CS 319 - Object-Oriented Software Engineering
Analysis Report

Defenders Of The Kingdom

Group 3-H

Alp Ege Baştürk

Barış Eymür

Emre Gürçay

Öykü Ece Ayaz

Table Of Contents

1 Introduction

Defenders of the Kingdom is a basic tower defence game that takes place in medieval ages, and concerns two kingdoms that are in a war. Tower defense is a popular type of strategy video game in the industry. The main aim of the tower defense games is to stop attackers from passing through the gates of the defending territory.

Similarly in Defenders of the Kingdom, the player will attempt to defend his/her base castle from the waves of attackers. Attackers will proceed following a path to the base castle. If they manage to enter the base, the player will lose health points in a correlation with the number of attackers that got through the castle. The path that the attackers follow will advance in different difficulty levels. Player will locate his/her towers around the path to defend the base as the attackers will try to pass through the base castle. As the difficulty level increases, the attackers will have increased attributes too. The player will have the option to upgrade his/her towers, increasing the performance of the towers. The other defence features include obstacles and heroes.

Defenders of the Kingdom is a PC - based game and will be played with the mouse and the keyboard.

We were influenced by GemCraft on Armor Games' website [1] while brainstorming on the game idea.

2 Overview

Tower defense is a subgenre of strategy games like Kingdom Rush. It is very easy and fun to play. Since the game has a story and the player needs to follow a strategy and consider his/her moves before he/she will act on. The difficulty of the levels will increase and new levels will be unlocked as the player manages to complete the level successfully. In the game, there are two kingdoms, A and B. The attackers of the kingdom A will try to invade the base castle of kingdom B. The player attempts to defend the gates of the kingdom B.

When a new game begins, the player is provided a certain amount of gold to purchase items while defending the kingdom B. Such items include towers to shoot the attackers, an obstacle that will be placed on the path of the attackers to stop them for a while, and also a hero from the kingdom B that will fight the attackers until he dies. The player may purchase and locate the towers near the path in order to shoot attackers. There are different kinds of towers which are different in defense strategies and power. Also there are two types of heroes which are different in strength and abilities. There is an obstacle as well which stops the attackers for a while then it tears down as the attackers gather up. Different towers are effective on different types of attackers. The strategy of the player directly plays a role on the flow of the game, only the right strategies will lead the player to the triumph.

2.1 Gameplay

The player will only require a mouse to play this game. The user will be able to place the towers in the locations he/she wishes by using the cursor on the screen, which is controlled by the mouse. The user will also be able to select the level that he/she wants to play from the main menu using the mouse. Then, if the user presses the button "PLAY" on the main menu, the game will start.

2.2 Levelling

There will be 5 different levels in Defenders of the Kingdom. The first level will be the easiest one to play. The attackers in this level will be more vulnerable to the attacks of the player. As the levels increase, the game will become harder to play. The final level of the game will be the most difficult one. The enemies in this level will be quite hard to destroy since they are more resistant to the attacks on their life points. A solid strategy will be necessary to stop the enemies from passing the city walls.

As the user completes levels, different kinds of towers and attackers will be unlocked. The user will be able to play the current level of the game and the previous levels that s/he had completed before.

2.3 Towers

There will be two kinds of towers in order to defend the castle and attack the attackers. The main goal of the towers is to destroy the attackers. The first type of the tower will shoot one attacker at a time. The second one will create a bomb effect on the attackers, damaging several attackers inside a certain radius. The player will purchase and place the towers using the gold s/he has earned. Later on, the player will be able to sell his/her tower.

The upgrades to the tower will be available once the player has purchased the item. The upgrades include increasing the rate of the shoot a tower can perform (for both towers), the damage one shoot causes on the attacker (for the first type) and the increase in the radius of the damage (for the second type)

2.4 Attackers

There will be 3 different types of attackers. As the player passes the levels, new kinds of attackers will be unlocked.

First type of attacker will be the standard type for attackers. It will be the less powerful one among the attackers.

The second type of attacker will be more resistant to the attacks of the defense towers. Its health will decrease slower than the standard attacker.

The third type of attacker will move faster than the previous ones. Therefore, it will be harder to kill for the towers.

2.5 Obstacles

In order to block the attackers' path to the castle, there will be obstacles available for the player. The player will purchase it and then sell the obstacle if (s)he wishes to do so. The player can use obstacles to stop the attackers for a while. Then, the towers will have more time to shoot the attackers. However, the obstacles can be damaged by the attackers as they gather up. As a result, the obstacle will save the player some time, but eventually the obstacle will tear down. The player can locate the obstacles to the desired places on the path. When

placing obstacles it will be more beneficial for the player if the player places obstacles in the shooting range of the towers. Then, the attackers will be trapped in the shooting range and get more damage.

2.6 The Heroes

Another feature of the game is the heroes of the defending kingdom. The heroes are basically the protectors of the kingdom. They have various kinds of abilities and strength. As they are very powerful and useful, they are expensive to purchase. The player would purchase heroes in the cases of an emergency. For example, if the attackers are very close to the gate, then the player should purchase them to avoid attackers to invade the castle. Like towers and an obstacle, the player will select a hero in order to integrate a hero to the game field. When a hero is purchased, the player is not able to locate the hero. However, the hero leaves base castle in order to fight the attackers who are close to the gate. When the heroes are called to the battle field, as they are so brave and fearless, they fight until death and the player is not able to sell them later. The heroes die if they have no life points left.

3 Requirement Specification

3.1 Functional Requirements

3.1.1 Play Game

Tower Defense is a strategy game which can be played with basic commands. In the game user is trying to defend the gates of his/her kingdom. In order to accomplish this, the player should buy towers, heroes and an obstacle. At the beginning of the game player is given a certain amount of gold. The gold allows the player to purchase such items during the gameplay. In order to buy towers, heroes and an obstacle, the player needs to choose one of the desired items by clicking on the item and selecting it. Then, the player can locate the selected item

to the desired location on the game field. When placing towers, the player is not able to place them on the path of attackers, but near the path. Also when the player selects a hero, (s)he is not able to locate it to the game field because the hero walks from the gates of the kingdom and follows the path towards the attackers.

During the game, the player can also upgrade the towers which are placed on the field. Also, the player is able to sell the towers for a less amount of gold than s(he) bought them. In the beginning of each level, the player is given 5 life points. As the attackers pass from the gates, the number of life points decreases with respect to the number of attackers that got through the gates. The player is also able to pause the game and resume.

