Mandatory Assignment 1

(SWC3 + TECH2)

Chat system

Dat15A/B/I hand in date:

This is one of the compulsory assignments you need to hand in, and pass, to qualify for 3'rd semester exam.
Hand in date: 16 October 2016, latest at 23:55, on Fronter.
Hand in info: <u>one zip</u> file with all your code.

Overview:

You should code a Chat system, with a chat client that can connect to a chat server.

You may need to use threads in client and/or in server. The client should at the start ask the user his chat-name and then send a join message to the server.

The server should accept clients to join the chat system, using a protocol specified below. When a client joins, the server should maintain and update a list of all active clients. Server will need to save for each client the user name, IP address and Port number.

If a new user tries to join with same name as an already active user, then an error message should be sent back to client. Client can try again with a new name.

An active client can send user text message to the server that will just send a copy to all active clients in the client list.

The Client must send a "heartbeat alive" message once every minute to the Server. The server should (maybe with a specialized thread) check the active list, and delete clients that stop sending heartbeat messages. Maybe the active list should include last heartbeat time.

The Client must send a Quit message when it is closing.

Protocol between Chat server and client:

List of allowed messages (and their meaning):

```
JOIN {user_name}, {server_ip}:{server_port}
```
(From client to server. The user name is given by the user. User
name is max 12 chars long, only chars, digits, '-' and '_' allowed.)

```
J_OK
```

(From server to client. Client is accepted.)

J_ERR
(From server to client. Client not accepted. Duplicate name, please
try again with different user name.)

DATA {User_Name}: {free text…}
(From client to server. And from server to all clients. First part of message
indicates from which user it is, the colon : indicates where the
user message begins. Max 250 user characters.)

ALVE
(From client to server. Client sends this heartbeat alive every 1
minute.)

QUIT
(From client to server. Client is closing down and leaving the group.)

LIST {name1 name2 name3 …}
(From server to client. A list of all active user names is sent to all
clients, each time the list at the server changes.)