Test Plan for Mobile Bird Application; v0.5

Create Date: 9/25/2013; By: Anna Barendt

Edit Date:<>; By<>

Table of Contents

[Objectives 2](#_Toc367961663)

[Purpose 2](#_Toc367961664)

[Scope 2](#_Toc367961665)

[References 3](#_Toc367961666)

[Requirements for Test 4](#_Toc367961667)

[Data and Database Integrity Testing 4](#_Toc367961668)

[Function Testing 4](#_Toc367961669)

[Graphical User Interface Testing 4](#_Toc367961670)

[Performance Testing 5](#_Toc367961671)

[Load Testing 5](#_Toc367961672)

[Stress Testing 5](#_Toc367961673)

[Volume Testing 5](#_Toc367961674)

[Installation Testing 5](#_Toc367961675)

[Recovery Testing 5](#_Toc367961676)

[Configuration Testing 5](#_Toc367961677)

[Security Testing 5](#_Toc367961678)

[Map Testing 5](#_Toc367961679)

[GPS Testing 6](#_Toc367961680)

[Camera/Image File Testing 6](#_Toc367961681)

[Voice Activation 6](#_Toc367961682)

[Audio File Testing 6](#_Toc367961683)

[Date Testing 6](#_Toc367961684)

[Test Strategy 6](#_Toc367961685)

[Testing Types 7](#_Toc367961686)

[Glossary 11](#_Toc367961687)

[Business Owner Proposal: Catalyst Mobile Bird App 12](#_Toc367961688)

[Milestones 14](#_Toc367961695)

[Deliverables 15](#_Toc367961696)

# Objectives

## Purpose

This document describes the plan for testing a prototype Android application that is being created for learning purposes, to give our development engineers the opportunity to work within the Android framework.

Please note: While the goals appear lofty, collection of data information, collection of GPS coordinates, etc., at this time there are no requirements to enforce even minimal data integrity in the database. As per the owner’s request, these requirements may or may not be installed at a later time.

## Scope

This Test Plan describes the tests that will be conducted on the prototype application.

This Test Plan assumes that JUnit testing is provided by the design engineers. The SDET will provide integration of the JUnit tests into an automation program.

The purpose of assembling this prototype is to test feasibility and performance of the selected architecture, devices, user interfaces.

1. The interfaces between the following subsystems will be tested:
   1. Icon
   2. Splash screen
   3. Menu page
   4. Data Input form
   5. Data Edit form
   6. Google Maps
2. The most critical performance measures to be tested are (This is a rough draft . . . ):
   1. Allowing data to be entered into the database
   2. Retrieving the correct data from the database
   3. Capturing the date/time when selecting a category.
   4. Being able to create a GIS object.
   5. Being able to turn on/turn off the GIS object.
   6. If the GIS object is being used, extract data from it and place it in the Input Form to be saved to the database.
   7. Create a camera object unique to the application.
   8. Allow the user to preset and save camera settings.
   9. Allow the user to change the camera settings “on the fly” because they want to.
   10. Allow the user to take a picture with the picture object and save the picture to a dedicated location.
   11. If GIS latitude/longitude data is available, encode the picture with the information.

## References

Template for Test Plan Creation

<http://sce.uhcl.edu/helm/RUP_course_example/courseregistrationproject/artifacts/test/plans/test_plan_arch.htm>

Wikipedia: <http://en.wikipedia.org/wiki/Graphical_user_interface_testing>

Verifying Code by Using Coded User Interface Tests (interesting aspects/not testing in ms<http://msdn.microsoft.com/en-us/library/dd286726.aspx>

# Requirements for Test

## Data and Database Integrity Testing

* Verify access to the device’s database.
* Verify that data can be entered.
* Verify that data can be retrieved.

## Function Testing

* Verify that the data in the input/edit form can be placed into the database.
* Verify that the user can see a list of their saved data.
* Verify that the user can select a data entity and edit it.
* Verify that the edit screen displays the entity data that the user wants to change.
* Verify a picture can be taken.
* Verify a picture can be stored.
* Verify a picture can be retrieved.
* Verify that latitude/longitude information can be ‘attached’ to a picture if the GPS function is turned on.
* Verify that latitude/longitude information can be auto-entered into the proper field on the input form if the GPS function is turned on.
* **Please note**, that phone software compatibility testing may affect the visual appearance and application functionality; more research needs to be done on this.

