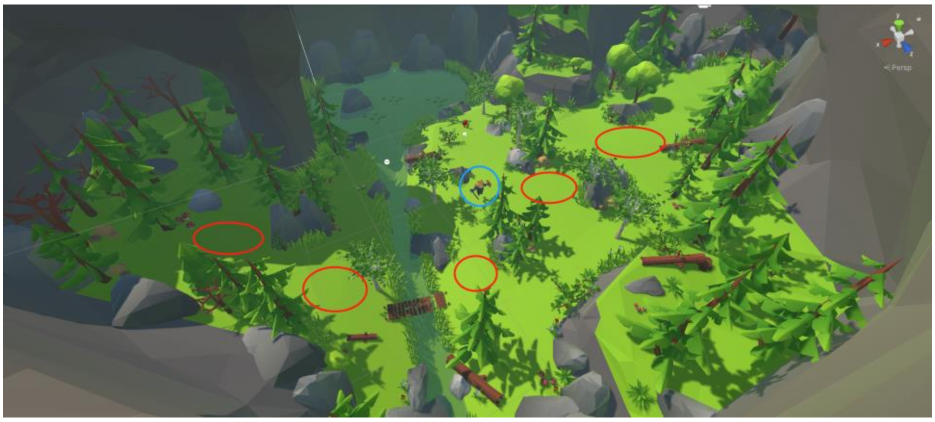


Figure 15: The generation point of player and monsters which are marked as blue and red.

5. User Interfaces

Considering the main art style tends to be low-poly style, here incorporate low-poly aesthetics into the UI elements to maintain visual consistency with the game's overall style. This project imports "Funny, clean cartoon GU / UI kit" in the Unity asset store for better visualization and interaction experience, including the panels, buttons, icons, and font. Generally, the main elements of this UI kit are consisted of angular shapes, flat shading, and limited color palettes.

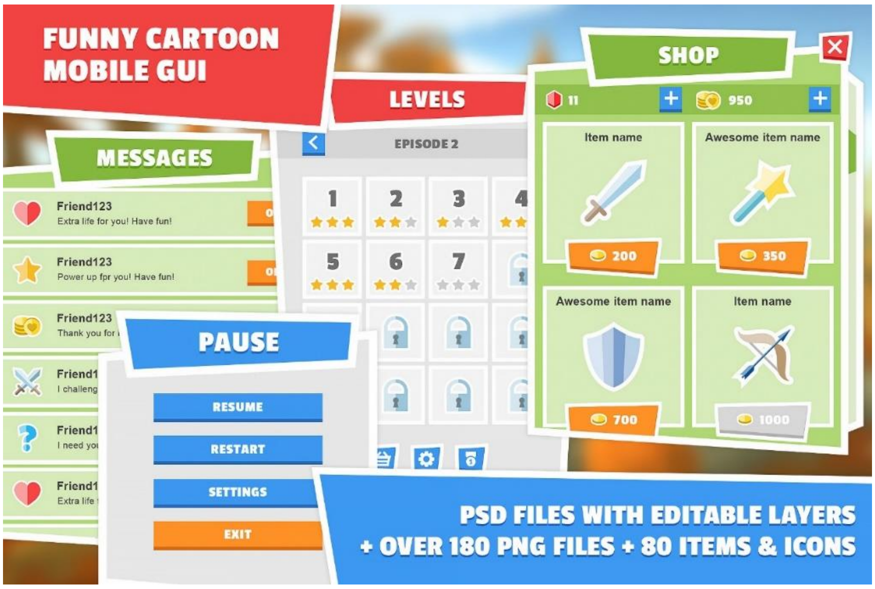


Figure 16: "Funny, clean cartoon GU / UI kit" in the Unity asset store.

The menus are modified from the same template, due to uniformed art style and familiar functionalities. The headers use different colors in background to differentiate from each other for users to identify the type of menu. All of them have the "Exit" option for users to exit the game.

