# **Quick Cart App**

# **Android Mobile Application**

Version 1.0.0>

Prepared by

|  |  |  |
| --- | --- | --- |
| Dejeon Battick | 180914 | Dejeon.battick11@gmail.com |
| Shemar Lundy | 180916 | [lundyshemar@yahoo.com](mailto:lundyshemar@yahoo.com) |
| Shemar Henry  Mark Anthony | 180915 | [shemarhenry24@yahoo.com](mailto:shemarhenry24@yahoo.com) |
|  |  |  |

|  |  |
| --- | --- |
| Course Instructor: | Thomas Xu |
| Course: | Web and Mobile Application development II |
| Date: | June 23, 2019 |

# Overall Description

## Product Context

The Quick Cart application was designed for everyday shoppers who visits a supermarket. The Quick Cart mobile application for android also has an iOS counterpart and allows the user to shop easily in the comfort of their homes. The application allows users to search for items in the supermarket and add it to their cart. They would then check out such items from the cart and a QR code would be generated for those items, then a message would be sent and their items would be prepared and packaged by the workers in the supermarket. The customer would then go to the supermarket and present the QR code and collect their items which have been prepared. The customer could collect their items after payment has been made. The Application is also very useful for customers who visits the supermarket and cannot find the items they are looking for. The locate page allows users to search for items and they would see where the item is located rather than to be walking and searching for the items while in the supermarket. The application also has a online community blog. This community blog will allow users to share recipes with each other and be kept abreast on posts made by the Quick Cart blog administrator.

## Product Functionality

**Major functions of the system will be to:**

1. Allows the user to search for items.
2. Allows all users to add items to a cart.
3. Allows all users locate items in the supermarket.
4. Allows a users to check out items.
5. Allows a user visit the community blog.

**Non-functional requirements of the system will be to:**

1. users information should be stored on a secured database.
2. User password should be hashed to ensure security.
3. System must check credentials against the user database to ensure that there isn’t duplication in the email field.
4. System should ensure search results are present in a timely manner.
5. System should ensure that a user login session is kept until the user manually logout .
6. System should ensure that the main flow of the application is smooth.

## Stakeholders and Users Characteristics

**The key stakeholders of this system are:**

**Shoppers:**

- Persons who use the Quick Cart application to shop from their homes.

- Persons who visit the supermarket to shop in store.

