Thrift Slug

CSE 118 – Mobile Applications

December 8th, 2019

Gregory Bork  
Computer Science Student  
 University of California, Santa Cruz  
 Santa Cruz, California, USA  
 gbork@ucsc.edu

Matthew Halteh  
Computer Science Student  
 University of California, Santa Cruz  
 Santa Cruz, California, USA  
 mhalteh@ucsc.edu

Rafi Haque  
Computer Science Student  
 University of California, Santa Cruz  
 Santa Cruz, California, USA  
mrhaque@ucsc.edu

1 Abstract

For our CSE 118 Mobile Applications project, we decided create a thrifting mobile application called Thrift Slug. The mobile application will allow users, who we classified as UCSC students, to buy and sell clothing, shoes, and other accessories. The application allows for users to create and update their profile, contact sellers, open Venmo, and most importantly, purchase and sell quality clothing!

In this final report, we will be covering an overview of our entire project, comprising of the following information:

1. A high-level overview regarding the purpose, reasoning, and influence behind our application
2. The various features and components our application consists of (including images of our User Interface model)
3. Our hierarchical developmental process, comparing our initial application model with how our application actually implemented and built
4. A rough estimate of each member’s contribution to the overall application and modeling
5. Future progress and exactly what features we hope to implement in the near feature

2 Objective

As previously mentioned, for our CSE 118 Mobile Applications project, we decided developing a thrifting application called Thrift Slug. The mobile application is intended for University of California, Santa Cruz students to affordably sell and shop for used clothing, shoes, and accessories. Thrift Slug essentially allows users to clear out their wardrobes, while making money at the same time. Although the application allows users to post any type of clothing or shoe brand for sale, we envisioned the application to be used for middle-to-high end clothing. We see our application as the future go-to thrifting platform of all UCSC students. By creating the application, we wanted to encourage student-to-student commerce that essentially gives students a secondary source of income and apparel.

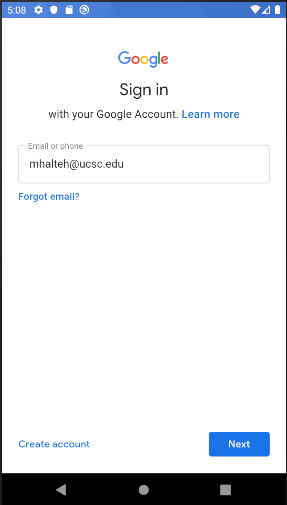
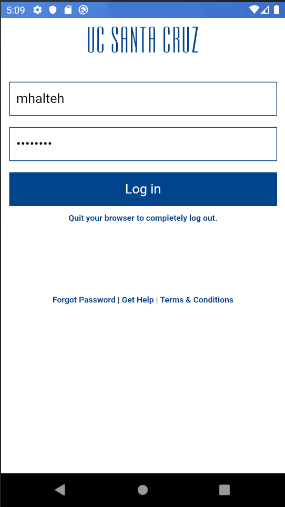
There are few other applications like our mobile application, and our influence for this project was a web based application called Craigslist. There are other similar mobile applications like Poshmark and ThredUp, however our application is a little different. We were thinking of a Craigslist-type platform, but turned into a mobile platform, specifically for clothing, shoes, and accessories. We also wanted to build an application that is specifically intended for set group of individuals: UCSC students. Unlike other thrifting applications, our application enables for students to organize a meet-up location in Santa Cruz to exchange the product and money. This actually cuts any sort of shipping cost the seller or buyer would generally have to pay for. Our application also encourages a peer-to-peer communication model, which encourages

3 Components  
3.1 Logo and Login

Like any other mobile application, our application began with us developing a logo. So, that’s actually the first thing the user sees as an icon:



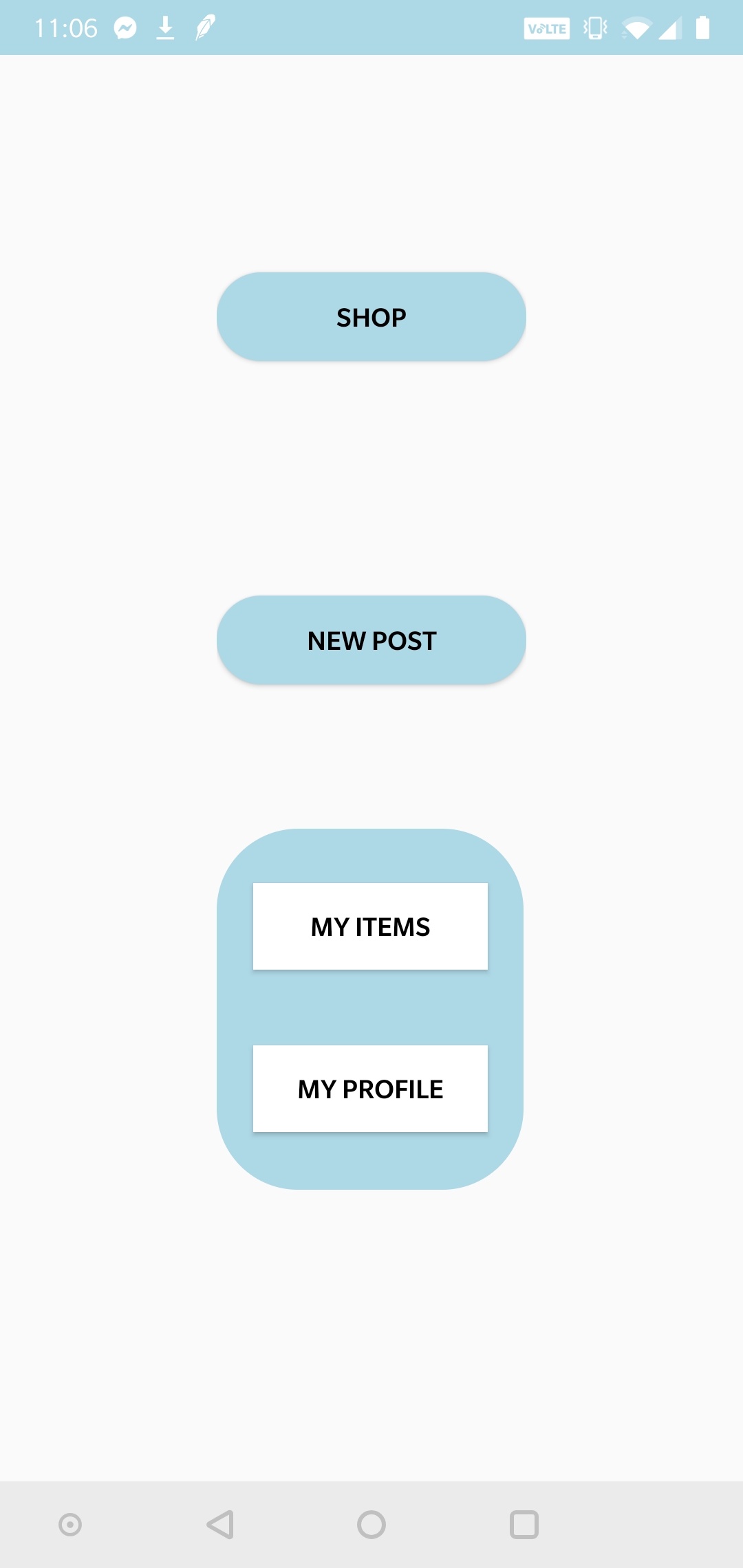
Upon clicking the icon from the home screen the user is prompted to login via Google sign in. The following process looks like this:

As you can see, our application login is quite simple, and the user must use google to login. We ultimately chose a Google login platform because of the fact that all UCSC students have a valid google account through their “@ucsc.edu”. We were then able to restrict login to accounts ending with “@ucsc.edu” rather than “@google.com”. If a user were to try and create an account using a non-UCSC email, the mobile application would restrict entry for that specific user email.

