

TeamSpeak Administration Tool

Anwendung zum Verwalten von
TeamSpeak 2-Servern

Dokumentation einer Projektarbeit

**für Fachinformatiker der
Fachrichtung Anwendungsentwicklung**

Vorgelegt von
Marco Jäger

Für die
Klara-Oppenheimer-Schule Würzburg

Schuljahr 2008/09, Klasse 12 FI 1

Inhaltsverzeichnis

1. Projektdefinition.....	1
1.1. Einführung: TeamSpeak 2.....	1
1.2. Projektinitiierung: Telnet.....	1
1.3. Problemanalyse.....	1
1.4. Lösungsansatz.....	2
1.5. Projektziele.....	2
1.5.1 Funktionen.....	2
1.5.1.1 Auslesen von Daten.....	2
1.5.1.2 Funktionen zum Eingreifen.....	3
1.5.1.3 Funktionen zum Ändern von Datenbanken.....	3
1.5.2 Ergonomie.....	4
1.5.3 Features.....	5
1.6. Systemvoraussetzungen.....	5
2. Projektplanung.....	5
2.1. Analyse der Kommunikation per Telnet.....	5
2.1.1 Befehle	5
2.1.2 Empfangen von Tabellen.....	6
2.2. Aufbau der Anwendung.....	6
2.2.1 Telnet-Kommunikations-Schicht.....	6
2.2.2 Business Logic.....	6
2.2.3 Benutzeroberfläche.....	6
3. Projektrealisierung.....	7
3.1. Analyse.....	7
3.1.1 Befehle.....	7
3.1.2 Empfangen von Tabellen.....	8
3.2. Erstellung der Anwendung.....	8
3.2.1 Telnet-Kommunikations-Schicht.....	8
3.2.1.1 Telnet-Verbindung.....	8
3.2.1.2 Verwaltung der Telnet-Funktionen.....	9
3.2.1.3 Fehlerbehandlung.....	9

3.2.1.4 Klassen zur Datenrückgabe.....	9
3.2.2 Business Logic.....	10
3.2.2.1 Klassen zum Halten von Daten.....	10
3.2.2.2 Handler.....	10
3.2.2.3 Verwaltung der Objekte.....	11
3.2.2.4 Neuladen von Daten.....	11
3.2.3 Präsentationsschicht.....	11
3.2.3.1 Hauptfenster.....	12
3.2.3.2 Logindialog.....	12
3.2.3.3 Bannliste.....	13
3.2.3.4 Anzeige von Datenbanken.....	13
3.2.3.5 Anzeigeoptionen.....	14
3.3. Test der Anwendung.....	15
3.3.1 Telnet-Schicht.....	15
3.3.2 Business Logic.....	15
3.3.3 Benutzeroberfläche.....	17
3.3.3.1 Hauptfenster.....	17
3.3.3.2 Login.....	17
3.3.3.3 Bannliste.....	17
3.3.3.4 Datenbanken.....	18
3.3.3.5 Anzeigeoptionen.....	18
3.3.3.6 Fehlermeldungen.....	18

1. Projektdefinition

1.1. Einführung: TeamSpeak 2

TeamSpeak 2 (TS2) ist eine Server-Client-Anwendung, die darauf ausgerichtet ist, dass sich mehrere Personen über das Internet unterhalten können. Es muss ein TeamSpeak 2-Server eingerichtet werden und jeder, der mitreden möchte, muss sich den TeamSpeak 2-Client runter laden und installieren.

Auf einem TS2-Server können mehrere Channels und Subchannels erstellt werden. Alle Personen, die sich in einem Channel befinden, hören sich gegenseitig.

Es gibt verschiedene Rechte, die man auf einem TS2-Server haben kann. Die hier interessanten sind Superserver-Administrator (SSA) und Server-Administrator (SA).

Ein TS2-Server kann über den Client, über eine Weboberfläche, als auch per Telnet administriert werden.

1.2. Projektinitiierung: Telnet

Per Telnet stehen im Vergleich zu den beiden anderen Zugriffsmöglichkeiten, die meisten Funktionen zu Verfügung. Außerdem sind bestimmte Features ausschließlich über Telnet aufrufbar.

Die Befehle sowie deren Syntax sind für den Menschen jedoch schwer zu merken. Die Ausgabe von vielen Daten ist teilweise unübersichtlich.

Auch ist es umständlich, wenn bestimmte Aktionen, zum Beispiel zu einem User ausgeführt werden soll. Über verschieden Befehle muss sich immer erst die User-ID und unter Umständen noch die davon abweichende Datenbank-User-ID geholt werden.

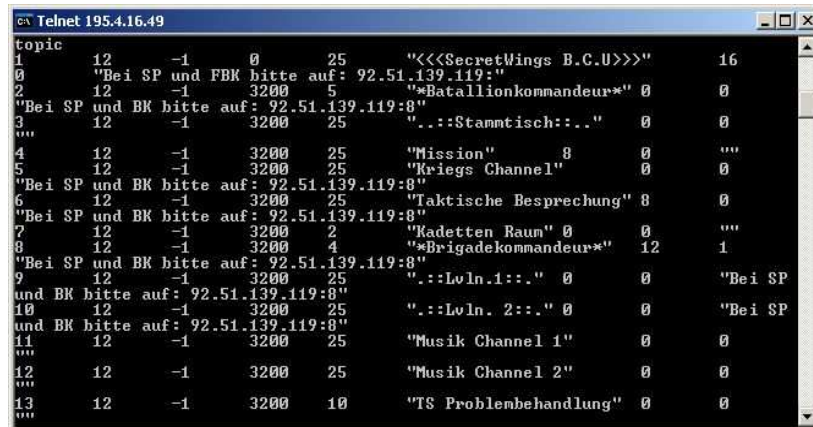
Diese Gründe führten zur Initiierung des vorliegenden Projekts.

1.3. Problemanalyse

Die Verwendung von Telnet zur Administration eines TS2-Servers ist für den Menschen ziemlich unhandlich.

- Unübersichtliche Ausgabe von Tabellen
- Unter Umständen sehr lange Befehle

- Teilweise sehr gewöhnungsbedürftige Syntax der Befehle
- Korrektur mit Backspace nicht möglich



```

Topic      12      -1      0      25      "<<<SecretWings B.C.U>>>"      16
0      "Bei SP und FBK bitte auf: 92.51.139.119:"      0
2      12      -1      3200      5      "*Battalionkommandeur*"      0
0      "Bei SP und BK bitte auf: 92.51.139.119:8"      0
3      12      -1      3200      25      "...:Stammtisch:..."      0
0
4      12      -1      3200      25      "Mission"      8      0
5      12      -1      3200      25      "Kriegs Channel"      0
0      "Bei SP und BK bitte auf: 92.51.139.119:8"      0
6      12      -1      3200      25      "Taktische Besprechung"      8      0
7      12      -1      3200      25      "Bei SP und BK bitte auf: 92.51.139.119:8"      0
8      12      -1      3200      4      "Kadetten Raum"      0      0
0      "Bei SP und BK bitte auf: 92.51.139.119:8"      0
8      12      -1      3200      4      "*Brigadekommandeur*"      12      1
9      12      -1      3200      25      "...:Lvln.1:..."      0
0      "Bei SP und BK bitte auf: 92.51.139.119:8"      0
10     12      -1      3200      25      "...:Lvln. 2:..."      0
0      "Bei SP und BK bitte auf: 92.51.139.119:8"      0
11     12      -1      3200      25      "Musik Channel 1"      0
0
12     12      -1      3200      25      "Musik Channel 2"      0
0
13     12      -1      3200      10     "TS Problembehandlung"      0
0

```

Rückgabe der Channels eines TS2-Servers per Telnet

1.4. Lösungsansatz

Gefordert wird ein Programm, mit dem über eine grafische Oberfläche bequem auf die Telnet-Funktionen eines TS2-Servers zugegriffen werden kann. Dieses soll für den Benutzer die Kommunikation mit dem Server übernehmen. Außerdem soll eine übersichtliche Ausgabe der zurückkommenden Daten erfolgen.

1.5. Projektziele

1.5.1 Funktionen

Das Programm soll eine Verbindung zu jedem beliebigen TS2-Server per Telnet aufbauen können. Der Port soll vom Benutzer frei wählbar sein. Sowohl ohne Login, als auch mit Server- oder Superserver-Rechten soll ein Login möglich sein.

1.5.1.1 Auslesen von Daten

Um mit dem Programm arbeiten zu können, müssen Daten von dem Server ausgelesen werden können. Folgende Informationen sind hier wichtig:

- Channelstruktur
- Angemeldete User und Channel in dem sie sich befinden

- Alle vorhandenen Informationen zu jedem Channel
- Alle vorhandenen Informationen zu jedem User
- User-Datenbank
- SSA-Datenbank
- Bannlist
- Offene Telnet-Verbindungen

1.5.1.2 Funktionen zum Eingreifen

Per Telnet lassen sich sehr viele Aktionen ausführen, die sich sofort auf die angemeldeten User auswirken. Folgende Funktionen müssen von dem Programm unterstützt werden:

- Nachrichten schreiben
- Erlauben der Registrierung
- Vergeben von SA-Rechten
- Verschieben von Usern
- Disconnecten (trennen) von Usern
- Kicken und bannen von Usern
- Festsetzen von Usern

1.5.1.3 Funktionen zum Ändern von Datenbanken

Ein TS2-Server verwaltet intern eine Datenbank. Per Telnet stehen die Grundfunktionen zum Verwalten von einzelnen Tabellen zur Verfügung. Folgende Möglichkeiten sollten durch das Programm gegeben sein:

- User-Datenbank
 - Auslesen
 - Anlegen von Datensätzen
 - Löschen von Datensätzen
 - Ändern von Passwörtern
 - Vergeben von SA-Rechten

- SSA-Datenbank
 - Auslesen
 - Anlegen von Datensätzen
 - Löschen von Datensätzen
 - Ändern von Passwörtern
- Bannliste-Datenbank
 - Auslesen
 - Anlegen von Datensätzen
 - Löschen von Datensätzen

1.5.2 Ergonomie

Da das Programm leicht zu bedienen sein soll und auch Personen, die es nicht kennen, sich schnell darin zurechtfinden, sollen folgende Kriterien im Bezug auf die Benutzerfreundlichkeit und Ergonomie erfüllt sein:

- Ähnliche Benutzeroberfläche und Bedienung wie der TS2-Client
- TreeView-Element zum Darstellen der Channels und User
- Intuitive Bedienung soll möglich sein
- Funktionen sollen per Kontext-Menü aufgerufen werden können
- Darstellung der Tabellen (User, SSAs, offene Telnet-Verbindungen) durch ein DataGridView
- Bei Standard-Einstellung nicht die Anzeige aller Informationen
- Vorbelegung der Eingabefelder mit den Standard-Ports
- Verwendung von SplitContainer so, dass der Benutzer selbst die Größe der einzelnen Teile bestimmen kann

1.5.3 Features

Da manche Aktionen nur mit sehr großem Aufwand möglich sind, sollen durch Kombination weitere Features verfügbar sein.

