

Cocos2d project

Tower defense game By Wouter Van Hecke

Content

1. [Introduction 2](#_Toc497843279)

[What is a tower defense game?](#_Toc497843280)

[What is different in this tower defense game?](#_Toc497843281)

[Content of the game](#_Toc497843282)

1. [Objectives 2](#_Toc497843283)

[Objective of the project](#_Toc497843284)

[Objective of the game](#_Toc497843285)

1. [Use cases 3](#_Toc497843286)

[Starting the application](#_Toc497843287)

[Starting the tutorial level](#_Toc497843288)

[Playing the tutorial level](#_Toc497843289)

[Change the background music](#_Toc497843290)

[Look up an enemy in the adventure book](#_Toc497843291)

[Choosing a level](#_Toc497843292)

[Playing a normal level](#_Toc497843293)

1. [Usage workflow 7](#_Toc497843294)

[Main menu](#_Toc497843295)

[A standard level](#_Toc497843296)

[Endless mode](#_Toc497843297)

1. [Mockups 8](#_Toc497843298)

[Logo screen](#_Toc497843299)

[Main menu](#_Toc497843300)

[Stage screen](#_Toc497843301)

[World screen](#_Toc497843302)

[Choose level screen](#_Toc497843303)

[Tutorial level screen](#_Toc497843304)

[Normal level screen](#_Toc497843305)

[Level ending screen](#_Toc497843306)

[Adventure book (tower/enemy choice) screen](#_Toc497843307)

[Adventure book (tower /enemy) screen](#_Toc497843308)

[Store screen](#_Toc497843309)

[Settings screen](#_Toc497843310)

1. [UML 11](#_Toc497843311)
2. [Database planning 12](#_Toc497843312)
3. [Workload and time planning 12](#_Toc497843313)

Introduction

In this project, I will make a tower defense game with Cocos2d.

What is a tower defense game?

In a tower defense game, there is always a beginning and an ending position. Enemies spawn at the beginning place and there goal is to get to the other side. Your goal is to stop them from reaching the end by placing towers that kill the enemies.

What is different in this tower defense game?

In this tower defense game, it is not your choice which tower you get, nor which position it will be placed. You upgrade your tower to do more damage to the enemies and in a later stage evolve your towers to get new powers.

Content of the game

In this tower defense game, you have a few options how you want to play the game. There is a standard version with a few worlds and a detailed tutorial. You can play themed levels, like snow levels, Halloween levels, summer levels and so on. Then in the end there will be an endless mode, so you can compete with your friends to get the farthest.

There are a lot of different enemies and turrets to experiment with. Which will be the best one to upgrade. Turrets and enemies all have unique features.

For the levels, I will see how far we get with making a few levels. But normally, when I have the basic implementation ready, the levels would come quicker.

Objectives

Objective of the project

Because cocos2d uses JavaScript to build up the project, I will also learn more about JavaScript and how to use certain API’s. During this project, I will learn more about how different objects in a game communicate and act towards each other. Collision detection will be a very important component in this project because that is how towers know when an enemy is nearby and for the projectile to find its target. I might be able to work with adds in the game if I can figure out how that has to be implemented. The end objective is to be able to compile it to an Android phone (maybe be able to get it on Google Play).

Objective of the game

The basic objective of the game is to defeat every level, which get harder as you progress. You can also try to collect all different enemies and towers in your adventure book. There is also an endless game mode in which, the goal is to get as far as possible.

Use cases

## Starting the application

#Use Case 1: Starting the application

**Primary actor:** the player

**Preconditions:** /

**Postconditions:** The application started up and is positioned on the main menu.

**Basic flow of events**

1. The user opens the application on his mobile
2. The System starts the music
3. The system shows the logo screen
4. The system shows the main menu screen

**Alternative flows**

1. The user closes the app during start up

## Starting the tutorial level

#Use Case 2: Starting the tutorial level

**Primary actor:** the player

**Preconditions:** The application started up and is positioned on the main menu

**Postconditions:** The tutorial level has started

**Basic flow of events**

1. The user choses the start option
2. The system shows all the stages
3. The user choses the standard stage
4. The system shows all the worlds
5. The user choses the tutorial option
6. The system starts the tutorial level

**Alternative flows**

1. The user choses the adventure book option UC\_Look\_up\_an\_enemy\_in\_the\_adventure\_book
2. The user choses the themed stage or endless mode
3. The user choses the first-second-third world

