



Report Projectwork: Chat TCP in C

Group:

- Canzolino Gianluca 0622701806 g.canzolino3@studenti.unisa.it
- Gambardella Giuseppe 0622701666 g.gambardella23@studenti.unisa.it
- Provenza Alberto 0622701850 a.provenza1@studenti.unisa.it

Index

Task..... 3

Software Environment 3

File List..... 3

Communication Protocol 4

Documentation..... 8

How To Run 8

Demo Interfaces..... 8

Task

Implementation of a simple chat in C.

The architecture must be client server and the communication must be implemented with sockets.

Client must connect to the Server to communicate with other Clients.

Each Client is identified by username.

The Server must manage connections and allows the exchange of messages between Clients.

A Client can also broadcast a message to all active Clients connected to the Server.

Software Environment

Distributor ID: Ubuntu

Description: Ubuntu 20.04.3 LTS

Release: 20.04

Codename: focal

File List

Here is a list of all documented files with brief descriptions:

▼ include	
client.h	Client header file
server.h	Server header file
▼ library	
▼ include	
gui.h	
hashtable.h	
infoHashtable.h	
infoList.h	
list.h	
message.h	Definition of struct Message
queue.h	
TBST.h	
tcp_socket.h	Implementation of tcp socket both for client and server
TinfoBST.h	
▼ src	
gui.c	
hashtable.c	
infoList.c	
list.c	
message.c	
queue.c	
TBST.c	
tcp_socket.c	
TinfoBST.c	
test.c	
▼ src	
client.c	Client Implementation of gagchat
server.c	Server implementation of gagchat

