

BAI4-RN	Praktikum Rechnernetze	HBN/SLZ
WS 2015	Aufgabe 2 – Entwicklung eines „Chat“ – Systems in Java	Seite 1 von 1

Aufgabenstellung

Entwickeln Sie in Java ein „Chat“-System (siehe <https://de.wikipedia.org/wiki/Chat>) mit den folgenden Komponenten:

Ein **Chat-Client**, der dem Benutzer folgende Funktionen anbietet:

- Beitritt zu einem Chat-Raum mit beliebig vielen anderen Benutzern (Clients) durch Angabe
 - ... eines Chat-Servers (Hostname) zur Angabe des Chat-Raums
 - ... eines eigenen Chat-Namens (frei wählbar)
 → Anmeldung am Chat-Server
- Eingabe einer beliebigen Textzeile in ein Eingabefenster. Nach Abschluss einer Eingabezeile wird der eingegebene Zeilentext sofort an alle anderen Benutzer (Chat-Clients desselben Chat-Raums) geschickt.
- Anzeige des Protokolls der Chat-Session: Alle Chat-Zeilen (inkl. Chat-Namen), die der Chat-Client von den anderen Benutzern (Clients) erhalten hat (inkl. der eigenen Zeilen) – in chronologischer Reihenfolge.
- Verlassen des Chat-Raums → Abmeldung vom Chat-Server

Ein **Chat-Server**, der lediglich die Aufgabe hat, einen Chat-Raum verwalten (die Liste aller aktuell angemeldeten Chat-Clients zu führen und jedem Client auf Anfrage zur Verfügung zu stellen).

1. Skizzieren Sie für diese Anwendung eine Architektur mit entsprechender Begründung, warum Sie für welche Kommunikationsaufgabe welches Transportprotokoll einsetzen.
2. Spezifizieren Sie für Chat-Clients und Chat-Server die notwendigen Befehle bzw. Formate für geeignete Anwendungsprotokolle inkl. aller Rahmenbedingungen (z.B. Portnummern, Zeichencodierung, Anfragehäufigkeit, nötige Fehlermeldungen, ...).
3. Erstellen Sie eine Spezifikation (Aufgabe der Klasse und jeder Methode) für jede verwendete Java-Klasse!
4. Implementieren Sie die Chat-Server- und die Chat-Client-Komponente in Java.
5. Testen Sie Ihre Implementierung mit einem Chat-Server und mindestens zwei Chat-Clients auf zwei verschiedenen Rechnern.

Tipps

- Orientieren Sie sich bei der Implementierung am Beispielcode aus der Vorlesung!
- Setzen Sie mehrere Threads ein, wobei jeder Thread genau eine spezifizierte Aufgabe im Rahmen eines Anwendungsprotokolls besitzt.