## Playing the tutorial level

#Use Case 3: Playing the tutorial level

**Primary actor:** the player

**Preconditions:** The tutorial level has started

**Postconditions:** The tutorial level is completed

**Basic flow of events**

1. The system gives information about aspects on the screen
2. The system starts the first wave
3. The user completes the wave
4. The system gives information about how money works in the game
5. The system gives information about towers
6. The system starts the second wave
7. The user completes the wave
8. The system gives information about new enemies
9. The system starts the third and fourth wave
10. The user completes the waves
11. The system gives information about evolving towers
12. The system gives information about the boss level
13. The system starts the last wave
14. The user completes the wave
15. The system gives a message about the completion from the tutorial level

**Alternative flows**

1. The user can not complete a wave
2. The user stops the game
3. The user stops the level

## Change the background music

#Use Case 4: Change the background music

**Primary actor:** the player

**Preconditions:** The application started up and is positioned on the main menu

**Postconditions:** The background music has changed

**Basic flow of events**

1. The user choses the settings option
2. The system gives the choice of music
3. The user choses a different music track

**Alternative flows**

1. The user didn’t choose another music track

## Look up an enemy in the adventure book

#Use Case 5: Using the adventure book

**Primary actor:** the player

**Preconditions:** The application started up and is positioned on the main menu

**Postconditions:** The user has found a specific enemy

**Basic flow of events**

1. The user choses the adventure book option
2. The system gives a tower option and an enemy option
3. The user choses the enemy option
4. The system gives al found enemies
5. The user choses a specific enemy
6. The system gives information about the specific enemy

**Alternative flows**

1. The user choses the start game option
2. The user choses the tower option
3. The uses hasn’t found the specific enemy yet

## Choosing a level

#Use Case 6: Choose a specific level

**Primary actor:** the player

**Preconditions:** The application started up and is positioned on the main menu

**Postconditions:** The specific level has started

**Basic flow of events**

1. The user choses the start option
2. The system shows all the stages
3. The user choses the standard stage
4. The system shows all the worlds
5. The user choses the first world
6. The system shows all the levels in a list view
7. The user choses the third level
8. The system starts the third level

**Alternative flows**

1. The user choses the adventure book option UC\_Look\_up\_an\_enemy\_in\_the\_adventure\_book
2. The user choses the themed stage or endless mode
3. The user choses the tutorial-second-third world
4. The specific level is not unlocked yet

## Playing a normal level

#Use Case 7: Playing a normal level

**Primary actor:** the player

**Preconditions:** The specific level has started

**Postconditions:** The specific level is complete

**Basic flow of events**

1. The system created the level screen
2. The user buys a tower
3. The system spawn a new random tower
4. The user upgrades the tower
5. The user starts the wave
6. The system spawns the enemies
7. The system lets the towers find an enemy
8. The systems lets the towers shoot the enemy
9. The user can buy new towers
10. The user can upgrade more towers
11. Go to step 7 until the wave is done
12. Go to step 5 until the last wave is done
13. The system gives a message about the completion from the level
14. The system gives the user an amount of points
15. The system adds the new towers and enemies to the adventure book

**Alternative flows**

1. The user doesn’t buy towers
2. The user doesn’t upgrade towers
3. The user loses the level

Usage workflow

Main menu

From the main menu, you have a few options. You can choose the start button, store button, adventure book button, credits button, exit button and settings option. If you choose the start button, you will be guided to the stage selection where you can choose between the standard, themed or endless stage. In the standard stage you can find all the normal levels. In the themed stage you can find levels for special occasions and of course the endless mode. When you choose the store button, you will be guided to the store area, where you can buy special items with your in-game coins or real money. The items exist of exclusive towers, levels and world. You can even buy a better speed up button. If you choose the settings button, you have the option to change or disable the music. Then as last thing, if you choose the adventure book button, you will be guided to the adventure book screen. There you can find all information about all the found towers/enemies.

