# Shop With Me

Android Application Final Write-Up

SI 543

December 10, 2014

Team ShopWithMe:

Yuheng Chen

Michael Collins

Nick Malzahn

**The Pitch**

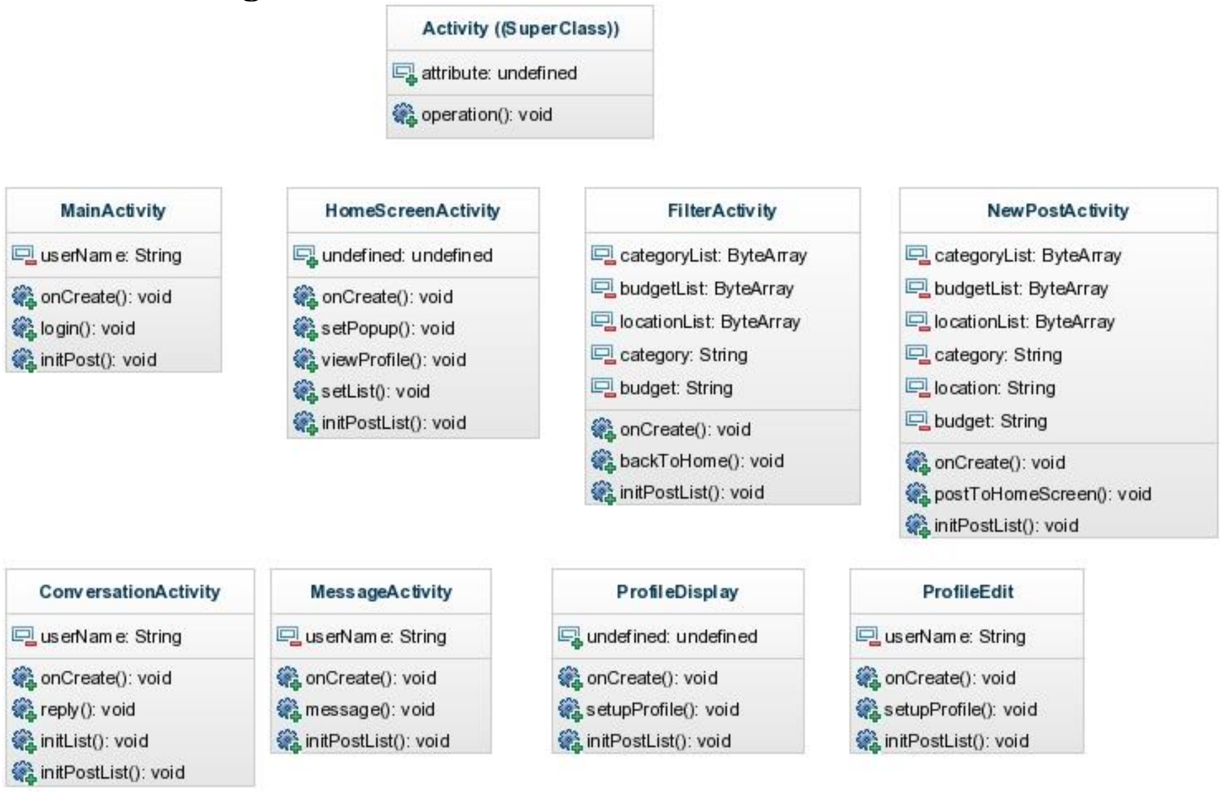
App Description
Our goal is to improve shopping outcomes by leveraging technology and the power of human connections to enhance the shopping experience. We offer a mobile application to help connect people across their favorite shopping activities; this can include shopping for clothes, antiques, music, food and other related shopping experiences. We want to fundamentally change the way people connect and provide a platform to make that process easier.

Target User
Our solution is being developed to target a variety of demographics. With that said, our target market is English speaking countries and in particular women in the United States. According to the US Census there are 57 million women from the ages of 18 to 44. We want to enhance their shopping experience by connecting them with someone in their community to share their shopping experience.

