

Requirements and Analysis Document for Maze (RAD)

1. Introduction

1.1 Purpose of the system

The aim of the project is to create a computer game which consists of a player getting through a labyrinth.

1.2 Scope of the system

Only one player at a time can play.

The application does not save half-finished games.

1.3 Objectives and success criteria of the project

It should be possible to play three maps with increasing difficulty.

1.4 Definitions, acronyms, and abbreviations

1.5 References

1.6 Overview

2. Requirements

2.1 Functional requirements

The player should be able to:

1. Make a new player (saving name, character type, and high score per level)
2. Load a previous player
3. Play the game with three different levels
4. See all-time high scores.

2.2 Non-functional requirements.

2.2.1 Usability

High priority.

2.2.2 Reliability

NA

2.2.3 Performance

TODO

2.2.4 Supportability

TODO

2.2.5 Implementation

The application will be implemented in Java.

2.2.6 Packaging and installation

TODO

2.2.7 Legal

We do not have any lawyers on the team.

2.3 Application models

2.3.1 Use case model

See separate page

2.3.2 Use cases written/priorities

For written description, see separate page. Use cases in order of priority?

1. Move
 - a. Answer questions
 - b. move onto a wall
 - c. Teleport / enter teleport square
 - d. Feed monster / want to enter monster square
 - e. Unlock doors / enter door square
 - f. Open chest / enter chest square
2. Choose new save slot
3. Load game / choose used save slot
4. View highscore
5. Return to main menu

2.3.3 Analysis model

See separate page

2.3.4 User interface

See separate page

2.4 References

None