- Das Verschieben mehrerer User gleichzeitig
- Möglichkeit, dass der Benutzer selbst entscheiden kann, welche Daten zu einem User oder Channel angezeigt werden
- Suche nach Usern
- Möglichkeit zum Speichern von Benutzernamen und Passwort
- Automatisches Speichern beim Schließen der Anwendung
 - Zuletzt eingegebene Serverdaten
 - Größe des Hauptfensters
 - Größe der einzelnen Datenbank-Dialoge
 - Einstellungen der SplitControls

1.6. Systemvoraussetzungen

Das Programm wird als Windows-Forms-Anwendung in Microsoft .NET 3.5 entwickelt. Dementsprechend muss zur Verwendung ein Rechner mit Microsoft Windows 2000, Server 2003, Windows XP oder Windows Vista mit installiertem Microsoft .NET Framework 3.5 vorhanden sein.

2. Projektplanung

2.1. Analyse der Kommunikation per Telnet

Um mit der Programmierung beginnen zu können, muss zuerst die gesamte mögliche Kommunikation mit dem TeamSpeak 2-Server analysiert werden.

2.1.1 Befehle

Um die Telnet-Schicht programmieren zu können, müssen zuerst alle Befehle, die der TS2-Server anbietet auf die Art ihrer Auswirkung als auch auf die Art ihrer Verwendung untersucht werden. Ebenfalls muss die sehr unterschiedliche Syntax genauestens analysiert werden. Auch die Rückgabewerte müssen betrachtet werden.

2.1.2 Empfangen von Tabellen

Da bei bestimmten Befehlen Tabellen zurückgeliefert werden muss die Art der Rückgabe und der Aufbau dieser Tabellen untersucht werden.

2.2. Aufbau der Anwendung

Das Programm soll im Drei-Schichten-Modell aufgebaut werden. Die Schichten sind sehr stark angelehnt an das Modell bei der Programmierung mit Datenbanken.

2.2.1 Telnet-Kommunikations-Schicht

Die unterste Schicht, die normalerweise die Verbindung zur Datenbank verwaltet, soll in diesem Programm die Telnet-Verbindung zum TS2-Server übernehmen. Zum Verwenden der verschiedenen Telnet-Funktionen soll eine Klasse dieser Schicht pro Funktion eine Methode mit den entsprechenden Übergabeparametern bereitstellen. Sollten Fehlermeldungen vom Server zurückgegeben werden, so müssen diese in der Telnet-Schicht in Exceptions umgewandelt werden. Das heißt, für jede mögliche Fehlermeldung muss eine Exception-Klasse angelegt werden. Um Daten von dieser Schicht an eine höhere Schicht weiterzugeben, müssen zusätzlich Klassen angelegt werden.

2.2.2 Business Logic

Diese Schicht soll die gesamte Logik enthalten. Es müssen also hier Klassen für User, Channel usw. erstellt werden. Diese sind aber nicht wie die Klassen aus der Telnet-Schicht nur zum Halten von Daten, sondern halten Referenzen auf die Objekte, anstatt nur IDs. So soll zum Beispiel ein Channel-Objekt eine Liste von seinen Subchannels, als auch eine Liste von den Usern halten, die sich momentan in diesem Channel befinden. Des weiteren sind Klassen für Verwaltung der Rechte zu erstellen.

2.2.3 Benutzeroberfläche

Die oberste Schicht übernimmt die Kommunikation mit dem Benutzer durch eine Windows-Forms-Anwendung. Sie soll die Daten der Business Logic übersichtlich darstellen und einen bequemen Zugriff auf die Funktionen des TeamSpeak 2-Servers erlauben. Die Oberfläche soll an das Layout des TS2-Client angelehnt sein, so dass der Benutzer sich leicht zurechtfindet.

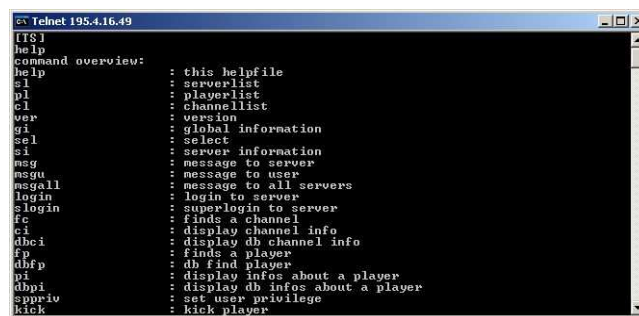
3. Projektrealisierung

Das Programm wurde in Microsoft Visual Studio 2008 mit dem Microsoft .NET Framework 3.5 entwickelt. Auf diesem Rechner wurde auch ein TeamSpeak 2-Server sowie ein TeamSpeak 2-Client zum Testen installiert.

3.1. Analyse

3.1.1 Befehle

Das Herausfinden der verschiedenen Befehle war einfacher als gedacht. Durch `help` erhält man eine Liste mit allen Befehlen und eine kurze Beschreibung zu jedem.



```
Telnet 195.4.16.49
[TS]
help
command overview:
help      : this helpfile
sl        : serverlist
pl        : playerlist
cl        : channellist
ver       : version
gl        : global information
sel       : select
sl        : server information
msg       : message to server
msgu      : message to user
msgall    : message to all servers
login     : login to server
slogin    : superlogin to server
fc        : finds a channel
ci        : display channel info
dci       : display db channel info
fp        : finds a player
dhfp      : db find player
pl        : display infos about a player
dpl       : display db infos about a player
sppriv    : set user privilege
kick      : kick player
```

Rückgabe bei Verwendung des Befehls help

Durch `help befehl` lässt sich für jeden Befehl eine Erklärung mit Beispielen anzeigen. Eine vollständige und kommentierte Liste aller Befehle befindet sich im Anhang.

→ Fehler: Referenz nicht gefunden Seite Fehler: Referenz nicht gefunden

Bei erfolgreicher Ausführung eines Befehls kommt immer die Meldung OK. Bei Fehlern können folgende Fehlermeldungen zurück kommen:

ERROR, not logged in	Es wurde versucht, ein Befehl auszuführen für den ein Login nötig ist
ERROR, invalid paramcount	Falsche Anzahl an Parametern wurde übergeben
ERROR, no server selected	Es wurde noch kein Server ausgewählt
ERROR, invalid permissions	Fehlende Rechte für diese Operation
ERROR, no data available	Keine Daten vorhanden

Your password failed 3 consecutive times, please wait 10 minutes before trying again!	Wegen dreimaliger Falscheingabe des Passwortes wird die IP-Adresse für 10 Minuten gebannt
error	Fehler ohne genauere Definition, zum Beispiel beim Verwenden von nicht definierten Befehlen

3.1.2 Empfangen von Tabellen

Wie oben bereits erwähnt, ist die Ausgabe von Tabellen sehr unübersichtlich. Schaut man sich aber den Aufbau etwas genauer an, so erkennt man, dass die einzelnen Spalten jeweils durch ein Tabulatorzeichen und die einzelnen Zeilen durch einen Wagenrücklauf (Carriage-Return) und einem Zeilenvorschub (line feed) getrennt sind.

Das heißt die Auswertung der Daten ist durch Splitten des empfangenen Strings an den eben genannten Trennzeichen sehr einfach möglich. Die Spalten haben zwar alle eine Überschrift, aber die Reihenfolge wird aus Gründen der Einfachheit hart kodiert. Da TeamSpeak 3 bereits in Entwicklung ist, ist nicht zu erwarten, dass sich die Reihenfolge ändern wird.

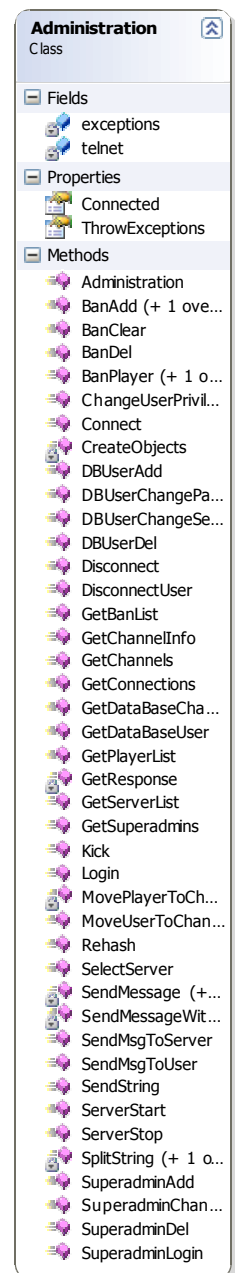
3.2. Erstellung der Anwendung

3.2.1 Telnet-Kommunikations-Schicht

Diese Schicht übernimmt die Kommunikation mit dem TS2-Server und die Kapselung der erhaltenen Daten in Objekte. Diese halten aber noch keine Referenzen auf andere Objekte.

3.2.1.1 Telnet-Verbindung

Um die Verwendung des Sockets zu vereinfachen wurde die Klasse Telnet geschrieben. Diese bietet Methoden zum Verbindungsaufbau und zum Senden von Strings. Nach jedem Senden wird auf eine Antwort vom Server gewartet und diese dann von der Klasse als String zurückgegeben.



3.2.1.2 Verwaltung der Telnet-Funktionen

Um die einzelnen Befehle, die der TS2-Server zur Verfügung stellt, einfach nutzen zu können, wurde die Klasse Administration entwickelt. Diese stellt Methoden zum Senden von Strings, als auch je eine Methode für jeden Befehl des TS2-Servers bereit.

Die Get-Methoden geben ein oder mehrere Objekte zurück. Diese Objekte werden bei jedem Aufruf der Methoden neu erzeugt. Das heißt, die Business Logic hat dafür zu sorgen, dass die Objekte anhand der IDs richtig zugeordnet werden.

3.2.1.3 Fehlerbehandlung

Sollten vom TS2-Server Fehlermeldungen, zum Beispiel aufgrund fehlender Berechtigungen zurückgegeben werden, so müssen diese durch die Schichten bis zur Benutzeroberfläche zurückgegeben werden. Aus diesem Grund wurden folgende Exception-Klassen zur Fehlerbehandlung erstellt:

- InvalidPermissionsException
- LockedFalsePasswordException
- NoDataAvailableException
- NoServerSelectedException
- NotLoggedInException
- ErrorException
- NoConnectionException

3.2.1.4 Klassen zur Datenrückgabe

Wie bereits mehrmals erwähnt, müssen zur Zurückgabe der Daten an die aufrufende Schicht Klassen angelegt werden.

