Distributed Collaboration - Assignment 1: 1-User and N-User IM

Date Assigned: Aug 23, 2012

1-user IM Completion Date: Aug 31, 2012

N-user IM Completion Date: Sep 7, 2012

Objectives:

- Implement a collaborative application..
- Understand Remote Method Invocation.
- Understand User Interface Toolkits

In this assignment you will implement an N-user instant messaging tool. We will make some simplifying assumptions about the UI and the collaboration features to make your task easy. Initially you will implement a 1-user IM and later extend it to N-users using remote method invocation.

The assignment is broken two multiple parts to allow you to incrementally build it. To make sure you do not procrastinate – there are two completion dates.

The assignment involves the use of Java RMI and the Java Swing/AWT toolkit.   Therefore you need to program in Java. Use the web to find tutorials on how to use RMI and the toolkit.

1-User IM

Implement a “single-user IM tool,” which seems like an oxymoron. This tool will be much like the single-user “Frost poem” tool we saw in class except that will be a GUI rather than a console-based tool. The UI should contain:

1. an editable message widget (e.g a JTextField or TextField) to enter the next message.
2. a non-editable history widget (e.g. JTextArea or TextArea) to show the history of messages entered using the message widget. A message is added to the history widget when the user presses ENTER.
3. one or more widgets (e.g. a JCombobox or Combobox) to set the user status to busy, idle and other choices you can dream of.
4. an editable topic widget (e.g. text widget) to display the topic of the IM (e.g. Recite a poem).

5. a non-editable editing status field that indicates if the user (a) has not entered any text in the message field, (b) is actively editing the message field, or (c) has entered some text but is not actively editing it. You will need to make assumptions about the minimum time between character insertions when the user is actively editing.

1-User IM

Create an N-user version of the tool. The user-interface will be the same as before except that:

1. the history widget will show the messages entered by all of the users.
2. the contents of the topic widget are displayed to and can be edited by all of the N users. Moreover, users see incremental changes, that is, each insertion and deletion. The ENTER key has no special meaning for this widget.
3. N user status and editing status fields are displayed, one for each user. A user can edit only his user status field.

The identities, locations and other properties of the collaborating users should not be known before the collaboration. Moreover is, any user should be able join the IM. It is up to you to decide how a latecomer should be accommodated.

Architecture and other Constraints

Carefully design the distributed architecture (the set of processes created and the communication among them) and the software architecture of each process (the classes in the program run by the process).

Submission

Byn midnight August 31, submit a link to a YouTube video showing your 1-user IM tool in action.

By midnight Sep 7, submit a link to a YouTube video showing your N-user IM tool in action. You can create all of the distributed processes on one computer. In addition, submit a document describing and defending the software and distributed architecture.