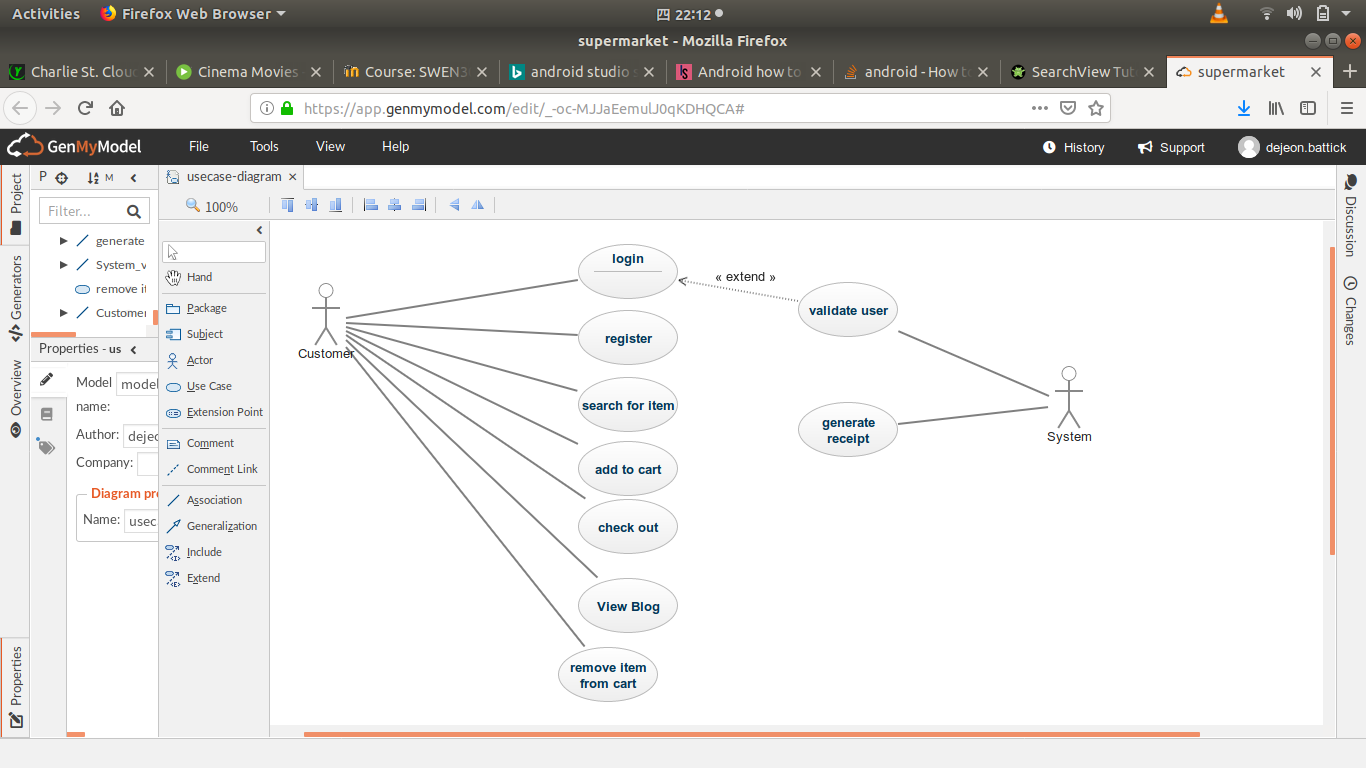
**Store Staff:**

- Persons who view shoppers receipt to prepare their packages.

## Operating Environment

The application is suited for android Operating Systems Oreo 8.1 and all newer versions. This is the android version of the application there fore it is suited for all android devices. It Internet connection is required to to check out for the supermarket to get a list of things to be prepared for that user. Internet connection i also required to visit the blog.

**Use Case Diagram**



## Use Case Description

**Use case name:** Login

**Summary:** the actor should be able to log into the application.

**Primary Actor:** Customer

**Secondary Actor:** n/a

**Precondition:** the user must have an account registered.

**Main Sequence:**

1. The user enter their email
2. The user enter their password.
3. The user click the submit button.
4. The user credentials are validated.
5. the user logs into the system.

**Alternative sequence:**

**Step 4:** if the user do not have an register account then they can go to the registration page.

**Post-condition:** Toast message saying login was successful.

**Use case name:** Registration

**Summary:** the actor should be able to create a new account.

**Primary Actor:** Customer

**Secondary Actor:** n/a

**Precondition:** the user must have the application installed.

**Main Sequence:**

1. The user clicks the registration text field
2. The user enter their user name.
3. The user enter their email.
4. The user enter their password.
5. The user enter their password again for confirmation.
6. The user click the submit button.
7. The user logs into the system.

**Alternative sequence:**

**Step 6:** if the user clicks the login button then the can return to login page.

**Post-condition:** A new user is created.

**Use case name:** add to cart

**Summary:** the actor should be able to search for item and add them to the cart.

**Primary Actor:** Customer

**Secondary Actor:** n/a

**Precondition:** The user must have logged into the application.

**Main Sequence:**

1. The user click on the search bar
2. The user clicks on the plus icon to increase the number of items
3. The user click on ht cart icon to add the item to the cart
4. the user click on floating cart button to go to the cart page

**Alternative sequence:**

**Step 2:** The user opens the navigation bar then click on the cart to go the cart page.

**Post-condition:** Items are added to the cart.

**Use case name:** search for item

**Summary:** the actor should be able to search for item to know where they are located in the supermarket.

**Primary Actor:** Customer

**Secondary Actor:** n/a

**Precondition:** The user must have logged into the application.

**Main Sequence:**

1. The user opens the navigation bar and click on locate
2. The user click on the search bar
3. The user search for the item they want to find

**Post-condition:** The item card is display with the location of the item.

**Use case name:** check out

**Summary:** the actor should be able to check out items added to the cart.

**Primary Actor:** Customer

**Secondary Actor:** n/a

**Precondition:** The user must have items added to cart.

**Main Sequence:**

1. The user go to the cart page
2. The user clicks on the cart floating button to check out items
3. The System generate a pop up of the QR code
4. The user click on the OK button to close the QR code and store it.

**Post-condition:** QR code is generated and is saved.

**Use case name:** remove item from cart

**Summary:** the actor should be able to remove items from the cart.

**Primary Actor:** Customer

**Precondition:** The user must have items added to the cart and on the cart page.

**Main Sequence:**

1. The user click on the menu option on the item
2. The user clicks on the delete
3. The item is removed from the users cart

**Post-condition:** Items are removed from the users cart.

**Use case name:** View Blog

**Summary:** The actor should be able to visit the community blog which is the web service of the application.

**Primary Actor:** Customer

**Precondition:** The user should have internet connection.

**Main Sequence:**

1. The user click on the navigation button.
2. The navigation bar is open.
3. The user clicks on the community text.
4. The online blog is launched.

