

Protokoll

TW-Mailer Basic | Raphael Pusam, Gerald Steinbrugger

Verwendung

1. make-Befehl verwenden (`make all`)
2. Server starten (in `/bin/` mit `./server`)
3. Client starten (in `/bin/` mit `./client`)
4. Kommunikation über client-Terminal
5. Beenden mit make-Befehl (`make clean`)

Architektur

Aufteilung in `myclient.cpp` und `myserver.cpp`

Server wartet auf Verbindung.

Client verbindet sich mit Server (PORT 6543 ist hard coded).

Client erfragt Input und enthält Logik zur Identifikation zu Befehlen, um beispielsweise zwischen SEND und READ unterscheiden zu können, damit Anzahl der Inputs berücksichtigt werden kann.

Nachdem Client alle Inputs erhalten hat, verschickt er sie als einen gemeinsamen String an Server.

Server teilt Input-String wieder auf und bearbeitet anhand der Logik die Calls.

Speichern der Nachrichten passiert in Ordner `/mail-spooler/`

Darin wird für jeden Nachrichtenempfänger ein neuer Ordner erstellt, in dem die empfangenen Nachrichten als Files mit dem Namen

`<username-sender>_<subject>.txt`

erstellt werden.

Technologien

Programmiersprachen C und C++

C hauptsächlich durch die Vorlagen (client/server-Beispiel), restlicher Code (Business Logik) besteht aus C++, weil für unser Team vertrauter durch PROGR in Semester 2. Erlaubt außerdem Kombination mit C.

Daten- und Versionenmanagement durch Git

Entwicklungsstrategie

API first-Ansatz: Kommunikation zwischen Client und Server sehr stark von client/server-Beispiel inspiriert, Formatierung der Calls und Responses vorgegeben, daher konnten Client und Server unter Berücksichtigung der Vorgaben fast unabhängig voneinander implementiert werden.

Nach API wurden die einzelnen Befehle hinzugefügt.