A standard level

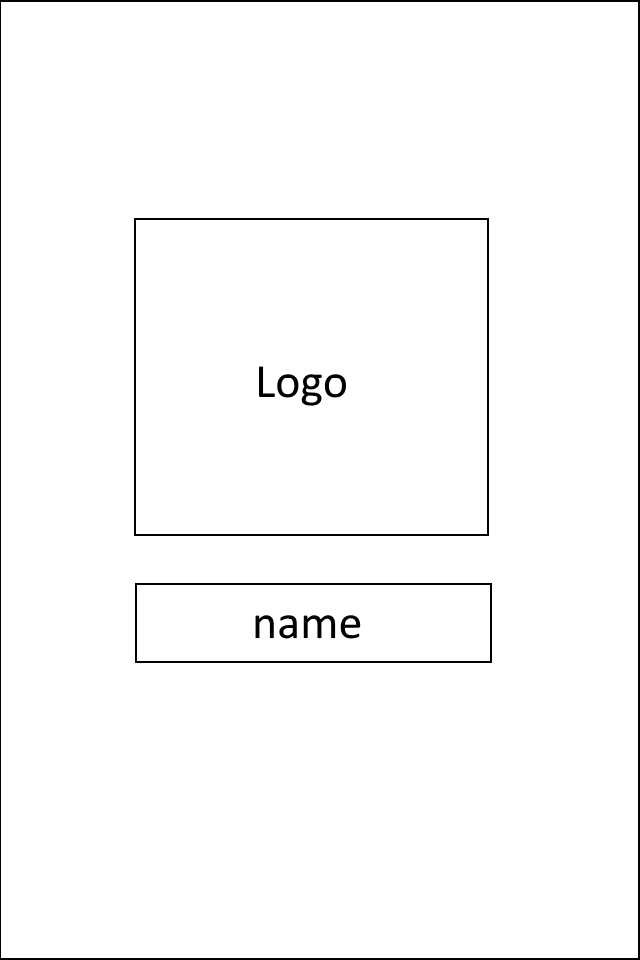
When you first start a level, you already start with some money to buy a tower. You have to press the button yourself to start the new wave. You get money for every enemy you kill and with that money, you can buy towers or upgrade your existing ones. Not every wave is the same, there will be different enemies distributed over all the levels, the amount of enemies per wave will also very. Every 5 waves and at the end of the level, there will also be a boss fight, which is one big, difficult enemy to beat. At the end of the level, you also get special coins, how much coins you get depends on how well you did on the level.

Endless mode

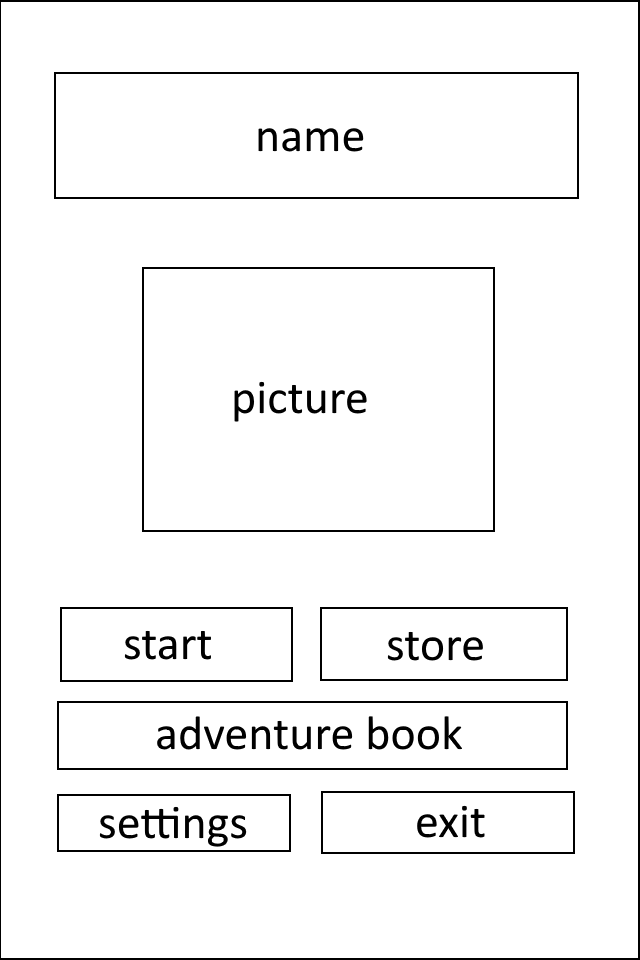
Pretty much the same rule apply here for a normal level, but there will be no and, not until 3 enemies pass the ending point.

