SMART LOGICS

# ReserveMe
# Master Test Plan

**Version 1.0**

# Revision History

| Date | Version | Description | Author |
|---|---|---|---|
| 05/June/2016 | 1.0 | Master test plan | Saranga Ekanayaka |
| | | | |
| | | | |
| | | | |

# Table of Contents

# Master Test Plan

## 1. Evaluation Mission and Test Motivation

The “ReserveMe” is a mobile application for make hotel bookings. Hotel owners can register their hotel into the system and hotel customers can make reservations. This documentation is about the overall test plane and methodologies used in testing process of this application.

Software testing is verify that “ReserveMe” meet the deferent requirement including functional, security, performance, memory, reliability and usability. And also this validate that the system being developed with all the user needs. Testing is help to find mistake done by the system developer. The main motivation for testing is to ensure that things we created do what it is supposed to do.

Later testing stages of the project is includes quality and performance testing and some of integration tastings.

The mission of this test plan includes

- Find out the errors and bugs inside the codes.
- Debugging.
- Discovering risks and important problems of the project.
- Ensure that the results of each methods are accurate, consistent and reliable.
- Making sure of the customers are satisfy with the system and system meet their all needs.
- To enhance the quality and performance of the application.
- Ensuring that the application should not result into any failures because it can be very expensive in the future or in the later stages of the development.
- Verify system is compatible with deferent devices and operating systems.

Finally, we need software testing because after we release the system to the users, we have to guarantee about all the works of the system.

## 2. Target Test Items

Target test items can be categories as follow,

- Software Items
    - UIs of the ReserveMe
    - Images compatibility of devices with deferent size of screen and resolutions.
    - Compatible with Android OS.
    - Testing all the functionalities of the user interface components.
    - Orientation of the component on the layout.
    - Data loading.
    - Event handling.

- Navigations across different activities.

- Hardware
  - Compatibility with the key board of deferent devices.
  - Change activity with the back button of the device.
  - Compatibility of with the deferent devices with deferent performance levels.

## 3. Test Approach

Testing of the ReserveMe is carried out under several categories. They are,

- Data and Database Integrity Testing
- Function Testing, User Interface Testing
- Performance Profiling
- Load Testing
- Security and Access Control Testing
- Failover and Recovery Testing
- Configuration Testing

These categories would comprise both manual and automated testing appropriately.

### 3.1 Testing Techniques and Types

#### *3.1.1 Data and Database Integrity Testing*

| | |
|---|---|
| Technique Objective: | The objectives of data and database integrity testing are to check whether data is properly translate with in short period without any corruption between database and ReserveMe mobile application and data access methods and processes function properly and without data corruption. This includes observing and logging incorrect functioning target behavior or data corruption. |
| Technique: | Several testing technique are used to verify the data and database integrity<br>Unit testing using several test cases.<br>Several techniques are used for checking the data and database integrity.<br>Unit testing. After loading the hotel data from the online web server, they are add into an array list. A unit test is written to invoke elements from this array list.<br>Inspecting the hotel map interface and check whether they are properly loaded onto the map or no.<br>Check values of variables by running application in debug mode. |

| | |
|---|---|
| Oracles: | This can be achieved through unit testing, debugging and manual inspecting.<br>In unit testing test outcomes of methods using test cases with known outputs.<br>This is achieved through method, "assertEqual(Expected,Actual)".<br><br>For an example<br>Assert.AreEqual("Gonapinuwala",getLocation("Rathmahara"))<br><br>Human oracles include manual verification of results can be carried out by inspecting the interfaces check if they are properly populated.<br><br>Running app with debug mode and check values of variables via android studio debug monitor. |
| Required Tools: | • Unit test – junit:4.12<br>• Android studio 1.4 debug monitor<br>• Genymotion virtual devices. |
| Success Criteria: | In unit testing assertEqual(Expected,Actual) method is determine actual outcomes are same as the expected values or no. And also this is detects the exceptions if there is an exception then test case will fail.<br>The success of manual testing can be determined by checking if the user interface and activities are populated as expected.<br>If debug monitor shows expected values for variable at particular point. |
| Special Considerations: | If internet connection will fail then cannot load data into application. |