- BanListEntry
- Channel
- ChannelInfo
- DataBaseChannel
- User
- DBUser
- Superadmin

Alle diese Klassen implementieren das Interface DataHolder. Dieses schreibt die Implementierung der Eigenschaft Key vor. Diese sollte immer die eindeutige ID zurückgeben. Somit kann in der Business Logic festgestellt werden, zu welchen bereits bestehenden Objekten die neuen Objekte der Telnet-Schicht gehören.

3.2.2 Business Logic

Die Business Logic ist dafür verantwortlich, die Daten in Objekte zu kapseln. Die IDs aus den Datenbanken sollen durch das Halten von Referenzen auf andere Objekte ersetzt werden.

3.2.2.1 Klassen zum Halten von Daten

Wie bereits erwähnt enthalten die Klassen der Telnet-Schicht lediglich die aktuell abgerufenen Daten. Um mit den Daten im objektorientierten Sinn arbeiten zu können müssen in diese in Objekte gekapselt werden, die Referenzen auf andere Objekte halten.

- Beispiel: Die Klasse Channel hält Referenzen auf die Subchannels sowie auf die User, die sich momentan in diesem Channel befinden.

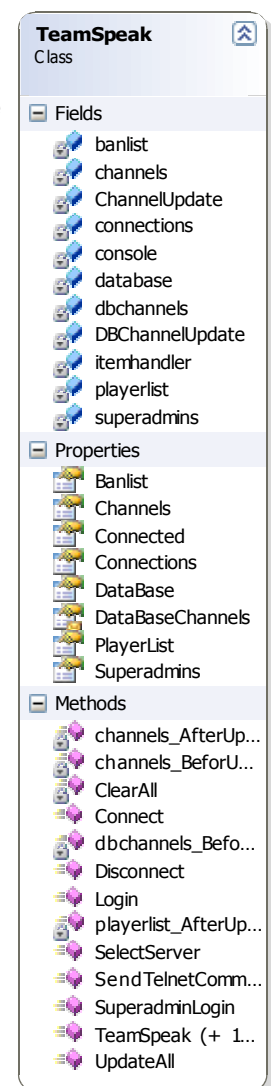
Alle Objekte erben von einer abstrakten Klasse, die vorschreibt, dass eine Methode zum Updaten der Werte implementiert werden muss.

3.2.2.2 Handler

Zum Speichern von mehreren Objekten eines Typs wurden eigene Handler-Klassen angelegt. Normale Lists reichen nicht, da spezielle Methoden zum Beispiel zum Updaten gebraucht werden.

- Beispiel: Objekte von Typ Channel werden von der Klasse Channels verwaltet.

Auch diese Klassen erben von einer abstrakten Klasse. Diese stellt die Grundfunktionen zur Verwaltung von Collections bereit.



3.2.2.3 Verwaltung der Objekte

Um das ganze System zu verwalten muss eine zentrale Klasse existieren. Deshalb wurde die Klasse TeamSpeak erstellt. Sie stellt den Mittelpunkt dieser Schicht da und muss von übergeordneten Schichten als einzige Klasse der Business Logic instanziiert werden. Im Konstruktor dieser Klasse geschieht dann die Erzeugung aller Handler.

Es existieren verschiedene Möglichkeiten sich auf dem TS2-Server einzuloggen. Je nachdem, ob man gar keine Rechte hat oder SA bzw. SSA ist.

Durch Aufruf der Methode UpdateAll werden die Daten durch Verwendung der Telnet-Schicht in die Handler geladen oder aktualisiert. Über die Properties kann darauf zugegriffen werden.

Aufbau der Business Logic siehe Anhang

- Fehler: Referenz nicht gefunden Seite Fehler: Referenz nicht gefunden
- Fehler: Referenz nicht gefunden Seite Fehler: Referenz nicht gefunden

3.2.2.4 Neuladen von Daten

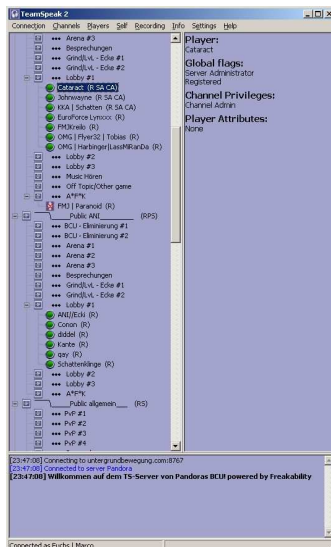
Kommt von einer höheren Schicht der Befehl, die Daten einzelner oder aller Handler zu aktualisieren, so werden von der Telnet-Schicht die neuen Daten angefordert. Es kommen jedoch neue Objekte zurück. Das heißt es existieren jetzt zweimal Objekte, die grundsätzlich die gleichen Daten halten. Nur einmal die noch alten Daten von der letzten Aktualisierung und einmal die neuen Daten. Anhand des Keys, der ja durch die abstrakte Klasse vorgeschrieben ist, können die Handler die Objekte zuordnen und die Daten aktualisieren.

3.2.3 Präsentationsschicht

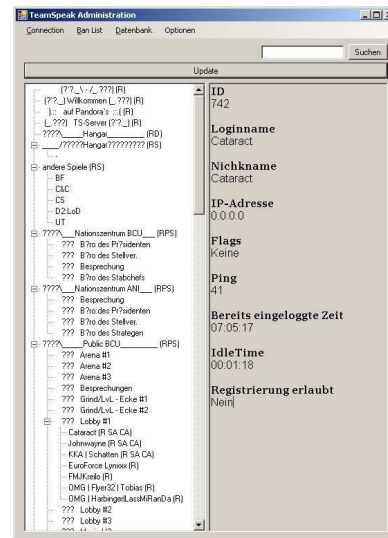
Diese Schicht soll die Kommunikation mit dem Benutzer realisieren. Dazu wurde eine Windows-Forms-Anwendung erstellt. Diese besteht aus einem Hauptfenster, das auch beim Start der Anwendung geladen und angezeigt wird. Zum Anzeigen von Daten wurden dann weitere Forms angelegt, die vom Hauptfenster aus aufgerufen werden können.

3.2.3.1 Hauptfenster

Der Aufbau und die Bedienung des Hauptfenster sollen wie bereits erwähnt an den TS2-Client angelehnt sein. Dies vereinfacht den Benutzern die Verwendung des Programms.



TeamSpeak 2 Client



TeamSpeak Administration Tool

Bilder im Großformat siehe Anhang

- Fehler: Referenz nicht gefunden Seite Fehler: Referenz nicht gefunden
- Fehler: Referenz nicht gefunden Seite Fehler: Referenz nicht gefunden

Bei beiden Programmen sind links die Channels und User in einer Baumstruktur dargestellt. Beim Klicken auf einen User oder Channel werden im rechten Teil Informationen dazu angezeigt.

Zusätzlich existiert oben in der Mitte der Button "Update". Mit diesem ist es möglich, die Daten neu vom TS2-Server zu laden. Ein weiteres Feature ist das Suche-Feld rechts oben, mit dem anhand des Nicknamen nach einem User gesucht werden kann. Auch eine Suche mit einem Sternchen als Platzhalter ist möglich.

3.2.3.2 Logindialog

Um mit der Anwendung arbeiten zu können, muss zuerst eine Verbindung mit dem TS2-Server aufgebaut werden. Beim Login müssen TS2-Serveradresse, TS2-Abfrageport und der Port des Servers, den man ansprechen möchte angegeben werden. Außerdem sind die folgenden Möglichkeiten zum Einloggen gegeben

- Ohne Login
- Login mit Serveradministrator-Rechten (SA)
- Login mit Superserveradministrator-Rechten (SSA)

Des weiteren kann der Benutzer auswählen können, ob Benutzername und Kennwort gespeichert werden sollen.

→ Fehler: Referenz nicht gefunden Seite Fehler: Referenz nicht gefunden

3.2.3.3 Bannliste

TeamSpeak 2 bietet die Möglichkeit IP-Adressen zu bannen. Auch das Verwalten der Bannliste ist per Telnet möglich. Deshalb wurde ein Dialog implementiert, der es dem Benutzer neben dem Anzeigen ebenfalls erlaubt, IP-Adressen hinzuzufügen und zu löschen.

Da bei einer ADSL-Verbindung sich die IP-Adresse normalerweise täglich ändert, macht es nicht viel Sinn sie für immer zu bannen. Deswegen kann der Benutzer beim Hinzufügen entscheiden, ob die IP-Adresse für immer oder nur für eine bestimmte Zeit gebannt ist.

→ Fehler: Referenz nicht gefunden Seite Fehler: Referenz nicht gefunden

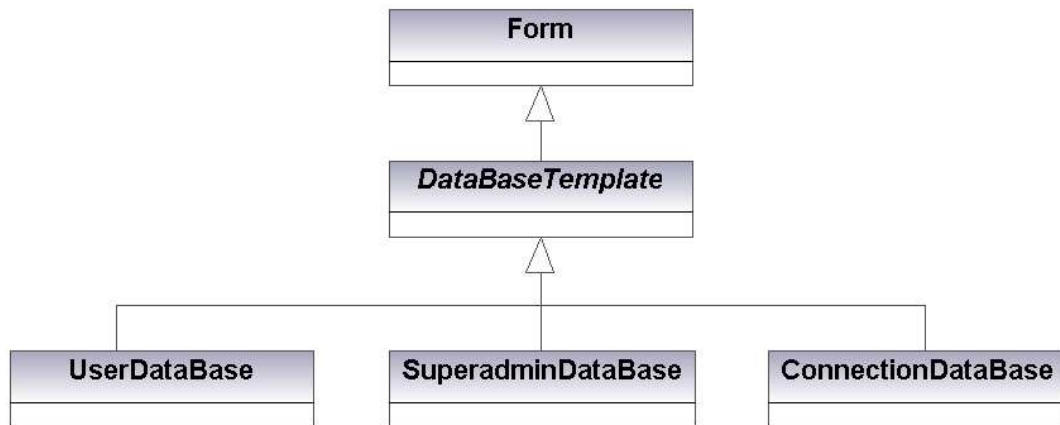
3.2.3.4 Anzeige von Datenbanken

Die User- und Superserveradministratoren-Datenbanken sowie die offenen Telnet Verbindungen werden vom Server in Tabellenform zurückgegeben werden. Um diese einfach und übersichtlich darzustellen wurden DataGridViews verwendet.

Das Dialog-Layout der Anwendung soll bei allen drei Tabellen gleich sein und manche Funktionen gehören in alle drei Tabellen. Aus diesem Grund wurde eine abstrakte generische Klasse DataBaseTemplate erstellt.

→ Fehler: Referenz nicht gefunden Seite Fehler: Referenz nicht gefunden

Die drei Klassen, die hiervon erben, müssen sowohl den Typ des ItemHandler als auch den des Items, das verwaltet wird angeben. Ebenso ist die Implementierung einzelner Methoden vorgeschrieben.