The game may end under two conditions: one of them is when there is no life points left or when the player manages to kill all the attackers. At the end of the game, the player earns success stars according to his/her left life points. If the player finishes the level successfully, then (s)he can either go to the next level or go to the main menu. If the player can not finish the level, then (s)he can either retry or go to the main menu as well.

3.1.2 View Help

In the "Help" section, the player will get the essential information and details about the game play. The essential information includes:

- The purpose and the rules of the game
- The information about the tower, attacker and hero types
- Controllers for the game

By reading this section, the player will learn about the rules and controllers of the game. After this, playing the game will become easier and more enjoyable for him/her.

3.1.3 Pause the Game

While playing, the player can pause the game. When the game is paused, the attackers and towers stop performing and the player can not purchase or sell items. When the player wishes, the game will resume as it was left. However, if the player selects Return to Main Menu, then the game is over.

3.1.4 Stars and Scoring

In the game stars indicate the success rate of the game. From a level user can get maximum 3 stars. Stars are given according to the player's success at the levels. The success of the player will be measured based on the life points s/he has got left. At the beginning of each level the player will be given 5 life points, as the attackers pass through the gates of the kingdom the number of the life points will decrease. If there are no life points left, the player loses the game and the player can not proceed to the next level until the player finishes the level successfully. If none of the attackers manage to pass the gate, then the player gets three stars which is the maximum. The meaning of the number of stars described below.

- 3 Stars: Indicate that the player finished the level with the best strategy.
- 2 Stars: Indicate that the player finished the level with a good strategy.
- 1 Star: Indicate that the player finished the level with a satisfactory strategy.

3.1.5 View Credits

In the game, it is possible for the players to reach the contact information of the developers of the game. In the main menu if the player clicks on View Credits then the player can reach to the mail addresses of the developers. The player can send his/her opinions about the game as well as the problems about the game in order to make the game better.

3.2 Non-functional Requirements

3.2.1 Game Performance

The game should work with good performance. There will be many objects interacting with each other. We do not want this to affect the performance of the game. It should work on standard computers.

3.2.2 Graphical Smoothness

The project will be a video-game. Therefore, graphics of the game should be pleasant for the user. We plan to add projectile graphics for tower attack and animations for the actions of the elements, which should be smooth.

3.2.3 Simple User Interface

We plan to make interface simple. Players should understand how to play the game without the need of documentation. However "View Help" menu will include details of the game. Our aim is to make it possible for the user to play the game without knowing these details. Optionally, the player can learn the details of the game to improve gameplay.

3.2.4 Extendibility

We plan to make the project suitable to be extended or reused. Our aim is to make the project follow these ideas.

4 System Model

Game

View Credits

View Help

Play Game <<Include>> Pause <<Include>> Quit

<<Include>>

Continue

Player

Select Level
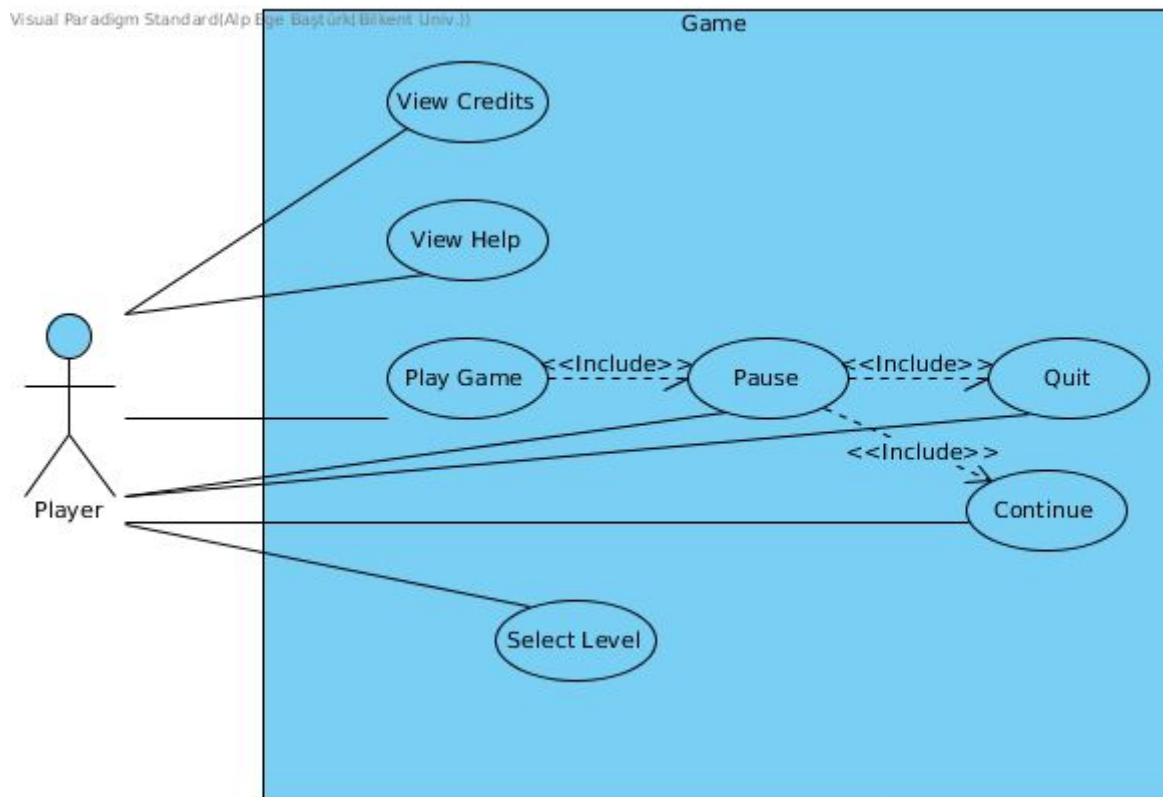
4.1 Use Case Model

This section provides information about the main use case model of Defenders of the Kingdom game.

4.1.1 Play Game

Use Case Name: Play Game
Primary Actor: Player
Stakeholders and Interests:
-Player tries to protect his or her base from waves of attackers.
-System keeps track of the score for different levels.

Pre-condition: The game settings (life points, initial gold) are set to default as a new game begins.
Post-condition: If the user completes the game with a score higher than the previous high score for the level, the new score is updated.

Entry Condition: The player chooses the difficulty level, and clicks on "Play Game" button.