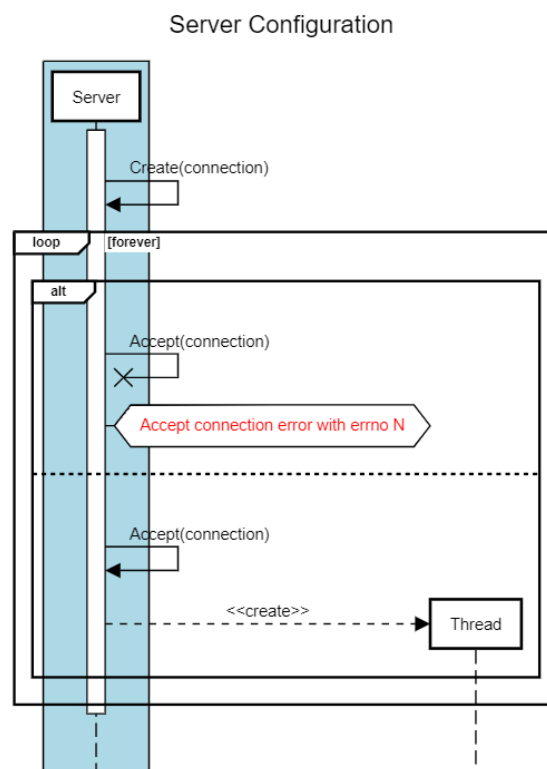
Communication Protocol

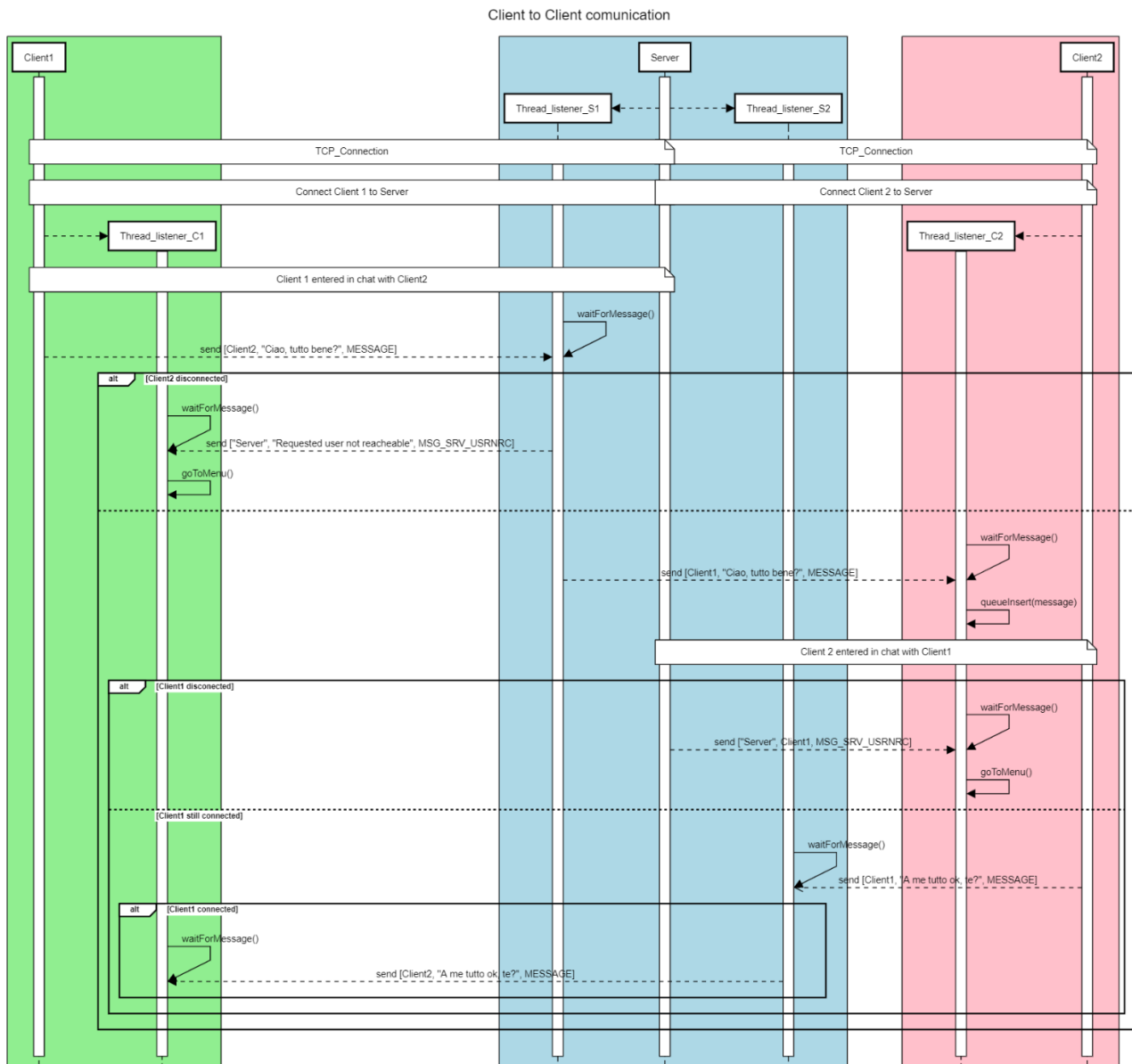
The communication protocol is characterized by the exchange of Messages between Client and Server. The basic message sent by parties is characterized by three params:

- user (string): the username string.
- text (string): the text string.
- code (int): a control code exchanged by parties, to simplify communication between client and server. There are many different codes:
 - **MESSAGE**: Normal Message directed to specified user.
 - **MSG_CLI_CREATE**: Request for creating a new user (specified by user).
 - **MSG_SRV_USRACK**: User is accepted by the server.
 - **MSG_SRV_USRNCK**: User is NOT accepted by the server.
 - **MSG_SRV_USRNRC**: User is NOT reachable from the server.
 - **MSG_CLI_LSTUSR**: Client request for the available list of users.
 - **MSG_SRV_AVLUSR**: Available user for client (specified by text).
 - **MSG_CLI_USR**: Client request user is available.
 - **MSG_SRV_USR_IS_AVL**: Server response "user is available".
 - **MSG_SRV_USR_IS_NAVL**: Server response "user is not available".
 - **MSG_SRV_NEW_USR**: New user logged.