**Post-condition:** The blog web page is open online.

**Use case name:** generate receipt

**Summary:** The system generate a QR code the the receipt information.

**Primary Actor:** System

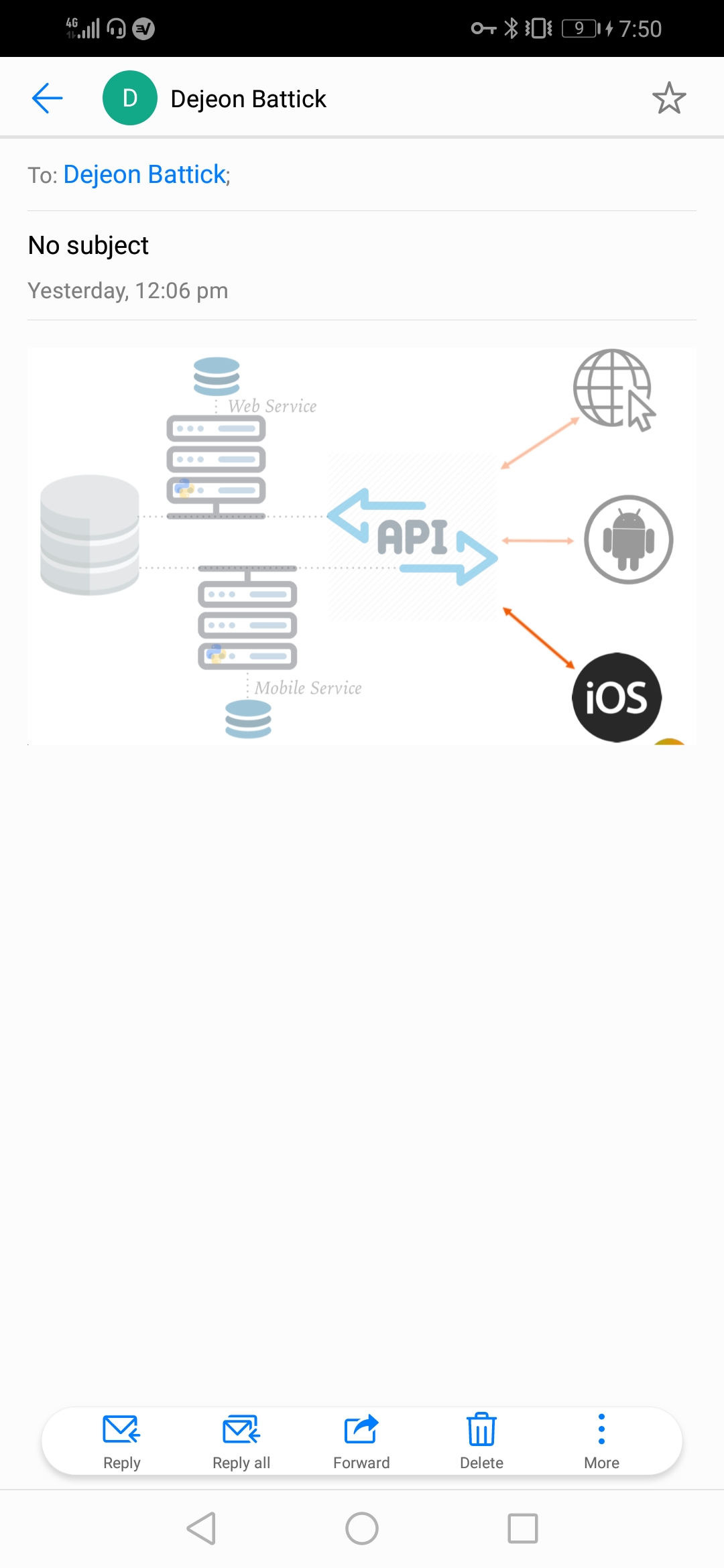
**Precondition:** The user check out their items out of the cart.

