Test plan

*Tianwei Zhang*

June 17, 2020

Table of Contents

[1 Introduction 2](#_Toc43386310)

[1.1 Objectives 2](#_Toc43386311)

[1.2 Team members 2](#_Toc43386312)

[2 Features/units to be tested 2](#_Toc43386313)

[3 Approach 2](#_Toc43386314)

[4 Test cases 2](#_Toc43386315)

[4.1 Test Case ID: 001 2](#_Toc43386316)

[4.2 Test Case ID: 002 4](#_Toc43386317)

[4.3 Test Case ID: 003 5](#_Toc43386318)

[4.4 Test Case ID: 004 6](#_Toc43386319)

[4.5 Test Case ID: 005 7](#_Toc43386320)

[4.6 Test Case ID: 006 8](#_Toc43386321)

[4.7 Test Case ID: 007 9](#_Toc43386322)

[4.8 Test Case ID: 008 10](#_Toc43386323)

[5 Schedule 11](#_Toc43386324)

[5.1 Test Schedule 11](#_Toc43386325)

[5.2 Deliverables 11](#_Toc43386326)

# Introduction

This test plan would test all the functions and all the UI elements in the assignment.

The database should be interact with the program and server API would be called as well

## Objectives

To make sure the program can match the requirement from client

## Team members

|  |  |
| --- | --- |
| **Name** | **Role** |
| *Qiao Li*  *Tianwei Zhang* | Test Manager  Developer, Tester |

# Features/units to be tested

* All the UI elements
* Page navigation
* Light and dark mode
* Sorting
* Searching
* Insert function with Local database
* Update function with Local database
* Delete function with Local database
* Mobile Features
* Server API call

# Approach

In this test plan the black box testing would be used.

# Test cases

## Test Case ID: 001

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID**: 001  **Executed by**: \_\_\_\_\_Tianwei Zhang\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Execution Date**: \_\_\_\_\_\_17/06/2020\_\_\_\_\_\_\_\_\_  **Scenario**: Run the program and test all the UI elements  **Test Case**: UI Elements and page navigation | | | | |
| Prerequisite | | Able to run the program in the emulator | | |
| Test Data | | *No test data for this test case* | | |
| Step | Action | | Expected Result | Success? |
| 1 | Start the program, check the welcome page UI | | The Welcome page should have a heading with welcome and three buttons Which are dark, light and camera. | **or X** |
| 2 | Swipe the Welcome page | | List page should show, in the list page, there should be a contact list, a hash table and five floating action buttons |  |
| 3 | Click on the add button | | In the add page, four text boxes should be display for name, phone, email and date |  |
| 4 | Click the back button, and click one of the contacts | | Details page should be shown, the details of this contact should be display in four different text box and three buttons should be displayed, which are call, SMS and back. |  |
| 5 | Click the SMS | | A SMS dialog box should be shown and ask for the message to send. |  |
| 6 | Click the back button, and click back again, swipe the contact | | The update button should be shown, click the update button would navigate to update page.  In the update page, four text box should be display with all the details inside, and there should be one update button in the screen. |  |
| 7 | Click the back button, and click one more back, navigate to welcome page, click on the use camera | | The camera page should be display with an empty image view and a camera button |  |

## Test Case ID: 002

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID**: 002  **Executed by**: \_\_\_\_\_Tianwei Zhang\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Execution Date**: \_\_\_\_\_\_17/06/2020\_\_\_\_\_\_\_\_\_  **Scenario**: Run the program and test the dark and light mode  **Test Case**: Dark mode and light mode | | | | |
| Prerequisite | | The program should be run successfully | | |
| Test Data | | *No test data required for this test case* | | |
| Step | Action | | Expected Result | Success? |
| 1 | Start the program click on the dark button | | The mode for the program should change to the dark mode | **or X** |
| 2 | Swipe the welcome page | | The list page should also be in dark mode |  |
| 3 | Turn off the program and start again | | The default mode for the program should be change to dark mode, hence, the welcome page should be in dark mode |  |
| 4 | Click on the light button | | The Program should now be on the light mode |  |
| 5 | Swipe the welcome page | | The list page should also be in light mode |  |
| 6 | Turn off the program and start again | | The default mode fort the program should be change to light mode, hence, the welcome page should be in light mode |  |

## Test Case ID: 003

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID**: 003  **Executed by**: \_\_\_\_\_Tianwei Zhang\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Execution Date**: \_\_\_\_\_\_17/06/2020\_\_\_\_\_\_\_\_\_  **Scenario**: Run the program and test the sorting, hash table and searching  **Test Case**: Sorting, hash table navigate and search | | | | |
| Prerequisite | | Able to run the program and display the list page | | |
| Test Data | | *No test data for this test case* | | |
| Step | Action | | Expected Result | Success? |
| 1 | Start the program, Swipe the welcome page, in the list page click on the z- a button | | All the record should be sorting from z to a | **or X** |
| 2 | Click on the a – z button | | All the record should be sorting from a to z |  |
| 3 | Click on the “H” in the hash table | | The page should be scroll to position which list start with h |  |
| 4 | Click on the “C” in the hash table | | The page should be scroll to position which list start with c |  |
| 5 | Click on the search button | | Search bar should show |  |
| 6 | Enter “som” | | All contact which the name contains som would be shown in the list. |  |
| 7 | Enter “$@!@#!” | | No record contains “$@!@#!” so there should show no contact. |  |