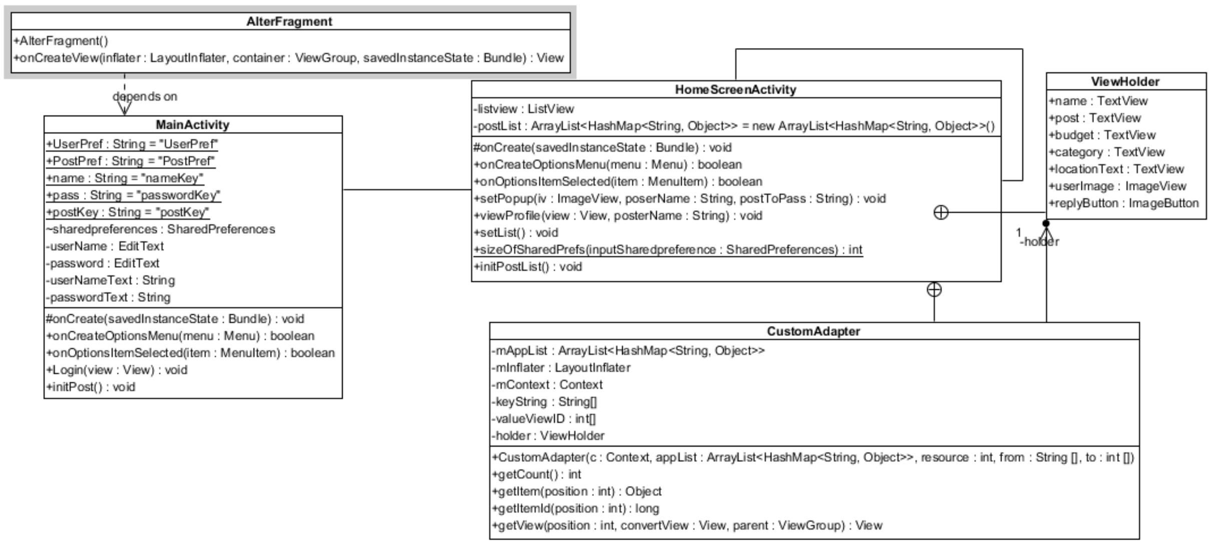
Importance
Life is all about the experiences. New experiences whether at home or while traveling enhance the moment and make it memorable. We want to make people's lives better by connecting them with others who want to share in some shopping experience. By connecting via an online platform we can make life more dynamic and enriching – not only can we help improve the overall shopping experience but also help people forge meaningful relationships.

Why us
Our team has the right mix of leadership, technical and operational experience to move this project forward. Michael is a Navy Veteran and has experience leading startups. Yuheng has many years of software development experience; she wrote her first line of code at 12. Nick has a background in interaction design and front-end development. Finally we can leverage the resources at the University of Michigan – Ann Arbor including the School of Information Entrepreneurship Program, the Center for Entrepreneurship and the Zell Lurie Entrepreneurship Center to build our platform.