**3.2 Home Screen: Buying, Selling, My Profile**

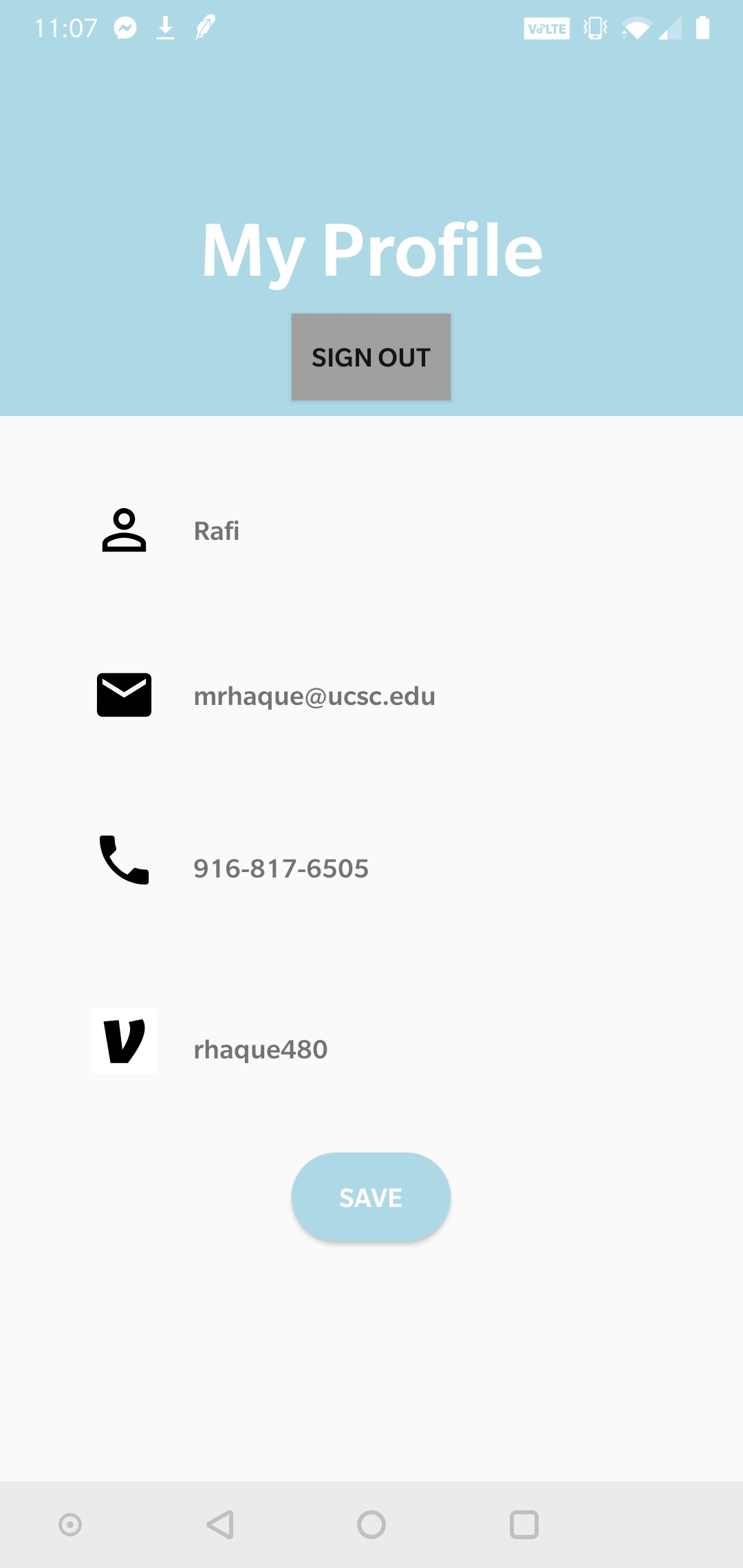
In order to keep our home screen as simple and user friendly as possible, we went with a clear approach that has four buttons for the user: *Buying* button, *Selling* button, *My Items* button, and a *My Profile* button. The following screen is presented below:



We tried highlighted the main buttons of the screen, *Buying* and *Selling*, by changing their button size, color, and shape. This would inevitably give the user a clear indication as to which buttons are meant for the actual process of buying and selling a product. *My Items* presents the items posted by the seller, and the *My Profile* section shows the information about the user.

