Basic Instructions:

1. In every file submitted you MUST place the following comments:
   a. Assignment #.
   b. File Name.
   c. Full name of the student.
2. Each group is required to submit the assignment on Canvas.
3. Please download the support files which include a Java project to be used for this assignment.
4. Submit Codes:
   a. Zip all the project folder to be submitted on canvas.
5. Submission details:
   a. The file name is very important and should follow the following format:
      InClass10.zip
   b. You should submit the assignment through Canvas: Submit the zip file.
6. Failure to follow the above instructions will result in point deductions.

In this assignment you will get familiar with Firebase and Firebase data management.
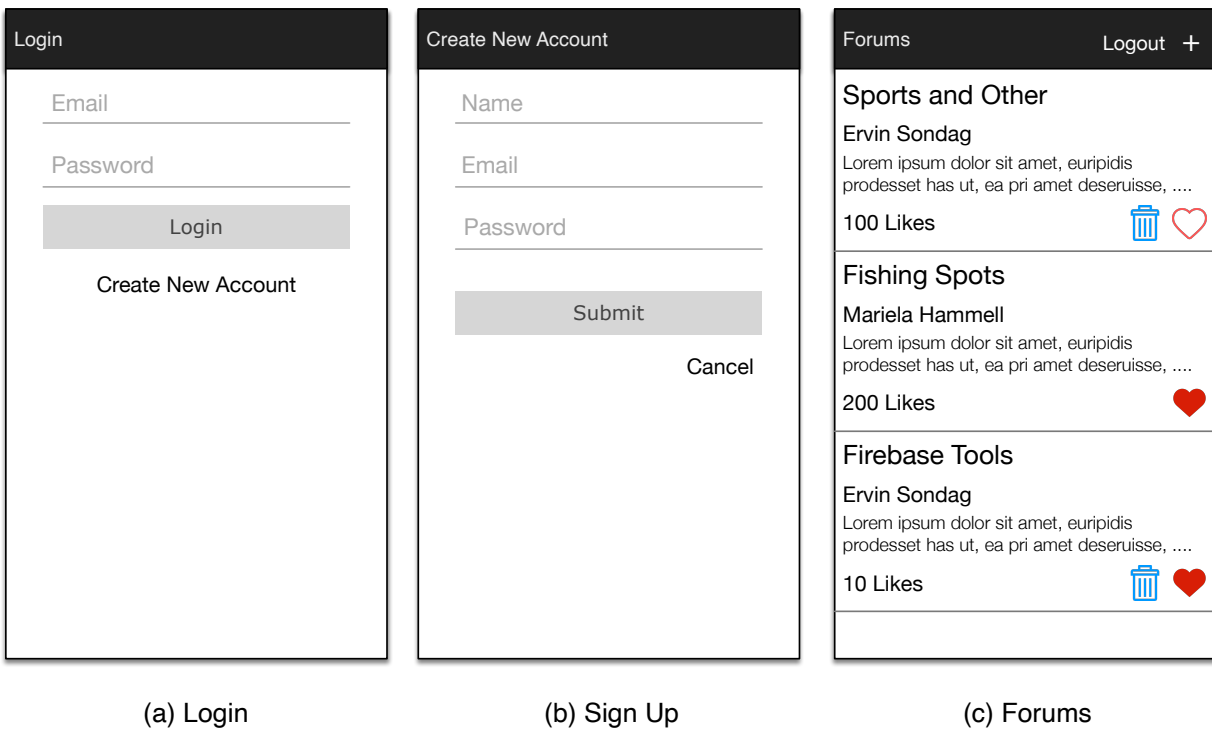

| (a) Login | (b) Sign Up | (c) Forums |

Figure 1, Application Wireframe

Part 1 : Login Screen (5 Points)
The interface should be created to match the UI presented in Figure 1(a). The requirements are as follows:
1. Upon entering the email and password, clicking the submit button should:
    a. Clicking "Login" button, if all the inputs are entered correctly, you should attempt to login the user using Firebase. If the login is successful then transition to the Forums Screen. If login is not successful show an alert dialog highlighting the error.
    b. If there is missing input, show an alert dialog indicating missing input.
2. Clicking the "Create New Account" should replace this screen with the Create New Account Screen.

Part 2 : Create New Account Screen (5 Points)
The interface should be created to match the UI presented in Figure 1(b). The requirements are as follows:
1. Upon entering the name, email and password, clicking the submit button should:
    a. If all the inputs are entered correctly, you should attempt to signup the user using Firebase. If the signup is successful then transition to the Forums screen. If signup is not successful show an alert dialog highlighting the error.

b. If there is missing input show an alert dialog indicating the error.
2. Clicking "Cancel" should replace this screen with the "Login" Screen.

Part 3 : Forums Screen (30 Points)
This screen enables the user to view the list of forums created by all the users on Firebase and to interact with these forums. As shown in Figure 1(c), The requirements are as follows:
1. Clicking the "+" menu button should display the Create New Forum Screen.
2. Clicking the "Logout" menu button should logout the currently logged in user and replace this screen with the Login Screen.
3. Each forum cell should display the following:
   a. Forum title, name of forum creator, two lines summary of the forum description, and the number of likes under each forum item indicates the number of unique users that liked the forum item.
   b. The trash "Delete" icon should only be displayed for forum items that were created by the currently logged in user. If the user clicks on the trash can icon it should present the user with an alert dialog asking the user if the selected forum should be deleted. If the user answers "OK" then the forum should be deleted from Firebase and the list should be refreshed to reflect this change.
   c. The like or unlike indicator which is represented by un-filled heart icon and filled heart respectively. The user can like the forum by clicking the un-filled heart icon, which should update the Firebase record to indicate that the user has liked this forum, and the heart icon should be changed to filled to indicate that the user has liked the forum.
4. The list of forums is stored by creation date in descending order.
5. Note that Clicking on a row item should segue to the the Forum Screen.

