Name: Cameron L'Ecuyer  
Class ID: 17  
Email: [cjlth5@mail.umkc.edu](mailto:cjlth5@mail.umkc.edu)  
Professor: Yugyung Lee  
[MyGitHub](https://github.com/camlecuyer/CS5551_11-1_LabAssignments)

Technical Partner: Name: Sneha Mishra  
Class ID: 21  
Email: [smccr@mail.umkc.edu](mailto:smccr@mail.umkc.edu)  
[GitHub](https://github.com/SnehaMishra28/CS5551_SnehaMishra_labassignments/wiki/CS5551_(11_1)_LabAssignment_lab%233)

**Objective**

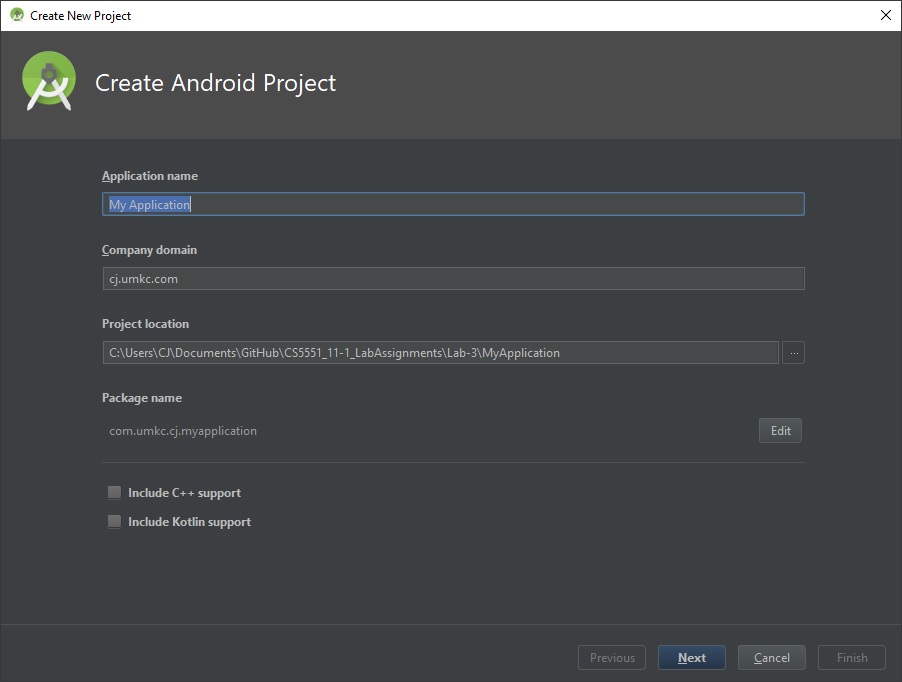
Create an Android app that possesses a login, a register page, and a home page that uses an Android feature and a knowledge service. The app also makes use of Firebase for OAuth services for login, and for real-time storage.

**Features**

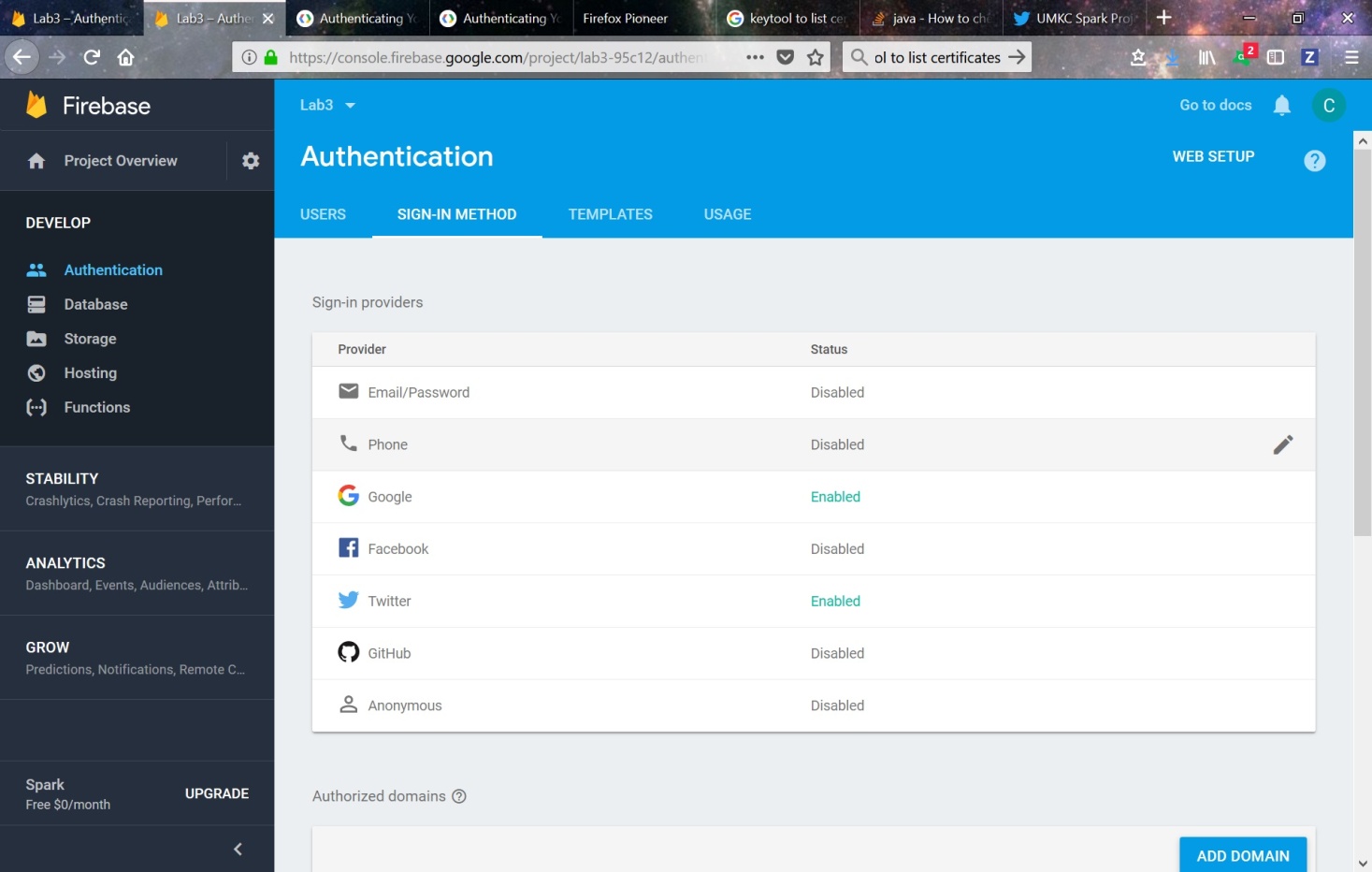
* Login page that uses OAuth
* Register page
* Use of the camera on an Android device
* Use of Watson's visual recognition software to identify what is in an image
* Firebase real-time storage

