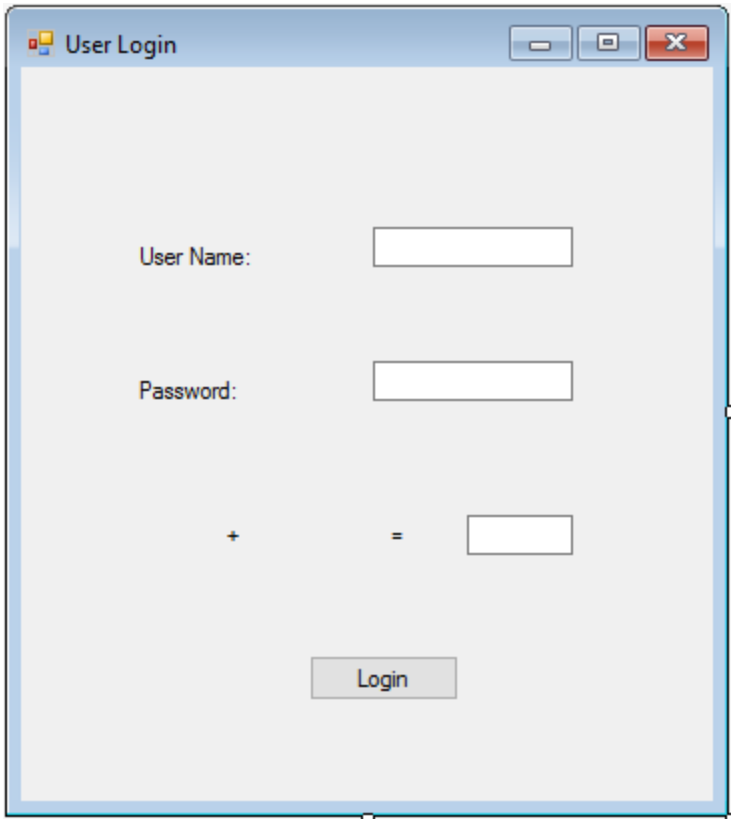
CIS 366 Introduction to .NET Development using C# (Spring 2019)

Assignment 3

Requirement

This assignment is to design a windows application to validate users' credentials for login. This application acts like a "fake" login systems because real one does not store user name and password on file in variables. In this assignment, you will use Visual Studio 2015 and write C# code the for following functionalities:

1. Build a user interface that looks like this. It allows users to input their user name and password. The third line is for a Captcha question asking the user to enter the answer of an easy math question (e.g. 3 + 4 = ?) to make sure that a real human being is using the system, not a robot. Both numbers of the addition operation will be randomly determined when the form loads, so place labels for those numbers and "+" and "=" signs. Change the password textbox's **passwordChar** property to make sure that password entered will be shown as *. Also put a textbox to receive user's input for the Captcha question.

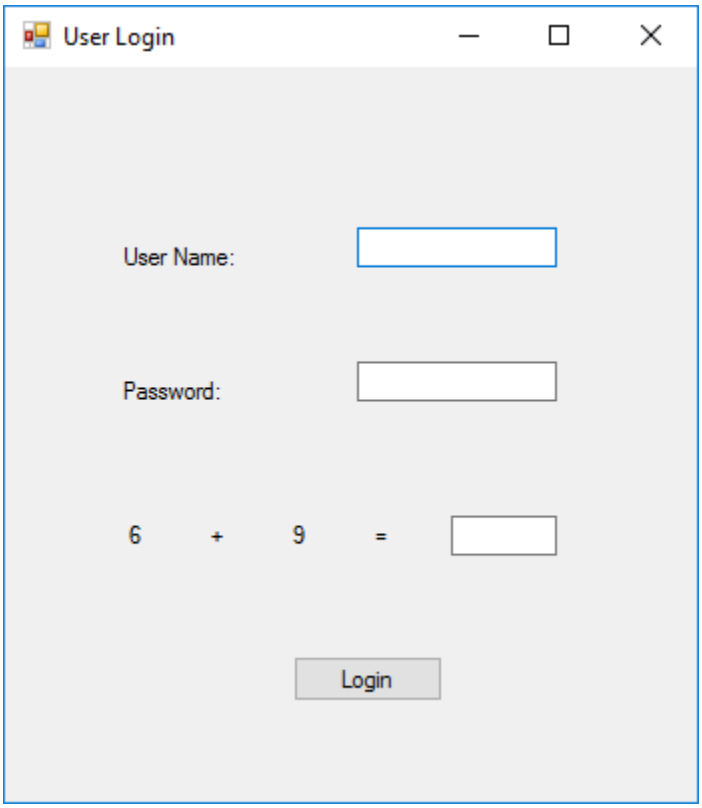


2. Double click anywhere on the form to write code under **Form1_Load** function. Those codes will be executed when the application loads. When loaded, the system generates

two random numbers in the range of 1-10 for Captcha question and put them on the corresponding labels. For random number generation, you can use the following method:

```
Random r = new Random();
int captchaNumber1 = r.Next(1, 10);
```

The user interface looks like this when loaded:



3. In the fields, define two **static string** variables **usernameOnFile** and **passwordOnFile** to store your testing case for user name and password. This simulates the process of reading real usernames and passwords from the database in a real system.

4. Program the Login button to perform the following functionalities when clicked:

    a. Check the user inputs of user name and password textboxes. If either of them is blank, show the user a message box indicating that they need to enter both username and password, also clear both textboxes and let the username textbox refocus when the error message box is closed by the user.

    b. If both user name and password are entered, check the user input of Captcha answer. If it is blank or input is not an integer, show a similar message box asking the user to re-enter. Use `TryParse()` to validate that.

c. If the Captcha answer is properly entered, check if the answer is correct based on the random numbers generated. If the answer is incorrect, show a similar message box asking user to correct.

d. If the Captcha answer is correct, compare the user's input of user name and password with **usernameonFile** and **passwordOnFile**. If any of those is not matched, show a similar message box asking user to re-enter. Show the user a message box to congratulate him/her on the correct information entered if both user name and password enter match with those on file.

Submission

Zip your ENTIRE project folder and name your zipped file to (yourlastname)_a3. Submit your zipped file to the Blackboard dropbox as an attachment.