Sliding Puzzle Game Application

Test Plan *Template* & Testcases)

Version 1.0

07/15/2017

VERSION HISTORY

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Version #** | **Implemented**  **By** | **Revision**  **Date** | **Approved**  **By** | **Approval**  **Date** | **Reason** |
| 1.0 | *Round Rock team* | *07/25/17* |  |  | Test Plan Template draft |
| 2.0 | *Round Rock team* | *08/2/17* |  |  | Updated test cases |
|  |  |  |  |  |  |

TABLE OF CONTENTS

[1 Introduction 4](#_Toc489364525)

[1.1 Purpose of The Test Plan Document 4](#_Toc489364526)

[2 COMPATIBILITY Testing 4](#_Toc489364527)

[2.1 Items to be Tested / Not Tested 4](#_Toc489364528)

[2.2 Test Approach(s) 4](#_Toc489364529)

[2.3 Test Pass / Fail Criteria 4](#_Toc489364530)

[2.4 Test Entry / Exit Criteria 4](#_Toc489364531)

[3 Functional Testing 4](#_Toc489364532)

[3.1 Test Risks / Issues 4](#_Toc489364533)

[3.2 Items to be Tested / Not Tested 4](#_Toc489364534)

[3.3 Test Approach 5](#_Toc489364535)

[3.4 Test cases 5](#_Toc489364536)

[3.4.1 Account Registration 5](#_Toc489364537)

[3.4.2 Login 5](#_Toc489364538)

[3.4.3 Logout 6](#_Toc489364539)

[3.4.4 New Game 6](#_Toc489364540)

[3.4.5 Solve 6](#_Toc489364541)

[3.4.6 Save 7](#_Toc489364542)

[3.4.7 Check Valid/invalid steps; 7](#_Toc489364543)

[4 Unit Testing 8](#_Toc489364544)

[4.1 Test Risks / Issues 8](#_Toc489364545)

[4.2 Items to be Tested / Not Tested 8](#_Toc489364546)

[4.3 Test Approach(s) 8](#_Toc489364547)

[4.4 Test Regulatory / Mandate Criteria 8](#_Toc489364548)

[4.5 Test Pass / Fail Criteria 9](#_Toc489364549)

[4.6 Test Entry / Exit Criteria 9](#_Toc489364550)

[4.7 Test Deliverables 9](#_Toc489364551)

[4.8 Test Suspension / Resumption Criteria 9](#_Toc489364552)

[4.9 Test Environmental / Staffing / Training Needs 9](#_Toc489364553)

[5 User Acceptance Testing 9](#_Toc489364554)

[5.1 Items to be Tested 9](#_Toc489364555)

[6 Conclusion: 9](#_Toc489364556)

# Introduction

## Purpose of The Test Plan Template Document

The Test Plan Template document documents and tracks the necessary information required to effectively define the approach to be used in the testing of end product. This document is created during the Planning Phase of the project and is leveraged when building test cases.

# COMPATIBILITY Testing

## Items to be Tested / Not Tested

|  |  |  |  |
| --- | --- | --- | --- |
| **Item to Test** | **Test Description** | **Test Date** | **Responsibility** |
| execute game in laptop | execute game in laptop with windows 10 | 8/2/2017 | IT Team |
| Execute game in mobile | Execute game in mobile with windows operating system 10 | 8/2/2017 | IT Team |
|  |  |  |  |

## Test Approach(s)

By executing the compatibility test cases we validate that application can be deployed and played in both mobile and laptops.

## Test Pass / Fail Criteria

Success criteria: User should successfully play game in both mobile and desktop editions of windows operating systems.

Fail criteria: If user is unable to play any of the functionality described in SRS, the test deemed as failed.

## Test Entry / Exit Criteria

Entry criteria: Deploy the application, complete the registration and play the game.

Exit criteria: User plays the game successfully.

# Functional Testing

## Test Risks / Issues

If the system is not connected to internet, the test cases related to login and registration would fail.

## Items to be Tested / Not Tested

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Item to Test** | | **Test Description** | **Test Date** | **Responsibility** | |
| 1 | Account Registration | Validate if the user is able to complete Account registration process | | 8/2/2017 | | IT Team |
| 2 | Login | Validate if the user is able to complete Account registration process | | 8/2/2017 | | IT Team |
| 3 | Logout | Validate if the user is able to complete logout process | | 8/2/2017 | | IT Team |
| 4 | Solve | Validate Solve functionality | | 8/9/2017 | | IT Team |
| 5 | Save | Validate Save functionality | | 8/9/2017 | | IT Team |
| 6 | New Game | Validate New Game functionality | | 8/9/2017 | | IT Team |
| 7 | Allowed Steps | Check Valid/Invalid steps | | 8/9/2017 | | IT Team |

## Test Approach

Deploy the application on the emulators before testing the application on actual device. After completion of emulator testing, game is deployed on a device and validate all scenarios again.