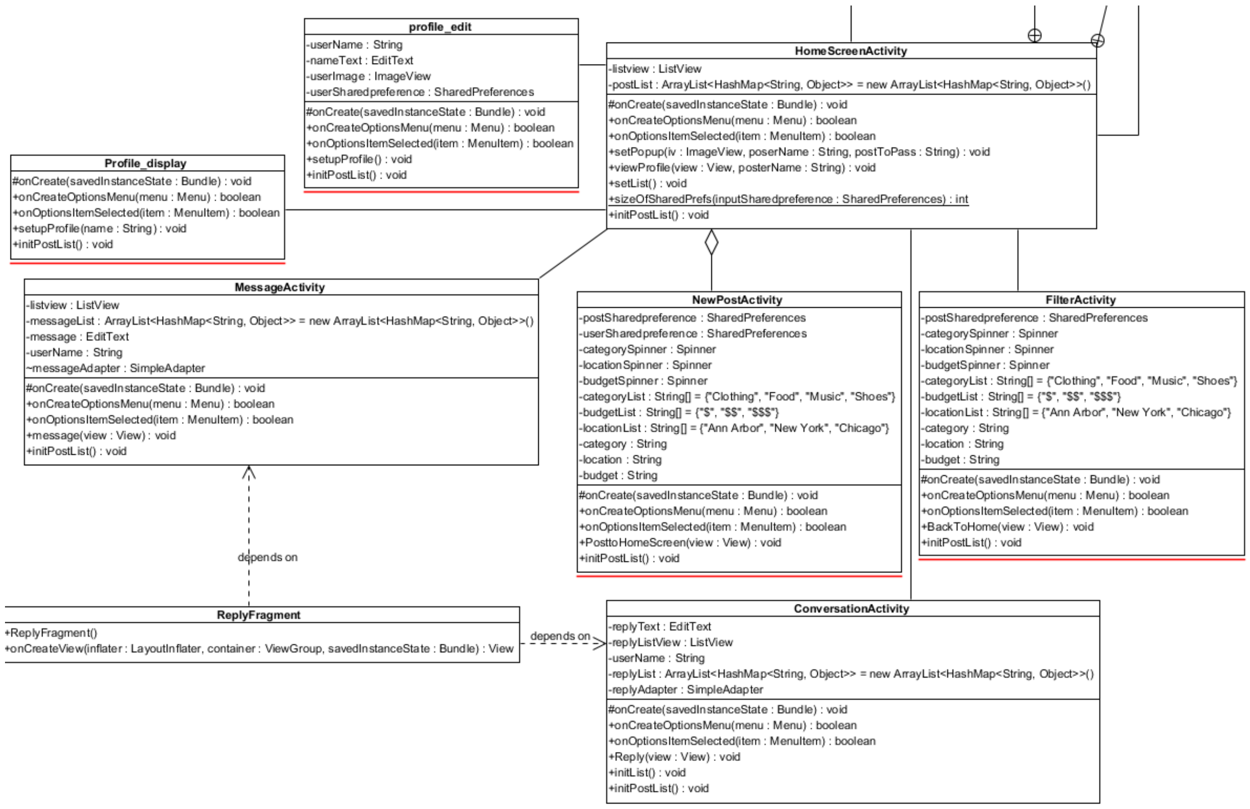
## UML Class Diagram



Because of space limit, I seperated the UML into two parts, the first part is the main activity and homescreen activity. The second part is the homescreen activity and all the other activities.