Mockups

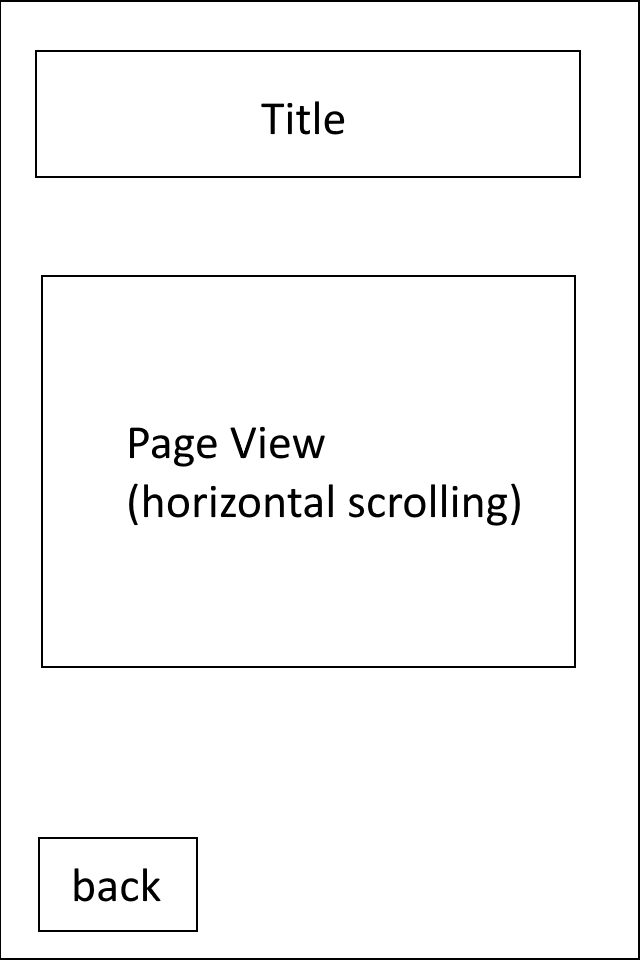
Logo screen



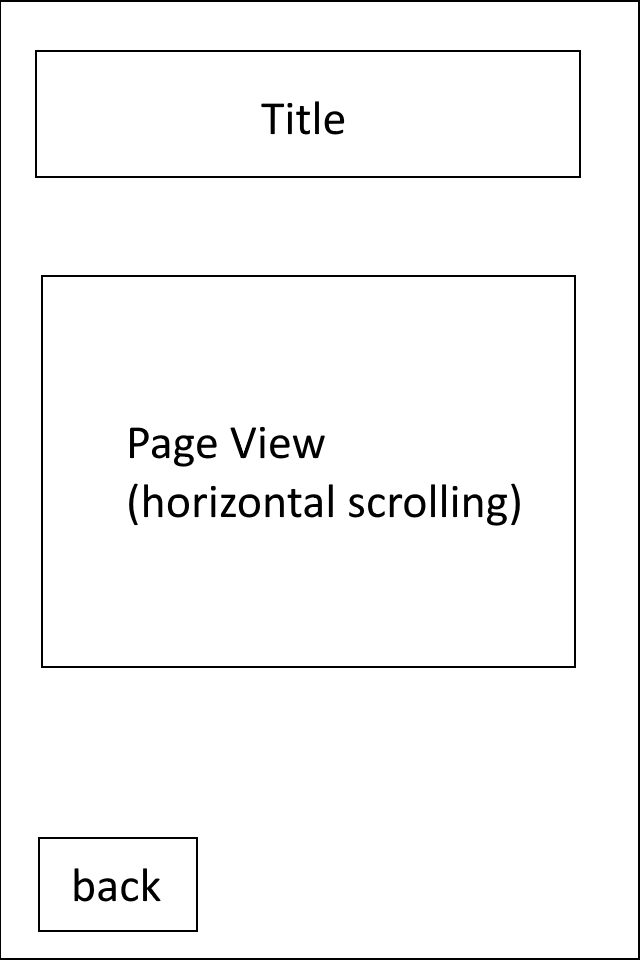
Main menu



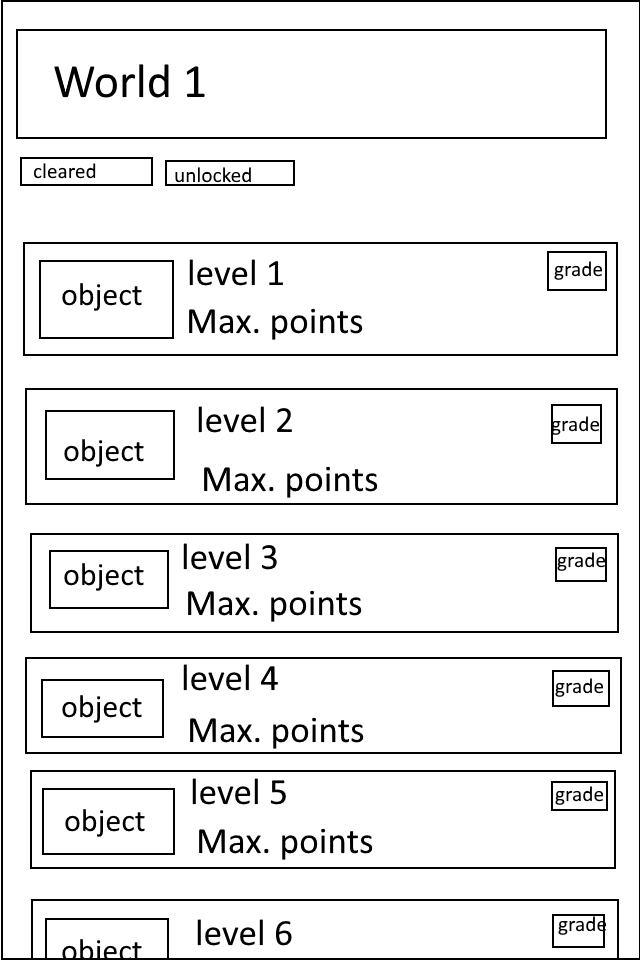