**3.3 My Profile**

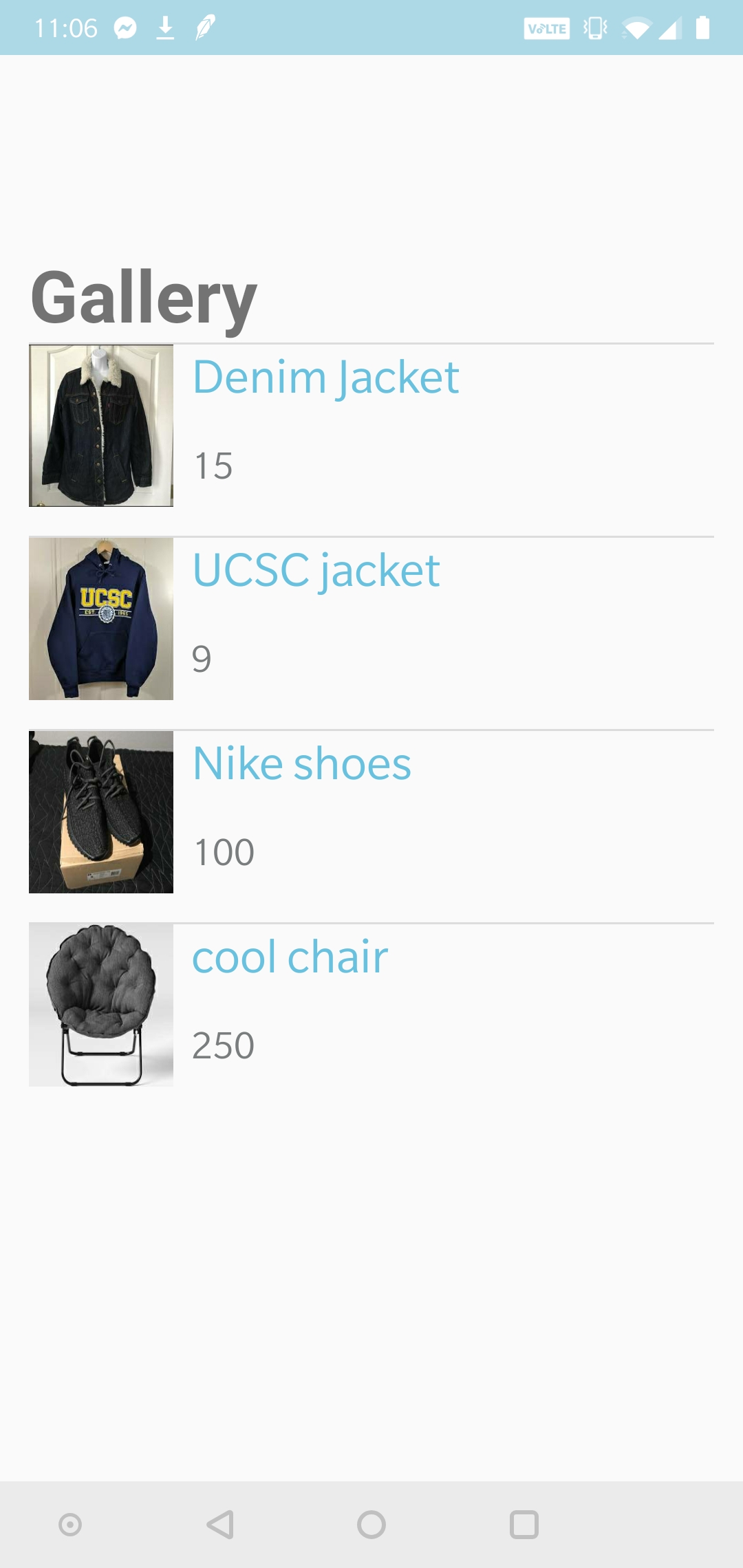
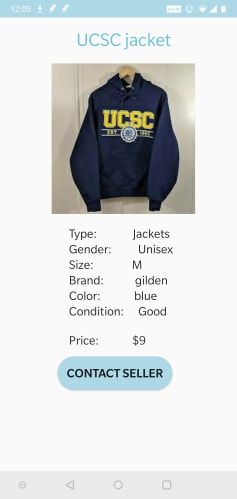
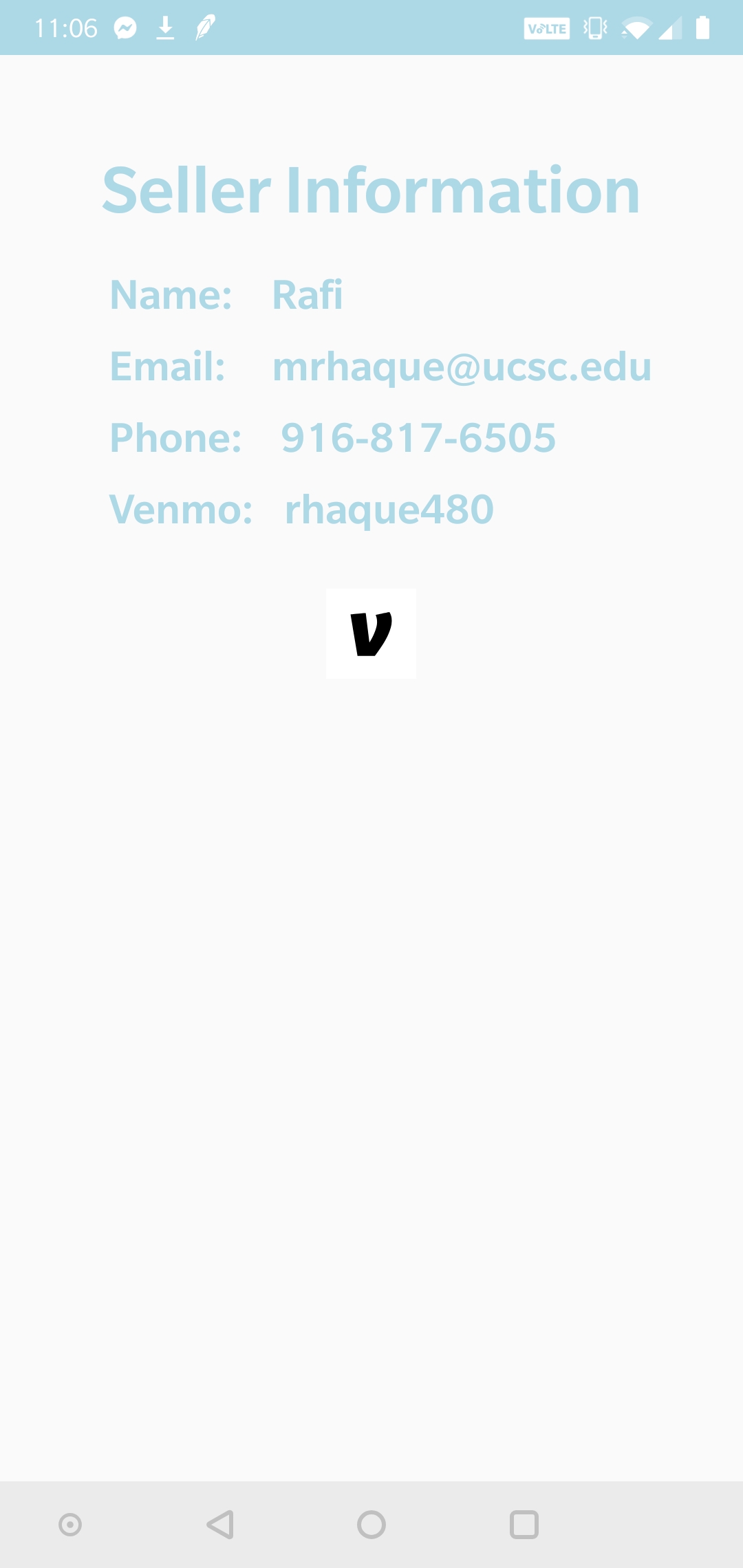
Upon clicking the *My Profile* button, the user is brought a page containing the user’s name, email, phone number, and Venmo account. The user can actually edit their phone number and Venmo username through the application, however the user’s name and email address are taken from their Google account, so that cannot be changed, unless the user logs in from a different UCSC account.