## Graphical User Interface Testing

* Verify ease of navigation through the screens/menus.
* Verify screens conform to GUI standards.
* Verify that input fields on the input/edit form can have data entered.
* Verify that the data in the input/edit form can be placed into the database.
* Verify that edited data can be retrieved.
* Verify Error Messages display correctly.
* Verify font size is readable.
* Verify text alignment is correct.
* Verify images have good clarity.
* Check GUI element positions in different screen resolutions.
* Check for clear demarcation of different sections of the screen.
* Check for aesthetically pleasing colors of fonts, background, etc.
* Check that the intended functionality can be executed through the GUI.

GUI Testing: <http://www.guru99.com/gui-testing.html>

## Performance Testing

* None.

## Load Testing

* None.

## Stress Testing

* None.

## Volume Testing

* None.

## Installation Testing

Verify that the application can install on:

1. Paul Wroe’s phone—Samsung Galaxy S3 running with Android 4.1
2. Anna Barendt’s phone—Samsung Galaxy S2 running Android 4.1.2; Touchwiz 4.1.2
3. Randy Miller’s phone—HTC DROID DNA running on Android 4.1 and HTC Sense 4.0 (Possible upgrade to Android 4.3 and HTC Sense 5.0 by the end of the project)
4. Mike Howell’s phone—Samsung Galaxy S3 (root access) running 4.2.2

**\*\*Please note,** at this time, we do not know the GUI overlays on all the above-mentioned phones, and subsequent upgrades may affect the overlay and/or back-end operating system, which could affect user experience and/or application functionality. This needs further research.

Gui overlays: <http://prowiki.isc.upenn.edu/wiki/Android_UI_Overlays>

## Recovery Testing

* None.

## Configuration Testing

* None.

## Security Testing

* None.

## Map Testing

* Verify that Google Maps can be accessed/imported/used by the application.
* Verify that vector point data can be marked on the Google Map.
* Verify that the marker for the vector point data is application specific, i.e., specific for Bird Activity, Nesting Area, and Misc.
* Verify that the icon size increases/decreases as the user scrolls in/out on their device.

## GPS Testing

* Verify that the GPS object can be used by the application.
* Verify that the latitude and longitude coordinates can be extracted from the GPS object and inserted into: the input form

## Camera/Image File Testing

* Verify that a camera object can be created.
* Verify that the user is prompted to set default camera settings of: white balance, f-stop, zoom, resolution, <add the rest here>
* Verify that the user-entered camera presets are saved.
* Verify that upon opening the application, the user can take a picture that uses the predefined camera settings.

## Voice Activation

* Verify that the application will accept voice recognition capabilities (this may be a function of the device, not the application).

## Audio File Testing

* Verify that a voice file can be created.
* Verify that the voice file can be stored.
* Verify that voice file can be accessed at a later date.

## Date Testing

* Verify that the date/time shows up within/on the application interface.
* Verify that date/time is entered into the database when the person

# Test Strategy

This Test Strategy presents the approach to the testing of this software on the above-mentioned devices. The previous section on Test Requirements described what will be tested; this describes how it will be tested. As a learning objective, this section may change over time.

The main considerations for the test strategy are the techniques to be used and the criterion for knowing when the testing is complete.

In addition to the considerations provided for each test below, testing should only be executed using known, controlled databases, in secure environments.

## Testing Types

|  |  |
| --- | --- |
| Data and Database Integrity Testing The database is an integral part of the Android system. Additional research into SQLLite needs to be performed to see if the database can be accessed without the user interface, otherwise, testing will be through the user interface. | |
| Test Objective: | Data input and extraction can be completed without corruption of the data. |
| Techniques: | Test each call to the database and verify that the correct data is saved and retrieved. |
| Completion Criteria: | * Test the initial input of text data, the retrieval and updating of the text date. * Test that the icons used in mapping are correct. * Test that the database is storing the correct auto-generated map points and placing them in the correct map location * Test that the user-created map points are being placed correctly on the map—Note: there is no inherent qualifying of the latitude and longitude points, may come up with incorrect hemisphere location * Test SQL injection into one/all of the data fields upon initial input/editing of data. |
| Special Considerations: | Using SQLLite, which is bundled in Android phones. |

|  |  |
| --- | --- |
| Function Testing Function Testing of the application focuses on any target requirements that can be traced directly to use cases/business rules. Testing goals are to verify proper data acceptance, processing, and retrieval, and the appropriate implementation of the business rules. Testing is based upon black-box techniques, i.e., verifying proper functioning of the application and its internal processes by using the GUI to interact with the application and analyzing the output/results. Attempts will be made to automate as much of this testing as possible. Further research/testing environment creation needs to be completed before automation of tests is possible. | |
| Test Objective: | Ensure that application navigation, data entry, processing, and retrieval function properly. |
| Techniques: | With valid or invalid data, execute each use case, use case flow, or function to verify:   * The expected results occur with valid data * Appropriate error messages display when invalid data is entered |
| Completion Criteria: | * Planned tests are executed and documented * Identified defects have been corrected |
| Special Considerations: | * None |

