SiQuoia

(Simple Intelligence Quotient Increasing Application)

Test Plan

Version 1

24 November 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	

Document Approval

Signature	Printed Name	Title	Date

TABLE OF CONTENTS

Revision History..... 2

Document Approval..... 2

1. Introduction..... 3

1.1 Purpose..... 4

1.2 Scope..... 4

1.3 Testing Strategy..... 4

1.4 References..... 4

1.5 Overview..... 4

2. Testing Plan……..... 5

2.1 Interface Testing……..... 5

2.1.1 Functional Testing/Unit Testing …….... 6

2.1.2 Non-Functional Testing……..... 7

2.2 Integration Testing……..... 7

2.3 System Testing …….... 8

2.4 Usability Testing……..……..……..……..…….. 9

2.5 Regression Testing……..……..……..…….. 9

2.6 Performance, Load and Stress Testing……..……..…….. 9

2.7 Installation Testing……..……..……..…… 10

2.8 Acceptance Testing……..……..……..…….. 10

1. Introduction

1.1 Purpose

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

1.2 Scope

Testing will be performed at several points in the life cycle as the product is constructed. Testing is a very 'dependent' activity. As a result, test planning is a continuing activity performed throughout the system development life cycle. SiQuoia's Test plans will be developed for each level of product testing. Testing will begin after each completed requirement.

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

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. Create automated tests to simulate specific user interactions on your application.
3. Compile our test cases into a JAR file and install it on our test device along with our app.
4. Run the tests and view the test results.
5. Correct any bugs or defects discovered in testing.

2.1.1 Functional / Unit Testing

Test Cases	Functional Design	Internal Design	Code	Testing Procedure
FR 01 User Login	Login Screen	SRS 2.1.2 Use Case: User Login Class Diagram State Diagram	LoginActivity	We plan to test by passing both correct and incorrect credentials and verifying that it responds appropriately.
FR 02 Register User	User Creation screen	SRS 3.2 Use Case: Register as a new user State Diagram	UserCreationActivity	We plan to create a new user using the application, then check database for the user. Do this for multiple users.
FR 03 Choice of list of questions	Quiz Screen	SRS 3.2 Use Case: Participating in a Quiz State Diagram	QuizActivity	We plan to test by choosing a subject, topic, and subtopic and check the list of questions to verify they are random questions from the database based on user's choice of subject, topic and subtopics.
FR 04 Unanswered Question are White	Quiz Screen	Use Case: Participating in a Quiz		When taking quiz, we will make sure unanswered questions stay white until they are answered.
FR 05 Wrong/Right Questions Red/Green	Quiz Screen	Use Case: Participating in a Quiz	QuizActivity	When taking a quiz in SiQuoia, we will visually ensure that answering questions incorrectly and correctly will change the question red and green respectively.

FR 06 Answering a question	Question Screen	SRS 3.2 Use Case: Participating in a Quiz Class Diagram Sequence Diagram	QuestionActivity	When taking a quiz, user selects an answer to a question and answer is correctly record. We will check both the database and the app data to see if the answer has been recorded.
FR 07 Interface to mySQL through Apache server	PHP and Android background scripts	SRS 2.1.4 Use Case: Getting data from the database Deployment Architecture	PHP scripts	We plan to test by executing a command on an android device (such as create a user account) and check the database and the php responses for correct outputs.
FR 08 Update User Information		SRS 2.1.7 Use Case: Exit Game Use Case: Logging out	QuizActivity	Similar to other tests, we will change the user information from the app (by answering a or starting a new quiz) and see if the new information has been stored correctly in database and in app.
FR 09 Leaderboard		SRS 2.2 Class Diagram State Diagram	LeaderBoardActivity	User is able to access the leaderboard from the home screen after login.
FR 10 Redeemable Codes	Redeem Code Screen	SRS 2.2 Use Case: Redeeming quiz-code	HomeActivity	We plan to test by verifying that the correct code will download a Branded Question and correctly display it to the user.

2.1.2 Non-Functional Testing

#	Non-Functional Requirements	SRS sec.	Testing Procedure
NF 01	Shall not be more than 60MB RAM	2.1.6	All testing devices will have more than 60MB of RAM and we will test to see how much memory SiQuoia uses.
NF 02	Shall not use > 30MB of HD space	2.1.6	All testing devices will have more than 30MB of data storage. We will verify the amount of HD space that SiQuoia uses.
NF 03	Database up 90% of Time	2.5	Daily tests will be administered to check Database activity. We will be pinging the database or running the application to get information.

2.2 Integration Testing

We will be combining different modules together to form one big module for a given task. We will combine all modules that relate to each functional requirement, both android code and php code and make sure that integrated module works as expected. For instant, we will integrate the android code and php scripts for creating a user and make sure that they work together with out any flaws.
The combining module will be deploy as Android .APK file for testing different functionality.

2.3 System Testing

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. 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 functions and use cases and make sure the application works as intended.

2.4 Usability Testing

We will test the application from the users' perspective and see if how easy it is to use the app and see if it is responsive. We will go through all the features and test the ease of use.

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.

2.5 Regression Testing

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 and that both the 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.

Every new software update shall be sent to the team to verify that all the requirements have been met and a full verification and validation testing will be performed to check for any new defects that may have been introduced. We will also make sure that all previous functions and modules works as expected and have not broken.

2.6 Performance, Load and Stress Testing

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. 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 to ensure that the performance requirements set out by the SRS are met.

2.7 Installation Testing

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

Upon installing the SiQuoia app on these devices, will use Acceptance tests to ensure that the application is working properly on each device and version of Android.

2.8 Acceptance Testing

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.

Acceptance Tests:

- User Account creation
- Login and logout of SiQuoia
- Buy and download a quiz
- Take a quiz from start to finish
- Leaderboard is accessible
- Multimedia in questions is accessible
- Ability to submit questions for review