- I started downloading de file “compose.yml” through Ubuntu utilizing WSL 2, with the following command listed on the documentation:

```
10  curl -L https://raw.githubusercontent.com/RocketChat/Docker.Official.Image/master/compose.yml -O
11  ls
```

- I used the following command to download the latest version of RocketChat image through Docker:

```
gregorytrovao@DESKTOP-VKODANU:~$ docker pull registry.rocket.chat/rocketchat/rocket.chat:latest
latest: Pulling from rocketchat/rocket.chat
26c5c85e47da: Pull complete
96da4c1974ec: Pull complete
286584c9c618: Pull complete
ec51043fad6b: Pull complete
10845595c672: Pull complete
23b3c9ae79f3: Pull complete
e8711a648170: Pull complete
ac96427e2f3a: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:2b45c666aafcf09fe6c83861303eaeab5fcd3850a39a30af4b4481c9f07febdc
Status: Downloaded newer image for registry.rocket.chat/rocketchat/rocket.chat:latest
registry.rocket.chat/rocketchat/rocket.chat:latest
```

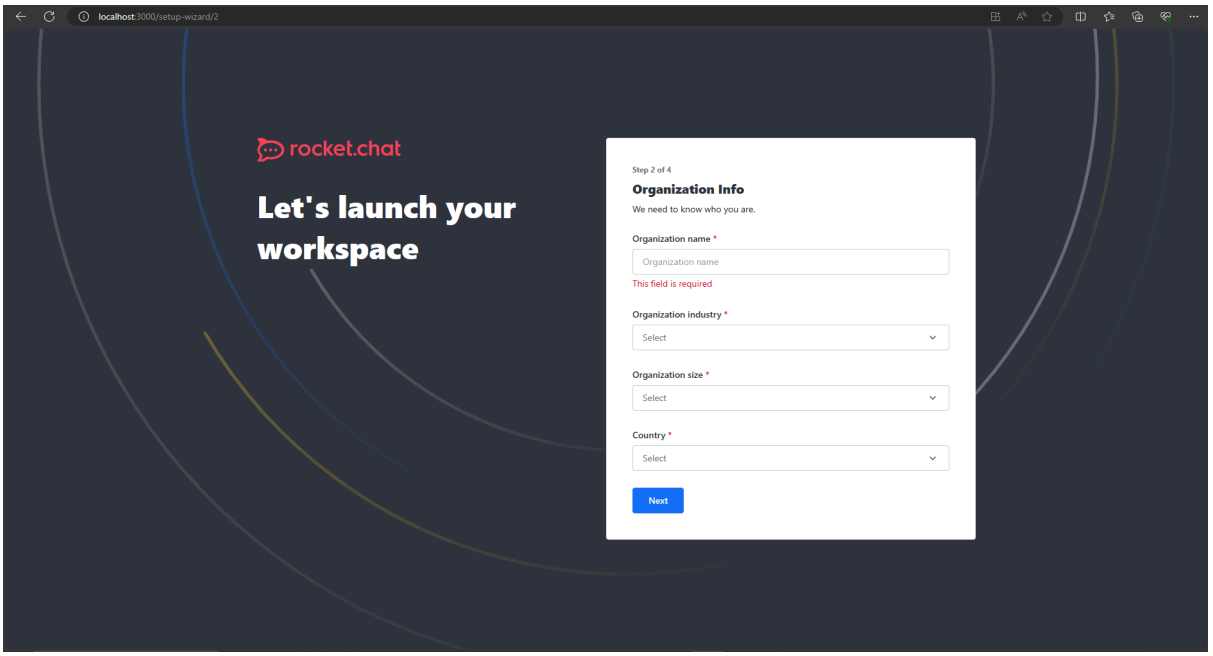
- Following the documentation I used the “nano” command to edit the file with the necessary changes:

```
  GNU nano 6.2                                                    .env
### Rocket.Chat configuration

# Rocket.Chat version
# see:- https://github.com/RocketChat/Rocket.Chat/releases
RELEASE=6.7.0
# MongoDB endpoint (include ?replicaSet= parameter)
#MONGO_URL=
# MongoDB endpoint to the local database
#MONGO_OPLOG_URL=
# IP to bind the process to
#BIND_IP=
# URL used to access your Rocket.Chat instance
ROOT_URL=http://172.21.225.18:3000
# Port Rocket.Chat runs on (in-container)
#PORT=
# Port on the host to bind to
#HOST_PORT=

### MongoDB configuration
# MongoDB version/image tag
#MONGODB_VERSION=
# See:- https://hub.docker.com/r/bitnami/mongodb

### Traefik config (if enabled)
# Traefik version/image tag
#TRAEFIK_RELEASE=
# Domain for https (change ROOT_URL & BIND_IP accordingly)
#DOMAIN=
# Email for certificate notifications
#LETSENCRYPT_EMAIL=
```