There are a couple supplemental features that we added on the *My Profile* page. One of which is the *Sign Out* button at the top of the page. The user can actually click that button and the application will log them out and require a google entry if the user tries to login again. Another feature that we added was the Venmo shortcut button. The venom icon, show above, in our mobile application is actually a shortcut to the venom app. So when the icon is clicked, the Venmo application will open on the user’s mobile device (as long as they have the Venmo application previously installed, of course). The user can also edit their phone number and Venmo username, and it will save and reflect the update in future screens within the application. After making any edits, the user can then click the save button and return to the main screen. The user can also return to the main screen by clicking the back button.

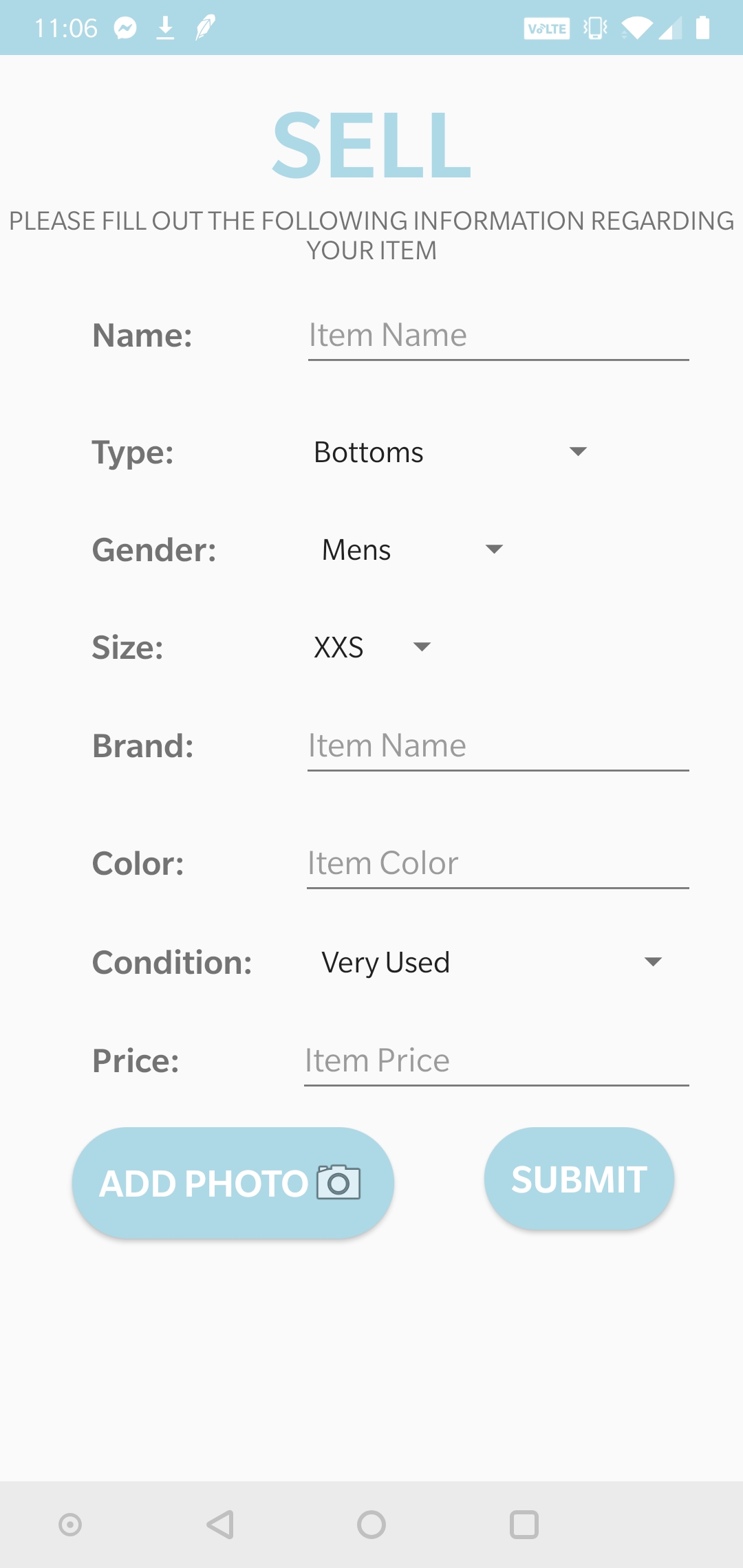
**3.4 Buying Feature**

If a user is interested in purchasing an item, they would essentially click the *Shop* button from the home screen. Upon clicking, the *Buying* screen presents a gallery of listings that the user can scroll through to view. If the user would like to get more information regarding a specific item, they could actually click any listing and it will open up an expanded screen containing all the information the seller posting about the item. Now, if