### *3.1.2 Function Testing*

Main functionalities of ReserveMe are,

- Reservation and booking
- View hotels
- Rate service of hotel
- Allows users to create a new account

Testing related to some of above functions are describe below

#### 3.1.2.1 Reservation and booking

| | |
|---|---|
| Technique Objective: | In reservation part most critical part which need a testing is calculating total payment. In this payment calculation take charge per day for each room and number of days booked as inputs and return the total payment for customer. The testing of this function is check if this payment calculation give the correct payment. |
| Technique: | Several techniques are used for checking the functionality.<br>Unit testing. A unit test method is generated to compare the calculated payment of the booking activity with the actual amount. Automated test could also be carried out for further clarification.<br><br>Debugging window of Android studio can be used to check the value changers with the payment calculations. |
| Oracles: | Heuristic oracles can be achieved through unit testing and debugging. In hear test outcomes of methods using test cases with known outputs. This is achieved through junit testing method, "assertEqual (Expected,Actual)".<br><br>For an example -:<br>assertEqual(3750,getPayment(750,5))<br><br>Human oracles include manual verification of results. It can be carried out by inspecting the booking table of the UI to check if is it contain the actual payment details.<br><br>Testing the implementation of an algorithm for validation and verification. |
| Required Tools: | • Unit test – junit:4.12<br>• Android studio 1.4 debug monitor<br>• Genymotion virtual devices. |
| Success Criteria: | The success of manual testing can be determined by checking if the booking table shows expected payment values.<br>In unit testing assertEqual(Expected,Actual) method is determine actual outcomes are same as the expected values. And also this is detects the exceptions if there is an exception then test case will fail. |
| Special Considerations: | Internet connectivity should be on for take price values from the server. |

#### 3.1.2.2 View hotel

| | |
|---|---|
| Technique Objective: | This function shows all the hotels system available for reservations. Those are display on a Google map. The main objective of this testing is check if this map displays all the available hotels in registered in the system with corresponding hotel names and location name. |
| Technique: | Several techniques are used for checking the functionality.<br><br>Inspecting the map activity of the all hotels interface to check if it is properly populated as intended and all records are included properly.<br><br>Inspecting the map activity of the all hotels interface to check if it is properly populated as intended and all records are included properly. |
| Oracles: | Human oracles include manual verification of results. It can be carried out by inspecting the map of the UI to check if the map contain markers for all the hotels.<br>Debugged hardware simulator to emulate hardware and software operations. |
| Required Tools: | • Android studio 1.4 debug monitor<br>• Genymotion virtual devices. |
| Success Criteria: | The success of manual testing can be determined by checking if the map shows all the markers of hotels as expected. |
| Special Considerations: | Internet connectivity is important to work with Google map. |

### *3.1.3 User Interface Testing*

| | |
|---|---|
| Technique Objective: | Main objective of user interface (UI) testing is to ensure that "ReserveMe" application meets its functional requirements and achieves good standard of quality such that it is more likely to be successfully adopt by user. Ultimately it is ensure that the user interfaces provide the user with the appropriate access and navigation through the functions.<br><br>Test the test the compatible of application with the back button ant key board of the mobile devices.<br><br>UI testing is includes,<br>• Alignments of UI component on the Activity layout.<br>• Deferent views of the layouts.<br>• Event handling functions of the Activities.<br>• Functionalities of Android manifest.<br>• Compatibility of the UIs with different size of devices.<br>• Testing all the functionalities of the user interface components. |

| | |
|---|---|
| Technique: | Run application and manually inspect the alignment of component on the each and every layouts.<br>Manually check activities of each and every clickable components.<br>Run the application on devices with deferent size of screens and resolutions.<br>Test UIs with uiautomator |
| Oracles: | Check the placement of the component on the interfaces. Check the appropriate access and navigation through the functions of the UI components.<br>Check the behavior of logouts on deferent size of screens. |
| Required Tools: | • Virtual devices with deferent size of screens.<br>• uiautomator |
| Success Criteria: | Activity layout with properly placed component with standard size. No changes of layouts with deferent size of android devices. |
| Special Considerations: | When add imagers to the application we have to add images with deferent resolutions (hdpi, ldpi, mdpi, xhpi, xxpi) otherwise they may not resize on deferent size of devices. |

