PRIME 2013

Japanese Friendship Garden Haiku Hunt

Jesus Rios

Dr Jason Haga, Dr. Shinji Shimojo

Abstract:

Android smart-phone application that uses a new location based technology, PlaceSticker, to serve as an interactive guide for the Japanese Friendship Garden in Balboa Park. PlaceSticker, provided by iSiD, is a novel location-based technology. The project was designed to give the user a different kind of experience in the garden. The user can learn more information about the collections in the garden. The Android application uses clues to navigate the user through a scavenger hunt across the garden. The clues are consisted of a haiku, contributed by the community, and a related drawing. The user can swipe between the ten riddles and view a map of the garden as well. The JFG Haiku Hunt application was designed with certain limitations of the garden. Wifi signal is not available in all areas of the garden which presents problems with an online based app. PlaceSticker creates its own Wifi signal, which is used by the application to detect the different items in the garden. The app associates the ten riddles with different PlaceStickers placed across the garden. The application detects the Wifi signal through the PlaceSticker API and its unique id of is associated with the appropriate riddle and displays information. This project will also be using a new solar powered prototype, which will be the first to be utilized in an outdoor environment. The solar PlaceSticker will be self powered which will replace the need to change batteries constantly. The app hopes to give a visitor in the garden different experience and a different way for them to enjoy the garden. The user can unlock detailed information about plant collections (e.g. fun facts) as they discover new locations and unlock a coupon if all locations are found. JFG Haiku Hunt hopes to give a visitor a deeper understanding of the collections in the garden as well as involve the user in a fun activity. [UCSD, Jason Haga, BPOC, Vivian Kung-Haga, NICT mentor, Shinji Shimojo]

Introduction:

The Japanese Friendship Garden (JFG) Haiku Hunt is a Android mobile application that uses a novel location based technology, PlaceSticker, to serve as an interactive guide for the Japanese Friendship Garden in Balboa Park. The app contains information about plant collection that users can unlock by exploring the garden. The app consists of 10 crowdsourced haikus along with a corresponding artwork that give clues to users to discover different collections in the garden.

Objective:

The objective is to give users a different kind of experience in the garden. Users can explore the garden while solving these haikus and learn more about the collections in the garden.

Materials/Methods:

My main method of development was done using the Eclipse IDE. The Android environment was set up alongside Eclipse using the Android Virtual Manager. I used Git as my version control for all of my code. To test the app, I mainly used a Samsung Galaxy S III for development. I tested the performance and user interface using mostly this phone.

The Android environment allows for users to emulate various models and environments. This allowed me to test the application on different phones and examine how the UI looks on different screen sizes. The emulator also works for testing the app functionality across different versions of Android. JFG Haiku Hunt was intended to reach as much as the public so it can be used across all Android phone devices which was the reason I used the ActionBarSherlock library. The ActionBarSherlock library allows older versions of Android that do not natively support the Android ActionBar to display it along with all of its functionality.

The backend of JFG Haiku Hunt is built around a Java class I created named Riddle.java.The

Riddle class is used alongside RiddleLab.java which stores a list of the Riddle objects. RiddleLab.java is a special kind of class which allows only one instance of itself to be created. This allows it to serve as a backing store for all of the Riddles and will exist as long as the application stays in memory.

The home screen of the app uses a modified version of the Android GridView class I created which allows GridView to display images and expand across the screen. The user can scroll through the grid of images/clues and tap on any of them to view the haiku and sound clue (if one exists).

All of the information the Riddle objects store was provided by Japanese Friendship Garden and Balboa Park.This includes the haikus, artwork, and photographs that are all stored in the application. The RiddleLab.java class is the main way the app can keep track of what the user has solved and what to display to the user.

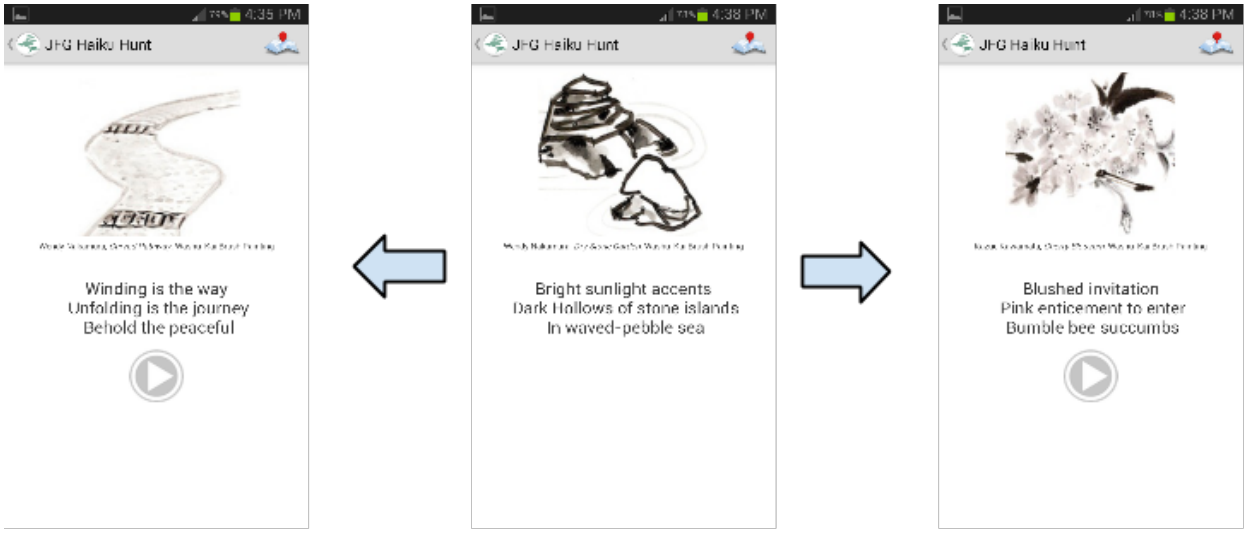
PlaceSticker is the main interactivity that a user has with the app. A PlaceSticker creates a unique WiFi signal that transmits to Android phones using the PlaceSticker API that was provided by iSiD. This WiFi signal is triggered once the user is within a close enough distance of the PlaceSticker.