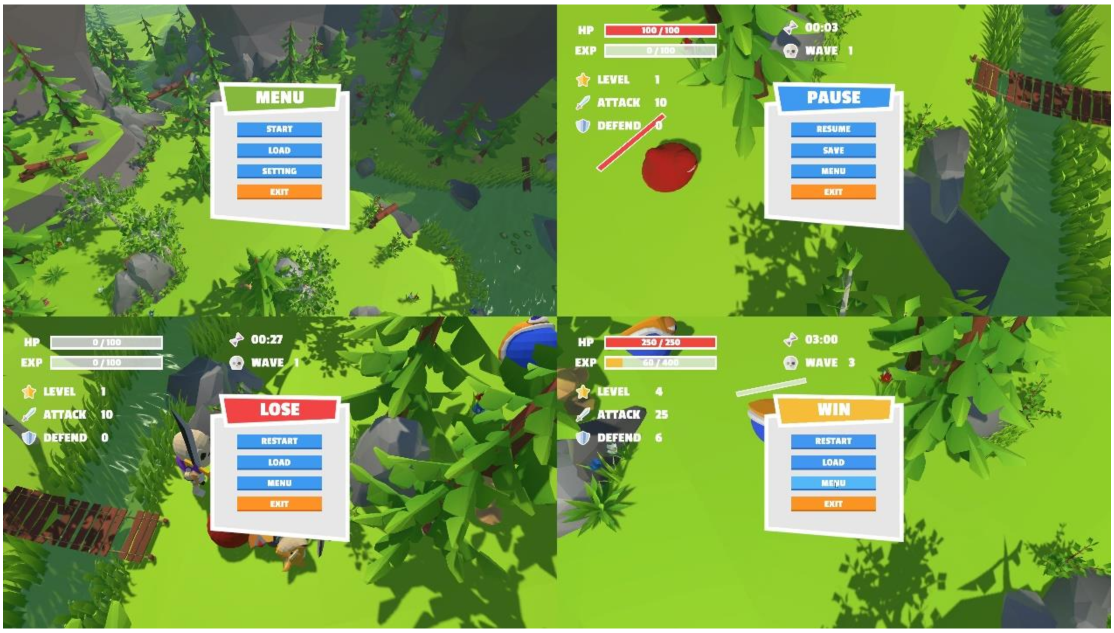


Figure 17: The interface of different menus (menu, pause, lose, and win).

For the in-game UIs, the main idea is to keep them as concise as possible in a way that the player can understand. Most of them are applied with the format of icon, label, content, such as level item shown in Figure 18. For player's HP and EXP, the filled bars, with exact current number and maximum number in it, provide a more visual representation of the player's current state. For monsters, the overhead simple bars make the scene more intuitive, which can follow with its body.

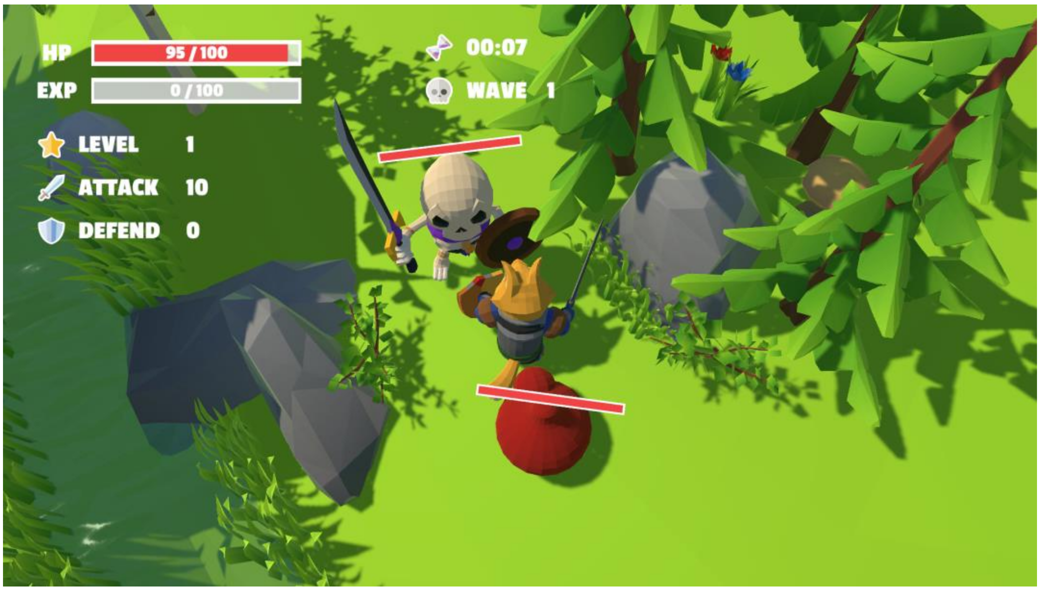


Figure 18: The in-game UI including game, player, and enemy information.

