EECS 581 Project Proposal Report

Team 4

Members

Siming Fu, Xiaoyi Lu, Fute Luo, Shujing Wu, Fengming Yang

Project Name

ESCAPE

Project Synopsis

Our team is going to make a 2D entertaining puzzle game which is built by unity and C#.

Project Description

- Why is the project being undertaken?

Computer games are common in modern society. A wonderful game can bring people happiness. Most of our team members are full of love for the game, developing a computer game is something we are interested in and want to try, and has the opportunity to bring happiness to people.

- Describe an opportunity or problem that the project is to address.

For a team without game development experience, how to cooperate well and how to accomplish the goal together is the hardest part. First, every team member needs to be familiar with the compiled language (C#) and learn to use Unity 2D alone, which will reduce the communication between teams, and lacked the initial game design planning. A successful game development team is often planned, and some team members will complete the framework to make the cooperation between team members easy. These are not available in our team, but at the same time, they are also opportunities for the team to learn these insufficient skills.

- What will be the end result of the project?

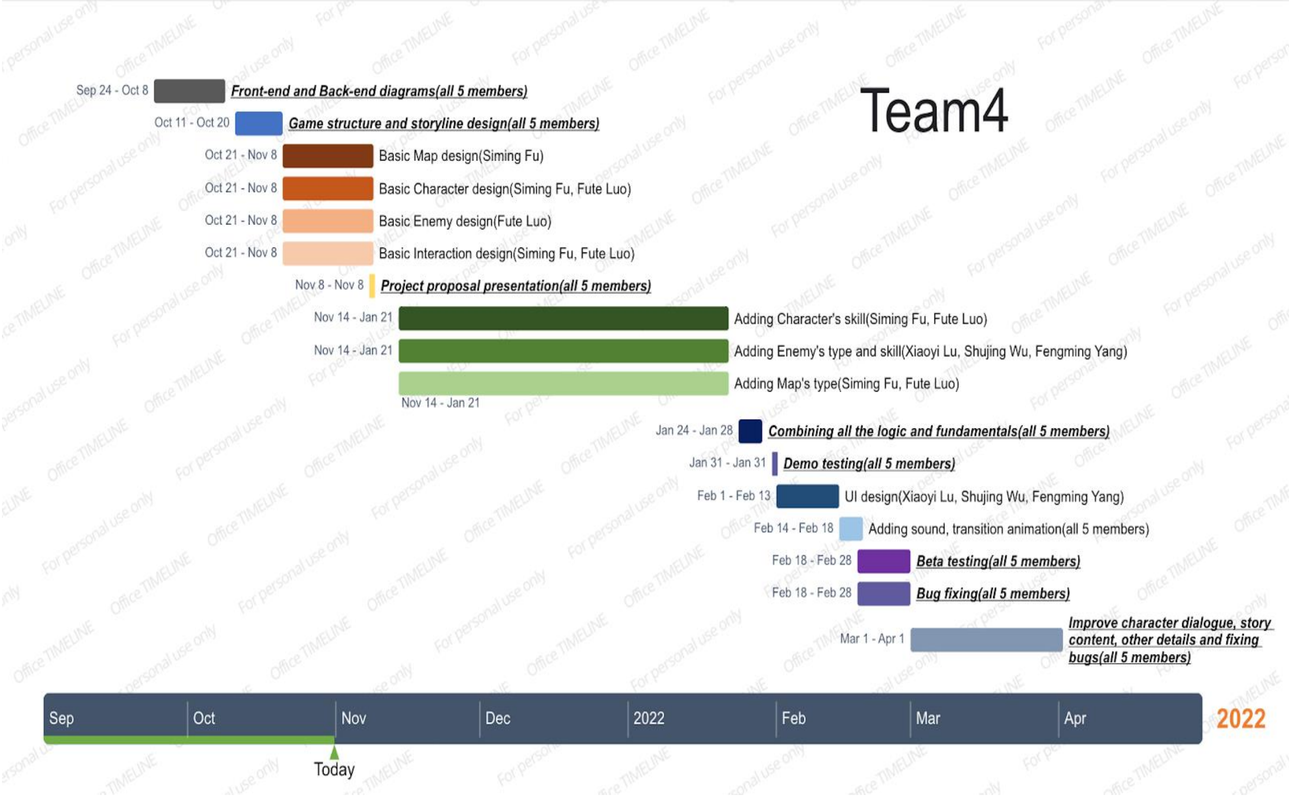
The end result of the project is a complete 2D roguelike game that tells a complete story. Players can operate the protagonist to experience, pass the levels after defeating the monsters, then randomly go to a map that hasn’t been to, and finally defeat the boss to win.

