

# Analysis Document for Project\_Titan

Version: Alpha 0.1

Date: 25/3 2015

Author: Martin, Oliver, Rasmus

## **1 Introduction**

- 1.1 Purpose of application**
- 1.2 General characteristics of application**
- 1.3 Scope of application**
- 1.4 Objectives and success criteria of the project**
- 1.5 Definitions, acronyms and abbreviations**

## **2 Requirements**

- 2.1 Functional requirements**
- 2.2 Non-functional requirements**
  - 2.2.1 Usability**
  - 2.2.2 Reliability**
  - 2.2.3 Performance**
  - 2.2.4 Supportability**
  - 2.2.5 Implementation**
  - 2.2.6 Packaging and installation**
  - 2.2.7 Legal**
- 2.3 Application models**
  - 2.3.1 Use case model**
  - 2.3.2 Use cases priority**
  - 2.3.3 Domain model**
  - 2.3.4 User interface**
- 2.4 References**

## **APPENDIX**

**GUI**

**Domain model**

**Use case texts**

# Introduction

## 1.1 Purpose of application

We are an independent game developer team that are going to create an application for PC computers. Our purpose is to build the game and distribute it online for a charge and make a profit. Since we are in the entertainment business, creativity in form of theme, style, interaction and music is of high priority to us but as for any company, high profit margin is the goal.

The target audience is everyone from the typical gamer to the ordinary person that just want to have fun. Even if our main group are people which enjoy “Action platform games” which is the category this game fall into.

## 1.2 General characteristics of application

This application is used to have fun and to challenge yourself. It is a good time-killer because it can be very time consuming to get the best scores in this game. The game is intended for all people who just want amusement and it is mostly played at home or when there is nothing to do, generally anywhere where a game console is available.

## 1.3 Scope of application

The application will contain:

- a singleplayer campaign with story mode
- fast paced jump shooter
- experience gaining and item collecting
- character customization with different gear and appearance

The application will not contain:

- multiplayer
- downloadable content (DLC)
- trademarked characters

## 1.4 Objectives and success criteria of the project

The primary objective is to get a function game with all the basics to make it as much enjoyable as possible. The standard and gameplay features will definitely make it for the

finished product. We strive to have 5 maps with at least one boss for each map. The database for the items and creatures will be big enough for just 5 maps which is around 20 items and 10 creatures. If there is time left we will implement more maps, make the databases bigger, and implements more features(multiplayer, easter eggs).