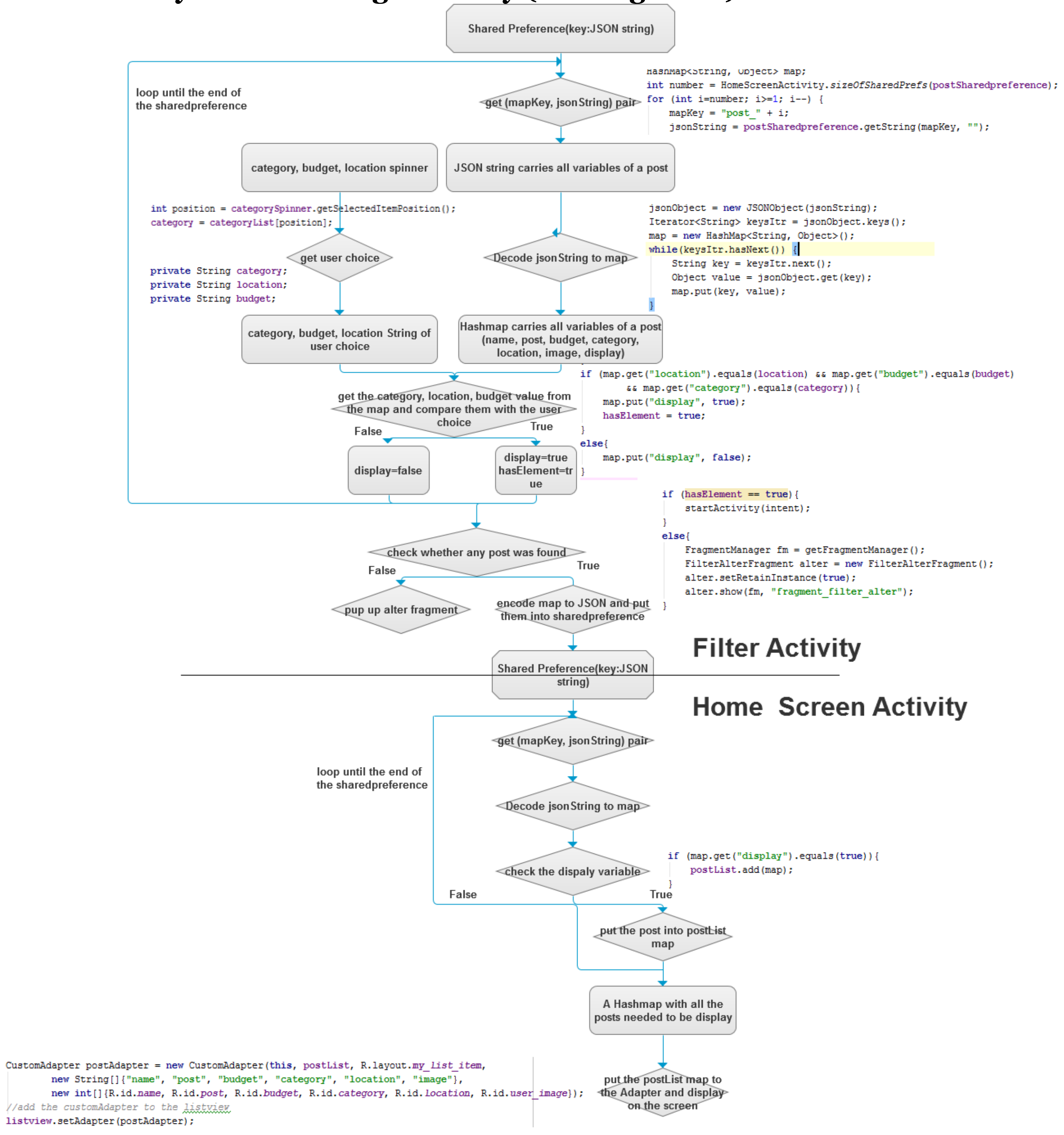

AlterFragment
+AlterFragment()
+onCreateView(inflater : LayoutInflater, container : ViewGroup, savedInstanceState : Bundle) : View
depends on
MainActivity
+UserPref : String = "UserPref"
+PostPref : String = "PostPref"
+name : String = "nameKey"
+pass : String = "passwordKey"
+postKey : String = "postKey"
~sharedpreferences : SharedPreferences
-userName : EditText
-password : EditText
-userNameText : String
-passwordText : String
#onCreate(savedInstanceState : Bundle) : void
+onCreateOptionsMenu(menu : Menu) : boolean
+onOptionsItemSelected(item : MenuItem) : boolean
+Login(view : View) : void
+initPost() : void
HomeScreenActivity
-listview : ListView
-postList : ArrayList<HashMap<String, Object>> = new ArrayList<HashMap<String, Object>>()
#onCreate(savedInstanceState : Bundle) : void
+onCreateOptionsMenu(menu : Menu) : boolean
+onOptionsItemSelected(item : MenuItem) : boolean
+setPopup(iv : ImageView, poserName : String, postToPass : String) : void
+viewProfile(view : View, posterName : String) : void
+setList() : void
+sizeOfSharedPrefs(inputSharedpreference : SharedPreferences) : int
+initPostList() : void
ViewHolder
+name : TextView
+post : TextView
+budget : TextView
+category : TextView
+locationText : TextView
+userImage : ImageView
+replyButton : ImageButton
1
-holder
CustomAdapter
-mAppList : ArrayList<HashMap<String, Object>>
-mInflater : LayoutInflater
-mContext : Context
-keyString : String[]
-valueViewID : int[]
-holder : ViewHolder
+CustomAdapter(c : Context, appList : ArrayList<HashMap<String, Object>>, resource : int, from : String [], to : int [])
+getCount() : int
+getItem(position : int) : Object
+getItemId(position : int) : long
+getView(position : int, convertView : View, parent : ViewGroup) : View