the user decides that they would like to purchase the item, they can click the *Contact Seller* button shown in the middle screen below. Once that button gets clicked, the seller’s information will appear for the user to view and contact. There is also a Venmo-linked button as shown in the rightmost photo. The UI images for the *Buying* feature is shown below:

**3.5 Selling Feature**

The selling feature in our application is very straight forward. Essentially, the user clicks the selling button from the main screen, and the following screen is presented:



As you can see, the screen prompts the user to *fill out the following information regarding [their] item*. When creating this screen, we envisioned a simple UI where the user can enter in key information regarding their product. We ask the user to provide the following product information:

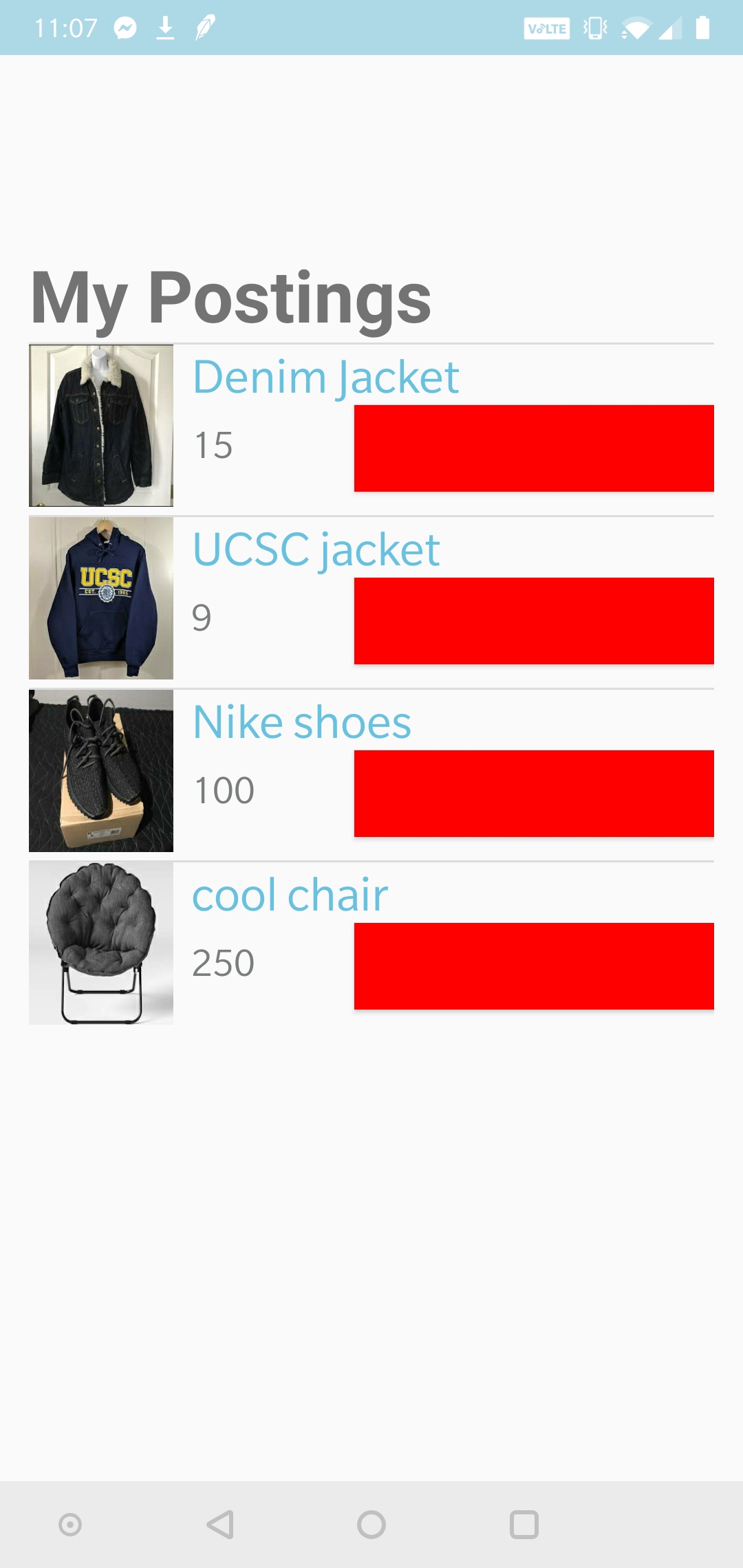
1. Name: The products designated name by its company (Ex: Bomber Jacket)
2. Type: The items wearable purpose (Ex: Jackets)
3. Gender: The gender correlated to the size of the item (Ex: Womens)
4. Size: The identified size of the article of clothing (Ex: S)
5. Brand: The brand specified on the article of clothing (Ex: Urban Outfitters)
6. Color: The recognizable color of the item (Ex: Black)
7. Condition: The condition of the item per the seller’s discretion (Ex: Excellent)
8. Price: The seller’s preferred item price (Ex. $18)