## Test Case ID: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID**: 004  **Executed by**: \_\_\_\_\_Tianwei Zhang\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Execution Date**: \_\_\_\_\_\_17/06/2020\_\_\_\_\_\_\_\_\_  **Scenario**: Run the program and test the adding  **Test Case**: Adding function | | | | |
| Prerequisite | | The program should run correctly, and the local database should be set up. | | |
| Test Data | | *“J Smith”,”049183901”,” Smith@abc.com”,”11-01-2001”* | | |
| Step | Action | | Expected Result | Success? |
| 1 | Start the program, go to the list page and click on the add button | | The add page should show. | **or X** |
| 2 | Enter all the detail and click the add button in add page | | The contact should be added successfully, and user would be navigating to the list page |  |
| 3 | Click on the J in the hash table | | The contact just entered display in the page |  |
| 4 | Open the device explorer and save the data to computer and open the DB Browser to check the data | | The contact: *“J Smith”,”049183901”,” Smith@abc.com”,”11-01-2001” should be in the database as well* |  |

## Test Case ID: 005

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID**: 005  **Executed by**: \_\_\_\_\_Tianwei Zhang\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Execution Date**: \_\_\_\_\_\_17/06/2020\_\_\_\_\_\_\_\_\_  **Scenario**: Run the program and test the update in both emulator and local database  **Test Case**: Updating function | | | | |
| Prerequisite | | Able to run the program and swipe one of the contact should show the edit button | | |
| Test Data | | *“J Smith for test”,”049183901”,” Smith@abc.com”,”11-01-2001”* | | |
| Step | Action | | Expected Result | Success? |
| 1 | Start the program, swipe the welcome page, and then swipe the J Smith contact | | The edit page should be show with all the data inside. | **or X** |
| 2 | Change the details base on the test data and click the update button | | The record should be successfully added to the database, and the emulator would be directed to the list page. |  |
| 3 | Click the “J” in the hash table | | The window would be scroll to the position start with J, and then we can see the contact that we just updated. |  |
| 4 | Open the device explorer and save the data to computer and open the DB Browser to check the data | | The data that we just update is successfully updated in the database as well. |  |

## Test Case ID: 006

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID**: 006  **Executed by**: \_\_\_\_\_Tianwei Zhang\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Execution Date**: \_\_\_\_\_\_17/06/2020\_\_\_\_\_\_\_\_\_  **Scenario**: Run the program and test the deleting in both emulator and local database  **Test Case**: Deleting function | | | | |
| Prerequisite | | Able to run the program and long click a contact can drag the contact. | | |
| Test Data | | *No test data for this test case* | | |
| Step | Action | | Expected Result | Success? |
| 1 | Start the program and swipe the welcome page, select one of the contacts, long click and drag it drop to the bin icon | | A confirm dialog box show to ask if user really want to delete this contact. | **or X** |
| 2 | Click the no button | | Message show delete canceled, and nothing was deleted |  |
| 3 | Redo the previous step, drag, and drop one contact to the bin icon | | Confirm dialog box show as for confirmation |  |
| 4 | Click the yes button | | The contact is deleted from the emulator |  |
| 5 | Open the device explorer and save the data to computer and open the DB Browser to check the data | | The contact is deleted from the local database |  |

## Test Case ID: 007

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID**: 007  **Executed by**: \_\_\_\_\_Tianwei Zhang\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Execution Date**: \_\_\_\_\_\_17/06/2020\_\_\_\_\_\_\_\_\_  **Scenario**: Run the program and test all the mobile features  **Test Case**: Mobile Features | | | | |
| Prerequisite | | The program should be successfully run and have the permission about send SMS, call, and use camera. | | |
| Test Data | | *“Sample Message”* | | |
| Step | Action | | Expected Result | Success? |
| 1 | Start the program, in the welcome page, click on the camera button | | Camera page should be show | **or X** |
| 2 | Click on the camera button | | The program would start using the camera |  |
| 3 | Click on the camera icon in the middle | | The program would take this photo |  |
| 4 | Click the **** Button | | The image would be save and display in the empty image view |  |
| 5 | Click the back button, swipe the welcome page, click one of the contacts | | The detail page would be show |  |
| 6 | Click on the call button | | The Program would make a phone call using the contact phone number |  |
| 7 | Click on the SMS button | | A dialog box would show, ask user to enter the Message that he want to send |  |
| 8 | Enter the test data and click on send | | Message sent, the message would be the text that user entered, the phone number would be this contact phone number |  |

## Test Case ID: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID**: 008  **Executed by**: \_\_\_\_\_Tianwei Zhang\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Execution Date**: \_\_\_\_\_\_17/06/2020\_\_\_\_\_\_\_\_\_  **Scenario**: The Server API program should be run during the testing  **Test Case**: Server API call | | | | |
| Prerequisite | | The program should be run successfully and able to connect to the internet, also the API should be working correctly | | |
| Test Data | | *No sample data should needed in this test case* | | |
| Step | Action | | Expected Result | Success? |
| 1 | Go to the server database, for example, SSMS (Microsoft SQL Server Management Studio) empty all the record in the database | | There would be no record in the database. | **or X** |
| 2 | Start the visual studio Server API Program | | The server is running in the background |  |
| 3 | Start the Android Program and swipe the list page, click on the sync button | | Nothing exactly changes in the list page |  |
| 4 | Go to the SSMS and refresh the database. | | All the data from the Android Program would be insert to the server-side database. |  |

# Schedule

## Test Schedule

|  |  |  |
| --- | --- | --- |
| **Task** | **Start** | **Finish** |
| Working on the test base on this plan | 18/06/2020 | 18/06/2020 |

## Deliverables

|  |  |
| --- | --- |
| **Task** | **Date** |
| Test Plan | 17/06/2020 |
| Test Results | 18/06/2020 |
| Test Report | 18/06/2020 |