**aMAZE**

**Game Report**

**By:**

**Goh Zheng Yuan (Darrus)**

**Lee Yu Xian (Shawn)**

**Chuan Guang Zhe**

Table of Contents

**Introduction2**

Game Idea/Concept2

Unique Selling Point2

Project Schedule3

Planned Timing4

Actual Timing4

**Features5**

Custom Maze Maker5

FOE5

High-score System6

Multiple Key and Door system6

Dynamic Flooring6

**Knowledge Applied**7

C++ Lessons7

Principle of Game Design7

**Challenges**8

Challenges encountered8

Things we’ve learnt8

**Future enhancements**9

**Introduction**

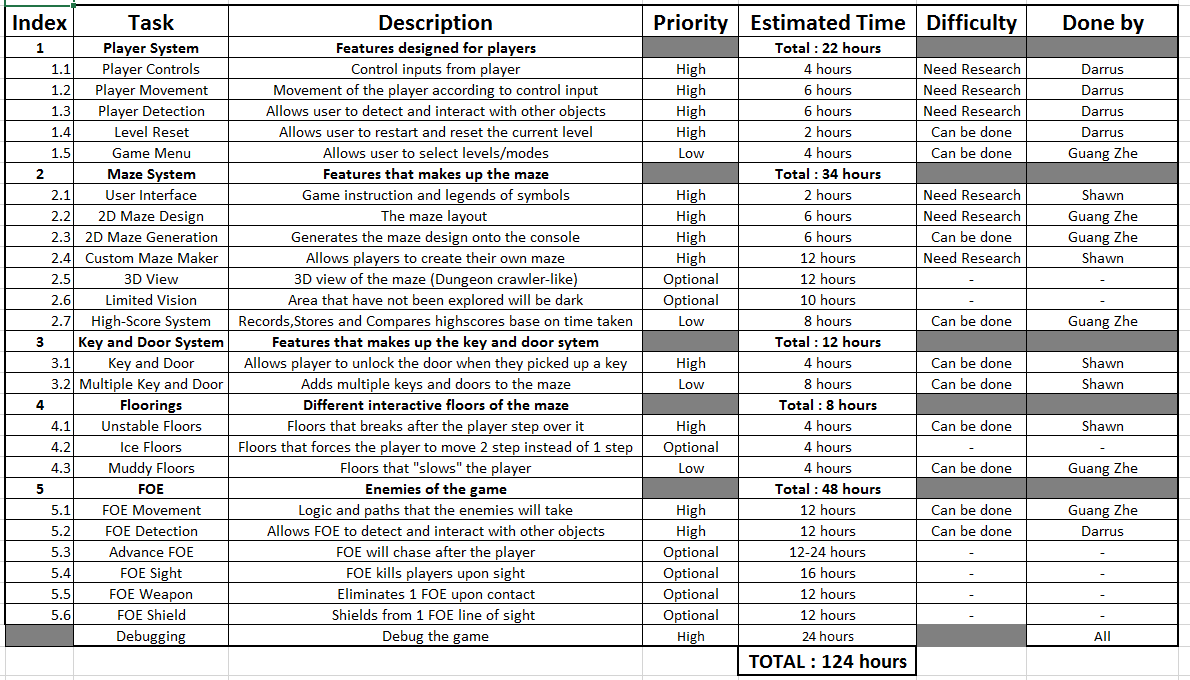
**Game Idea/Concept:**

Our game is a 2D maze exploration game with different types of level with each level having a different type of puzzle element. Players are to solve the maze while avoiding FOE and traps in the fastest time possible. We wanted to make a game that forces players to think carefully before taking each step hence the presence of floor traps and FOE.

