Final Project Documentation
CS 4720 - F17 - ChinYeung

Final Project Proposal

Device: iPhone 8 Platform: iOS

Name: Christopher Yeung Computing ID: cy4bv

Name: Barry Chin Computing ID: bxc2gn

App Name: GarageSale

Project Description:

Our app, GarageSale, will allow users to post items that are for sale in the area, and based off the location of that user, this will post that item for sale at that specific location. We will have a map of the immediate surrounding area which can allow users to visualize where exactly that item is located. This is especially relevant for better helping users see what items are closer to them, and can help them decide whether they can actually move an item, for example, especially for more cumbersome items like couches/beds. Users will be able to message other users if there is interest for an item within the app.

What we propose to do is create an app that will do the following:

1. The system shall allow a user to create a profile where they can enter basic contact information, and can also keep a record of any sales made by that user.
2. The system shall allow users to view a list of available items for sale, as well as a map to better visualize the location at which these items are being sold, in order for the buyer to find the item they want.
3. The system shall allow users to interact with each other via messaging/calling, allowing the seller and buyer to decide privately whether an agreement can be made for an item. 4. The system shall generate a dynamic map of the surrounding area, where items for sale will appear on this map. This will better help the user visualize what items are available for sale nearby.
5. The system shall allow users to take pictures of an item that is for sale, to allow buyers

to better understand what they are buying.

6. The system shall required items to be categorized and have descriptions for each to help users with browsing.

We will plan to incorporate the following features. We have revised our proposal to reflect the optional features we ended up incorporating (as having the ones we decided not to include seems unnecessary) :

● GPS/Location Awareness (15 points) - The user who places a sale of an item will have this item posted/placed on the map based on the GPS location where the post was made.

● Camera (15 points) - The user will be able to post pictures for an item they have for sale. This image will also show up on the map showing where that item is for sale to give buyers a better idea of what they are getting.

● Build and consume your own web service using a third-party platform (15 points)- We will use Firebase as our primary database storage for our app. This will store items that are on sale by users.

● Open Shared Activity/ Features (5 points)- The user will be able to interact with other users, by sending them messages or calling them. The app will help facilitate this sharing between users.

● Consume a pre-built web service (10 points)- We plan to use some sort of location-based API to create a better user experience for our item map feature.

Wireframe Explanation

The wireframe we have attached, though being a little bit different from some of the developmental choices we made for our app, still mostly reflects the general user flow of our application. The user first lands on a sign-in/sign-up screen where they can create their account or log in to our app. From there, they see an item map with a list. We ultimately decided against including the two features, map and list, on the same tab, as we thought this would be a bit overwhelming for the user, but the ideas of both remain the same. As you can see on the map, markers of nearby items are placed around the user's current location, and the user can click on those items and receive more information about them as desired. The item list shows a segmented control panel, which allows the user to filter based on category of item that they want to see, with the most relevant item details, such as name, picture and price, appearing for the user as they are browsing. We also kept most of the other features such as the user's profile tab, and editing and deleting item capabilities. Overall, our wireframe still gives a pretty good idea of what the user flow through our app would look like and what a typical user would expect on our app.

Final Project Documentation

Platform Justification - What are the benefits to the platform you chose?

We ultimately decided to build an iOS app using Swift and xCode due to a number of reasons. One of them was our enjoyment of using xCode and Swift when building our iOS mini app. Both of us found the Storyboard a much more intuitive way to understand the views of our app and how they all relate, and we also found that making layouts on xCode was a far more enjoyable process when compared to Android Studio and its XML layouts. Another major reason that played a factor is convenience; both of us had apple computers so making the switch to building an iOS app made sense for us. However, the biggest benefits were definitely how much easier and comfortable it was for us to make layouts. Ultimately, there are a lot of similarities programmatically between iOS and android, so it came down to some of the smaller details.

Major Features/Screens - Include short descriptions of each (at least 3 of these)

Our app can really be summarized into having 5 major features:

1. User login/signup/logout features
    a. We tried to make the login, signup, and logout process for the user as smooth as possible, and did a lot of testing in ensuring that logging in, signing up, and logging out all worked as intended, in terms of providing relevant error messages and also in terms of authenticating users properly.
2. Item Map Screen
    a. The Item map screen has a lot of features within that should be noted. First is that the user is first brought to their current location, where they can view the items that are on sale around them. If they want, they can also move to a different location, and again, to make this simpler for the user, we added autocomplete (via google's places API) so the user doesn't have to type in a full address to search for it. Different items are placed with markers depending on category (question mark for "Other", clothes sign for "clothes", couch sign for "furniture", and books sign for "school"). Annotations are also available providing more item details, when the item is clicked on.
3. Item List Screen
    a. The item list screen shows all the items that are available to the user in a clean list format, with the item name, price, and image showing up first in each cell. When clicking on a particular item, the user is brought to an item information view which shows all the item details they need, along with the ability to contact the seller directly, via a variety of different methods (e.g. text message or call). The user can also filter using the segmented control the category of item that they want to scroll through.
4. Sell Item Screen/s
    a. The Sell item process is broken into chunks since we thought it would have been overwhelming if the user were to have to go through the whole process at once. First, the user enters basic item details (such as price, name, etc.), second, they

are told to attach an image for their item, and last, a location. After clicking on post, the item will be added to the list and map accordingly.

