SiQuoia

(Simple Intelligence Quotient Increasing Application)

Test Plan

Version 2

7 December 2013

Team SQ03

CMPE 131-03/CS 160-02
Computer Engineering/Computer Science
Software Engineering
San Jose State University

Revision History

Date	Description	Author	Comment
11.24.13	Version 1	SQ03	
12.7.13	Version 2	SQ03	

Document Approval

Signature	Printed Name	Title	Date

TABLE OF CONTENTS

1. Introduction

1.1 Purpose

This document presents the Test plan for SiQuoia's functional and nonfunctional requirements.

1.2 Scope

The scope of this document extends to testing the application in eight perspectives: Integration, System, Usability, Regression, Performance, Load and Stress Installation and Acceptance testing. Each of the perspectives will include tests that will verify connectivity of the device to the server, test the use and functionality of the user interface of the application on multiple android devices and installation of the application on said devices.

1.3 Testing Strategy

Testing on different devices
SiQuoia, an application developed for the Android operating system for running and managing quizzes will be tested on multiple android devices. Each new version of SiQuoia will be tested with new .APK files.

Functional Testing
SiQuoia will provide end-users an interface to create their own personal user account, participate in quizzes, submit new questions for quizzes, and exchange a form of currency for perks in-game. Functional requirements are organized into task for the implementation team. Once each functional requirements are completed, the testing team will test and report the function.

Nonfunctional Testing
Developing SiQuoia on the Android platform will provide the benefits of the fast-growing market share of Android mobile devices. In order to keep up with Android's specification: performance, load/stress, reliability, availability, recoverability, liability, security, compatibility, installation, and serviceability testing will be performed.

1.4 References

IEEE Recommended Practice for Software Requirements Specifications
(IEEE Std 830-1998) 25 June, 1998

SiQuoia Inc. Request For Proposal, RFP No. 20130822

SiQuoia Inc. Traceability Matrix

SiQuoia Inc. SRS final version

Android 4.0 Compatibility Definition, Revision 4
Last updated: April 21, 2013

1.5 Overview

The remainder of this document will provide an overall vision of SiQuoia's testing plan by describing the functional and nonfunctional test cases.

2 . Test Plan

2.1 Interface Testing

Test Definitions

Interface testing include many unit tests that verify and validate each function and module in our implementation of the SiQuioa application.

Test Procedures

All tests will be done manually, on all hardware devices specified under Installation testing section or the Android Emulator provided with the Android SDK.
We will be using UI in different functional interfaces to test each functional requirements.
Here's a short overview of the steps required to automate UI testing:

1. Prepare to test by installing the app on a test device, analyzing the app's UI components, and ensuring that our application is accessible by the test automation framework.
2. Compile our test cases into a JAR file and install it on our test device along with our app.
3. Run the tests and view the test results.
4. Correct any bugs or defects discovered in testing.

Testing Details

Note: All tests are conducted on either a android device or emulator. Testing will require access to the database.

2.1.1 User Login:

Case 1- Correct username and password :

1. Input correct username and password on login screen.
2. Confirm that application login successfully and switches to main menu screen.
3. Check on database that the number of points displayed on home is current for user account.

Case 2- Correct username and incorrect password :

1. Input correct username and incorrect password on login screen.
2. Confirm that application does not login and “incorrect credential” alert pops up.

Case 3- Incorrect username and correct password :

1. Input incorrect username and correct password on login screen.
2. Confirm that application does not login and “incorrect credential” alert pops up.

Case 4- Empty username and Empty password :

1. Click login without inputting username or password on login screen.
2. Confirm “Please enter both fields” alert pops up.

Case 5- Empty username and non-empty password :

1. Input correct username and incorrect password on login screen.
2. Confirm that application does not login and “Please enter both fields” alert pops up.

Case 6- Non-empty username and empty password :

1. Input a username and no password on login screen.
2. Confirm that application does not login and “Please enter both fields” alert pops up.

2.1.2 Create User :

Case 1- Create new user :

1. Click “Create New Account” in login screen”.
2. Enter email, password and re-enter password for each field.
3. Click “Create” button.
4. Confirm that the new user has been added to the database.