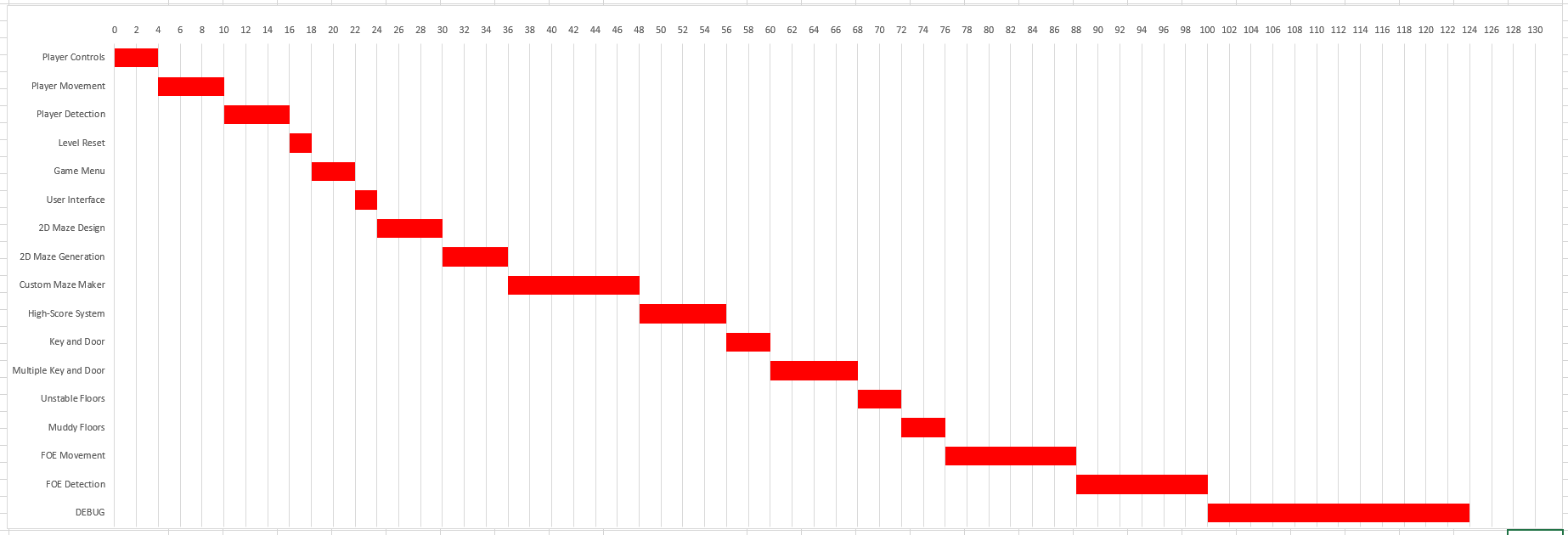
**Unique Selling Point:**

* Maze Customizer
* FOE
* High-score system, so that players will feel the challenge in trying to be the best.
* Traps
* Challenging levels

