Software Test Plan Document

HelpMate Mobile App

Sirisha Veeraganta

TABLE OF CONTENTS

1.0 INTRODUCTION

This product is unique and independent by its features but there are many mobile apps in the market which work on the similar concept of interacting with user and handling the user's task more efficiently. Most of the other products are standalone independent software while HelpMate will try to integrate these applications into one product. HelpMate needs to communicate with other apps on the phone in order to retrieve information and utilize that to alert or notify the user. HelpMate largely utilizes the GPS on the phone to communicate and keep the track record of its locations.
Helpmate also connects to the call history and updates its internal database with number of times calls made to each number. This information is used to classify the contacts list and favorites by prompting to user to help generate a favorite list.

2.0 OBJECTIVES AND TASKS

2.1 Objectives

HelpMate largely utilizes the GPS on the phone to communicate and keep the track record of its locations. This information is fed to the database and is used for parking lot assist and GPS tracker.
Helpmate also connects to the call history and updates its internal database with number of times calls made to each number. This information is used to classify the contacts list and favorites by prompting to user to help generate a favorite list. At regular intervals of time, the user is suggested to get back in touch with that one such person, the user made calls more than a certain number of times. Thus HelpMate takes care of the social aspects of the user when the user is stuck in the daily chores of life.

2.2 Tasks

Major tasks involve testing the user interface functionality and applications interaction with GPS and database.

3.0 SCOPE

General

The specifics about the operations within the features is going to be tested and debugged. The features have very few user interactions but the back ground work is intense which needs to be fool proof. The features have partial authority to make decisions for the user and study the user based on which the suggestions and recommendations to the user will be made. Hence it is very crucial and important for the app to be bug free.

Tactics

The tactics for testing the user interface will be running and operating all possible manipulations on the application.

For Park Mate –
The feature has three buttons, “Pin location”, “navigate”, “delete”.
When “pin location is chosen, the app should show a blue dot on the map. This ensures that the location is pinned.
When navigate button is chosen, the app should be able to open the maps and show the directions to the pinned location
When delete option is chosen, the blue pinned dot should be deleted.

The testing aspect – when the delete button is chosen without a pinned blue dot, the button should do nothing.
The navigate button should not be able to open maps when there is no pinned location or blue dot.
Stay Connect Mate has a favorites list. The list is generated and updated from time to time with new data. The user can choose to delete a particular number, when long pressed on the number, the delete options pops up and user can click on it to delete the number. This functionality is to be tested.
GPS Mate also less very less manipulation with the user. The user interface is minimal here, the only interaction with the feature is when the user chooses to delete a particular location, he/she can long press the location listing, the delete button pops up and upon clicking that, the location will be deleted.

4.0 TESTING STRATEGY

For each major group of features or feature combinations, specify the approach which will ensure that these feature groups are adequately tested. Specify the major activities, techniques, and tools which are used to test the designated groups of features.
The approach should be described in sufficient detail to permit identification of the major testing tasks and estimation of the time required to do each one.

4.1 Unit Testing

Definition:
Since the mobile application have any designated hardware, it does not have any direct hardware interfaces. The physical GPS is managed by the GPS application in the mobile phone and the hardware connection to the database server is managed by the underlying operating system on the mobile phone and the web server.

Methodology:

This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware, software and communication interfaces and provides basic prototypes of the user interface.

4.2 System and Integration Testing

Definition:
The communication between the different parts of the system is important since they depend on each other. The different app on the mobile are configures to the HelpMate and thus it retrieves the data and stores the required data to its database.

Methodology:
The ParkMate page looks very simple. It basically contains 3 buttons. A “Pin location” button, “Find the pin” button and “delete pin” button.
When the user arrives at a location and wants ParkMate to remember that location, he/she can simply go to the ParkMate page and click on the “Pin Location” button. What this basically does is, it connects to the GPS and fetches the Latitude, longitude details and sends that to its database to store it. When the user wants to go back to that pinned location, he/she can click on the “Find the Pin” button and the app goes to the database, retrieves the saved data and puts it on the GPS and gives the path to the location to the user. This app is used mainly for parking assistance. Often people forget where their car is parked and look around wasting time in order to spot it. This app can be best utilized then, it’s very efficient, quick and time saving.

4.3 Performance and Stress Testing

Definition:
When the HM app is downloaded, SCM feature communicates with the call history and screens the phone numbers of last 10 days and picks the number that was either dialed or received more than 3 times. This number is possible prospect of being added to the favorites of the SCM app. Before which, the app confirms with the user if the number could be added to favorites. Upon the approval of the user, the number is added to favorites. The app basically collects such numbers from the call history by screenings the call history on daily basis. The app makes a list of favorites and at regular intervals of time (The use can customize the interval), the app suggests the user to call or message the number added in favorites. That way the app ensures to alert the user to call or message all those numbers added to its favorites list.
The user also has a choice to discard the suggestion by simply clicking cancel.
This app is best app of all because it is unique, it is a friendly app giving suggestions to the user to catch up with his/her old friends and stay in touch. In this busy world, such app comes very handy as it not only filters the call list and makes a favorites list but also ensures the user is notified to keep in touch with his /her close connections.