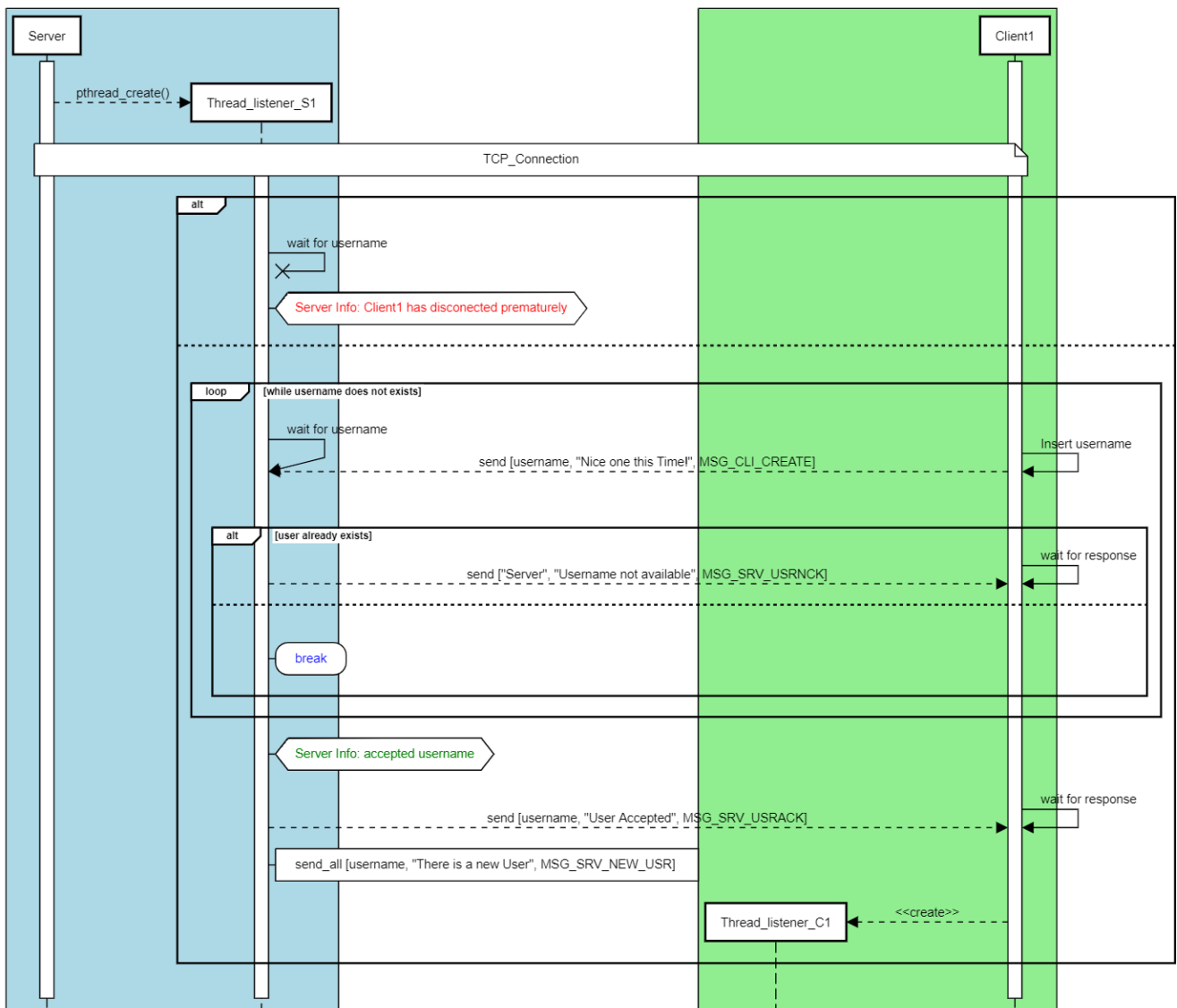
The protocol is simple. Clients connects to the server and needs to specify first his desired Username. If the server notice that the username is already present, it sends a message **MSG_SRV_USRNCK**, otherwise it sends **MSG_SRV_USRACK** message. When Client receives the USRACK message, he can perform three types of queries to server:

- **MSG_CLI_LSTUSR** message: client ask to server to show its list of users. Server answers with n **MSG_SRV_AVLUSR** messages (with n number of users)
- **MESSAGE**: Normal Message directed to specified user.
 - If the specified user is represented by keyword **all**, server sends the message to all connected Users.

