

Software Requirements Specification

for

<Animal Trivia>

Version 3.0 approved

Prepared by Arashpreet S. Pandher, Aman Vahora, Sophia Young

Animal inc.

4/17/2022

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		
	1.5 References.....	
1		
2.	Overall Description.....	
2		
	2.1 Product Perspective.....	
2		
	2.2 Product Features.....	
2		
	2.3 User Classes and Characteristics.....	2
	2.4 Operating Environment.....	
2		
	2.5 Design and Implementation Constraints.....	
2		
	2.6 User Documentation.....	
2		
	2.7 Assumptions and Dependencies.....	
3		
3.	System Features.....	
3		
	3.1 System Feature 1.....	
3		
	3.2 System Feature 2 (and so on).....	4

4. External Interface Requirements.....	..
5. Other Nonfunctional Requirements.....	5
5.1 Performance Requirements.....	5
5.2 Safety Requirements.....	..
5.3 Security Requirements.....	5
5.4 Software Quality Attributes.....	.
6. Other Requirements.....	5

4	
4.1 User Interfaces.....	
4	
4.2 Hardware Interfaces.....	
4	
4.3 Software Interfaces.....	
4	
4.4 Communications Interfaces.....	
4	

5

5

Appendix A: Glossary.....	
5	

Appendix B: Analysis Models.....	
6	

Revision History

Name	Date	Reason For Changes	Version
Edit 2.0	5/15/2022	Update what features we want to include and exclude	2.0
Edit 3.0	6/8/2022	Make final revisions and add final features and edits.	3.0

1. Introduction

1.1 Purpose

The purpose of this document is to give a detailed description of the Animal Trivia Game. This document will explain the purpose and features, interfaces, what the game does, the constraints of the game, and requirements. Intended for team members to build the Animal Trivia Game. The game is like a maze and to move throughout the maze you will have to answer questions about animals.

1.2 Document Conventions

Font: Arial, size 11

Bold are the sub titles and titles.

1.3 Intended Audience and Reading Suggestions

The document is Intended for team members and students in the TCSS 360 section A. The rest of the document contains information on the features, interfaces, constraints, and requirements for the Animal Trivia Game. A sequence for reading this document, go from top to down.

<Describe the different types of reader that the document is intended for, such as developers, project managers, marketing staff, users, testers, and documentation writers. Describe what the rest of this SRS contains and how it is organized. Suggest a sequence for reading the

document, beginning with the overview sections and proceeding through the sections that are most pertinent to each reader type.>

1.4 Project Scope

The purpose of this software is to bring knowledge about animals to the community, with some entertainment. Our goal is to bring awareness about animals, that they are not harmful but wonderful species. This will bring insight into different species and what they are which will benefit not only the audience but the animals as well. The Trivia game will also provide animal lovers the ability to test their knowledge by being confronted with questions that are challenging by selecting the hardest difficulty.

1.5 References

<https://sqlite.org/index.html> <https://www.figma.com/file/qtt4uC3d1PF9OGIfJD8yPx/Untitled?node-id=0%3A1>

<https://www.q4quiz.com/animals-true-or-false-quiz-questions-check-the-statement-is-fact-or-fiction-about-the-animals/>

<https://parade.com/1247897/marynliles/animal-trivia/>

<https://www.q4quiz.com/animals-true-or-false-quiz-questions-check-the-statement-is-fact-or-fiction-about-the-animals/>

2. Overall Description

2.1 Product Perspective

This is a new self-contained product.

2.2 Product Features

The major feature of this product is that the user can choose which direction to move in while there will be doors throughout the maze. If the user enters the door, A question will be prompted to the user that they will have to answer in order to get through the maze. The player has five lives in which they will lose a life if the question is answered incorrectly. If they answer all the questions wrong, it will result in a game over. If they answer each question correctly, they will be able to move through the maze and reach the exit which will be located somewhere within the maze. The questions consist of multiple choice, short answer, and true or false.

2.3 User Classes and Characteristics

This product will be used by students in TCSS 360, as well as anyone who wants to learn about animals. It can be played by users of any age that have at least some knowledge of animals. If

the user has no knowledge, then it can be used as a learning tool - although it will chip away at their hp.

2.4 Operating Environment

This software will operate in multiple operating systems such as Windows, Linux, and Mac OS because it is being developed through an IDE which will allow it to be run across different platforms such as laptops and PC's.

