Software Requirements Specification

for

Trivia Maze

Version 1.0 approved

Prepared by Aaron Young

Content by Bryce Fenske, Trae Rawls, Aaron Young

Cup-O Java

December 9, 2015

Copyright © 2002 by Karl E. Wiegers. Permission is granted to use, modify, and distribute this document.

Table of Contents

Table of Contents ..... **ii**
Revision History ..... **ii**
1. Introduction **1**
1.1 Purpose 1
1.2 Document Conventions 1
1.3 Intended Audience and Reading Suggestions 1
1.4 Project Scope 1
2. Overall Description **2**
2.1 Product Perspective 2
2.2 Product Features 2
2.4 Operating Environment 2
2.5 Design and Implementation Constraints 2
2.7 Assumptions and Dependencies 2
3. System Features **2**
3.1 Web Server 2
4. External Interface Requirements **4**
4.1 User Interfaces 4
4.2 Hardware Interfaces 4
4.3 Software Interfaces 4
4.4 Communications Interfaces 5
5. Other Nonfunctional Requirements **5**
5.3 Security Requirements 5
Appendix A: Glossary **6**
Appendix B: Analysis Models **6**
Appendix C: Issue List **8**

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

Create a maze traversable by the user from entrance to exit. The maze will be composed of rooms containing one or more doors. To move into a new room, the user must answer a trivia question. Incorrectly answered questions will lock that particular door. The game is won on completion of the maze, and lost when no path to the end exists.

1.2 Document Conventions

Throughout this document the terms...

Client Application refers to the installed application on the customer end

Web Portal refers to the website (www.aaronyoungproductions.com)

User refers to the end license user, the person, people, organization using the Client Application

Us, We refers to the developers

1.3 Intended Audience and Reading Suggestions

Users should jump to 2.3, 2.4, Section 4 for requirements for running this product.

Testers should pay attention to system requirements as well as Section 2 for normal operating behaviors.

All readers are encouraged to read on general use and system requirements.

1.4 Project Scope

Questions will be numbered, of multiple types (short answer, T/F, multiple choice) and stored in a SQLite database.
There will be a visual representation of the maze for the user to interact with (possibly multiple).
Variations on a theme are welcome (items to help the user, finite use powers, etc).
Min size of maze is for rooms (no max specified). Entrance/exit may be random or fixed at opposite ends.

User must be able to save/load (serialize game state)

Provide admin tools for database editing

Scrub user input

Unit Tests required

2. Overall Description

2.1 Product Perspective

This product is a new system consisting of a client side trivia game and a server side web portal and update server.

2.2 Product Features

The product will allow for a user to traverse a made made up of doors and corridors to reach an end point at which case the user will have won or move on to the next level. A user will also have the option of visiting the web portal to submit their own questions for other users to then have within their own version of the game.

2.3 Operating Environment

The Trivia Maze program is supported on the Windows 7, 8, 10 operating system.

The web portal is supported on Google Chrome, Microsoft Internet Explorer

2.4 Design and Implementation Constraints

It is up to the user to maintain and regularly check for updates as the system relies on outside and unpredictable inputs.

2.5 Assumptions and Dependencies

Dependencies: internet connection, sqlite, ascii panel

When the update server is down, the usability of the client application is unknown.