**Main Sequence:**

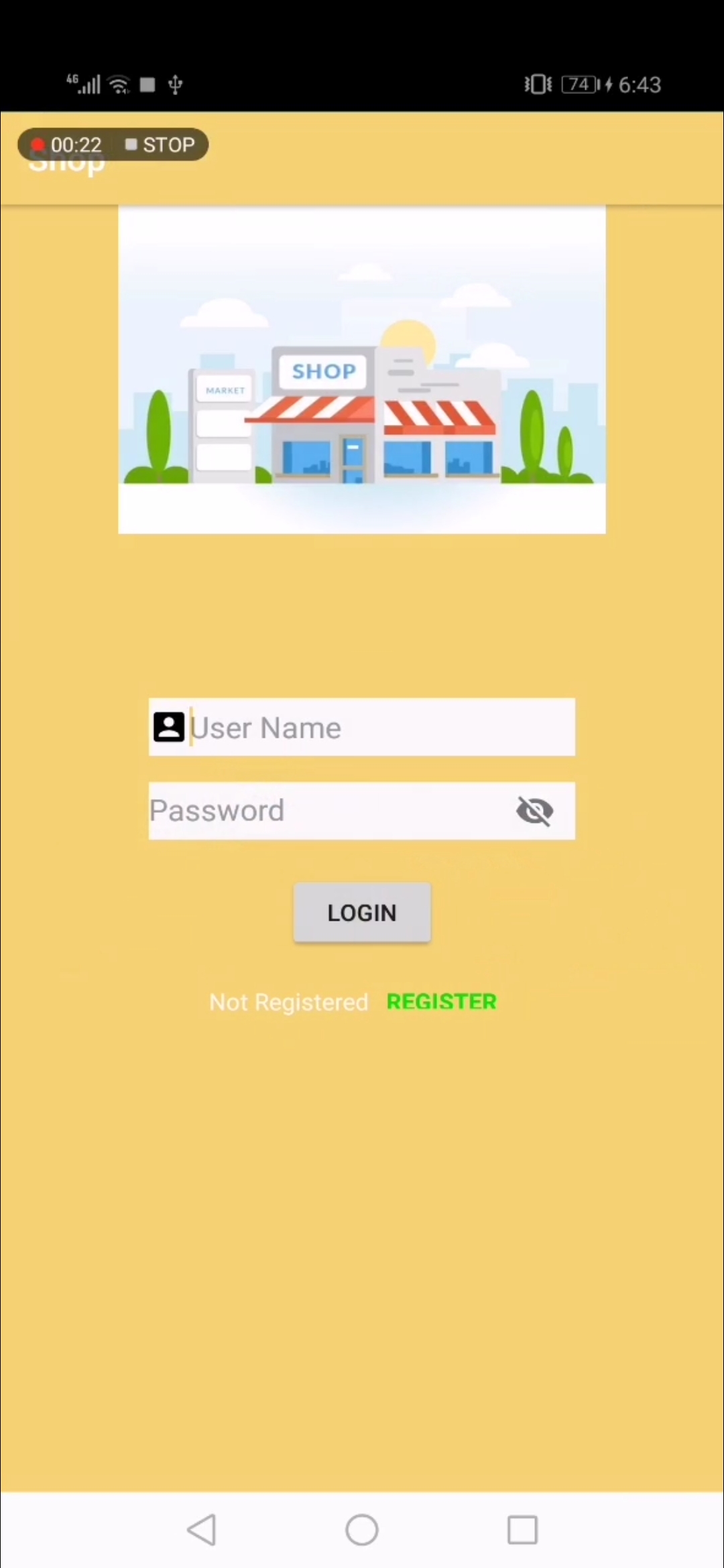
1. The user check out
2. The System generates a QR code
3. The system stores the QR code