Part 4 : New Forum Screen (10 Points)
The interface should be created to match Figure 2(a). The requirements are as follows:
1. Clicking the "Cancel" button should dismiss this screen and return back to the Forums Screen.
2. Clicking the "Submit" button, the app should check if all the fields are entered and display an alert dialog if any of the entries is missing indicating that the missing field is required.
   a. If all the fields are entered, the app should create the new forum on Firebase. The upon completion should dismiss this screen and signal the Forums screen to refresh the list of forums.
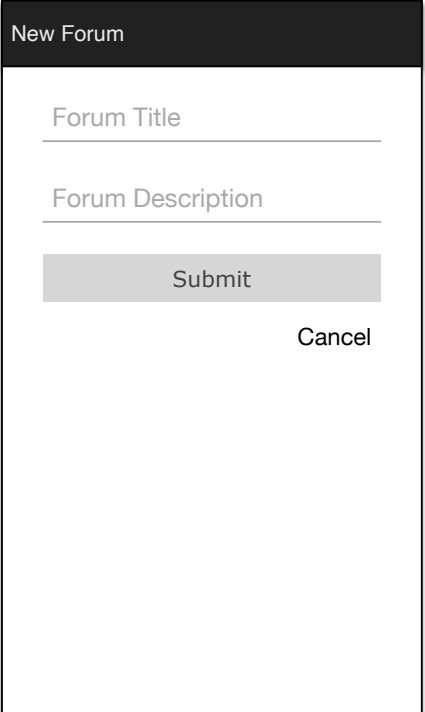   b. Upon returning to the Forums screen the list should be refreshed and should include the newly created forum.

Figure 2, Create Forum

Part 5 : Forum Screen (50 Points)

The Forum screen enables the user to view the forum details and comments made by other users on this forum. In addition, the view controller should enable the user to post comments on the forum. The requirements are as follows:

1. The Forum screen should display details about the selected forum, and show the list of comments as shown in Fig 3.
2. At the top of the screen, the forum title, forum creator name, and forum description should be displayed. The number of likes, the like/dislike heart and the delete button should follow the same functionality as in the Forums screen.
3. Creating a comment:
    a. The comment EditText should allow the user to enter a text comment.
    b. Selecting the "Add Photo" button allow the user to add a photo the comment. This should trigger the camera in order to take a photo, which should then be stored on Firebase, and the image url should then be stored with the comment. A preview of the selected photos should be displayed in an ImageView beside the "Add Photo" button as shown in Figure 3. The addition of a photo is not required when submitting a comment.
    c. After entering a comment, which could include a photo, clicking the "POST" button should store the comment and the photo url (if one added) to Firebase. Then the list should be refreshed to show the newly added image at the top of the list of comments.
4. The list of comment is stored by creation date in descending order as shown in Figure 3.
5. Note that the trash can "Delete" icon should only be displayed in comment row items that belong to the currently logged in user. If the currently logged in user clicks on the trash can icon it should present the user with an alert dialog asking the user if the selected comment should be deleted. If the user answers "OK" then the comment should be deleted and the list should be refreshed to reflect this change.
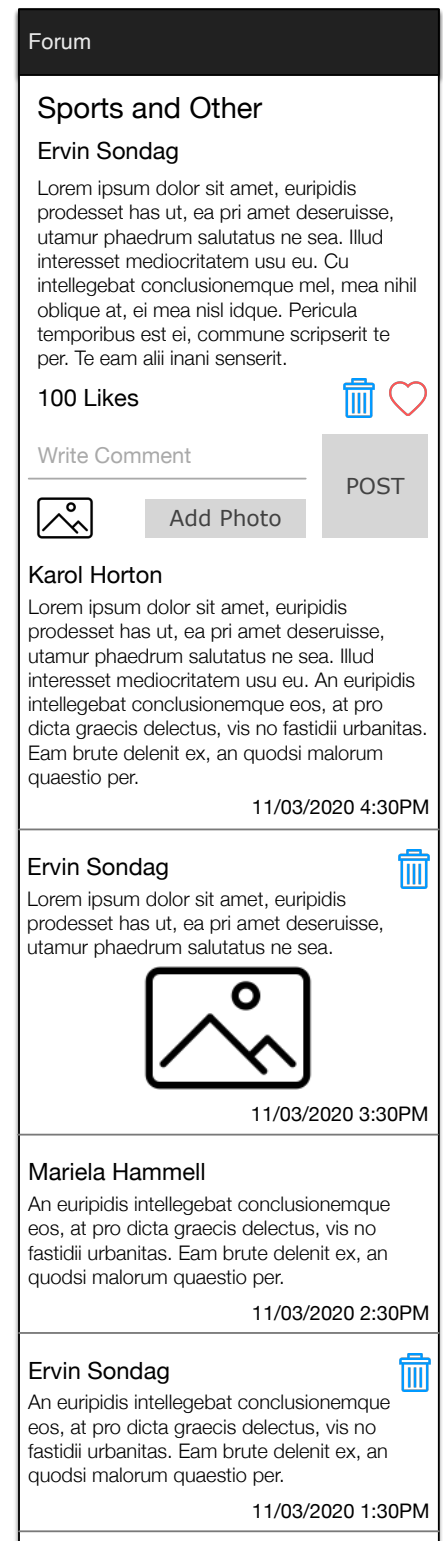

Figure 3, Forum Details