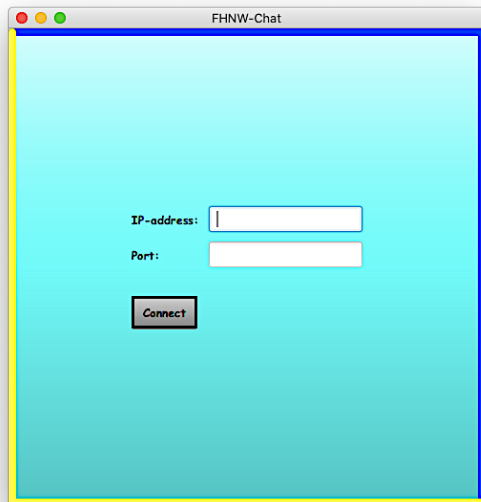


Mini-Projekt 2: Chat

<https://github.com/AliYuerek/CPM2.git>

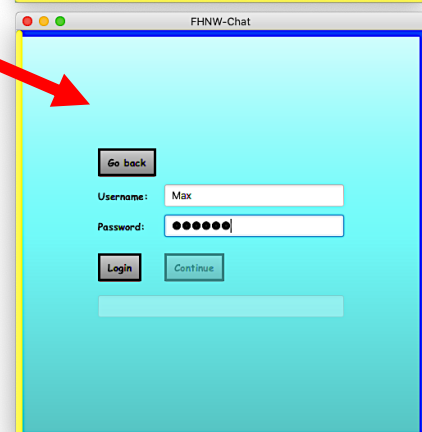
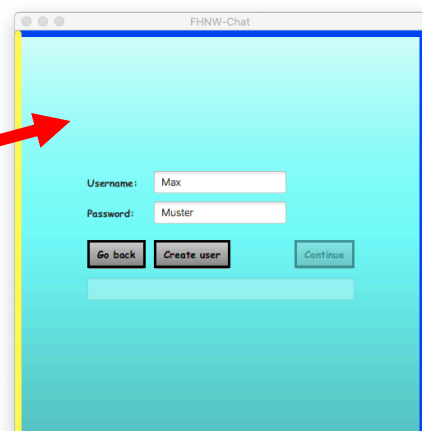
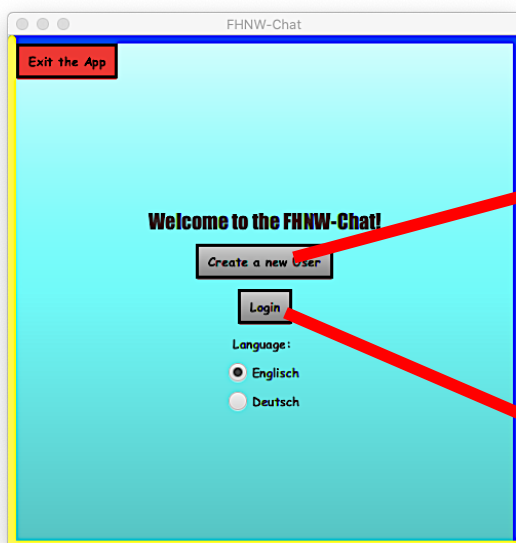
Dieses Programm wurde von mir, Ali Yürekirmaz, unter Verwendung der Vorlage von Bradley Richards weiterentwickelt. Das Programm ist wie folgt aufgebaut:

SCHRITT 1: Serververbindung



Der Benutzer gibt die IP-Adresse: 147.86.8.31 und die Port-Nummer: 50001 in die Felder ein und klickt auf den Button „Connect“, um eine Verbindung zum Server herzustellen. Wenn keine Verbindung hergestellt wird oder falsche Eingaben gemacht werden, erscheint ein Warnfenster.