**Post-condition:** The QR code can be viewed anytime by user.

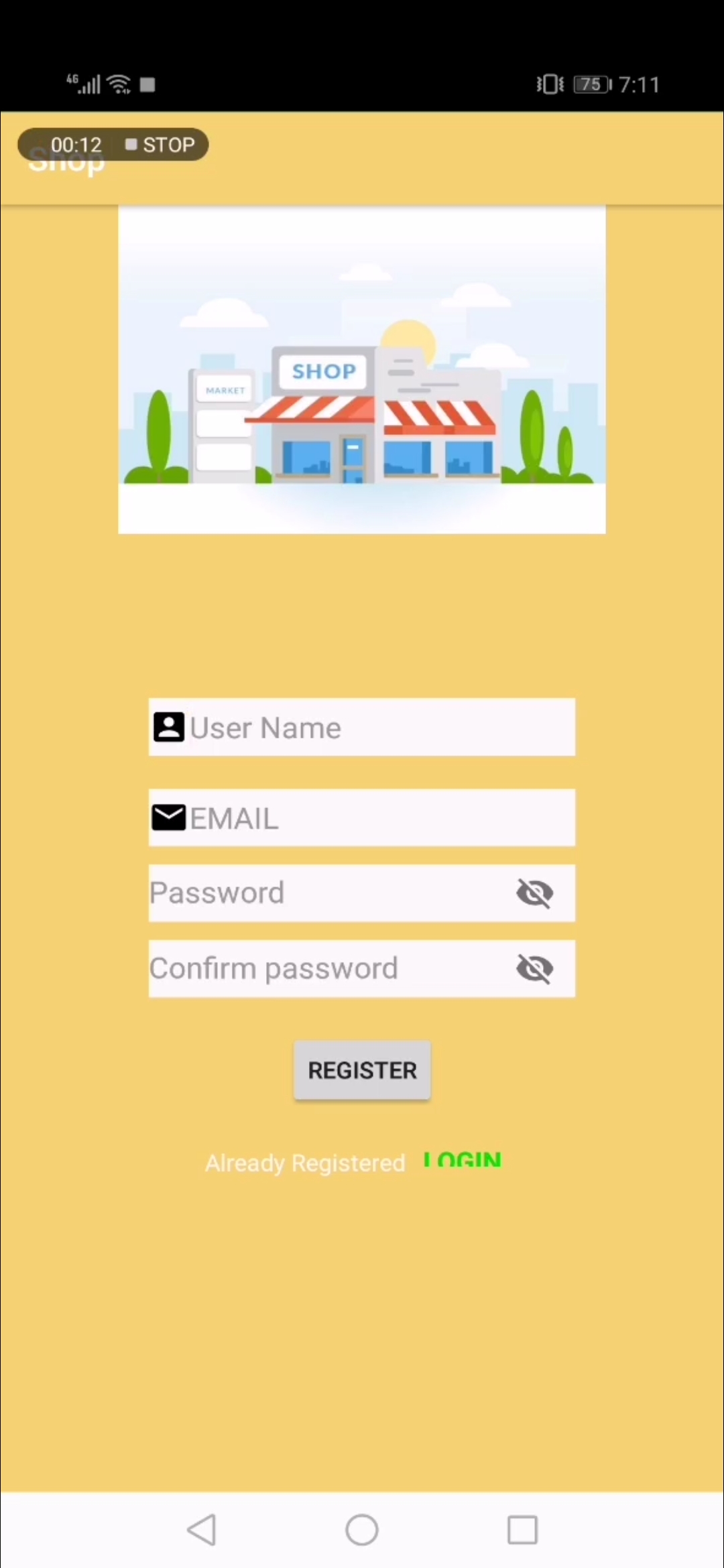
**Structure**



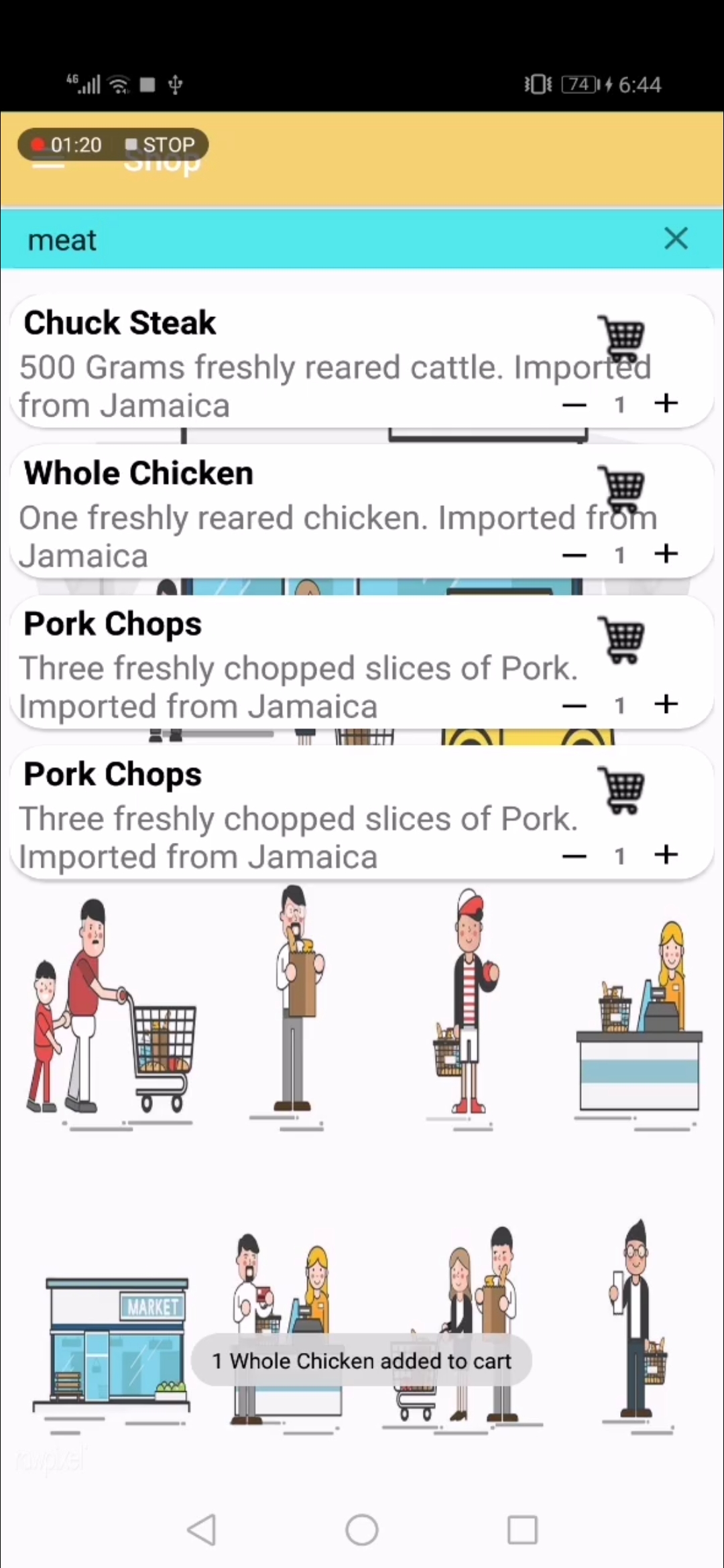
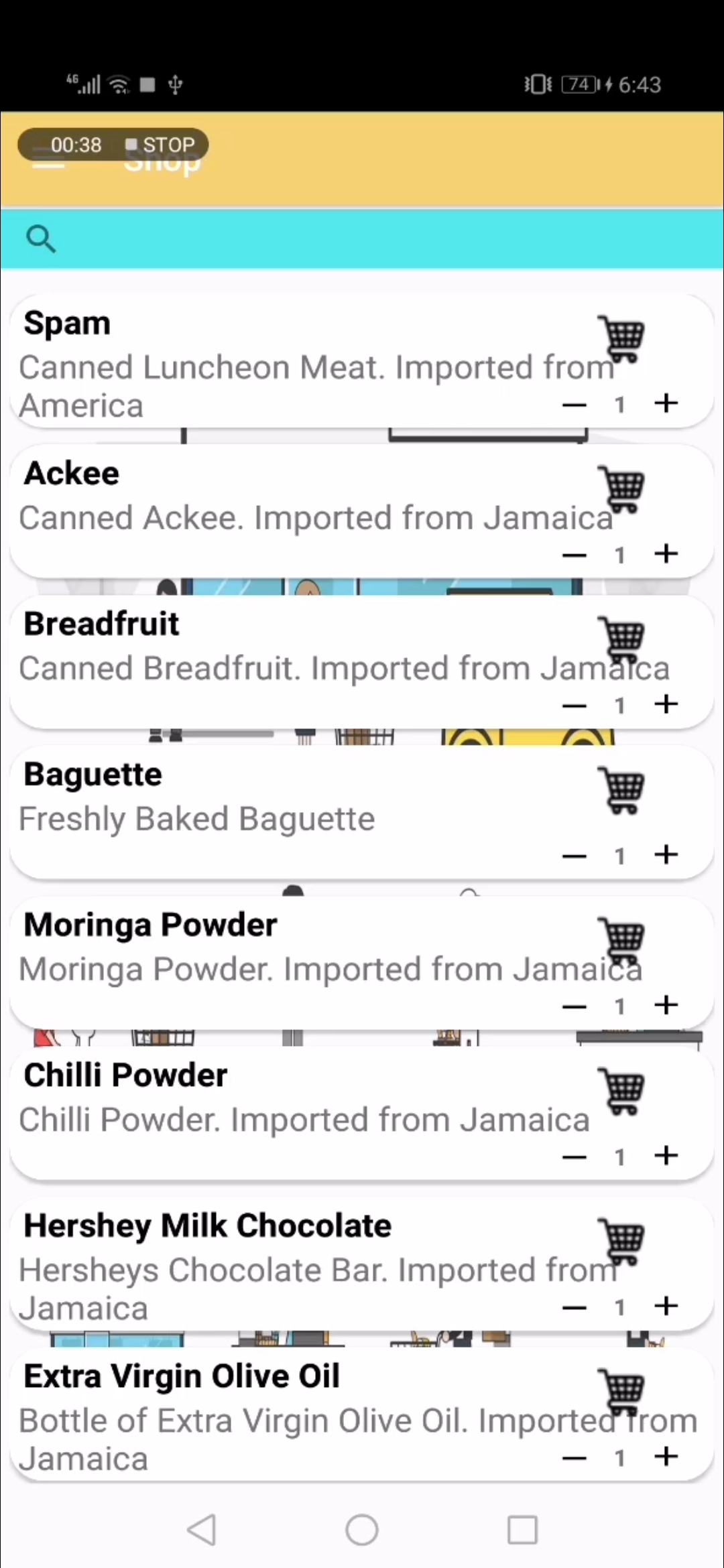
Above is an image of the structure of the quick cart application. As mentioned before there is an iOS version that is discussed in more details in the iOS submission. There is also a Web service community that is also discussed in the web submission. There is a middle layer API that handles all the heavy lifting and communication between layers.