Detailed UML(1st part)


profile_edit
-userName : String
-nameText : EditText
-userImage : ImageView
-userSharedpreference : SharedPreferences
#onCreate(savedInstanceState : Bundle) : void
+onCreateOptionsMenu(menu : Menu) : boolean
+onOptionsItemSelected(item : MenuItem) : boolean
+setupProfile() : void
+initPostList() : void
HomeScreenActivity
-listview : ListView
-postList : ArrayList<HashMap<String, Object>> = new ArrayList<HashMap<String, Object>>()
#onCreate(savedInstanceState : Bundle) : void
+onCreateOptionsMenu(menu : Menu) : boolean
+onOptionsItemSelected(item : MenuItem) : boolean
+setPopup(iv : ImageView, poserName : String, postToPass : String) : void
+viewProfile(view : View, posterName : String) : void
+setList() : void
+sizeOfSharedPrefs(inputSharedpreference : SharedPreferences) : int
+initPostList() : void
Profile_display
#onCreate(savedInstanceState : Bundle) : void
+onCreateOptionsMenu(menu : Menu) : boolean
+onOptionsItemSelected(item : MenuItem) : boolean
+setupProfile(name : String) : void
+initPostList() : void
MessageActivity
-listview : ListView
-messageList : ArrayList<HashMap<String, Object>> = new ArrayList<HashMap<String, Object>>()
-message : EditText
-userName : String
~messageAdapter : SimpleAdapter
#onCreate(savedInstanceState : Bundle) : void
+onCreateOptionsMenu(menu : Menu) : boolean
+onOptionsItemSelected(item : MenuItem) : boolean
+message(view : View) : void
+initPostList() : void
NewPostActivity
-postSharedpreference : SharedPreferences
-userSharedpreference : SharedPreferences
-categorySpinner : Spinner
-locationSpinner : Spinner
-budgetSpinner : Spinner
-categoryList : String[] = {"Clothing", "Food", "Music", "Shoes"}
-budgetList : String[] = {"$", "$$", "$$$"}
-locationList : String[] = {"Ann Arbor", "New York", "Chicago"}
-category : String
-location : String
-budget : String
#onCreate(savedInstanceState : Bundle) : void
+onCreateOptionsMenu(menu : Menu) : boolean
+onOptionsItemSelected(item : MenuItem) : boolean
+PosttoHomeScreen(view : View) : void
+initPostList() : void
FilterActivity
-postSharedpreference : SharedPreferences
-categorySpinner : Spinner
-locationSpinner : Spinner
-budgetSpinner : Spinner
-categoryList : String[] = {"Clothing", "Food", "Music", "Shoes"}
-budgetList : String[] = {"$", "$$", "$$$"}
-locationList : String[] = {"Ann Arbor", "New York", "Chicago"}
-category : String
-location : String
-budget : String
#onCreate(savedInstanceState : Bundle) : void
+onCreateOptionsMenu(menu : Menu) : boolean
+onOptionsItemSelected(item : MenuItem) : boolean
+BackToHome(view : View) : void
+initPostList() : void
depends on
ReplyFragment
+ReplyFragment()
+onCreateView(inflater : LayoutInflater, container : ViewGroup, savedInstanceState : Bundle) : View
depends on
ConversationActivity
-replyText : EditText
-replyListView : ListView
-userName : String
-replyList : ArrayList<HashMap<String, Object>> = new ArrayList<HashMap<String, Object>>()
-replyAdapter : SimpleAdapter
#onCreate(savedInstanceState : Bundle) : void
+onCreateOptionsMenu(menu : Menu) : boolean
+onOptionsItemSelected(item : MenuItem) : boolean
+Reply(view : View) : void
+initList() : void
+initPostList() : void

