Tower Defense Project Plan

By: Aditya, Xiong, Dung, Nguyen

Game description:

Our project for the Object oriented programming course in C++ is a tower defense game. The goal of the game is to survive 10 rounds of enemies with increasing difficulty each round in order to win the game. The enemies will attempt to reach the end of the predetermined path and the goal of the player is to stop any of them from doing so. Failure of such will result in a loss. The player may achieve this through the building of towers in exchange of credits, which is earned through defeating enemies. Towers are also upgradable to deal with enemies in the later parts of the game.

This is our intended look/UI for the game :

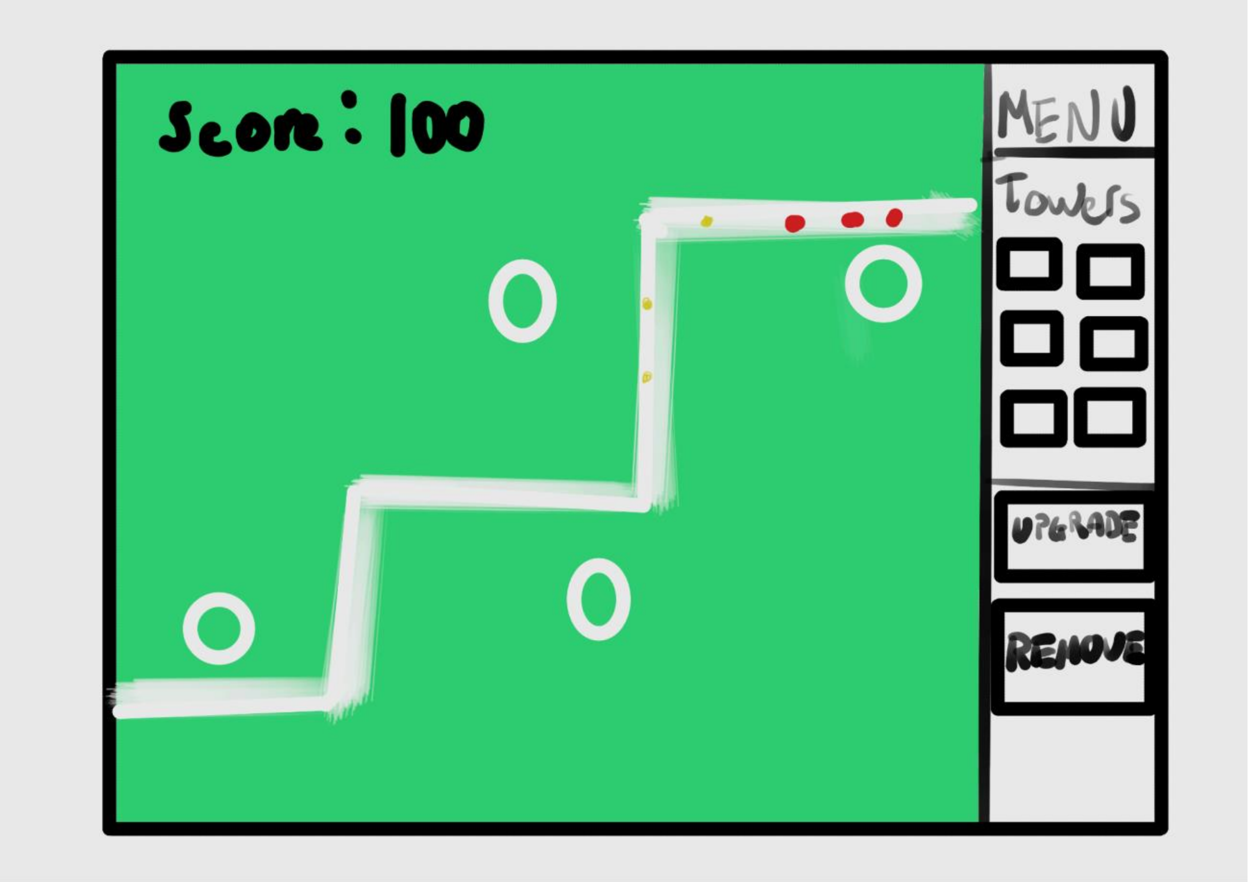


Fig.1 Intended GUI

As can be seen from the illustration, we also intend to include a score system. The current workings are not fully decided on but the score given will probably be based on how fast the enemies are killed.