Exit Condition: If the game ends or the user returns to the main menu.

Success Scenario Event Flow:
1. The game begins.
2. The player starts playing.
3. The player plays until all the attackers are dead.
4. System keeps track of the left gold.
5. System keeps track of the stars. If the number of stars are more than previous play, the star count is updated.
6. The player can proceed to the next level.
7. The player is given new life points and some gold.
8. The player starts playing the next level.
   The player repeats the steps 2 - 8 until all levels are completed or there are no points left.
9. The player returns to the main menu.

Alternative Flows:

3A. Player tries to kill all the attackers.

      3A.1. Attackers start to come.

      3A.2. The player places towers or an obstacle to the game field. Or the player calls heroes.

      3A.3. Towers start shooting to the attackers.

      3A.3. Attackers die because of the items that are placed to the game field.

      3A.4. The dead attackers disappear from the screen.

      3A.5. Player is given some amount of gold for of each dead attacker.

      3A.6. Player continues to play until all attackers are dead.

      3A.7. Player completes the level.

      3A.8. Player is given some amount of gold according to his/her success rate.

    A. The attackers pass by the gates.
      A.1. The player's life points decrease.
      A.2. The player is not able to get any gold from the attacker.
      A.3. The player can not finish the level.
      A.4 The player can retry or return to the main page.
      Events A.1 and A.2 continue until there are no life points left.

    B. The player decides to sell the towers.
      B.1. The player selects the desired tower.

B.2. The player selects the sell option.

B.3. The player sells the tower.

B.4. The tower disappears from the screen.

B.5. The player is given half of the value of the tower.

C. The player decides to upgrade the towers.

C.1. The player selects the desired tower.

C.2. The player selects the upgrade option.

C.3. The player upgrades the tower for some amount of gold.

C.4. The tower gets more powerful.

D. The player wants to quit the game.

D.1 The player selects the pause button.

D.2 Quit button comes to the screen.

D.3 The player selects the quit.

D.4 The player returns to the main menu.

4.1.2 View Help

Use Case Name: View Help
Primary Actor: Player
Stakeholders and Interests:
- The player wants to know about the instructions and the features of the game.
- System shows a text containing information about the tower types, attacker types, scoring, purchasing/ selling items, and pausing the game.

Pre-condition: The player should be in Main Menu in order to view help screen
Post-condition: -
Entry Condition: The player chooses View Help from Main Menu.
Exit Condition: The player chooses Back from the current page.

Success Scenario Event Flow:
1. Player selects View Help in the Main Menu
2. A document containing information on the flow of the game, types of towers, attackers and heroes are displayed.
   Alternative Flows:
A. The player can return to the Main Menu by clicking on the Back button.

4.1.3 Select Level

Use Case Name: Select Level

Primary Actor: Player

Stakeholders and Interests:
- Player will select the difficulty level that (s)he wants to play.
- The difficulty in the levels is maintained by the kinds of the attackers the level bears and the path followed by the attackers.

Pre-condition: The player should be in Main Menu in order to select the difficulty level.

Post-condition: The new game will be in the selected difficulty level.

Entry Condition: The player should click on any difficulty level in order to choose it.

Exit Condition: The player chooses a difficulty level and clicks on Play Game.

Success Scenario Event Flow:
1. The player will click on any level displayed on Main Menu.
2. When a level is selected, the high score that was previously accomplished in the current level will be displayed with the count of stars.
3. The player selects the play game, and the new game begins.

    Alternative Flow:
A. If the level has not yet been played before, the stars are not displayed.

4.1.4 View Credits

Use Case Name: View Credits

Primary Actor: Player

Stakeholders and Interests:
- The player wishes to reach the contact information of the developers of the game.
- As the user clicks on the button, the contact information of the developers are displayed.

Pre-condition: The player should be in Main Menu in order to select the difficulty level.

Post-condition: -

Entry Condition: The player clicks on View Credits.

Exit Condition: The player clicks on Back button to return to the Main Menu.

Success Scenario Event Flow:
1. The player clicks on the View Credits button
2. A text containing contact information of the game developers will be displayed.
3. The player uses Back button to return to the Main Menu.

4.1.5 Return to Main Menu

Use Case Name: Return to Main Menu

Primary Actor: Player

Stakeholders and Interests:
- The player wishes to return to the main menu while playing the game
- As the user clicks on the button, (s)he will be directed to the main menu screen.

Pre-condition: The player should be in the game and the game must be paused in order to navigate back to the Main Menu.

Post-condition: The game is over.

Entry Condition: The player needs to pause the current game and select Return to Main Menu.

Exit Condition: The player clicks on Back button to return to the Main Menu.

Success Scenario Event Flow:
1. The player pauses the game while playing.
2. The player selects Return to Main Menu.
3. The game is over, and the player is navigated back to the Main Menu.