Die Spalten werden in jeder der drei Klassen einzeln erstellt. Ebenso die Methoden zum neu laden der Daten und zum Anlegen, Ändern und Löschen von Datensätzen. Bei der Anzeige der offenen Verbindung entfallen die Möglichkeiten zum Ändern der Datenbank, diese Tabelle kann nur angezeigt werden.

Die User-Datenbank hat noch eine Spezialfunktion. Die Checkboxes, die anzeigen, ob ein User SA-Rechte hat, können angeklickt werden und lösen damit eine sofortige Änderung der Rechte in der Datenbank aus.

3.2.3.5 Anzeigeeoptionen

Wird im Hauptfenster im Tree ein Channel oder ein User angeklickt, so erscheinen im rechten Teil Informationen dazu. Da sowohl zu Channels als auch zu Usern sehr viele Informationen verfügbar sind, wurde ein Menü erstellt, über das der Benutzer steuern kann, welche Daten angezeigt werden sollen. Auch Schriftart, Schriftgröße und Schriftfarbe können in diesem Menü festgelegt werden.

Um festzulegen, welche Werte dem Benutzer zur Auswahl stehen, wurden die öffentlichen Eigenschaften von User und Channel mit dem selbst angelegten Attribut ShowPropertyAttribute versehen. Diese werden dann durch Reflection wieder ausgelesen.

→ Fehler: Referenz nicht gefunden Seite Fehler: Referenz nicht gefunden

Das heißt, die Anzeige im Hauptfenster wird komplett dynamisch aufgebaut.

3.3. *Test der Anwendung*

Um sicherzustellen, dass das Programm zuverlässig und stabil läuft, wurden einige Tests auf allen Schichten durchgeführt.

3.3.1 Telnet-Schicht

Da diese Schicht der Grundbaustein der Anwendung ist, muss sie stabil und fehlerfrei laufen.

Es wurde die Klasse Telnet ausführlich getestet. Verschiedene existierende und funktionierende Befehle, als auch welche bei denen eine Fehlermeldung wegen zum Beispiel fehlenden Berechtigungen kommt, wurden übergeben. Auch die Verhaltensweise bei nicht existierenden Befehlen wurde getestet. Ebenso wurden alle möglichen Fehlermeldungen provoziert.

Ergebnis: Die Telnet-Klasse arbeitet zuverlässig und die Verbindung mit dem TS2-Server ist stabil.

Anschließend wurden alle Methoden der Klasse Administration geprüft und die Rückgabewerte mit den Daten des Servers verglichen. Auch hier wurden sowohl erlaubte, als auch nicht erlaubte Werte übergeben und die Exceptions abgefangen und überprüft, ob sie mit den vom Server erwarteten Fehlermeldungen übereinstimmen.

Ergebnis: Es gab Problemen durch Fehlermeldungen wegen falscher Parameteranzahl. Diese treten auf, wenn die Befehle ausgeführt werden, ohne dass vorher ein Server ausgewählt wurde. Durch entsprechende Programmlogik wurde dieses Problem beseitigt.

3.3.2 Business Logic

Das Testen der Business Logic war etwas komplizierter, da die Beziehungen zwischen den Klassen beachtet werden mussten.

Es wurde versucht die Daten zu laden und sie zu verändern. Ebenfalls wurden die Methoden, die für das updaten der Daten zuständig sind, ausführlich getestet. So wurden sowohl die Daten einzelner Handler neu geladen, als auch alle Daten gleichzeitig.

Ergebnis: Laden und verändern der Daten durch die Methoden funktioniert ohne Probleme.

Problem: Beim neu laden der Channels oder User ohne das jeweils andere upzudaten traten Probleme auf, wenn sich die Channels mittlerweile geändert hatten.

Lösung: Das Programm wurde so angepasst, dass das Neuladen der Daten einer der beiden Handler einen Aufruf der Update-Methode des anderen Handlers zur Folge hat.

Da das Programm in einem Thread abläuft, musste hierbei auf nicht zu lange Laufzeiten geachtet werden, da sonst die Benutzeroberfläche hängen würde.

Dann wurde die richtige Zuordnung der Objekte anhand der ID überprüft. Ebenso das Leeren des Caches, das beim Trennen vom Server geschehen muss. Des weiteren wurden die gesamten Methoden zum Bearbeiten der Datenbanken ausgiebig getestet. Hierbei zeigten sich auch die Auswirkungen von Leer- und Sonderzeichen.

In Black-Box-Tests wurden zuletzt noch einmal alle von außen möglichen Methoden aufgerufen und die Rückgabe der Werte überprüft.

Ergebnis: Die Laufzeiten beim Laden aller Daten befinden sich unterhalb von 3 Sekunden und sind somit akzeptabel.
Leer- und Sonderzeichen verursachen keine Probleme, da in der Anwendung selbst immer über die verschiedenen IDs zugegriffen wird.

Problem: Beim Abrufen der Daten werden alle Sonderzeichen und Umlaute durch Fragezeichen ersetzt. Sollten in dem Namen oder Passwort, mit dem man sich versucht per Telnet einzuloggen, Leer- oder Sonderzeichen sein, so schlägt der Vorgang fehl.

Lösung: Das Problem mit den Leer- und Sonderzeichen konnte nicht gelöst werden, da es eine Eigenheit des TeamSpeak 2-Servers und durch Programmlogik nicht zu umgehen ist.

3.3.3 Benutzeroberfläche

Da man nie davon ausgehen kann, dass der Benutzer stets korrekte und sinnvolle Werte eingibt, müssen beim Testen der Benutzeroberfläche sowohl die erwarteten, als auch die unerwarteten und die komplett sinnlosen Eingaben berücksichtigt werden.

3.3.3.1 Hauptfenster

Im Hauptfenster wurde die richtige Anzeige der Channels und User im Tree überprüft. Ebenso wie der Update-Button und die Anzeige der jeweiligen Informationen auf der rechten Seite. Die Suchen-Funktion wurde auf Servern mit ca. 50 Personen getestet, um sicherzugehen, dass sie auch bei vielen Usern noch richtig und schnell arbeitet.

Außerdem wurden die Funktionen der unterschiedlichen Kontext-Menüs bei Usern und Channels getestet.

Ergebnis: Das Hauptfenster arbeitet ohne Probleme und vollkommen sicher und zuverlässig. Die Suchen-Funktion erledigt auch bei vielen Usern ohne erkennbare Verzögerung ihre Aufgabe.

3.3.3.2 Login

Hier wurde als Test das Einloggen auf verschiedenen Servern mit verschiedenen Rechten ausgeführt und überprüft, ob die Server- und Login-Daten gespeichert werden.

Ergebnis: Der Login-Dialog funktioniert ohne Probleme. Die Server-Daten sind nach dem Neustarten des Programms noch vorhanden und die Login-Daten nur dann, wenn die CheckBoxen zum Speichern gesetzt waren.

3.3.3.3 Bannliste

Bei der Bannliste wurde überprüft, ob IP-Adressen hinzugefügt und gelöscht werden können. Ebenso wurde mit Falscheingaben der reguläre Ausdruck getestet, über den festgestellt wird, ob der Benutzer eine gültige IP-Adresse eingegeben hat.

Ergebnis: Die Bannliste läuft fehlerfrei und das Programm kann auch durch fehlerhafte Eingaben nicht zum Absturz gebracht werden.

3.3.3.4 *Datenbanken*

Bei den verschiedenen Datenbanken wurden die jeweiligen Buttons zum Einfügen, Passwort Ändern, Löschen und Updaten getestet. Bei der User-Datenbank zusätzlich noch die Möglichkeit zum Vergeben und Entziehen von SA-Rechten.

Ergebnis: Alle Funktionen arbeiten wie gewünscht.

3.3.3.5 *Anzeigeoptionen*

Bei den Anzeigeoptionen wurden verschieden Werte ausgewählt und die Eigenschaften der Schrift mehrmals umgestellt. Anschließend wurde geprüft, ob sich die Änderung im Hauptfenster auswirken.

Ergebnis: Die Anzeige im Hauptmenü wird sofort nach Bestätigen des Einstellungen wie gewünscht ausgeführt.

3.3.3.6 *Fehlermeldungen*

Bei fast allen Tests traten wie erwartet Fehlermeldungen auf, die sich durch Popup-Fenster mit der Fehlermeldung bemerkbar machten.

So wurden zum Beispiel beim Verwenden von Funktionen ohne ausreichende Berechtigung Fehlermeldungen angezeigt. Ebenso wie bei dem Versuch, die User-Datenbank anzuzeigen, wenn man sich "Ohne Login" mit dem Server verbunden hatte.

Ergebnis: Das Werfen der Exceptions und das Auffangen durch das Front-End funktionieren problemlos.

Anhang

Anhang

1. Selbstständigkeitserklärung.....	IV
2. Projektantrag.....	V
3. Zeiteinteilung.....	VI
4. Glossar.....	VII
5. TeamSpeak2 Client.....	IX
6. Ausgabe einer Tabelle per Telnet.....	X
7. Telnet-Befehle des TeamSpeak2-Servers.....	XI
8. Klasse Telnet.....	XIV
9. Klassendiagramme der Business Logic.....	XVI
9.1. Vererbung.....	XVI
9.2. Halten der Handler.....	XVII
10. Vergleich: TeamSpeak 2 – TeamSpeak Administration Tool.....	XVIII
10.1. TeamSpeak 2 Client.....	XVIII
10.2. TeamSpeak Administration Tool.....	XIX
11. Dialoge.....	XX
11.1. Login.....	XX
11.2. Bannliste.....	XX
11.3. User-Datenbank.....	XXI
11.4. SSA-Datenbank.....	XXI
11.5. Offene Telnet-Verbindungen.....	XXII
11.6. Anzeigeoptionen.....	XXII
12. Klasse DataBaseTemplate.....	XXIII
13. Klasse ShowPropertyAttribute.....	XXV

1. Selbstständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Hausarbeit mit dem Titel

TeamSpeak Administration Tool

selbstständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe.

Mainbernheim, 18.01.09

Marco Jäger

2. Projektantrag

Projektbeschreibung

Ziel des Projekts ist die Erstellung einer Windows-Anwendung zum Administrieren von TeamSpeak 2.

TeamSpeak 2 ist ein Programm, das es ermöglicht, dass sich mehrere Personen über das Internet unterhalten können. Das Programm hat einen Client der das Sprechen ermöglicht und es kann per Telnet administriert werden. Da sich der Superadmin nicht mit dem Client einloggen kann, muss er eine Telnet-Verbindung mit dem TeamSpeak-Server aufbauen, um aktiv eingreifen zu können.