Schedule plan:

Week No.	Implementation plan for the week
Week 1	We will implement the basic modules for the following - Tower class and its subclasses - Enemy class and its subclasses - Tower and Enemy classes which have an HP/XP/Special attack statistics system implemented - Map class
Week 2	- graphics - sound - wave/placement modes implementation (including remove/replace functions)
Week 3	- level editor - high score + username save function - upgradable towers
Week 4	- general code polishing/bug squashing - optimizing gamer happiness
Week 5	- same as week 4 - demo alert

Program design:

Main classes of the game will include a tower class, an enemy class, and a map class. Each class is developed independently in the file with the same name.

Tower class includes towers that shoot at the closest enemy according to the path, different towers have different ranges and different power. More specifically, towers class has the following subclasses:

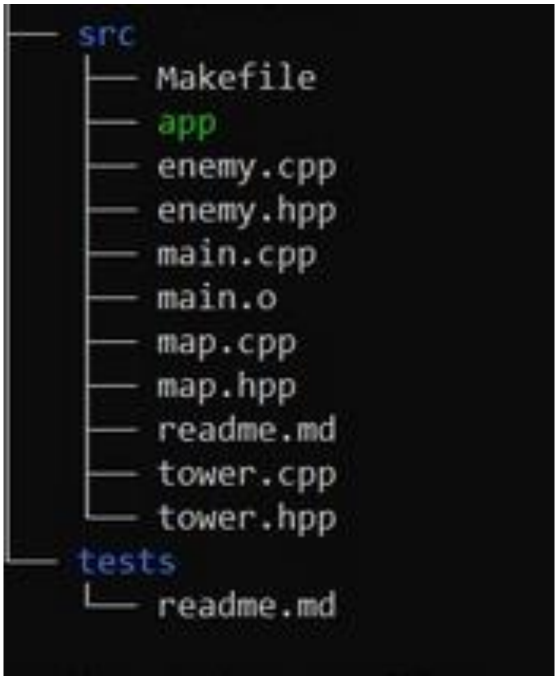
- Plant type (basic tower, can only attack one enemy at a time with set damage)
- Fire type (damage over time to multiple enemies)
- Water type (slows enemies in their path, does not deal significant damage)
- Bomb type (Attacks multiple enemies)
- Ground type (Blocks enemies)
- Magic type (Gives other towers bonus HP/XP)

Enemy class includes various types of enemies that can and cannot attack towers, different enemies also have different passive, health, and strength. Enemy subclasses are:

- Plant type (basic enemy, can deal basic damage)
- Tree type (similar to plant type, but splits into multiple plant types enemies when HP reaches 0)
- Fire type (can deal damage over time to towers)
- Bomb type (can attack multiple towers)
- Magic type (Gives other enemies bonus HP/XP)
- Ground type (blocks towers from dealing damage)
- Water type (Makes towers only deal 50% damage as long as it is alive)
- Boss type (Can kill towers in one hit and has a lot of HP, but moves very slowly and has smaller detection radius)

Map class implements the function to read saved maps from a text file. This is how the 10 preset levels will be saved and played. The map design will be based on a 16x16 grid.

The UI of the game will be implemented in the main.cpp file.



```
├── src
│   ├── Makefile
│   ├── app
│   ├── enemy.cpp
│   ├── enemy.hpp
│   ├── main.cpp
│   ├── main.o
│   ├── map.cpp
│   ├── map.hpp
│   ├── readme.md
│   ├── tower.cpp
│   └── tower.hpp
└── tests
    └── readme.md
```

Fig.2 Main modules of the project

The project will make use of the Simple and Fast Multimedia Library (SFML) for graphics, audio, and processing mouse inputs.

Division of work:

This will be discussed and decided at the start of each week.

First week	
Name	Responsibility
Aditya Agrawal	Implementing the game UI
Dung Nguyen	Map class
Nguyen Hoang	Basic Enemy class
Xiong Xiong	Basic Tower class structure