**Steps**

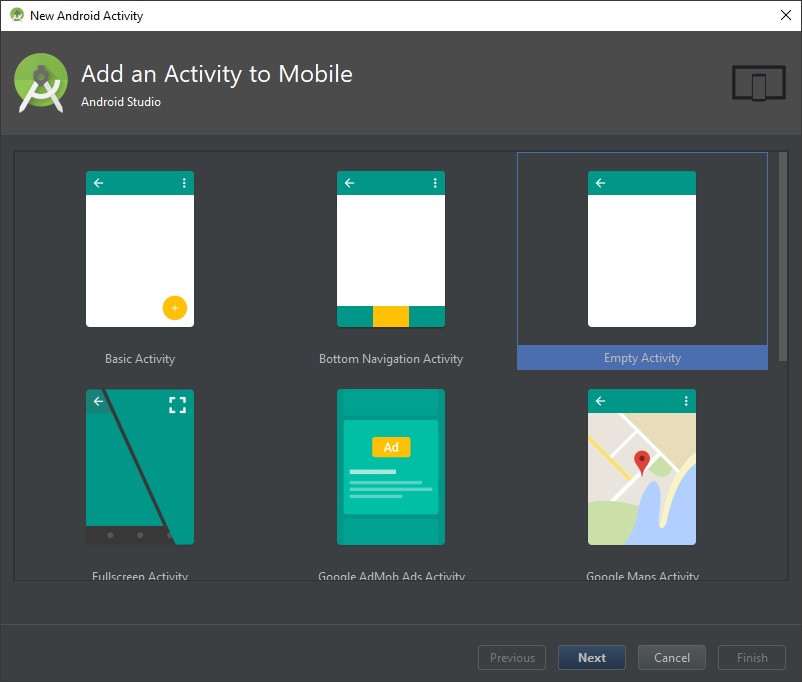
**Step 1 - Create an Android app**

This image is the creation of an Android app 

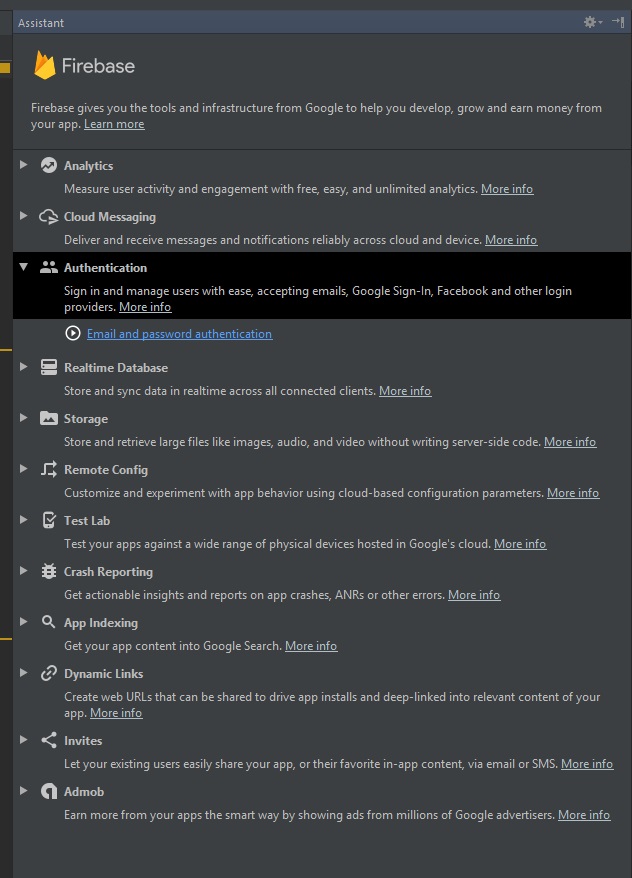
**Step 2 - Register for Firebase and connect your Android app and Twitter app**