Das zu schreibende Programm soll eine Windows-Anwendung sein, die dem TeamSpeak 2 Client ähnlich sieht und auf ähnliche Art bedient werden kann. Es soll per Telnet die Befehle ausführen und die empfangenen Daten wieder interpretieren. Das Programm soll ein Drei-Schichten-Modell werden, wobei die unterste Schicht die rein textbasierte Verbindung mit Telnet übernimmt. Das heißt Daten sendet und den empfangenen String wieder interpretiert. Die Business Logic ist zuständig für die Verwaltung des Systems. Die Anwendungsschicht übernimmt die Kommunikation mit dem Benutzer.

Dieser soll hierbei die Möglichkeiten haben, sich ohne Login anzumelden, als Serveradmin oder als Superadmin. Es soll wie im TeamSpeak 2-Client ein Tree mit den Channels und den Usern angezeigt werden. Die Funktionen sollen im Kontextmenü erscheinen. Ebenso sollen die Player per Drag & Drop zwischen den Channels verschoben werden können. Zu dem markierten Player bzw. Channel sollen die aktuellen Daten angezeigt werden, wobei der Benutzer auswählen können soll, welche Daten alles angezeigt werden. Die Datenbanken sollen durch ein DataGridView bereitgestellt werden, wobei die Daten aus der Datenbank genauso so wie die vom Benutzer vorgenommenen Änderungen natürlich wie alles andere auch per Telnet übertragen werden müssen.

Zeitplanung

Analyse der durch Telnet bereitgestellten Funktionen	1 Stunde
Telnet-Verbindung und Methode für jede Funktion	4 Stunden
Business Logic	
Erstellung des Designs	1 Stunde
Programmierung	10 Stunden
Windows-Anwendung	
Erstellung des Designs	1 Stunde
Programmierung	12 Stunden
Einbauen des DataGridViews	2 Stunden
Testphase	3 Stunden
Projekt-Dokumentation	6 Stunden

3. Zeiteinteilung

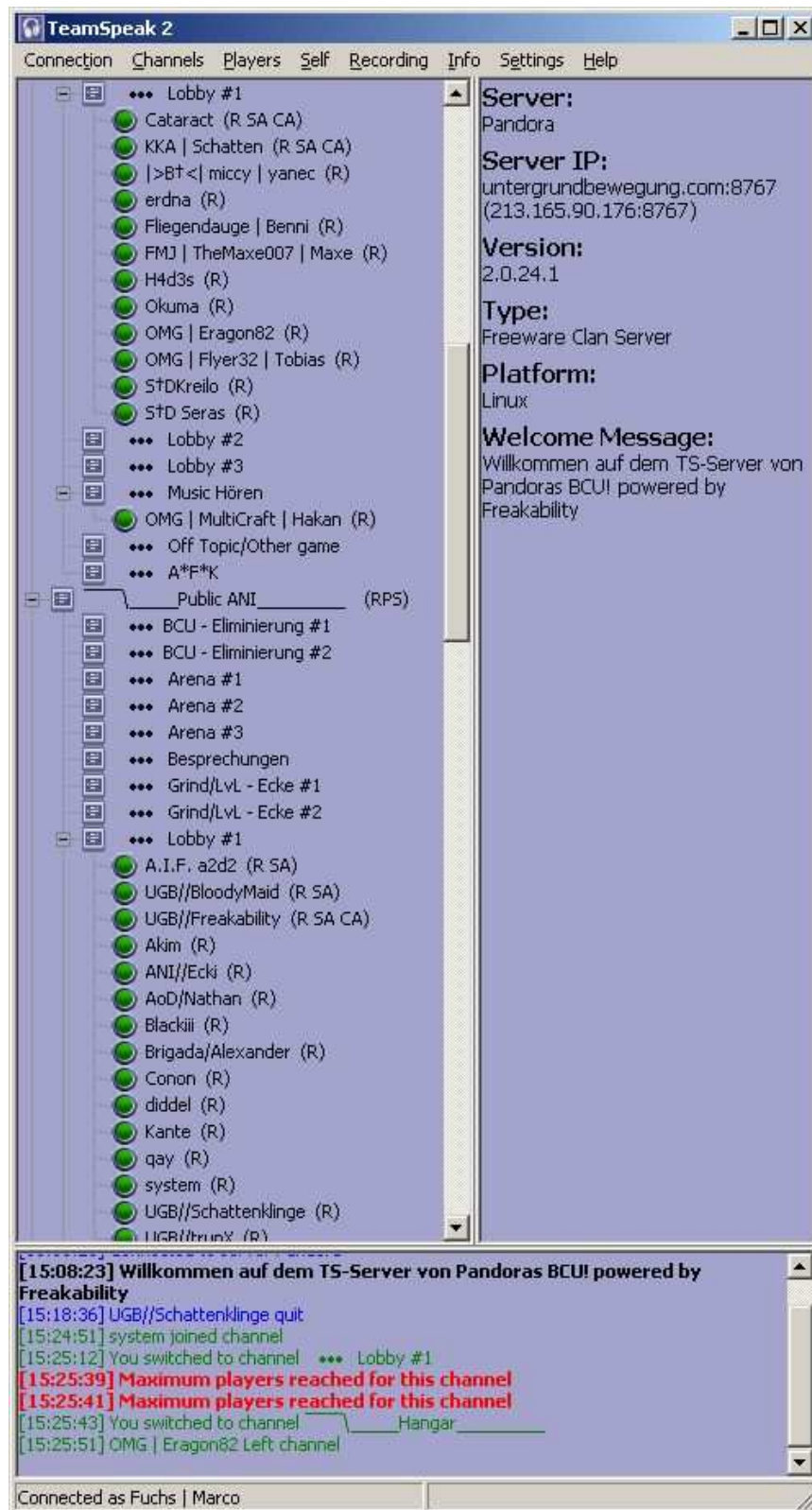
Stunden	Arbeitsschritt
1	Projektplanung
1	Analyse der Telnet-Funktionen
1	Erste Versuche, eine Telnet-Verbindung mit dem TS2-Server aufzubauen
5	Implementierung der Telnet-Kommunikations-Schicht
1	Planung des Klassendesigns der Business Logic
8	Umsetzung der Business Logic
1	Entwurf des Designs der Benutzeroberfläche
1	Entwurf der verschiedenen Dialoge
8	Erstellung und Implementierung des Hauptfensters
5	Erstellung und Implementierung der Dialoge
3	Testen der Anwendung und Beseitigung von Fehlern
5	Erstellung der Projektdokumentation

4. Glossar

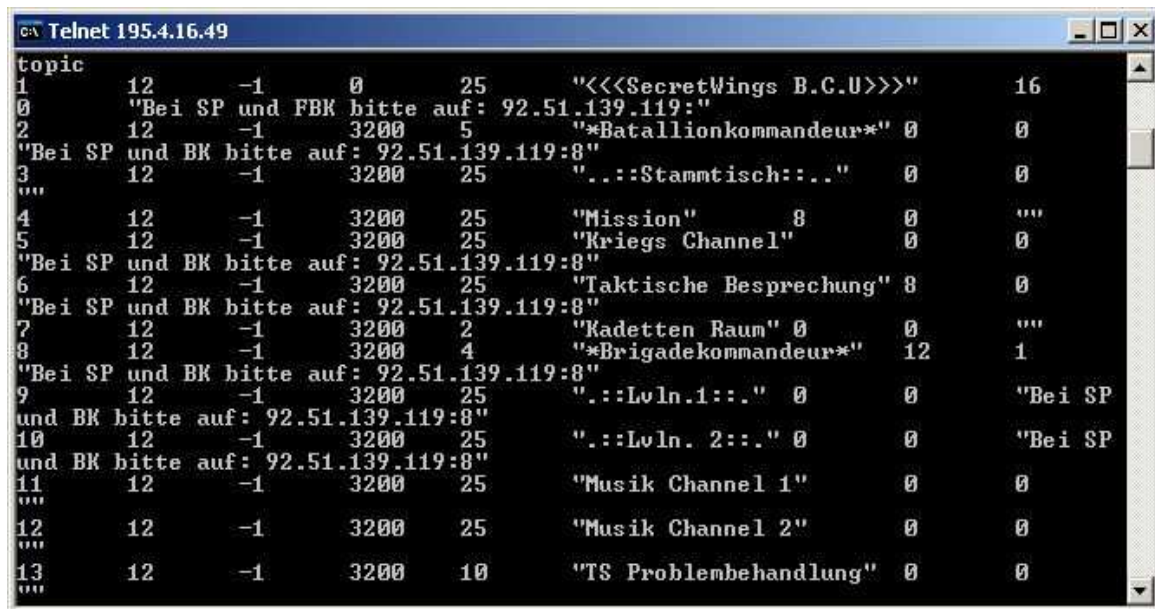
TeamSpeak 2	Programm zum Unterhalten/Telefonieren über ein Netzwerk oder das Internet, bei dem mehrere Personen miteinander reden können
DataGridView	Element in C#, mit dem Daten in Tabellenform dargestellt werden können
TreeView	Element in C#, mit dem Daten in einer Baumstruktur dargestellt werden können
TeamSpeak 2-Server	TeamSpeak 2-Server meint das gesamte System mit allen auf den verschiedenen Ports laufenden Servern
TeamSpeak 2-Client	TeamSpeak 2-Client, mit dem man sich auf einem TS2-Server verbinden kann
Server	Mit Server ist in dieser Dokumentation nur ein Server auf einem Port eines TS2-Servers gemeint
TS2	siehe TeamSpeak 2
Telnet	Ein im Internet weit verbreitetes Netzwerk-Protokoll, das TCP zur Übertragung verwendet
Benutzer	Benutzer bezeichnet in dieser Dokumentation den Benutzer des TeamSpeak Administration Tools; nicht zu verwechseln mit den Usern in TeamSpeak
User	User bezeichnet in dieser Dokumentation Daten, die auf sich auf in TS2 angemeldete User beziehen; nicht zu verwechseln mit den Benutzern des TeamSpeak Administration Tools
Serveradministrator (SA)	Rechte auf einem Server eines TS2-Servers
Superserveradministrator (SSA)	Uneingeschränkte rechte auf dem gesamten TS2-Server
SA	Siehe Serveradministrator
SSA	Siehe Superserveradministrator
kicken	Bezeichnet das hinauswerfen eines Users von einem Server
bannen	Bezeichnet das Eintragen einer IP-Adresse in eine Tabelle, wodurch sich User von dieser IP-Adresse aus nicht mehr mit dem TS2-Server verbinden können
Bannlist	Tabelle mit gebannten IP-Adressen; siehe bannen
Channel	Ein "Raum" auf einem Server; alle User in einem Channel können sich gegenseitig hören

SplitContainer	Element in C#, durch das in Fenster in zwei Teile geteilt wird, von denen der Benutzer die Größe verändern kann
Socket	Eine bidirektionale Software-Schnittstelle zur Netzwerk-Kommunikation
Reflection	Bezeichnet das Auslesen der Klasseninformationen aus den Meta-Daten
abstrakt	Abstrakte Klassen können nicht instanziiert, sie sind lediglich als globale Schnittstelle und zum Wiederverwenden von Programmcode gedacht
generisch	Generisch heißt, dass die Klasse nicht vollständig typisiert ist, will man sie verwenden, so muss man die "fehlenden" Typen angeben
Handler	Wird in dieser Dokumentation von Handlern gesprochen, so sind Klassen gemeint, die von der Klasse ItemHandler erben. Sie halten Referenzen auf andere Objekte und übernehmen die Verwaltung dieser.

