

Programmierpraktikum

(Michael Barth, Axel Rauschmayer)

Aufgabe 2 (12 Punkte)

Entwickeln Sie ein Chatprogramm, welches beliebig vielen Personen erlaubt, miteinander, über Texteingaben, zu kommunizieren. An einem Loginserver kann sich jede Person anmelden. Dies geschieht, indem diese Person einen graphischen Chatclient startet, der auf einem bestimmten (nicht notwendig dem gleichen) Rechner und an einem bestimmten Port versucht, mit dem Loginserver Kontakt aufzunehmen. Der Loginserver erzeugt für den ersten Client einen Chatserver. Jeder folgende Client wird mit dem bereits existierenden Chatserver verbunden.

Der Benutzer darf seinen Nicknamen für die Anmeldung am Chatserver frei wählen. Der Chatclient erlaubt dem Benutzer die Eingabe einer Textzeile über die Konsole, die dann zum Chatserver übertragen und von diesem an alle Clients verteilt wird. Alle Clients zeigen diese neue Nachricht und seinen Absender graphisch in einem Textfeld an. Das Textfeld soll den Ablauf der ganzen Kommunikation darstellen (Textfeld mit Scrollbar). Die gesamte Anwendung muss synchronisiert (threadsafe) implementiert werden.

1. Entwickeln Sie je ein Protokoll:
für den Client (`ProtokollClient.txt/ProtokollClient.jpg`)
und den Server (`ProtokollServer.txt/ProtokollServer.jpg`).
2. Implementieren Sie eine Klasse `StartChat.java`, die den Loginserver realisiert, am Port horcht, den Chatserver erzeugt und die Verbindung mit jedem neuen Client zu diesem weiterleitet.
3. Implementieren Sie eine Klasse `ChatServer.java`, die den Chatserver realisiert.
4. Implementieren Sie eine Klasse `ChatPartner.java`, die, als eigener Thread, das serverseitige Ende des Sockets verwaltet.
5. Implementieren Sie eine Klasse `ChatClientInput.java`, die auf der Clientseite die Verbindung hält und Eingaben, über den `ChatPartner` zum `ChatServer` übermittelt.
6. Implementieren Sie eine Klasse `ChatClientOutput.java`, die (clientseitig - als eigener Thread im Hintergrund) von anderen Teilnehmern ankommende Nachrichten in der graphischen Oberfläche ausgibt.

Abgabe bis Mittwoch, den 1.11.2006, 22.00 Uhr über UniWorx.
Geben Sie ein Zip-Archiv mit den Dateien ihrer Lösung ab.