This image is the tab in Firebase to access Authentication section 

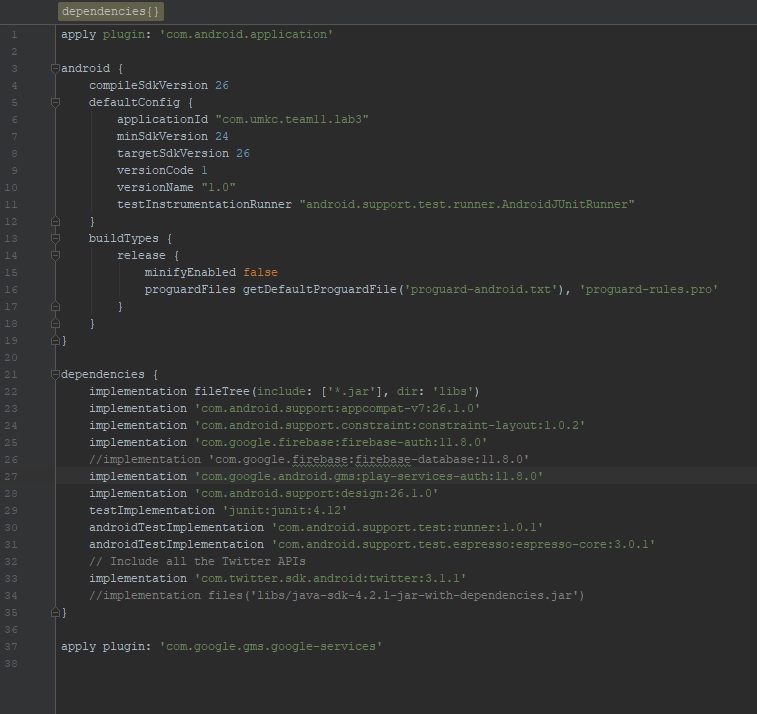
**Step 3 - Add activities to the app**

This image is the selection of an activity 

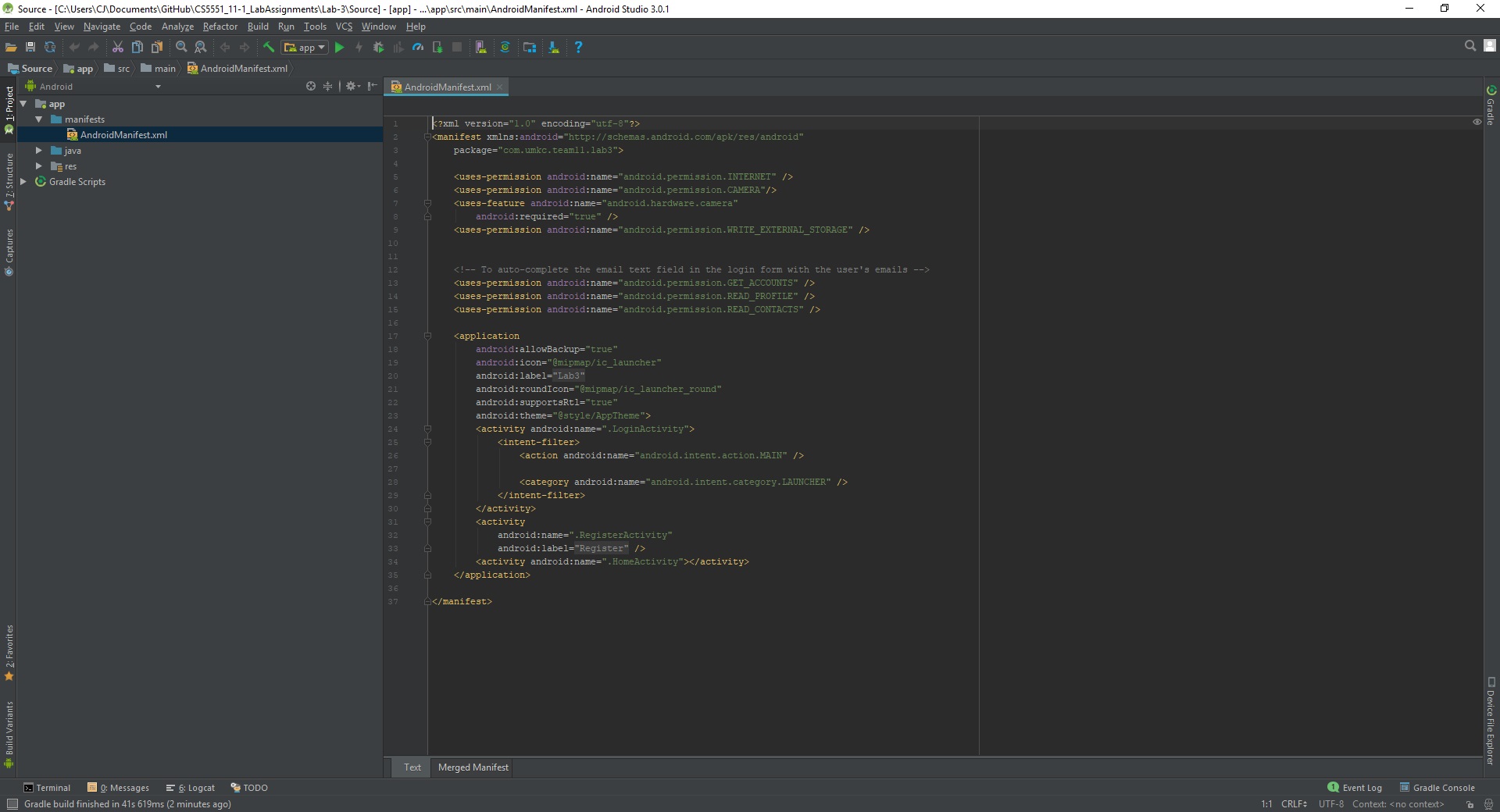
**Step 4 - Activate Firebase in Android**