Project Milestones

First Semester:	Due Date:
1. Front-end and Back-end diagrams completed	(10/08)
2. Game structure design	(10/20)
3. Basic Map, Character, Enemy, Interaction design	(11/08)
4. Improve Map, Character, Enemy’s types and skills	(12/20)

Second Semester:	Due Date:
1. Improve Map, Character, Enemy’s type, and skill	(01/21)
2. Combining all parts	(01/28)
3. UI design	(02/13)
4. Adding sound and animation	(02/18)
5. Beta testing	(02/28)
6. Bug fixing	(02/28)
7. Improving	(04/01)

Gantt Chart



Project Budget

- Hardware, software, and/or computing resources
  - Unity 2019.4.10f1 LTS (free) (Fall and Spring semester)
  - Unity assets (depending on the amount of time for asset development) (Fall and Spring semester)
- Graphics and assets development
  - Art, music, voice acting (Spring semester)
- Pluralsight Training Tools

- Estimated cost: $400-800

Preliminary Project Design

ESCAPE will be a subgenre of role-playing video games characterized by a dungeon crawl through procedurally generated levels, turn-based gameplay, grid-based movement, and the permanent death of the player character. During our research, we found several high-quality roguelike games in Steam and had much of a good rate in those games' feedback. At all, our game will correspond with video games and sample interactive storylines.

The software we created will be an application on Steam and may add controller adaptation to the PS platform or Xbox platform. The basic organization of the game is that when a player accesses the main menu, there are three buttons that the player can choose, which are "Start Game", "Setting", and "Help". First, when he chooses "Help" it will show the keyboard button that allows the player to play the game. Second, when he chooses "Setting", it will show the sound control and language change.

Right now, we don't allow you to change keyboard buttons, which may change in future updates. Finally, when the player chooses "Start Game", as Figure 1 presented, players will enter a random map and they need to find the exit door at each floor in order to climb up to a higher-level floor. If players die before meeting the boss in the fifth level, the interface of the game failure will be displayed and then the system will ask players if they want to continue the game. If not, players will go back to the main menu for quitting the game. If players would like to start again, they will go back to the main level floor (first level floor) and play another random map, and continue climbing up the floor. If he stays on the fifth floor and finishes all the enemy and finds the exit door, he will be against the final boss. If he wins that boss, he will trigger the final CG of this game, and successfully escape the room, and back to the main menu, if he loses that boss, he will trigger the lost CG and also back to the main menu. On each floor, players can find the skills and tools that increase character ability and get some medkit. Skills can increase such as a player's attack rate, and change his attack way, or speed up his speed, or increase his defense. Tools are put on character bags and allow the player to use them at any time, such as medkit, attack potion, or defense potion. Those skills and tools can be gathered from enemies, treasure, and small bosses. Even if you are moving to a special spot, you will surprisingly gather skills and tools. In Figure 2, we can see that there are five types of maps designed on each floor and each floor will have distinct characteristics for the enemy and small boss.

