**Software Requirements Specification**

**for**

**<Trivia Maze>**

**Version 2.0 approved**

**Prepared by < Sado Iman & Rohit Ark>**

**<TCSS 360 Team 3>**

**<June 7th, 2024>**

**Table of Contents**

**Table of Contents ii**

**Revision History i**

**1. Introduction**

1.1 Purpose

1.2 Document Conventions

1.3 Intended Audience and Reading Suggestions

1.4 Project Scope

1.5 References

**2. Overall Description**

2.1 Product Perspective

2.2 Product Features

2.3 User Classes and Characteristics

2.4 Operating Environment

2.5 Design and Implementation Constraints

2.6 User Documentation

2.7 Assumptions and Dependencies

**3. System Features**

3.1 Game Mechanics

3.2 Character and Monster Mechanics

3.3 Level Mechanics

**4. External Interface Requirements**

**5. Other Nonfunctional Requirements**

**6. Other Requirements**

**Appendix A: Glossary**

**Appendix B: Analysis Models**

**Appendix C: Issues List**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Rohit Ark | 4/17/2024 | Initial Draft | 1.0 |
| Rohit Ark | 6/7/2024 | Final Documentation | 2.0 |

# **1.** **Introduction**

## **1.1** **Purpose**

This SRS describes the software requirements for release 2.0 of the Trivia Maze Game. This SRS covers the entire system of the Trivia Maze Game including the mechanics, interfaces, performance, requirements, safety requirements, as well as other aspects relevant to the entire system. This document is to be used by the developers of the project that will implement and also verify the proper functioning of the system.

## **1.2** **Document Conventions**

Standard formatting conventions for SRS documents have been followed in this document.

## **1.3** **Intended Audience and Reading Suggestions**

This document is intended for various types of readers, including developers, project managers, testers, and documentation writers. It contains information about the software requirements, organization, and scope of the project. It is recommended to begin reading with the overview sections and proceed through the sections sequentially. Certain sections may be more relevant depending on the reader, for example users may find section 1 more useful for them while developers and testers may find sections 2 and 3 more relevant for them.

## **1.4** **Project Scope**

The Trivia Maze game is an adventure game featuring a character tasked with reaching the end of the maze through answering questions. The goal is to create a functional, well-designed game that incorporates the given specifications outlined in the rubric and utilizes object oriented principles as well as design patterns such as Model, View, Control.

## **1.5** **References**

There were no references used in this document.

# **2.** **Overall Description**

## **2.1** **Product Perspective**

*This project was created in order to help students learn to work in a group and effectively communicate and cooperate with each other. Along with learning how to work effectively as a team, new concepts will also be explored to further our education as computer scientists. These include, working on GitHub, using Pivotal Tracker, building SRS documents (like this one), working with SQLite, along with improving our overall efficiency as programmers.*

## **2.2** **Product Features**

*The major features of this project include everything that was described in the product perspective, along with working in Intellij, using Java, creating mechanics within the game to deal with answering questions, healing, movement mechanism, and saving/loading the game.*

## **2.3** **User Classes and Characteristics**

*There are 3 packages that are going to be created in order to have this application run correctly. There is the model package, view package, and the controller package. Each package will have multiple different class to help ensure abstraction.*

*The model package will help manage the behavior and data of the application domain it also responds to requests for information about its state. This ranges from room/door classes to the question factory.*

*The view package will be used to show visuals to the user. This will range from the maze panel itself to the question panel which will display the question, and the arrow panel which is how you will move have buttons to move your character.*

*The controller package will receive input and initiate a response by making calls on model objects.*

## **2.4** **Operating Environment**

*The operating system that we will be using is MacOS.*

## **2.5** **Design and Implementation Constraints**

*There is no design or implementation consternates on this project.*

## **2.6** **User Documentation**

*There will be a simple how to menu in the help section of the game explaining the different heroes, and how to play the game.*

## **2.7** **Assumptions and Dependencies**

*N/A*

# **3.** **System Features**

## **3.1** **Game Mechanics**

3.1.1 Description of general game mechanics

* Start new game
* Save game
* Load game
* Answer question to move to the next door
* Questions will be:
  + Short answer
  + Multiple choice
  + True/False
* Potions will give hints of what surrounding doors questions are

3.1.2 Description of in game mechanics

* Ability to answer questions
* Lock door if answer is false
* Find in game items
* Get to the end to escape maze

3.1.3 Description of in game items

* Health potion (acts as hint)

3.1.3 Stimulus/Response Sequences

Stimulus: Player moves in a selected direction

Response: Player will be prompted with a question to answer

Stimulus: Player answer question correctly

Response: Player is allowed to move into that room

Stimulus: Player answer question incorrectly

Response: Door becomes locked and player will be unable to enter door all game.

Stimulus: Player enters a room with a potion

Response: Pop-up will appear showing what each question around him is.

Stimulus: Player answers every question around them wrong

Response: Pop-up will appear saying you lost the game.

Stimulus: Player makes it to the end and answers final question correctly

Response: Pop-up appears indicating that you won the game

# **4.** **External Interface Requirements**

## **4.1** **User Interfaces**

N/A

## **4.2** **Hardware Interfaces**

N/A

## **4.3** **Software Interfaces**

N/A

## **4.4** **Communications Interfaces**

N/A

# **5.** **Other Nonfunctional Requirements**

## **5.1** **Performance Requirements**

N/A

## **5.2** **Safety Requirements**

N/A

## **5.3** **Security Requirements**

N/A

## **5.4** **Software Quality Attributes**

N/A

# **6.** **Other Requirements**

N/A

**Appendix A: Glossary**

N/A

**Appendix B: Analysis Models**

N/A

**Appendix C: Issues List**

N/A