As shown, some entries have a built-in drop down menu that has a set list of options. This makes it easier for the user to enter in information, and it also enables us to later filter entries based on these item options.

Once the user has entered in all the information regarding their item, they can take a photo of their item to post alongside their product information. As the user clicks the *Add Photo* button at the bottom of the page, they are brought to screen that has another button at the bottom of the page labeled *Upload Image From Camera Roll*. Upon clicking that button, the application will ask the user for access to the phone’s camera and photos. The user can then take a photo or choose one from the camera role. Once the photo is chosen, the user can then hit the *Post Item* button.

**3.6 My Items Feature**

In order for the user to access their listings, we created a *My Items* feature as shown below:



As you can see, the user’s listings are shown in the order they posted them. Because this is essentially a user-programmed application, the seller deletes an item after they sell it. The large red button next to each item indicated a deletion from the database. So essentially, the user can delete any entry they have posted by simply clicking the red button.

4 Development

Once we came up with a rough idea regarding our application, we began analyzing the steps in order to produce a respectable outcome. We created a flow chart to map our ideal application protocol:



As you can see, we initially planned on and created a screen that would prompt the user to login with a traditional login platform, and if they don’t already have an account, they would create one in a new screen. As we began the project, we realized that it would be logistically more efficient to use a Google login platform because it enabled us to restrict access to only UCSC students. This was actually the first major change we made from our originally planned flow chart. After logging in, we envisioned there to be some sort of Main Menu screen that contained two feature: Buy and sell. We did implement this, however, we decided to add a *My Profile* feature to our Main Screen so that the user can access their information, name and email, which we pulled from their Google login. As we were finishing up our project, we decided to add another feature to our Main Menu that would allow the user to look at their listings. This would help the user with deleting their entries once they have been sold. We called this feature *My Items*.

Once we created our login and main menu features, we decided to continue with the creation of our User Interface before we jumped into the backend. We wanted to ensure that our product looked visibly pleasant prior to making sure every single feature worked. So, continuing with our flow chart, we decided to work on the selling feature of our application first, because it’s impossible for a user to *purchase* a product if no products exist. Essentially, we wanted to create a *selling* platform that the user can easily navigate. In the flow chart, we decided to ask the user for the item’s condition, price, picture, and price. However, we later decided to add the item’s color, type, name, and gender. We tried modeling these options after online shopping websites in order to give users the ultimate shopping experience. In order to make the navigation easier, we thought it would be even easier to add a drop down menu of pre-written options for some of the item descriptors. As previously mentioned, this would help us with filtering certain aspects of an item. In our flow chart we had an option give the user an option to the seller’s listed items. Unfortunately, at the time we felt that the feature was bit out of reach for the scope of this project. But as we continuing working on the project, we realized that this feature was a necessity to the overall success of our project, so we managed to include it as a last minute addition.