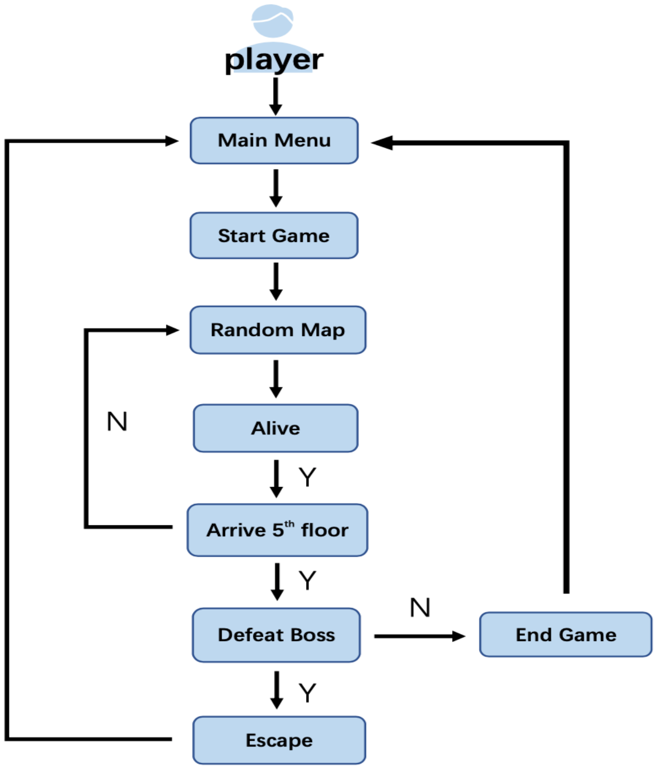


Figure 1. The Flow Chart of the Game

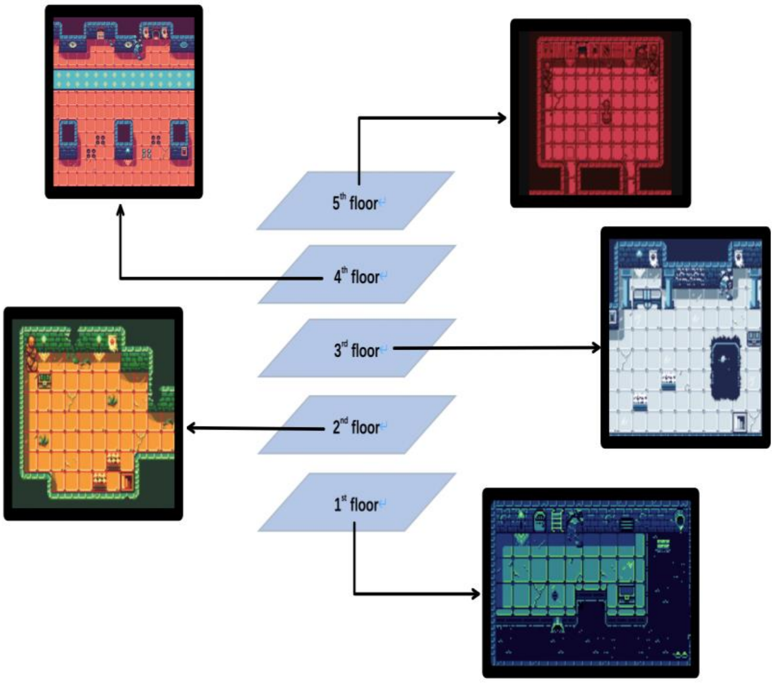


Figure 2. The Designed Map of the Game

Every time players defeat an enemy or boss, they can get a certain amount of bonus points, which can be used in the game to buy new weapons or upgrade existing weapons. These points can be inherited as well, which means that players will have the opportunity to obtain more advanced weapons before entering the map so that it increases the opportunity of successfully defeating the enemy and boss. In addition, in order to increase the difficulty, we will set a power bar for the player. When there is no power, the player cannot attack but can dodge. The power will be naturally recovered at a slower rate or through potions. Therefore, players need to plan to use their power and supplies to maximize their gains and then escape successfully. We are also putting some bonus scenes in-game that players don't have to against the final boss but still can escape the room, we call it "God Escape", and let players find that secret in order to trigger a special CG ending.

The design constraints in our team can be included in the technical and budget.
Technical constraints:
Operating system supported, software and programming language:

Our team has no experience in game development, so the choice of language and compiler is a technical constraint. We finally decide to use Unity2D and C# as IDE, and the unified use of Unity is 2020.3.21f1LTS (free) version as a compilation. Additionally, the keyboard operation is required in our game design, and it must work on Windows, Linux, or macOS and cannot support on mobile platforms.

Framework:

In the early of development, our team chose to use common open-source game frameworks to design our product but found that these frameworks are too bloated (most of these frameworks are suitable for 3D games, but our team develop a 2D game), which is not very adapt to our game design. As far as the design requirements of our 2D roguelike games are concerned, a lightweight game framework could reduce a lot of our work, and it is also convenient for post-production modifications. Figure.3, is the game architecture diagram we designed. This architecture can only be used to obtain the UI of all game pages, switch various scenes, control audio, obtain and modify all data, moving shot, and battle system. However, in this framework, there is no online feature. In other words, this game does not have in-game updates (If we have new features or story plot update, we can only adapt by downloading the update package) and cannot design multiple players, which is a design constraint. The battle system needs (In Figure.4) to meet the interactive functions between characters, and the specific implementation also requires some special requirements. We simplify the design of the combat framework into AI management, skill management, and event management. The entire game framework cannot be used in Figure4. Battle Architecture Diagram outside of Unity because it is based on the Unity platform.