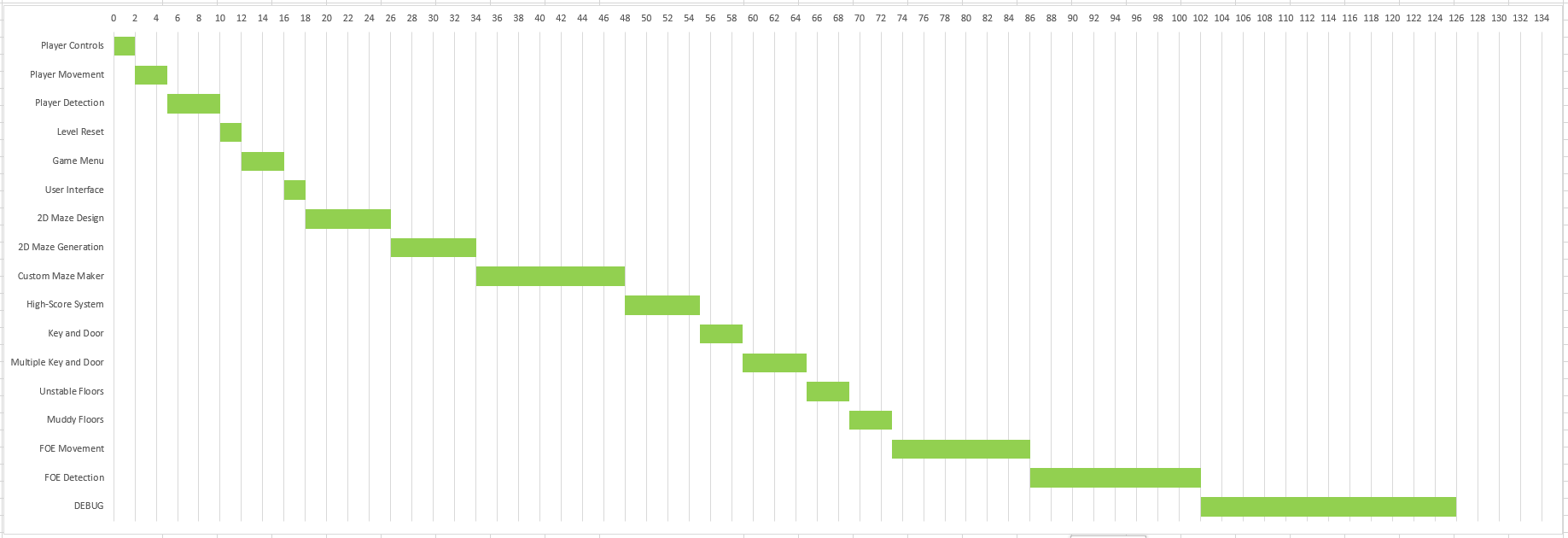
**Project schedule:**

****

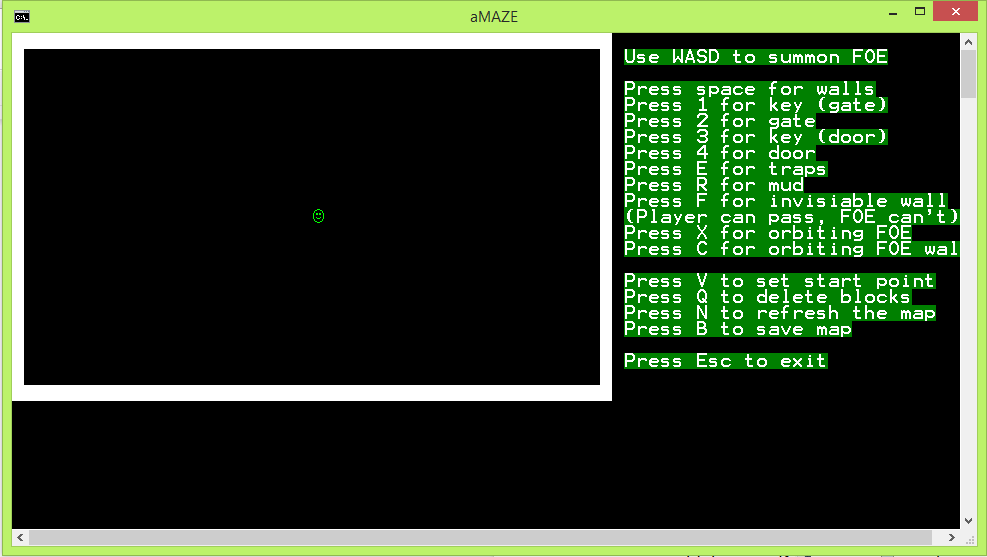
**Planned Timing:**



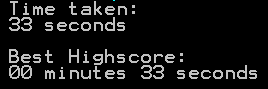
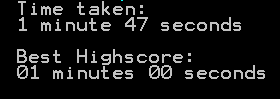
**Actual Timing:**



**Features**

* **Custom Maze Maker**
  + Players can use the in-game custom maze maker to create their very own maze. There is 3 save slot for players to save their creation in and all of them can be playable by selecting them in the level select menu. There is an user interface to show all the controls to let players create their very own mazes.
* **FOE** 
  + The game features enemies named FOE. They move one step for each step the player takes. There are a total of three types of FOE: Horizontally moving, Vertically moving and Orbiting. Players who walks into or in front of the FOE will be killed thus failing the level.



* **High-score System**
  + The game features a high-score system which saves the fastest clearing time in a text file for each level. It will automatically compare the player’s clearing time with the saved high-score and determine which one is the fastest and replaces the high-score if the player achieved better timing.



* **Multiple Key and Door System**
  + To clear the level, players must collect a yellow key which unlocks a corresponding yellow door. But there are blue gates which must be unlock by collecting a blue key.



* **Dynamic Flooring**
  + There are different type of floors that will affect gameplay. Unstable floors breaks after the player steps over it while stepping on Muddy floors causes FOE to move two steps instead of the normal one step

tea.png

**Knowledge Applied**

* **C++ Lessons**
  + Everything that we’ve learned in the C++ lessons helped a lot in our project.
  + This includes, but not limited to :
    - Loops
    - 2D Arrays
    - Strings
    - Vectors
    - Pointers
    - Structs
    - Enums
* **Principle of Game Design**
  + Game Design Document
  + Fundamentals of Level Design

**Challenges**

**Challenges encountered:**

* Function for different types of FOE sharing the same variables causing issues.
  + Used struct to store variables for each individual FOE.
* Writing to and reading from text file
  + Researched online and learnt to use fstream and its functions
* Array size was not dynamic unable to store different sizes of maps.
  + Using Vector to store the map, it is more dynamic to different sizes of map.
* Understanding the framework
  + Lots of testing using my own cpp file

**Things we’ve learnt:**

* Teamwork is extremely important, if one of us is not cooperating, the project would have not been successful.
* fstream is very similar to HTML and provides a new way to store and read data inside text files.
* Multiple files makes our codes way more organized

**Future enhancements**

* Add an FOE that will chase the player
* Add an FOE that will shoot in a direction
* More variety of difficulty
* Step counter, take the least amount of steps to beat the game. Or limit the amount of steps the player can take per level.
* Limited vision
* Add more variety of traps
  + Ice floors, once a player steps on it, the player will slide till a player steps on a normal floor.