Requirements Analysis Document

Mattias Torstensson, Karl Wikström, Agnes Mårdh, Luka Mrkonjic

May 2017

Version: 2

Date: 26-05-2017
Revised by: Mattias Torstensson
This version overrides all previous versions.

1   Introduction

In this modern day and age games have inclined towards a more and more casual and wider audience than ever before. Nowadays games more than often serve as a tool for defeating boredom and wasting time while for example waiting for the bus or the lecture to start, which is Hook Hero's main goal. In simple terms the application's purpose is to provide quick entertainment by challenging the user to beat their previous record in the game, similar to an arcade game. The game is easy to learn but hard to master, which provides a great user experience for both experienced gamers as well as new ones. The target audience for the game are people who like challenging themselves in games via trial and error. Due to the easy controls of the game it's suitable to people of all ages. Since one "playthrough" of the game probably won't last longer than a couple of minutes due to the constantly increasing difficulty, the game can practically be picked up and played whenever the user feels like it.

1.1   General characteristics of the application

The application will be a offline singleplayer desktop game, that will be runnable on Mac/Windows/Linux operating systems.

The game will be a sidescrolling platformer of the endless runner variety. As such the game will feature a single player character that is always running forward relative to the world. The world will consist of endless procedurally generated obstacles that the player has to overcome by using the character's available movement abilities: jumping aswell as attaching its hook to an object in the world and swining across gaps with it. The player will accumulate score over time until they trigger the game's fail state either by crashing into something

in the world or by falling off it. There is no win condition in the game, it will keep running until the fail state triggers.

1.2 Definitions, acronyms and abbreviations

- Player: The player playing the game, the player controls the Character.

- Character: The character that the player controls.

- World: The environment in which the character exists.

- Side scroller: A video game genre in which the gameplay action is viewed from a side-view camera angle.

- Hook: A tool a character uses to swing from the ceiling.

- In-game: When the player is actively playing the game, in other words while the game is unpaused and the player is not in a menu.

2 Requirements

2.1 User interface

2.2 Functional requirements

- In the main menuy the player will be able to:

  - Start the game
  - View previous highscores
    * Close the view of previous highscores

- In-game the player will be able to:

  - Jump
  - Use grappling hook
    * Detach grappling hook.
  - Trigger a game over
    * By colliding with an obstacle
    * By falling off the map

- In the game over screen the player will be able to:

  - View and save their score
  - Restart the game
  - Go to the main menu

Use cases sorted by priority, highest priority first:

- Jump

- Use grappling hook

- Trigger a game over

- View and save their score

- Restart the game

- Main Menu

- Go to the main menu

2.3   Non-functional requirements

2.3.1   Usability

The game is supposed to be easy to pick up and play for anyone and as such it should not require anything other than the short list of controls available in-game in order to be played.

2.3.2   Reliability

N/A

2.3.3   Performance

Input from the player should appear to be instantaneous at all times and the game should not suffer from any noticeable stuttering.

2.3.4   Supportability

The game should be runnable on Windows, Mac, and Linux operating systems.

The application should have automated tests for all parts that are viable for automated testing, any part of the application that doesn't have automated testing should be tested manually.
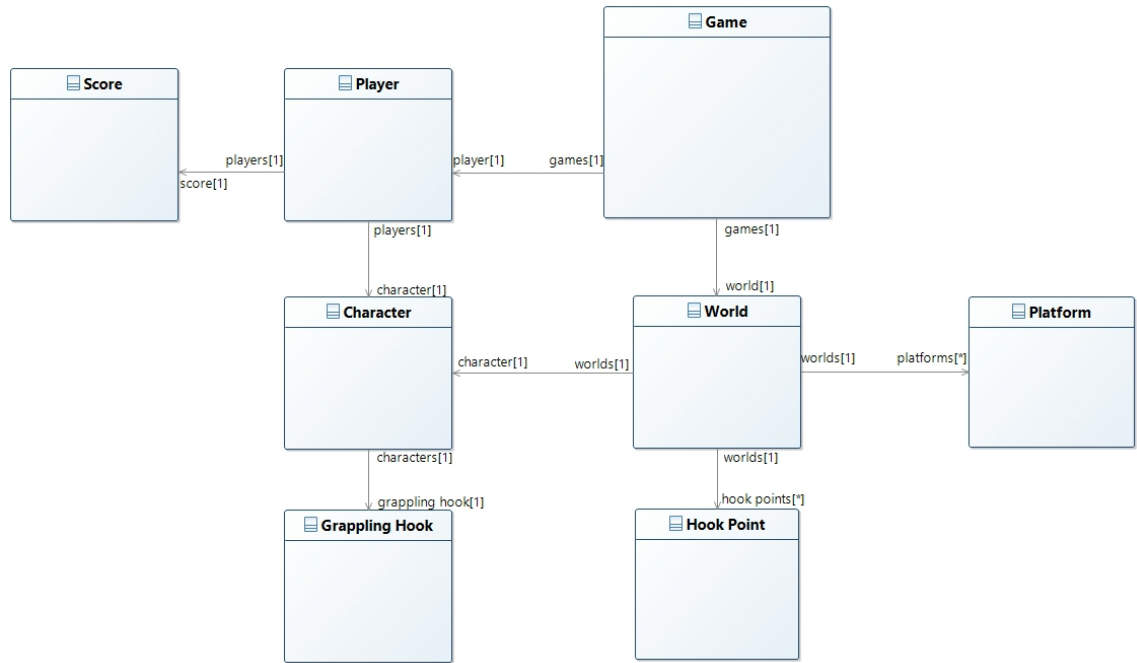
2.3.5   Legal

TODO: https://github.com/libgdx/libgdx/blob/master/LICENSE

3 Use cases

3.1 Use case listing

4 Domain Model

4.1 Class responsiblities

- Game, the overall representation of the game.

- Player, the person playing the game.

- Score, the score of the player has in the current session of the game.

- Character, the character the player is playing as.

- Grappling Hook, the grappling hook the character can use to get around.

- World, the world the character is in. Contains procedurally generated Platforms and Hook Points.

- Hook Point, a part of the world that the character can attach its hook to.

- Platform, a platform in the world that the character can stand on top of, or collide with the side of.

5   References