|  |  |
| --- | --- |
| Graphical User Interface Testing GUI testing verifies the user’s interaction with the software with the goal of ensuring that the GUI provides the user with the needed navigation and access through the functions of the application. Additionally, this testing ensures that the objects within the UI behave as expected and conform to business owner standards. Research needs to be completed for automation of GUI testing. | |
| Test Objective: | Verify the following:   * Navigation through the application reflects business owner requirements, including window-to-window, field-to-field, and access methods of dropdown lists, menus, information boxes, swipe motions. * Screen objects and characteristics such as menus, position, and size conform to business owner and/or industry standards. |
| Techniques: | * Create/modify tests for each screen to verify proper item placement and navigation. |
| Completion Criteria: | * Each screen successfully verified to stay consistent within acceptable standards. |
| Special Considerations: |  |

|  |  |
| --- | --- |
| Installation Testing Installation testing verifies that the application can run on different devices. At this point all the installations are on Android devices/emulators. | |
| Test Objective: | Validate that the application can be installed on the previously listed devices and emulator setting. |
| Techniques: | Attempt to install and run the application on the various devices. |
| Completion Criteria: | The application can install and run correctly on the device. |
| Special Considerations: |  |

|  |  |
| --- | --- |
| Map Testing Map testing verifies the viewing of Google Maps within the application. Research needs to be done to see if this testing can be automated for regression testing. | |
| Test Objective: | Test functionality of Google Maps within the application. |
| Techniques: | Functionality will be tested by:   * Selecting to view the map * The map has an indicator showing user’s position within the map |
| Completion Criteria: | Mapping application passes above tests and continues to function in a similar manner in consequent versions of the application. |
| Special Considerations: | Users can edit latitude and longitude coordinates, and may not know how to properly enter them, which may lead to inaccurate mapping. |

|  |  |
| --- | --- |
| GPS Testing GPS Testing verifies that the latitude and longitude coordinates are captured from the GPS object in a usable format. Additionally this testing verifies the ability to create and insert a vector data point on the Google Map view within the application environment and do so with some measure of accuracy. Research still needs to be completed for test automation. | |
| Test Objective: | Ability to extract latitude/longitude data from the GPS object to store the latitude/longitude data in the database and attach the information to other objects within the application. |
| Techniques: | Functionality will be tested by:   * Selecting to view the map * Selecting to insert data points on the map, using latitude and longitude points from the database * Verifying that application icons are associated with map points * Re-sizing the map to see if the points automatically resize * Verifying that the latitude and longitude can be auto-placed in the text fields of the original data input form and in the data edit form * Verify that the data is returned accurately * Verify that the latitude/longitude data can be attached to a picture properly. * Verify that the user has the ability to turn the GPS function on or off.   Enter the data point latitude and longitude into Google Earth and visually validating that the points are in a similar area of the earth. |
| Completion Criteria: | Users are able to edit latitude and longitude coordinates, and may not know how to properly enter them, which may lead to inaccurate mapping. At this time, |
| Special Considerations: | Unknown how this functionality may be affected based on the individual phone/emulator. |

|  |  |
| --- | --- |
| Camera Testing The camera is an integral function of the application. Additional research needs to be performed to see if testing can be automated. | |
| Test Objective: | User can manipulate a camera within the application. |
| Techniques: | Functionality will be tested by   * Verifying that the user can preset the camera settings of : * Verifying that the settings are saved upon shutting the application. * Verify that the settings are automatically applied when the camera is opened within the application and not applied when the camera is opened outside of the application. * Verify that the user can easily change the camera settings after the initial preset * Verify that there is a unique location to store the pictures * Verify that pictures can be retrieved for viewing * Verify that one location can have many pictures associated with it * Verify that one picture can be associated with many locations * Verify that a picture can be deleted. * Verify that the database reflects the deletion of an image. |
| Completion Criteria: | Users are able to manipulate the camera object to meet their needs. |
| Special Considerations: | Unknown how this functionality may be affected based on the individual phone/emulator. |

# Glossary

Capture Event, Basic: When the user selects a category.