Methodology:
Describe how Performance & Stress testing will be conducted. Who will write the test scripts for the testing, what would be sequence of events of Performance & Stress Testing, and how will the testing activity take place?

4.4 User Acceptance Testing

Definition:
A first-time user of the mobile application should be able to download the app on to the mobile and run the app to install it. It also comes with manual for the first time user and the interface is very user –friendly and application is self-explanatory. It does not require any login username or Password. The first time user can open the app and maneuver the app to set the filters and criteria for the notifications

If the user is not a first-time user, he/she should be able to operate the software with ease. The user can open the app and maneuver the app to change the filters and criteria for the notifications if required.

Methodology:
Describe how the User Acceptance testing will be conducted. Who will write the test scripts for the testing, what would be sequence of events of User Acceptance Testing, and how will the testing activity take place?

4.5 Batch Testing

4.6 Automated Regression Testing

Definition:
Regression testing is the selective retesting of a system or component to verify that modifications have not caused unintended effects and that the system or component still works as specified in the requirements.

Methodology:
The mobile application communicates with the GPS application in order to get geographical information about where the user is located. The communication between the database and the application consists of operation concerning both reading and modifying the data, while the communication between the database and the mobile application consists of only reading operations.

4.7 Beta Testing:

Methodology:
When the user clicks on the “find the pin” button, the app goes to the database, retrieves the saved data from the database and invokes Google maps app and gives the path to the location to the user. If a pinned location does not exist then the app should do nothing.

5.0 HARDWARE REQUIREMENTS
Smart Phone

6.0 TEST SCHEDULE

Include test milestones identified in the Software Project Schedule as well as all item transmittal events.

Define any additional test milestones needed. Estimate the time required to do each testing task. Specify the schedule for each testing task and test milestone. For each testing resource (that is, facilities, tools, and staff), specify its periods of use.

7.0 CONTROL PROCEDURES

Problem Reporting
Document the procedures to follow when an incident is encountered during the testing process. If a standard form is going to be used, attach a blank copy as an "Appendix" to the Test Plan. In the event you are using an automated incident logging system, write those procedures in this section.

Change Requests
Document the process of modifications to the software. Identify who will sign off on the changes and what would be the criteria for including the changes to the current product. If the changes will affect existing programs, these modules need to be identified.

8.0 FEATURES TO BE TESTED

Identify all software features and combinations of software features that will be tested.

9.0 FEATURES NOT TO BE TESTED

In order to get a view of how to divide the requirements into different releases and what requirements should be included in which release, a prioritization of the requirements is needed. This section discusses the choice of prioritization methods and gives a suggestion of how the release plan for these requirements could look like.

10.0 RESOURCES/ROLES & RESPONSIBILITIES

Specify the staff members who are involved in the test project and what their roles are going to be (for example, Mary Brown (User) compile Test Cases for Acceptance Testing). Identify groups responsible for managing, designing, preparing, executing, and resolving the test activities as well as related issues. Also identify groups responsible for providing the test environment. These groups may include developers, testers, operations staff, testing services, etc.

11.0 SCHEDULES

Major Deliverables
Identify the deliverable documents. You can list the following documents: -
Test Plan
- Test Cases
- Test Incident Reports
- Test Summary Reports

12.0 DEPENDENCIES

HelpMate requires a smartphone with Wifi or (4/3) G connectivity and basic working knowledge of a smartphone such as being able to install the app on the phone and operating an app. The smart phone should be GPS enabled. The smart phone should have tactile screen for interacting with the application. The phone should also have enough hardware requirements such as memory to run and store the data.

The user is required to download the app and install it in the phone. While downloading the user has to ensure the installation is complete without any errors. Partial download will not run the software.
The features are completely independent of each other although some features completely depend on the GPS software on the phone without which they cannot operate.

13.0 RISKS/ASSUMPTIONS

HelpMate requires a smartphone with Wifi or (4/3) G connectivity and basic working knowledge of a smartphone such as being able to install the app on the phone and operating an app. The smart phone should be GPS enabled. The smart phone should have tactile screen for interacting with the application. The phone should also have enough hardware requirements such as memory to run and store the data.

The user is required to download the app and install it in the phone. While downloading the user has to ensure the installation is complete without any errors. Partial download will not run the software.
The features are completely independent of each other although some features completely depend on the GPS software on the phone without which they cannot operate.

14.0 TOOLS
Visual Studio 2013

Windows phone 8.1 SDK