This image is the path to activate the Authentication portion of Firebase in Android 

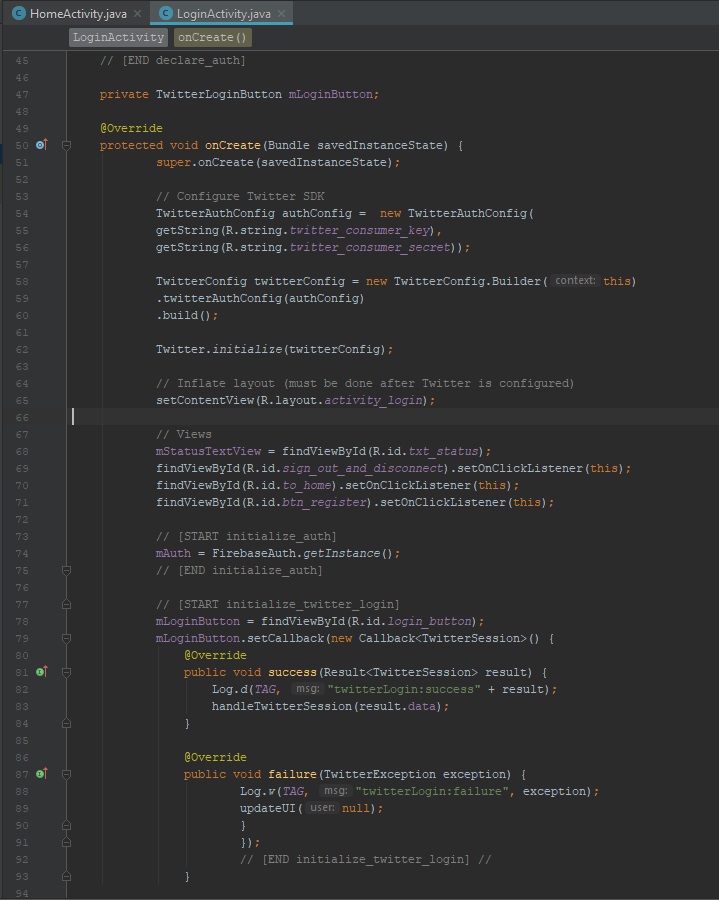
**Step 5 - Add dependencies for Twitter, Firebase, and Watson Visual Recognition**

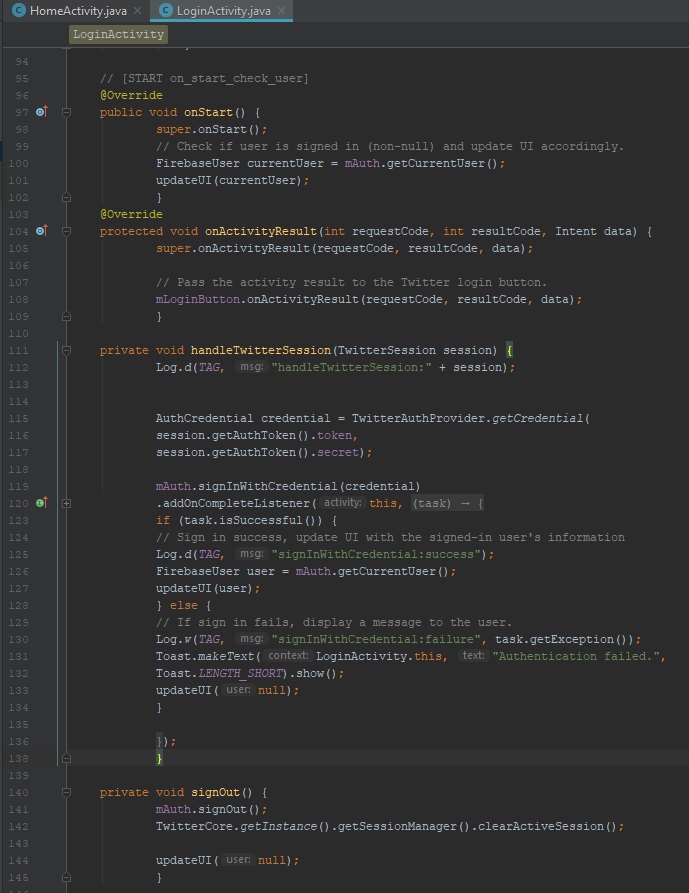
This image shows the dependencies of the app 

