Context

The backend of a chat application, which makes it possible to connect to any room and communicate directly with other users.

The task

Develop a backend application, which supports:

1) One-to-one communication
2) One-to-many communication (group channels)

The following methods should be available:

1) Connect - connect to the server
    a) In: username (no password required)
    b) Out: messages **stream** or error
2) JoinGroupChat - join group chat
    a) In: the group channel name
    b) Out: empty message or error
3) LeaveGroupChat - leave group chat
    a) In: the group channel name
    b) Out: empty message or error
4) CreateGroupChat - create group chat
    a) In: the group channel name
    b) Out: empty message or error (including the case when the name is already taken)
5) SendMessage
    a) In: the channel name or username, the text of a message
    b) Out: empty message or error
6) ListChannels
    a) In: empty message
    b) Out: the list of channels that includes both usernames and groups with the specified type (username/group)

To connect use the gRPC stream, and the server would be sending messages through this server stream. After the connection, any user can send a message to any user just specifying their username. In case there is no such username, an error should be returned.

Users can’t send a message to the group chat without prior entering it. After entering, they would receive all messages that would be sent to this chat.

When a user creates a new chat, they automatically enter it. In case all users leave a group chat, it should be automatically deleted.

In case of disconnection, the user automatically leaves all group channels.

Communication should be done using gRPC, contracts would be developed using protobuf. To authorise and identify a particular user in the following queries after connecting please use **Authorization** header, but pass for simplicity user login to it.

No need to add any database, store everything in memory. Code organisation is up to you, but would be also assessed.

Result

We are expecting a ZIP archive with source codes, an example of a client, which interacts with such a backend, README.md with instructions for running the app and any other information you want to provide, and an example of run in logs.txt (if there are any logs)