Student Registration App (WPF)

Last Updated: November 18, 2015

Assignment

Write a small application that will enable the end user to display/add/edit/remove a list of student registration info.

For this assignment you should be using WPF for the UI part and you should apply all the knowledge and good programming practices you learned so far.

This assignment does not cover the persistence layer and everything will be held in memory as far as this assignment is concerned.

There is a Balsamiq Mockup included with this assignment. Please pay close attention to the mockup that is provided as well as the notes that are attached on the design sheets.

Your application should conform to the following workflows:

- **User Login**
  - As soon as the app launches the user should be presented with this screen
  - User should not be able to proceed unless [s]he provides the correct username and password
  - Max three unsuccessful attempts and the application quits forcing the user to restart the app again

- **Main Window**
  - Once the user passes the login screen this is the screen that [s]he should be seeing first
  - Please write a helper class to randomly generate the fields here
    - I would like to see the usage of a helper class and the Randomizer function to randomly generate records in memory
    - 10 random student records would suffice
    - Department field should have a static list that is already composed of what is shown on the Balsamiq docs (i.e. A department can be only one of the following: Information Systems, International Affairs, Nursing, Pharmacy, Professional Studies, Psychology, and Public Administration) but the department will be assigned to the students at random
    - For generating Student IDs – Make sure generated string is in the format that is presented on the docs – i.e. It should follow the format of "xxx-xx-xxxx"

- **New Student Form**
  - Functionality of this form is outlined in the Balsamiq document
  - The user will be presented a confirmation window if the user misses any of the fields – i.e. leaves them blank (You should be covering for spaces as well)
    - When this is the case, the user should be warned and then brought back to this form again so [s]he can provide the missing info from the registration info

- **Remove Student Form**
  - Functionality of this form is outlined in the Balsamiq document
  - The user should be presented a confirmation window since this is a deletion operation and there is no way to recover
    - Related confirmation window is also outlined on the provided documents

- **Edit Student Form**
  - Functionality of this form is outlined in the Balsamiq document
  - The user should be presented a confirmation window once [s]he is done and there is no missing field – i.e. all the fields are filled in (Similar to the workflow mentioned above with Remove Student Form)
  - The user should be presented a warning form if [s]he starts modifying the fields but leaves at least one of them blank (Similar to the workflow described above with New Student Form)

Additional Notes:

- You will need Balsamiq installed on your system since that is the way we chose to communicate our UI design and ideas – If you have not done so already, please download and install Balsamiq first
- There are no unit tests for this assignment since this assignment is UI heavy and automating UI tests is out of our scope and is tedious and consumes a significant amount of time
- An example skeleton code was already shown in the class and this code can be found on GitHub (Already posted an announcement about this)
- Please use your own judgment for good design – i.e. Use everything that you learned so far such as OOP Pillars (encapsulation, inheritance, polymorphism), helper methods,

properties, utility classes etc. This includes designing the forms in such a way as well – Remember forms are just plain classes at the end of the day
- Pay close attention to the Balsamiq mockups (This includes the form and component sizes and the yellow attention stickers) – You don't have to go crazy about the positions of the controls (i.e. x,y coordinates) but make sure to at least align them so the design looks tidy and nice

Developer Notes

- Please pay attention to the deadlines. Your commit timestamp should not be later than the hard deadline
- Before committing your changes please make sure
    - Your solution files build and run fine
    - If any other document files other than the code files are required, then please make sure you included these as well
- Once you committed your changes, please test it one last time to make sure everything is ok
- Please send the github link to the location of your solution to me and to your TA

Deadlines

- For this assignment deadline is **12/7/2015 @ 4pm** and hard-deadline is **12/12/2015 @ 4pm.** Unfortunately assignments sent over after this deadline will not receive any credit