Stage screen



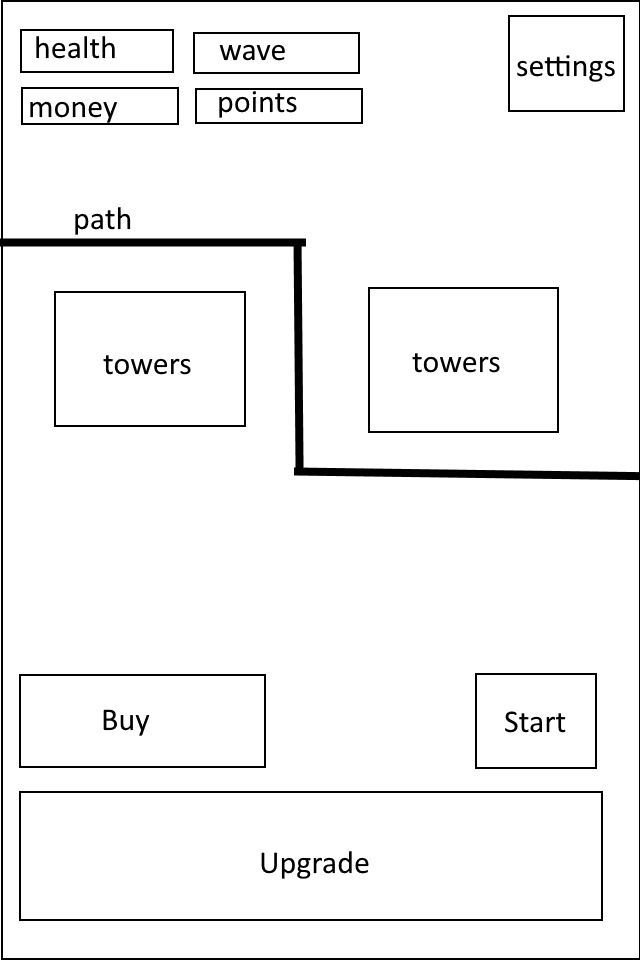
World screen



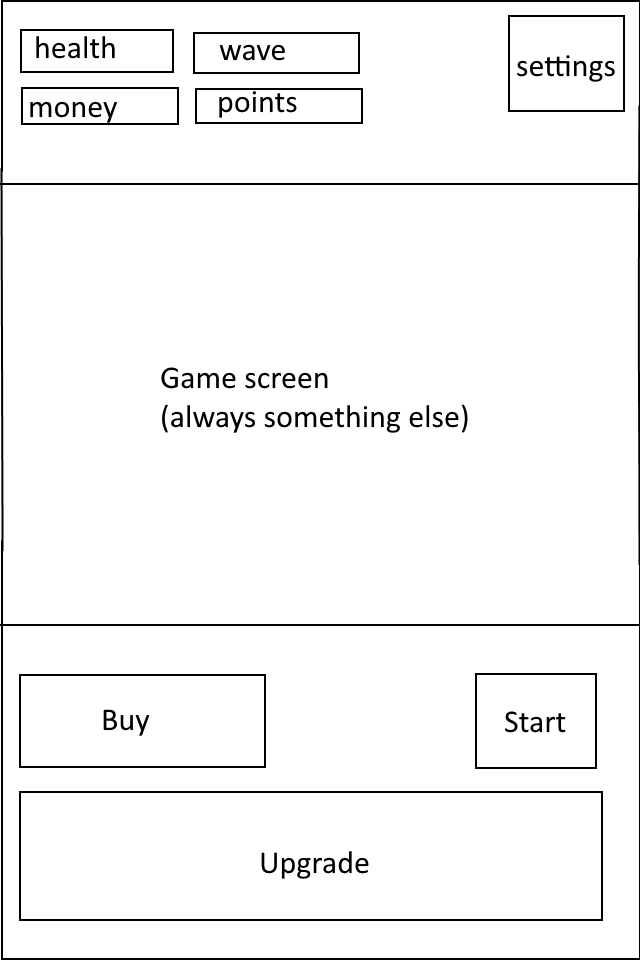
Choose level screen