4. Implementation

This game is developed by Unity Engine and C# scripts. The UML diagram in the Figure 19 shows the main structure of the core components, which can be divided into six parts including game, camera, player, enemy, UI and save. Each part is managed by a related manager respectively, which is inherited from a Singleton pattern class to provide easy access from each other. Furthermore, the project introduces a "Property" generic class for data binding by delegating actions.

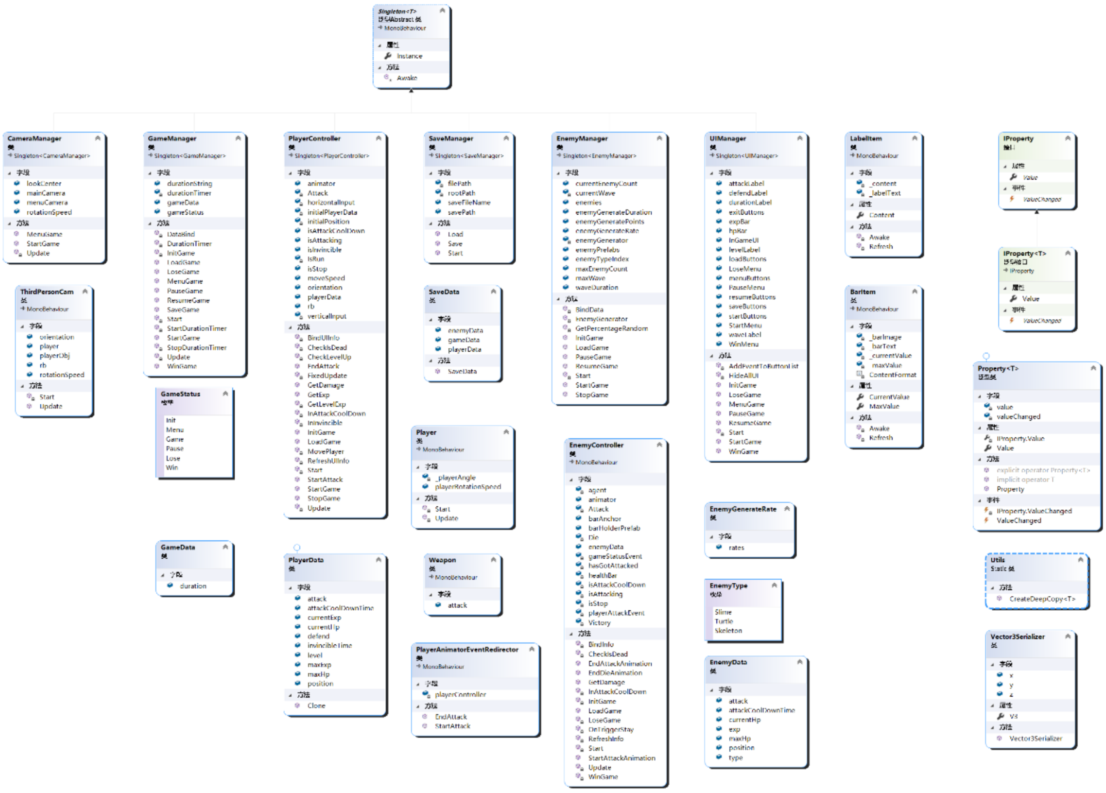


Figure 19: The UML diagram of the game structure.

1. Game Manager

Game manager determines the global game states such as in-menu, in-game, and paused, and provide global entrance functions for switching between states. It is clearer and easier to maintain, considering frequent external invocation and different strategies when switching. For example, the game starting is high relied on the "Start" button click events, which is implemented in UI manager, but player manager and enemy manager need to react to this event as well. Moreover, game manager stores some game parameters including game duration.

2. Camera Manager

This project utilizes "Cinemachine" to realize the camera following with player in a third-person top-down view. By editing "follow", "look at", and three types of Rigs, the main camera can keep moving in a stable perspective smoothly. The rounding camera displays the overview of the terrain in the menu background, which aims to familiarizing the player with the map.

3. Player Manager

Player manager defines the basic attributes of the player in the player data class, including maximum and current HP and EXP, level, attack, and defense. Through "Property" classes, these data can be easily bind to the in-game UI for refreshing. This manager also contains the player movement and attacking controls, which are depended on multiple animator triggers and intermediate variables.

