Test plan

*Jonathan Richardson*

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 is for functionality of all the UI elements as well as the database/API functionality.

## Objectives

To make sure the program can match the requirement from client

## Team members

|  |  |
| --- | --- |
| **Name** | **Role** |
| *Qiao Li*  *Jonathan Richardson* | 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**: Jonathan Richardson  **Execution Date**: 02/07/2020  **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 saying welcome and there should be buttons for the light and dark mode, as well as camera. | **** |
| 2 | Swipe the Welcome page | | The list page will show. | **** |
| 3 | Click on the add button | | The add page should show with the text boxes to enter a new contact and the add button. | **** |
| 4 | Click the back button, and click one of the contacts | | Contact details should be shown. | **** |

## Test Case ID: 002

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID**: 002  **Executed by**: Jonathan Richardson  **Execution Date**: 02/07/2020  **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 colour scheme should change. | **** |
| 2 | Swipe the welcome page | | The list page will be in dark mode | **** |
| 3 | Turn off the program and start again | | The previously selected colour scheme should be the same as before you closed it. | **** |
| 4 | Click on the light button | | The colour scheme should change back to light mode | **** |
| 5 | Swipe the welcome page | | The list page will be in light mode. | **** |
| 6 | Turn off the program and start again | | The previously selected colour scheme should be the same as before you closed it. | **** |

## Test Case ID: 003

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID**: 003  **Executed by**: Jonathan Richardson  **Execution Date**: 02/07/2020  **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 records will be sorted reverse alphabetically. | **** |
| 2 | Click on the a – z button | | All the records will be sorted alphabetically. | **** |
| 3 | Click on the “z” in the hash table | | The page will center on the contacts under Z. | **** |
| 4 | Click on the “D” in the hash table | | The page will center on the contacts under D. | **** |
| 5 | Click on the search button | | Search bar should show. | **** |
| 6 | Enter “Jon” | | All contacts named Jon should show. | **** |
| 7 | Enter “!!!!” | | No recorded contact with that name so it won’t show. | **** |

## Test Case ID: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID**: 004  **Executed by**: Jonathan Richardson  **Execution Date**: 02/07/2020  **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 | | *“Jon”,”05123123”,” jon@gmail.com”,”02/04/2000”* | | |
| 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. | **** |
| 2 | Enter all the detail and click the add button in add page | | The contact will be added and you will be redirected to the list view. | **** |
| 3 | Click on the J in the hash table | | The contact just entered should be in the center. | **** |
| 4 | Open the device explorer and save the data to computer and open the DB Browser to check the data | | *“Jon”,”05123123”,”* [*jon@gmail.com”,”02/04/2000*](mailto:jon@gmail.com”,”02/04/2000)*”*  *Added to database.* | **** |

## Test Case ID: 005

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID**: 005  **Executed by**: Jonathan Richardson  **Execution Date**: 02/07/2020  **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 | | *“Jon”,”05123123”,” jon@gmail.com”,”02/04/2000”* | | |
| 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. | **** |
| 2 | Change the details base on the test data and click the update button | | The Record will be successfully added to the local db. | **** |
| 3 | Click the “J” in the hash table | | The window will head to the letter in the hash table and show all contacts in the list starting with j. | **** |
| 4 | Open the device explorer and save the data to computer and open the DB Browser to check the data | | Database updated successfully. | **** |

## Test Case ID: 006

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID**: 006  **Executed by**: Jonathan Richardson  **Execution Date**: 02/07/2020  **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 | | Confirmation box should display asking if the user wishes to delete the contact. | **** |
| 2 | Click the no button | | A message will show saying that the contact wasn’t deleted. | **** |
| 3 | Redo the previous step, drag, and drop one contact to the bin icon | | Confirmation box should display asking if the user wishes to delete the contact. | **** |
| 4 | Click the yes button | | The contact is deleted. | **** |
| 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**: Jonathan Richardson  **Execution Date**: 02/07/2020  **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 show. | **** |
| 2 | Click on the camera button | | The camera screen will show. | **** |
| 3 | Click on the camera icon in the middle | | The picture will be taken and you’ll be directed to the gallery page. | **** |
| 4 | Click the **** Button | | Image is saved to gallery. | **** |
| 5 | Click the back button, swipe the welcome page, click one of the contacts | | Detail page will show. | **** |

## Test Case ID: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID**: 008  **Executed by**: Jonathan Richardson  **Execution Date**: 02/07/2020  **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 | | No records currently in the database. | **** |
| 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 | | No changes in the list page. | **** |
| 4 | Go to the SSMS and refresh the database. | | All data from the app will be inserted into the server side DB. | **** |

# Schedule

## Test Schedule

|  |  |  |
| --- | --- | --- |
| **Task** | **Start** | **Finish** |
| Working on the test base on this plan | 02/07/2020 | 02/07/2020 |

## Deliverables

|  |  |
| --- | --- |
| **Task** | **Date** |
| Test Plan | 02/07/2020 |
| Test Results | 02/07/2020 |
| Test Report | 02/07/2020 |