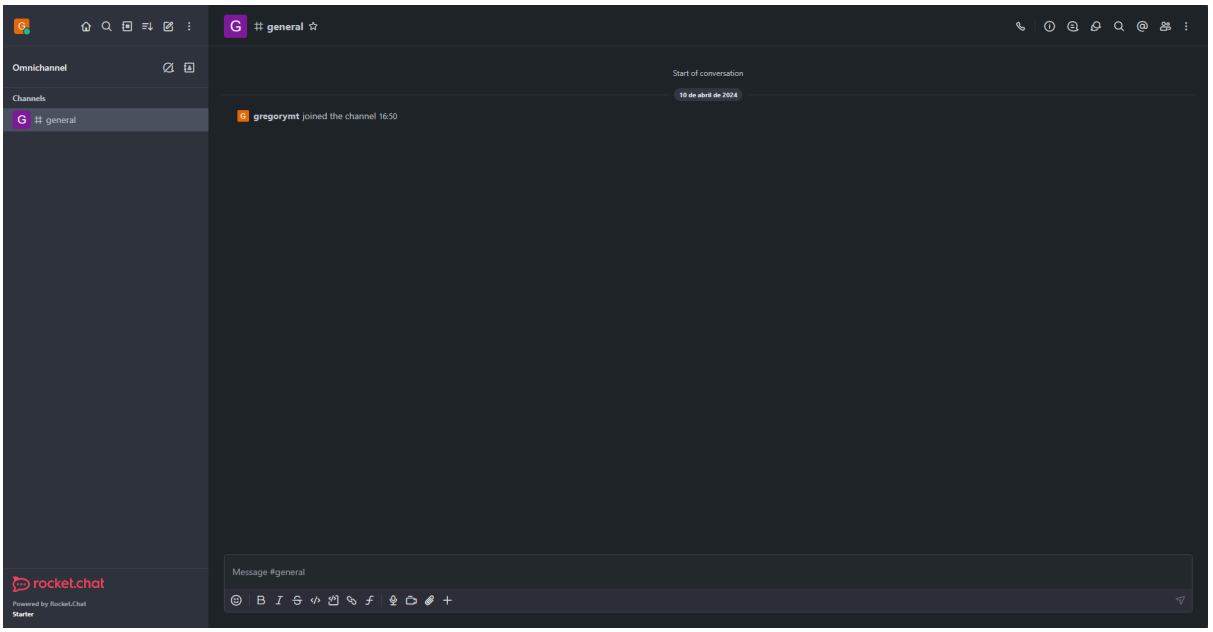
- With Docker Compose I started the container:

```
gregorytrovao@DESKTOP-VKODANU:~$ docker compose up -d
[+] Running 3/3
 ⠿ rocketchat Pulled                                                    3.2s
 ⠿ mongodb Pulled                                                      41.6s
   ⠿ 6754bb5cae91 Pull complete                                        23.5s
[+] Running 4/4
 ⠿ Network gregorytrovao_default          Created                       0.1s
 ⠿ Volume "gregorytrovao_mongodb_data"    Created                       0.0s
 ⠿ Container gregorytrovao-mongodb-1      Started                       3.1s
 ⠿ Container gregorytrovao-rocketchat-1   Started                       0.0s
```

- I opened the link inserted on the compose.yml to test and it was working correctly:

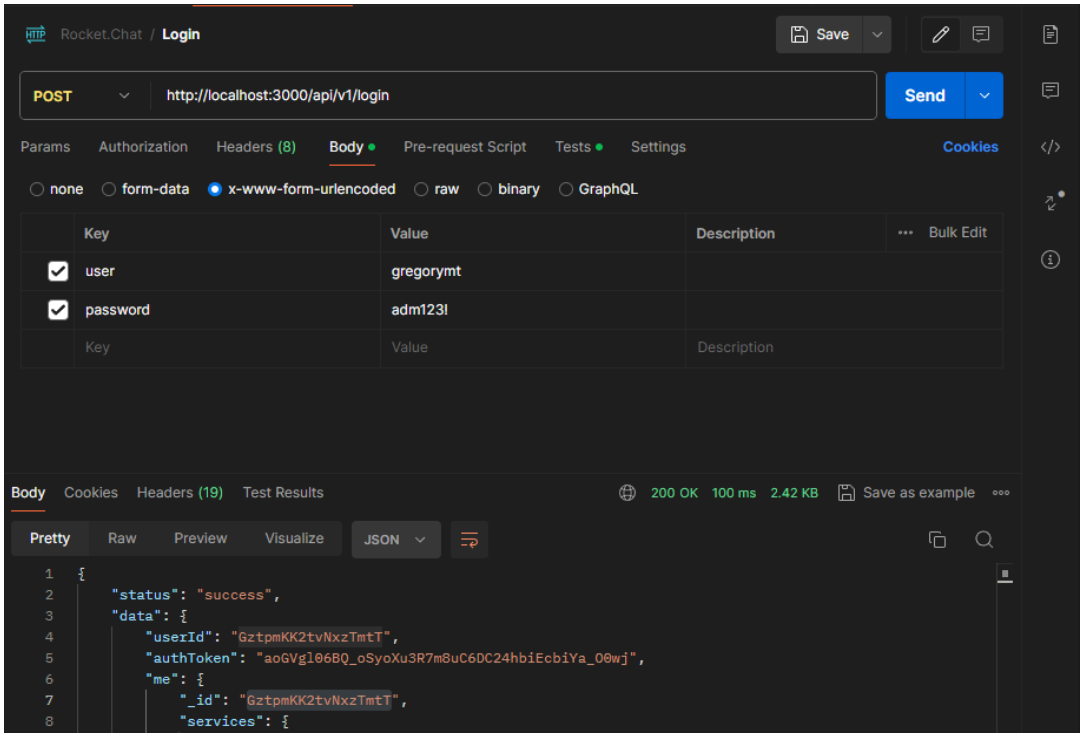


- I registered myself and logged in as an admin:



Then I started doing the 3 tasks asked by the challenge, I used Postman.

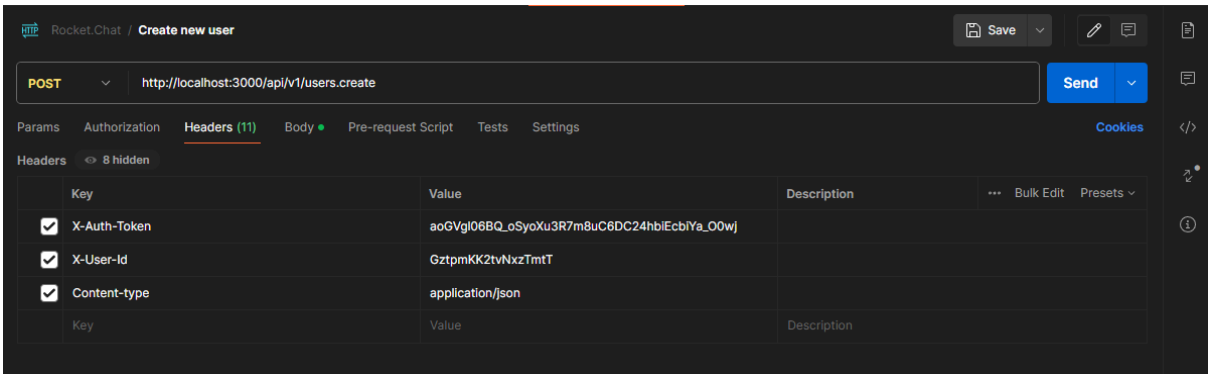
- **Create a new user via an API endpoint:**