## Test cases

### Account Registration

|  |  |  |  |
| --- | --- | --- | --- |
| **Execution step** | **Expected output** | **Actual output** | **PASS/ FAIL** |
| START |  |  |  |
| Execute application to start the game | Game login screen is displayed |  |  |
| Click ‘New user’ | Account registration page is displayed |  |  |
| Fill in required details and hit “Register” | User is created in database and game home page is displayed |  |  |
| END |  |  |  |

### Login

|  |  |  |  |
| --- | --- | --- | --- |
| **Execution step** | **Expected output** | **Actual output** | **PASS/ FAIL** |
| START |  |  |  |
| Execute application to start the game | Game login screen is displayed |  |  |
| Click login | Text box to enter username is displayed |  |  |
| Fill in required details and hit “OK” | If existing user, previous session of the game is displayed |  |  |
|  | If new user, user can select a new game |  |  |
| END |  |  |  |

### Logout

|  |  |  |  |
| --- | --- | --- | --- |
| **Execution step** | **Expected output** | **Actual output** | **PASS/FAIL** |
| START |  |  |  |
| Execute application to start the game | Game login screen is displayed |  |  |
| complete login functionality | User is successfully logged in |  |  |
| Click logout button | User is logged out and a login button is displayed. |  |  |
| END |  |  |  |

### New Game

|  |  |  |  |
| --- | --- | --- | --- |
| Execution step | Expected output | Actual output | PASS/FAIL |
| START |  |  |  |
| Execute application to start the game | Game login screen is displayed |  |  |
| Click New game button to start a new game | New game shall be started |  |  |
| END |  |  |  |

### Solve

|  |  |  |  |
| --- | --- | --- | --- |
| Execution step | Expected output | Actual output | PASS/FAIL |
| START |  |  |  |
| Execute application to start the game | Game login screen is displayed |  |  |
| Click New game button to start a new game | New game shall be started |  |  |
| Play the game for few steps | User shall be able to slide the numbers based on the expected moves |  |  |
| Click solve | Game would revert to initial state and numbers shall start sliding to re organize in ascending order. |  |  |
| END |  |  |  |

### Save

|  |  |  |  |
| --- | --- | --- | --- |
| Execution step | Expected output | Actual output | PASS/FAIL |
| START |  |  |  |
| Execute application to start the game | Game login screen is displayed |  |  |
| Click New game button to start a new game | New game shall be started |  |  |
| Play the game for few steps | User shall be able to slide the numbers based on the expected moves |  |  |
| Click Save button | User game data is saved |  |  |
| Close the application | Application is closed |  |  |
| Restart the application and login as the same user | Game shall be resumed from where left off. |  |  |
| END |  |  |  |

### Check Valid/invalid steps;

|  |  |  |  |
| --- | --- | --- | --- |
| Execution step | Expected output | Actual output | PASS/FAIL |
| START |  |  |  |
| Execute application to start the game | Game login screen is displayed |  |  |
| Click New game button to start a new game | New game shall be started |  |  |
| Play the game for few steps | User shall be able to slide the numbers based on the expected moves |  |  |
| Check Valid steps:  Move slide by 1 slot | The tile can slide only into adjacent empty slot |  |  |
| Check invalid steps  Move tile into a non empty slot | The tile cannot slide into non empty slot |  |  |
| Check invalid steps  Move tile into an empty slot whose distance is more than one slot. | The tile cannot slide into a slot which is not adjascent. |  |  |
| END |  |  |  |

# Unit Testing

## Test Risks / Issues

View cannot be tested. Only model and view model can be tested.

## Items to be Tested / Not Tested

All model classes must be tested. Unit test goal is 75 % coverage using Resharper testing tool.

## Test Regulatory / Mandate Criteria

Unit test goal is 75 % coverage.

## Test Pass / Fail Criteria

All assert statement must pass or return true for a test to pass.

## Test Deliverables

Test project with unit test case code, screenshot of tests passing and coverage results will be the deliverables.

# User Acceptance Testing

## Items to be Tested

|  |  |  |  |
| --- | --- | --- | --- |
| **Item to Test** | **Test Description** | **Test Date** | **Responsibility** |
| New User is able to register | Validate New user registration process |  |  |
| User can resume from a saved state | Validate if user can resume a previous game |  |  |
| User interface has the expected functionality as displayed in SRS. | UI is as shown in the diagram in SRS document |  |  |
| User can solve a puzzle | Validate if user can solve the puzzle |  |  |
| Tiles movement | Tiles are moved only to adjacent empty slots |  |  |

# Conclusion:

The test cases mentioned in this document ensures the application developed is thoroughly tested and validated against expected outputs.