Exercise:

The challenge is to create the backend for messaging app. Users will have the ability to create a user, get and create chat groups, invite users to chat groups, and send messages.

To build this you must use a nodejs API framework. You must use a relational db to store the data (sqllite, mysql, postgres, etc). Below are instructions of what you must do but feel free to take creative liberties where appropriate.

Functionality:

Users:

Users APIs that need to be supported

1. Get users
    a. A name
    b. Created at
    c. Email address
2. Create user
    a. This should be used during sign up

Chat Groups:

Users should be able to do the following:

3. Get chat groups. Only users who are part of a chat group can get a chat group. Chat groups will have the following features
    a. A name
    b. Created at
    c. Owner (creator)
    d. A list of users who have access to the chat
4. Create a chat group. Anyone can create a chat group. By default, the creator is a member.
5. Update chat groups. Only the owner can update a chat group. Fields that can be updated
    a. Name
    b. List of users (invite or uninvite)
6. Get all messages for a chat group. Users can only get messages if they are a member of the chat group.
    a. This needs to be paginated since chat groups could have a lot of messages

Chat Messages:

Users should be able to do the following:

1. Get messages. Messages will have the following features
    a. Text

   b. Created at
   c. Updated at
   d. Owner (who created the message)
   e. Chat Group

2. Create messages. Messages must belong to a chat group. Users must be a member of a chat group to create a message in that chat group.

Analytics:

We also need to support the ability for users to get analytic measures to understand chat usage. They will want to see:

1. The 5 users who have sent the most messages
   a. This should return a user’s name and the # of messages
   b. This API should optional take start and end dates to support before, after or in a date range
2. The 5 chats with the most messages
   a. This should return the chat name and the # of messages

Notes:

- All APIs must be paginated