

**SYSTEM TEST
DOCUMENTATION
for
QuoteShot**

v1.0 - BlueQuoters

Mertcan Boz

Zeliha Yılmaz

Zeynep Havva Dinç

Burak Aydemir

1. Introduction	4
1.1. Problem Definition	4
1.2 Purpose and Scope	4
1.2 Document Identifier	4
1.3 References	4
1.4 Level in the Overall Sequence	5
2. Details for System Test Plan	5
2.1 Test Items and Their Identifiers	5
2.2 Test Traceability Matrix	6
2.3 Features to be Tested	7
2.4 Features not to be tested	7
2.5 Approach	7
2.6 Item pass/fail Criteria	7
2.7 Test Deliverables	8
3. Test Management	8
3.1. Testing a sub component within itself	8
3.2. Testing the communication protocols and interactions between adjacent sub-system components	8
3.3. Integration of the complete system and testing	9
4. TEST CASE DETAILS	9
4.1 General Testing	9
Use Case - 1 Register	9
Use Case - 1.1 Register	10
Use Case - 2 Login	10
Use Case - 2.1 Login	11
Use Case - 3 Share Quote	11
Use Case - 4 Edit Profile	12
Use Case - 5/6 Like / Unlike	12
Use Case - 7 Comment	13
Use Case 8 / 9 Follow / Unfollow	13
Use Case 8.1 / 9.1 Follow / Unfollow	14
Use Case 10 Send Message	14
Use Case 11 - Convert Image to Text	15
Use Case 12 Search a New User	15
Use Case 13 Save Address Information	16
Use Case 14 Logout	16
4.2 System Testing	17
System Test 1	17
System Test 2	17
System Test 3	18
System Test 4	18
System Test 5	19

<u>5. SYSTEM TEST REPORT DETAILS</u>	<u>19</u>
<u>5.1 Overview of the Test Results</u>	<u>19</u>
<u>5.2 Detailed Test Results</u>	<u>19</u>
<u>5.2.1 General Testing</u>	<u>20</u>
<u>5.2.2 System Testing</u>	<u>20</u>
<u>5.3 Rationale for Decisions</u>	<u>21</u>
<u>5.4 Conclusions and Recommendations</u>	<u>21</u>

1. Introduction

1.1. Problem Definition

The purpose of the project “QuoteShot” is to provide a social platform project where the users can share their favourite snapshots of the quotes from the books they read, in their feeds or other social media platforms such as Facebook. They shall also be able to share the quotes in text form generated by the application's text recognition. Trending lists of books of different categories will be generated periodically, based on the sharings of users. On top of all the features, users who are locally close enough, shall be able to exchange books.

1.2 Purpose and Scope

The purpose of this document is to provide the test cases of the QuoteShot project. It defines the objective, scenario, expected outcomes and procedural requirements for each test case. It also includes a table showing which test case is related to which one. The software will be tested using guidance of this document. Although it covers all the test cases specifically in detail, a little portion of the details is subject to change in test phase.

1.2 Document Identifier

This document is prepared by BlueQuoters with the purpose of testing the QuoteShot software. The document explains design of test cases and procedure with great detail so that any tester would be able to run and observe the outcomes. This document is the first version of Software Testing Documentation.

1.3 References

- QuoteShot Software Requirements Specification (SRS)
- QuoteShot Software Design Descriptions (SDD)
- IEEE STD 829-2008

1.4 Level in the Overall Sequence

The hierarchy between different level testing methods such as unit testing, which checks the correctness of the individual modules of the software; or integration testing that verifies compatibility between software elements of the project, does not take part in this document. Only system integration testing, which is defined as the testing of the sub-systems together in order to ensure that they work correctly as a system, is described.

2. Details for System Test Plan

This section describes the specific items to be tested at different levels and provides a Test Traceability Matrix that links the items to be tested with the requirements.

2.1 Test Items and Their Identifiers

The test items are the functionalities specified in QuoteShot Software Requirement Specification document Functional Requirements(2.1) section. These functionalities are described by means of use cases. The test items are listed below with their use case identifiers.

Use Case 1 - Registration: A non member user registers to the system.

Use Case 2 - Login: User logs in to the system.

Use Case 3 - Follow: A member user follows another member.

Use Case 4 - Unfollow: A member user unfollows another member.

Use Case 5 - Share a Quote: A member user shares quotes from books.

Use Case 6 - Edit Profile: A member user edits their profile information.

Use Case 7 - Like: A member user likes a post in the system.

Use Case 8 - Unlike: A member user unlikes a post in the system.