SCHRITT 2: User erstellen oder Login

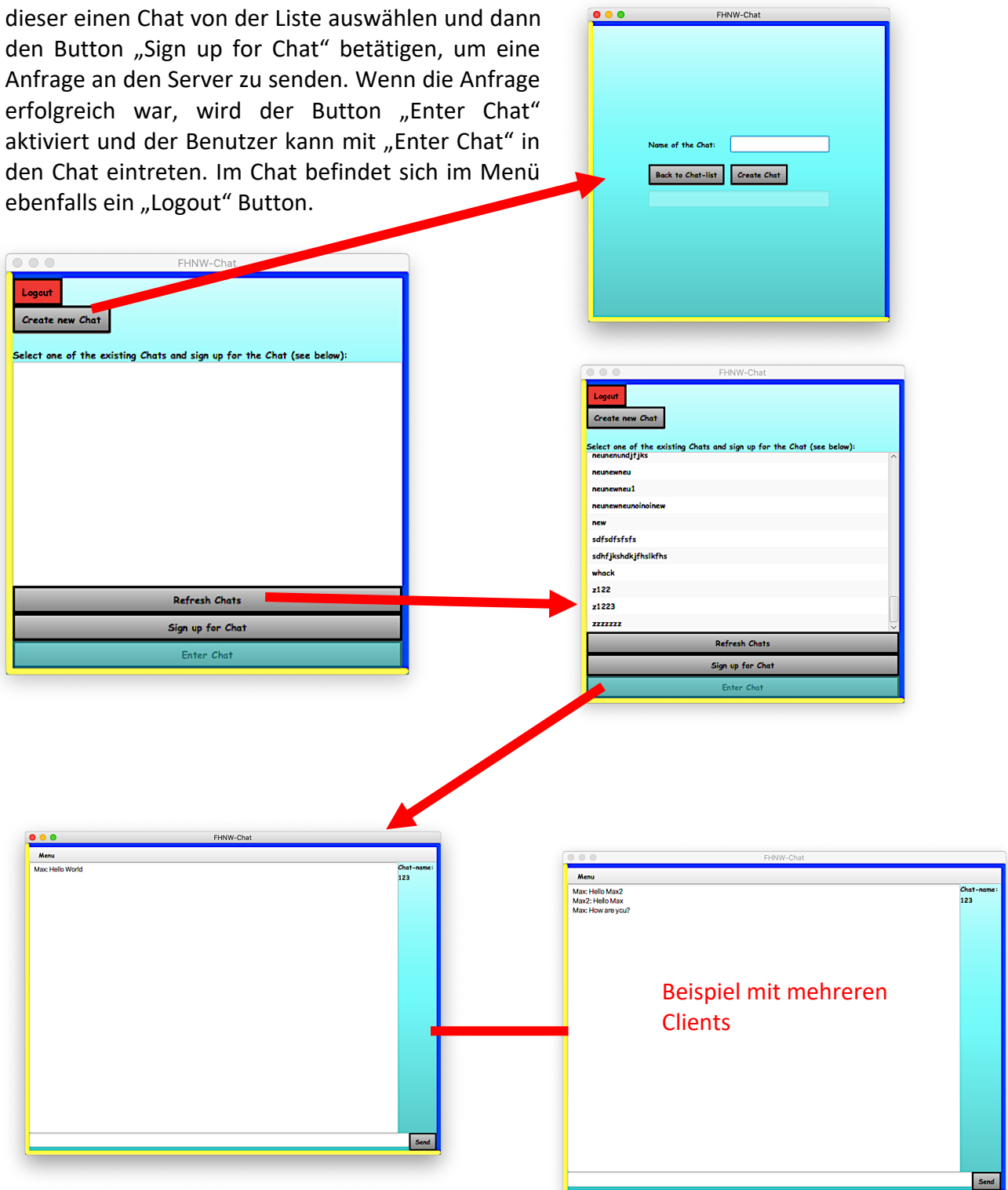


Im zweiten Schritt hat der Benutzer die Möglichkeit, entweder ein Konto zu erstellen oder sich direkt zu registrieren. In beiden Fenstern gibt es Textfelder und Buttons, die auf die Rückmeldungen des Servers reagieren und den Benutzer zwingen, entsprechend zu handeln. Erfolgt in den Feldern keine Eingabe, erscheint wieder ein Warnfenster. War die Eingabe erfolgreich, so werden die Buttons aktiviert.

SCHRITT 3: Chat erstellen oder Chat beitreten

Nun hat der Benutzer die Möglichkeit, entweder einen neuen Chat zu erstellen oder auf den Button „Refresh Chats“ zu klicken und alle aktuellen Chats in einer Liste zu sehen. Wenn der Benutzer einen neuen Chat erstellt hat, muss er die Liste erneut aktualisieren, um die aktuellen Chats zu sehen. **ACHTUNG: Der Benutzer muss zweimal auf den Button „Refresh Chats“ klicken.** Der Benutzer kann sich auch abmelden mit dem Button „Logout“ und befindet sich somit im Login-Fenster (siehe Schritt 2).

Wenn der Benutzer einen Chat betreten möchte, muss dieser einen Chat von der Liste auswählen und dann den Button „Sign up for Chat“ betätigen, um eine Anfrage an den Server zu senden. Wenn die Anfrage erfolgreich war, wird der Button „Enter Chat“ aktiviert und der Benutzer kann mit „Enter Chat“ in den Chat eintreten. Im Chat befindet sich im Menü ebenfalls ein „Logout“ Button.



In diesem Projekt habe ich viel Wert auf die GUI und ihre Gestaltung gelegt. Dadurch wurden die folgenden Ziele erreicht (siehe grüne Markierung):

Required features

- **Administrative**
 - Set server address and port, test connection with "ping"
 - Create new login
 - Login
 - Logout
- **Chat room**
 - List chatrooms, join a particular chatroom
 - See messages in the chatroom
 - Send a message to the chatroom

Optional: GUI features

- **Implement a *nice* GUI**
 - Nice design, good use of color
 - Good error messages
 - Sensible resizing behavior
 - All controls enabled/disabled when it makes sense
- **"Wow" factor**
 - Subjective...
 - For example: animations for user feedback

Required features

- **MVC structure: Separate the UI from the logic**
- **Multilingual application: at least 2 languages**
 - One language is English or German, the other is your choice
- **Reasonable user interface**
 - Not totally ugly
 - Platform independent (test on Windows/Mac)
 - User errors do not crash the program
- **Controls enable/disable as appropriate**
 - Example: If not logged in, you cannot send a message