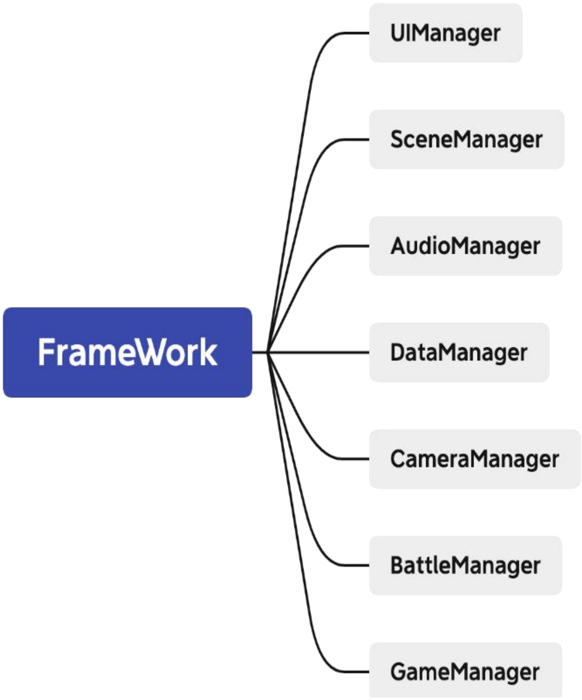


Figure3. Game Architecture Diagram

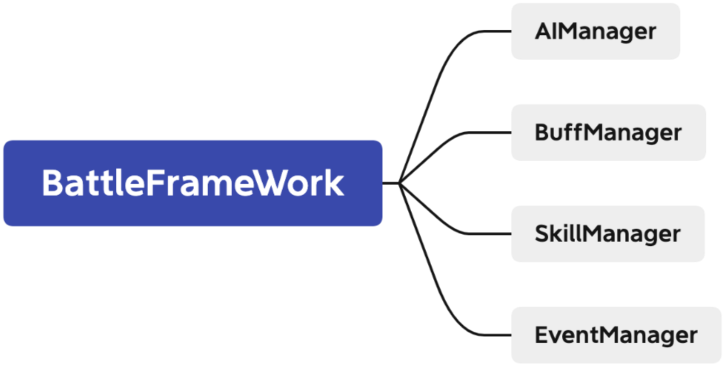


Figure 4. Battle Architecture Diagram

Budget constraints:

Moet game development requires art, music, and game dubbing support. Because of our budget, most of the design materials are free to use or simply made by ourselves. This game may not have exquisite art design and can only meet basic art needs.

Ethical Issues

1) Data privacy issue

When we do research and build games, we must consider the issue of protecting players' game data and personal privacy. We must proceed accurately and ask appropriate questions and permissions about privacy protection to protect each of our players' data. We will plan everything carefully so that there are no issues when getting feedback from the user's testers. In addition, if we conduct any formal real-time user testing, we will create a user testing agreement and a privacy confidential account so that users and game players can fully understand the privacy requirements related to participating in the game.

2) Violence and blooding scene issue

Another ethical issue is the problem of violence and blood on the mental health of young people. Due to the nature of our game, the main story in the game is about fighting monsters and escaping the map. We recommend young players and users participate in our game. And our game screen style is pixel style, which greatly reduces the visual effects of violence and blood so that our users can better experience the story of the game and enjoy the story of the game. We will determine the user and the age of the game player before starting the game, to avoid harming the mental health of young gamers.

Intellectual Property Issues

In the process of making games, it is indispensable for us to collect a large amount of material, and there will inevitably be related problems related to intellectual property rights. To avoid the risk of infringing intellectual property rights as much as possible, we double-check the assets are free for commercial and personal uses to build our games. An intellectual property issue is that all art or music we include in the game must be used fairly in our game. In addition, if we decide to charge for apps in the app store, then art and music must be used fairly in games for profit. And we will protect the game from unauthorized use after the game is completed.

Change Log

- Changing in Milestone
    - Separating the Map, Character, Enemy design into two-part

      After we began to develop the project, we realized that the component for designing map, character, and enemy is much more than we thought before. Therefore, we decided to develop the base in the first semester and then add other components in the second semester.

    - Moving the UI design to the second semester

The style of the game’s beginning interface required more time to determine and design, so we decided to implement the UI design in the second semester.