5. TeamSpeak2 Client



6. Ausgabe einer Tabelle per Telnet



The screenshot shows a Telnet window titled "c:\ Telnet 195.4.16.49". The window displays a table of channel information. The table has 8 columns: a line number, a numeric ID, a numeric value, a numeric value, a channel name, a numeric value, and a numeric value. The channel names include "SecretWings B.C.U", "Bei SP und FBK", "Bei SP und BK", "Mission", "Kriegs Channel", "Taktische Besprechung", "Kadetten Raum", "Brigadekommandeur", "Lvl n. 1", "Lvl n. 2", "Musik Channel 1", "Musik Channel 2", and "TS Problembehandlung".

Line	ID	Value 1	Value 2	Channel Name	Value 3	Value 4
1	12	-1	0	"<<<SecretWings B.C.U>>>"	16	
0				"Bei SP und FBK bitte auf: 92.51.139.119:"		
2	12	-1	3200	"*Batallionkommandeur*" 0	0	
				"Bei SP und BK bitte auf: 92.51.139.119:8"		
3	12	-1	3200	"...:Stammtisch:..." 0	0	
4	12	-1	3200	"Mission" 8	0	
5	12	-1	3200	"Kriegs Channel" 0	0	
				"Bei SP und BK bitte auf: 92.51.139.119:8"		
6	12	-1	3200	"Taktische Besprechung" 8	0	
				"Bei SP und BK bitte auf: 92.51.139.119:8"		
7	12	-1	3200	"Kadetten Raum" 0	0	
8	12	-1	3200	"*Brigadekommandeur*" 12	1	
				"Bei SP und BK bitte auf: 92.51.139.119:8"		
9	12	-1	3200	"...:Lvl n. 1:..." 0	0	"Bei SP
				und BK bitte auf: 92.51.139.119:8"		
10	12	-1	3200	"...:Lvl n. 2:..." 0	0	"Bei SP
				und BK bitte auf: 92.51.139.119:8"		
11	12	-1	3200	"Musik Channel 1" 0	0	
12	12	-1	3200	"Musik Channel 2" 0	0	
13	12	-1	3200	"TS Problembehandlung" 0	0	

7. Telnet-Befehle des TeamSpeak2-Servers

Befehl	Erklärung vom TS2-Server	Beschreibung
help	help	Anzeige aller Befehle; Anzeige der Hilfe zu einem bestimmten Befehl
sl	serverlist	Auflistung aller Ports, auf denen ein Server läuft
pl	playerlist	Liste aller momentan angemeldeten User, Channel-ID, in dem er sich befindet + Informationen zu den Usern
cl	channellist	Liste aller momentan existierenden Channels + Informationen zu den Channels
ver	version	Serverversion
gi	global information	Globale Informationen über diesen Server
sel	select	Auswahl eines bestimmten Servers (Portes)
si	server informationen	Abrufen von Werten globaler Servervariablen
msg	message to server	Nachricht an alle User eines Server schreiben
msgu	message to user	Nachricht an einen User schreiben
msgall	message to all servers	Nachricht an alle User aller des gesamten TS2-Servers
login	login to server	Login als SA auf einem Server
slogin	superlogin to server	Login als SSA auf dem TS2-Server
fc	finds a channel	Suchen eines Channels anhand des Names
ci	display channel info	Informationen zu einem Channel abrufen; Channel-ID wird gebraucht
dbci	display db channel info	Datenbankinformationen zu einem Channel abrufen; Datenbank-Channel-ID wird gebraucht
fp	finds a player	Suchen eines Users anhand des Nicknames

Befehl	Erklärung vom TS2-Server	Beschreibung
dbfp	db find player	Suchen eines Users in der Datenbank anhand des Login-Names
pi	display infos about a player	Informationen zu einem momentan angemeldeten User anhand der User-ID auslesen
dbpi	display db infos about a player	Informationen zu einem User aus der Datenbank anhand der Datenbank-User-ID auslesen
sppriv	set user privilege	Setzen eines User-Rechtes, bei einem User, der gerade angemeldet ist. Die Wirkung tritt unmittelbar nach Absenden des Befehls auf.
kick	kick player	Kicked einen User vom Server; gebraucht wird die User-ID
removeclient	remove client	Disconnected einen User vom Server; gebraucht wird die User-ID
mptc	move player to channel	Verschiebt einen User in einen anderen Channel; gebraucht wird User-ID und Channel-ID
gapl	get average packet loss	Abrufen, wie viele Prozent der Pakete auf einem Server verloren gegangen sind
serverstart	start server	Startet einen Server
serverstop	stop server	Stoppt einen Server
serveradd	adds a server	Erzeugt einen neuen Server auf einem noch offenen Port
serverdel	deletes a server	Entfernt einen Server
serverset	set a server variable	Setzen von Werten, die pro Server gelten
dbserverlist	list all server variable	Auflistung aller in der Datenbank existierenden Server
checkserverok	checks if server hangs	Prüft, ob ein bestimmter Server hängt
rehash	reload server settings	Neu laden der Servereinstellungen
dbuserlist	display server user list	Anzeige aller User in der Datenbank eines Servers
dbuseradd	adds a user to server	Hinzufügen eines neuen Users in die Datenbank

Befehl	Erklärung vom TS2-Server	Beschreibung
dbuserdel	deletes a user from server	Löschen eines Users aus der Datenbank; Datenbank-User-ID wird gebraucht
dbuserchangepw	changes password of a user	Ändert das Passwort eines Users; Datenbank-User-ID wird gebraucht
dbuserchangeattr ibs	changes rights of a user	Ändert die Rechte eines Users in der Datenbank; Änderungen treten erst in Kraft, wenn sich der User das nächste mal einloggt
dbuserid	display dbid from loginnam	Gibt die Datenbank-User-ID zurück; gebraucht wird die User-ID
dbsuserlist	display superadmin serverlist	Anzeige aller SSAs in der globalen SSA-Datenbank eines TS2-Servers
dbsuseradd	adds a superadmin to server	Hinzufügen eines neuen SSAs
dbsuserdel	deletes a superadmin from server	Löschen eines SSAs; Datenbank-SSA-ID wird gebraucht
dbsuserchange pw	changes password of a superadmin	Ändert das Passwort eines SSAs; Datenbank-SSA-ID wird gebraucht
lc	list all open tcpquery connections	Anzeige aller offenen Telnet-Verbindungen
ki	kick idlers	Kickt User, die eine bestimmte Zeit inaktiv waren
banlist	show banlist	Anzeige der Einträge in der globalen TS2-Server Bannlist
banadd	add ip ban	Hinzufügen einer IP zur Bannlist
bandel	del ip ban	Löscht einen Eintrag aus der Bannlist
banplayer	bans a player	Fügt die IP eines Users zur Bannlist hinzu; User-ID wird gebraucht
globalset	set a global variable	Setzen einer globalen Servervariable
quit	close connection	Beenden der Telnet-Verbindung

8. Klasse Telnet

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Net;
using System.Net.Sockets;
using TeamSpeak_Exceptions;

namespace TeamSpeak_Telnet
{
    class Telnet
    {
        private Socket socket;

        public bool Connect(string host, int port)
        {
            try
            {
                socket = new Socket(AddressFamily.InterNetwork,
                    SocketType.Stream, ProtocolType.Tcp);
                IPAddress address = Dns.GetHostAddresses(host)[0];
                EndPoint ep = new IPEndPoint(address, port);
                socket.Connect(ep);
                if (socket.Connected)
                    socket.Receive(new byte[8]);
                return socket.Connected;
            }
            catch (SocketException)
            {
                return false;
            }
        }

        public bool Connected
        {
            get { return socket != null && socket.Connected; }
        }

        public void Disconnect()
        {
            if (socket != null && socket.IsBound)
                socket.Close();
        }

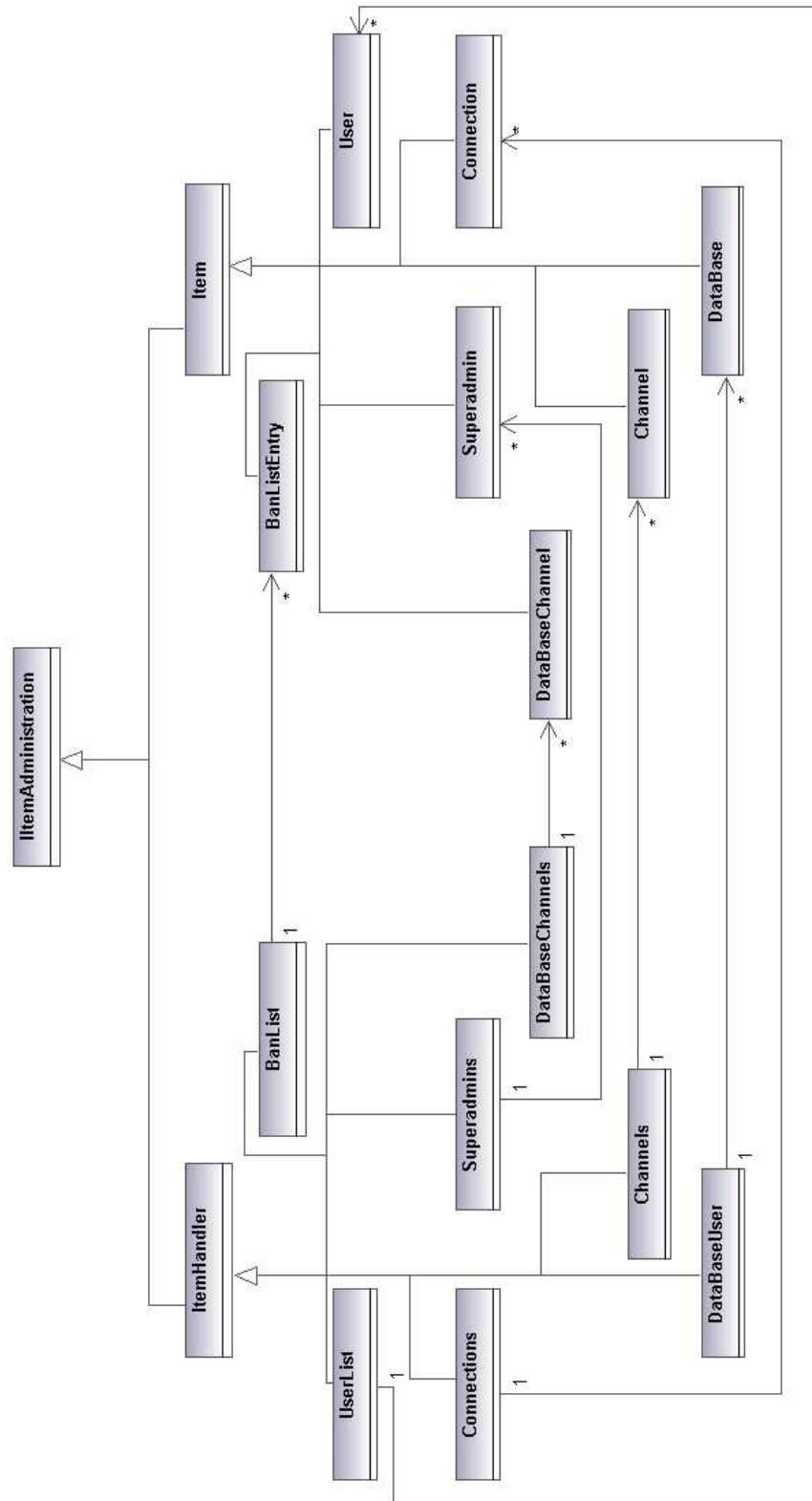
        public string SendMessage(string str)
        {
            if (str.Length == 0)
                return null;
            str = str + "\r\n";
            byte[] msg = Encoding.ASCII.GetBytes(str);
            try
            {
                socket.Send(msg);
            }
            catch (SocketException)
            {
                return null;
            }
        }
    }
}
```