**Step 6 - Add permissions to the Manifest**

This image shows the updated Manifest 

**Step 7 - Add code to the Login Page to use Twitter OAuth**

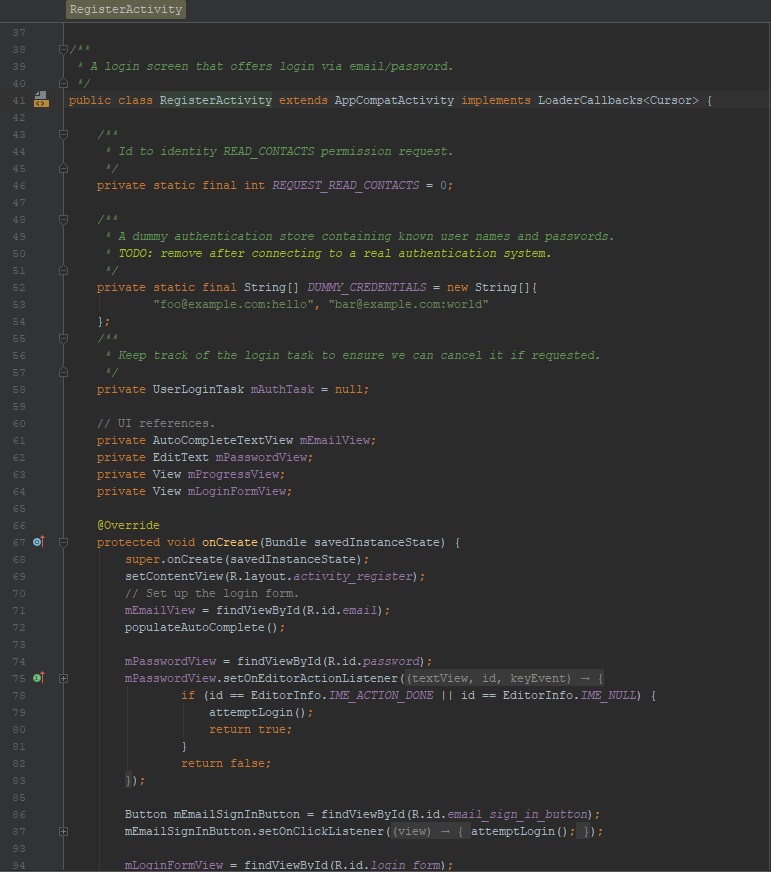
This image is the first part of the code for the Login page, it shows the creation of a link to Twitter, the listener for Twitter login button, and the activation of the Twitter Login button 

This image is more of the code for the login page, this part shows the handler for the Twitter button, the sign out function, and the call to get Firebase user 

[The full code for the login page](https://github.com/camlecuyer/CS5551_11-1_LabAssignments/blob/master/Lab-3/Source/app/src/main/java/com/umkc/team11/lab3/LoginActivity.java)

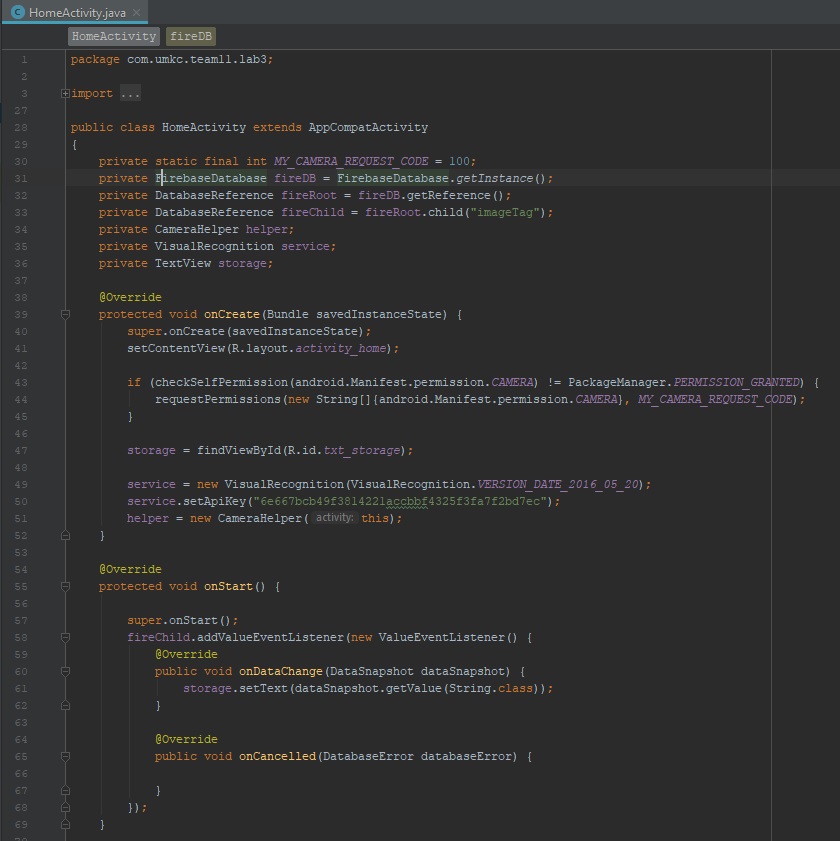
**The login code is based off the Firebase Twitter GitHub example, I tried to implement based on their walk through but it did not work. I copied their code from their GitHub into the app, changed some variables and it worked fine.**  
[The link to the Firebase Twitter example](https://github.com/firebase/quickstart-android/blob/master/auth/app/src/main/java/com/google/firebase/quickstart/auth/TwitterLoginActivity.java#L109-L141)