3. System Features

3.1 Web Server -- Appendix B 4

3.1.1 Description and Priority

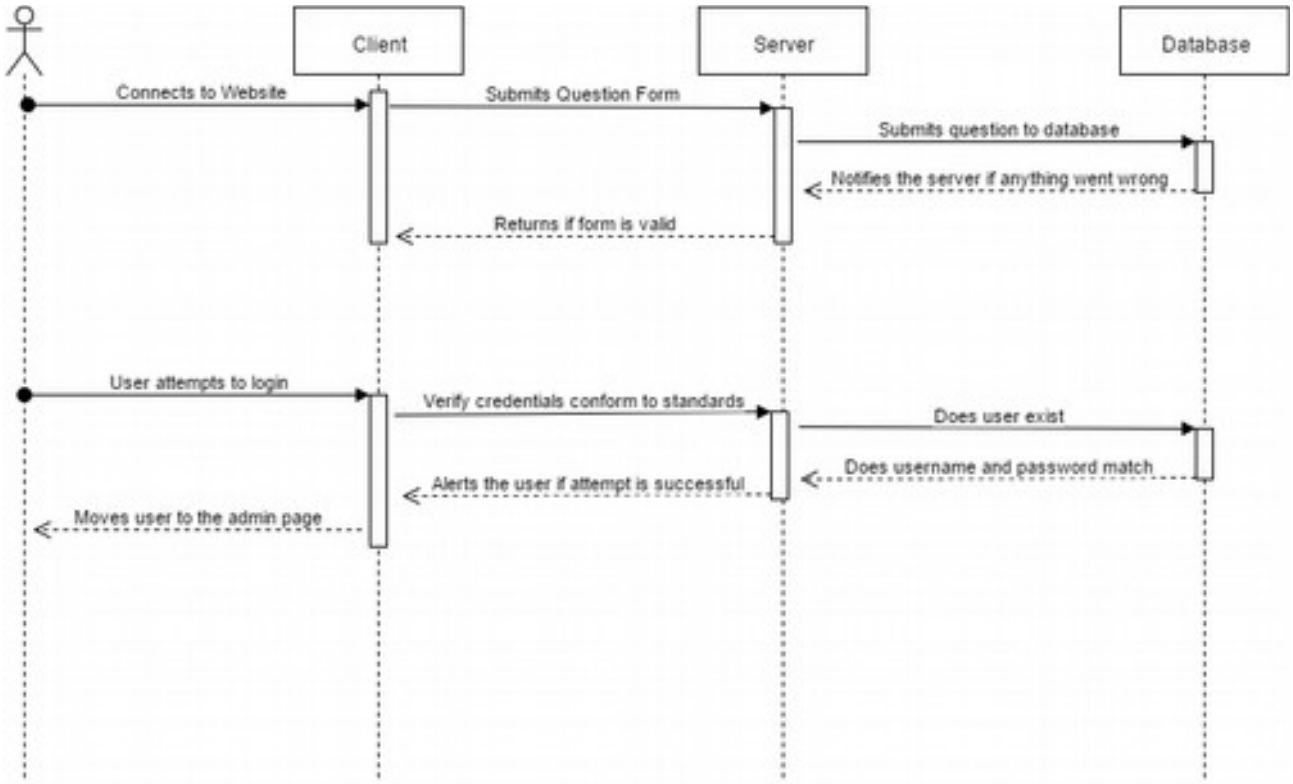
The web server allows for all client applications to be on the same question set while giving the users a way to add their own 'flair' to the game aspect.
Priority : 7
Costs a yearly fee while needing to be maintained while needing security to prevent malicious actions.

3.1.2 Stimulus/Response Sequences

User submits a question that is then added to the database. That question is then added to the client application's local Sqlite database to be used as a possibility to be selected for a normal play through as one of the questions to be answered.