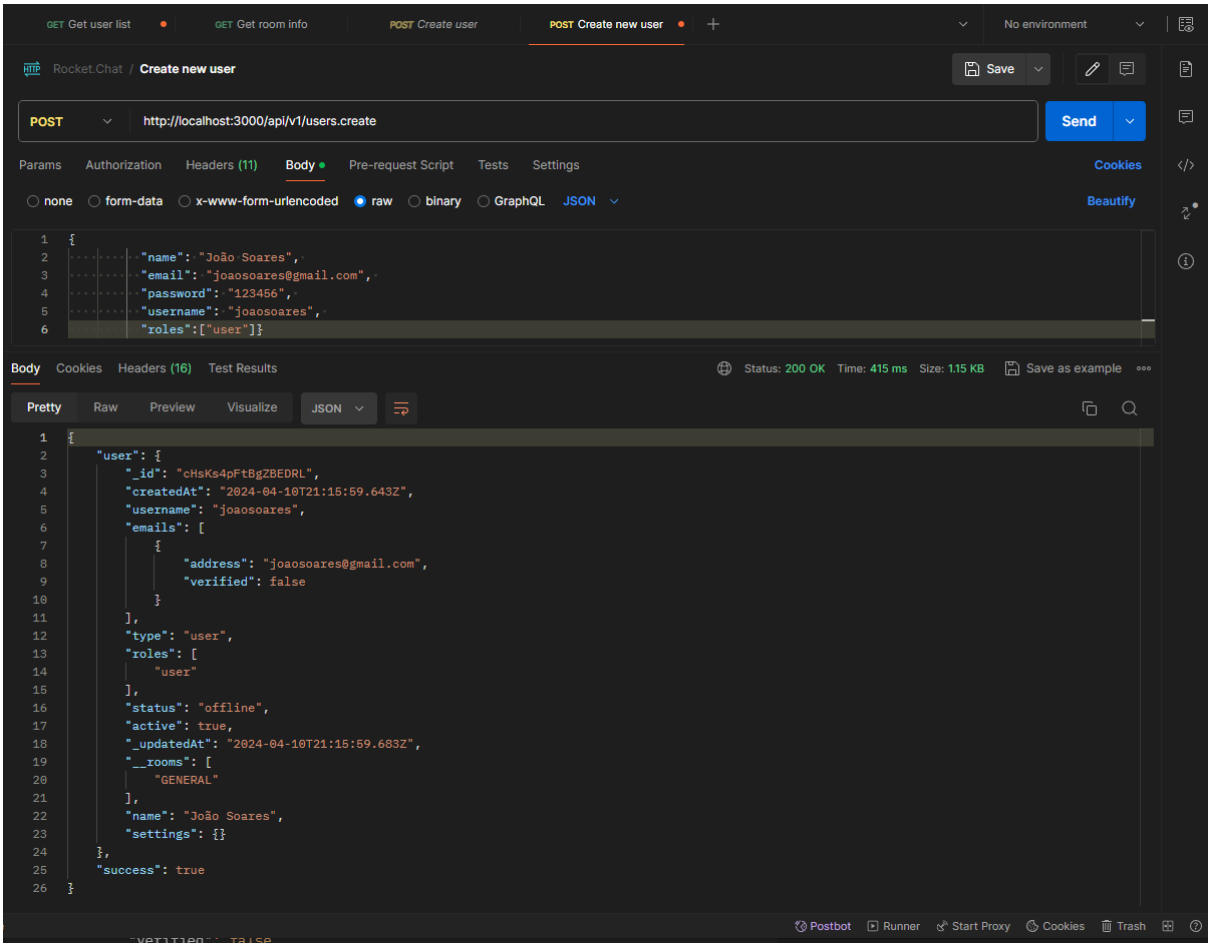


- I noticed that on the documentation the option to create a user required an authentication, so I used the Login with Username and Password API endpoint to get my userID and authToken, using them to create a new user.

- Login Function used:

  curl http://localhost:3000/api/v1/login \
  -d "user=gregorymt&password=adm123!"