Use Case 9 - Comment: A member user comments on a post in the system.

Use Case 10 - Send Message: A member user sends a message to another member.

Use Case 11 - Convert Image To Text Form : A member can convert his image form of quote into the text form.

Use Case 12 - Save Address Information : A member can save his address information by using GPS technology.

Use Case 13 - Search a New User : A member can search another users of QuoteShot.

Use Case 14 - Book Exchange: A user either offers their book for an exchange or searches a book.

Use Case 15 - Logout: A user logs out from the system.

2.2 Test Traceability Matrix

The test traceability matrix is given below. The first row specifies the corresponding use case. Use cases are denoted by UC and use case number. They correspond to the use cases listed in section 2.1. The test cases are denoted by TC and test case number and are listed in the first column. Since we have 7 different experiments we will denote for each experiment only one traceability matrix to show each use case is matched the shown test case for each scene.

	TC1	TC2	TC3	TC4	TC5	TC6	TC7	TC8	TC9	TC10	TC11	TC12	TC13	TC14	TC15
UC1	+														
UC2		+													
UC3			+												
UC4				+											
UC5					+										
UC6						+									
UC7							+								
UC8								+							
UC9									+						
UC10										+					
UC11											+				
UC12												+			
UC13													+		
UC14														+	
UC15															+

2.3 Features to be Tested

The features to be tested are listed in section 2.1. In addition to testing system's functionality and user interface we will test the system's results whether it works correctly or not. Moreover we will run the system in multiple environment and test performance.

2.4 Features not to be tested

Open source libraries usually conduct their own tests, so tests for those libraries will only take place in integration and unit tests of software implemented. Also, hardware of all phones can not be tested. Security tests will also be postponed for this version of document.

2.5 Approach

All of the functionalities will be tested with knowing their internal structure and workings of the application.

Mobile Application Component Level: All tests regarding to mobile application will be handled using analysis test method.

System Database Level: Database integrity tests shall be held using black-box method.

2.6 Item pass/fail Criteria

For use cases testing for each experiment a test if and only if pass when we give an input that is specified in this use case and produces a result that matches which were described in SRS. If the results do not match the test will fail. The test will fail also when it produces wrong results.

2.7 Test Deliverables

Test deliverable for the project is this document. This document includes level test plans, level test cases and level test reports. Section 1, Section 2 and Section 3 cover test plans. Section 4 covers level test cases and Section 5 covers level test report.

3. Test Management

Best way to test the whole system is creating sub-system tests. This approach is superior to complete system testing because sub components are pretty big already. Their interactions and communications with each other should be well defined and work steadily. Finally the complete system integration and testing must be performed. Thus the workload of testing schema is as follows:

3.1. Testing a sub component within itself

This process ensures that the component is working without any interaction by any other sub components. Our processes has 4 module. First of them is social media issues. We tested that user can register/login in out application and share photo of quote, like/unlike, comment of quote follow/unfollow another user and update his profile. Second is OCR which converts image from text. This part does not work %100 correctly. Third part is GPS. We tested the application can take user's location. Final part is Recommendation. We only test book to book recommendation.

3.2. Testing the communication protocols and interactions between adjacent sub-system components

Purpose of this part is stabilize inter sub-system communications. Techniques are mostly composed of erroneous situation generation. As defects reveals the real life situations more realistically. Sending an invalid JSON object from intermediate component to cloud server might be an example. In fact, when exactly an exception occurs and how sub-systems reacts those are the heart of this test activity.

3.3. Integration of the complete system and testing

This is the final test scenario and describes how will the complete system work when it is deployed in real life. We tested GPS, OCR and recommendation part with integrated of our database. Our application save users location users booklist and users quotes database.

4. TEST CASE DETAILS

4.1 General Testing

In this section test case identifiers, objectives, inputs and outcomes, environmental needs, special procedural requirements, intercase dependencies for each functional test case are provided as tables. We will provide all use case's test cases for each experiment in this section.

Use Case - 1 Register

Test Case Identifier	QuoteShot - Register - 01
Objective	To test the register component of the system.
Scenario	<ol style="list-style-type: none">1. User clicks Do you like to register ? button on application launch page.2. User fills the required fields with valid informations.3. User clicks register button.
Input	email, username, password.
Outcome	A new user is created in the system User is redirected to Feed page.
Requirements	The application shall be started.