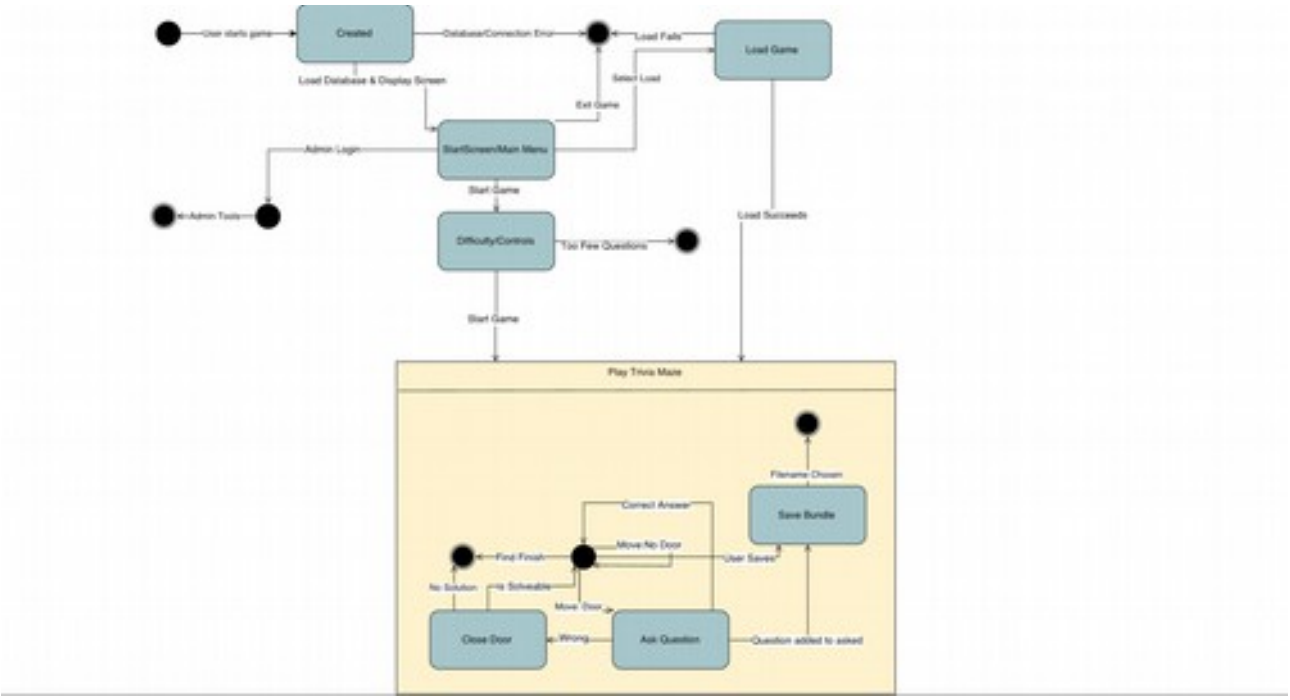
3.1.3 Functional Requirements

Internet Connection
Web Client : Mobile supported
Keyboard : Physical or Virtual



4. External Interface Requirements

4.1 User Interfaces



4.2 Hardware Interfaces

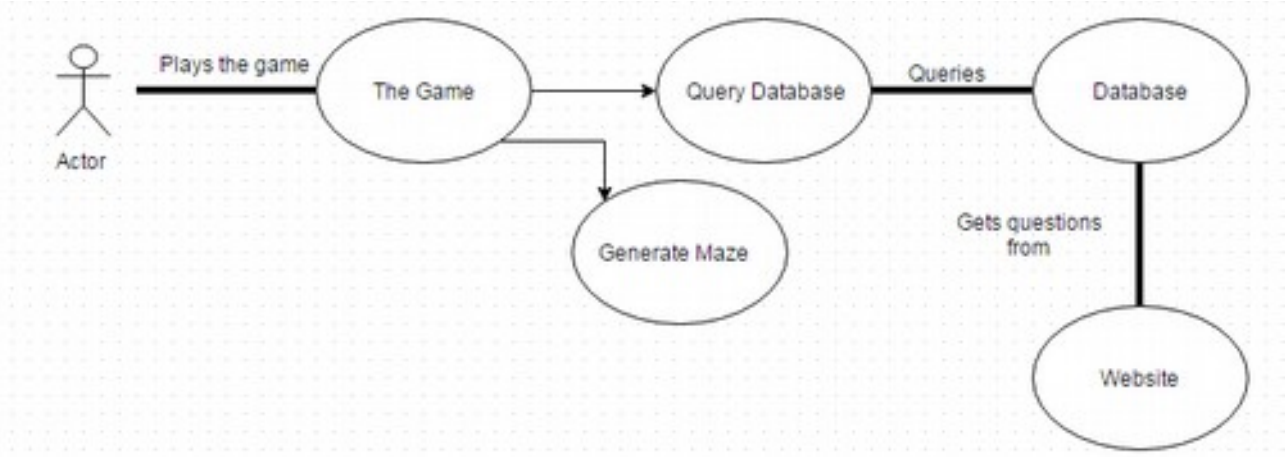
The client application supports exclusively alphanumeric keyboard entry and processing. No mouse interactions supported.

The web portal fully supports keyboard input to proceed with any activity the user would need to submit a question from start to finish. The web portal also allows for mouse input. The web portal does require a stable internet connection and interactions with unsupported browsers are unknown.

4.3 Software Interfaces

The client application uses a Sqlite database locally while connecting to an update server. The client application relies on Sqlite v.3.8.11.2, AsciiPanel v.1.0, and jsoup 1.8.3.
The client application will be using a HTTP request to pull questions from an update server to keep the local Sqlite database populated and up to date. No sensitive data of any kind is transmitted.
Appendix B 1-3

Name: Play the Game
Description: Traverse the selected maze and attempt to reach the winning tile.
Primary Actor: User
Main Scenario: 1. Start the game
2. Choose a difficulty
3. Traverse the maze
4. Answer questions as they appear
5. Either reach the end or lose your path