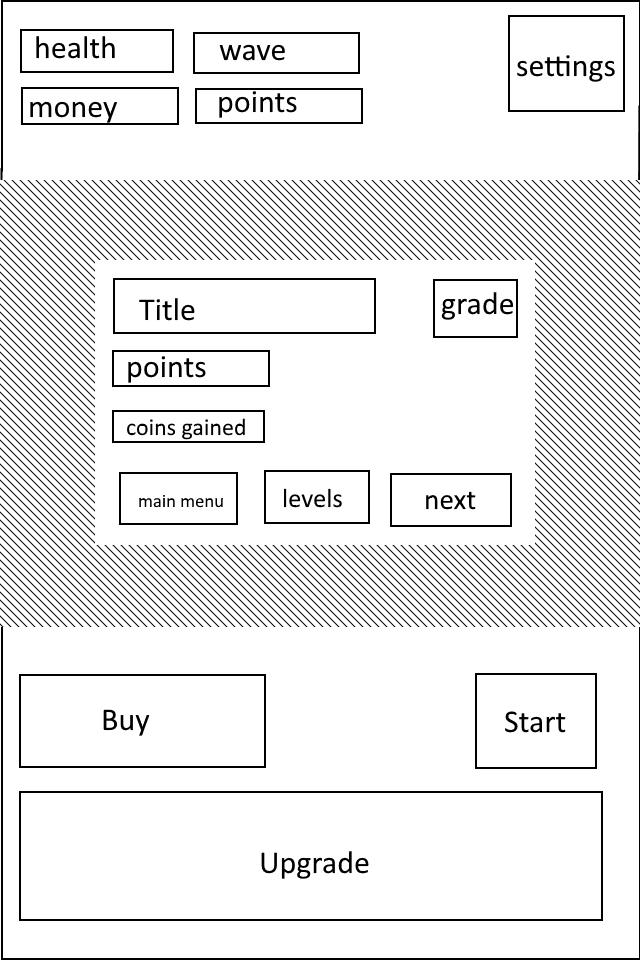
Tutorial level screen



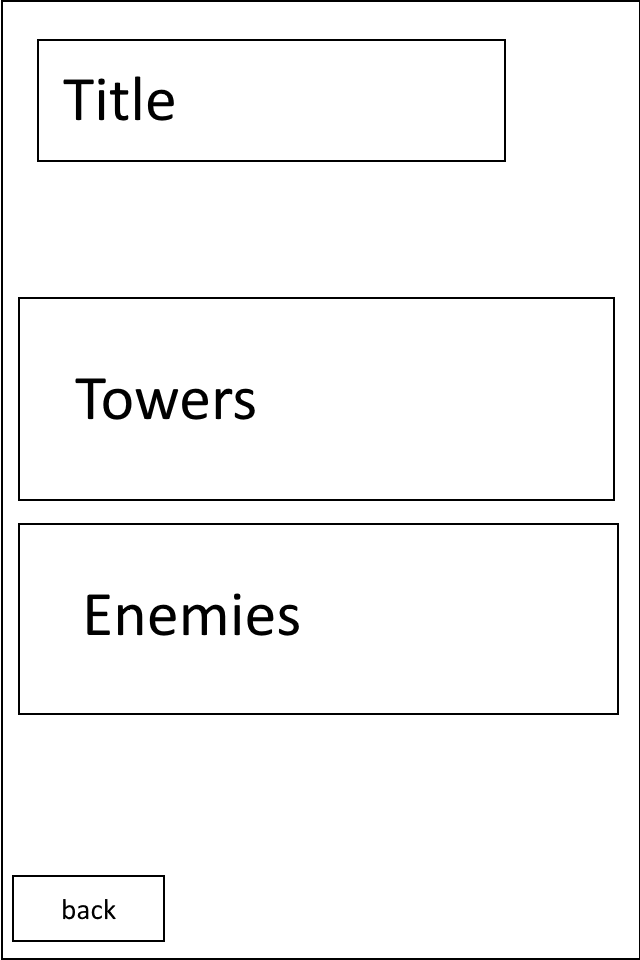
Normal level screen