5. My Profile Screen
    a. The user can manage all their information here, and save the changes they make and log out as well. Another important feature here is that the user can also see the items that they have posted. **When swiping these items (right to left), they are prompted with edit and delete options.** When clicking on edit, the user can edit their item and save the changes they make to each individual item from the edit item view. When deleting the item, the item is removed from the item list and the user's own list.

Optional Features - Include specific directions on how to test/demo each feature and declare the exact set that adds up to ~60 pts

15 pts - GPS / Location-awareness (includes using Google or Apple Maps)

- The item map screen displays the user's current location, and uses this to show items (if any) that are on sale that are near them. The app also allows users to search a desired location and see items in those areas as well. Annotations are placed to mark out where each item is relative to the user so the user can better visualize how far these items are from them.

15 pts - Camera

- The app allows users to either take a picture or take a picture from their camera roll to display for their item. This image is used in the item list and item information pages to show the user what that item looks like. The images are saved into Firebase Storage upon creation, and loaded to the item list and information pages as needed using the image's download URL which are saved in Firebase Database for each item.

15 pts - Build and consume your own web service using a third-party platform (i.e Firebase)

- We used firebase extensively for this app. We first used firebase to handle authentication for our app. Login, signup, and logout all handle users for our app. We also used the firebase real time database for our items and users. This information was retrieved for the My Profile, Item Map, and Item List tabs. When a user added an item to sell, this was added to the maps and lists accordingly. Similarly, when an item was deleted or edited by a user, all these changes were tracked in firebase database. We also use firebase storage to handle how we stored images, and this was used to save/load the images for our items.

10 pts - Consume a pre-built web service

- Google Maps offers a very powerful Places API, a ready-made web service which allowed us to give the user the ability to autocomplete their searches on the Item Map tab. This is especially useful as we know most users would not want to type in a whole address, and doing so also leads to a greater chance for a typing error and also greater frustration for the user. The autocomplete API we utilized allowed us to suggest addresses based on what the user entered,

which we found to be a very powerful addition to our app.

5 pts - Open shared activity / features (i.e. Create an email to send, share with a text message, etc.)

- In order to contact the seller, when an item is clicked on from the item list page, the user has a couple options, among them the ability to message the seller of their interest via text message or call, along with email. The seller's preferred contact method is listed so the user knows the best way to reach him/her. All three of these open shared activity/features between the buyer and seller.

Testing Methodologies - What did you do to test the app?

To test the app, we did a number of things. The first thing is that we put print statements throughout our code, validating the data that we expected to receive in order to ensure our app was doing what we intended for it. Another crucial part of our testing was anticipating any potential errors the user would potentially run into (e.g. entering passwords that do not match during signup). We wanted to test our app to anticipate all these different scenarios and respond to them accordingly. Another important part of our testing, especially since Firebase was such a large component of our app, was ensuring that everything in the Firebase database was being updated accordingly. This meant that we would constantly have the Firebase console open as we made changes, making sure that when changes are being made, these are being reflected in firebase as expected. Lastly, we went through what we viewed as typical user experiences of our app. This went from logging in to signing up to forgetting passwords, to entering the app, viewing items, buying items, and being able to manage their own settings/items in profile.

Usage - Include any special info we need to run the app (username/passwords, etc.)

- **As discussed in the proposal, we have designed our app for iPhone 8 on iOS.** For best user experience and design, please use the iPhone 8 simulator or an actual iPhone 8.
- the formal xCode project for our application is within the directory "FinalProjectiOS". From the git clone of this project, please cd into that directory in order to run the application.
- Depending on the size of the list, the initial loading of images from firebase may take a bit of time (possibly, at worst, a couple minutes, depending on how many images there are). This is expected. Since we are caching the images, loading images after this initial load, however, is very quick.
- In order to start using the app, please sign up. We have also provided a demo account with the following sign in credentials in case you wanted to test the app

quickly and forgo the signup process.
    - Username: cy4bv@virginia.edu
    - Password: password
- Camera does not work well in simulator, as well as location since the simulator is defaulted in San Francisco, whereas most items we post are in the UVA area. A good starting point to see items is to move to Jefferson Park Avenue, Charlottesville, VA, since the majority of our test items were placed around there.

Lessons Learned - What did you learn about mobile development through this process?

- Wireframing is an extremely good use of time when designing an app. One thing that is unique about mobile app wireframing is better visualizing the screen by screen user flow. When mapping this out in the storyboard when developing the app, this kind of wireframing is extremely valuable.
- Understand the design limitations of your platform. For example, there were times when we spent hours figuring out why a navigation or tab bar item disappeared, and this was due to not designing our flow based on common design principles in iOS (for example, having the initial controller for each tab bar item be a navigation controller, and only then, will the view controllers following have the desired tab bar item). Small things like this make a big difference to the user experience and can take a lot of time to figure out if one does not understand some of these design constraints.
- Screen square footage matters significantly more when it comes to mobile development. With the screen being much smaller, design decisions carry greater weight, since each decision takes up more space. This is actually one of the reasons we decided to have a step-by-step sell item process instead of having it all on one view.