Detailed UML(2rd part)

## Functionality of the Message Activity (Yuheng Chen)


Shared Preference(key:JSON string)
loop until the end of the sharedpreference
get (mapKey, jsonString) pair
category, budget, location spinner
JSON string carries all variables of a post
get user choice
Decode jsonString to map
category, budget, location String of user choice
Hashmap carries all variables of a post (name, post, budget, category, location, image, display)
get the category, location, budget value from the map and compare them with the user choice
False
True
display=false
display=true hasElement=true
check whether any post was found
False
True
pup up alter fragment
encode map to JSON and put them into sharedpreference
Filter Activity
Shared Preference(key:JSON string)
Home Screen Activity
get (mapKey, jsonString) pair
loop until the end of the sharedpreference
Decode jsonString to map
check the dispaly variable
False
True
put the post into postList map
A Hashmap with all the posts needed to be display
put the postList map to the Adapter and display on the screen

```java
HashMap<String, Object> map;
int number = HomeScreenActivity.sizeOfSharedPrefs(postSharedpreference);
for (int i=number; i>=1; i--) {
    mapKey = "post_" + i;
    jsonString = postSharedpreference.getString(mapKey, "");
```

```java
int position = categorySpinner.getSelectedItemPosition();
category = categoryList[position];
```

```java
private String category;
private String location;
private String budget;
```

```java
jsonObject = new JSONObject(jsonString);
Iterator<String> keysItr = jsonObject.keys();
map = new HashMap<String, Object>();
while(keysItr.hasNext()) {
    String key = keysItr.next();
    Object value = jsonObject.get(key);
    map.put(key, value);
}
```

```java
if (map.get("location").equals(location) && map.get("budget").equals(budget)
        && map.get("category").equals(category)){
    map.put("display", true);
    hasElement = true;
}
else{
    map.put("display", false);
}
```

```java
if (hasElement == true){
    startActivity(intent);
}
else{
    FragmentManager fm = getFragmentManager();
    FilterAlterFragment alter = new FilterAlterFragment();
    alter.setRetainInstance(true);
    alter.show(fm, "fragment_filter_alter");
}
```

```java
if (map.get("display").equals(true)){
    postList.add(map);
}
```

```java
CustomAdapter postAdapter = new CustomAdapter(this, postList, R.layout.my_list_item,
        new String[]{"name", "post", "budget", "category", "location", "image"},
        new int[]{R.id.name, R.id.post, R.id.budget, R.id.category, R.id.location, R.id.user_image});
//add the customAdapter to the listview
listview.setAdapter(postAdapter);
```

In this application, we hope to have a functionality which can let the users choose the posts meet their requirement. Thus, I implement the filter functionality.
In this application, I encode every post into a JSON string and store them into the shared preferences. In the filter activity, I will first retrieve every post and decode them into Hash map again.  I added a “display” (default true) key to the Hash map of each post.
I will then retrieve the user input from the location, budget and category spinner. The “display” value of the posts with the required location, post and budget will be set to “true”, while all the others will be set to “false”.

