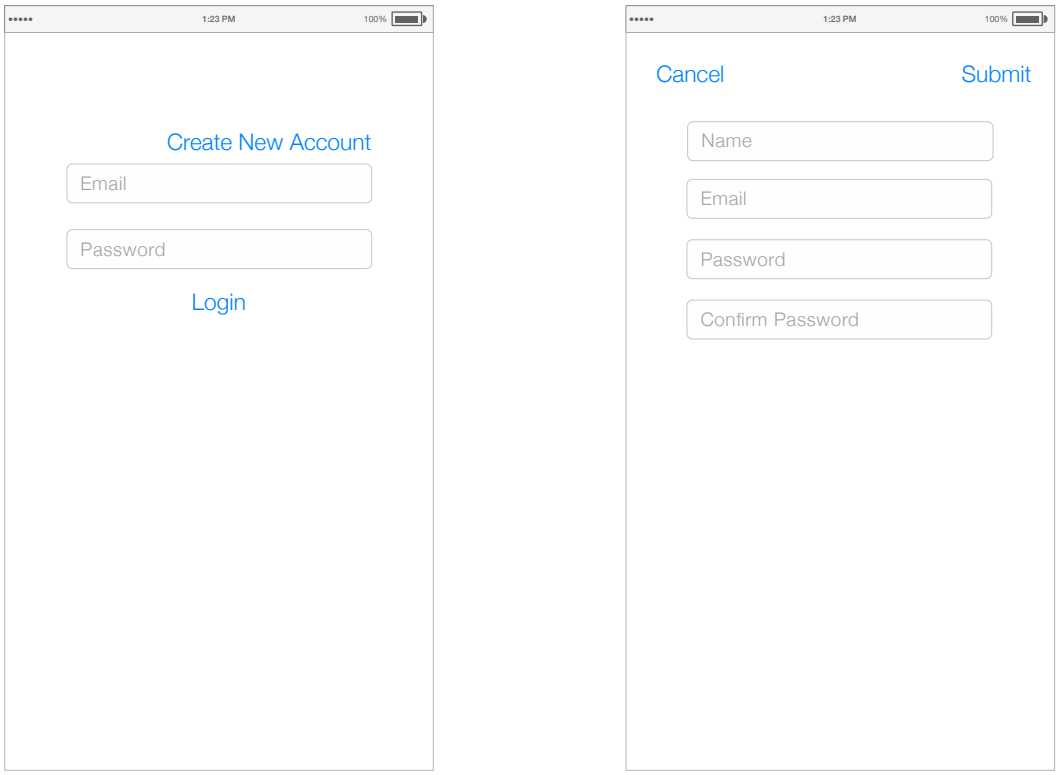
ITIS/ITCS 6010 – Advanced Topics for Mobile App Development
In Class Assignment 6

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 Assignment 6 (100 Points)

In this assignment you will get familiar with Firebase to create a notes application.



(a) User Login (b) User Signup

Figure 1, User Login Related Application Wireframe

Part 1: User Signup and Login (30 Points)

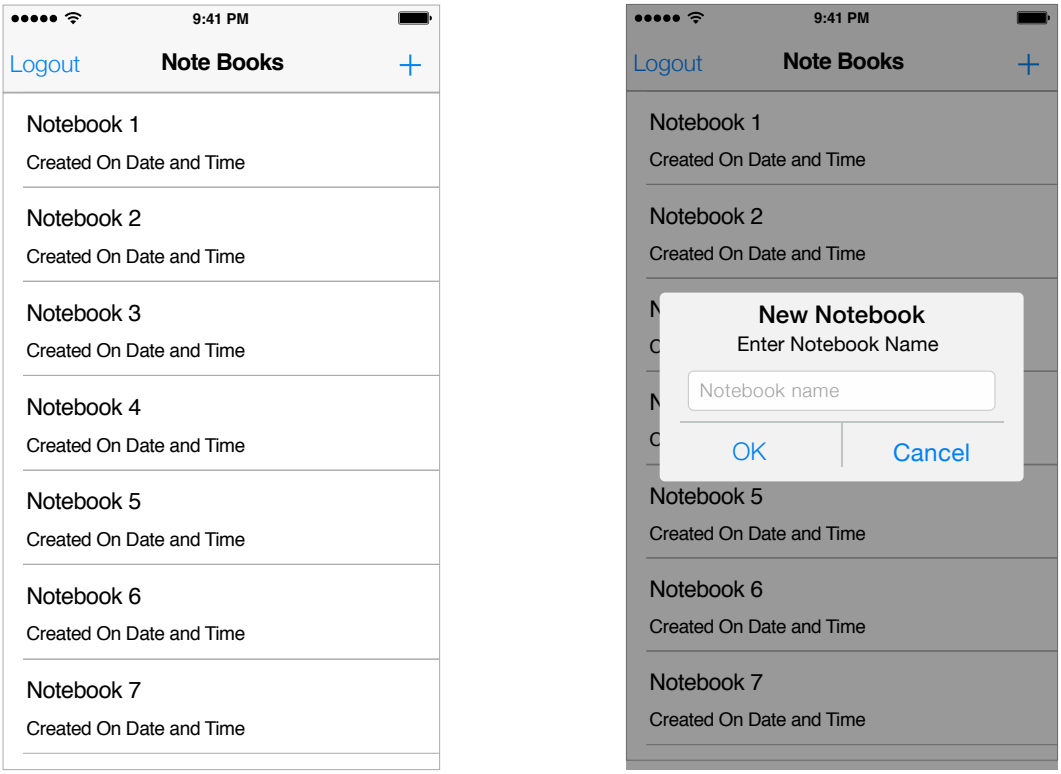
Your app should implement both login and signup functionalities. You should use Firebase to Store the user's display name, email and password. The requirements are as follows:

1. The main view controller should be set to the Login view controller. When the app first starts, the Login view controller should check if there is a current user session, by using the Firebase provided methods to check if there is a valid current user:
   a) If there is a current valid user, then the Notebooks view controller should **replace** the Login view controller.
   b) If there is no current valid user, then the Login view controller should be used to provide user login.
2. Create a Login view controller (Figure 1(a)):
   a) The user should provide their email and password. The provided credentials should be used to authenticate the user using Firebase. Clicking the "Login" button should submit the login information to Firebase to verify the user's credentials.
      • If the user is successfully logged in then the Notebooks view controller should replace the Login view controller.

