# 1 Introduction

## 1.1 Design goals

The design must follow a clear MVC-pattern, to facilitate eventual switches of GUI and game engine. This means that the model will only contain the game logic and no graphics whatsoever, thus becoming completely independent from the view and the controller. The design must be testable, which means that it should be possible to isolate modules and classes for test. For usability see RAD.

## 1.2 Definitions, acronyms, and abbreviations

* GUI, graphical user interface
* Java, platform independent programming language.
* MVC, Model-View-Controller.

# 2 System design

## 2.1 Overview

The application will use an MVC model based on the jMonkeyEngine framework. The control layer will call the model for information every update frame and use it to update the graphics via the view layer.

### 2.1.1

### 2.1.2

### 2.1.3

### 2.1.4

## 2.2 Software decomposition

### 2.2.1 General

### 2.2.2 Decompositions into subsystems

### 2.2.3 Layering

### 2.2.4 Dependency analysis

## 2.3 Concurrency issues

## 2.4 Persistent data management

## 2.5 Access control and security

## 2.6 Boundary conditions

# 3. References