Case 2- Creating duplicate user :

1. Click “Create New Account” in login screen”
2. Enter same email, same password and re-enter same password for each field.
3. Click “Create” button.
4. Confirm that “email is already” alert pops up.
5. Confirm that new entry in database has not been created.

Case 3- Empty email :

1. Click “Create New Account” in login screen”.
2. Leave email field empty, enter password and re-enter password for each field.
3. Click “Create” button.
4. Confirm that “email is already” alert pops up.
5. Confirm that new entry in db has not been created.

Case 4- Empty email empty password :

1. Click “Create New Account” in login screen”.
2. Leave email field empty, enter password and re-enter password for each field.
3. Click “Create” button.
4. Confirm that “email is already” alert pops up.
5. Confirm that new entry in db has not been created.

Case 5- Non-formated email :

1. Click “Create New Account” in login screen”.
2. Leave email field empty, enter password and re-enter password for each field.
3. Click “Create” button.
4. “Please enter valid email” alert pops up.

Case 6- Empty email empty password :

1. Click “Create New Account” in login screen”.
2. Leave email field empty, enter password and re-enter password for each field.
3. Click “Create” button.
4. Confirm that the “please enter valid email” alert pops up.

Case 7- Non-empty email and empty password :

1. Click “Create New Account” in login screen”.
2. Leave email field empty, enter password and re-enter password for each field.
3. Click “Create” button.
4. Confirm that “password does not match” alert pops up.

Case 8- Non-empty email, but different password input :

1. Click “Create New Account” in login screen”.
2. Enter email and password, but enter different password.
3. Click “Create” button.
4. Confirm that “password does not match” alert pops up.

2.1.3 New Quiz

Case 1- Obtain a new quiz without specifying any subject, topic, subtopic :

1. Click “New Game” in the main menu screen.
2. Click “yes” when asked to spend SiQuoia points.
3. Click on “Create Quiz”.
4. Log onto the database and confirm:
    a. the quiz has been stored for user.
    b. the user current answers are now equal to empty string.
    c. that the user has 5 point taken off from user’s account.
5. Go through quiz and confirm quiz is random.

Case 2- New quiz by specifying subject, topic and subtopic :

1. Click “New Game” in the main menu screen.
2. Click “yes” when asked to spend SiQuoia points.
3. Select subject “English”.
4. Select topic “Vocabulary”.
5. Select Sub-Topic “Definition”.
6. Click on “Create Quiz” .
7. Log onto the database and confirm:
    a. the quiz has been stored for user.
    b. the user current answers are now equal to empty string.
    c. that the user has 5 point taken off from user’s account.
8. Go through quiz and confirm that the question are from definition packet on the database.

Case 3- New quiz by specifying only subject and topic :

1. Click “New Game” in the main menu screen.
2. Click “yes” when asked to spend SiQuoia points.
3. Select subject “English”.
4. Select topic “Vocabulary”.
5. Select Sub-Topic “Any”.
6. Click on “Create Quiz”.
7. Confirm confirm that the questions are from “Definition” and “Synonym” quiz packet on the database.
8. Log onto the database and confirm:
    a. The quiz has been stored for user.
    b. The user current answers are now equal to empty string.
    c. That the user has 5 point taken off from user’s account.

Case 4- New quiz by specifying only subject :

1. Click “New Game” in the main menu screen.
2. Click “yes” when asked to spend SiQuoia points.
3. Select subject “English”.
4. Select topic “Any”.
5. Select Sub-Topic “Any”.
6. Click on “Create Quiz”.
7. Confirm “Definition”, “Synonym”, “Vocabulary” and “Writing” quiz packet.
8. Confirm that SiQuoia points have been spent by returning to main menu.

Case 5- New quiz before completing a started quiz :

1. Click “New Game” in the main menu screen.
2. Click “yes” when asked to spend SiQuoia points.
3. Click on “Create Quiz”
4. Go back to main menu and click “New Game”
5. Click “yes” when asked to spend SiQuoia points and overwrite current quiz.
6. Click on “Create Quiz”.
7. Confirm that SiQuoia points have been spent by returning to main menu.
8. Confirm all previous quiz data is gone and replaced by new quiz data on database.