The development for *Buying* feature was a little more complex than initially anticipated. In order to get this feature completely set up, we realized we needed to start the backend of mobile application. In order to implement the backend, we decided on using a MySQL database in order to store each entry created in the *Selling* feature. Essentially, the first step is that a user creates and posts an item to sell. Once the user clicks *Post Item*, the information about the item gets pushed to the database. In the *Buying* feature, the information from the database is pulled and represented on the screen. Our *Buying* screen, as shown above in section 3.4, has a list view of various entries, presented in the order they were submitted into the database. As mapped out in our flow chart, each item can be clicked for more information. A few aspects of this feature that we initially didn’t quite plan correctly are the actions of contacting the seller, completing a purchase, and deleting an entry. We initially thought all these options would happen at the same time, but we realized there were some logistical issues with our initial plan. Because we don’t have a pay-through-application option, we realized we can’t delete the entry once the buyer clicks on a purchase now button. We figured the buyer and seller need to meet up first, exchange the item for money, and then the seller can later delete the entry. In order to accomplish this, we created a *My Items* option in the Main Menu, as previously mentioned. Now, once the seller clicks the item, the item specifics pop up, and the user can contact the seller by clicking on the *Contact Seller* button. We tried our best to model our system after Craigslist in that we want the seller to have the authority to delete an item which a purchase has been completed or anytime at will. Once we achieved this last part of our mobile application, we wrapped up our execution and fixed minor flaws and UI aspects.

As you could already tell, we definitely had some major execution changes from when we initially started the project, to when we completed it. The development process was a long, stressful task, but it was key to the completion of our application. Our approach of completing the UI first and the backend second worked very well, up until we needed the backend to proceed with the UI, specifically in the *Selling* feature of the application. It was very rewarding to see our initial proposal and flow chart evolve and turn into a successful finishing product.

5 Contribution

In regards to overall contribution throughout the entire project, each member took a very hands-on approach to whole of the project. Essentially, each member took *lead* on a specific task but we each encouraged each other to contribute to nearly each aspect so that we could get the best possible implementation of our application. Although we all aided each other with every task, each member’s lead tasks are outlined below:

* Matthew’s Lead Tasks: Project Proposal (Application Description and Flow Chart), initial project UI (Main Menu, *My Profile* feature, *Selling* feature), Venmo shortcut, photo uploading task, interim presentation PowerPoint, and final report primary writer
* Rafi’s Lead Tasks: Project Proposal (Details segment), Google Login and Google data retrieval in *My Profile*, backend MySQL database implementation, initial project UI (*Buying & My Items* feature), PowerPoint presentation editing, and final report draft editing
* Greg’s Lead Tasks: Project Proposal (final editing), Logo and icon development, initial project UI (*Buying* & spinning features on buttons), completion of photo upload conversion for database, interim presentation PowerPoint editing, and final report draft editing

In regards to the individual work presented above, we held team meetings almost every Monday and Friday, and spent a combined 12 hours a week working and executing each part of the application. We really enjoyed working with one another, and we are very proud of the work we have put forward and accomplished. Although we feel like we have “finished” the application for the scope of this class, we will definitely be working some more on our project, and we would like to see it evolve into full-scale, student-used application.

6 Future Work  
 Although we were actually quite happy with the work we put forward, there are definitely improvements that we would like to make to our application before we actually publish it. First and foremost, we would like to upgrade our UI. We felt like our UI was easy to use and quite accessible for the user, however, we would like to make an upgrade so the looks are a little more visually captivating. We are actually very happy with how the functionality of our application worked out. Everything worked as originally planned, so we are glad our hard work payed off. One minor detail that would also like to address and fix in the future is the compression of an image when uploading. This would save us lots of storage in our MySQL database and would actually improve the run time of uploading an entry to the database. Other than these two edits, we are glad the three of us had the opportunity to work with one another, and we look forward to adding further work to Thrift Slug.

ACKNOWLEDGMENTS

We would like to thank our TA Vihang Godbole for his support and guidance on this project. We really appreciate him taking the time to explain various Android Studio and concepts that we were able to implement in our application. Thanks Vihang!Conference Name:ACM Woodstock conference

Conference Short Name:WOODSTOCK’18

Conference Location:El Paso, Texas USA

ISBN:978-1-4503-0000-0/18/06

Year:2018

Date:June

Copyright Year:2018

Copyright Statement:rightsretained

DOI:10.1145/1234567890

RRH: F. Surname et al.

Price:$15.00