**Overview:**

This protocol describes a chat system involving a server and multiple users.

**Sections**

1. Joining the server
2. Users
3. Channels
4. Messages
5. User commands
6. Operator commands
7. Server responses

**1 - Joining the servers.**

1.1 - Unique nickname / identifiers.

When a new client connects to the server, the server will respond with prompt username

(7.5). The user will respond by sending a username to the server, the string may not begin with the ‘!’, ‘/’, ‘.’ or ‘#’ characters and may not contain a space. Nicknames are case sensitive. Nicknames may contain lowercase and uppercase a-z as well as numbers 0-9 and underscores ‘\_’.

If the username is unique the server will respond with an acknowledgement (7.1). If the username is not unique, the server will respond with prompt new username (7.6). This continues until a unique username is found.

**2 - Users.**

1.1 - Users

This is the default type of user. Users can send plain messages (4.1) and can also use all User commands (5).

1.2 - Operators

Operators have all of the abilities of users, however can also use all Operator commands (6).

**3 - Channels**

3.1 - Channel names

Each channel will have a unique name. Channel names are stored in strings and will begin with the (#) character. This is to distinguish them from other strings. E.g:

#General

#Code

#Help

**4 - Plain messages**

4.1 - Sending a message

Plain messages may not begin with a (!), (/), (.) or (#) character. If the message sent is not valid the server will respond with Invalid (7.2). If the message is valid the server will respond with Acknowledgement (7.1).

**5 - User commands**

All user commands begin with an exclamation mark (!) character. This is to distinguish them from plain messages and other commands. Commands are case sensitive. If any command is mistyped or invalid (e.g. the user missed a parameter), the server will respond with Invalid command (7.3). Otherwise, the server will respond with Acknowledgement (7.1).

5.1 - Switching rooms

The user can only be in one room at a time. In order to switch rooms, the user will send a (!) followed by the word “join”, a space and then the channel name (Including the (#) character). If the user attempts to join a channel that does not exist, the server responds with invalid channel (7.7).

Example:

!join #general

5.2 - Private message

Sending a private message will display the sent message only to the specific user. Private messages begin with a (!) character, followed by the word “message”, a space and then the user’s name, followed by another space and then the message. Private messages from a user you have muted (5.3) will not be displayed.

Example:

!message Nick\_name Hello World!

5.3 - Ignore/Un-ignore users

When a user (a) ignores another user (b), user a will not be able to see any messages sent by user b. To ignore a user, the command begins with a (!) character, followed by the word “ignore”, a space and then the unique identifier of the user they wish to ignore.

Example:

!ignore Nick\_name

To undo this, the user can use “unignore”.

Example:

!unignore Nick\_name

5.4 - Help

Users can use the Help command to see a list of all commands available to them, along with a description of what they do. If the user is also an operator, it will also list all operator commands. The help command begins with the (!) character followed by the word “help”.

Example:

!help

5.5 - List rooms

This command will list all available rooms. This command begins with the (!) character followed by the word “rooms”.

Example:

!rooms

**6 - Operator commands**

Operators have unique commands with regular users cannot use. All Operator commands begin with a (/) character. Operator commands are case sensitive. If a user attempts to use an operator command, the server will respond with Invalid permission (7.4). If any command is mistyped or invalid (e.g. the user missed a parameter), the server will respond with Invalid command (7.3). Otherwise, the server will respond with Acknowledgement (7.1).

6.1 - Mute/unmute users

The mute command allows an operator to prevent any users messages from being displayed, this applies to all channels. Muted users can still private message other users. Operators can mute users but cannot mute other operators.

The mute user command begins with a (/) followed by “mute”, a space and then the user’s nickname.

Example:

/mute Nick\_name

To unmute a user, the operator can replace “mute” with “unmute”.

Example:

/unmute Nick\_name

6.2 - Kick users

Kicking a user will disconnect the user from the server. Operators can kick users but cannot kick other operators.

The kick command begins with a (/) followed by the word “kick”, a space and then the user’s nickname.

Example:

/kick nickname

**7 - Server responses**

All server responses begin with a dot (.). Server commands are case sensitive and are always typed in full-caps.

7.1 - Acknowledgement.

General acknowledgement. This tells the client that their message/command was valid.

Response:

.OK

7.2 - Invalid.

General command. This tells the client that the message/command they sent was invalid.

Response:

.INV

7.3 - Invalid command

This response tells the client that a command they attempted to send was invalid.

Response:

.INVCMD

7.4 - Invalid permission

This response tells the client that they do not have permission to perform the command they tried to send.

Response:

.INVPRM

7.5 - Prompt username

This response prompts the client to enter a username, following the rules specified in (1.1).

Response:

.USER

7.6 - Prompt new username

This response tells the client the username entered was invalid, and to enter a new username, following the rules specified in (1.1).

Response:

.NEWUSER

7.7 – Invalid channel

This response tells the client that the user attempted to join a channel that does not exist.

Response:

.INVCHANNEL

**References:**

Kalt, C. and 1, rfcmarkup version (2000) *Internet relay chat: Client protocol*. Available at: <https://tools.ietf.org/html/rfc2812.html> (Accessed: 22 November 2016).