Praktikum Rechnernetze

Aufgabe 3 – Chat-System Fortsetzung

Spezifikation des Chat-Anwendungsprotokolls

Inhalt

[1. Abstract 3](#_Toc499664754)

[2. Grundlegende Technische Konzepte 3](#_Toc499664755)

[3. Genereller Session Aufbau 3](#_Toc499664756)

[4. Befehle Server 4](#_Toc499664757)

[5. Befehle Client 4](#_Toc499664758)

# Abstract

Dieses Dokument ist eine Spezifikation unseres Anwendungsprotokolls für einen einfachen Chat. Orientiert haben wir uns dabei an dem SMTP-Protokoll der RFC’s.

Unser Anwendungsprotokoll heißt „*Simple Chat Server Communication Protocoll*“ (SCSCP) und soll den Austausch von Nachrichten innerhalb eines Chatrooms zwischen mehreren Clients gewährleisten.

Das SCSCP ist komplett textbasiert, weshalb auf Server Seite einige Voraussetzungen erfüllt sein müssen, damit z.B. Clients keine Server-Befehle senden können.

# Grundlegende Technische Konzepte

Der Transport der übergebenen Nachrichten erfolgt über TCP. Dadurch ist ein Datenaustausch in beide Richtungen möglich. Bei den zu übertragenden Datenobjekten handelt es sich ausschließlich um Strings.

Das Protokoll ist ein Pull/Push-Protokoll, da der Client einmal eine Verbindung zum Server aufbaut, die solange bestehen bleibt. Der Server ist jedoch auch in der Lage durch einen „äußeren“ GUI Eingriff oder als Reaktion auf bestimmte Vorkommnisse, die Verbindung zu einem Client zu trennen (KICK <username>). Der Server informiert alle User, die im jeweiligen Chatroom sind, über neue Aktionen der User. Der Server „pusht“ sozusagen alle Neuigkeiten in den Chatroom.

Die Kommunikation läuft über den Port 9000. Der User befindet sich standardmäßig nach dem Login in dem Chatroom „lobby“. Anschließend kann der User auch in einen der anderen Standard-Räume mit dem Befehl „GOTO\_ROOM“ wechseln. Die Standardräume, die zur Auswahl stehen, sind die Chaträume „music“ und „gossip“.

# Genereller Session Aufbau

Alle gesendeten Nachrichten sind Textnachrichten und bestehen immer aus einer Zeile, die mit "\n" beendet wird.

Nachdem der Server gestartet ist, horcht dieser auf ankommende Anfragen von Clients. Solange die Anzahl der angemeldeten Clients bei der Anmeldung eines neuen Clients die 50 nicht überschreitet, wird eine Verbindung zum Client aufgebaut. Wenn dies nicht der Fall sein sollte, sendet der Server folgendes Kommando: „>>002\_<optionale Nachricht>". Damit zeigt er, dass der Server voll ist und keine Verbindung zum Chat möglich ist. Der Client sollte im besten Fall damit reagieren, sich selbst mit einer geeigneten User-Nachricht zu beenden.

Anschließend wird der User aufgefordert, einen Usernamen einzugeben. Dies geschieht mit dem Befehl: „<<300“.

Die nächste vom Client gesendete Textnachricht wird automatisch als Nutzername interpretiert. Sollte nun der Nutzername bereits vergeben sein, sendet der Server die Nachricht „>>001\_<optionale Nachricht>“. Wenn der Name ungültig ist und nicht in die Regularien des Servers passen sollte, erhält der Client folgende Server-Nachricht: „>>000\_<optionale Nachricht>>“ (z.B. „>>000\_Usernamen müssen mindestens drei Zeichen lang sein“). Falls eine der oben genannten Fehler-Meldungen/-Kommandos (>>000\_,>>001\_) vom Server übermittelt worden sind, fragt der Server bis zu fünf Mal nach einem neuen und gültigen Usernamen. Im Falle einer erfolglosen Beendigung aller Schleifendurchgänge, sendet der Server folgenden Befehl: „>>>>601\_<optionale Nachricht>“, mit dem er den Client veranlasst, die Verbindung zum Server zu beenden. Dieser Befehl sollte unabhängig davon vom Client jederzeit als eine Aufforderung zum Verbindungsabbruch interpretiert werden.

# Befehle Server

Server Kommandos

|  |  |
| --- | --- |
| <<300 | Befehl an Client, den Usernamen zu senden. |

Den Client-Zustand ändernde Server Kommandos

|  |  |
| --- | --- |
| >>>>601\_<optionale Nachricht> | Zwingt den Client zum Verbindungsabbruch (Disconnect). |

Positive Server Antworten

|  |  |
| --- | --- |
| >>100\_<optionale Nachricht> | Der Username des Clients ist gültig. |
| >>101\_<optionale Nachricht> | Der User wurde erfolgreich eingeloggt. |

Negative Server Antworten

|  |  |
| --- | --- |
| >>000\_<optionale Nachricht> | Der Username ist ungültig. |
| >>001\_<optionale Nachricht> | Der Username wird bereits verwendet. |
| >>002\_<optionale Nachricht> | Der Server ist voll. |

# Befehle Client

Nachdem der User erfolgreich eingeloggt wurde, kann er folgende Befehle an den Server senden. Dieser sollte darauf automatisch mit einer passenden Nachricht antworten.

|  |  |
| --- | --- |
| /commands | Listet alle dem Client zur Verfügung stehenden Befehle auf. |
| /rooms | Listet alle existierenden Chaträume auf. |
| /goto<leerzeichen><room> | Gehe in den Chatraum <room>. |
| /users | Zeigt alle im jeweiligen Raum befindenden User. |
| /whisper<leerzeichen><username> | Wechselt in den Flüstermodus mit User <username> |
| /unwhisper | Deaktiviert den Flüstermodus, sodass alle ausgehenden Nachrichten wieder für alle User des gleichen Chatrooms sichtbar sind. |
| /log | Zeigt das aufgezeichnete Protokoll aller den User betreffenden Nachrichten dieses Chatraumes. |
| /quit | Beendet die Chat-Session und schließt die Streams und Socket. |