

CS 329E: Bulko

Programming Assignment 2

Simple User Interface

1 Problem Definition

Your goal in this assignment is to create an iOS app with a simple user interface. The app that you create will have seven elements:

1. A label reading, “User Login”.
2. A label reading, “userID:”.
3. A text field where the user can type in a userID.
4. A label reading, “Password:”.
5. A text field where the user can type in a password.
6. A button labeled “Login”.
7. A label reading, "Not currently logged in"

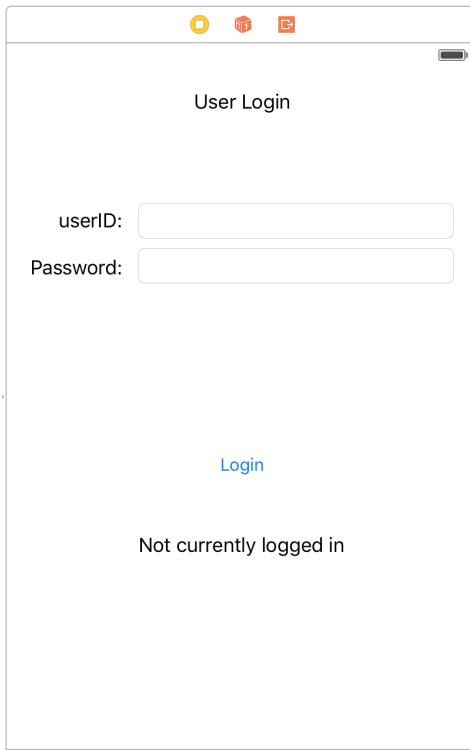
Expected behavior of your app:

- The user uses the keyboard to enter strings into the “userID” and “Password” fields. (You do not have to worry about encrypting or hiding the password for this assignment.)
- If the user touches anywhere in the background outside of the keyboard, the keyboard is dismissed.
- When the user clicks the “Login” button, if all is well, the text in the status label should change to "`<userID> logged in`".
- If the user clicks on the “Login” button when one or both of the text fields are empty, the text in the status label should change to "`Invalid login`".

2 Detailed Instructions:

Create a Single View application project named `<lastName><firstName>-HW2`.

Switch to the Storyboard. Set the size of the View Controller to the newest model supported by the version of Xcode you have installed.



- Add the seven elements listed in Section 1 above to the Storyboard. Space them so that the screen looks something like the image shown. Use constraints to ensure that:
 - the "User Login" label, the Login button, and the status label are all centered horizontally.
 - the text fields and the status label all have a width of 250.
 - the contents of the "User Login" label and the status label are centered, and the contents of the "userID:" and "Password:" labels are right-justified.
- Create @IBOutlet s for each of the two text fields and the status label.
- Create an @IBAction linking the button to a method called buttonPressed.

Switch to ViewController.swift.

- Add code to dismiss the keyboard.
- Write the code for buttonPressed.
 - If data has not been input into both of the two text fields, update the status field with the "Invalid login" message.
 - If the data is fine, update the status field with the message, "<userID> logged in".

3 Grading criteria

1. The user interface conforms to the design shown. (50%)
2. The code behaves according to the above specifications. (50%)

- 3. If the app does not build and run, ZERO points will be given.**
4. The Coding Standard is followed. One point deducted for each violation.

4 General criteria

1. Set the simulator to the newest model supported by the version of Xcode you have installed.
2. I will be looking for good documentation, descriptive variable names, clean logical structure, and adherence to all coding conventions expected of an experienced programmer, as well as those outlined in the Coding Standard document. There will be penalties for failure to meet these standards.
3. Your code must compile and run before submission.
4. Xcode will automatically generate standard headers to your .swift files. Add two lines to each Swift file so that the header includes the following:

```
// Project: LastnameFirstname-HW2  
// EID: xxxxxx  
// Course: CS329E
```

5 To submit your assignment:

Zip up your project directory ("LastnameFirstname-HW2") into a .zip file and attach it to the HW2 assignment on Canvas before the due date/time.