### *3.1.4 Performance Profiling*

| | |
|---|---|
| Technique Objective: | The main objective of performance profiling is to verify performance of the "ReserveMe". This is the process of determining the speed or effectiveness of the application.<br>The testing should measure the response time for the load hotels onto map, Find location through the Google map, calculate payment for a reservation and calculate ratings etc.<br>Performance is an important consideration when develop an application for mobile devices, because most of mobile devices has a limited central processing unit (CPU) and graphical processing unit (GPU). And also mainly focused customers of "ReserveMe" are hotel owners in rural areas and most of the have devices with low performance. So compatibility of such a devices is more impotent to "ReserveMe" |
| Technique: | • Run application and check memory, GPU, network performance via android device monitor. |
| Oracles: | • Android device shows memory, GPU, network and thread usage for each action done for the application. |
| Required Tools: | • Android studio 1.4 – Android Device monitor.<br>• Android mobile device |
| Success Criteria: | When we carried out any action for application CPU, GPU, network activity going up in the live graph can be observed. |
| Special Considerations: | • Results of performance are depend on device. So it is batter to test performance with devices has low performances.<br>• Always need to use physical devices for performance testing. |

### *3.1.5 Load Testing*

| | |
|---|---|
| Technique Objective: | Performance of the system may be lag when the work loads and number of user increases. So objective of this testing is to test the workloads measure and evaluate the performance behaviors. The goal of load testing is to determine and ensure that the system functions properly beyond the expected maximum workload. Additionally, load testing evaluates the performance characteristics, such as response times and other time sensitive issues.<br>As example when registered hotel count is increases time take to load hotels onto the map also may be increases. |
| Technique: | Modify data file to increase the number of records and carrying out testing performance measures. Use android device monitor for check performance. |
| Oracles: | Android Studio device monitor the responsiveness of the application responsiveness tools on Phone device.<br>Human oracles include detecting the performance of the application by modifying the quantity of data contained in database. |
| Required Tools: | • Android studio 1.4 – Android Device monitor.<br>• Android mobile device |
| Success Criteria: | As we run the view hotel activity, CPU activity and the UI responsiveness going up in the live graph can be observed. This test can be carried out for a certain time period and several times increasing hotel count. |
| Special Considerations: | This testing should be carried on the Phone device not in the Emulator. |

### *3.1.6 Security and Access Control Testing*

| | |
|---|---|
| Technique Objective: | Objective of this testing is verify the security of this system. Use a user login system for provide security. In here we test the functionality of the login system. |
| Technique: | Use a user login system for provide security. Use a unit test for test functionality of the login system.<br>Run application in debug mode and check values of phone number and password fields while run the login activity. |
| Oracles: | Heuristic oracles can be achieved through unit testing and debugging. In hear test outcomes of methods using test cases with known outputs. This is achieved through junit testing method, "assertEqual (Expected,Actual)".<br><br>Check values of phone number and password fields from android studio debug monitor.<br><br>Human oracles include manual verification of results. It can be carried out by entering wrong password and try to login. |

| | |
|---|---|
| Required Tools: | • Unit test – junit:4.12<br>• Android studio 1.4 debug monitor<br>• Genymotion virtual devices |
| Success Criteria: | The success of manual testing can be determined if login will success with only when use the correct mobile numbers and passwords.<br>In unit testing assertEqual(Expected,Actual) method is determine actual outcomes are same as the expected values. And also this is detects the exceptions if there is an exception then test case will fail. |
| Special Considerations: | Since login system take login data from online server, internet connectivity is necessary fact to do this testing. |

