

# Assignment 6: Chat Server

CMPS 109 - Winter 2003

Ira Pohl

# General Specs

- Write a program that will function as a chat server.
- Allow multiple users to connect
- Any message sent by one user is broadcast to all other users
- Make a GUI with an area to type a message and an area to display the conversation.

# The Client

- ④ No threads needed.
- ④ User specifies server address, server port, and login name
  - ④ Command line
  - ④ GUI Dialog

# The Client

- A JTextField is provided for the user to enter new messages.
- A JTextArea is provided to display the overall conversation.

# The Server

- ④ Needs to create threads
  - ④ One listens for new connections.
  - ④ Client handling threads
    - ④ One thread per user
    - ④ One thread per X number of users (allows more connections).
- ④ Messages from a client are broadcast to all others.
- ④ Connections are broadcast to all users.

# The Server

- ④ Needs a list of connected users
  - ④ Access to the list must be guarded by thread locks.
- ④ A GUI showing the state of the server is optional.
  - ④ The list of connected users.
  - ④ The conversation history.

# Messages

- ④ Client to Server messages
  - ④ Login: sends username
  - ④ Post: sends a new message
  - ④ Logout: tells the server to remove this client
- ④ Server to Client messages
  - ④ Login accept: acknowledge client and username
  - ④ Login reject: sends rejection and reason
  - ④ UserConnected: reports a new client
  - ④ Message: a message sent by a client

# Design Hints

- ④ Message superclass
- ④ Share message classes between client and server (use a sub-package)
- ④ Make a text-only version first!
- ④ When broadcasting a message, make a copy of the current client list

# Steps

- ④ 1) Write a server.
- ④ 2) Make the client connect.
- ④ 3) Make the server accept or reject the client.
- ④ 4) Make the client send something to the server (random text?)

# Steps

- ⑤ 5) Make the server broadcast to all connected users.
- ⑥ 6) Write the client GUI.

Questions  
and  
Answers