4. Enemy Manager

Enemy module is consisted of enemy manager for all enemies and enemy controller for single monster. The enemy manager provides inspector fields to manage the detailed information of monsters in each wave. For instance, there is a 60% chance a monster in wave one that it is a Slime, a 30% chance that it is a turtle, and the remaining 10% is a skeleton. They will be generated in the five generation points in sequence through 5 seconds generation duration. The data can be easily modified for better balance fix without touching the code part. The enemy controller gives a monster the complete set of behavioral logic. They will attack the player or die immediately at the right moment, which are also powered by their relevant animator triggers and parameters. Monsters have their own values stored in the enemy data class, such as HP, attack, or EXP the player gained after being killed.

5. UI Manager

Due to data binding in "Property" and main entrance function in game manager, UI manager only declares the references to different UI items. These UI items can be seen as black box wrapped by predefined components, such as assigning content in "LabelItem" or max/current value in "BarItem". For menu part, there are multiple buttons in different menu calling the same function. The "AddEventToButtonList" function adds events in button batches referred in the inspector.

6. Save Manager

As all the data are serializable including "Property" and custom "Vector3" type, it is simple to stringify or parse the game data in the format of JSON by integrating game data, player data, and enemy data list into a "SaveData" class. The deep copy function based on the binary formatter supports the overwriting original data by a new one. What needs to mention is that the serialization transformation factually ignores and clears data binding action in the "Property", which require reasonable re-binding or unbinding.

5. Ethical and Social Issues

1. Animal Welfare and Conservation

As the game revolves around animals and their adventures, it's crucial to address the ethical treatment of animals. Developers should ensure that the game promotes respect for wildlife and portrays responsible interactions with animals. Additionally, highlighting conservation efforts and raising awareness about environmental issues can be a positive aspect of the game.

2. Cultural Sensitivity

The portrayal of different cultures and their representation in the game should be approached with sensitivity and respect. It's important to avoid perpetuating stereotypes or engaging in cultural appropriation. Consulting with cultural experts or diverse perspectives can help navigate potential challenges and ensure a more inclusive and respectful portrayal.

3. Violence and Aggression

The game's combat mechanics and interactions involving violence should be carefully designed to strike a balance between entertainment and ethical considerations. Developers should consider the impact of violence in the game, particularly in relation to the portrayal of animal aggression, and take measures to mitigate any negative effects.

4. Inclusivity and Diversity

It is essential to consider the representation of diverse characters and perspectives within the game. Ensuring that players can relate to and identify with the animal protagonists, as well as other characters, regardless of gender, race, or other identities, promotes inclusivity and broadens the game's appeal.

6. Limitations and Future Work

1. Limitations

Although this game prototype provides the main framework of the game design, there are still lack of abundant game elements to fulfilled the game process, such as new characters, new monsters and new scenes. For animation, expressiveness and sense of combat become the main drawbacks of the combat system, as player and monsters have no hit animation and back-swing to simulate the realistic conditions. For level design, the game win/lose conditions are much ambiguous to stimulate the player for exploration and combat. The player has limited means to enhance their roles or make more personalized choices.

2. Future Work

1. Equipment and Prop System

The equipment system allows players to acquire and equip various items that provide benefits or enhance the abilities of their animal protagonists. These items can include armor, weapons, accessories, and utility tools. Each piece of equipment may offer stat bonuses, special abilities, or unique effects that influence gameplay. The prop system allows players to interact with the game world by using or manipulating various props. Props are objects found within the environment that can serve different purposes, such as solving puzzles, unlocking secret areas, or aiding in survival.

2. Skill System

The skill system allows players to develop and improve their animal protagonist's abilities over time. As players gain experience points through various actions like exploration, combat, and completing quests, they earn skill points that can be invested in different skill trees or categories. These skill trees may include combat skills, survival skills, social skills, and exploration skills. Each skill tree focuses on specific gameplay aspects and offers a unique set of abilities and bonuses.

3. Boss System

Bosses are formidable adversaries that stand out from regular enemies in terms of their size, strength, and abilities. Each boss possesses unique characteristics, visual design, and combat mechanics that make them memorable and engaging encounters. Bosses typically follow a progressive difficulty curve, becoming increasingly challenging as players advance in the game. Each boss encounter may require players to learn and adapt to new attack patterns, mechanics, or phases. Defeating bosses yields substantial rewards, including rare loot, powerful equipment, unique abilities, or significant story advancements. Boss battles often mark key milestones in the game's narrative and character progression, making them pivotal moments of achievement and progression.