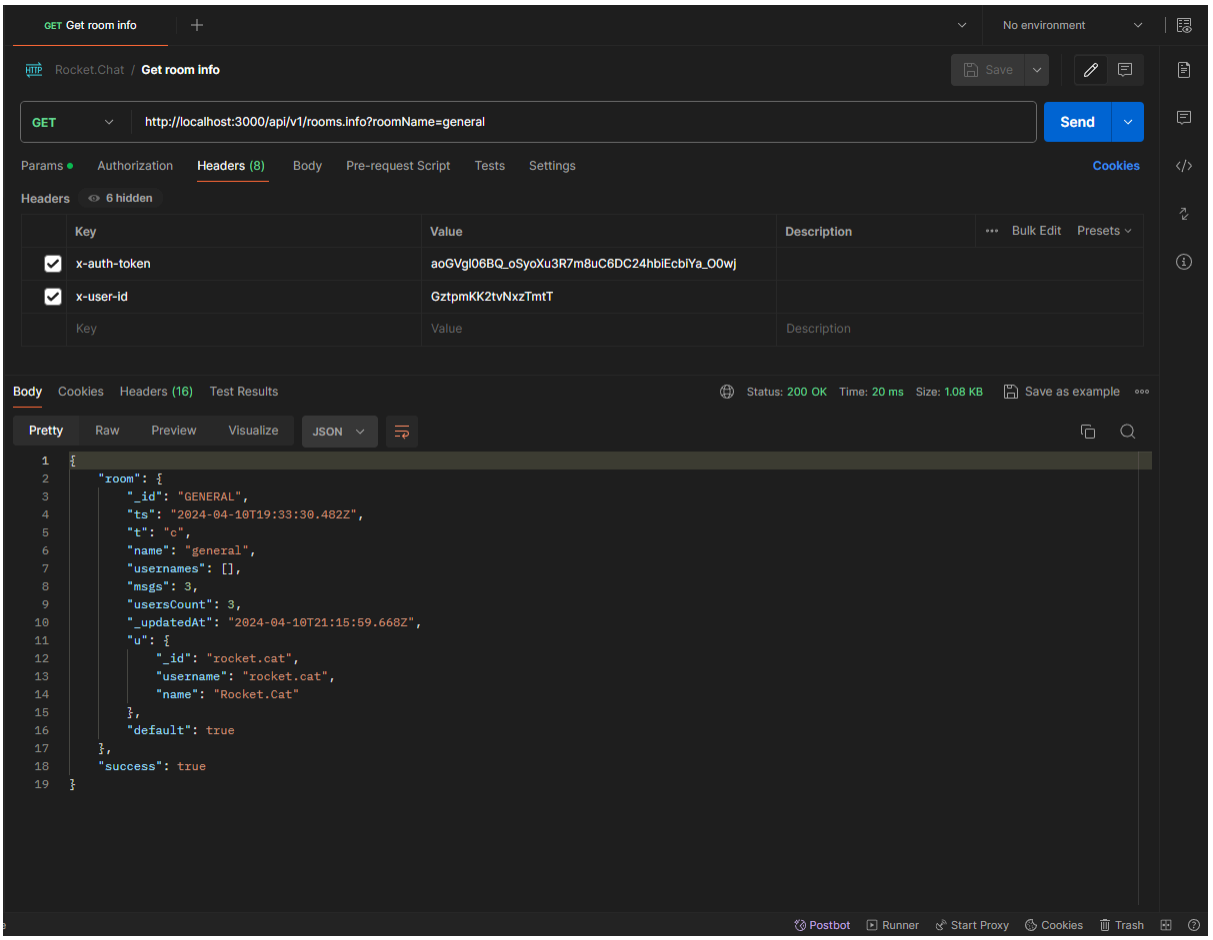
- After getting my token and ID, I used the command “create user” available on the documentation, changing the user infos after running it:

```
curl -H "X-Auth-Token: 9HqLlyZOugoStsXCUfD_0YdwnNnunAJF8V47U3QHXSq" \
    -H "X-User-Id: aobEdbYhXfu5hkeqG" \
    -H "Content-type:application/json" \
    http://localhost:3000/api/v1/users.create \
    -d '{
        "name": "name",
        "email": "email@user.tld",
        "password": "anypassyouwant",
        "username": "uniqueusername",
        "roles":["bot","user"]}'
```

- **Get the room information via an API endpoint:**

- I used the command available on the documentation, but opted to use the roomName instead of the roomID, using the name of the room that was created when I deployed the local instance:

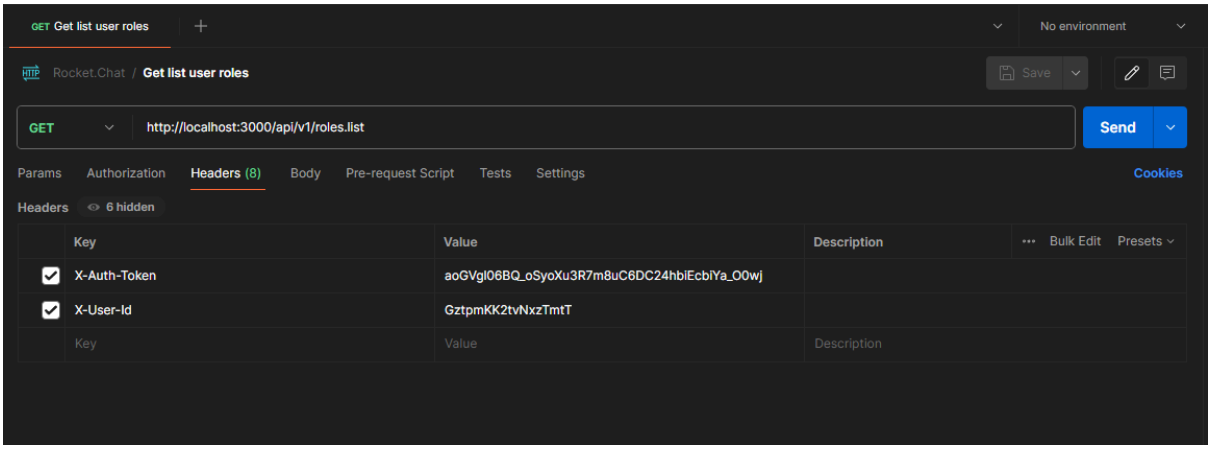
```
curl -H "X-Auth-Token: 9HqLlyZOugoStsXCUfD_0YdwnNnunAJF8V47U3QHXSq" \
    -H "X-User-Id: aobEdbYhXfu5hkeqG" \
    http://localhost:3000/api/v1/rooms.info?roomName=general
```

