

Sriram Atluri

Professor Faramarz Mortezaie

CS 146: Data Structures & Algorithms

20 May 2020

Social Media App Project

The Social Media App Project is intended to provide an intuitive app for people to have a virtual social network to know how many friends are in their contacts, the events going on in their life, as well as allowing for the adding and removal of friends from a person's social network. Moreover, ensuring the code can flow well as per the user's intuition was the key challenge. In order to solve this problem, the Java language was used to create the program which involved the use of four different classes packaged together and the use of a Graphical User Interface (GUI) program, which included `Social_Media_Profile`, `Profile_Manager`, `Driver`, and the GUI class by the name `Window`. Prior to starting all the coding work, the logistics by how a profile was created and how the profiles were to be managed had to be thought out in order to have a successful back-end in the place where the actual adding and removal of friends will take place as per the user's input. The specific process by how the code was actually created will be described in the remaining of this report as well as some aspects that could have been done better.

To implement the classes, four different classes that had to be made were `User`, `Social_Media_Profile`, `Profile_Manager`, `Driver`, and `Window`, as stated earlier. The last class that was created was used for the Graphical User Interface. In the `User` class, the means by configuring a user's name was laid out. The process consisted of creating two private `String` variables by the names `first`, and `last`. The step following that was the constructor which depends

on the setName method with the first name and last names passed in as string values to the first and last variables. Following that, the getName method was created to return the first and last names. Following the User class, the Social_Media_Profile class was created where four private class variables were needed, which were the BufferedImage for the photograph, User for the persons_name, a String variable to hold the status, and an ArrayList to hold the contact profiles. Social_Media_Profile class was the program file where attributes such as images, user name, user status, and the list of friends. Just like the previous class, a constructor was required to instantiate those variables and follows the process of set and get to access the private variables in the subsequent methods introduced into the code. The Profile Manager incorporates code that is dependent on the add_friend and remove_friend methods listed in the Social_Media_Profile class, so that the specific profiles in the array list could be allocated or removed out. The adding or removal of the profiles required the creation of an ArrayList stored in the variable contact_profiles. The adding of the profiles simply required the creation of a profile from the Social_Media_Profile class, whereas the removal of a profile involved calculating the size of the array as well as using that in a for loop and setting up an if condition that made use of the equality operator to determine that if second profile variable held the same value as the first profile variable, then it was to be removed. Coming to the Driver class, the code built upon the Social_Media_Profile and Profile_Manager classes. This time an instance of the Profile Manager had to be created in order to hold the profiles and the individual social media profiles were created by creating individual instances of different people and calling upon the addProfile, createFriend, getFriendsList, and splitFriend methods.

```

--- Profiles Data ---
Person name: John Walker
Status: Enjoying Switzerland!
Number of friends: 0
Person name: Preet Agarwal
Status: Enjoying the Rajasthan and Kerala tour!
Number of friends: 0
Person name: Vignesh Thaman
Status: Flying off to Chicago!
Number of friends: 0

--- Friendships are now made! :) ---
Person name: John Walker
Status: Enjoying Switzerland!
Number of friends: 1
Person name: Preet Agarwal
Status: Enjoying the Rajasthan and Kerala tour!
Number of friends: 2
Person name: Vignesh Thaman
Status: Flying off to Chicago!
Number of friends: 1

Viewing friends list
Friend: Preet Agarwal
List of this person's friends
John Walker
Vignesh Thaman
Friend: John Walker
List of this person's friends
Preet Agarwal
Friend: Vignesh Thaman
List of this person's friends
Preet Agarwal
--- Friendships broke apart! :( ---
Person name: John Walker
Status: Enjoying Switzerland!
Number of friends: 0
Person name: Preet Agarwal
Status: Enjoying the Rajasthan and Kerala tour!
Number of friends: 0
Person name: Vignesh Thaman
Status: Flying off to Chicago!
Number of friends: 0
Process finished with exit code 0

```

Figure 1. The output of the Driver class.

Coming to the GUI class was a different process altogether as it was more about event-driven programming. To create the GUI as how it is formally done, I had a home screen as well as the actual interface where the user can proceed about doing the adding and removal of profiles. To create the home screen as well as the second interface, I had to use the JFrame component which makes the program visible to the user. In order to input the texts for the title as well as within the program, I made use of the setTitle and JTextField. To transition from the first interface to the second interface, I used the JButton widget which acts upon the onClickListener. In order to make the first screen disappear, I had to dispose off the JFrame used for the home screen. In the second screen, I made extensive use of JLabels to display the normal text such as Name, Status, and the number of friends.