The PlaceSticker transmits a unique ID which is tied to one of the ten Riddle objects. Once the PlaceSticker triggers the app it updates the view accordingly. Each Riddle object is tied to one of the PlaceSticker id. The API allows the program to continuously check for updates on the network.

One of the first issues I had with coding was Fragmentation. Android has two ways of creating an application -- using Activity and Fragment class. Activity is a method of creating a fast and simple app. However, Activities do have their limitations as only one item can be on the screen at a time. An Activity is still needed in order to host a Fragment. The Fragment is associated with a base Activity, but it also has its own lifecycle. This allows the Activity to display multiple Fragments and even host more than one Fragment.

Fragmentation allows the user to have multiple views on the screen. Some downsides of this is that Fragments require you to implement more methods than a regular Activity. Fragmentation though is that it is more difficult to update the current view of the screen. Using Activities, one can update the screen with a simple call. Since the Activity instance is still on the screen, the program has access to all of its elements.

I was able to allow the users to swipe left or right to change in between different screens of the app by using Fragments. The screen the app displays depends on what the user seen. For example here is a screen that shows what a user sees before he solves any riddles:



Fragmentation turned out to be an issue since I had initially constructed the application using only Activities. Converting the code to work with Fragments was a minor problem. The good thing is that there are no compatibility differences with using Fragments. As long as you use the library that is provided by Android, android.support.v4. I did not encounter any problems running the application on older versions of Android.

One limitation that I came across with Fragments was updating view dynamically. The way I

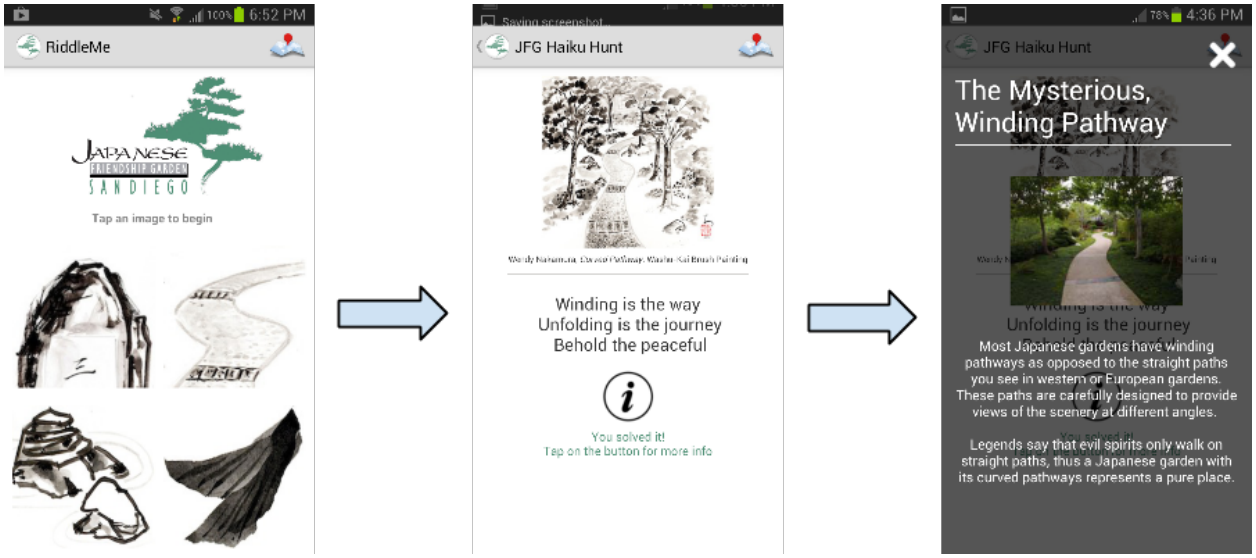
resolved this case was by reconstructing the View that holds the Fragment. Instead of modifying the current elements that are on the screen, I was able to destroy the current View and reconstruct a new one to show the correct elements. Because Android handles this operation smoothly, there are no visible delays.

Conclusion:

In conclusion, the PlaceSticker devices were integrated successfully with the application. The application was completed and there are no known issues as of now. The PlaceSticker functionality is still available outdoors with no issues as long as the user is within range the PlaceSticker covers

Results:

The app is in the process of being deployed in the Japanese Friendship Garden. The final view a user may see once they have solved a riddle is shown below:



These screens shows the information button that display information of the collections once a riddle has been found by approaching the PlaceSticker.

Acknowledgements

University of California, San Diego (UCSD):
• Dr. Gabriele Wienhausen
• Dr. Jason Haga
• Dr. Peter Arzberger
• Teri Simas

National Institute of Information and Technology (NICT):
• Dr. Shinji Shimojo
• Takata Tomoaki

Balboa Park Online Collaborative (BPOC):
• Vivian Haga
• Christina DePaolo
• Wesley Hsu

Japanese Friendship Garden:
• Marisa Espinosa
• Marisa Takeuchi
• Luanne Kanzawa

Information Services International Densu Ltd. (iSiD)

PRIME alumna Haley Hunter-Zinck

National Science Foundation (NSF)