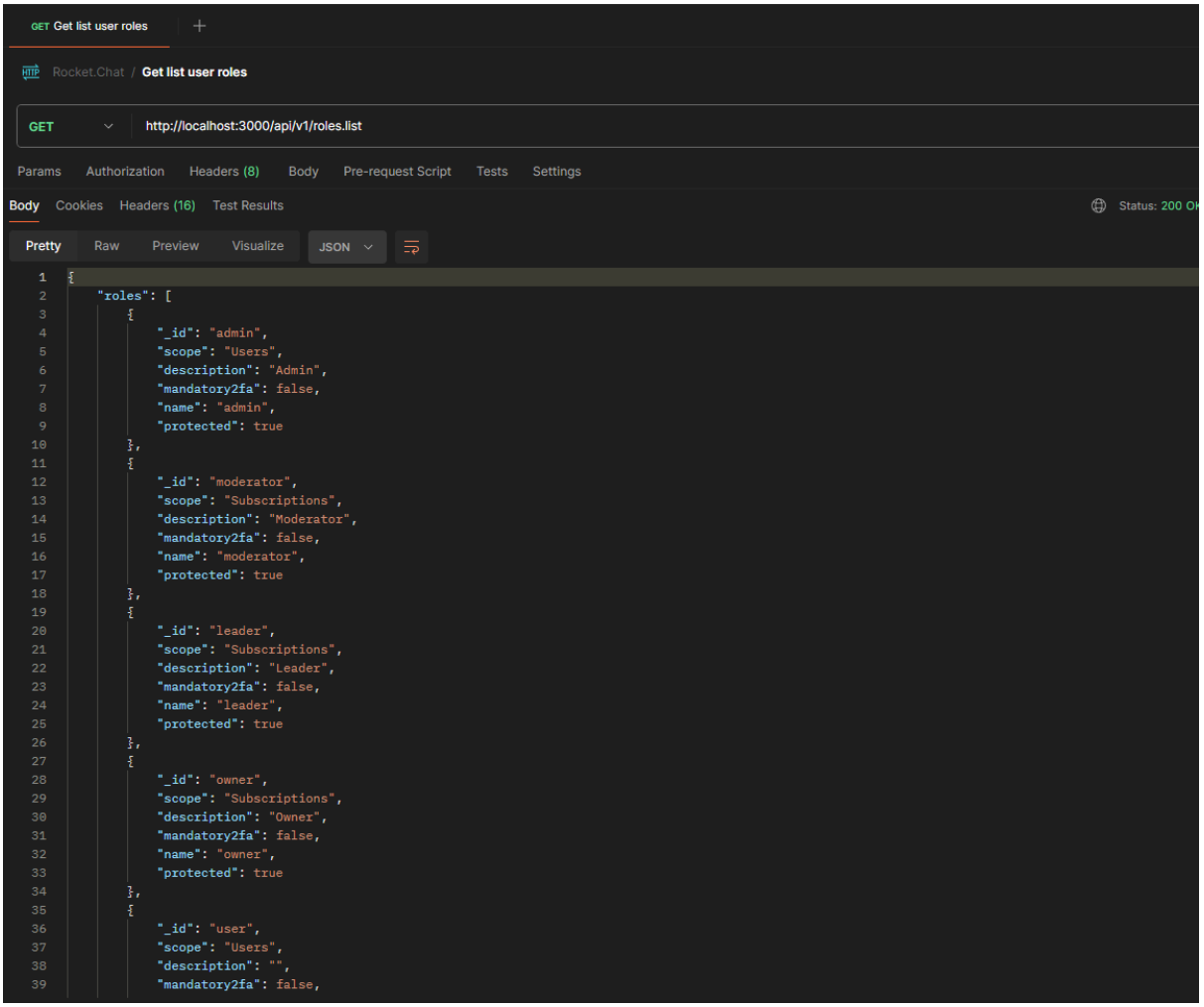


- **Get a list of all user roles in the system via an API endpoint:**

- I used the List Roles Endpoint available on the documentation, just changing to my authToken and userID:

```
curl -H "X-Auth-Token: 9HqLlyZOugoStsXCUfD_0YdwnNnunAJF8V47U3QHXSq" \
    -H "X-User-Id: aobEdbYhXfu5hkeqG" \
    http://localhost:3000/api/v1/roles.list
```



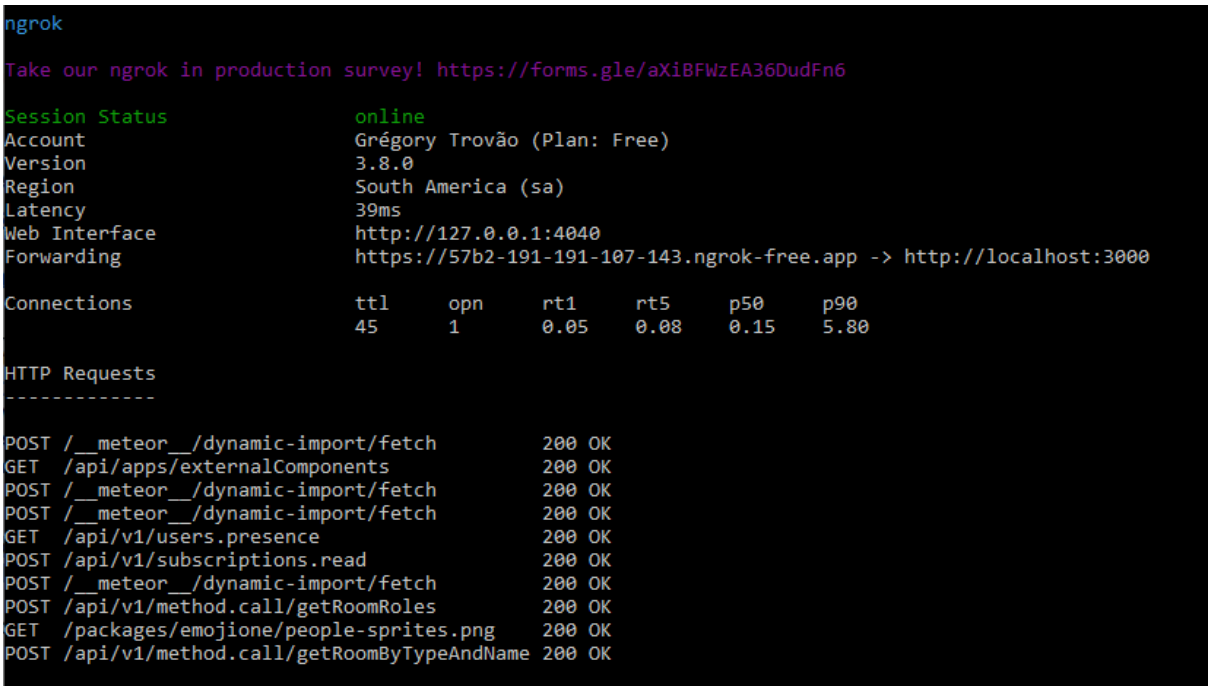


- **Optional: Feel free to go a little crazy and integrate your Rocket.Chat server whatever tool, platform, or service you see fit, ok? Please, amaze us :)**

