Tower Defense

Mikhail Romanov, Aki Oura, Leo Teodosin and Someone

October 2023

Contents

1. Scope of the project
2. High-level structure
3. External libraries
4. Division of work
5. Schedule
6. Scope of the project

Our scope is to develop fun game that will have a lot of features. Game will have enjoyable graphics and sound effects. Also game will have main menu for user to navigate between levels.

Game will start with main menu where user can choose level to play or randomly generate fully new one. After that level will start and user will be given some fixed amount of money and first round will start. Every round there will be some given amount of enemies coming to field and users mission is to not allow them to cross the finish line. Player will be given money for every enemy destroyed and after every round. User can win the game by deafening some fixed amount of rounds or lose the game by allowing to many enemies to cross the finish line. After winning the game user will have option to continue game, then game will continue until player loses it.

Planned features:

* Main menu
* Audio effects
* Visual effects
* Excellent graphics
* Random generated level
* Multiple Tower types
* Multiple Enemy types
* High score
* User interface
* Mouse controls
* Multiple difficulties

1. High-level structure

Game will consist on single Game-object, which manages rest of the game logic. It will consist of 3 simple methods, one of them starts the game, one quits and one of them updates.

We will have one Object class that will have all moving parts stored in it. Object class will have to big subclasses Tower and Enemies. And this subclasses will have smaller subclasses for every type of defender or attacker. All objects have variables HP, damage, texture, shape, attack speed and range so this variables will be stored in Object class. Also Object class has two virtual functions playSound and attack. Enemies will have two separate variables defense and speed and own function move. Towers will have cost and level as their own variable.

We are planning to make map 10x10 grid. Map data will be stored in 2d vector and every object will have shape class that will store their position on the map as (x, y) coordinates. We will have existing maps in our game that form which user can select one to play or user can generate fully random level using our map generator.

All basic parameters of the game such are resolution and graphics will be stored in the game class. Also all game data such are vector of towers or vector of enemies will be stored in Game class to.

1. External Libraries

We are planning to use only one external library

* SFML

In out opinion we will need only one library for the graphics of our game. We haven’t yet worked with the SFML library so out opinion might change in the process. This change is possible on the first week of the development phase.

1. Division of work

Primary plan:

* Object class: Leo and Aki
* Level class: Mikhail and Leo
* Graphics: Aki and Leo
* UI / GUI: All
* Sounds: All

1. Planned schedule

Plan is to start working on object class and level class simultaneously, because object class requires a lot of work and level class is required to test object class. We are planning that two persons start working on object class on first week and two others will start working on level class. After finishing level class one of group members will switch to help with object class development and one will start working on graphics of the game. After finishing object class one person will fully test it and two others will start developing UI, GUI and sound. To stay in timing we are planning to meet weakly to check on the progress and make changes in the plan if someone is not on time.

Plan week by week:

* Week 41 Planning
* Week 43 Finishing plan, starting work on the classes and getting to know external libraries
* Week 44 Basic class coding and implementing grid structure
* Week 45 Class coding and working on basic graphics
* Week 46 Connecting classes and graphics with the grid structure
* Week 47 Finishing touches / Documentation
* Week 48 Polishing up the game