4.4 Communications Interfaces

HTTP : No secure traffic is used so minimal to no encryption is needed

5. Other Nonfunctional Requirements

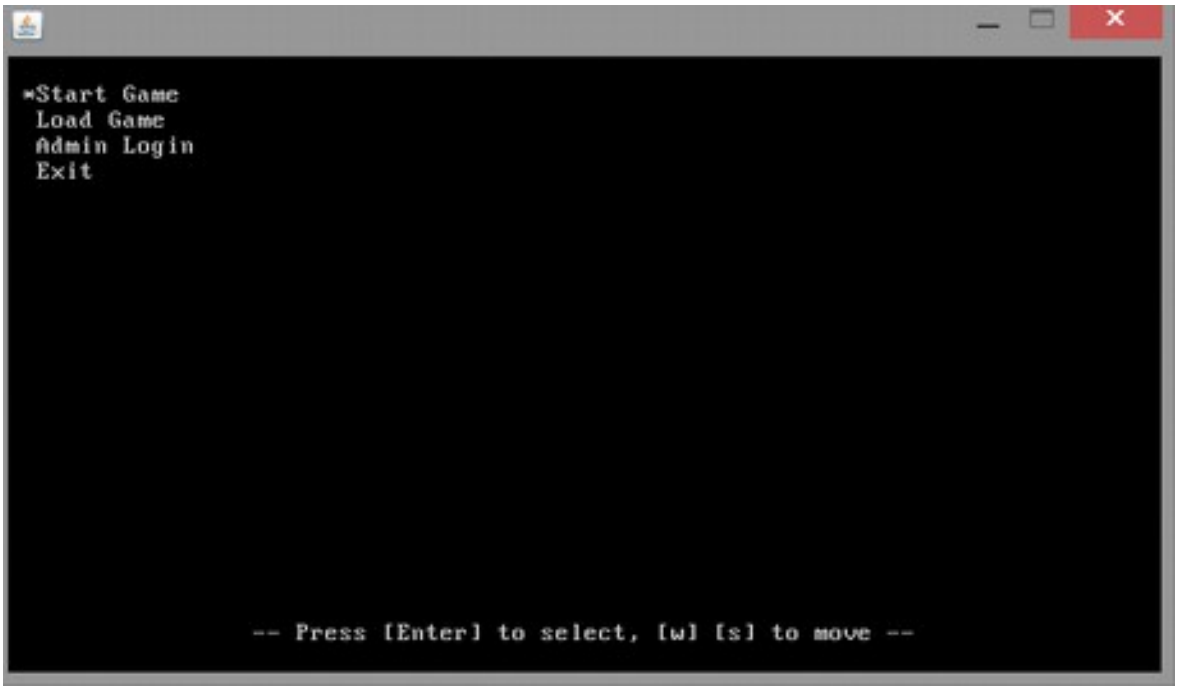
5.1 Security Requirements

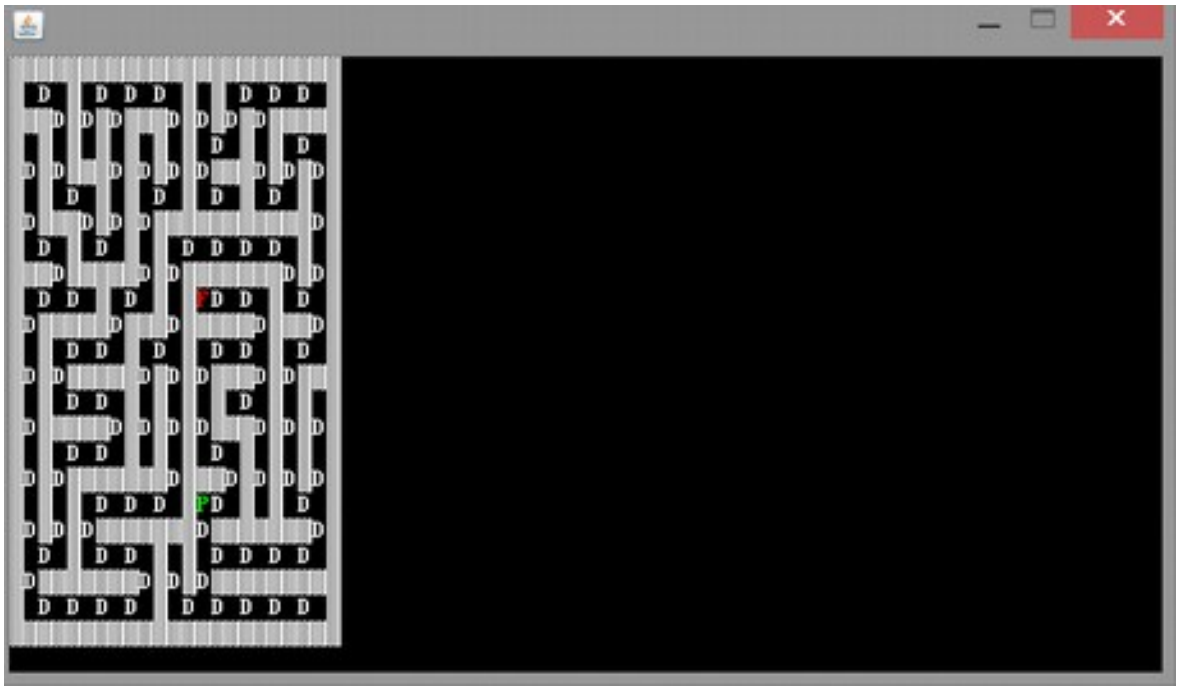
Web portal admin tools require security clearance to modify the question database.