4.2 Dynamic Models

4.2.1 Sequence Diagrams

Scenario: The player wants to start the game by selecting a level from the Main Menu and pressing the "Play Game" button. System initializes the game window, the list of towers box and graphics objects to draw. Game map is loaded from a configuration file. According to this file, the path which the attackers will follow and the tiles which the user can build towers will be determined. Also attacker number and types will be read from a configuration file. After this part, game loop is initialized and player starts to play.
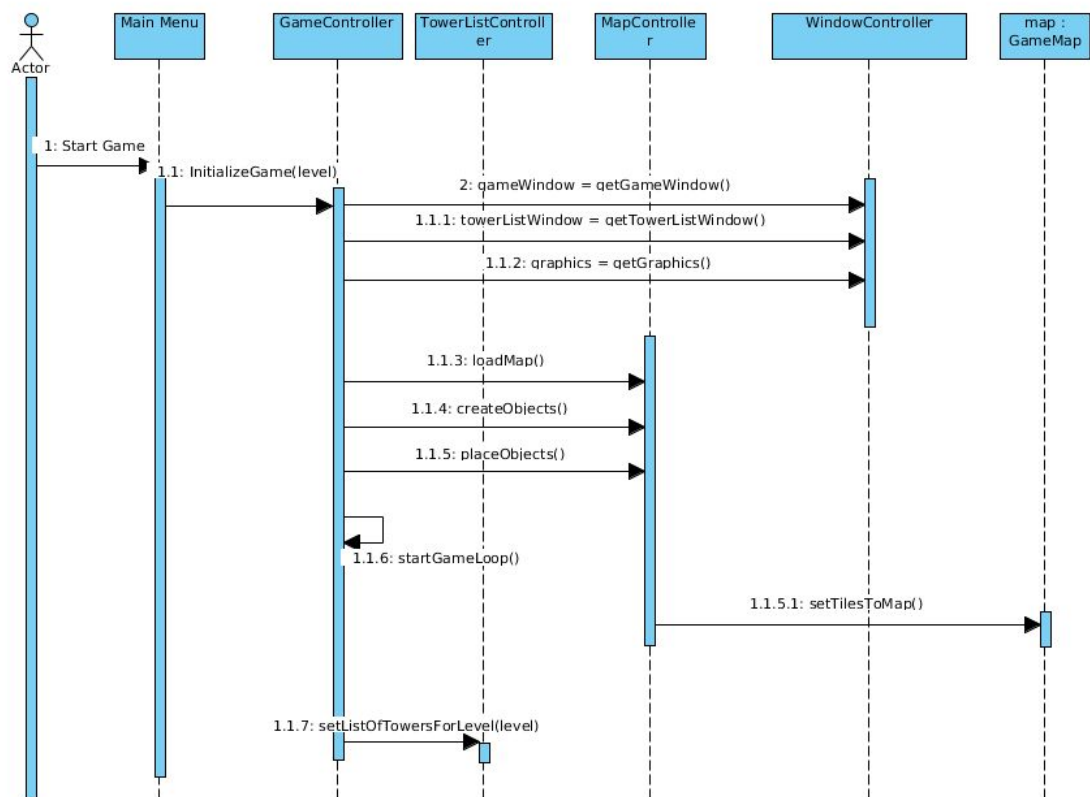
Scenario: Basic tower placement scenario. Player selects a tower from the tower box with the mouse input. Then, the player clicks on the location which he or she wants to place the tower. If the tile is suitable and the player has sufficient resources, then tower is placed and resources are deducted from the player. The map gets updated.
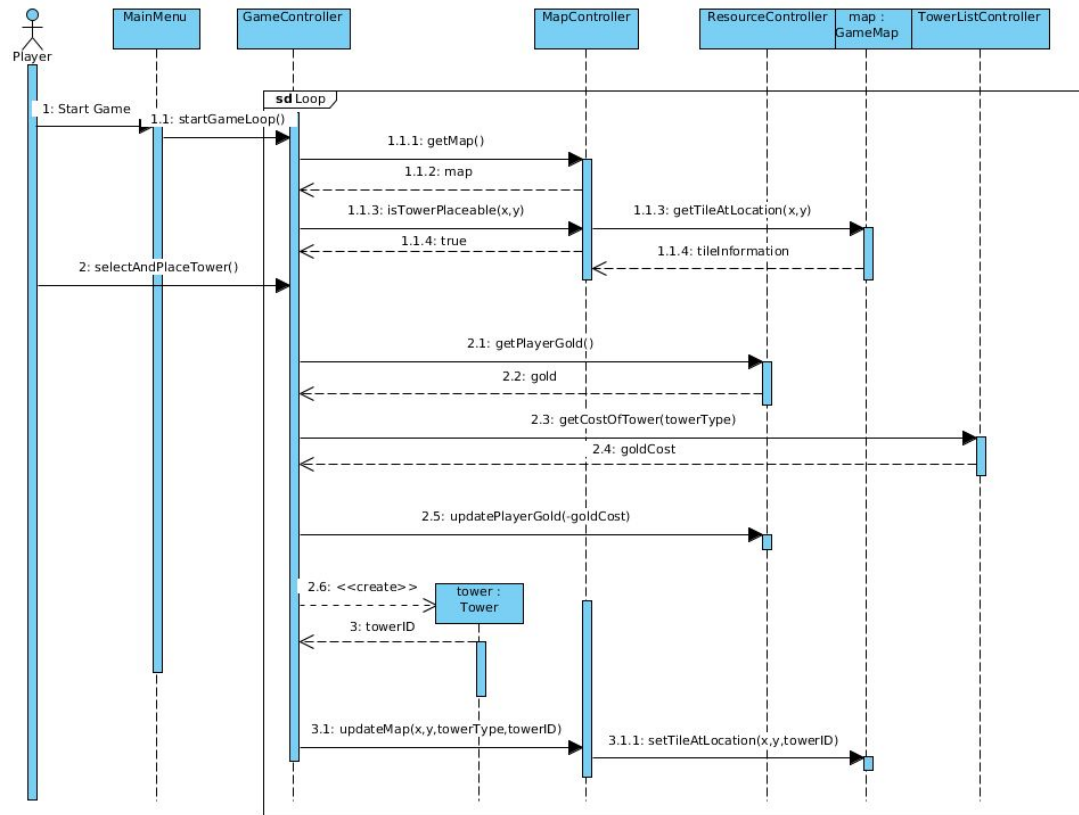

//ToDo Scenario: Tower sell. Player selects a tower, selected tower is destroyed, controllers are notified.

