CSE 118 Mobile Applications Final Report

Yza Velasco
My Pham
Sean Jossy

ABSTRACT
SwapIt is a mobile application whose purpose is to allow nearby users to trade with one another without monetary transactions. In this report, our objectives, app components, and app development process will be discussed. Screenshots of certain activities will be provided along with a brief description for better visualization.

OBJECTIVE
SwapIt removes the hassle of having to use money to obtain used items. Most apps require monetary transactions. Instead, our mobile application allows registered users to trade items of similar value with one another.

One similar application that already exists is “Letgo”, an app for buying and selling locally. Another similar app is “Offer Up”, which has a similar idea as “Letgo”. There are also buying and selling websites such as Facebook Marketplace and Craigslist. Our app, Swap It, is similar to the apps mentioned, but it incorporates features from most, if not all of these platforms.

COMPONENTS
Our mobile application *SwapIt*, provides functionalities such as user login, user signup, homepage viewing, individual user posting, profile viewing and editing, in-app messaging, and external sharing. Upon initial use, the user will be directed to the main activity, containing the log in and sign up options. *Figure 1* below shows the initial main screen.

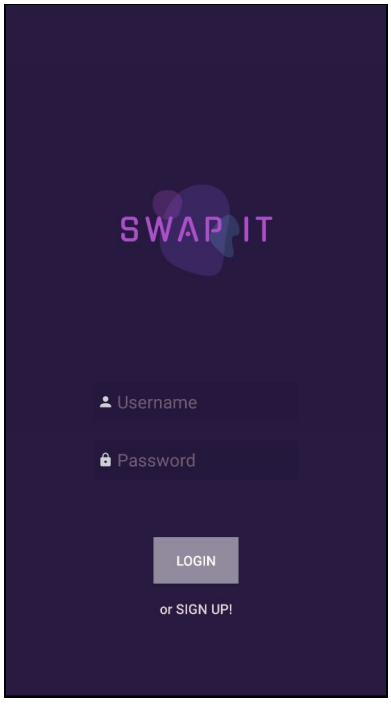

Figure 1: Login/Signup screen with login and signup functionality.

After logging in, the user’s login information is saved so they remain logged in and do not have to login every time they open the application. When logged in, the user is immediately directed to the homepage, where they can view the various items posted by other users within the current user’s zip code. Each post contains an item name, description, share button, and message button. *Figure 2* below demonstrates a view of the homepage. The message and share activity can be accessed via the homepage.

Figure 2: Homepage with view of sellers' posts containing item name, description, share, and message button.

From the homepage, the sellers' profiles can be accessed by clicking the name at the top of every post. The seller's profile consists of the seller's name and profile picture, an overview of their posts, and a message button that also directs the user to the message activity.

At the bottom of the homepage is the bottom navigation bar. This aspect remains constant through four activities; Homepage, Add Post, User Profile, and Message.

Upon clicking the Messages button on the bottom navigation bar, users are directly a collection of all their messages. Each component of the list displays a preview of the recipient's profile photo, name, and the most recent messages between the users demonstrated in *Figure 3*. Within the message activity, users and sellers are able to communicate directly and privately. The activity features an online status (online, offline, or last seen), message delivery status, and read status. *Figure 4* demonstrates a message thread between a user and seller. The top of the view displays the name and online status of the recipient.

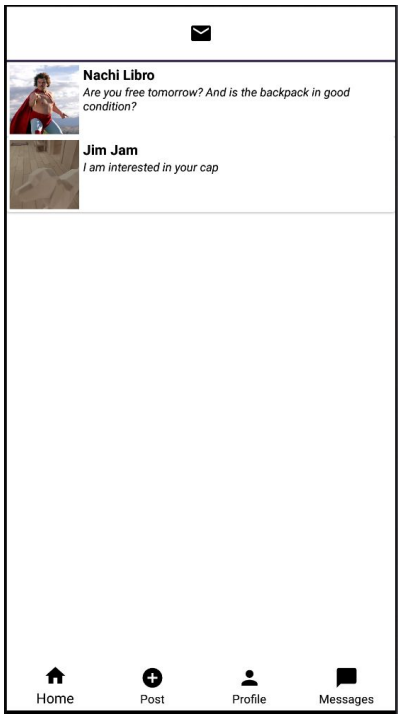

Figure 3: A list view of the user's messages.