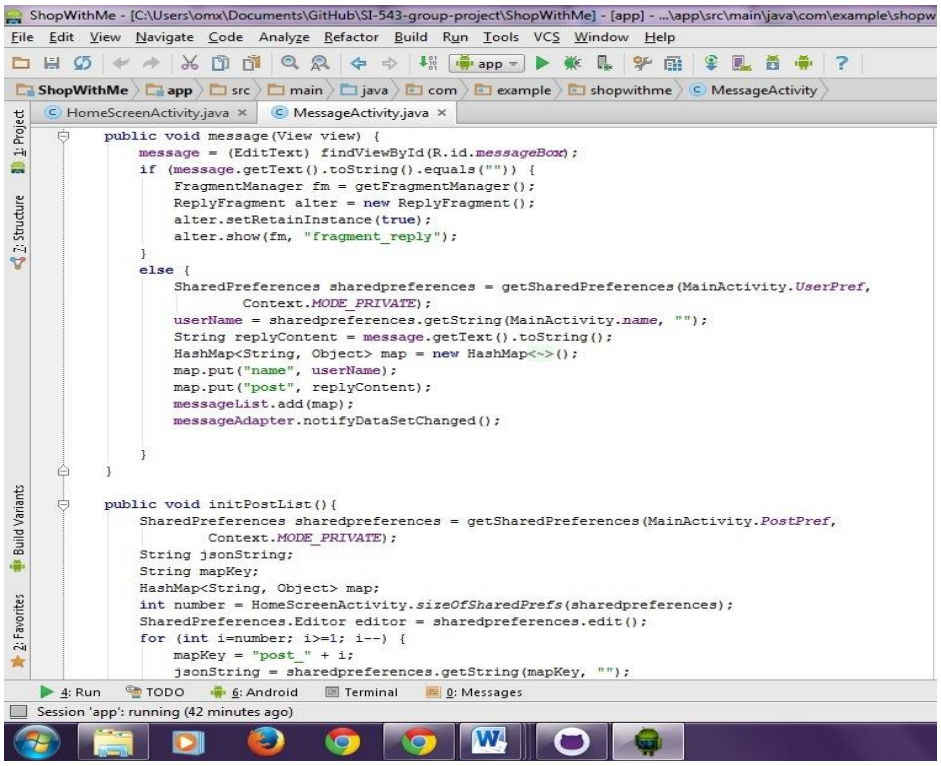
In the home screen activity, only the posts with “display” set to true will be put into the post list. I also write a function “initList” to set all the “display” of all posts to “true” to deactivate the filtering function after one successful filtering.
I set up a “hasElement” variable to check whether any post was found in the filtering. If the value is still false after searching the whole map, it will pop up an alert to tell the user “No result found and try again”.

## Functionality of the Message Activity (Michael Collins)

Our system connects people to shop together. We want people, who are new to an area or simply want to meet new people, to have a platform to connect and collaborate on a common interest – shopping. There are a variety of activities in our application; they include the conversation, filter, home screen, main, message, new post, profile display and profile edit.

I worked on a variety of activities including the message and conversation activity. The message activity allows a user to contact to a user posting; this area is a private two way chat area that can be used to share personal information that users do not want to made public such as phone number. The conversation activity allows users to reply to user posting; this is a public multi chat area that can be user to talk generally about a shopping event and get additional details that may be helpful in deciding if different users want to shop together.

In this write-up I will focus on the message page and elaborate on the code. First, the username that was entered during login was associated with the user via sharedpreferences; this is a critical first step as it allows a username to correctly associate with a conversation reply. After that a HashMap was setup that basically allows me to store a certain value with a variable. In this, case it would be a user name and a user posting so when a user reply the message is posted with both the correct name and the message entered; the code is listed below.


ShopWithMe - [C:\Users\omx\Documents\GitHub\SI-543-group-project\ShopWithMe] - [app] - ...\app\src\main\java\com\example\shopw
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
app
ShopWithMe app src main java com example shopwithme MessageActivity
HomeScreenActivity.java
MessageActivity.java
1: Project
7: Structure
Build Variants
2: Favorites
4: Run TODO 6: Android Terminal 0: Messages

```java
public void message(View view) {
    message = (EditText) findViewById(R.id.messageBox);
    if (message.getText().toString().equals("")) {
        FragmentManager fm = getFragmentManager();
        ReplyFragment alter = new ReplyFragment();
        alter.setRetainInstance(true);
        alter.show(fm, "fragment_reply");
    }
    else {
        SharedPreferences sharedpreferences = getSharedPreferences(MainActivity.UserPref,
                Context.MODE_PRIVATE);
        userName = sharedpreferences.getString(MainActivity.name, "");
        String replyContent = message.getText().toString();
        HashMap<String, Object> map = new HashMap<~>();
        map.put("name", userName);
        map.put("post", replyContent);
        messageList.add(map);
        messageAdapter.notifyDataSetChanged();

    }
}

public void initPostList(){
    SharedPreferences sharedpreferences = getSharedPreferences(MainActivity.PostPref,
            Context.MODE_PRIVATE);
    String jsonString;
    String mapKey;
    HashMap<String, Object> map;
    int number = HomeScreenActivity.sizeOfSharedPrefs(sharedpreferences);
    SharedPreferences.Editor editor = sharedpreferences.edit();
    for (int i=number; i>=1; i--) {
        mapKey = "post_" + i;
        jsonString = sharedpreferences.getString(mapKey, "");
```