2.5 Design and Implementation Constraints

The questions must be stored within an SQLite database. The game must include a save, load, and exit option. The game must have instructions and a description. The program must use or at least include the MVC design pattern. The minimum size of the room can be 4 by 4 rooms. The questions that can be prompted must be of three types: 1) Multiple choice. 2) Short answer. 3) True or False.

2.6 User Documentation

The system will include a user manual that will be displayed at the start of the game to inform the user of what the game is about and how the game will function. It will explain what types of questions there are and how the difficulty system works.

2.7 Assumptions and Dependencies

- The difficulty of creating a functional GUI.
- The save and load functionality using serialization within the GUI version.
- Algorithm to solve the maze in multiple ways.
- We will be using a recursive backtracking algorithm that we sort of made ourselves to create a maze.

3. System Features

3.1 System Feature 1: Choose Difficulty

3.1.1 Description and Priority

This feature will allow the user to choose a level of difficulty. There will be three difficulties the user can choose from. It will consist of easy, medium, and hard. The difficulty system will increase the chances of question popping up and select questions that will be more difficult. It is a low priority feature because this is an add on, so we are not too worried about it.

3.1.2 Stimulus/Response Sequences

Stimulus: User clicks option for choosing difficulty

Response: The maze will be set to the difficulty level the user picked

3.1.3 Functional Requirements

- Maze.Select: The system will allow the user to choose from the difficulty that is selected in the main menu screen.
- Maze.difficulty: The selected difficulty will instruct the program to choose questions from the database related to that difficulty.

3.3 System Feature 3: Save, Load, and Exit game:

3.3.1 Description and Priority

The user will be able to save the game which will store their current location within the maze and the questions they answered and have not answered. The user will also be able to load the game to restore the current location and questions, then continue from where they left off. The load button loads the most recent saved file. Lastly, an exit function will be added to make sure the user can exit from the game at any time which will prompt the user to save the game once again. The priority is high on this because it is one of the requirements.

3.3.2 Stimulus/Response Sequences

Stimulus: User clicks the save option.

Response: The program will store the location and health of the user. It will display a prompt stating the game has been saved.

Stimulus: User clicks the load option.

Response: The program will restore the location and health of the user. This allows user to carry on from where they left off. The program will execute from the saved state.

Stimulus: User clicks the exit option.

Response: The exit feature asks the user to save the game once again. It will stop the program and close the game.

- GUI.Save: The system will use serialization to store the information of the maze into a text file.
- GUI.Load: The system will access the save file and take that information from the file to deserialize and restore the information from where the user saved.
- GUI.Exit: The Program will terminate and the GUI will close.

3.4 System Feature 4: Sound option

3.4.1 Description and Priority

The game will include a sound feature that will be playing a song throughout the game and the user will have an option to mute or unmute it. This is low priority since this is just a small feature to make the game more interesting and interesting.

3.4.2 Stimulus/Response Sequences

Stimulus: User clicks the mute option.

Response: The audio is either turned on or off.

Stimulus: User click one of the arrows to move.

Response: An audio is played for the user.

- GUI.Sound: The system will play audio while the game is run

3.5 System Feature 5: Movement

3.5.1 Description and Priority

The character will be able to move left, right, up, or down. The user can control the movements of the character to navigate through the maze. This is a high priority because without this feature the user cannot traverse through the maze.

3.5.2 Stimulus/Response Sequences

Stimulus: A movement button is pressed.

Response: If able, the character will move towards the direction inputted by the user.

- GUI.Movement: The program will display movement keys that the user can click to move in the direction they choose.

3.7 System Feature 7: Victory

3.7.1 Description and priority

This feature is only available when the user gets to the exit with health remaining. Otherwise the game over feature will be displayed.

3.7.2 Stimulus/Response sequences

Stimulus: The user reaches the exit.

Response: The program shows a victory screen and then closes.

- GUI.Victory: The system will display a Victory prompt

3.8 System Feature 8: Help

3.8.1: Description and Priority:

This feature will include an about page for what the game is about and a game instruction manual to tell the user how to play the game and what to expect within the game.

3.8.2 Stimulus/Response sequences

Stimulus: The user clicks help menu:

Response: An about and game instructions is displayed.