- If the user is not successfully logged in, then show an alert dialog indicating that the login was not successful.

b) Clicking the “Create New Account” button should navigate to the the Signup view controller.

3. Create a Signup view controller (Figure 1(b)):
   a) Clicking the “Cancel” button should navigate back to the Login view controller.
   b) The user should provide their display name, email, password and password confirmation. The provided credentials should be stored in Firebase. Clicking the “Sign Up” button should submit the user’s information to Firebase to verify the user’s credentials.
      - If an account with the same email already exists, display an error message in an alert dialog indicating that the account account was not created and the user should select a different email.
      - If the sign up process is completed successfully then the Notebooks controller should replace the Signup view controller.



(a) Notebooks List (b) Add New Notebook

Figure 2, Notebooks Application Wireframe

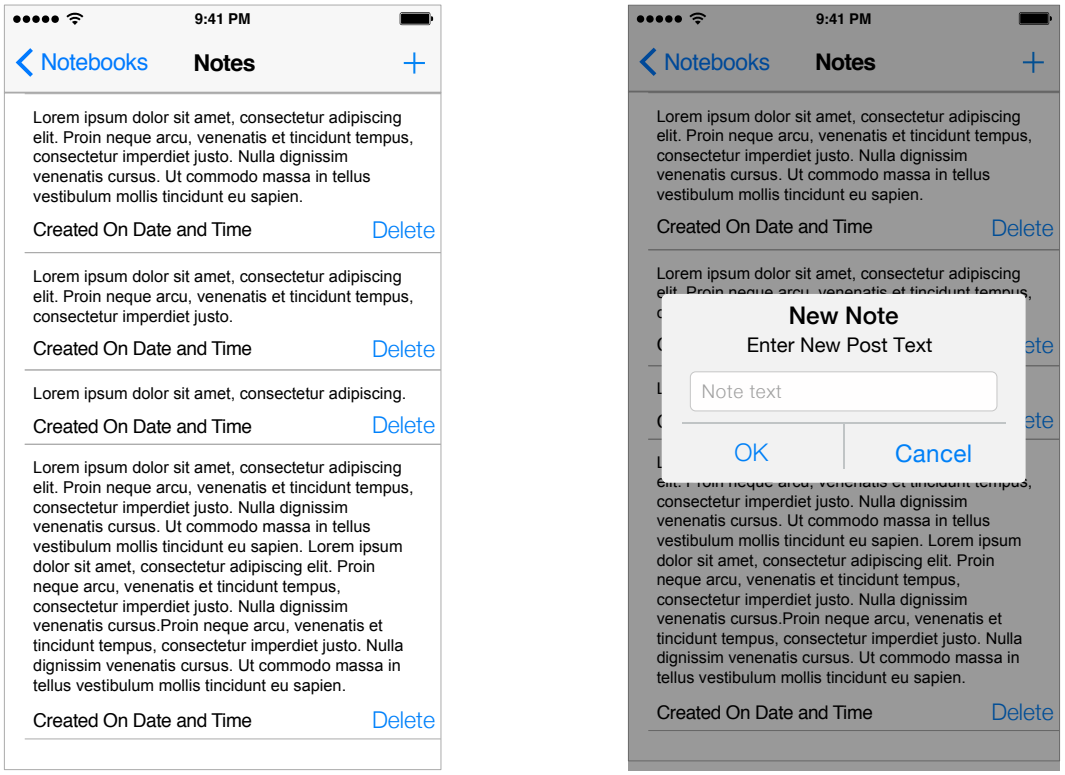
Part 2 : Notebooks View Controller (40 Points)

The interface should be created to match the UI presented in Figure 2(a). The requirements are as follows:

1. Retrieve the list of notebooks stored on Firebase only for the logged in user and display the TableView as shown in Figure 2(a). Notice you should show each

notebook name and the create at date for each notebook.

2. Clicking on a row item should segue to the Notebook Notes view controller to display the notes for the selected notebook.
3. Clicking the “+” button should show an alert dialog as shown in Figure 2(b). Upon clicking the “OK” button a new notebook should be created for the current user and should be stored on Firebase. The tableview should be refreshed to reflect the newly added notebook.
4. Clicking on the Logout button should logout the user, and replace the current view controller with the Login view controller.



(a) Notes List in a Notebook

(b) Add New Note to Notebook

Figure 3, Notebook Notes Application Wireframe

Part 3 : Notebook Notes View Controller (30 Points)

The interface should be created to match the UI presented in Figure 3(a). The requirements are as follows:

1. Retrieve the list of notes stored on Firebase for the selected notebook and display the TableView as shown in Figure 3(a). Notice you should show each note, the note text and the create at date.
2. Clicking the “+” button should show an alert dialog as shown in Figure 3(b). Upon clicking the “OK” button a new note should be created for the selected notebook and should be stored on Firebase. The tableview should be refreshed to reflect the newly added note.
3. Clicking the “Delete” button should delete the selected note from Firebase and should refresh the list to show this change.