COMP4801 Final Year Project

Project Plan for
The Road to Castle 3D

Supervisor:

Dr. T. W. Chim

Group Member(s):

Leung Chiu Yuen (3035857294)

Ko King Nam (3035864297)

Table of Content

1. Project Background 3
2. Project Objective 4
3. Project Methodology 6
   3.1 Integrated Development Environment 6
   3.2 Blender 6
   3.3 Unity and Its Built-in Systems 6
4. Project Schedule and Milestones 8

1. Project Background

In recent years, game development has become more accessible to individual game developers. Various tools and game engines, such as Unity and Unreal Engine 5, are now available to create immersive game experiences. Moreover, the game industry has witnessed tremendous growth in the market, along with the success of classic fantasy role-playing games (RPGs), like Runescape, Baldur's Gate, and Monster Hunter. We are greatly inspired by these masterpieces and are motivated to create our own game for this project.

There are several goals to be achieved in this project. We want to imitate the game design principles behind those masterpieces, and to rework our previous game project, The Road to Castle (https://github.com/pystander/ENGG1340_Gp106_RTC). Programming skills and data structures we learnt in school can also be applied in the project to enhance game performance and code organization. Furthermore, the project serves as a great way for us to gain a deeper understanding of game development, as well as some new AI tools for game development, such as voice overs, image generation, role-playing, and even in the development process itself.

Game development requires strong coordination and teamwork among all members in different roles. For instance, a developer needs to routinely catch up with the progress made by others, communicate with the designers, and to implement new features under the plan. This would be a valuable opportunity to build our communication and collaboration skills, by utilizing platforms like GitHub.

Besides collaboration skills, this is also a valuable opportunity to expand our skill sets for game development, such as programming, UI/UX design, computer graphics and animation, and 3D modelling. These technical skills and the project itself could be an impressive addition to our portfolio, showcasing our technical expertise to potential employers in the game industry.

2. Project Objective

The main purpose of our game is to provide players with a captivating RPG experience. This would be accomplished by developing various gameplay systems, such as combat and level systems. We hope to learn about the workflow, and to gain hands-on experience on game engines and development tools.

There are 9 objectives (including game features) to be achieved in this project:

1. **Game storyline and tutorials**

   An engaging and compelling storyline will be created for the game. The game tutorials will be integrated into the main storyline, allowing players to learn as they progress through the game.

2. **User-friendly user interface**

   The user interface (UI) is designed to be intuitive and easy to navigate. This ensures that players can focus on the game itself without being distracted or confused.

3. **Well-designed maps**

   Maps are separated into different parts, with diverse weather and eco-systems. For instance, there would be wolves in snowy forest. This adds variety and depth to the game world, making it more immersive and engaging.

4. **Inventory system**

   There would be an inventory system for players to manage their own resources effectively.

5. **Balanced combat system**

   The combat mechanism plays a crucial role in this game, which is designed to be challenging and yet not overwhelming to the players. For example, it may involve some reasonably complicated controls for certain actions in the battle.

6. **State management and serialization**

   There would be save and load functions for players to resume back to where they left off, either by local or over the network. For instance, we could store the serialized data files on AWS or Azure. This ensures that players can access their saves when they switch their devices or locations.

7. **Level system**

   Level is considered an important player development mechanics in RPG. When characters defeat enemies and complete quests, they will be rewarded with experience points (EXP). When the accumulated EXP reaches a certain limit, the character will level up, with increases in attributes and abilities.

8. **Dungeon editor**

   We want players to express their creativity by designing their own dungeons, so that they can share with their friends. In this game, a dungeon editor is included to create custom levels with various enemies, terrains, and events.

9. **Software integration**

   Different software like Unity and Blender will be synergized in this project. For instance, 3D models built in Blender will be imported to Unity.

3. Project Methodology

The project will be built with Unity, Blender, and any IDE (integrated development environment) of our choice. The game will be mainly developed in Unity, where we integrate all the resources, for example, models and sounds, into the game. It is also where we design and implement the logic of the system. And the most important of all, it can create cross-platform games.

3.1 Integrated Development Environment

Integrated development environment (IDE) is the tool where we implement the logic for the game. Since we have chosen to use Unity as our game engine, C# will be the programming language we will be working with. This means that we can use Visual Studio Code or Visual Studio Community as our IDE to start our game development. Using a specific IDE that we are comfortable with is the key.

3.2 Blender

Blender provides a lot of functionality in 3D modeling, animation, shading and visual effects creation. This time, we will be using it to create 3D models and animations. Visual effects will also be created for animations like sword animation. Since our 3D virtual world is set in the medieval age, we are inclined to create medieval age style models like houses, bridges, castles, characters, swords and so on. For animations, we will need to create animations for walking, running, climbing, sword swinging and so on.

3.3 Unity and Its Built-in Systems

Unity provides us with a wide variety of built-in systems, which we will leverage to build our RPG game.

The first one is the Entity Component System (ECS). We will apply logic scripts to the game objects and allow some of the objects to hold RPG game specific attributes and logics. Besides adding scripts to game objects. Other built-in components such as a camera and a mesh renderer can also be added to the objects to allow them to perform a specific role. Many of the RPG game systems or logic, for example, inventory, combat,

leveling system and so on, will be built with this Entity Component System. This is also the main system used to power our Dungeon Level Editor system where players can create their own dungeon levels and share it to other players to play with.

The physics system is another system already built into the game engine itself. Many physics related mathematics are done by system. The objective here is to build on top of the system by adjusting and configuring system parameters to create a working and enticing 3D RPG game hopefully within the 2 semesters.

The audio system will also be used in the project to play medieval age style background music. Battle music will also be played once the play is engaged in combat. Music will be composed by AI or bought from the asset store. Our asset store policy will be mentioned in the last paragraph of this section.

There will also be a networking system, either to be designed by us or provided by API from Unity's built-in package. This system allows us to create multiplayer games for the game. Currently, we will use the system to power our dungeon stage. Specifically, the dungeon can be played in single player or multiplayer mode. Dungeons designed by players can also be shared by leveraging the networking system.

An animation system provided by Unity will also be utilized. Animations created from Blender to be imported into Unity to animate character movements such as walking, running, climbing, sword swinging and so on. Since animals will exist in the game, animations for animals will also be created and run on Unity.

4. Project Schedule and Milestones

Objectives	Deadline
Deliverables of Phase 1 (Inception) • Detailed project plan • Project webpage	1 October, 2023
Minimum Viable Project Development • Inventory System • Combat System • Level System • Create 3D models • Basic Map Design	31 December, 2023
First Presentation	8-12 January, 2024
Deliverables of Phase 2 (Elaboration) • Preliminary implementation • Detailed interim report	21 January, 2024
Final Project Development • Create Dungeon Level Editor • Create models for characters • Create animation	15 March, 2024
Game Testing and Adjustments	25 March, 2024
Final Presentation	15-19 April, 2024
Deliverables of Phase 3 (Construction) • Finalized tested implementation • Final report	23 April, 2024
Project Exhibition	26 April, 2024