Use Case - 1.1 Register

Test Case Identifier	QuoteShot - Register - 02
Objective	To test the register component with the same username.
Scenario	<ol style="list-style-type: none">1. User clicks Do you like to register ? button on application launch page.2. User fills the username field with the already existed username.3. User clicks Register button
Input	email, username, password
Outcome	The user is warned to enter a valid username to register to application.
Requirements	The application shall be started.

Use Case - 2 Login

Test Case Identifier	QuoteShot - Login- 03
Objective	To test login system successfully.
Scenario	<ol style="list-style-type: none">1. User fills username2. User fills password.3. User clicks "LOGIN" button.4. Application shows successfull message.5. Application shows Feed Page.
Input	Username and password
Outcome	Successfull message and feed page
Requirements	Correction of username and password

Use Case - 2.1 Login

Test Case Identifier	QuoteShot - Login - 04
Objective	To test logging with wrong inputs
Scenario	<ol style="list-style-type: none">1. Users fills username.2. User fills password.3. User clicks "LOGIN" button.4. Application shows error message.
Input	Username and password
Outcome	Error Message
Requirements	Wrong username and password

Use Case - 3 Share Quote

Test Case Identifier	QuoteShot -Share Quote-05
Objective	To test that user can share post or not
Scenario	<ol style="list-style-type: none">1. User clicks camera button.2. Users takes a photo of any quote.3. User selects cropped area.4. User saves photo.5. User writes comments.6. User writes tags.7. User clicks "QuoteShot" button.8. The photo of quote shall be posted in Feed and Profile page.
Input	Photo, comment and tags.
Outcome	Showing shared Quote Feed and Profile page.
Requirements	Phone should has a camera.

Use Case - 4 Edit Profile

Test Case Identifier	QuoteShot - Edit Profile - 06
Objective	To test user can edit his profile.
Scenario	<ol style="list-style-type: none">1. User goes to his profile page.2. User opens hamburger menu from his profile page.3. User fills the field which he wants to change.4. User clicks save button.
Input	User credentials which will be edited.
Outcome	Succesfull edit profile.
Requirements	User should be loggin to the application.

Use Case - 5/6 Like / Unlike

Test Case Identifier	QuoteShot- Like/Unlike-07
Objective	To test user can like quote.
Scenario	<ol style="list-style-type: none">1. User clicks "Like" button under the quote.2. If the user did not click before the quote will be liked and the number of like will be increased.3. If the click "Like" button before the quote will be unliked and the number of like will be decreased.
Input	Shared quote
Outcome	Successful like or unlike Showing number of users who like
Requirements	User should be loggin to the application.

Use Case - 7 Comment

Test Case Identifier	Quoteshot - Comment -08
Objective	To test user can comment a post.
Scenario	<ol style="list-style-type: none">1. User clicks the comment button under the quote.2. User enters his comment in the opened field.3. User clicks Send button.
Input	Shared Quote
Outcome	Successful comment.
Requirements	User should be loggin to the application.

Use Case 8 / 9 Follow / Unfollow

Test Case Identifier	QuoteShot-Follow/Unfollow(Search)-09
Objective	To test user can follow/unfollow another user with searching.
Scenario	<ol style="list-style-type: none">1. User clicks "Search" button.2. User fills username.3. User clicks any of users in search list.4. User clicks "follow/unfollow" button.5. The application shows successful message.
Input	User
Outcome	Successful Follow/Unfollow
Requirements	User should be loggin to the system.

Use Case 8.1 / 9.1 Follow / Unfollow

Test Case Identifier	QuoteShot-Follow/Unfollow(List)-10
Objective	To test user can follow/unfollow another user with editing following and follower list.
Scenario	<ol style="list-style-type: none">1. User clicks "Profile" button.2. User clicks following/follower lists button.3. User clicks follow/unfollow button any of user in the list.4. The application shows successful message.
Input	User
Outcome	Successful Follow/Unfollow
Requirements	User should be loggin to the system.

Use Case 10 Send Message

Test Case Identifier	QuoteShot- Send Message -11
Objective	To test if user can send a message to another user.
Scenario	<ol style="list-style-type: none">1. User goes to other user profile.2. User clicks Send Message button.3. User writes his message.4. User clicks sends icon.
Input	Message
Outcome	Successfull sending message.
Requirements	User should be loggin to the application.

Use Case 11 - Convert Image to Text

Test Case Identifier	QuoteShot - Convert Image to Text-12
Objective	To test if user can convert his image form of quote to text form.
Scenario	<ol style="list-style-type: none">1. User opens camera by clicking Camera icon.2. User takes a photo of quote which he wants to share.3. User crops his image.4. User clicks Convert Text button.
Input	Image form of Quote.
Outcome	Text form of a quote.
Requirements	User should be loggin to the system. Phone should has camera.