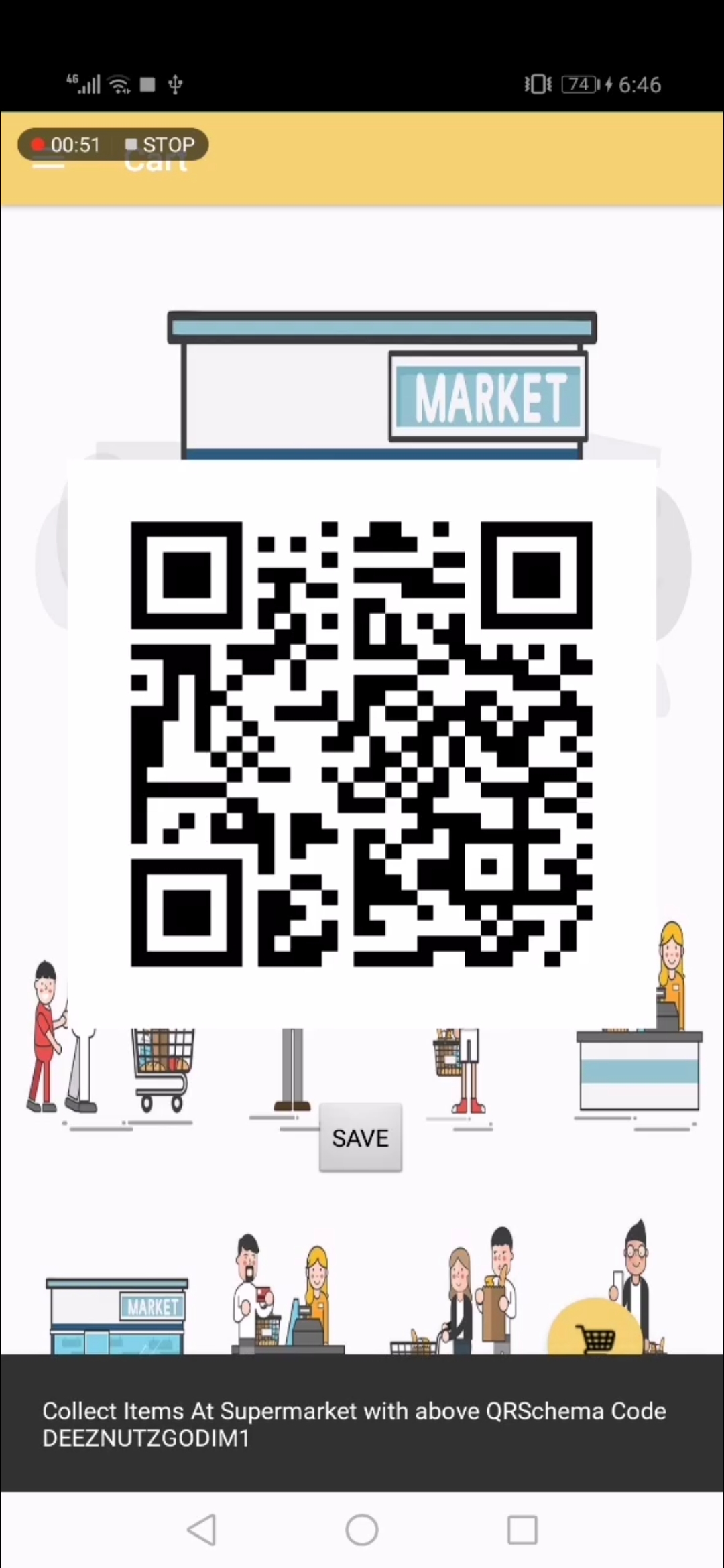
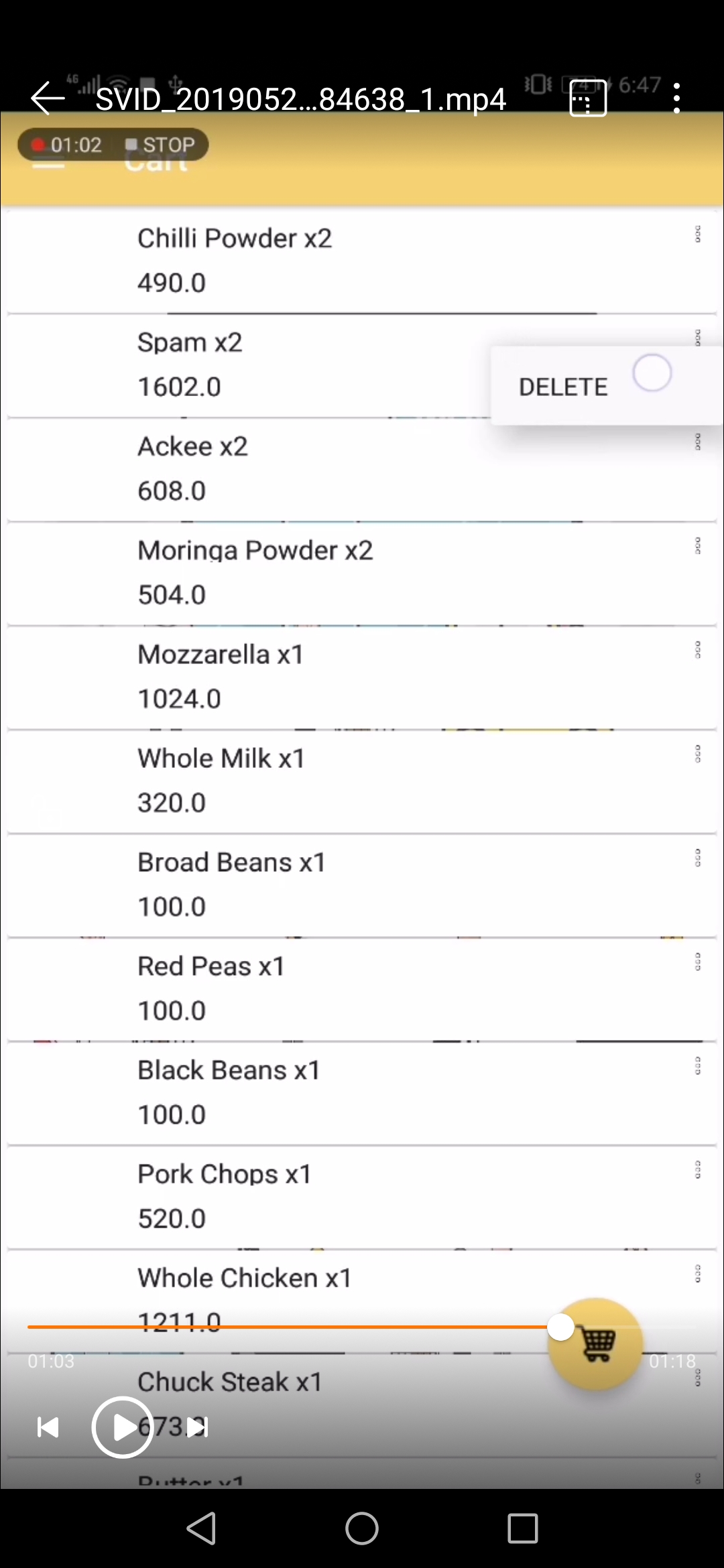
The above image is the login page where existing users can login with their information, the users credentials are stored in the My SQL database. The validation is done on the back-end of the application to ensure that passwords and usernames match. If the user is a new user they can click on the register text field to go there and register.



