

Project 6 Status Summary

Project name: Boulder FM - Farmers Market

Team member: Qinglu Sun, Qihang Mao, Yibo Yang

Work Done:

Using Flutter Framework, Programmed in Language **Dart**.

- Login Page
 - Users can login to the application, and the back end is supported by Google Firebase Authentication.
 - Login fail notification update
- Register
 - Users can register their account and specify their role (Customer/Store). The data is stored in the Google Firebase Firestore Database.
 - Users need to register with required information instead of empty
- Home Page
 - Customer Home Page
 - When logged in as customer, users can see all the store's listing
 - Store Home Page
 - When logged in as Store Manager, users can see the store's detailed information and have access to post notifications, make changes to their products.
 - Users can jump to other pages with the footer icons
- Profile Page
 - Users can change their information.
- Notification Page
 - Using Observer Pattern, customers can receive notifications from the stores they subscribed to. Stores can post updates to all the customers.
- Created class structures for types of products, customer/store.
- Built partial connections between the front-end and backend.

Change/Issues Encountered:

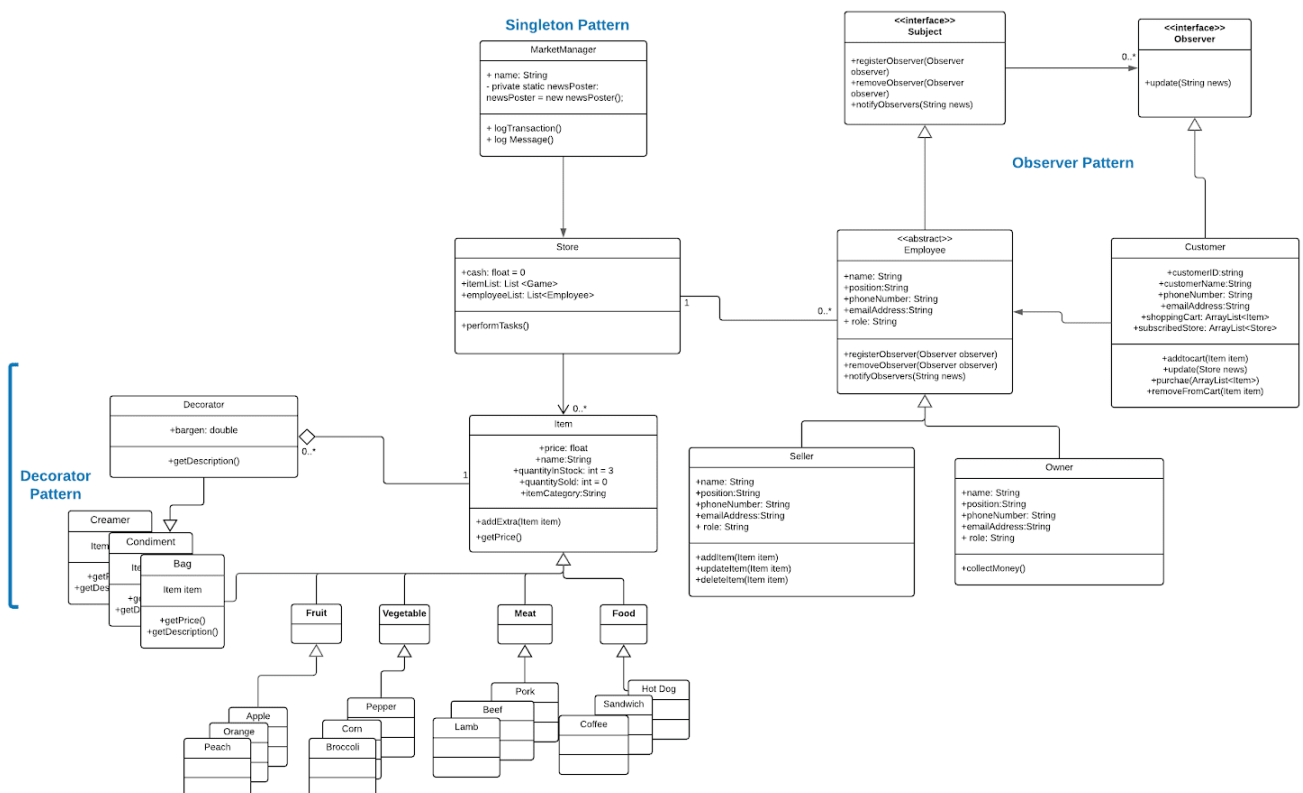
- We changed from using Python Django framework to using Flutter Framework to create mobile applications.
- Change coding language from Python to Dart
- Forget password function remove due to Firebase authentication required

Patterns:

- Observer Pattern:
 - Using Observer Pattern, customers can receive notifications from the stores they subscribed to. Stores can post updates to all the customers.
- Singleton Pattern:
 - Create object *MarketManager*, which will handle issues related to the market itself from both customers and the store manager.

- Decorator Pattern:
 - When adding items into their cart, customers can choose additional add-ons for the product.
- Factory:
 - Stores can have pre-made Fruit Baskets (combination) for their customers. Then customers can directly add it to their cart and pay for the item.
- Adaptor Pattern:
 - Creating adapter connecting the data stored in Firebase database and local class objects.

Class Diagram (20 points)



Plan for Next Iteration (10 points)

- We need to work on the purchase process, as users add items into their shopping cart and make the purchase.
- Sellers need to make modifications on their products such as changing quantities, adding new items, and removing sold items.
-

Project 6 presentation

- Work done part above
- Singleton Pattern finished

Project 7

- Need to work on Factory Pattern for stores to create their special fruit basket, and list it on the store page.
- Need to work on MarketManager functions to enable it to receive messages from both stores and customers.
- Need to work on Billing Page for making transactions

Project due

- All excepted work finished