

Dokumentation TW Mailer Basic

Architecture:

Two Tier Architecture. Client und Server sind über einen Socket miteinander verbunden, welcher vom Server erstellt wird.

Der Client kommuniziert seine Commands an den Server. Dieser empfängt diese und wertet mittels einer Folge an if else statements aus, welche Anfrage gestellt wurde. Die Commands werden vom Server zeilenweise eingelesen und gespeichert bis es vervollständigt wurde. Kann der Anforderung nachgekommen werden, so sollte der Server mit einem OK antworten, sonst mit ERR.

Nachrichten werden in .txt Files gespeichert, welche das Subject der Nachricht als Namen bekommen. Diese werden in die jeweiligen Ordner gelagert, in die sie gehören. Ein Ordner wird für jeden Empfänger einer Nachricht angelegt. Somit findet man in jedem Ordner alle Nachrichten, welche der jeweilige User empfangen hat.

Used Tech:

Der Code ist in C++ geschrieben. Server und Client kommunizieren über `send()` und `recv()`.

Für das Speichern, Suchen und Löschen der Nachrichten wurde die filesystem Library verwendet.

Für das Handling des Buffers werden größtenteils Methoden der C String Library verwendet. Für das clientseitige Einlesen wird `fgets()` verwendet.

Development strategy and needed adaption:

Client und Server Programmierung wurde aufgeteilt. Ziel war clientseitig die Eingaben auf ihre Richtigkeit zu prüfen (Username muss mit dem RegEx `[a-z0-9]{1,8}` übereinstimmen, Subject darf nur maximal 80 Zeichen lang sein). Nach erfolgreicher Validierung der Eingabe sollte diese an den Server geschickt werden. Dies führte jedoch unerwartet zu einer falschen Ausgabe auf dem Server. Beispielsweise wurde vom Client die Zahl 1 eingegeben, beim Server kam jedoch 11 an.

Ein weiteres Problem war das Management des Buffers. Zeitweise wurden vom Server mehr Zeichen an den Client geschickt als erwartet wurden. Beispielsweise wurde ein LIST Command vom Server nur mit OK beantwortet und bei einem darauffolgenden Command wurde die Liste vom Server geschickt. Bei anderen Versuchen empfing der Client die richtige Ausgabe, jedoch hängte sich daran noch ein OK an.

Es wurde versucht eine Funktionalität zu implementieren, welche es möglich macht, besonders lange Nachrichten segmentiert an den Server zu schicken. Da dies nicht mehr rechtzeitig funktionsfähig war, wurde die Funktionalität auskommentiert.