The above image is the registration page in which a new user can sign up. The page takes the user’s, name, email and password. It also requires the password to be entered twice for verification. The new user’s information is stored in the database as a new field is created with their in formation.



The above image the main page when logged in, the shopping page where persons can view items and add them to their cart. There is also a search bar foe persons to search for a specific item they are looking for. The search bar accepts the user’s request and make a query of its own gathering the information from the database and displaying the results in the recycler view.



The above is an image of the cart where users can view their items they added, they can also delete items before they check out. After they check out they see a QR code which is their receipt to be scanned to collect their items.

The above is an image of the find item page which is very useful for in store shoppers to locate items they are searching for. The view for this page only shows the name of the item and where to find it. This page also has a search view which works similar to the on on the search page.

This is an image of the profile page which has the information of the user, a link to the online blog and a link to their transactions. Person can click on blog to take them to the online website, internet connection is needed for this. Persons can click on transaction to view their stored QR codes for their packages to be collected

**Future Implementations:**

* Implementation of an in app payment method
* Implementation for users to change their user information
* Implementation for a delivery service and for users to keep tract of their delivery via map
* Implementation for users to view all their previous purchases
* Implementation of a favorites feature for users to add items to their favored list

**Group members contributions**

Our group members, Dejeon Battick, Shemar lundy, Shemar Henry and Mark Anthony Jones worked acidiously to achieve our goals for this project. We are

proud of what we have done and with each other contributions to the

project.

Dejeon battick: worked on the login and registration features and user interface and middle layer(API).

Shemar Lundy: worked on the user interface the web server

Shemar Henry: worked on the middle layer(API) with interaction between the app and the database

Mark-Anthony Jones: worked on back-end and the database and servers