Entity: Database term; person, place, thing, or concept about which data can be collected. For the purpose of this application the entity is loosely described as a bird.

GUI: Graphical User Interface

HTC Sense: HTC user interface overlay

Implementation: how the application does what it does, which is not visible by the user

Interface: how something interacts with the world that specifies what will be accepted as input, and what output will be produced; the various screens/views of the application that the user interacts with

TouchWiz: Samsung user interface overlay

Unit Testing: To test the components of a program one by one

# Business Owner Proposal: Catalyst Mobile Bird App

For Android, iOS, and Windows Phone

## A Bench Project Proposal By Paul Wroe

8/26/2013

## Purpose

The Catalyst Mobile Bird App is a bench project that can provide some focus for skills in the area of field data collection. Users should be able to go to parks and other nature areas to collect relevant data on birds for their personal enjoyment. They should also have the ability to share their data with others. Additionally, users should be able to extract their data from their mobile device and generate reports.

## Product Goal

The product goal is to produce a useable, friendly mobile application that has desirable features that showcase Catalysts' abilities in the mobile market.

## Training Goal

The training goal is to develop mastery of the various mobile platforms by way of feature implementation in an agile environment.

## Product Features

* Camera access - user can pre-configure camera settings and store the settings locally so that when the camera is engaged, it will be ready to go with minimal hassle.
* Geo-tagged pictures - pictures taken will be encoded with their GPS location.
* Localization - program is written so that other languages can be integrated and then selected.
* Data Storage - collected data will be stored locally in a lightweight SQL database.
* Data Export - data can be exported from the SQL database to either a CSV or PDF.
* KML/KMZ export - spatial data can be exported in KML or KMZ format so that it can be viewed in Google Earth.
* Voice notes - user can record audio notes and have them automatically associated with a location.
* Voice Digitization - Voice notes are transcribed into text or may be entered manually.
* 3rd party web services - Application connects to online bird database.
* Social Media - Bird observations, after being entered, can be uploaded to Facebook or other social media with little effort or configuration.

## Discussion of the Project

The Catalyst Bird App can be a showcase for our skills in the area of field data collection. It is possible that gaining skills in mobile data collection could open up opportunities in governmentally run natural resource departments.

Microsoft dropped support for [Windows Mobile](http://en.wikipedia.org/wiki/Windows_Mobile) and created a gap for operating systems and software related to field data collection. ESRI, a front runner in Geospatial Information Systems, has finally come out with an Android version of their mobile mapping software but many other applications are still frozen in the now deprecated Windows Mobile platform.

Although we would not be creating an application as full featured as ArcPad with regards to GIS data collection, we could make an application that could be used to demonstrate the companies’ prowess. This could create some opportunities making task specific software.

# Milestones

# Deliverables

Create Date: 9/25/2013; By: Anna Barendt

Edit Date: 9/26/2013; By Anna Barendt

The development team will deliver to the business owner by the end of the year 2013 a mobile application that will be able to:

1. Capture, store, recall and delete pictures with custom camera configuration
2. Store pictures with GPS information encoded in the picture
3. Capture, store and recall user text data via manual input
4. Capture, store, and recall user text data via voice input
5. Capture, store, recall and delete audio samples
6. Allow for data records to be edited
7. Allow for data records to be deleted
8. Capture, store, and recall GPS locations
9. Plot GPS locations on a map
10. Have customized icons for data points based on record category
11. Have information windows associated with data points that displays both picture and data.
12. Allow for center of mass of displayed data points to be displayed as a unique icon
13. Allow for selection of data points to be displayed on map
14. Have a cool application icon
15. Have great splash screen
16. Generate KMZ files from database and associated pictures
17. Generate a PDF from database and associated pictures
18. Upload single records to Facebook
19. Export data from the database as a CSV
20. Allow for data to be pulled from eBird website.

The application will be tested for defects using the appropriate technologies:

1. MonkeyTalk
2. Calabash/Cucumber
3. RoboElectric Test Runner
4. Program Emulators
5. JUnit

The application will be able to run on these four devices, which are Android devices:

1. Paul Wroe’s phone—Samsung Galaxy S3 running with Android 4.1
2. Anna Barendt’s phone—Samsung Galaxy S2 running Android 4.1.2; TouchWiz 4.1.2
3. Randy Miller’s phone—HTC DROID DNA running on Android 4.1 and HTC Sense 4.0 (Possible upgrade to Android 4.3 and HTC Sense 5.0 by the end of the project)
4. Mike Howell’s phone—Samsung Galaxy S3 (root access) running 4.2.2