Software Design - Project Final Meeting

Due Date

- Final Meeting - 100 points: Thursday, December 15

Final Project Meeting. You must meet with me before **3:00 pm** on Thursday December 15th for a final project meeting. At this time you must turn in the required documentation for the system (see below). You will also demonstrate your project for me one final time. At this meeting you will also be able to demonstrate additional features of your system that will be considered for extra credit.

Grading. The points for this final meeting will be distributed as follows:

- 10 points for having the meeting
- Improvements from previous meetings: 20 points (If you were not given areas to fix in the previous meetings then you do not need to do anything to receive these points)
- 30 points for the documentation
- Overall Execution and code review: 40 points (Will happen after the meeting)

Project documentation. This document must include the following pieces:

- A cover page that has your name and a title for your project.
- A **use case diagram** that documents the specific actions that a user would make in your GUI.
- An up-to-date **class diagram for the server**. Make sure your classes include the methods that you actually implemented and the instance variables. Your diagrams will be checked against your code to make sure they correspond with each other! You should include Java's Collection (HashMap, ArrayList) classes in your document if you used them (but only list the methods that you used for each class). You do NOT need to include other Java classes that you used such as Scanner and Socket.
- A set of **sequence diagrams for the server**. You should produce a sequence diagram for the use cases of registering, following someone, and retrieving messages with a certain hashtag. You only have to produce these for the server side. You only have to produce these 3 sequence diagrams As with your class diagram(s), these diagrams will be checked against your code for accuracy and consistency. Does your diagram accurately document what your code actually does?

This document will be graded based on how accurate it is (does it accurately describe how your system works), and how complete it is (are methods or fields missing from the class diagram, do you have all the method calls in the sequence diagrams, do you have all the use cases).

Turning in your project. Zip your client and your server folders and submit those zipped folders as well as your documentation on Blackboard.