## **1.5 Definitions, acronyms and abbreviations**

JS2k = JumpShooter2000

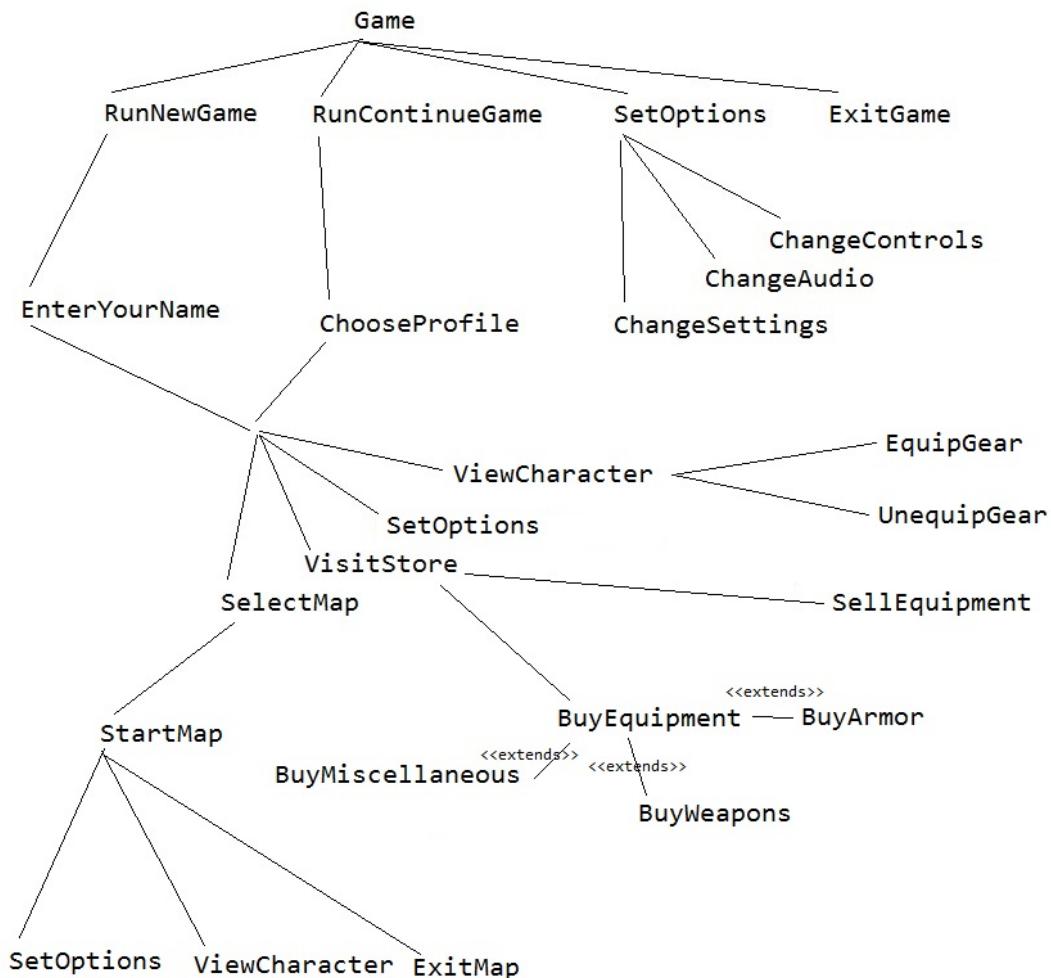
GUI - Graphical User Interface

Java - Programming Language

# Requirements

## Application models

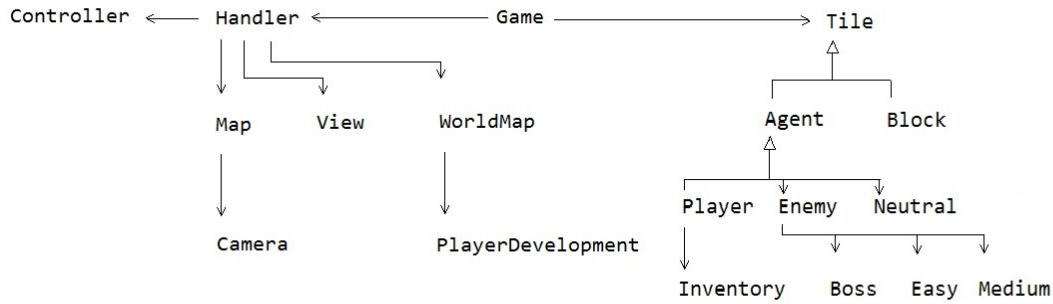
### 2.3.1 Use case model (Lägga till meny)



### 2.3.2 Use cases priority

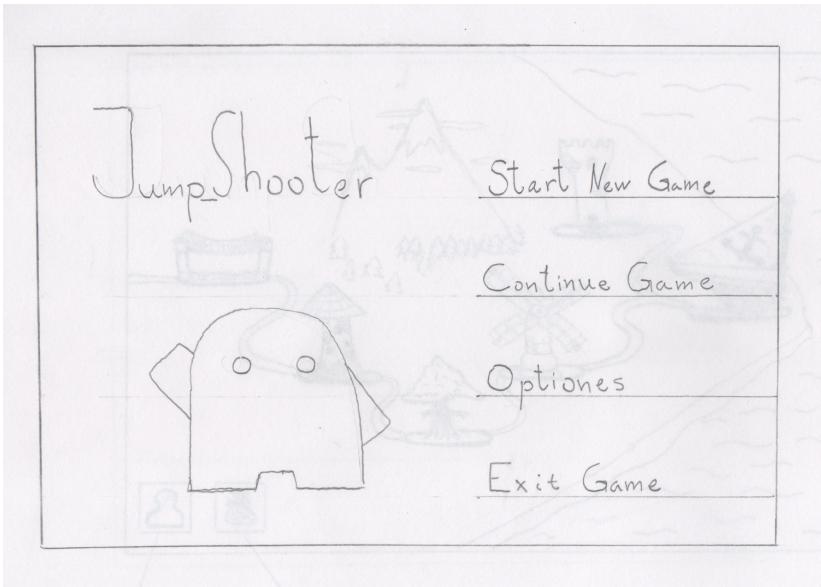
- High:** RunNewGame, RunContinueGame, EnterYourName, ChooseProfile, SelectMap, StartMap
- Mid:** VisitStore, ViewCharacter, BuyEquipment, SellEquipment, EquipGear, UnequipGear, ExitGame, ExitMap,
- Low:** SetOptions, ChangeSettings, ChangeAudio, ChangeControls, BuyMiscellaneous, BuyWeapons, BuyArmor

### 2.3.3 Domain model



Early diagram showing relations between different parts of the game. The game is built upon two core classes, Handler and Tile. The Handler controls the underlying engine and runs the game. The Tile Class is the building blocks of the visual components of the game, including characters and their functionality such as AI-behaviour or Players Inventory.