```
        int bytesSent = socket.Send(msg);
    }
    catch
    {
        throw new NotLoggedInException(str);
    }
    return Read();
}

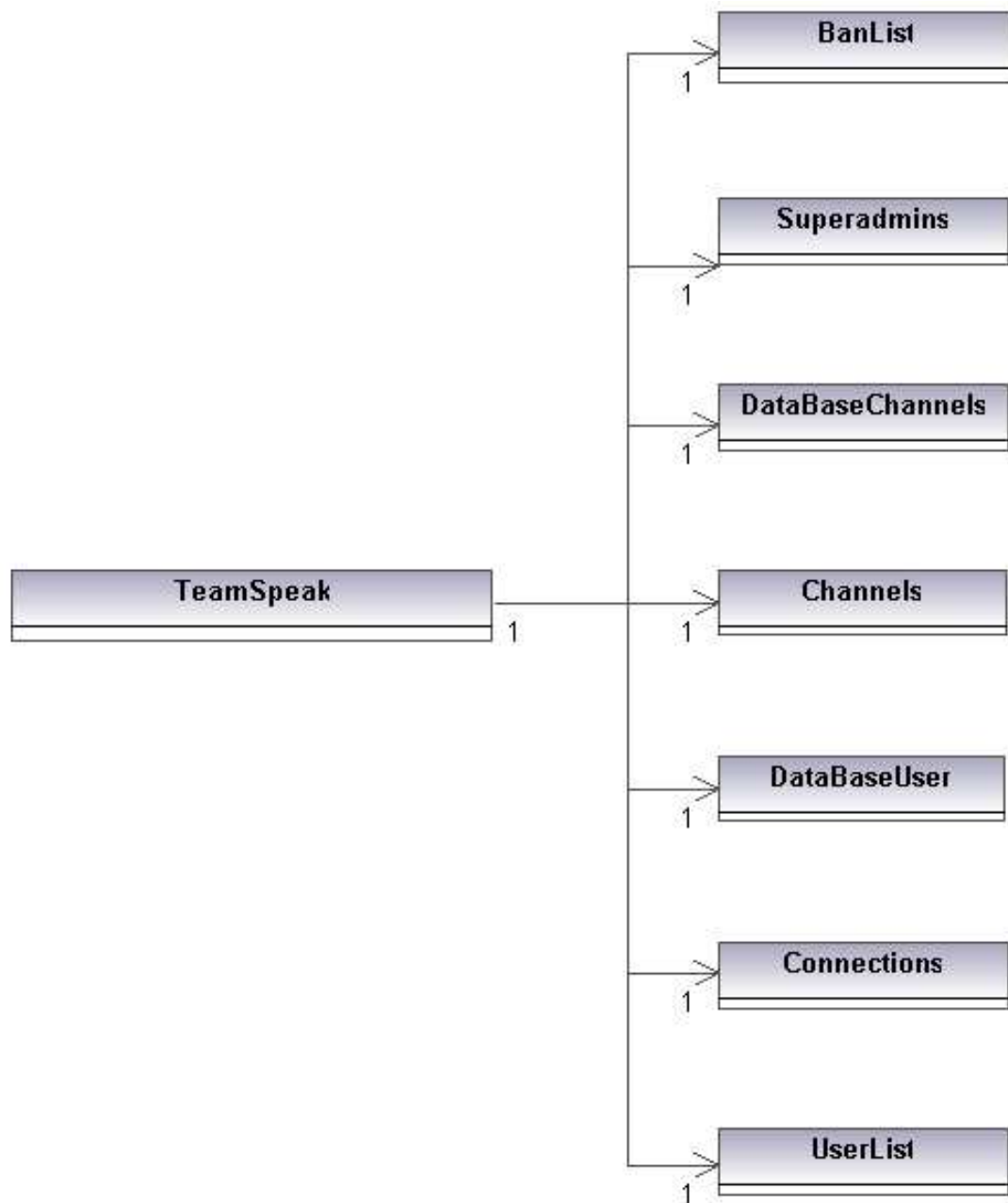
private string Read()
{
    byte[] bytes = new byte[1024];
    int bytesRec;
    string str = string.Empty;
    do
    {
        bytesRec = socket.Receive(bytes);
        str += Encoding.ASCII.GetString(bytes, 0, bytesRec);
    } while (str.Length >= 4 && str.Substring(str.Length - 4) != "OK\r\n"
        && !str.Contains("ERROR") && str != "Your password failed 3
        consecutive times, please wait 10 minutes before trying
        again!\r\n" && !str.Contains(" error") && !str.Contains("\nerror")
        && !
        str.EndsWith("error"));
    return str;
}
}
```

9. Klassendiagramme der Business Logic

9.1. Vererbung

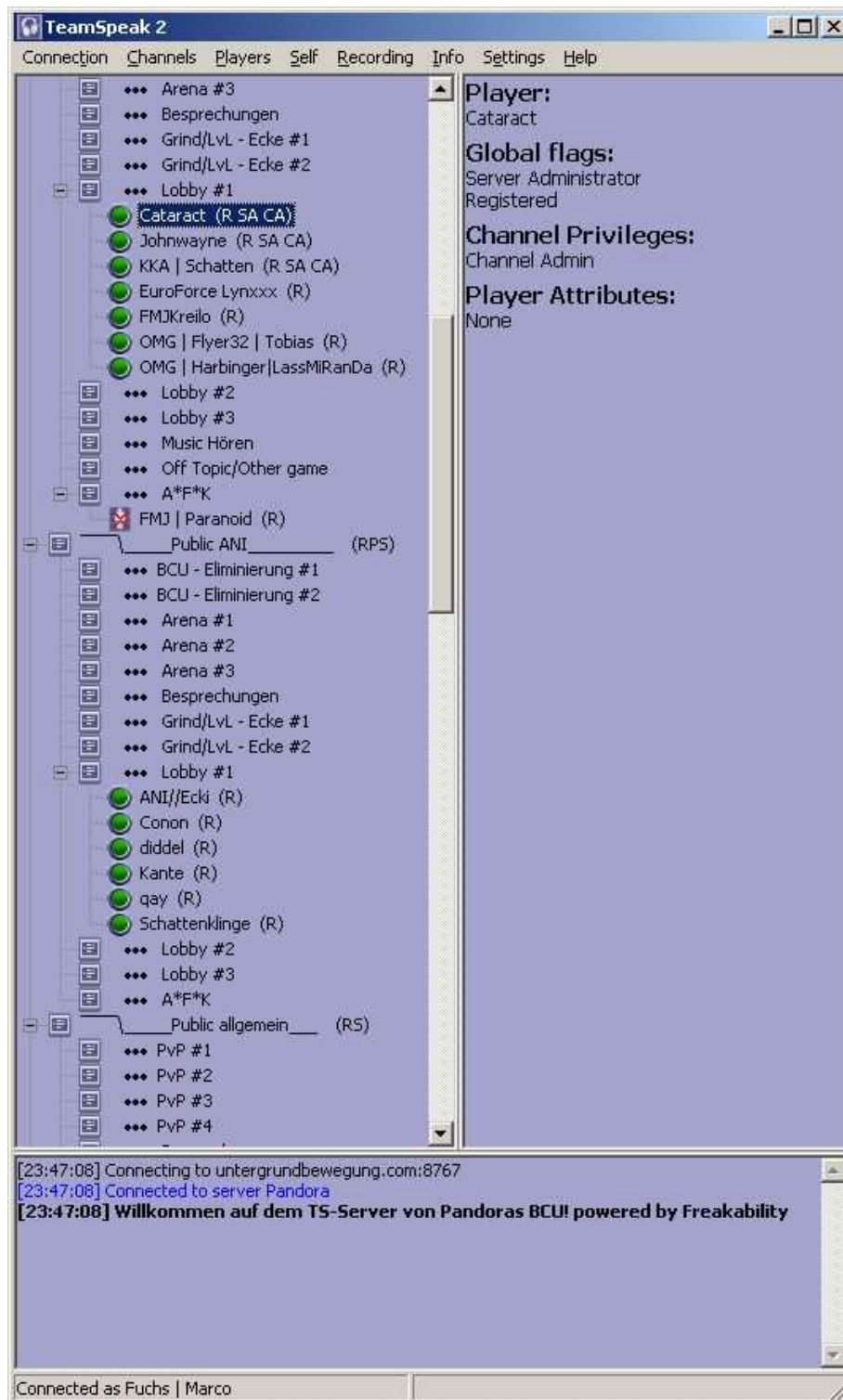


9.2. Halten der Handler

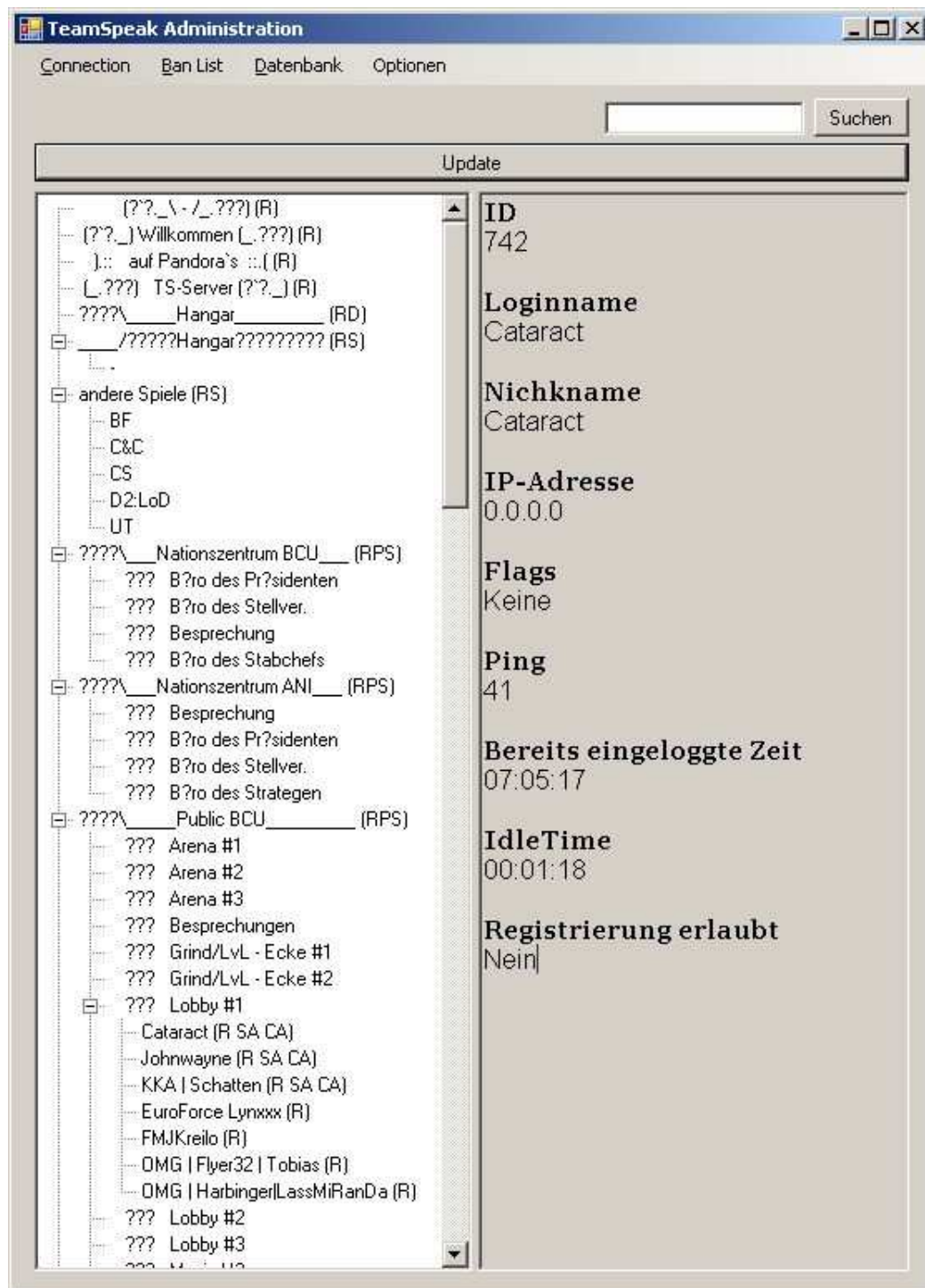


10. Vergleich: TeamSpeak 2 – TeamSpeak Administration Tool

10.1. TeamSpeak 2 Client




10.2. TeamSpeak Administration Tool



11. Dialoge

11.1. Login



The screenshot shows the 'Login' dialog box with the following fields and options:

- Server Adresse: [Empty text box]
- Server Port: [51234]
- Port: [8767]
- Radio buttons: ☒ Ohne Login, ☐ Serveradmin, ☐ Superadmin
- Loginname: [Empty text box] with checkbox ☐ Loginname speichern
- Passwort: [Empty text box] with checkbox ☐ Passwort speichern
- Buttons: OK, Abbrechen

11.2. Bannliste



The screenshot shows the 'BanList' dialog box with a table of banned users and controls at the bottom.

ID	IP-Adresse	Verbleibende Zeit
1	87.80.223.45	unendlich
2	80.86.125.123	unendlich
3	83.53.220.56	02:50:27

Below the table, there is an input field for a new IP address, a dropdown menu for duration, and buttons for 'Hinzufügen', 'Löschen', and 'Update'.

11.3. User-Datenbank

User Datenbank					
	ID	Server Admin	Erstellt	Letzter Login	Name
▶	7622	<input checked="" type="checkbox"/>	22.08.2008 13:24:06	04.01.2009 08:40:54	Day
	7943	<input type="checkbox"/>	10.09.2008 16:45:06	17.01.2009 10:57:24	lordtiger
	7951	<input type="checkbox"/>	11.09.2008 11:27:45	21.10.2008 20:56:04	dow
	7965	<input type="checkbox"/>	13.09.2008 13:32:35	13.10.2008 15:29:50	mooSe
	8038	<input checked="" type="checkbox"/>	19.09.2008 00:18:34	17.01.2009 13:49:36	Fuchs
	8042	<input type="checkbox"/>	19.09.2008 18:14:01	21.11.2008 17:34:00	Grillinho
	8062	<input type="checkbox"/>	21.09.2008 21:28:04	25.09.2008 13:27:52	Razor
	8071	<input type="checkbox"/>	24.09.2008 15:06:36	16.01.2009 17:50:15	Bau
	8134	<input type="checkbox"/>	04.10.2008 23:52:01	25.11.2008 21:21:23	Hyperia
	8135	<input type="checkbox"/>	05.10.2008 00:12:06	21.11.2008 18:44:03	helmchen
	8136	<input type="checkbox"/>	05.10.2008 13:35:48	25.12.2008 09:00:42	GhstAngel
	8447	<input type="checkbox"/>	23.12.2008 12:00:19	23.12.2008 15:42:05	Kues
	8449	<input type="checkbox"/>	23.12.2008 13:30:50	23.12.2008 18:03:54	Hiob
	8487	<input type="checkbox"/>	28.12.2008 14:07:35	16.01.2009 20:27:36	Smoker
	8513	<input type="checkbox"/>	01.01.2009 20:50:52	07.01.2009 19:04:48	112233
	8517	<input type="checkbox"/>	02.01.2009 15:04:33	08.01.2009 18:44:42	cypher
	8523	<input type="checkbox"/>	03.01.2009 17:24:23	17.01.2009 13:39:21	HaddaRieme
	8531	<input type="checkbox"/>	05.01.2009 14:48:47	16.01.2009 22:57:04	hunter91
	8545	<input type="checkbox"/>	07.01.2009 13:43:53	17.01.2009 13:47:45	Crazy29
Update					
Player hinzufügen		Kennwort ändern		Player löschen	

11.4. SSA-Datenbank

Superadmin Datenbank				
	ID	Erstellt	Letzter Login	Name
▶	17	30.03.2008 13:34:04	11.05.2008 16:30:42	D?MON
	15	16.03.2008 19:44:48	29.03.2008 19:00:13	ANI
	2	24.06.2007 19:09:20	17.01.2009 14:00:01	superadmin
Update				
Player hinzufügen		Kennwort ändern		Player löschen

11.5. Offene Telnet-Verbindungen



	Loginname	Server ID	Superadmin	Severadmin	IP-Adresse	Letzter Befehl
▶	superadmin	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	192.168.1.2	lc

Update

11.6. Anzeigooptionen



Daten auswählen

Beschreibung

- ☒ ID
- ☒ Loginname
- ☒ Nickname
- ☒ IP-Adresse
- ☐ Globale Rechte
- ☐ Channel Rechte
- ☒ Flags
- ☒ Ping
- ☒ Bereits eingeloggte Zeit
- ☒ IdleTime
- ☒ Registrierung erlaubt
- ☐ Gesendete Bytes
- ☐ Gesendete Pakete

Up

Down

Überschrift

Schriftart

Farbe

Text

Schriftart

Farbe

☒ Diese Schrifteinstellungen auch für Channels verwenden

OK

Abbrechen

Überschrift
Text unter der Überschrift

12. Klasse DataBaseTemplate

Hier die Felder, Eigenschaften und Methoden der Klasse DataBaseTemplate, um zu zeigen, welche Methoden von den Unterklassen implementiert werden müssen.