Use Case 12 Search a New User

Test Case Identifier	QuoteShot - Search a New User - 13
Objective	To test if user can search other users.
Scenario	<ol style="list-style-type: none">1. User goes Search Page.2. User enters username of other user who is wanted to be search in the search field.3. The application display this user.
Input	Username of the other user.
Outcome	Other users profile page.
Requirements	User should be loggin to the system.

Use Case 13 Save Address Information

Test Case Identifier	QuoteShot - Save Address Information-14
Objective	To test user can update address information.
Scenario	<ol style="list-style-type: none">1. User clicks "Profile" button.2. User clicks "Edit Profile" button.3. User clicks "Change Address" button.4. If the user's GPS is not open the application directs settings of phone.5. If the user's GPS is open the application saving his address by using GPS.
Input	Location of users.
Outcome	Saving location of users.
Requirements	Phone should has GPS.

Use Case 14 Logout

Test Case Identifier	QuoteShot - Logout- 15
Objective	To test if user can logout from the system.
Scenario	<ol style="list-style-type: none">1. User goes to his profile page.2. User opens hamburger menu from the profile page3. User clicks logout button.
Input	-
Outcome	Succesfull logout.
Requirements	User should be loggin to the system.

4.2 System Testing

System Test 1

Test Case Identifier	Add Book To Database
Objective	Testing the adding book to book database.
Scenario	When user wants to add book to his book list if the wanted book is not available in the database, the system search this book on the internet and after user clicks ADD button, the book is added to database successfully.
Input	Book Name
Outcome	Adding book to database.
Requirements	User should enter the valid book name.

System Test 2

Test Case Identifier	Adding Tags About Related Quote
Objective	Testing to adding tags.
Scenario	When user wants to share a quote from a specific book after he choosing the book name, the application brings related tags from internet.
Input	Book name.
Outcome	Related tags.
Requirements	User should choose a book name.

System Test 3

Test Case Identifier	Book to Book Recommendation
Objective	Testing to book to book recommendation.
Scenario	When user shares a quote, after he choosing the book name, author and category, the application recommends a book from the same author and category.
Input	Book Name, Author Name and Category.
Outcome	Recommended book with similar features.
Requirements	User should select the book name and author while sharing a quote.

System Test 4

Test Case Identifier	Quote to Quote Recommendation
Objective	Testing to quote to quote recommendation.
Scenario	When user shares a quote the application recommends a similar quote by using tags and context based algorithms.
Input	Quote
Outcome	Recommended similar quotes.
Requirements	User should share a quote with meaningful tags.

System Test 5

Test Case Identifier	Sending Notification
Objective	Testing to notification functionality of the application.
Scenario	After one user sends a message to another user or like / unlike / comment a post the notification is sent.
Input	Message or post.
Outcome	Notification is sent.
Requirements	Messaging between two user or a post.

5. SYSTEM TEST REPORT DETAILS

The following sections describe the overview of the test results, the detailed test results, the rationale for decisions and the conclusions and recommendations.

5.1 Overview of the Test Results

All test cases whose details are explained in Section 4 are applied to the product.. Environmental needs, procedural requirements and intercase dependencies are taken into account for each test case. Consequently, some test cases are performed by one user, while others are performed by two users simultaneously.

5.2 Detailed Test Results

Test results for each test case are specified in this section.

5.2.1 General Testing

General Test 1	PASSED
General Test 2	PASSED
General Test 3	PASSED
General Test 4	PASSED
General Test 5	PASSED
General Test 6	PASSED
General Test 7	PASSED
General Test 8	PASSED
General Test 9	PASSED
General Test 10	FAILED
General Test 11	FAILED
General Test 12	PASSED
General Test 13	PASSED
General Test 14	PASSED

5.2.2 System Testing

System Testing 1	PASSED
System Testing 2	PASSED
System Testing 3	PASSED
System Testing 4	FAIL
System Testing 5	PASSED

5.3 Rationale for Decisions

Since the project is not completed to its final extent, some test cases failed to result in expected outcomes. These failures can be fixed by changing the structure of some parts.

5.4 Conclusions and Recommendations

The Revenge system passes 70% of test cases. This figure reveals the fact that the project is 70% completed. Since the system is developed well-structured from the beginning, it will be easy to introduce missing features. However, in order to release the product prototype in an acceptable time, product development should be completed as soon as possible to obtain 100% success on system test.