2.1.4 Branded Quizzes

Case 1 - Inputting a valid code:

1. Select “Redeem Code” from menu.
2. Enter valid code (CS160 for example) and click “enter code”.
3. Confirm that a new quiz opens up.
4. Log into database and confirm that quiz is sames to user’s current Quiz field.

Case 2 - Inputting an invalid code:

1. Select “Redeem Code” from menu.
2. Enter a non-valid code and click “enter code”.
3. Confirm that “Incorrect Code” message pops up.

2.1.5 Submit a Question

Case 1 - Submit a question form is completely filled out :

1. Select ‘Submit Question Option’.
2. Fill out all the question fields.
3. Click Submit button.
4. Make sure email sender opens up.
5. Send email.
6. Log on to siquoiaquiz@gmail.com and confirm email has arrived.

Case 2 - Submit a question form is incompletely filled out :

1. Select the 'Submit Question Option'.
2. Fill out some fields and leave some other fields to be empty.
3. Click submit button.
4. Confirm that "Please fill all fields" alert pops up.

2.1.6 User Profile

1. Click "User Profile" from menu.
2. Confirm that information shown in the "User Profile" screen matches information in the database. Information that needs to be confirmed is:
    a. user email.
    b. current number of SiQuoia Points.
    c. number of packet bought.
    d. number of memorabilia bought.
    e. total points spent.
    f. and total points earned.

2.1.7 Continuing a quiz:

Case 1 - No saved Quiz :

1. On Home screen, press "Continue Button".
2. Confirm that the "No Saved Quiz" pops up.

Case 2 - Saved, but not completed quiz Quiz:

1. On Home screen, press "Continue Button"
2. Based on saved info in database:
    a. Confirm that the right number of questions have been answers.
    b. Confirm that the correct question are displayed in green and incorrect ones are displayed in red.
    c. Confirm number of correct questions is correctly displayed.

Case 3 - Saved, completed Quiz

1. On Home screen, press "Continue Button".
2. Confirm that app goes directly into quiz results screen.
3. Based on saved info in database:
    a. Confirm that the number of correct answers is displayed properly.

2.1.8 Answering Question:

Case 1 - Attempting to answer question out of order :

1. Open up a quiz (new or save).
2. Click on the any question except the "next question" (next question being the first question after the last question answered).
3. Confirm an alert pops up prompting users to select the "next question".

Case 2 - Answering Questions :

1. Open up a quiz (new or save).
2. Click on the first question after the lasted answered question.
3. Answer the questions.
4. Confirm that app goes directly to next question.
5. Confirm that if answer was correct, that the total points on upper right side has been increased.
6. Confirm in the database
    a. That the user's answers has been appended to their correct answers.
    b. If question is answered correctly, make sure its rank has been increased by one.

2.1.9 Completing Quiz:

Case 1 - Completing normal quiz :

1. Open up a quiz (new or save).
2. Answer all the questions.
3. After answering last question, make sure quiz goes to "quiz completion" page.
4. Confirm that number of correct answers gets added to user total points in database.

Case 2 - Completing "Branded" quiz :

1. Answer all questions.
2. After Last question question, make sure quiz goes to "Quiz completion" page.
3. In database, confirm that the user's siquioa points value has not change.

2.1.10 Leaderboard

1. On Home screen, click on Leaderboard.
2. Log into database and go to questions table and arrange table by descending order based on rank.
3. Confirm that the top 20 questions in the database match the questions show in the applications leaderboard.

2.1.11 Logout:

1. On Home screen, click on the menu and select logout.
2. Make sure app takes users to the Login Screen.
3. Quit the application completely.
4. Open the application and confirm it opens to Login Screen.

2.2 Integration Testing

Test Definitions

During development, we will be combining different features (as show the traceability matrix and unit tests) to work together. We will combine all features that relate to each functional requirement, both android code and php code and make sure that integrated module works as desired. For instance, we will link the Android login code with the PHP login code and confirm that they work together.