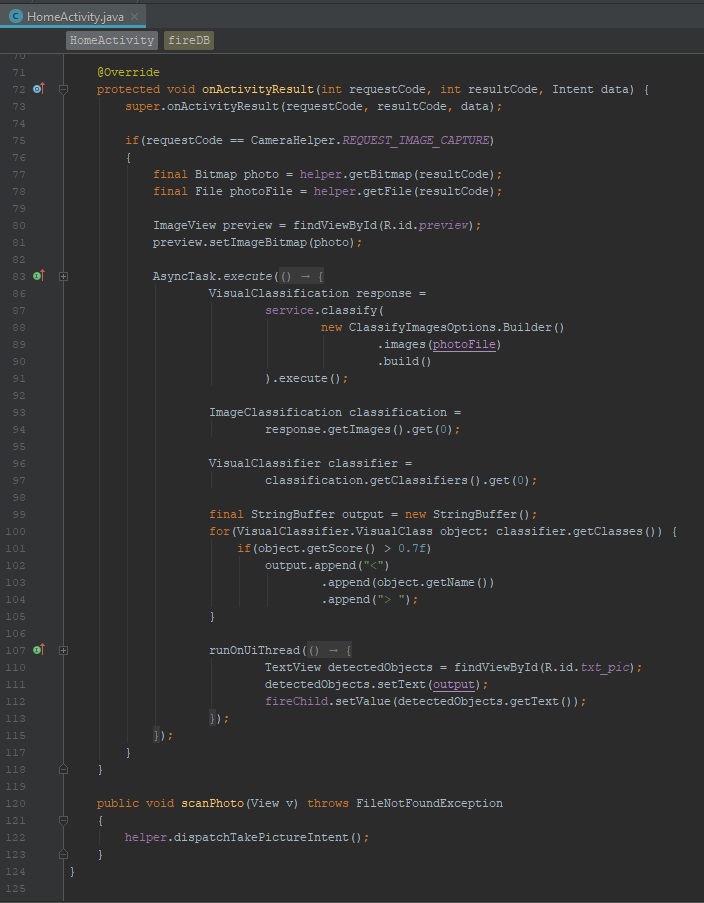
**Step 8 - Create Register activity and add code**

This code is generated by default when selecting the login activity in the Android examples, small changes were made to make it a register activity 

[This is the full code for Register](https://github.com/camlecuyer/CS5551_11-1_LabAssignments/blob/master/Lab-3/Source/app/src/main/java/com/umkc/team11/lab3/RegisterActivity.java)

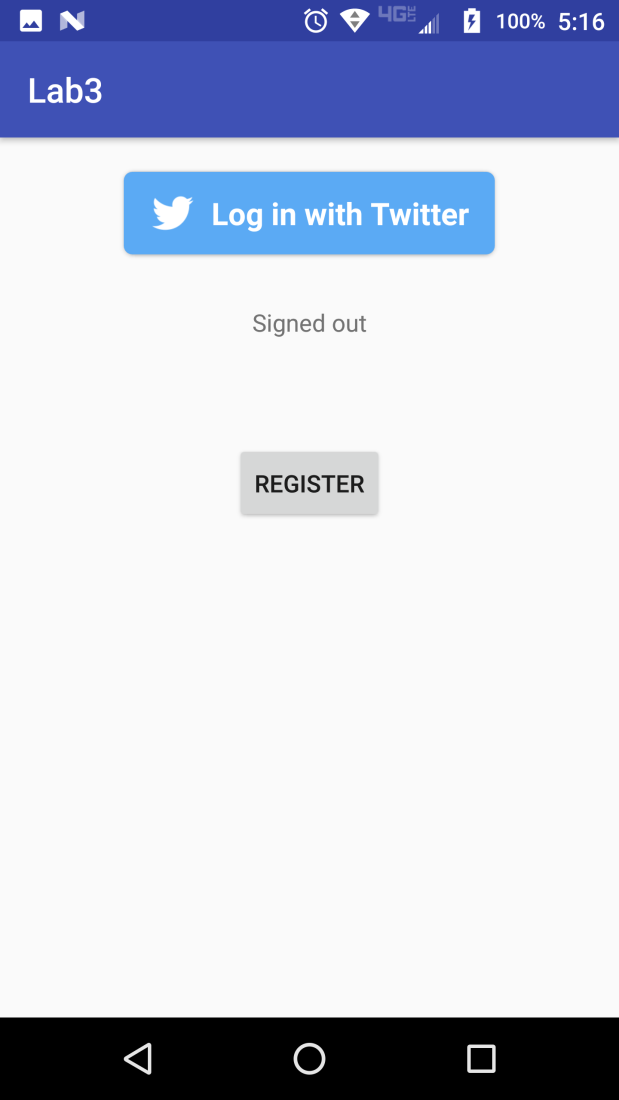
**Step 9 - Create Home activity and add code**