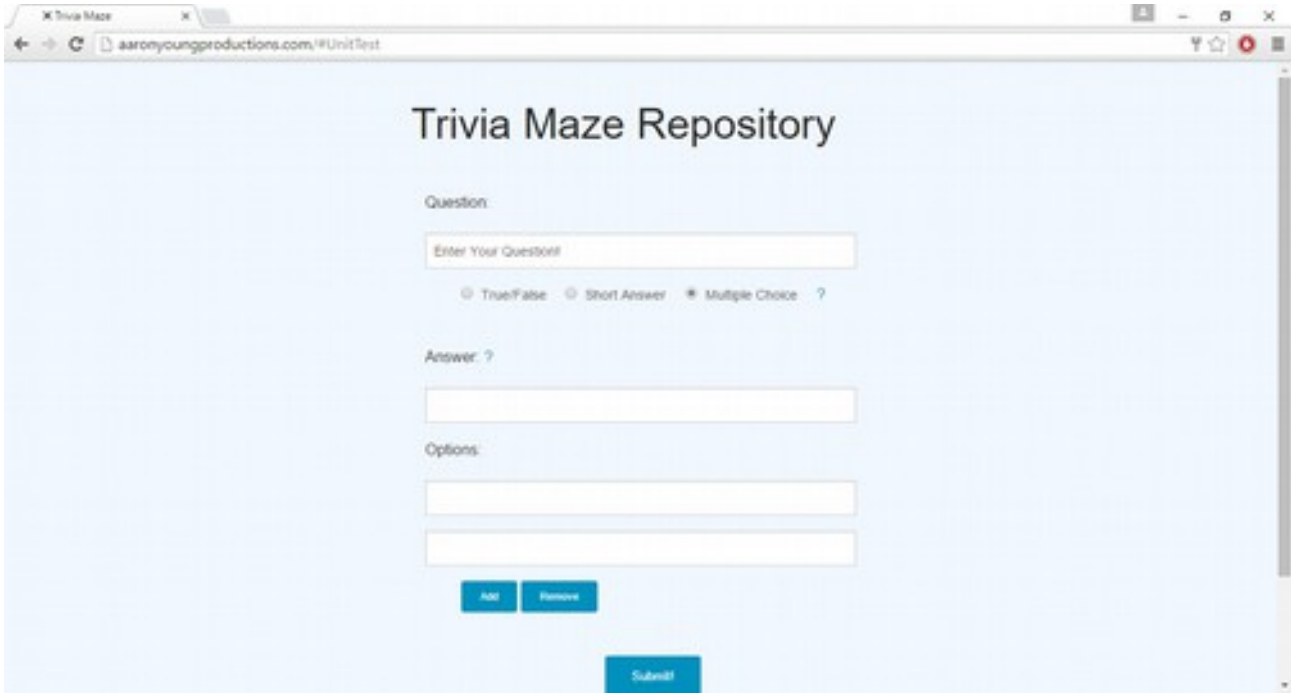
Appendix A: Glossary

To Be Implemented

Appendix B: Analysis Models


*Start Game
Load Game
Admin Login
Exit
-- Press [Enter] to select, [w] [s] to move --





Appendix C: Issues List

Occasionally the client application will hand on start up. Re running the client application will solve this most of the time.