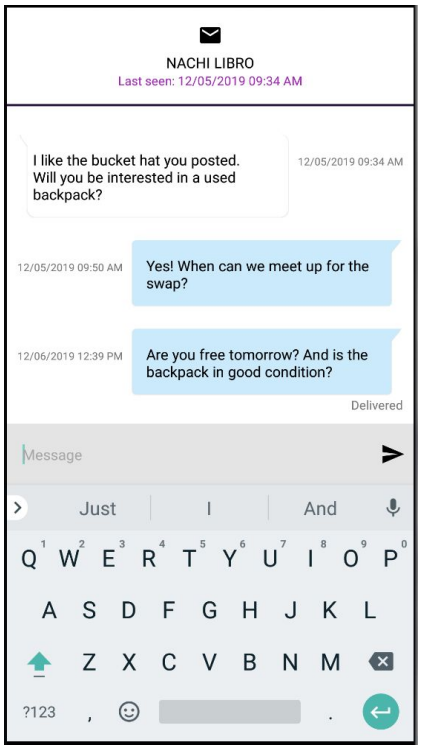

Figure 4: Individual user and seller message.

The share post activity shows the "Share via" pop up screen, which allows the user to share posts through text, email, google drive, or etc. *Figure 5* below shows a view of a shared post via text containing an image of the

shared item and a default message include the item and app name.

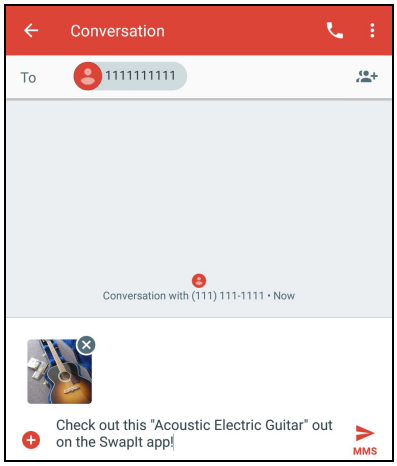

Figure 5: Message with an image of the shared item and a default message.

Clicking on the user profile button on the bottom navigation bar will direct the user to their own profile. From there, the user is able to see their name, profile photo, and posts displayed. Additionally, the user can also see their personal email address and zip code, which can only be seen by them. The floating action bar allows the user to edit their profile photo and display name. The logout button allows the user to logout and be redirected to the homescreen. *Figure 6* shows a view of a user's profile.

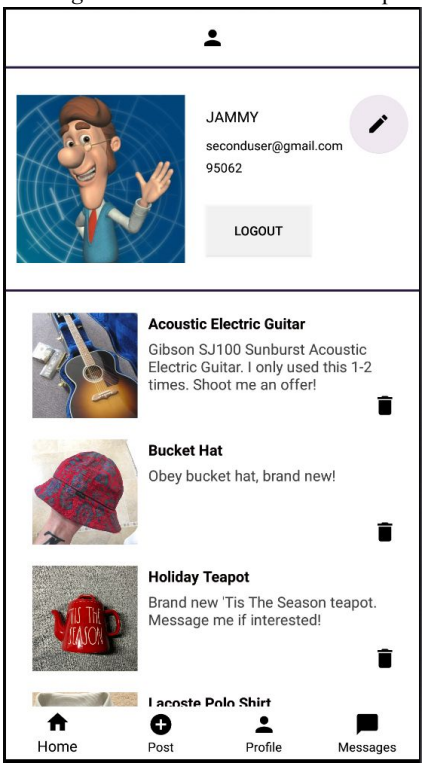

Figure 6: A view of user Jammy's profile.

Lastly, clicking the Post button directs the user to the Add Post activity. From here, the user can list their own items. Users must add input an item name, description, and photo in order for the post to be valid. When adding a photo, the user can select one from their gallery or take one with the external camera. *Figure 7* demonstrates a view of the Add Post activity.

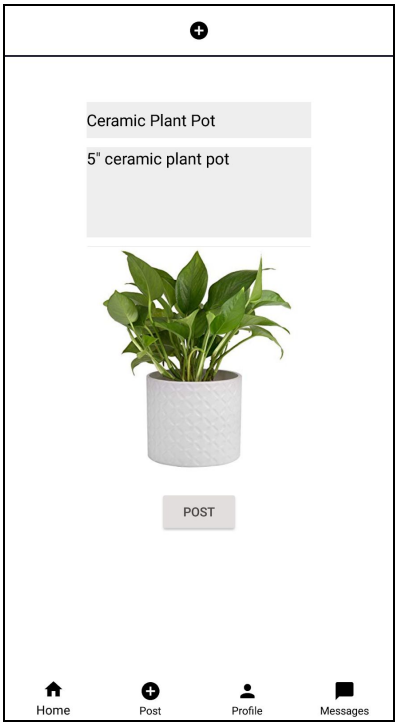

Figure 7: View of Add Post activity.