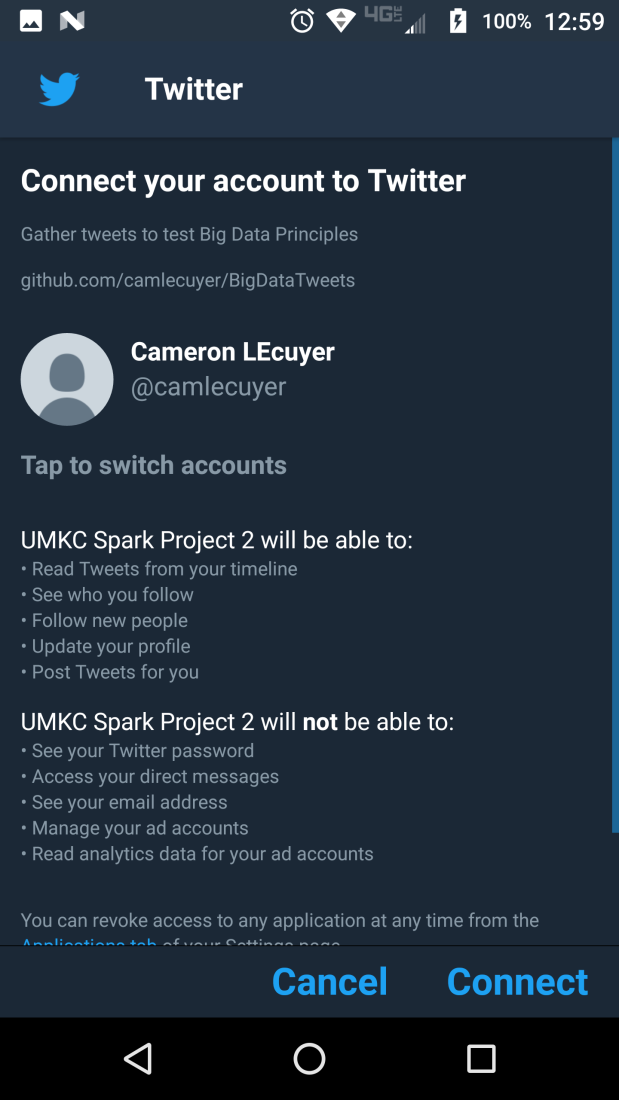
This image is the first half of the home activity code. It shows the setup for the Firebase storage, and the call to initialize Watson's visual recognition. It also shows the creation of the Firebase storage listener and the permissions for using the camera 

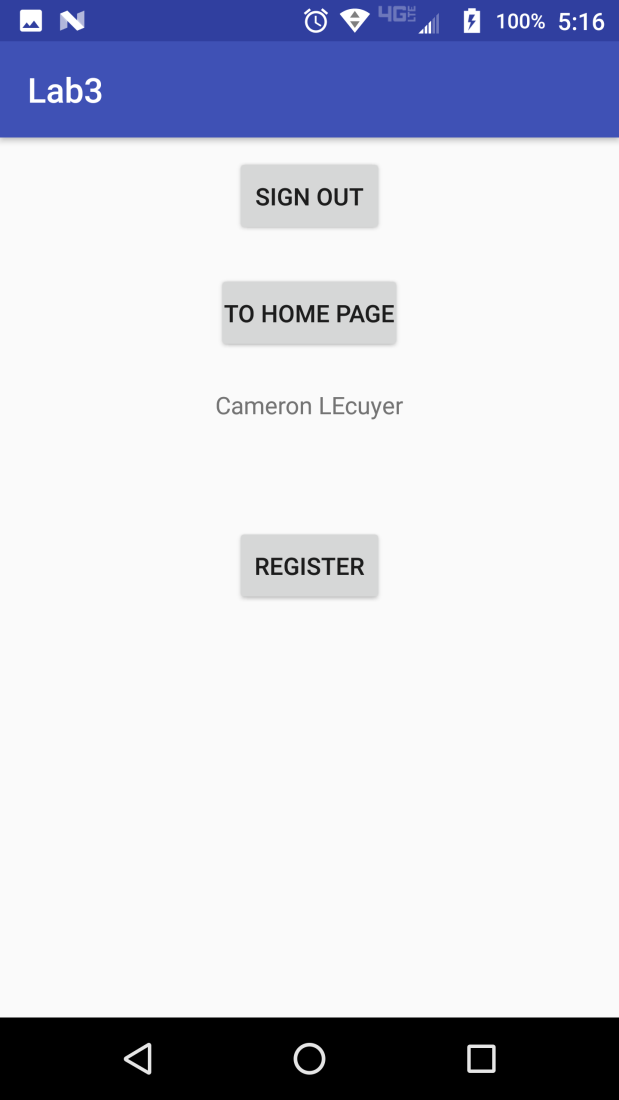
This image is the rest of the code for the home activity. It shows the function that calls the camera using Watson's cameraHelper, the call to take and create the photo, and the asynchronous call to get the message from Watson. After Watson returns the call, a text field on the home page is populated, and the data is saved to the Firebase storage which will update a separate text field on the home page when a change is detected in the storage 

[This is the full code for Home](https://github.com/camlecuyer/CS5551_11-1_LabAssignments/blob/master/Lab-3/Source/app/src/main/java/com/umkc/team11/lab3/HomeActivity.java)