### *3.1.7 Failover and Recovery Testing*

| | |
|---|---|
| Technique Objective: | Failover and recovery testing ensures that the target-of-test can successfully failover and recover from a variety of hardware, software or network malfunctions with undue loss of data or data integrity.<br>For ReserveMe that must be kept running failover testing ensures that, when a failover condition occurs, the alternate or backup systems properly "take over" for the failed system without any loss of data or transactions |
| Technique: | • Interruption when tracking location using GPS.<br>• Power interruption to the application.<br>• Interruption for internet connection when use the application. |
| Oracles: | • Testing fallers by adding incorrect data for inputs fields.<br>• Test the application function of add location of 'Register hotel' activity with GPS setting of the phone set to off.<br>• Run application with devices in low battery state and check what will happen when device shout down.<br>• Turn off internet connection and try to run application.<br>• Run application connecting WIFI connection. |
| Required Tools: | • Android device mobile phone.<br>• Android studio 1.4 |

| | |
|---|---|
| Success Criteria: | • If GPS setting is not set to on, a message should be displayed to the user requesting GPS to set on when try to track location via GPS.<br>• If mobile data is not set to on, a message should be displayed to the user requesting turn on mobile data or go to wifi settings.<br>• User should log out in power frailer.<br>• If invalid inputs are entered by the user, application should be displayed an error massage showing what kind of input that field can get. For an example user should not be able to enter latters in phone number text field. |
| Special Considerations: | For check power frailer situation application should run on actual device. |

### *3.1.8 Configuration Testing*

| | |
|---|---|
| Technique Objective: | Configuration testing is to verify the operation of the application on different software and hardware configurations |
| Technique: | • Test "ReserveMe" with deferent android mobile devices.<br>• Test application with deferent virtual devices. They have deferent API levels, resolutions and performance (Ex :- Custom Phone – 5.0.0 – API 21 – 768x1280, Samsung Galaxy S4 – 4.2.2 – API 17 – 1080x1920 )<br>• Test with the android emulators<br>• Configuration using gradle and the android plugin. |
| Oracles: | This application should tested with different virtual devices and android phones for proper functionalities. In each device test that the layouts application are properly placed with proper sizes on the device screen. |
| Required Tools: | • Deferent Android devices<br>• Genymotion virtual devices.<br>• Android studio 1.4 |
| Success Criteria: | If application is pass the configuration test then it can be release as an application which compatible for any kind of android device. The layout sized should keep standard sizes on deferent size of screens. |
| Special Considerations: | This test also can be done with the user feedbacks and can improve system based on those. |

## 4. Deliverables

- In this section, list the various artifacts that will be created by the test effort that are useful deliverables to the various stakeholders of the test effort.

### 4.1 Test Evaluation Summaries

Testing is carried out throughout the whole project development period. Not only that, after released the initial versions testing is happened with the user feedbacks. After adding new component or modification into the system it is necessary to carried out a testing process. It is important to ensure that the application works as intended and the new modifications had not caused any problems to the existing working application.
The testing of the application is carried out under several categories such as Data and Database Integrity Testing, Function Testing, User Interface Testing, Performance Profiling, Load Testing, Security and Access Control Testing, Failover and Recovery Testing and Configuration Testing.
These categories would comprise both manual and automated testing appropriately.

### 4.2 Reporting on Test Coverage

Test report would generally include the testing activities, results, bugs are found out and how they have been fixed out. Iteration wise tests are carried out and results are released as reports. Every iteration releases a report on testing. An iteration period very with the work done in particular period. And also this include testing technologies, test case use for each test cases and objectives of each kind of testing. And also need to add separate section for customer acceptance test part.

## 5. Risks, Dependencies, Assumptions, and Constraints

| Risk | Mitigation Strategy | Contingency (Risk is realized) |
|---|---|---|
| **GPS setting turn off.**<br><br>**Likelihood of occurrence:** High<br><br>**Impact if the risk is realized:** Low | To take current location by using GPS is a must in turn on. If relevant data is not returned through GPS, runtime exceptions can occur. The risk can be avoided by making aware the users that GPS setting of the device is not set to on when use the Google map. | Aware the user prior to the loading of the add location activity that GPS of the device is not set to ON, so that the services cannot be provided. To get the service, request the user to set it on. |
| **Internet connectivity is turn off.**<br><br>**Likelihood of occurrence:** High<br><br>**Impact if the risk is realized:** Low | For use this application internet connectivity is necessary fact. | Internet shod turn on always for use “ReserveMe” |