# Mandatory Assignment 2, Tech2 class

## September 2019, KEA, Dat18D

The student should program a chat server and a matching chat client.

The solution should be programed in Java.
The chat server and client must use UDP for communication.
The server port number is 20202.
The chat server and client must use the below application protocol.
The user interface is a plain terminal window. No GUI is required.

The chat client functionality:

- When the chat client starts, the user should be asked where the chat server is, i.e. the IP-address of the chat server.
- Next the user should be asked which chat username he wants to use.
    - No login is required in this initial version. Usernames are free.
- The chat client tries to contact the chat server and join the main chat room.
- If the clients chat username is already in use, the server will reject the join.
- If the join is successful, the chat client should run two independent threads.
    - One thread is accepting text from the user and sending it to the chat server.
    - The other thread is receiving UDP packets from the server and displaying the contents to the user.
- Any chat messages from the server incoming to the chat client should be presented to the user in the format:
    - From user: username  Message: message text
- The user can quit the chat room any time by typing QUIT

The chat server functionality:

- When the chat server starts, it will initialize a common chat room.
    - The server will keep an updated list of participants.
    - The list contains one entry for each connected chat client.
    - Each entry contains: client IP address, client port number, client chat username.
- When a new chat client tries to connect (with a join message), the server will check if the username is already used.
    - If the username is free, the server will give a positive answer to the client and add an entry to the internal list of clients.
    - If the username is already occupied, the server should send an error message to the client.
- When an accepted client sends a message to the chat server, the server will send a copy of this message to all connected clients (all clients in the list), indicating which client originated the message.
- If the chat server receives a disconnect message from a client, then the client is removed from the internal list of clients.
- When a new client is accepted at the server, the server will inform all current clients that a new client has joined the chat room.

The student solutions should be interchangeable.

- If a student starts a chat server then other students' clients should be able to connect and cooperate with that server.
- One student's client should be able to connect to another student server and cooperate there.

For different chat servers and chat clients to be able to cooperate, we have to follow a common chat server-client application layer protocol. The users should not be aware of any protocol used. The protocol and messages in the protocol is defined here:

In any message:
The first word in upper case is a constant implying some action.
The text inside [] is free text from the user. Any text is truncated to maxiumum 200 characters.
Characters [ and ] are not part of the message, just placeholders in the protocol description.

Possible messages from the chat client:

JOIN: [username]
MESSAGE: [user text]
QUIT

Possible messages and responses from the chat server

JOIN OK
JOIN ERROR username occupied
NEW CLIENT [username]
FROM [username] MESSAGE [user text]

In case of any doubts or need for help, contact the teacher.