Video Game Trivia Maze

Jake McGillicuddy, Jonathyn Komorita,

Joel Kalich

**Software Requirements**

**Specification Document**

**Version: 1** **Date:02/07/2020**

# Table of Contents

[**Table of Contents**](#_heading=h.gjdgxs) **2**

[**1. Introduction**](#_heading=h.30j0zll) **4**

[1.1 Purpose](#_heading=h.1fob9te) 4

[1.2 Scope](#_heading=h.3znysh7) 5

[1.3 Definitions, Acronyms, and Abbreviations.](#_heading=h.2et92p0) 5

[1.4 References](#_heading=h.3dy6vkm) 6

[1.5 Overview](#_heading=h.1t3h5sf) 6

[**2. The Overall Description**](#_heading=h.4d34og8) **7**

[2.1 Product Perspective](#_heading=h.2s8eyo1) 7

[2.2 Product Functions](#_heading=h.17dp8vu) 7

[2.3 User Characteristics](#_heading=h.3rdcrjn) 8

[2.4 Constraints](#_heading=h.lnxbz9) 9

[2.5 Assumptions and Dependencies](#_heading=h.35nkun2) 9

[**3. Specific Requirements**](#_heading=h.44sinio) **9**

[3.1 External Interfaces](#_heading=h.2jxsxqh) 9

[3.2 Functions](#_heading=h.z337ya) 11

[3.3 Logical Database Requirements](#_heading=h.3j2qqm3) 20

[**4. Change Management Process**](#_heading=h.1y810tw) **20**

[**5. Document Approvals**](#_heading=h.4i7ojhp) **20**

# **1. Introduction**

## **1.1 Purpose**

The purpose of this document is to provide a detailed description of the requirements for the Trivia Maze game. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval and referenced for developing the second version of the system for the development team.

## **1.2 Scope**

The Trivia Maze game is going to be a Java GUI-based game where a ‘player’ will navigate a nxn maze [where n is the length of the row/column], answering Trivia questions to move through each room in the maze. It will be free, and available for anyone who may wish to play it.

The Trivia Questions and ‘admin’ data will be stored in a SQLite database which will be located on a web-server.

## **1.3 Definitions, Acronyms, and Abbreviations.**

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Player | User who is using the application to play the Trivia Maze Game |
| Admin/Administrator | User who will be able to “sign-in” to add/delete trivia questions |
| GUI | Graphical User Interface |

**1.4 Overview**

The remainder of this document includes three chapters. The second chapter provides an overview of the system and functionality, and how the system interacts with other systems. This chapter will also introduce the system constraints and assumptions about the product.

The third chapter provides the requirements specification in detailed terms and a description of the different system interfaces. Different specification techniques are used in order to specify requirements more precisely for different audiences.

# 

# 

# 

# 

# 

# 

# 

# 

# **2. The Overall Description**

This section will give an overview of the whole system. The system will be explained in its context to show how the system interacts with other systems and introduce the basic functionality of the basic structure. Lastly, the constraints and assumptions for the system will be presented.

## **2.1 Product Perspective**

The game will consist of two parts: a Java GUI application that is the actual Trivia Maze game, and a web-based database that will be used to store the trivia questions

The application will need to communicate a database to store the Trivia Questions. All database communications will take place over the internet.

## **2.2 Product Functions**

The final product will be a GUI based trivia maze game. In the game, the player will begin at a starting point in the maze, and have to navigate through each room to eventually reach the finish line. To move from one room to another, the player will have to correctly answer a trivia question. Player moves and other actions will be taken in through the GUI.

There will be several different types of questions, such as multiple choice, true/false, short answer, and if we have time we will add questions that will ask you to identify sounds and/or portraits.

The admin account will be given a different screen from the player, and through this screen they will be able to add additional trivia questions into the database to be used in the game.

## 

## 

## **2.3 User Characteristics**

**2.3.1 Players:**

Players will be the users playing the game, but will not have access to adding new questions.

**2.3.2 Admin:**

Admins will be able to load new questions into the game

## **2.4 Constraints**

The game will be limited by the number of questions available. At some point, questions will begin repeating questions to the player if the board is big enough and there aren’t enough questions in the database.

An Internet connection will also be required in order to communicate with the database to load the questions.

## **2.5 Assumptions and Dependencies**

It is assumed that anyone playing the game will have a basic knowledge of video games in order to answer the trivia questions. It is also assumed that anyone running the application will be doing so on Windows, as there may be issues running on a different OS.

Another assumption is that anyone running the application will have an internet connection.

## 

## 

## 

## 

## **3. Specific Requirements**

## **3.1 Player Interface**

A player will start at a “Welcome” screen that will check if you want to enter as a “Player” or an “Admin”. Upon selecting “Player”, they will be taken to the game GUI. The GUI will have a section that will display the maze and currently player progress, a section that will hold the up, down, left, and right movement buttons, a section where the trivia question is displayed, and a dropdown menu bar so that we can exit back to menu, or exit the game completely.