Scenario: Hero spawn. Hero is spawned according to the selection of the player. Process is similar to tower placement scenario. However, player does not specify the placement location. Hero always spawns at the player base building.
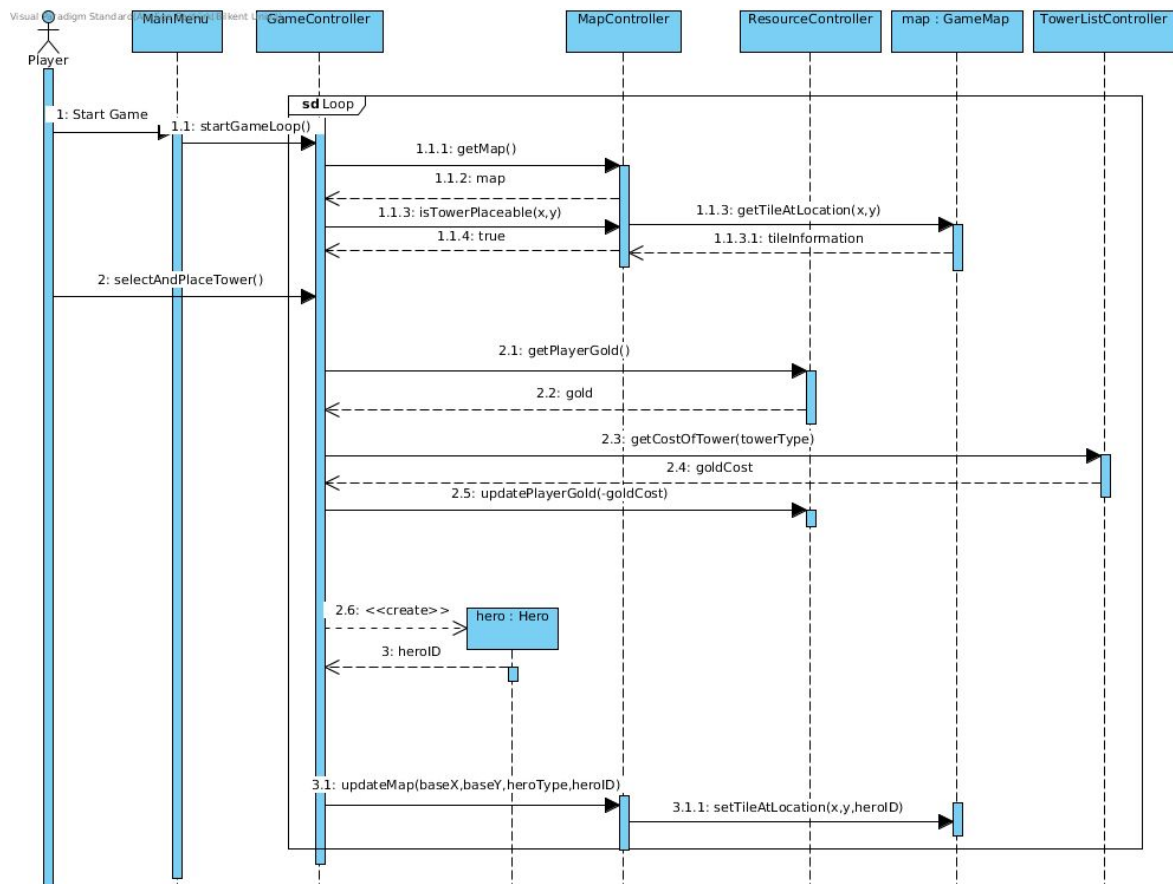
Scenario: Basic tower attack scenario during game play. For each tower on the map, game requests list of adjacent targets from the MapManager. Map manager gets location of each tower and checks adjacent tiles. Then, it returns a list of targets for a given tower depending on the tower range. The list is sent to the related tower which then performs attack operation. This operation is implemented differently for different tower types. When a tower performs the attack operation, it reduces the health of the target. This information is given to the target in the attack operation.
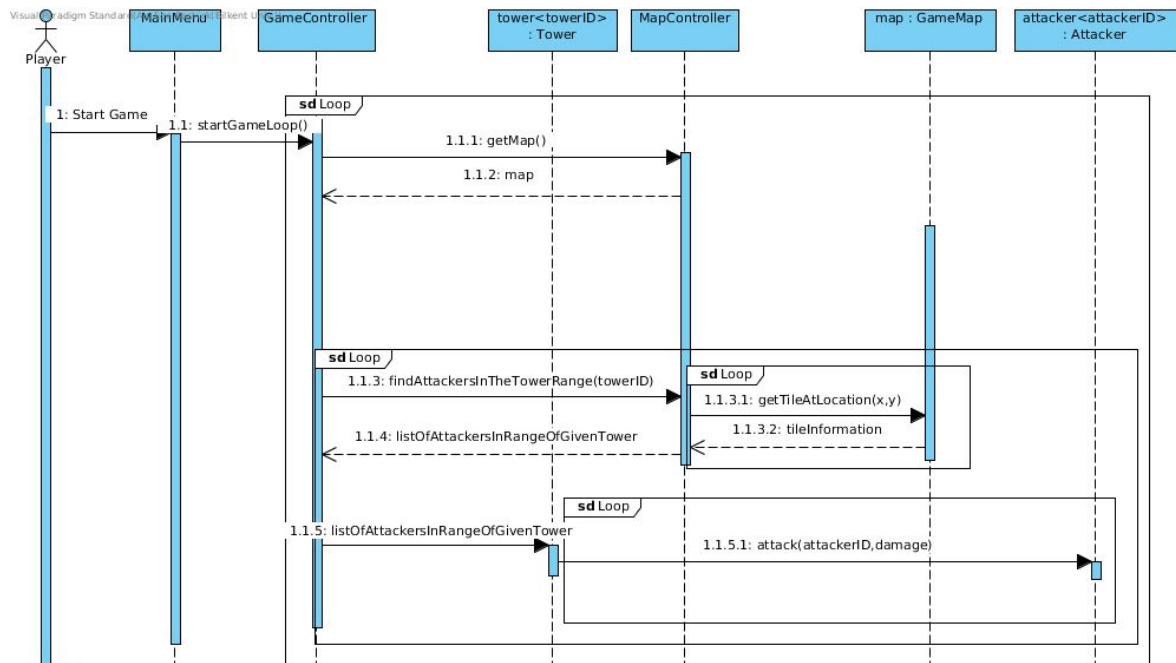
Scenario: Basic attacker spawn scenario for the game. GameController gets list of attackers from the AttackerController, which is decided according to the level. In the game loop, GameManager creates attackers and stores their list. These attackers are passed to the MapController to update the map according to their existence.
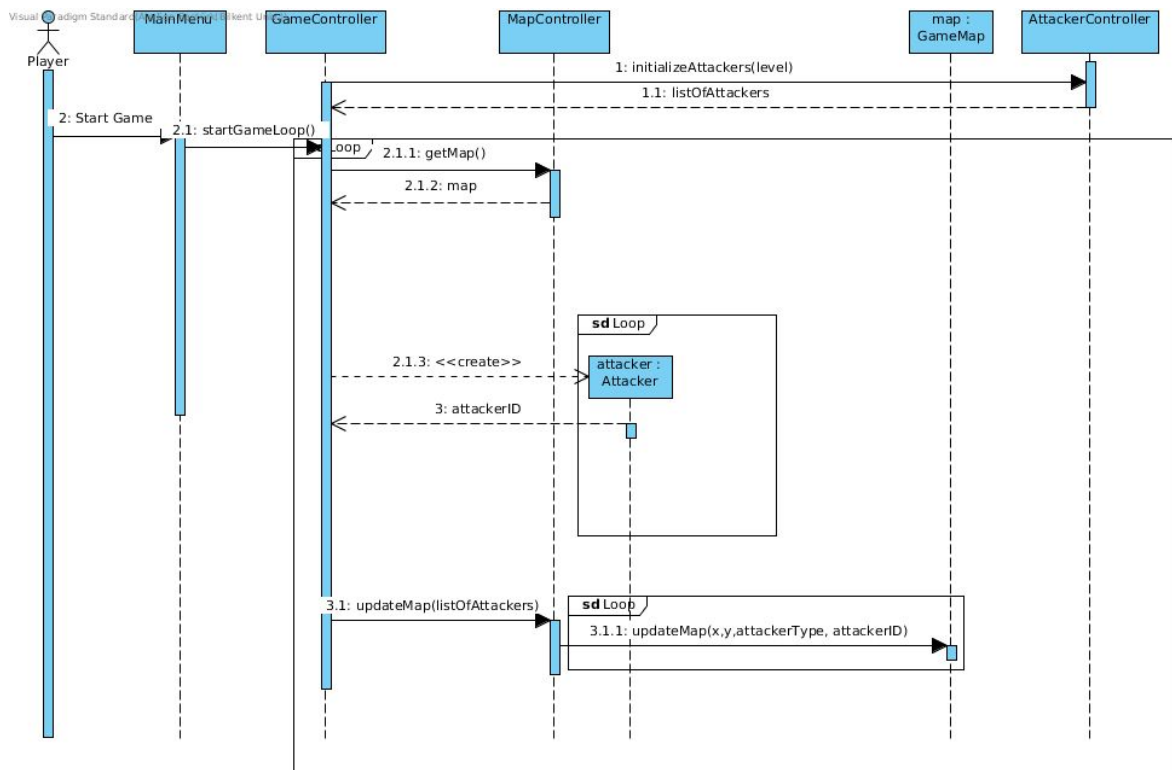
Also attack of a hero is similar to the attack of towers.