Session 'app': running (42 minutes ago)

## Functionality of the ReplyFragment (Nick Malzahn)

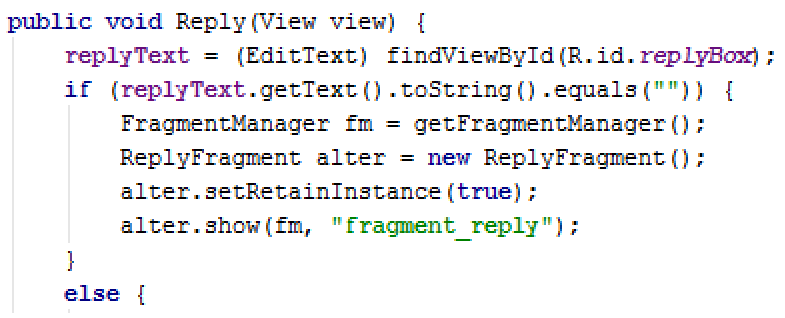
For the ConversationActivity class, we wanted to create an alert dialog fragment that appears when a user tries to submit a reply without any content. Therefore, I first created a layout file called fragment_reply.xml and coded a TextView object. I used 'android:text="Please check your input"' to establish the message users receive when they cause the alert dialog to appear.

Next, I created the ReplyFragment class and extended the DialogFragment class in order to make this fragment function as a dialog. Within the ReplyFragment class, I implemented the onCreateView method and wrote

```java
return inflater.inflate(R.layout.fragment_reply, container, false);
```

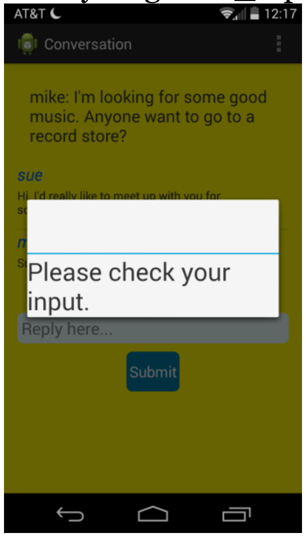
in order to inflate the fragment_reply layout file.

Lastly, in the ConversationActivity class, I instantiated a new ReplyFragment object with a conditional statement when calling the Reply method:



```
public void Reply(View view) {
    replyText = (EditText) findViewById(R.id.replyBox);
    if (replyText.getText().toString().equals("")) {
        FragmentManager fm = getFragmentManager();
        ReplyFragment alter = new ReplyFragment();
        alter.setRetainInstance(true);
        alter.show(fm, "fragment_reply");
    }
    else {
```

“replyText” is the variable for what the user types into the EditText object called “replyBox.” The “if” statement takes the “replyText” variable and gets the text from it, changes the text to a string, and recognizes the string as empty. Therefore, if this statement is found to be true, I call the FragmentManager method to be able to interact with ReplyFragment object and then create a new ReplyFragment object called “alter.” I set “alter” to retain its instance across the entire activity and then have it finally show itself, maintaining the view described by fragment_reply.xml:



mike: I'm looking for some good music. Anyone want to go to a record store?

sue

Hi. I'd really like to meet up with you for

Please check your input.