**3.1.2 Administrator**

Upon starting up the game, the admin will be taken to the same welcome screen as the player. When they select the admin tab, they will be taken to a [log-in panel? Once their log in credentials are confirmed, they will be taken to their main screen] their main screen where they will be able to add questions to the database. They can select in a drop down menu what type of question to enter, and the appropriate question template will come up.

## **3.2 Functions**

3.2.1 Functional Requirement

ID: FR1

TITLE: Welcome Screen

DESC: The system will take the general user to a welcome screen

RAT: In order for the users to be able to use the application

DEP: None.

3.2.2 Functional Requirement

ID: FR2

TITLE: Admin Log-in Screen

DESC: The system shall allow a general user into the Admin Log-In screen

RAT: In order for the admin to log-in and use the application

DEP: FR1

3.2.3 Functional Requirement

ID:FR3

TITLE: Player Main Screen

DESC: When a general user clicks on the “Player” tag, it should take them to the game screen

RAT: So that the player can player the game

DEP FR1

3.2.4 Functional Requirement

ID:F4

TITLE: Player Move

DESC: When the player clicks the up, down, left, or right button, the player should be moved in that direction through the maze

RAT: In order for the player to traverse the maze

DEP: F1, F3

3.2.5 Functional Requirement

ID:F5

TITLE: Prompt Player with Question

DESC: After starting the game or moving to a new room, player will be given a new question to answer before continuing.

RAT: Provides the main function of the game

DEP: F3, F4

3.2.6 Functional Requirement

ID: F6

TITLE: Player Answers Question

DESC: After being prompted with a question, Player will click on an answer, or type in the short answer. They will be notified if Correct or not.

RAT: In order for the player to move on in the game, or not

DEP: F3, F5

3.2.7 Functional Requirement

ID: F7

TITLE: Player Exits

DESC: Player will be able to select “EXIT” from a dropdown menu bar to exit game

RAT: In order for the player to exit the game

DEP: F3

3.2.8 Functional Requirement

ID: F8

TITLE: Admin Main Screen

DESC: When logged in as an Admin, the game should take them to a screen where they will be able to add questions to the database

RAT: To keep an updated list of questions, and to add more in case there aren’t enough

DEP: F1, F2

3.2.9 Functional Requirement

ID:F9

TITLE: Admin Submit Question

DESC: On a button click, the application will pull and store the question and answers into the database

RAT: To allow the question to be registered into the system.

DEP: F8

3.2.10 Functional Requirement

ID:F10

TITLE: Dropdown Bar for Question Type

DESC: Allow the admin to select what type of question, and the appropriate question template will show up

RAT: To provide different types of questions besides just M/C or T/F

DEP: F8

3.2.11 Functional Requirement

ID: F11

TITLE: Dropdown Menu Item to Exit

DESC: In the File dropdown menu bar, there will be an Exit option

RAT: To exit the application

DEP: F8

3.2.12 Functional Requirement

ID:F12

TITLE: 100 character limit on submitted questions

DESC: The application will require a question to be less than or equal to 100 characters

RAT: Character limit to ensure questions are an acceptable length

DEP:F8,F9,F10

3.2.13 Functional Requirement

ID: F13

TITLE: 50 character limit on submitted answers from Admin

DESC: The application will require an answer from the admin to be less than or equal to 50 characters

RAT: Character limit to ensure answers are an acceptable length

DEP: F8,F9,F10

3.2.14 Functional Requirement

ID: F13

TITLE: 50 character limit on submitted short answers from Player

DESC: The application will require a short answer response from the player to be less than or equal to 50 characters.

RAT: Character limit to ensure answers are an acceptable length

DEP: F8,F9,F10

## **3.3 Database Requirements**

All questions submitted by the admin will be stored in a SQLite database.

Database will include tables for:

Questions

This table will include the questions and their assigned ID.

Multiple Choice Answers

This table will include the relevant question ID, 3 incorrect answers, and the correct answer.

Short Response Answers

This table will include the relevant question ID and the correct answer.

True False Answers

This table will include the relevant question ID and the correct answer.

Question Comments

This table will include the relevant question ID and two comments, for when the user gets the answer wrong or right. The comment for a wrong answer will always include the correct answer.

Tables may be added by the developers or modders for different question formats.

**4. Change Management Process**

Changes should be discussed with the team, then documented within the SRS document.

# 

# **5. Document Approvals**

|  |  |
| --- | --- |
| Use Case 1 | The software will allow submission of questions to the database |
| Actor | Admin |
| Use Case Overview | The admin logs into the application and begins to fill out the question template. They will fill out the question name, as well as the answers to the question. They will then have the option to submit it to the database. |

|  |  |
| --- | --- |
| Use Case 2 | The software will check player answer submissions |
| Actor | Player |
| Use Case Overview | When they player gets into the game, there will be a section with a trivia question pulled from the database. The player will then answer the question and submit their answer, which will be checked and tracked if correct or wrong |

|  |  |
| --- | --- |
| Use Case 3 | The software will track user movement through the maze |
| Actor | Player |
| Use Case Overview | The player will be allowed to move one room over when they answer a question correctly, using the up, down, left, or right buttons in the GUI to traverse the maze. |