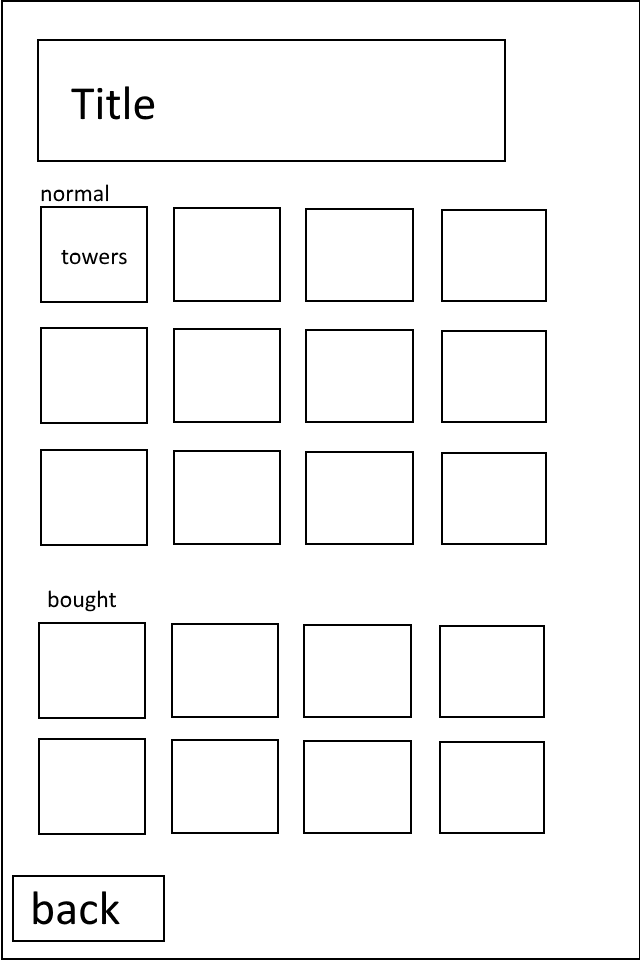
Level ending screen



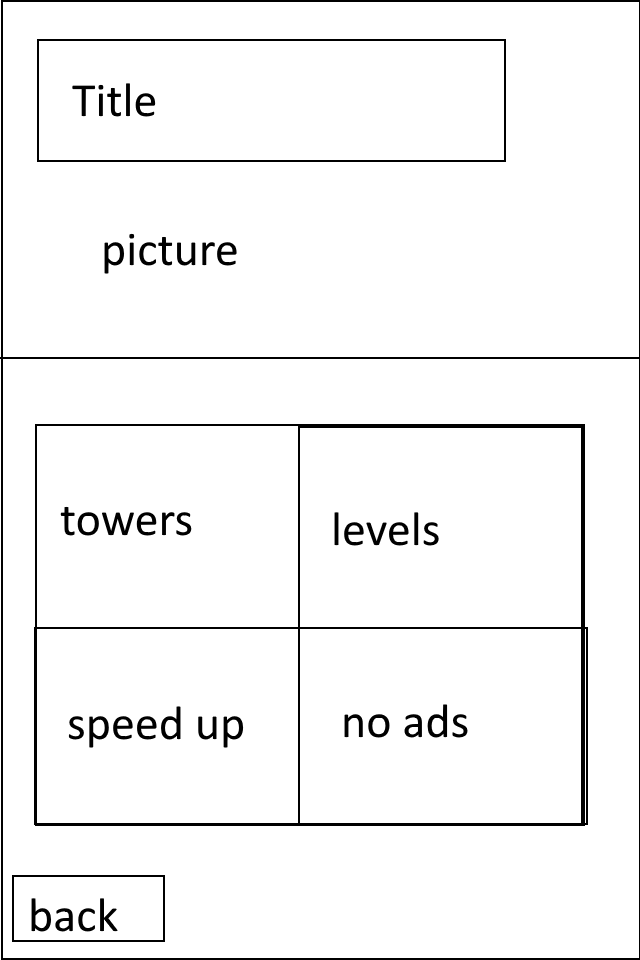
Adventure book (tower/enemy choice) screen