DEVELOPMENT

The development of *SwapIt* went very quickly once we figured out the most important elements in our application. We worked on the layouts for each activity we intended to implement. Creating the layouts allowed us to have a framework of the smaller tasks at hand. Tasks such as individual users' posts and add post are just a couple examples of items that we wanted to implement from the start.

Once the layouts were ready, we worked on creating the bottom navigation bar and one of the most crucial elements, implementing the Firebase Database. The bottom navigation bar was first implemented as empty fragments until we were finished with our database implementation. At first, getting the database to work correctly was a struggle. Unfamiliar with databases, we relied on extensive research and bug testing. Overcoming the threshold became a great victory during the development. Only with the database working, were we able to work on other important aspects of the application.

After finally getting our Firebase Database set up, we were able to implement user account creation/login with Firebase Authentication. This was foundational to getting

the Add Post and Homepage activities up and running, as they need to be able to see/show which user posted what.

The Add Post and Homepage activities were the next large tasks on our agenda. With the ability to navigate through each fragment via the bottom navigation bar and proper access to our database, we began working on Add Post. We coded the activity and used it to bug test our database and fragments making sure the buttons and other views on the fragments worked properly. Once posts were being properly added, worked on getting posts in the HomePage to display. Using a Firebase RecyclerView Adapter, we were able to display and filter posts by zip code.

Firebase RecyclerView Adapters were also used for the individual user profile and seller profile posts as well. Instead of filtering by zip code, we filtered the posts by user id, so only the user's posts would be displayed. A recyclerview was also used for the Messages list view as well as the messages thread themselves. Having a recyclerview allowed for messages and posts to be updated in real-time.

For the most part, we stayed true to our initial idea in our proposal. Although, some small items were added on to our project. After finishing the large tasks in our agenda, we created buttons and perfected our layouts.

Initially, we wanted users to click on the posts to access the message button. Instead, we added a message button on every post on the Homepage for convenience. This allowed users to access and message sellers more easily. Another addition to our initial idea was adding a share button. Though we have fulfilled our tasks in our proposal, we definitely look forward to adding more to app and continuing to improve it.

INDIVIDUAL CONTRIBUTION

Figure 8 displays a graphical chart of every member's contribution for every part of the project from the proposal, presentations, app development, and final report.

FUTURE WORK

In the future, we plan to implement a more intensive filtering functionality, which will be shown by a drop down menu on the homepage. This drop down menu will show a list of cities, so users can view posts in the city they are in or cities they are close to. If we were to continue developing this application, we would also implement an "add photo" function in the messages activity to allow users to send photos directly through the app.

We see this application progressing in the future because there has been a significant rise in the ecommerce industry throughout the years. What differentiates our application from other buy-and-sell apps is the idea of simply swapping items without using a medium of exchange.

	Proposal	Presentations	Final Report	Final Project (App)
Yza Velasco	Brainstorming Proposal Document: (Components Flowchart, Screen previews)	Interim presentation: Code Contribution- Activity layouts, Log In/ Sign Up/ Log Out, User profile activity, Firebase Auth, Debugging Final presentation: Code Contribution- See Final Project column	App Components, Abstract, Objective Future work	Activity layouts Log In/ Sign Up code Firebase Authentication Log out code Post Item code Share post code User profile - edit name,edit picture, floating action bar Message Activity Messages list (show all messages) Firebase Database Debugging
My Pham	Brainstorming Proposal Document: (Screen descriptions/ components)	Interim presentation: Code Contribution - Activity layouts, Bottom nav bar, User profile activity, Debugging Final presentation: Code Contribution - See Final Project column	App Components, Abstract, Objective, Development	Activity layouts Post Item code Delete Item code Bottom Nav Bar code Home Page code User profile code View Seller profile code Edit user database update code Zip code filtering code Firebase Database Debugging
Sean Jossy	Brainstorming	Interim presentation: Code Contribution: User profile activity, Debugging Final presentation: See Final Project Column	App Components, Development	Firebase Database Camera test- edit user, add post Debugging

Figure 8: Chart of contribution for each person on the Proposal, Presentations, Final Report, and Final Project