Scenario: Death of an attacker. This is similar to the tower attack scenario. This scenario is executed when the health of an attacker drops to zero or below by the attack of a tower. Each attacker stores its health and controls this attribute. If it is equal or less than zero, code for the death is initialized. GameController and ResourceController are notified so that they can make updates. Death of a hero is similar to the death of an attacker.

4.2.2 Activity Diagram

Wait for user to select a level

Check if the selected level has been played before

If user selects an item

Start the game → Item selected

Level has been played before

Display the highest score of the current level by stars

Check if there are enough golds to purchase the item

Not enough gold → Item not purchased

Enough gold

Tower shoots the attacker ← Item purchased

Sell Items

If it is a tower

Check if attacker has life points left

Check if item is a hero

No life points

Item is hero

Attacker dies

Attacker gets through the base castle → Check if the player has life points left

Hero fights the attackers

No

Check if all attackers are dead

The player wins, game is over

The player loses, game is over

Hero is dead

No life points left

Yes

4.3 Object and Class Model

GameController
+gameLoop()
+updateTime()
+notifyDeath()

Listens player inputs
handles inputs

InputController

TowerListController
-imageForTower1
-imageForTower2
-costOfTower1
-costOfTower2

AttackerController
-listOfAttackersForLevel1
-listOfAttackersForLevel2
-listOfAttackersForLevel3
-listOfAttackersForLevel4
-listOfAttackersForLevel5
+initializeAttackers(level)

provides frame

WindowController
+getGameWindow()
+getGraphics()
+getTowerListWindow()

ResourceController
-playerGold
+updatePlayerGold(amount)

MapController
-map
-pathColour
-nonPathColour
+loadMap(level)
+drawMap()
+isTowerPlaceable(x, y)
+updateMap(x, y, objectType, objectID)
+updateMap(ListOfObjects)
+findAttackersInTheTowerRange(towerID)

Menu

MainMenu

PauseMenu

GameMap
-tiles
+updateMap(x, y, objectType, objectID)

1..*

GameObject
+draw(image, posX, posY)
+getObjectLocation()

Tower
-damage
-rateOfFire
-range
-attackType
-image
+attack(attackerID, damage)

SinlgeAttackTower
+attack(attackerID, damage)

AreaAttackTower
+attack(attackerID, damage)

Hero
-damage
-health
-rateOfAttack
-image
+attack(attackerID, damage)
+notifyDeath()
+initializeDeath()

HeroType1

HeroType2

Attacker
-speed
-health
-image
-bounty
+move()
+initializeDeath()
+notifyDeath()

AttackerType1

AttackerType2

AttackerType3

GameController class is the main class which interacts with other classes to control the game.

ResoureController class stores the resource information of the player. // TODO: storing may be persistent across the levels.

WindowController class provides the game window and graphics objects to the GameController.

InputController class listens to user mouse inputs.

TowerListController class controls the list of towers in the right menu. It allows use of towers according to level and stores their cost.

AttackerController class stores the information about the attackers. This class stores information such as the number and types of attackers. Scenario:

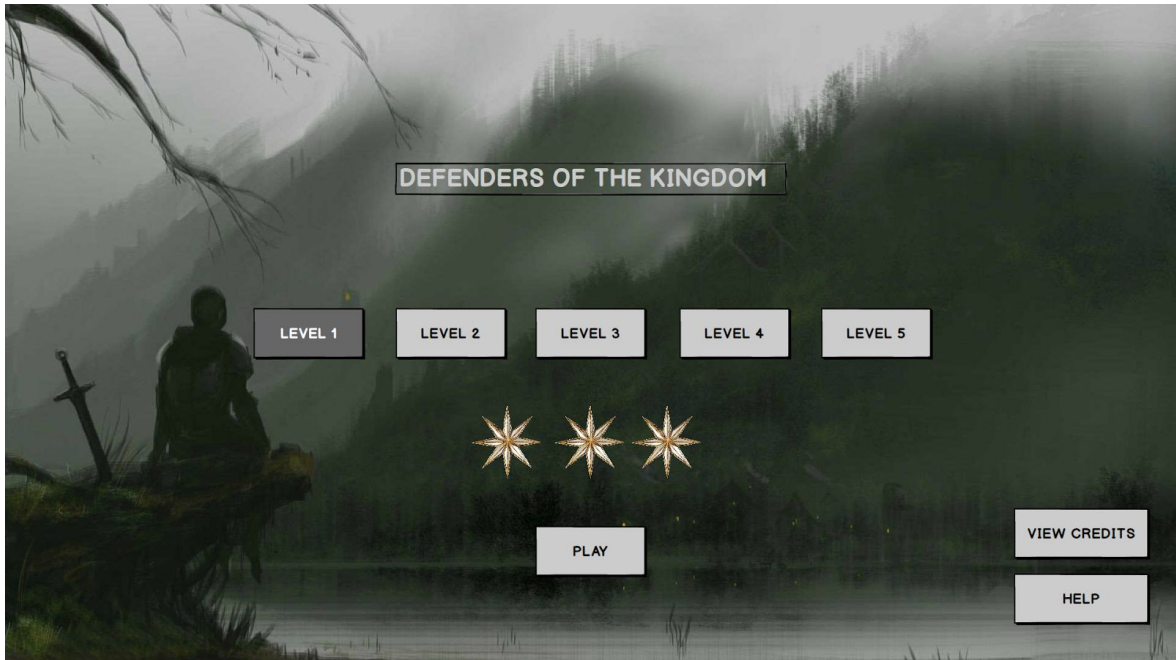**MapController** class handles the object placement and map operations of the game.