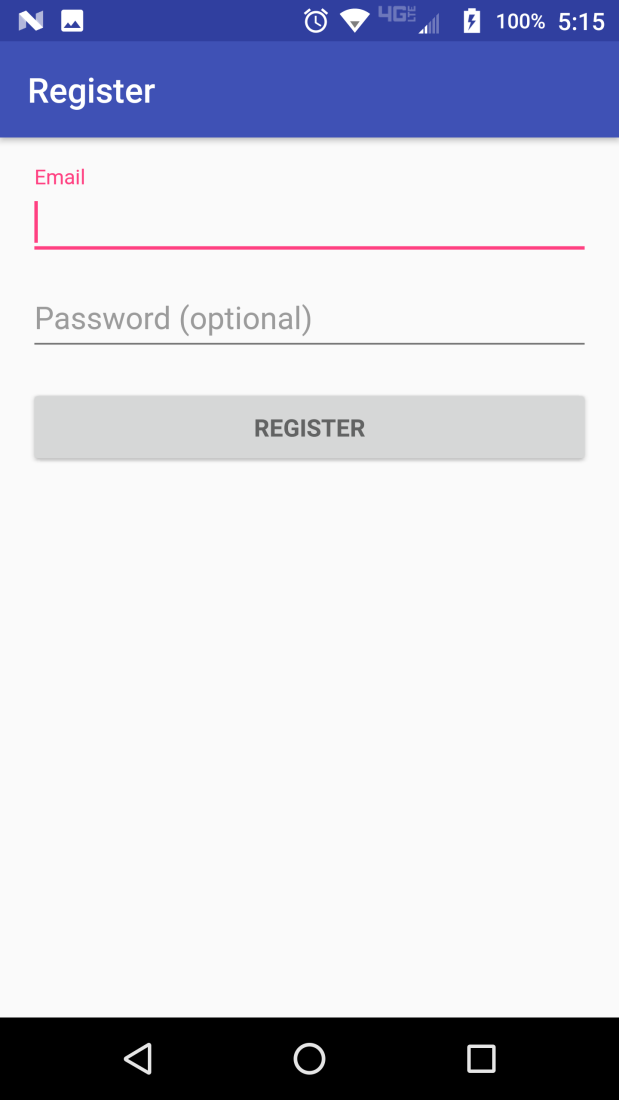
**Results**

This image shows the page that displays when the app starts 

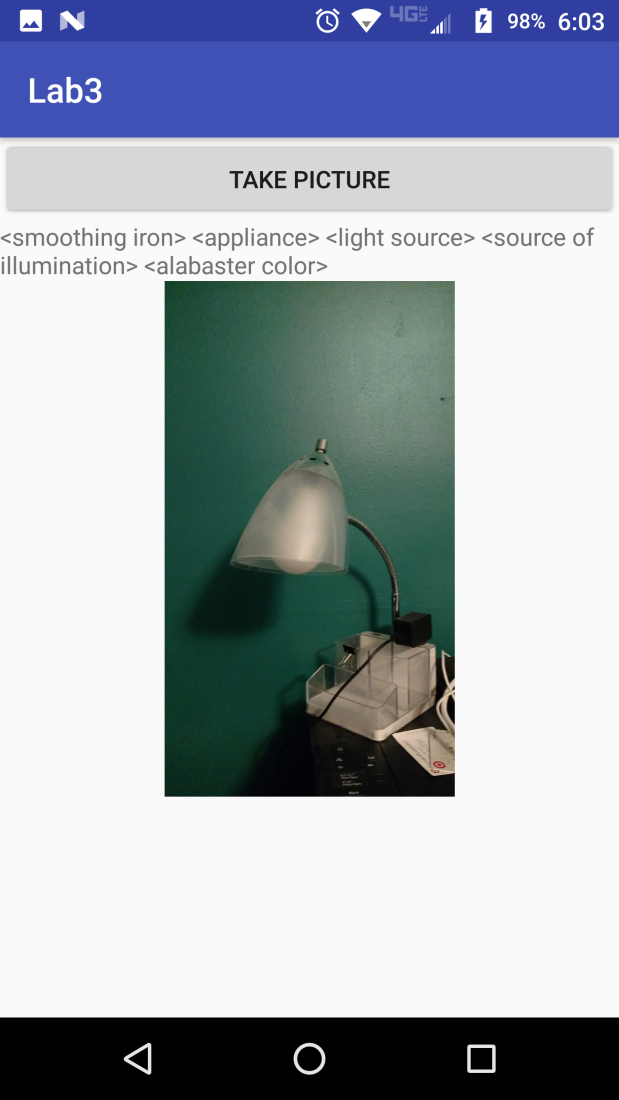
This image shows what happens after the Twitter button is pushed 

This image shows the screen after accepting the Twitter connection 

This image shows the Register page



This image shows the Home page before taking a picture 

This image shows the Home page after Watson returns a result 

This image shows the Home page after Watson returns a result, and displays the Firebase storage results



**References**

This link helped with the Twitter SDK code that ended up not working: <https://github.com/twitter/twitter-kit-android/wiki/Getting-Started>  
This link helped with setting up Firebase Twitter OAuth, the code ended up not working: <https://firebase.google.com/docs/auth/android/twitter-login>  
This link is to the GitHub example of Firebase's Twitter OAuth example, it did work: <https://github.com/firebase/quickstart-android/blob/master/auth/app/src/main/java/com/google/firebase/quickstart/auth/TwitterLoginActivity.java#L109-L141>  
This link helped set up Watson's visual recognition: <https://code.tutsplus.com/tutorials/using-the-ibm-watson-visual-recognition-api-in-android-apps--cms-29542>  
These videos helped set up Firebase's real-time storage:  
<https://www.youtube.com/watch?v=QETnC6SEwa0&feature=youtu.be>  
<https://www.youtube.com/watch?v=1qoR9XnWRBc&t=953s>