Programmierprojekt

Lastenheft

Chat-Programm

Anforderungen:

* Chatoberfläche
* Chat jeweils zwischen 2 Personen
* Mehrere Chats parallel führen, mit Kontaktliste
* Nachrichten können gespeichert/geladen werden
* Nicht versendete Nachrichten werden bei nächster Gelegenheit versendet
* Eigene Nachrichtenklasse & weitere Klassen
* Ein Nutzer soll auch mehrere Nachrichten nach einander schicken können
* Dokumentation
  + Anwender/Technisch

Erweiterungen:

* Gruppenchats
* Anhänge verschicken (z.B. Bilder)
* Verlorene Nachrichten minimieren, indem der Empfänger antwortet, dass er die Nachricht erhalten hat. Und der Sender die Nachricht nochmals schickt, wenn keine Empfangsbestätigung kommt.

Pflichtenheft

Nachrichtenklasse:

Eigenschaften:

* Nachricht (string)
* Sender (string/Zahl evtl beides/evtl. MAC-Adresse)
* Empfänger (wie Sender)
* Zeitpunkt (Datum und Uhrzeit DateTime)
* Abgeschickt (Bool)

Oberfläche (An Whatsapp orientiert):

* Auswahlmöglichkeit für Chats
* Schriftfeld
* Anzeige für Chatverlauf (Noch mehrere Gedanken dazu evtl. Farben oder eingerückt)
* Senden-Button
* Kontakt hinzufügen Button

Software:

* Peer-to-Peer (UDP-Protokoll)

Dokumentation

* Anwender/Technisch
* Theorie über unterschiedliche Bereiche:
  + Threads
  + Protokolle (UDP)
  + IP-Adressen
  + Ports

Quelle:

https://www.dreamincode.net/forums/topic/231058-peer-to-peer-chat-advanced/

Probleme:

* Benutzernamen können doppelt vorkommen. Dann können Nachrichten doppelt zugestellt werden. Mögliche Lösung: neben dem Namen auch den Erstellungszeitpunkt des Benutzernamen zur Identifikation verwenden

Arbeitspakete:

* Chat für 2 Personen zum Funktionieren bringen in Konsole
* Grafische Oberfläche gestalten (J.)
* Punkt 1 in GUI (L.)
* Erstellung der Klasse Nachricht
* Speicherung der Nachrichten beim Schließen
* Laden der Nachrichten beim Programmstart
* Nachrichten speichern wenn nicht beide Online sind und erneutes Senden
* 2. Chat einfügen parallel
* Kontaktbuch einfügen
* Grafische Oberfläche programmieren
* Antwort des Empfängers bei angekommener Nachricht

Tatsächliches Vorgehen:

* Chat für 2 Personen zum Funktionieren bringen
* Grafische Oberfläche
* Nachrichtenklasse
* Eigenes Protokoll zum Versenden von „Nachrichten“
* Userklasse
* UsernameDialog
* Schönes Darstellen der Nachrichten (Nachrichtenverlauf)
* Senden mit Enter
* Kontaktliste + Kontakt erstellen + Darstellung
* Sender in Klasse auslagern

Ausblick:

* Empfänger auslagern (Herr Bauer fragen)

To-Do:

* Zwischen Chats wechseln -> Speicher
* Problem: Luca schreibt Jacqueline zuerst: Nachrichten von J. werden bei J nicht angezeigt.

Herr Bauer:

* Seine Anforderungen
  + Programm
  + Doku (wie ausführlich?)
* Bewertungskriterien
* Unser bisheriger Ansatz
* Probleme:
  + Doppelter Username
  + Funktioniert bei J nicht