Software Requirements Specification

for

Shell Game

Version 3.0 approved

Prepared by Paul Kelting, Akshobh Mirapurkar, Cody Ridge, and Daniel Schell

Client: Dr Julie Suhr

Russ College of Engineering at Ohio University

April 19, 2021

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 Product Scope 1

2. Overall Description 1

2.1 Product Perspective 1

2.2 Product Functions 1

2.3 User Classes and Characteristics 1

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

2.6 Assumptions and Dependencies 2

3. External Interface Requirements 2

3.1 User Interfaces 2

3.2 Hardware Interfaces 3

3.3 Software Interfaces 3

3.4 Communications Interfaces 3

4. System Features 3

4.1 Main Menu 3

4.2 Shell Game 3

5. Other Nonfunctional Requirements 4

5.1 Performance Requirements 4

5.2 Security Requirements 4

5.3 Business Rules 4

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Paul Kelting | 12/2 | SRS Update Assignment | 2 |
| Paul Kelting | 4/19 | Final Sprint | 3 |

# Introduction

## Purpose

Create a modifiable shell game to test participants. The design will be clean and be able to keep a constant memory of patient results and trial settings.

## Document Conventions

The content of the document is in Times New Roman at font size 12. The main titles are written in

bold Times New Roman at font size 18, and the subtitle follow the same format but are at font size

14. The left/right margins are .9” with top/bottom at 1”. 1.0 spacing is used between lines.

## Intended Audience and Reading Suggestions

Psychologists who are looking to test the cognitive abilities of an individual.

## Product Scope

At this point it is being used to test cognitive abilities of someone.

# Overall Description

## Product Perspective

This will be used in a professional environment by psychologists.

## Product Functions

The test proctor will be allowed to modify the number of trials, number of cups per trial, number of swaps per trial, and speed of the swaps per trial. These test settings can be saved for later use. The person taking the test will receive a positive notification when the ball was correct. After the trials have been completed the test proctor will be given an excel spreadsheet displaying the results for each specific trial, along with each trials settings. A patient number will be associated with each chunk of data.

## User Classes and Characteristics

The program is designed to be proctored by a trained psychologist. The test itself is designed to be accomplished by any sighted person.

## Operating Environment

The program can be operated via any desktop computer via an executable. This executable will be written in C# and exported from Unity. This program will be used in a professional workspace.

## Design and Implementation Constraints

This program isn’t graphically intense and requires only a small portion of memory to save trial settings and trial results.

## Assumptions and Dependencies

The device must have some graphical capabilities to run the executable.

# External Interface Requirements

## User Interfaces

The launch menu will contain a password to enter the admin settings. For the admin menu the proctor can enter in the details for the test. A patient ID number will be prompted in order to update the correct spreadsheet. The proctor can enter X number of trials, which will open up interfaces to modify the specifics for X number of trials. Each trial has different speeds, swaps, and cups. The stack of trials will allow saving presets and applying previous settings. The interface for the test taker will display the appropriate settings in the shell game environment. A non admin can only select an output file and a trial preset to use.

## Hardware Interfaces

The hardware required to interact with this program is a mouse, to click buttons, and a keyboard to enter in specific settings.

## Software Interfaces

This program is written in C# and has be exported via unity to an executable file.

## Communications Interfaces

This program will not communicate with other devices while running.

# System Features

This program will be an executable created by Unity and can be run on desktop devices. The user will be greeted with the first of the two environments, the main menu. The second environment, the shell game, will be the 3D environment that is transitioned to after the user has given trial stack settings to the main menu.

## Main Menu

4.1.1 Description and Priority

This will be a medium priority. It would not have a cluttered design and will serve to enter in settings that will be used in the test.

4.1.2 Stimulus/Response Sequences

The user will type in the settings that they wish to use.

4.1.3 Functional Requirements

REQ-1: Clean design

REQ-2: Receives user input

REQ-3: Give the test the correct settings

REQ-4: Allow settings to be saved

REQ-5: Admin lock for modifying settings

REQ-6: Detailed guide

## Shell Game

4.2.1 Description and Priority

This will be high priority since it is the focus of the project. It should be correctly customized with the settings provided by the proctor. The data will then be exported to excel when trials are finished.

4.2.2 Stimulus/Response Sequences

The player will be allowed to select a cup that they think correctly hides the ball. They will receive a notification if that choice was correct.

4.2.3 Function Requirements

REQ-1: Correctly uses applied settings

REQ-2: Clean animations

REQ-3: Notification for user on when they answer

REQ-4: Data exported to excel after trials

# Other Nonfunctional Requirements

## Performance Requirements

This isn’t graphically intensive so all desktop computers should be able to run it

## Security Requirements

Patient confidentiality with results.

## Business Rules

The person proctoring the test should not show the test settings to the person taking the test.