Homework 5: Graphical User Interfaces

Disaster has struck at Facelook! The team that moved to Giggle sabotaged Facelook's computers and ruined their entire code base. The Facelook CEO, Zark Muckerberg, wants to rebuild Facelook from scratch and has asked you to implement their graphical user interface. Your goals in this assignment are to gain experience designing and building GUIs and to learn how to use Java Swing.

To facilitate your work, Zark (a.k.a. "the Muck") Muckerberg has provided a sample back-end data API you must use when building your GUI. Make sure you understand the back-end API before starting your GUI construction; eventually (in a later assignment) your GUI will be the front-end interface for Facelook's distributed system, which will use Java networking to implement this same API.

Your Facelook GUI must include the following features:

Login screen:

- Two text fields:
  - Email address
  - Password, using password-style ("***") text
- One button that allows the user to submit the information and attempt to log in
  - If successful, go to the NewsFeed in a separate window
  - If unsuccessful, give a pop-up dialog with an error message
- One button that goes to the Registration screen

Registration screen:

- Four text fields:
  - Full name
  - Email address
  - Password, using password-style text
  - Confirm password, using password-style text
- One button that will submit the information and attempt to register the user
  - If successful, give a pop-up dialog to inform the user
  - If unsuccessful, give a pop-up dialog with an error message
- One button that goes back to the Login screen

News Feed screen:

- One text field to search for other users
- One button to execute this search
  - If successful, move to the target user's Profile screen

        - If unsuccessful, give a pop-up dialog to inform the user
    - One button to go to your Friends screen
    - One button to go to your own Profile screen
    - A panel of 10 Status objects, the most recent 10 posts by your friends
        - When a Status object is clicked, go to the Profile screen for the user who posted that Status

Profile screen:

- One button to go to the NewsFeed screen
- A label indicating whose Profile is being displayed
- A panel of 5 Status objects, the most recent 5 posts from this user
    - These Status objects should only be visible if you are friends with the user
- If this Profile is the current user's own profile, have a text field and button to allow the user to post a Status object
- If this Profile is not the user's own profile, have a button to add or remove this person as a friend

Friends screen:

- One button to go to the NewsFeed screen
- A panel of buttons that indicate pending friend requests
    - When a friend request is clicked, go to the requester's Profile; the user can then confirm the friend by clicking the add friend button there
- A panel of buttons that indicate current friends
    - When clicked, these buttons go to the friend's Profile

Status:

- Contains text for the status, the poster, and the timestamp
- When a Status is clicked, go to the poster's Profile

Zark doesn't want the shareholders to notice the new implementation; make sure you use the classic Facelook blue (RGB (59,89,152)) in your panels.

To launch the GUI, create a Main class (inside Main.java) with a main method. In the main method you should initialize the backend database and launch a JFrame containing the Login screen.

Submitting your work

To submit your work, commit your lab assignment and work for this assignment to your SVN `trunk/homework/5` directory by Tuesday, October 23rd at 11:59 p.m.

Evaluating your homework submission

Overall this assignment is worth 100 points, including the lab portion of the assignment. Points will be allocated as follows:

Login screen: 9 points
Registration screen: 14 points
News Feed screen: 19 points
Profile screen: 16 points
Friends screen: 10 points
Status button: 2 points
Separation of Login/Registration and main app: 5 points
Miscellaneous: 5 points

Lab component: 10 points

Style: 10 points

You can also earn a small amount of extra credit if you re-implement a backend database so that data persists between runs of the application. To do this, your persistent database must implement the `Backend` interface; place your solution into a new package.