_____
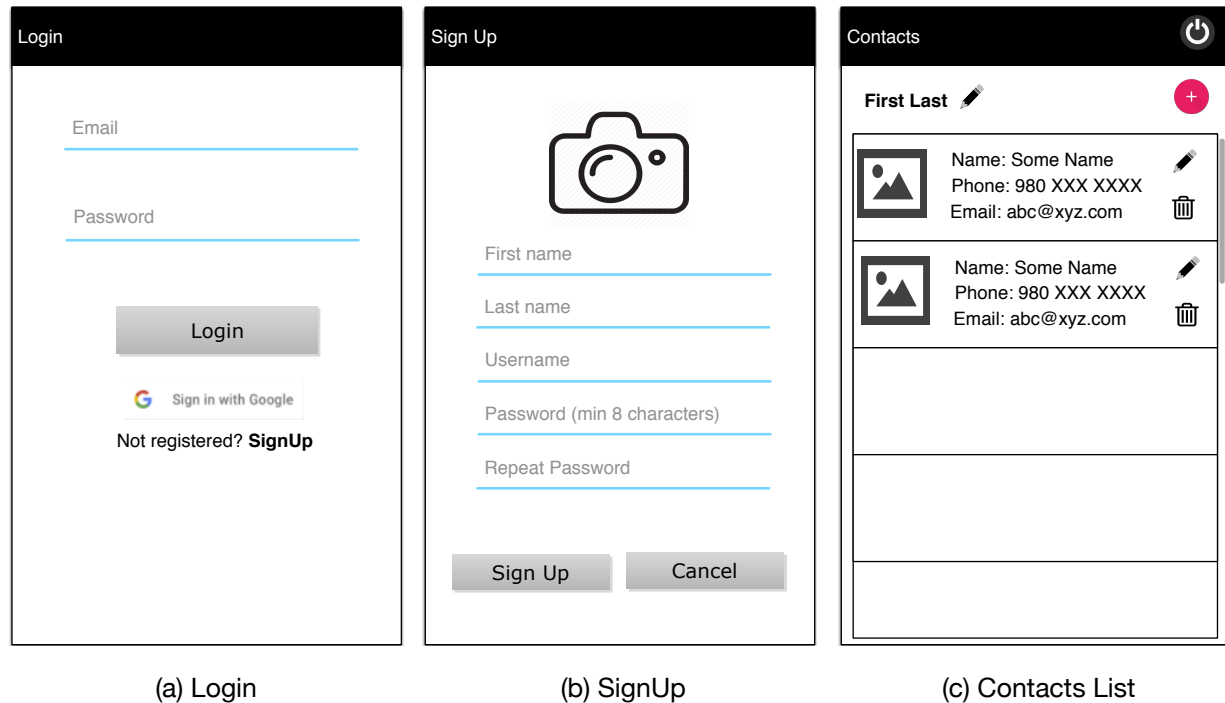
Basic Instructions:

1. In every file submitted you MUST place the following comments:
    a. Assignment #.
    b. File Name.
    c. Full name of each person in your group.
2. Each team is required to submit the assignment on Canvas.
3. Please download the support files provided with this assignment and use them when implementing your project.
4. Submission details:
    a. Compress the contents of your project folder. The file name is very important and should follow the following format: **InClass11_Group#.zip**
    b. Only one group member is required to submit on behalf of the whole group.
    c. You should submit the assignment through Canvas: Submit the zip file.
5. The required Android Virtual Device (AVD) should have **minimum SDK version set to 20 and target SDK at 25**.
6. **Failure to follow the above instructions will result in point deductions.**

In Class Assignment 10 (100 points)

In this assignment you will be building an application that uses Firebase to manage he user contacts. You are free to use either fragments or activities.

| (a) Login | (b) SignUp | (c) Contacts List |

Figure 1, App Wireframes.

Part 1: Login (10 points)

This is the launcher screen of you app. The wireframe is shown in Figure 1(a). The requirements are as follows:

1. The user should provide their email and password. The provided credentials should be used to authenticate the user using firebase login (use only Email and Password authentication). Clicking the "Login" button should submit the login information to firebase to verify the user's credentials.
   a) If the user is successfully logged in then start the Contacts List Screen, and finish the Login Screen.
   b) If the user is not successfully logged in, then show a toast message indicating that the login was not successful.
2. Clicking the "Sign Up" button should start the Signup Screen Figure 1(b), and finish the login Screen.
3. "Sign in with Google" should be added.
   a) You have to add every other details by editing the profile.

Part 2: SignUp (25 points)

Create the Signup screen to match Figure 1(b), with the following requirements:

1. Clicking the "Cancel" button should finish the Signup Screen and start the Login Screen.

2. There should be a Image Button to enable user for getting a profile photo.
   a) Use implicit intent to use either Camera or Gallery to get the image.
3. The user should provide their first name, last name, email, password and password confirmation. Preform the required validation(the given password and the repeated password must match). Clicking the "Sign Up" button should submit the user's information to firebase signup.
   a) If the signup is not successful display an error message indicating the error message received from firebase.
   b) If the signup is successful, then display a Toast indicating that the user has been created. Then start the Contacts List Screen and finish the Signup Screen.

Part 3, Contacts List (40 Points):
The contact list should display a ListView or RecyclerView showing the list of contacts retrieved from firebase as showing in Figure 1(c). Below are the requirements:
1. Retrieve the contacts stored for the currently logged in user from firebase. Note that this is a multi user app and each user should have access to only their stored contacts.
2. The list of contacts should be displayed using a ListView or RecyclerView as shown in Figure 1(c).
3. Clicking on the Edit icon should enable you to edit the contact. You need to design 'Edit Contact' screen yourself.
4. Clicking on the Delete icon should prompt you for deleting the contact. If confirmed, delete the contact, and refresh the list.
5. Clicking on the add action button should display the "Create New Contact" screen.
6. Clicking on the Power Action Bar button should logout the current user, start the "Login" screen and finish the current screen.

Part 4, Create New Contact (25 Points):
This screen is used to create a new contact. The requirements are as follows:
1. Just like the Sign Up screen should prompt the user to take a photo using Camera, or Gallery.
2. It should enable user to take inputs for First Name, Last Name, Phone, and Email.
3. Upon clicking the "Submit" button the new contact should be created and sent to firebase to be stored under the contacts for the currently logged in user. Finally the screen should be finished, which will show the "Contacts List".
   • The "Contacts List" screen should be refreshed to display the updated list.