Stimulus: The user clicks about:

Response: The about will display what the game is about

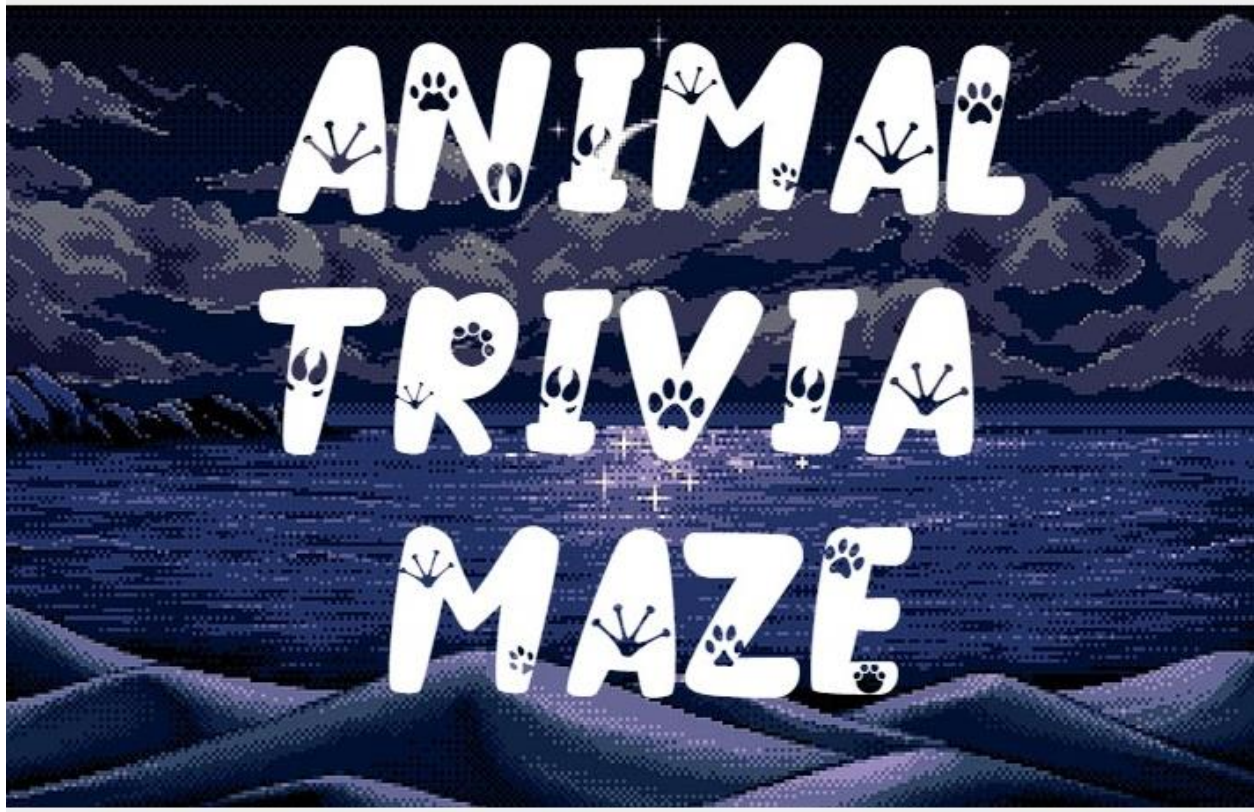
Stimulus: The user clicks game instructions:

Response: The game instructions for the Trivia game are displayed.

4. External Interface Requirements

4.1 User Interfaces

UI-1: The Main Menu: The user will be able to select the difficulty in this menu.



New Game

Load Game

Exit



UI-2: Gameplay: This will be the visual components with which the user can interact with. It will display the maze and movement.



4.2 Hardware Interfaces

The program should be supported on any device that can compile java.

4.3 Software Interfaces

The game uses an SQLite database to store the questions, answers, difficulty, and the type of question. The database will be accessed and the data will be stored into a door object which is part of the maze.

This Animal trivia Maze game version 1.5. Version 1 was console version and version 1.5 is the GUI version.

Java Libraries:

- Java standard Libraries
- Junit
- Apache Commons

6. Other Requirements

Appendix A: Issues List

Error showed up once indicating SQL questions arraylist was out of bounds.

Pending decisions:

- Add feature to choose character
- Pause menu
- Make the user interface better
- Add installer for game