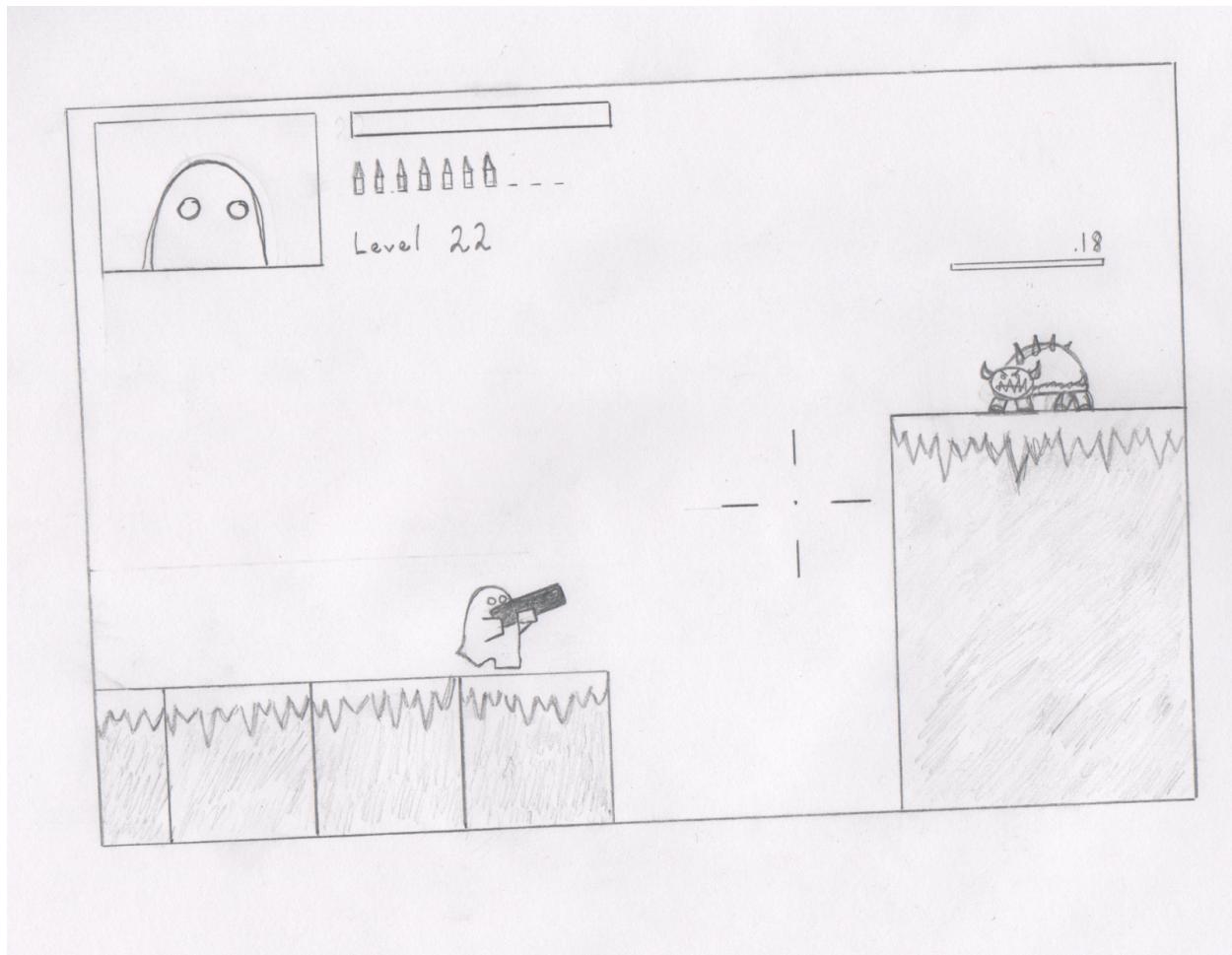
## 2.3.4 User interface



**Start Menu**  
Providing you with the alternatives to start a new game, continue on a previous saved one, change some settings or exit the application.



**World Map**  
Choose which level to play, or go to the shop to buy items or upgrade your character and view your inventory



First graphical view of actual gameplay. Stages contains of short to medium sequences of platform jumping and shooting action. Once stage is completed a window displaying result statistics in form of experience and currency gain will show. After the stage is completed the player will be returned to the world map.