Test Procedures

The following list of modules and functional will all be tested together with continuous integration throughout the development:

- New Quiz
    - Selection of categories: Subject, Topic, Subtopic
    - Fetch a new quiz from the database
- Continue Quiz
    - Continues a quiz the user was previously playing
    - Fetch the quiz from the database
- Leaderboard
    - Fetch information about question ranking in our database
- Submit a Question
    - Submit a question interface on the application
    - The server sends the prospective question to the administrators.

The combining module will be deploy as Android .APK file for testing different functionality.

Test Expectations

Upon integrating all modules and functions into a single application as a Android .APK file, the application will have all necessary functionality and will be tested to function as set-out by the System Requirement Specification.

2.3 System Testing

Test Definitions

The methodology for performing System Testing on SiQuoia consists of black-box testing that will be administered by everyone on the team outside of the Development team.

Test Procedures

System test will be performed on the application running on a supported Android device. Once all modules have been integrated and the application is complete, we will run through all functionality on the client application and ensure that the server is correctly tracking and updating the database. Specific functionality includes: registering as a new user, obtaining a quiz packet, continuing a quiz and all other unit tests that were used in Interface Testing (2.1).

Test Expectations

The client application and server will be communicating with each other so that all quiz packets, user information and leaderboard status will be updated on the user's device and on the database in real time and accurately.

2.4 Usability Testing

Test Definitions

We will test the application from the users' perspective and see how easy it is to use the app and how responsive the app is.

Test Procedures

Our development team will evaluate and ease of use and responsivity of the application. If time is permitted, we will have additional users test the application and obtain their feedback. We will take feedback and take the appropriate measure to ensure that the application is usable.

Test Expectations

The expectation of the application is to provide an easy to use and intuitive touch design that will not slow down the user when taking quizzes.

2.5 Regression Testing

Test Definitions

Regression Testing will be performed via black-box testing new software updates to verify and validate that the update has not introduced new defects into SiQuoia. In addition, we will ensure that both functional and nonfunctional requirements have been met.

White-box testing will done by the developer team to check that the internal functionality of SiQuoia performs as expected.

Test Procedures

Each time a new commit has be pushed to the repository each memeber will test the new function on their given devices or emulators. Should a new feature or function change another feature's behavior the development manager will be notified to make a correction.

Test Expectations

This set of testing with ensure that each module or function in our application will be safely implemented without the risk of having an nonfunctioning application.

2.6 Performance, Load and Stress Testing

Test Definitions

Due to lack of resources we are unable to fully stress or load test our server. However, we will be do performance tests on said Android devices for speed and fidelity.

Test Procedures

By simultaneously using hardware specified in installation testing, we record the amount of time required for loading the application as well as doing actions such as create new user and view leaderboards.

Test Expectations

The recorded amount of time for loading the application should all be under 3 seconds, loading a quiz shall be under 4 seconds, and all other other operations will be under 3 seconds.

2.7 Installation Testing

Test Definitions

Installation Testing will include installing the application on multiple Android platforms on multiple devices.

Test Procedures

These will include the Android Emulator and mobile devices running Android 4.0.0 above on the following hardware:

- LG Nexus 4
- Asus Nexus 4
- Motorola Droid Razr
- Samsung Galaxy Tab 7.7

After installing the SiQuoia app on these devices, our client and team will conduct Acceptance tests to ensure that the application is working properly on each device and version of Android.

Test Expectations

The expected result after installation testing is that the application was able to be install and started without compatibility conflicts and the application is able to be ran.

2.8 Acceptance Testing

Test Definitions

Acceptance testing will consist of a subset of the System tests in order to validate the major functional and nonfunctional requirements. The subset of System tests that will be used for Acceptance testing have been mutually agreed upon between the customer and the SQ03 Development team and client will sign off on agreed System tests.

Test Procedures

Acceptance Tests:

- User Account creation
- Login and logout of SiQuoia
- Buy and download a quiz
- Start and finish a quiz (Answer all 20 questions)
- Leaderboard is accessible
- Multimedia in questions is accessible
- Ability to submit questions for review

Test Expectations

After presenting these features and functions to the client, we would expect them to accept the application and its implementation. If the client is not acceptive of the implementation then the development team will have change the implementation as necessary.