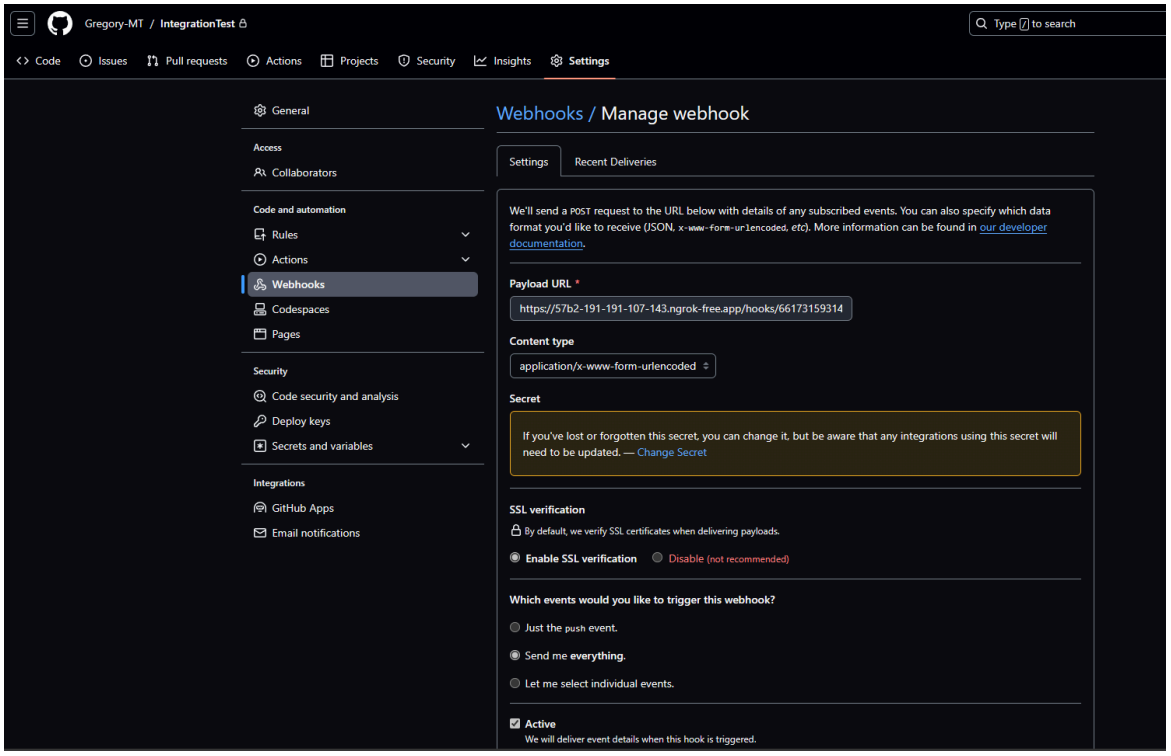
- I created a new channel through Postman to test the integration, changing the "channelname" to "Testes":

```
curl -H "X-Auth-Token: 9HqLlyZOugoStsXCUfD_0YdwnNnunAJF8V47U3QHXSq" \
    -H "X-User-Id: aobEdbYhXfu5hkeqG" \
    -H "Content-type: application/json" \
    https://localhost:3000/api/v1/channels.create \
    -d '{
        "name": "channelname" }'
```

- Using Ngrok I exposed my localhost to external access, then I used the documentation to integrate a webhook from GitHub (In case I close Ngrok, the URL I used will change, and I will have to change the Auth Token and URL on GitHub, because my plan on Ngrok is free and I can't set a fixed URL):



- Following the documentation I created a new incoming integration with the first example script, generating the URL and the Token, then I used them in the GitHub WebHook from the repository I created for this Challenge Test:



- After saving it, I received a message on the “Teste” channel, validating the integration