```
public abstract partial class DataBaseTemplate<TItemHandler, TItem> : Form
    where TItemHandler : ItemHandler
    where TItem : Item
{
    protected Configuration config;
    protected TItemHandler database;
    protected Dictionary<int, TItem> dictionary;
    protected Type dbusertype;

    public DataBaseTemplate(TItemHandler database, Configuration config);

    protected abstract void CreateColumns();

    protected void CreateColumn(string headerText, bool readOnly);

    protected void CreateColumn(string headerText, bool readOnly, Type valueType);

    protected void CreateColumn(string headerText, bool readOnly, Type valueType,
                                bool checkBoxColumn);

    protected abstract void FillDictionary();

    public new virtual void Show();

    private object GetPropertieValue(TItem dbuser, int index);

    private void dgvUserDataBase_CellValueNeeded(object sender,
                                                  DataGridViewCellValueEventArgs e);

    private void dgvUserDataBase_CellValidating(object sender,
                                                  DataGridViewCellValidatingEventArgs e);

    protected abstract void CellValidating(object sender,
                                            DataGridViewCellValidatingEventArgs e);

    private void btUpdate_Click(object sender, EventArgs e);

    protected abstract void UpdateDataBase();

    private void UpdateList();

    private void UpdateContent();

    protected abstract int DataBaseCount { get; }

    private void TrimRowCount();
```

```
protected void btAddUser_Click(object sender, EventArgs e);  
protected abstract void AddUserDialog();  
private void btDelete_Click(object sender, EventArgs e);  
protected abstract bool DeleteUser(TItem dbuser);  
private void btPlayerEdit_Click(object sender, EventArgs e);  
protected abstract void ChangePassword(TItem dbuser);  
private void DataBaseTemplate_FormClosing(object sender, FormClosingEventArgs e);  
}
```

13. Klasse ShowPropertyAttribute

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace TeamSpeak_Server
{
    [AttributeUsage(AttributeTargets.Property)]
    public class ShowPropertyAttribute : Attribute
    {
        public string DisplayName { get; private set; }
        public string Description { get; private set; }
        public int Order { get; private set; }
        public bool DefaultValue { get; private set; }

        public ShowPropertyAttribute(string displayName, string description, int order,
                                     bool defaultValue)
        {
            this.DisplayName = displayName;
            this.Description = description;
            this.DefaultValue = defaultValue;
            this.Order = order;
        }
    }
}
```