Project: Send/Receive

Points:

Design: 20 points
Implementation: 20 points
Meeting: 10 points

due dates:

- Friday, November 4 – Class and Sequence Diagrams, Protocols and GUI picture(s)
- Wednesday, November 16 - Implementation meeting

You will need to implement these six pieces of the project at this stage:

- **Sending a public message**
- **Retrieving a set of messages sent by the people the user is following that have not been read yet AT LOGIN TIME (retrieving messages while logged in is the next stage)**
- **Retrieving all messages that have a hashtag**
- **Viewing sent messages since login**
- **Making sure the X button logs a user out if it closes the program**
- **Being able to shut the server down**

Your GUI needs to provide a way for the user who is logged on to create and send a message which will be saved on the server. Your server needs to save these messages in such a way that they can be retrieved in 2 ways. A user should be able to retrieve messages sent by people they are following, **but only retrieve messages they have not read yet – only new messages** when they log in (or when they open the window that will display the messages). A user should also be able to search for a hashtag and all messages (whether they have been read previously or not) with that hashtag will be retrieved. In addition, a user should be able to read all the messages they have sent while they have been logged on.

NOTES:

- If you are using the same area to display all messages you need to be able to go back and forth between those sets of messages without going to the server
- You should NOT have to go to the server to retrieve messages sent by the user

Design requirements (for Friday, November 4th).

<u>Class Diagram</u>

You should update the class diagram for the server that you turned in for the last design to include any new classes, methods, and instance variables needed to store messages. Please remember these requirements for your class diagram:

- You must include any of Java's collection classes that you use in your project (LinkedList, ArrayList, HashMap, etc.) in the class diagram. You do NOT have to include any other Java classes such as Scanner or Random.
- For the classes that you will be writing, you must include your best estimation of what instance variables you will have. You should also include any methods you have listed in the sequence diagrams.
- For Java's collection classes you only need to include the class name. You do not need to include any instance variables or methods in the diagram.

Sequence Diagrams

Provide a sequence diagram for sending a message on the server. This is the only sequence diagram you need to complete for this design. You may wish to complete sequence diagrams for the other aspects of this phase and I will look at them, but they are not required.

Make sure your sequence diagram and class diagram are consistent. If you indicated that a method will be used in your sequence diagrams, but it doesn't appear in your class diagram then you will lose points.

Your documents must be electronic!

Submit your diagrams on Blackboard as pdf or image files (this is preferred over the html file that you have to zip). Please do not zip the files together.

Protocols

Provide protocols that show the communication between the client and the server for each of the three pieces you are required to implement that involve the server (sending, receiving, and searching). These are the messages that the client and server will send to each other of the input and output streams created from the sockets. It is suggested that you follow the format of the examples that will be presented in class.

These may be hand-written and a picture can be submitted on Blackboard.

GUI Picture

Add the ability to your GUI to send and receive messages. Submit a picture of the pieces of your GUI where this will occur. You can code this in your project and take a picture of the GUI when you run it, or if you aren't ready to do that you can just draw the GUI on paper and take a picture

of that. If your GUI has multiple windows you only need to submit the pieces relevant to sending/receiving.

Submit your picture(s) on Blackboard.