ITIS/ITCS 5180 Mobile Application Development
Homework 06

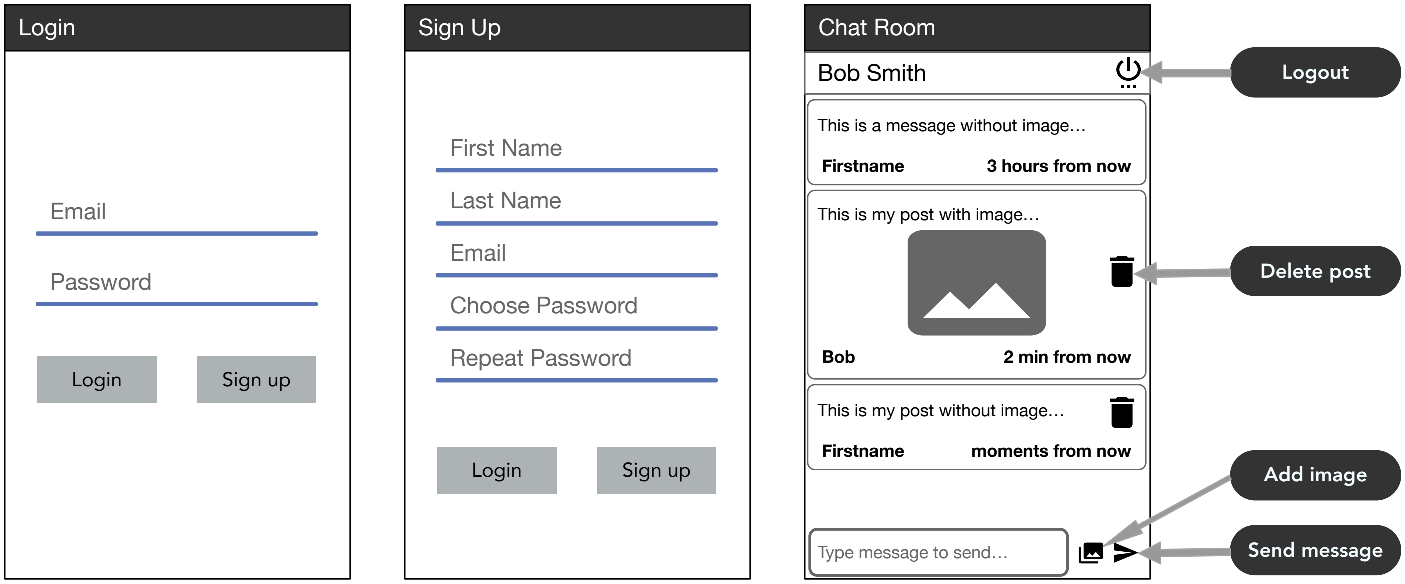
---

**Basic Instructions:**

1. In every file submitted you MUST place the following comments:
   a. Assignment #.
   b. File Name.
   c. Full name of all students in your group.
2. Each group should submit only one assignment on behalf of all the other group members.
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. Export your Android project and create a zip file which includes all the project folder and any required libraries.
5. Submission details:
   a. Compress the contents of your project folder. The file name is very important and should follow the following format: **Group#_HW06.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.
6. **Failure to follow the above instructions will result in point deductions.**

# Homework 06 (100 points)

In this assignment you will create a chat room application that uses Firebase realtime database and storage features.



**(a) Login screen** **(b) Sign Up screen** **(c) Chat screen**

**Figure 1: App Wireframes**

## Part A: Login (10 Points)

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

1. The launcher screen should be set to the Login Screen. When the app first starts, the login screen should check if there is a current user session:
    a. If the user is already logged in to Firebase, then start the Chat Screen and close the login Screen.
    b. If there is no current session, then the Login Screen should be used to provide user login.
2. The user should provide their email and password. The provided credentials should be used to authenticate the user to Firebase. 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 Chat Screen and finish the Login Screen.
    b. If the user is not successfully logged in, then show a toast message indicating the login was not successful.
3. Clicking the Sign Up button should start the Sign Up screen and finish the login screen.

## Part B: Sign Up (15 Points)

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

1. Clicking the “Cancel” button should finish the Sign Up Screen and start the Login Screen.

2. The user should provide their first name, last name, email, password and password confirmation. Clicking the “Sign Up” button should submit the user’s information to Firebase.
    a. If the sign up is not successful display an error message indicating the error message you received from Firebase.
    b. If the signup is successful, then display a Toast indicating the user has been created. **Then directly go to the Chat screen.**

**Part C: Chat Room (75 Points)**

The screen should show the list of messages retrieved from Firebase. Also the screen allows the user to add new text and image messages. The requirements are as follows:

1. Display the list of messages as shown in Figure 2(a). The messages should contain both text and images.
    a) Each message object contains a message text, an image, time, first name, and last name of the user.
    b) Each message includes a Delete button. Find the images for the buttons in Resource folder provided.
    c) Display the time the message was posted using the Prettytime library.
    d) Display the First name of the user that created the message.
2. When the user types a text message and clicks the “Send message” icon, it should store the new message in Firebase. Then list should be updated to show the latest messages.
3. When the user clicks on the “Add image” icon, it should open the image gallery for the user to select the image to be sent to Firebase storage. It should load the image replacing the “Add image” icon. Then if the user clicks on the “Send message” icon, it should store the image in Firebase, get the downloadable link, and store the message in Firebase realtime database. Then list should be updated to show the latest messages.
4. When you click on the delete icon on a message, you should be able to delete the message.
5. After successfully deleting a message, the list should be refreshed to show the messages stored on Firebase.
6. Clicking the logout icon should logout the user from Firebase and take the user back to Login Screen.