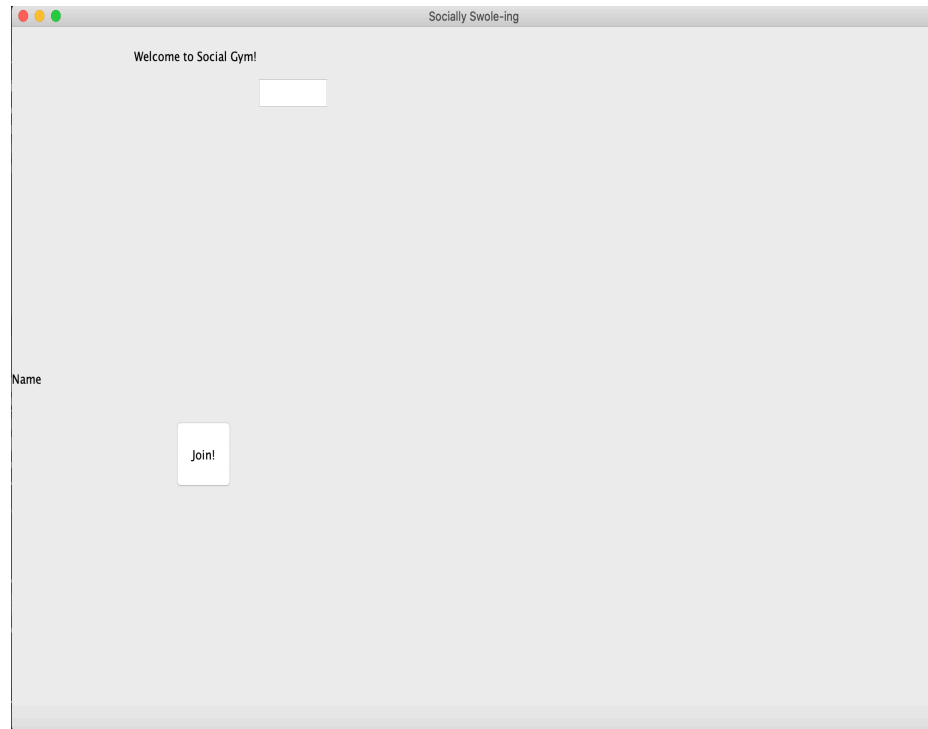


Figure 2. Home Screen of the GUI

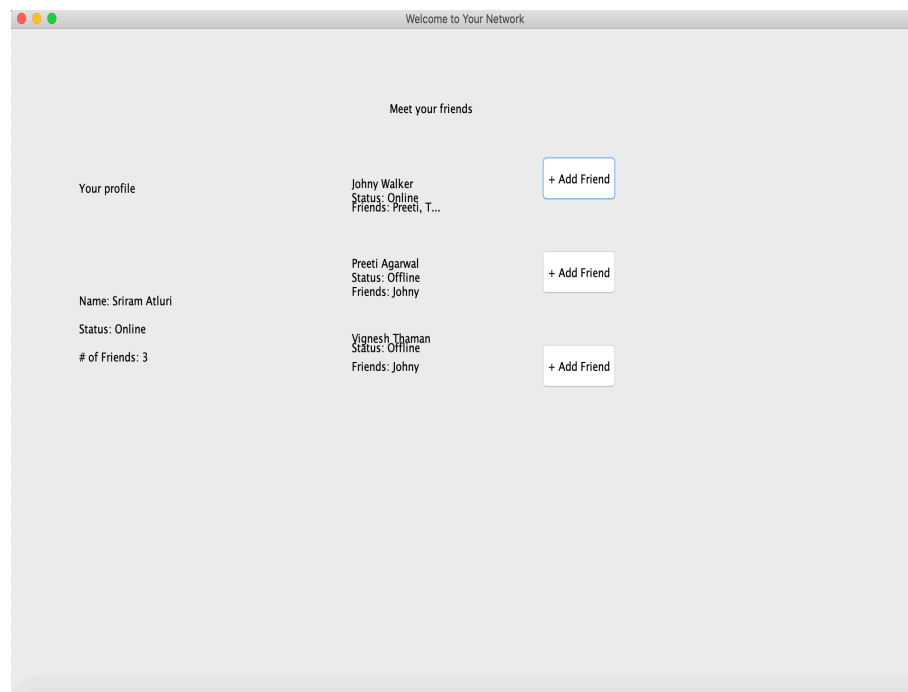


Figure 3. The screen of the second interface in the GUI.

The project was a learning experience in multiple ways. I would say I need more experience with the Swing part of Java language, considering that Java is still quite new to me and I am in need of some extra exposure to it. If there was something I would improve if I had more time from a coding point of view, is become more familiar with Swing to see if I could give a more stylish layout and spend some more time on the BackEnd portion. To provide clarity on this, I really would hope to ensure my “Add Friend” button actually adds the friend. I personally think I am in need to brush up on my skills with the Swing toolkit of the Java language.