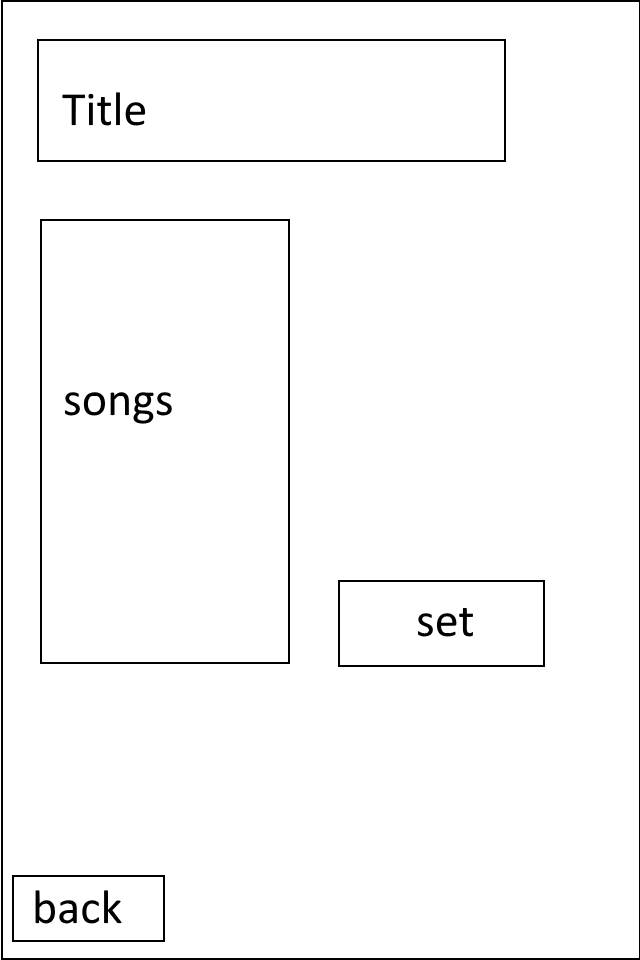
Adventure book (tower /enemy) screen



Store screen

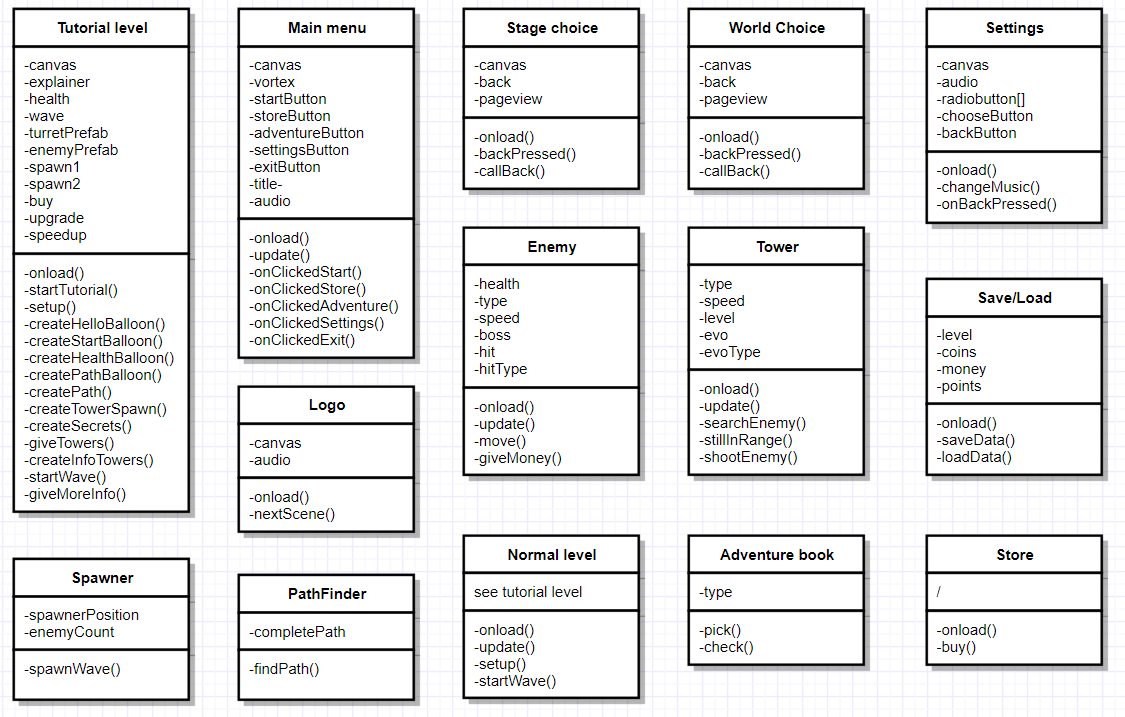


Settings screen



UML

Cocos2d uses JavaScript, so the classes are not really connected to each other, only a few are.



Database planning

For this game, there is no need to use a Database. There might be a need to use xml files to know which paths will be used in every level. For normal things like level progress, I will use Local Storage and JSON.

Workload and time planning

When it comes to planning, I’m more of a type that just keeps working on the project and doesn’t look at what still needs to be done, what is done already and so on, I just keep on working on the project. But here are the things that need to happen.

* + Make design for the main menu, logo, store, adventure book, every level.
  + The enemies have to be programmed perfectly to move on every level and apply certain types like fire or electricity.
  + The towers have to be programmed perfectly so they always find the right enemy and shoot right at them. They need to be able to upgrade and evolve in a correct way.
  + The levels need to feel error free, everything has to run smooth, the money/damage system has to be on point.
  + I will try to find a way to integrate google play services into Cocos creator.
  + I will try to find a wat to integrate ads into Cocos creator.