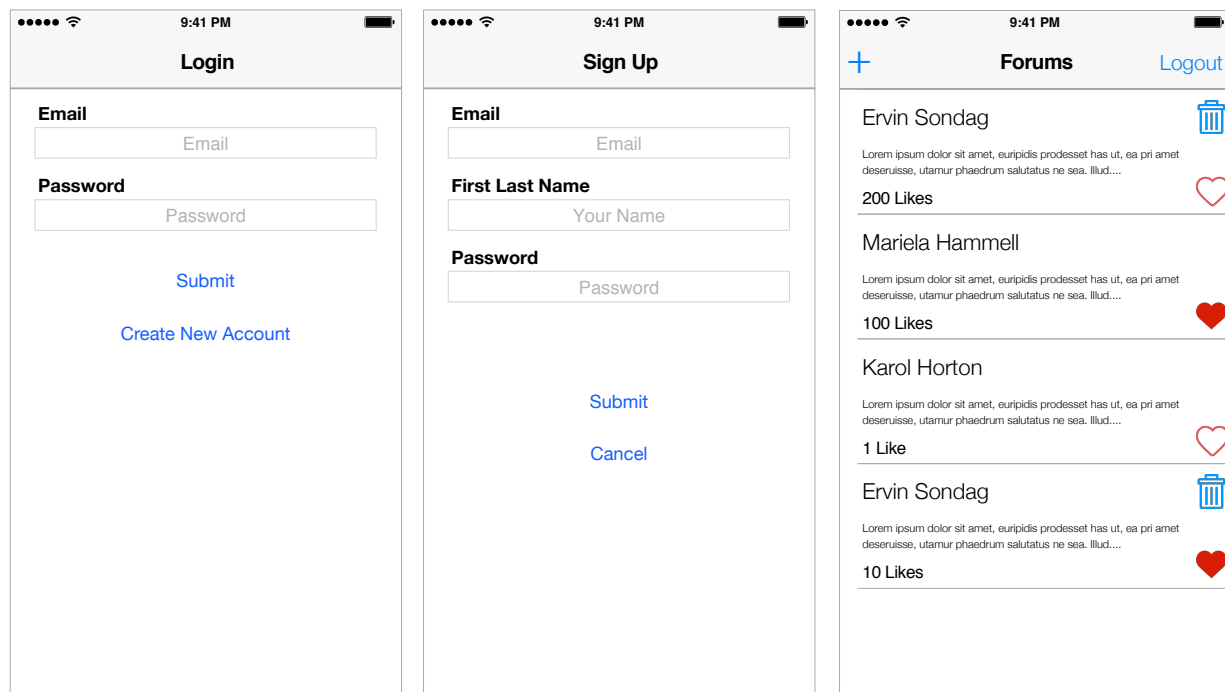**Basic Instructions:**

1. In every file submitted you MUST place the following comments:
   a. Assignment #.
   b. File Name.
   c. Student Full Name.
2. This is an individual assignment, each student is expected to work alone and submit their own work.
3. Your assignment will be graded for functional requirements and efficiency of your submitted solution. You will loose points if your code is not efficient, does unnecessary processing or blocks the UI thread.
4. Please download the support files provided with this assignment and use them when implementing your project.
5. Create a zip file which includes all the project folder, any required libraries, and your presentation material.
6. Submission details:
   a. You should submit the assignment through canvas: Submit the zip file.
7. **Failure to follow the above instructions will result in point deductions.**

In Class 10 (100 Points)

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

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

Figure 1, Application Wireframe

Part 1 : Login View Controller (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. 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 Contacts View Controller. 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.

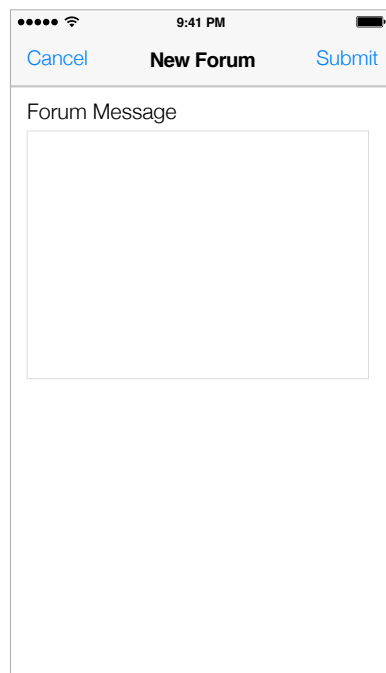## Part 2 : Sign Up View Controller (5 Points)

The interface should be created to match the UI presented in Figure 1(b). The requirements are as follows:
1. Upon entering the email, name 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 Contacts View Controller. If signup is not successful show an alert dialog highlighting the error.
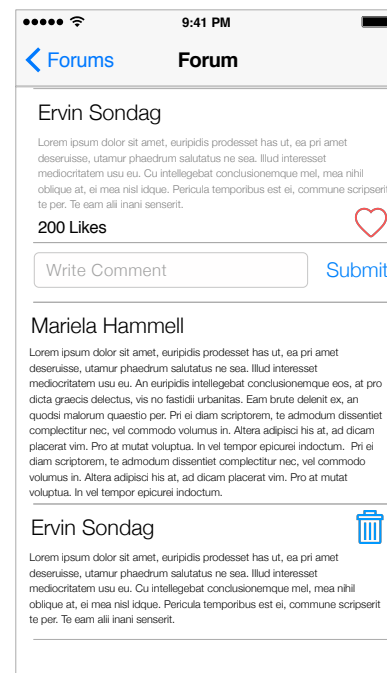   b. If there is missing input show an alert dialog indicating the error.

Part 3 : Forums List View Controller (40 Points)
This ViewController 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 "+" button should segue to the New Forum ViewController, you should use the "Present Modally" segue.
2. Clicking the "Logout" button should logout the currently logged in user and show the Login View Controller.
3. Each forum cell should display the following:
    a. Name of forum creator, two lines summary of the forum message, 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 UITableView 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. Note that Clicking on a row item should segue to the the Forum ViewController and pass it the required data.

(a) New Forum   (b) Forum ViewController

Figure 2, Application Wireframe

Part 4 : New Forum View Controller (10 Points)
The interface should be created to match the UI presented in Figure 2(a). The requirements are as follows:
1. This view controller should enable the user to enter a message that should be used to create a new forum.
2. Clicking the "Cancel" button should dismiss this ViewController and return back to the Forums ViewController.
3. 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 ViewController and signal the Forums View Controller to refresh the list of forums.
    b. Upon returning to the Forums View Controller the TableView should be refreshed and should include the newly created forum.

Part 5 : Forum View Controller (40 Points)
The Forum view controller 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 view controller should display details about the selected forum, it should include a UITableView. It is required that the UITableView rows accommodate the **variable heights** of their content and display all the messages and comments content as shown in Figure 2(b).
    a. The first row should display the forum entry, which is the forum topic that was created by the owner. It should show the full form topic message, number of likes and the like or not like heart have the same functionality as in the Forums ViewController.
    b. The second row should display the comment entry area, which includes a UITextField, and Post button. After entering a comment and clicking the "Post" button, the comment message should be stored in Firebase and the UITableView should be refreshed to reflect this change.
    c. The remaining rows should display the comment fields associated with the displayed forum.
2. 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 UITableView should be refreshed to reflect this change.