5 User Interface

5.1 Navigational Path
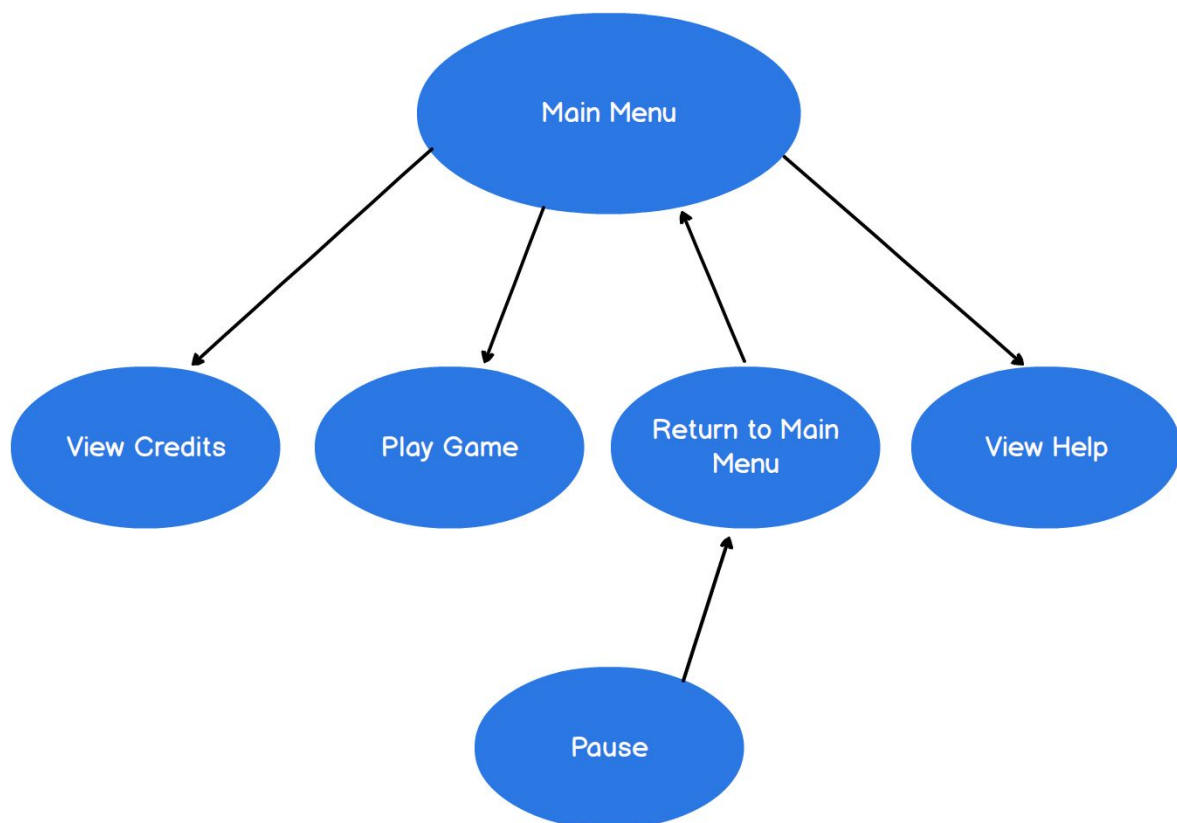
5.2 Screen Mock-ups

5.2.1 Main Menu

When the player runs the game. A menu screen will be shown at first. In the menu, the player can choose between the levels, see with how many stars (s)he passes it before and play the desired level. The player can see the credits of the game and also get help about the game.

5.2.2 View Credits

When the player chooses to see the credits of the game, s(he) will be able to see contact information of the developers of the game.

5.2.3 Help
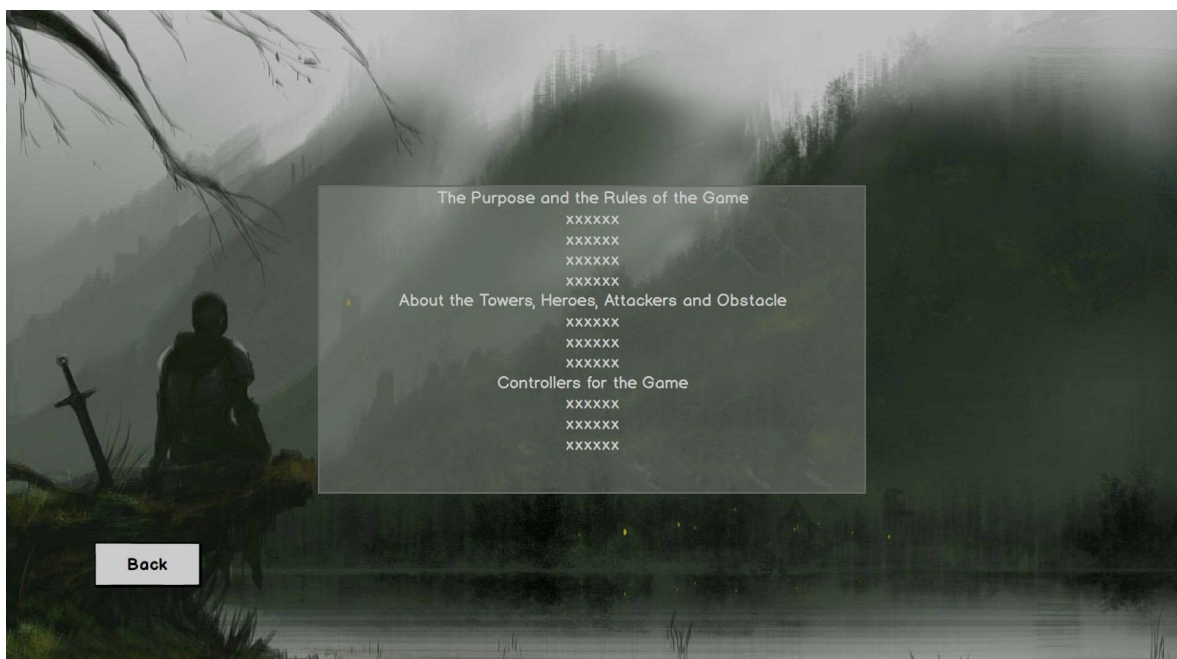
If the player wants to get help about the game s(he) can see the information about the purpose of the game, about the towers types, attackers, heroes and an obstacle. Also the player will be able to learn about controllers of the game.