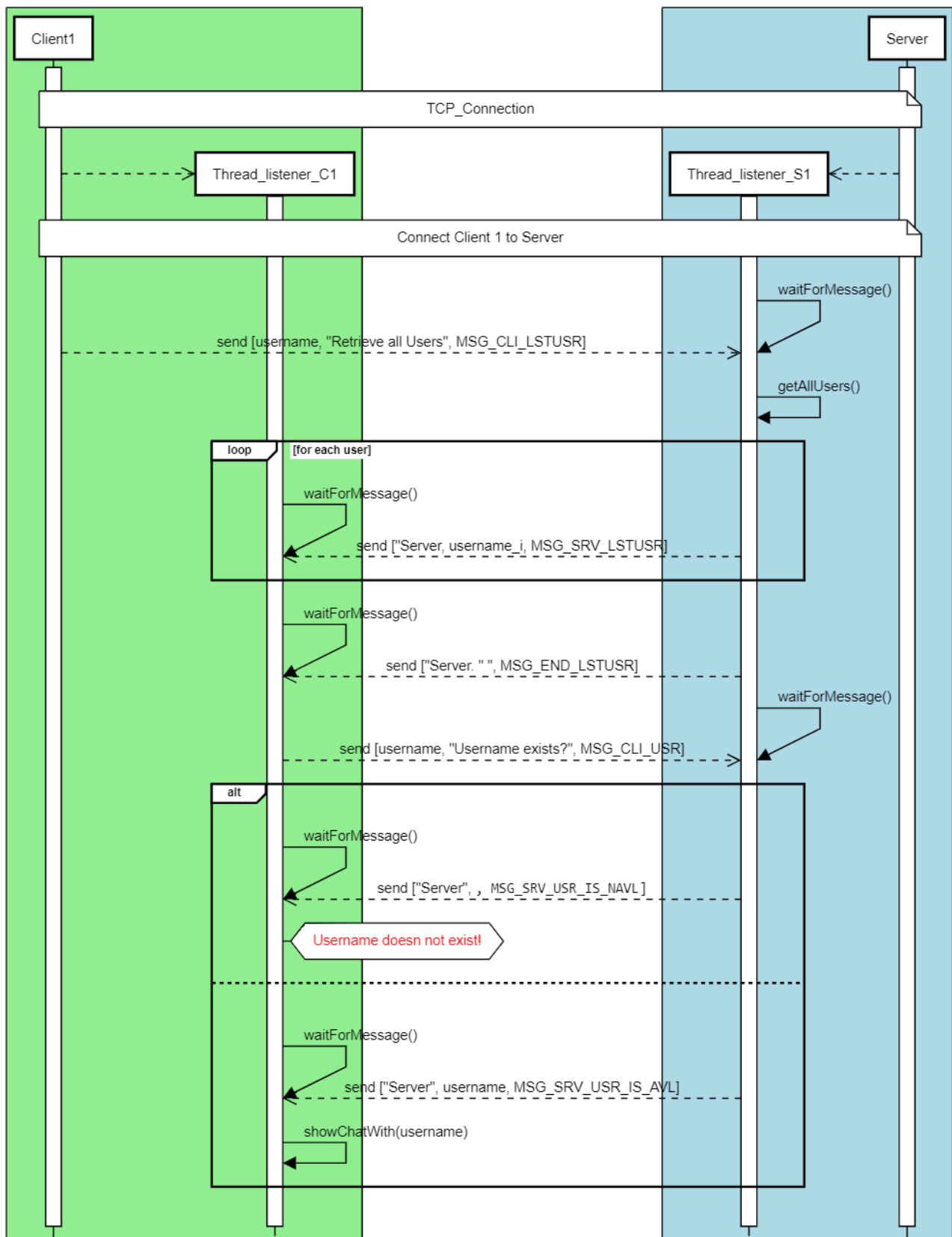




Connect Client1 to Server



Client enter in a chat with another Client



Documentation

For the code documentation, see the attached file refman.pdf.

