

# Requirements and Analysis Document for GruppTvå

Version: 1.0

Date: 2017-03-27

Author: Luka Mrkonjic, Agnes Mårdh, Mattias Torstensson, Karl Wikström

This version overrides all previous versions.

## 1 Introduction

*Background explaining why this application is needed (besides mandatory in course). What's the problem addressed (use imagination)? What will it do? Who will benefit/use from this application? In what situation will the application be used? Define the application. General characteristics of application.*

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.

*//Combine following points into one text segment when done*

*-Why needed?*

*-What problem does it address?*

*-What does it do?*

*-Who will use it?*

*-When will they use it?*

*-Define application:*

## 1.2 Definitions, acronyms and abbreviations

*Create word list to avoid confusion.*

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

*Sketches, drawings and explanations of the application user interface (possible navigation).*

### 2.2 Functional requirements

*What will the user be able to do ? Write a list of use case names (id's) in the language of the customer. The specific flows for each use case is recorded below. Specify a use cases in priority order.*

In the main menu the player will be able to:

1. Start the game
2. View previous highscores
  - a. Close the view of previous highscores

In-game the player will be able to:

1. Jump
2. Use grappling hook
  - a. Detach grappling hook
3. Collect points
4. Pause the game
  - a. Unpause the game
  - b. Close the game
5. Trigger a game over
  - a. By colliding with an obstacle
  - b. By falling off the map

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

1. Save their score
2. View previous highscores
3. Restart the game
4. Close the game

## 2.3 Non-functional requirements

*Any special considerations besides functionality? Usability, reliability, performance, supportability, legal, implementation, ... NOTE: Testability mandatory (must have tests)*

### 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 application should run on the three major desktop operating systems, Mac, Windows and Linux.

-tests

### 2.3.5 Legal

TODO

## 3 Use cases

*An UML use case diagram*

### 3.1 Use case listing

*Use case texts (using the use case template)*

## 4 Domain model

*An UML class diagram.*

### 4.1 Class responsibilities

*Explanation of responsibilities of classes in diagram*

## 5 References