5.2.3 Game View

This is basically how a particular level will look like. There will be a path and attackers will follow that path. The player will be able to select the desired towers and place them near to the path. Also the player can buy heroes. If the player wants to stop the attackers for a while, (s)he can put an obstacle to the path of the attackers. Player can also pause the game and resume.

5.2.4 Success Pop-up
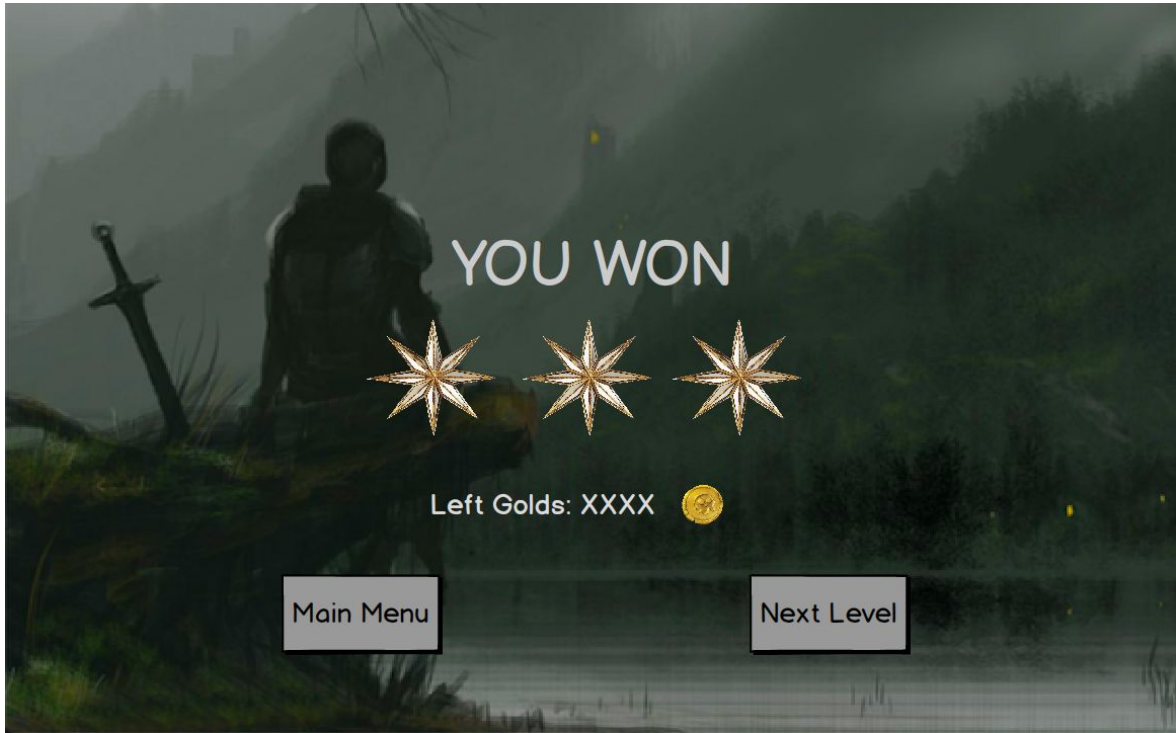
When the level is finished the player will be able to see a success pop-up like the figure above. The player can either proceed to the next level or return to the main menu.

6 Important Decisions In Overall Analysis

- We have decided that the attackers can not make an attempt to the towers.
- When the player purchases a hero, the hero stands by the base castle and fights the coming up attackers. The hero is not navigated by the player.
- We have decided to make objects such as attackers and heroes destroy themselves. In other words, they call their destructors and required methods when they die.

7 Conclusion

In this report, we have introduced and analyzed our tower defense game, Defenders of the Kingdom. There are 5 essential categories in the report: introduction, overview of the game, requirement specifications and the system model parts and the user interface mock-ups.

In introduction and overview sections, we have mainly introduced and described the core elements of the game. The attackers, and the defence mechanisms are defined in detail.

For the Requirements Specifications, we have analyzed functional and nonfunctional requirements. We have specified the game rules and instructions. The specifications that we have done in the analysis report will hopefully help us in the design and implementation parts.

For the modeling part, we have prepared Use Case Diagrams, Sequence Diagrams, Activity Diagram and also the Object and Class Diagrams (UML). Especially, the Object and Class Diagrams will lead us in the implementation part of the project.

Lastly, we have designed the UI mock-ups.

8 References

1 "GemCraft". *Armor Games*. Accessed 23 Sep 2017.
http://armorgames.com/play/1716/gemcraft

Image References
Towers: https://www.artstation.com/artwork/xB29Y
Item Background: http://weknowyourdreams.com/single/grey/grey-08
Obstacle: https://www.artstation.com/artwork/q5r0R
Gold: https://www.artstation.com/artwork/XvJRy
Path: http://www.claytus-towerdefense.com/images/tuto_1/8_paths_simple.jpg
Background:
https://i.pinimg.com/originals/97/e3/8f/97e38fb563124715573cab16b83cf6e2.jpg
Star:
https://cdn.decorpad.com/photos/2016/04/12/decorative-star-antique-brass-framed
-mirror.jpeg
Attacker # 1: https://www.artstation.com/artwork/nxAYe
Attacker # 2: https://www.artstation.com/artwork/rwke2
Hero # 1: https://www.artstation.com/artwork/8nwkq
Hero # 2:https://www.artstation.com/artwork/D1xX9