How To Run

1. Extract the project on a generic dir.
2. Launch the command `make`.
3. `./server` command to execute Server
4. `./client ip` command to execute Client.
 - a. Ip must be a valid ip address. If you are running client on the same machine as server, run `./client 127.0.0.1`

Demo Interfaces

Client username insertion.

```
xeymad@DESKTOP-PLIAVCS:/mnt/c/Users/Fisso_Casa/Documents/GitHub/gagchat$ ./client 127.0.0.1
Please Enter a Username
> peppe
Server: Requested User is not available. Try again.    3
> xeymad

+-----+
| Welcome xeymad                                     |
|                                                     |
| Make your choice, insert:                          |
| -show_all-> show all connected users                |
| -username-> open the chat with the user "username"  |
|                                                     |
+-----+

Insert-> 
```

Client interface for commands.

```
+-----+
| Welcome giuseppe                                   |
|                                                     |
| Make your choice, insert:                          |
| -show_all-> show all connected users                |
| -username-> open the chat with the user "username"  |
|                                                     |
+-----+

Insert-> show_all[]

+-----+
| List of all connected user                         |
|                                                     |
| ~ alberto                                          |
| ~ gianluca                                        |
| ~ mario                                           |
| ~ xeymad                                          |
|                                                     |
| Insert "__menu" to return to the menu              |
| Insert "__exit" to exit from a chat                |
|                                                     |
+-----+

Insert user-> 
```


Server visualization.

```
keymad@DESKTOP-PLIAVCS:/mnt/c/Users/Fisso_Casa/Documents/GitHub/gagchat$ ./server
Server running...waiting for connections.
[ServerInfo]: request with connection_fd 4
mario: Nice one this time!      1
[ServerInfo]: accepted username mario
[ServerInfo]: request with connection_fd 5
alberto: Nice one this time!    1
[ServerInfo]: accepted username alberto
[ServerInfo]: request with connection_fd 6
giuseppe: Nice one this time!  1
[ServerInfo]: accepted username giuseppe
[ServerInfo]: request with connection_fd 7
keymad: Nice one this time!    1
[ServerInfo]: accepted username keymad
User -> giuseppe: Retrieve all Users      code: 5
[ServerInfo]: request with connection_fd 8
gianluca: Nice one this time!  1
[ServerInfo]: accepted username gianluca
User -> giuseppe: Retrieve all Users      code: 5
User -> giuseppe: Retrieve all Users      code: 5
User -> gianluca:      code: 8
User -> gianluca: ciao code: 0
User -> gianluca: come stai code: 0
User -> keymad:      code: 8
User -> keymad: hello code: 0
User -> keymad: how are you? code: 0
User -> alberto:      code: 8
User -> alberto: i'm fine ty! code: 0
User -> all:      code: 8
User -> all: Good morning everyone! code: 0
User -> giuseppe: Retrieve all Users      code: 5
User -> all:      code: 8
User -> mario: code: 8
User -> keymad:      code: 8
[ServerInfo]: Disconnecting user alberto
User -> keymad: Retrieve all Users      code: 5
```

Client send to all. Everyone will receive a message from Mario.

```
+-----+
| Welcome mario                                     |
| Make your choice, insert:                         |
| -show_all-> show all connected users              |
| -username-> open the chat with the user "username" |
+-----+
Insert-> all
```

```
                                <-mario
+-----+
| Good morning everyone!      |
+-----+

█
```

Client to Client communication.

```
+-----+
| Welcome alberto            |
|                             |
| Make your choice, insert:  |
| -show_all-> show all connected users
| -username-> open the chat with the user "username"
|                             |
+-----+

Insert-> xeymad█

Press any command
█

                                <-alberto
+-----+
| hello                      |
+-----+

                                <-alberto
+-----+
| how are you?              |
+-----+

xeymad->
+-----+
| i'm fine ty!              |
+-----+
```

Client “alberto” disconnects while communicating with client “xeymad”. Client xeymad will automatically retrieve the full list of users

```
                                <-alberto
+-----+
| hello                               |
+-----+

                                <-alberto
+-----+
| how are you?                       |
+-----+

xeymad->
+-----+
| i'm fine ty!                      |
+-----+

^C

Closing connection
xeymad@DESKTOP-PLIAVCS:/mnt/c/Users/Fisso_Casa/Documents/GitHub/gagchat$
```

```
+-----+
| List of all connected user          |
|                                     |
| ~ gianluca                        |
| ~ giuseppe                        |
| ~ mario                          |
|                                     |
| Insert "__menu" to return to the menu |
| Insert "__exit" to exit from a chat  |
|                                     |
+-----+

Insert user-> 
```

This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/4.0/> or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA