



Summary

Setting the Environment

- 1 - WSL Instalation on Windows 10
- 2 - Ubuntu Update through WSL 2
- 3 - Deploying Docker and creating MongoDB service
- 4 - Getting the Rocket.chat repository with git
- 5 - Setting Localhost on .env file

Deploying Rocket.chat

- 6 - Pulling the Rocket.chat image
- 7 - Building and running the Containers
- 8 - Rocket.chat platform first steps

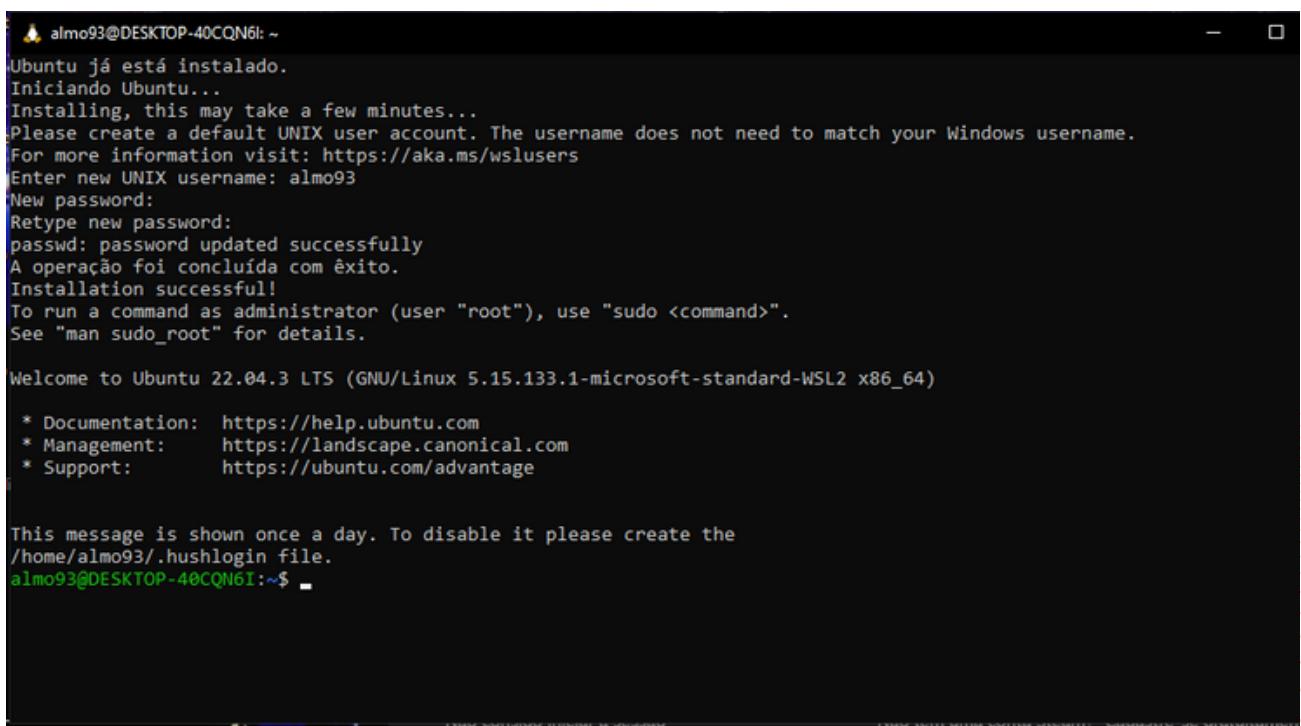
Rocket.Chat API

- 9 - Generating Token
- 10 - Postman Login
- 11 - Creating User / API
- 12 - Getting Room Information
- 13 - Getting all Permission Roles

Setting the Environment

After evaluation, I decided to deploy Rocket.Chat via through terminal to make the process faster and more efficient. Since I was using the Windows 10, I used WSL 2, a feature of Windows that allows running a Linux environment on Windows to deploy Docker for creating MongoDB and Rocket.Chat containers.

1 - In the PowerShell terminal, the command `wsl --install -d ubuntu` was used to install Ubuntu via WSL. After this process, I updated from WSL 1 to WSL 2 with the command `wsl --update` because Docker installation on Windows requires WSL 2.



```
almo93@DESKTOP-40CQN6I: ~
Ubuntu já está instalado.
Iniciando Ubuntu...
Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: https://aka.ms/wslusers
Enter new UNIX username: almo93
New password:
Retype new password:
passwd: password updated successfully
A operação foi concluída com êxito.
Installation successful!
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.133.1-microsoft-standard-WSL2 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

This message is shown once a day. To disable it please create the
/home/almo93/.hushlogin file.
almo93@DESKTOP-40CQN6I:~$
```

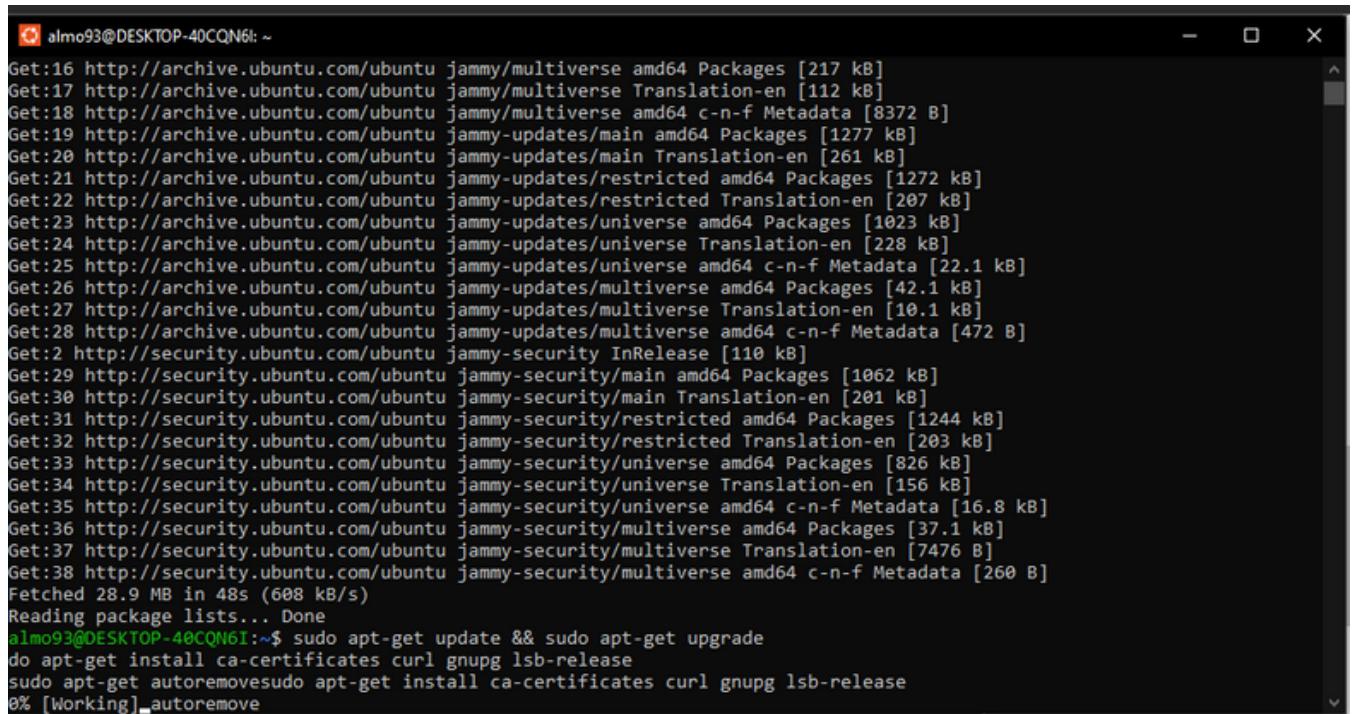


```
Selecionar Administrador: Prompt de Comando - wsl --update
Microsoft Windows [versão 10.0.19045.3803]
(c) Microsoft Corporation. Todos os direitos reservados.

C:\Windows\system32>wls --update
'wls' não é reconhecido como um comando interno
ou externo, um programa operável ou um arquivo em lotes.

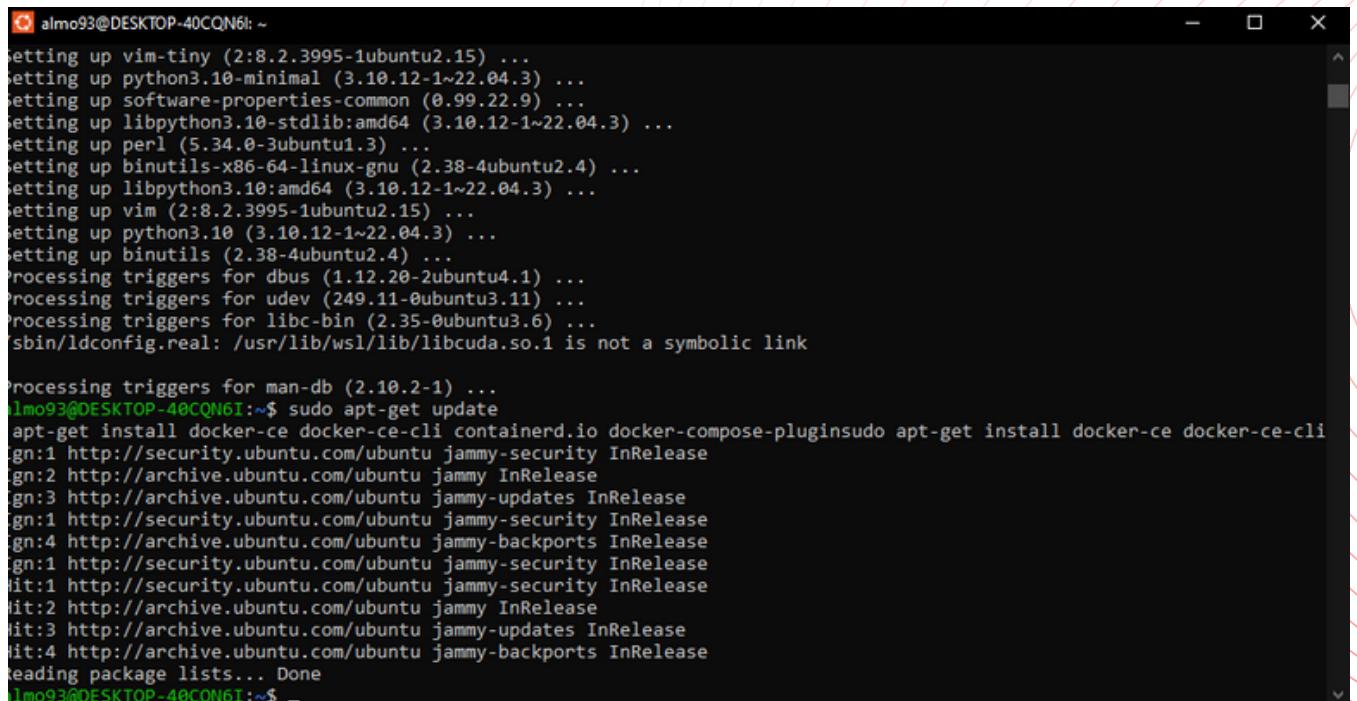
C:\Windows\system32>wsl --update
Verificando se há atualizações.
Atualizando Subsistema do Windows para Linux para a versão: 2.0.14.
```

2 - After updating WSL, the Ubuntu version was also upgraded to the latest version.



```
almo93@DESKTOP-40CQN6I:~$ Get:16 http://archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:17 http://archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:18 http://archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
Get:19 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1277 kB]
Get:20 http://archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [261 kB]
Get:21 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1272 kB]
Get:22 http://archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [207 kB]
Get:23 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1023 kB]
Get:24 http://archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [228 kB]
Get:25 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 c-n-f Metadata [22.1 kB]
Get:26 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [42.1 kB]
Get:27 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse Translation-en [10.1 kB]
Get:28 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 c-n-f Metadata [472 B]
Get:2 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:29 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1062 kB]
Get:30 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [201 kB]
Get:31 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1244 kB]
Get:32 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [203 kB]
Get:33 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [826 kB]
Get:34 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [156 kB]
Get:35 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 c-n-f Metadata [16.8 kB]
Get:36 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [37.1 kB]
Get:37 http://security.ubuntu.com/ubuntu jammy-security/multiverse Translation-en [7476 B]
Get:38 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 c-n-f Metadata [260 B]
Fetched 28.9 MB in 48s (608 kB/s)
Reading package lists... Done
almo93@DESKTOP-40CQN6I:~$ sudo apt-get update && sudo apt-get upgrade
do apt-get install ca-certificates curl gnupg lsb-release
sudo apt-get autoremove
udo apt-get autoremove
udo apt-get install ca-certificates curl gnupg lsb-release
0% [Working] autoremove
```

3 - Starting the Docker installation with the command: ***sudo apt-get install docker-ce*** and ***sudo apt-get update*** for updating to the latest version. After completing this process, I pulled a MongoDB image with the command ***docker pull mongodb/mongodb-community-server***, and to start the newly created container, I used the command ***docker run --name mongo -d mongodb/mongodb-community***



```
almo93@DESKTOP-40CQN6I:~$ setting up vim-tiny (2:8.2.3995-1ubuntu2.15) ...
setting up python3.10-minimal (3.10.12-1~22.04.3) ...
setting up software-properties-common (0.99.22.9) ...
setting up libpython3.10-stdlib:amd64 (3.10.12-1~22.04.3) ...
setting up perl (5.34.0-3ubuntu1.3) ...
setting up binutils-x86-64-linux-gnu (2.38-4ubuntu2.4) ...
setting up libpython3.10:amd64 (3.10.12-1~22.04.3) ...
setting up vim (2:8.2.3995-1ubuntu2.15) ...
setting up python3.10 (3.10.12-1~22.04.3) ...
setting up binutils (2.38-4ubuntu2.4) ...
Processing triggers for dbus (1.12.20-2ubuntu4.1) ...
Processing triggers for udev (249.11-0ubuntu3.11) ...
Processing triggers for libc-bin (2.35-0ubuntu3.6) ...
/sbin/ldconfig.real: /usr/lib/wsl/lib/libcuda.so.1 is not a symbolic link

Processing triggers for man-db (2.10.2-1) ...
almo93@DESKTOP-40CQN6I:~$ sudo apt-get update
apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin
ign:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
ign:2 http://archive.ubuntu.com/ubuntu jammy InRelease
ign:3 http://archive.ubuntu.com/ubuntu jammy-updates InRelease
ign:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
ign:4 http://archive.ubuntu.com/ubuntu jammy-backports InRelease
ign:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
ign:2 http://archive.ubuntu.com/ubuntu jammy InRelease
ign:3 http://archive.ubuntu.com/ubuntu jammy-updates InRelease
ign:4 http://archive.ubuntu.com/ubuntu jammy-backports InRelease
Reading package lists... Done
```

```
almo93@DESKTOP-40CQN6I: ~
Setting up perl (5.34.0-3ubuntu1.3) ...
Setting up binutils-x86-64-linux-gnu (2.38-4ubuntu2.4) ...
Setting up libpython3.10:amd64 (3.10.12-1~22.04.3) ...
Setting up vim (2:8.2.3995-1ubuntu2.15) ...
Setting up python3.10 (3.10.12-1~22.04.3) ...
Setting up binutils (2.38-4ubuntu2.4) ...
Processing triggers for dbus (1.12.20-2ubuntu4.1) ...
Processing triggers for udev (249.11-0ubuntu3.11) ...
Processing triggers for libc-bin (2.35-0ubuntu3.6) ...
/sbin/ldconfig.real: /usr/lib/wsl/lib/libcuda.so.1 is not a symbolic link

Processing triggers for man-db (2.10.2-1) ...
almo93@DESKTOP-40CQN6I:~$ sudo apt-get update
  apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin
  sudo apt-get install docker-ce docker-ce-cli
Ign:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Ign:2 http://archive.ubuntu.com/ubuntu jammy InRelease
Ign:3 http://archive.ubuntu.com/ubuntu jammy-updates InRelease
Ign:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Ign:4 http://archive.ubuntu.com/ubuntu jammy-backports InRelease
Ign:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:2 http://archive.ubuntu.com/ubuntu jammy InRelease
Hit:3 http://archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://archive.ubuntu.com/ubuntu jammy-backports InRelease
Reading package lists... Done
almo93@DESKTOP-40CQN6I:~$ docker --version
Docker version 24.0.7, build afdd53b
almo93@DESKTOP-40CQN6I:~$ docker compose version
Docker Compose version v2.23.3-desktop.2
almo93@DESKTOP-40CQN6I:~$
```

```
almo93@DESKTOP-40CQN6I: ~
See 'docker run --help'.
almo93@DESKTOP-40CQN6I:~$ docker run --name mongo -d mongodb/mongodb-community-server:latest
b3c35b69c32d5fadcb4cd6bd2f75d59099eb158d9b1ac36edf72a0e57bd11efc
almo93@DESKTOP-40CQN6I:~$ docker container ls
CONTAINER ID   IMAGE                               COMMAND                  CREATED             STATUS              PORTS
 NAMES
b3c35b69c32d   mongodb/mongodb-community-server:latest   "python3 /usr/local/..."   30 seconds ago    Up 21 seconds   27017
/tcp mongo
almo93@DESKTOP-40CQN6I:~$
```

4 - Next, copy the link to the Rocket.Chat repository, and with Git initialized, use the command git clone <https://github.com/allanmelo93/Rocket.Chat.git> to clone the repository to your machine.

```

almo93@DESKTOP-40CQN6I:~$ git init
Reinitialized existing Git repository in /home/almo93/.git/
almo93@DESKTOP-40CQN6I:~$ git clone https://github.com/allanmelo93/Rocket.Chat.git
Cloning into 'Rocket.Chat'...
remote: Enumerating objects: 348837, done.
remote: Counting objects: 100% (498/498), done.
remote: Compressing objects: 100% (360/360), done.
remote: Total 348837 (delta 181), reused 290 (delta 104), pack-reused 348339
Receiving objects: 100% (348837/348837), 439.96 MiB | 13.96 MiB/s, done.
Resolving deltas: 100% (217718/217718), done.
Updating files: 100% (7531/7531), done.
almo93@DESKTOP-40CQN6I:~$ curl -L https://raw.githubusercontent.com/RocketChat/Docker.Official.Image/master/compose.yml
-0
  % Total    % Received % Xferd  Average Speed   Time     Time      Time  Current
          Dload  Upload   Total   Spent    Left  Speed
100  1904  100  1904     0      0  432      0  0:00:04  0:00:04  ---:--  435
almo93@DESKTOP-40CQN6I:~$ docker compose up -d
[+] Running 2/2
  ⬤ Container almo93-mongodb-1     Running                               0.0s
  ⬤ Container almo93-rocketchat-1  Started                               0.0s
almo93@DESKTOP-40CQN6I:~$
```

5 - Continuing with the documentation provided at <https://docs.rocket.chat/deploy/deploy-rocket.chat/deploy-with-docker-and-docker-compose>, I used the command ***sudo nano .env***, setting `http://localhost:3000` as `ROOT_URL`, and saved the file.

```

almo93@DESKTOP-40CQN6I:~$ .env *
GNU nano 6.2
#RELEASE=6.5.2
# 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=
# 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=
```

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo M-A Set Mark
 ^X Exit ^R Read File ^V Replace ^U Paste ^J Justify ^/ Go To Line M-E Redo M-6 Copy

Rocket.Chat Deployment

6 - Using the command **curl -L**

https://raw.githubusercontent.com/RocketChat/Docker.Official.Image/master/compose.yml -O to download the *compose.yml* file, and then performed a pull with the command **docker pull registry.rocket.chat/rocketchat/rocket.chat**.

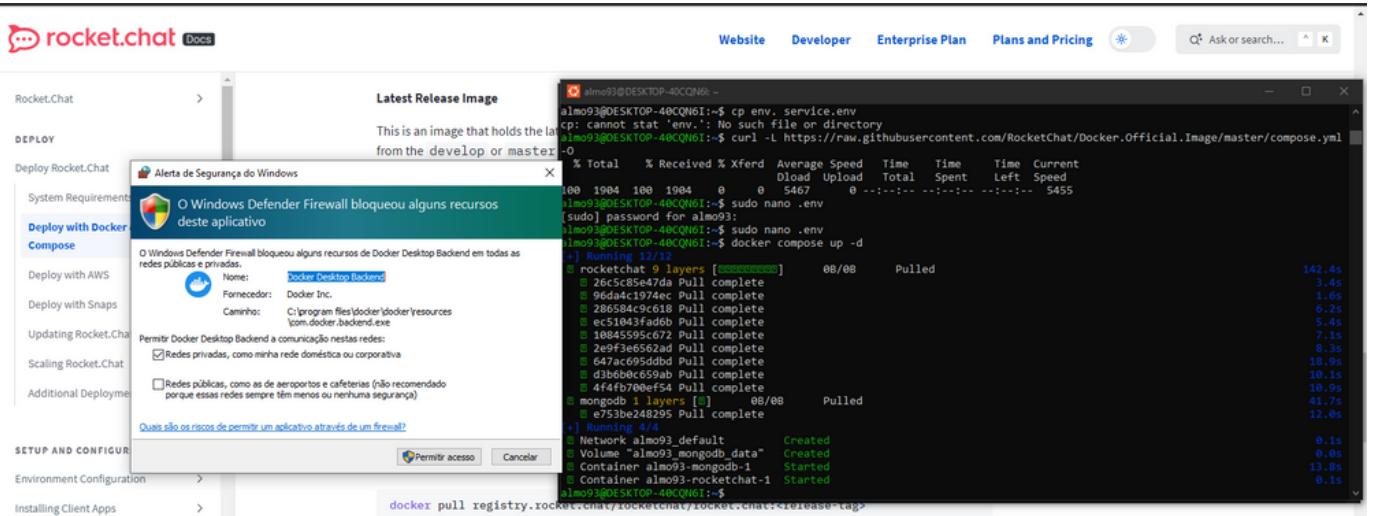
```
almo93@DESKTOP-40CQN6I:~$ git init
Reinitialized existing Git repository in /home/aldo93/.git/
almo93@DESKTOP-40CQN6I:~$ git clone https://github.com/allanmelo93/Rocket.Chat.git
Cloning into 'Rocket.Chat'...
remote: Enumerating objects: 348837, done.
remote: Counting objects: 100% (498/498), done.
remote: Compressing objects: 100% (360/360), done.
remote: Total 348837 (delta 181), reused 290 (delta 104), pack-reused 348339
Receiving objects: 100% (348837/348837), 439.96 MiB | 13.96 MiB/s, done.
Resolving deltas: 100% (217718/217718), done.
Updating files: 100% (7531/7531), done.
almo93@DESKTOP-40CQN6I:~$ curl -L https://raw.githubusercontent.com/RocketChat/Docker.Official.Image/master/compose.yml -O
% Total    % Received % Xferd  Average Speed   Time     Time      Time  Current
          Dload  Upload   Total   Spent    Left  Speed
100  1904  100  1904    0     0   432      0  0:00:04  0:00:04  --:--:--   435
almo93@DESKTOP-40CQN6I:~$ docker compose up -d
[+] Running 2/2
  ⬤ Container almo93-mongodb-1     Running                               0.0s
  ⬤ Container almo93-rocketchat-1  Started                               0.0s
almo93@DESKTOP-40CQN6I:~$
```

```
almo93@DESKTOP-40CQN6I:~$ 
stop      Stop services
top       Display the running processes
unpause   Unpause services
up        Create and start containers
version   Show the Docker Compose version information
wait      Block until the first service container stops
watch     Watch build context for service and rebuild/refresh containers when files are updated

Run 'docker compose COMMAND --help' for more information on a command.
almo93@DESKTOP-40CQN6I:~$ docker pull registry.rocket.chat/rocketchat/rocket.chat:latest
latest: Pulling from rocketchat/rocket.chat
Digest: sha256:69ae057740edc9c039aa66a5fe8b65151f3ad0b4f2f67c1f551b324574942353
Status: Image is up to date for registry.rocket.chat/rocketchat/rocket.chat:latest
registry.rocket.chat/rocketchat/rocket.chat:latest

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview registry.rocket.chat/rocketchat/rocket.chat:latest
almo93@DESKTOP-40CQN6I:~$
```

7 - Using the command **docker-compose up -d** to start the containers of Rocket.Chat and MongoDB simultaneously, and docker-compose logs -f rocketchat to check the newly created server.



```

almo93@DESKTOP-40CQH6I: ~
2024-01-12T18:41:17.863Z] INFO 30395fe4-ae6e-4d88-aaba-f5f324c9a977/TRANSIT: Connecting to the transporter...
2024-01-12T18:41:18.003Z] INFO 30395fe4-ae6e-4d88-aaba-f5f324c9a977/TRANSPORTER: TCP server is listening on port 39671
2024-01-12T18:41:18.004Z] INFO 30395fe4-ae6e-4d88-aaba-f5f324c9a977/TRANSPORTER: UDP Discovery is disabled.
2024-01-12T18:41:18.005Z] INFO 30395fe4-ae6e-4d88-aaba-f5f324c9a977/TRANSPORTER: TCP Transporter started.
fs: temp directory created at "/tmp/ufs"
node:1) [MONGODB DRIVER] Warning: cursor.count is deprecated and will be removed in the next major version, please use collection.estimatedDocumentCount` or `collection.countDocuments` instead
-----+
          SERVER RUNNING
-----+
Rocket.Chat Version: 6.5.2
  NodeJS Version: 14.21.3 - x64
  MongoDB Version: 5.0.23
  MongoDB Engine: wiredTiger
    Platform: linux
  Process Port: 3000
    Site URL: http://localhost:3000
ReplicaSet OLog: Enabled
  Commit Hash: 3ebc8e0868
  Commit Branch: HEAD
-----+
2024-01-12T18:41:18.519Z] INFO 30395fe4-ae6e-4d88-aaba-f5f324c9a977/REGISTRY: '$node' service is registered.
2024-01-12T18:41:18.520Z] INFO 30395fe4-ae6e-4d88-aaba-f5f324c9a977/REGISTRY: 'matrix' service is registered.
2024-01-12T18:41:18.521Z] INFO 30395fe4-ae6e-4d88-aaba-f5f324c9a977/$NODE: Service '$node' started.
2024-01-12T18:41:18.521Z] INFO 30395fe4-ae6e-4d88-aaba-f5f324c9a977/MATRIX: Service 'matrix' started.
2024-01-12T18:41:18.522Z] INFO 30395fe4-ae6e-4d88-aaba-f5f324c9a977/BROKER: ServiceBroker with 2 service(s) started successfully in 670ms.

```

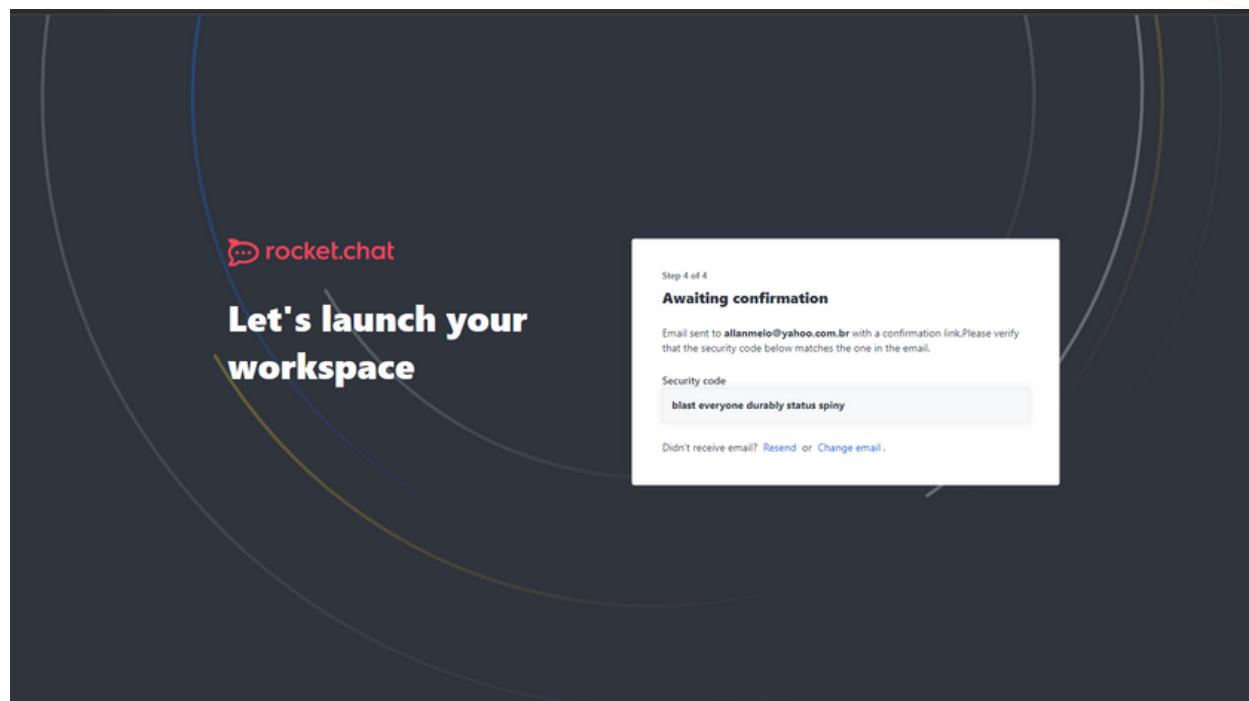
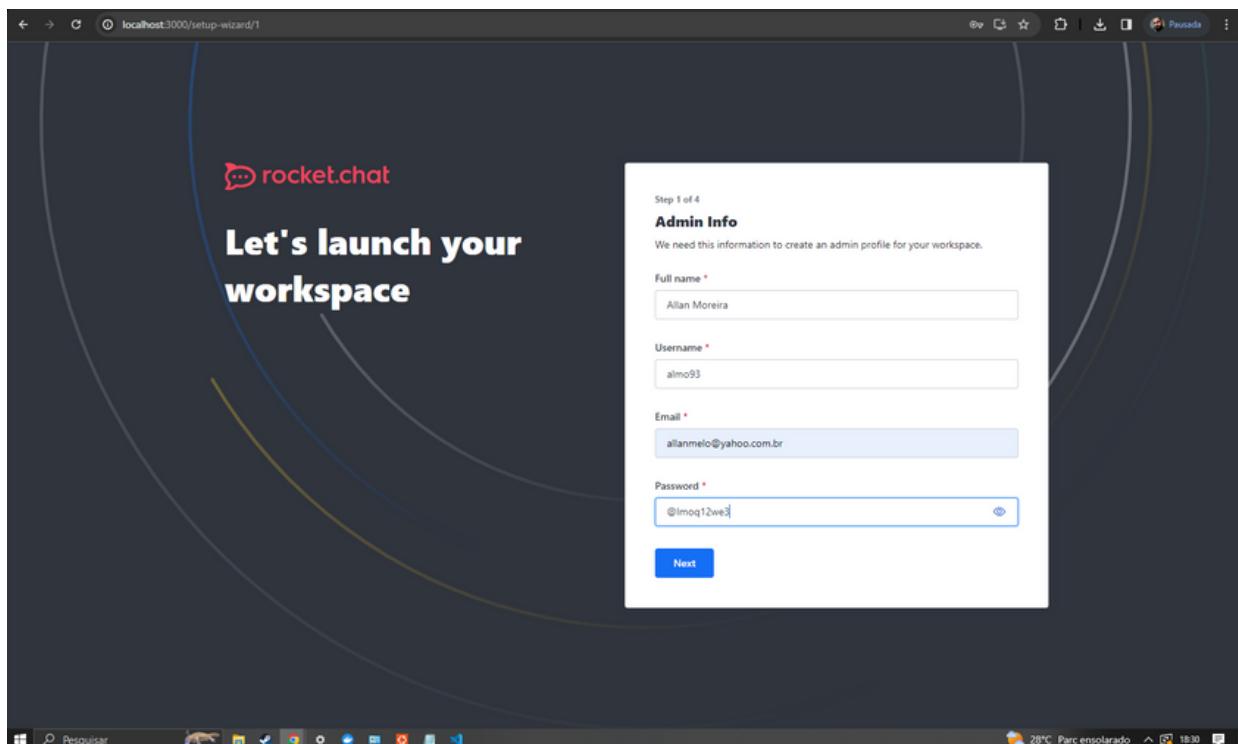
8 - Accessing the browser at <http://localhost:3000/>, the initial configuration for access data begins:

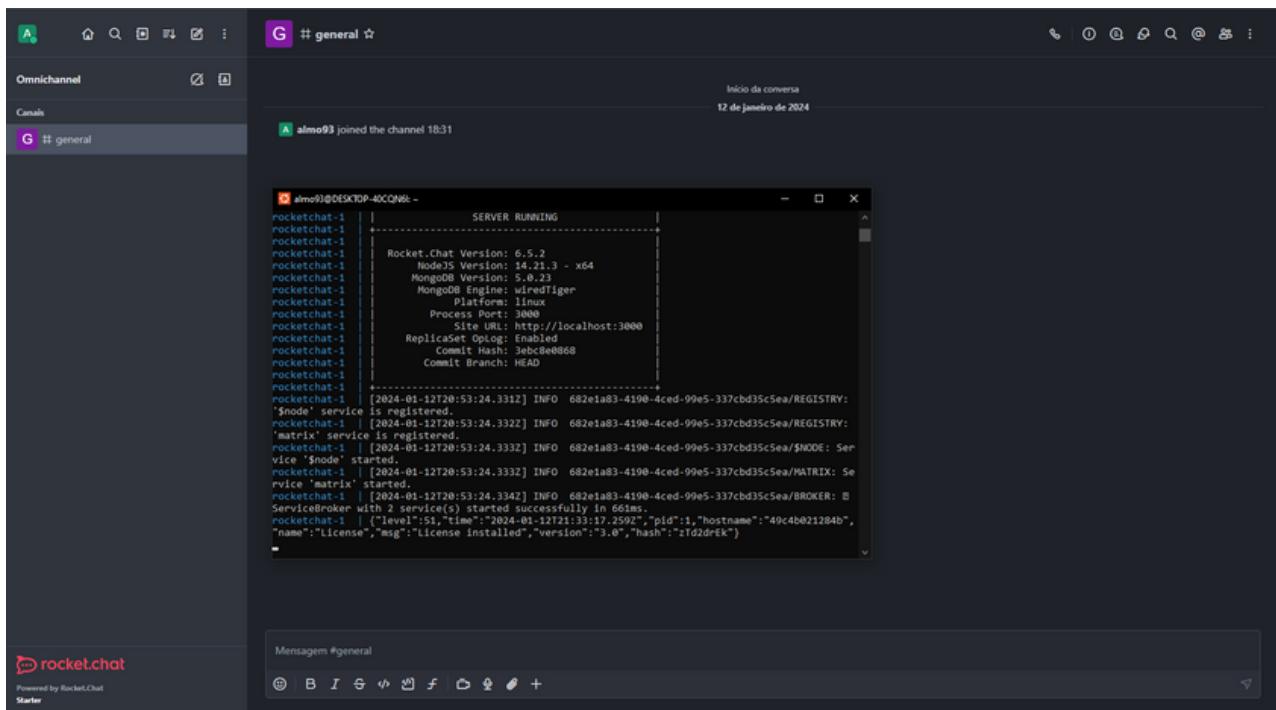
Username: almo93

Email: allanmelo@yahoo.com.br

Password: @lmoq12we3

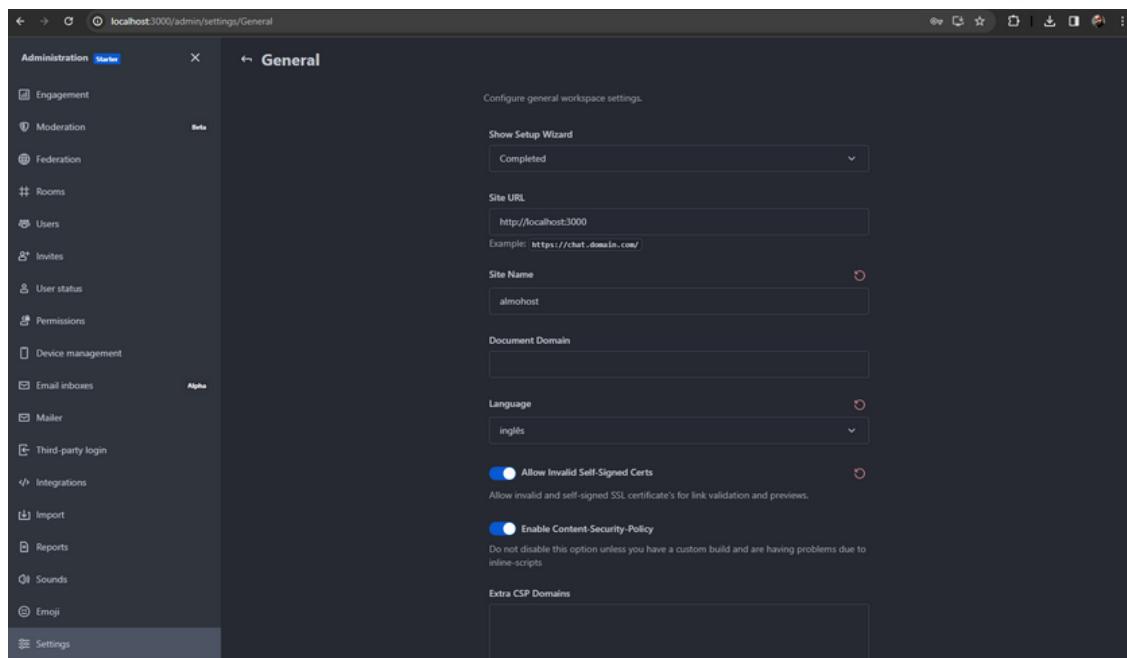
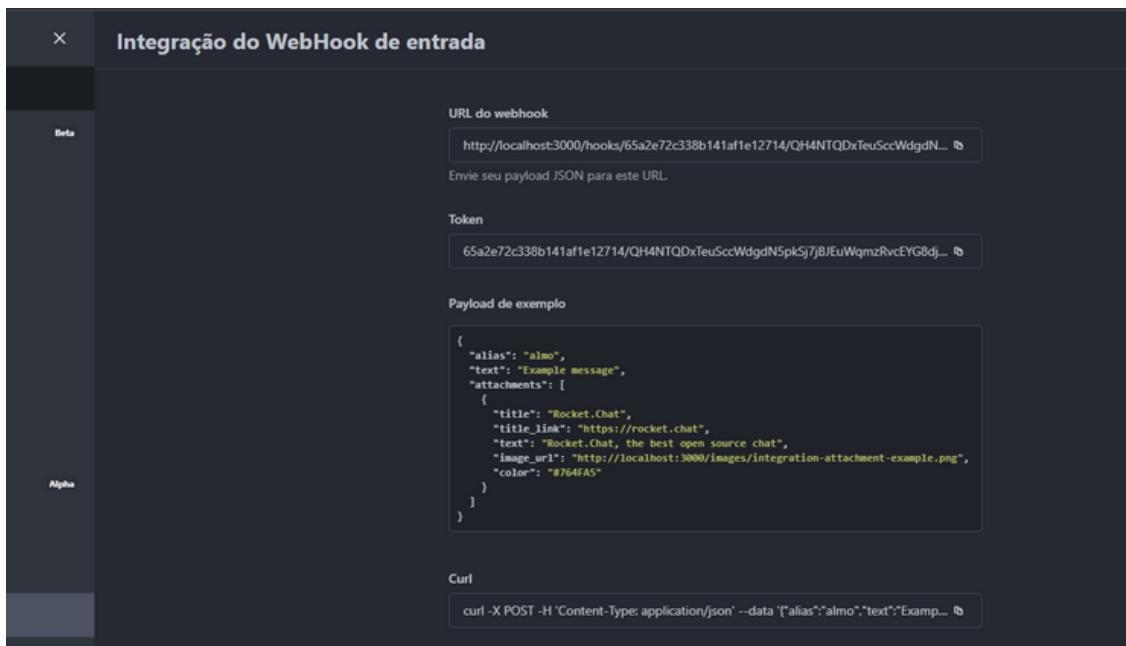
The email was verified, and access was successfully completed.





Rocket.Chat API

9 - After analysis, it was decided to use Postman for automated testing to make the process more agile. Within the Rocket.Chat platform, go to Settings -> Integrations, and create an ENDPOINT to acquire the Token (65a2e72c338b141af1e12714/QH4NTQDxTeuSccWdgdN5pkSj7jBJEuWqmzRvcEYG8djqcewT). Then, in the REST section, follow the configurations provided at <https://developer.rocket.chat/reference/api/rest-api>



10 -With Postman open, I performed the following command:

`curl http://localhost:3000/api/v1/login -d "user=almo93&password=@lmoq12we3"`
and obtained the following result:

The screenshot shows the Postman interface with a collection named 'API Rocket.chat Test'. Under the 'POST Login' request, the body is set to 'raw' JSON with the following content:

```
1 ["status": "success", "data": {"userId": "vk077pd0gct8Q9PcL", "authToken": "vYu2I8vxXENoDx547KmPZ-AuP2f369PdPhRj-c0_A", "me": {"_id": "vk077pd0gct8Q9PcL", "services": {"password": "bcrypt": "$2b$12$0hJtaSvNQmuw/gpU0Ujg0sfuUtvYcoMwixjywhufl"}, "email2fa": {"enabled": true}}, "emails": [{"address": "alamo@yahoo.com", "verified": false}, {"roles": [{"user": "admin"}, {"status": "active"}]}, {"language": "en"}, {"username": "almo93"}, {"utcOffset": -3}, {"settings": {"profile": 1}, "preferences": [{"enableAutosave": true, "idleTimeout": 3000, "desktopNotificationRequiresInteraction": false}, {"saveMobileBandwidth": true, "collapseMediaByDefault": false, "hideUsernames": false, "hideFlexTab": false, "displayAvatar": true, "desktopNotifications": "all", "pushNotifications": "all", "unreadAlert": true, "useEmoji": true, "convertAccents": true, "autoImageLoad": true, "sidebarGroupType": "activity", "themeAppearence": "auto", "sidebarViewMode": "medium", "sidebarDisplayAvatar": true, "sidebarShowUnread": false, "sidebarSortby": "activity", "showThreadsInMainChannel": false, "alsoSendThreadToChannel": "default", "sidebarShowFavorites": true, "sendDocEnter": "normal", "emailNotificationMode": "mentions", "newRoomNotification": "door", "newMessageNotification": "chime", "muteFocusedConversations": true, "notificationsSoundVolume": 100, "omnichannelTranscriptEmail": false, "notifyCalendarEvents": true, "enableMobileRing": true, "omnichannelTranscriptPDF": false}], "avatarUrl": "http://localhost:3000/avatar/almo93"}]}
```

The response status is 200 OK, time 146 ms, size 2.27 KB.

This screenshot is identical to the one above, but the 'Headers' tab is selected in the request configuration. The headers listed are:

- Content-Type: application/json
- Content-Length: <calculated when request is sent>
- Host: <calculated when request is sent>
- User-Agent: PostmanRuntime/7.36.0
- Accept: */*
- Accept-Encoding: gzip, deflate, br
- Connection: keep-alive
- X-Auth-Token: 65a2e72c33bb14afe12714/QH4NTQdxTeuScCwWgdGNS...
- X-User-Id: almo93

The response body and status are the same as in the previous screenshot.

11- To create the user through the endpoint, I used the following command:

```
curl -H "X-Auth-Token: bDEi8JlFOZ_e9ydWdfoiTcb-PCUP_ZQ3YFW0nad6_m7" \-H "X-User-Id: vk87PpGLgCEBQ89CL" \-H "Content-type:application/json"
\http://localhost:3000/api/v1/users.create \-d '{"name": "Cecilia", "email": "cecilia@gmail", "password": "26015180", "username": "cecilia.rocket", "roles": ["bot", "user"]}'
```

And this was the obtained response:

The screenshot shows the Postman interface with a successful API call. The request URL is `http://localhost:3000/api/v1/users.create`. The response status is 200 OK, and the response body is a JSON object representing the created user.

```
1 {
2   "user": {
3     "_id": "23ExgzSS1v3egyZg",
4     "createdAt": "2024-01-14T01:50:56.763Z",
5     "username": "cecilia.rocket",
6     "emails": [
7       {
8         "address": "cecilian@gmail.com",
9         "verified": false
10      }
11    ],
12    "type": "user",
13    "roles": [
14      "bot",
15      "user"
16    ],
17    "status": "offline",
18    "active": true,
19    "updatedAt": "2024-01-14T01:50:56.843Z",
20    "_rooms": [
21      "GENERAL"
22    ]
23 }
```

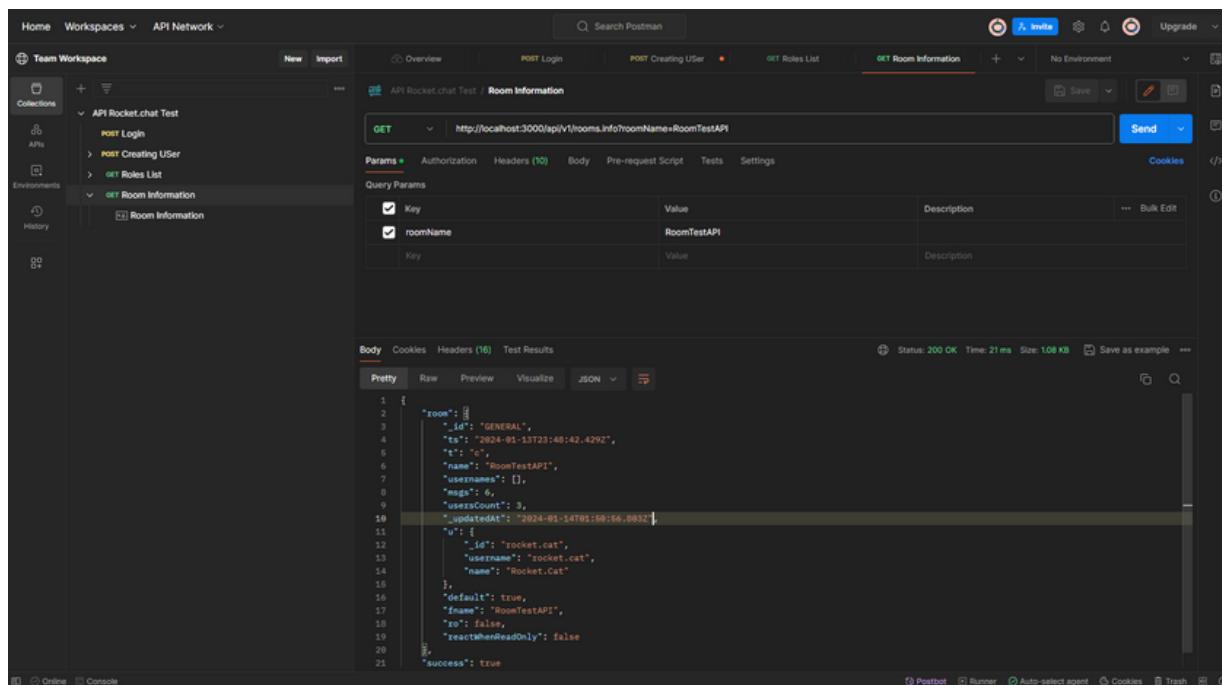
The screenshot shows the Postman interface with a successful API call. The request URL is `http://localhost:3000/api/v1/users.create`. The response status is 200 OK, and the response body is a JSON object representing the created user.

```
1 {
2   "user": {
3     "_id": "23ExgzSS1v3egyZg",
4     "createdAt": "2024-01-14T01:50:56.763Z",
5     "username": "cecilia.rocket",
6     "emails": [
7       {
8         "address": "cecilian@gmail.com",
9         "verified": false
10      }
11    ],
12    "type": "user",
13    "roles": [
14      "bot",
15      "user"
16    ],
17    "status": "offline",
18    "active": true,
19    "updatedAt": "2024-01-14T01:50:56.843Z",
20    "_rooms": [
21      "GENERAL"
22    ]
23 }
```

12 - For the room information, I used the following command:

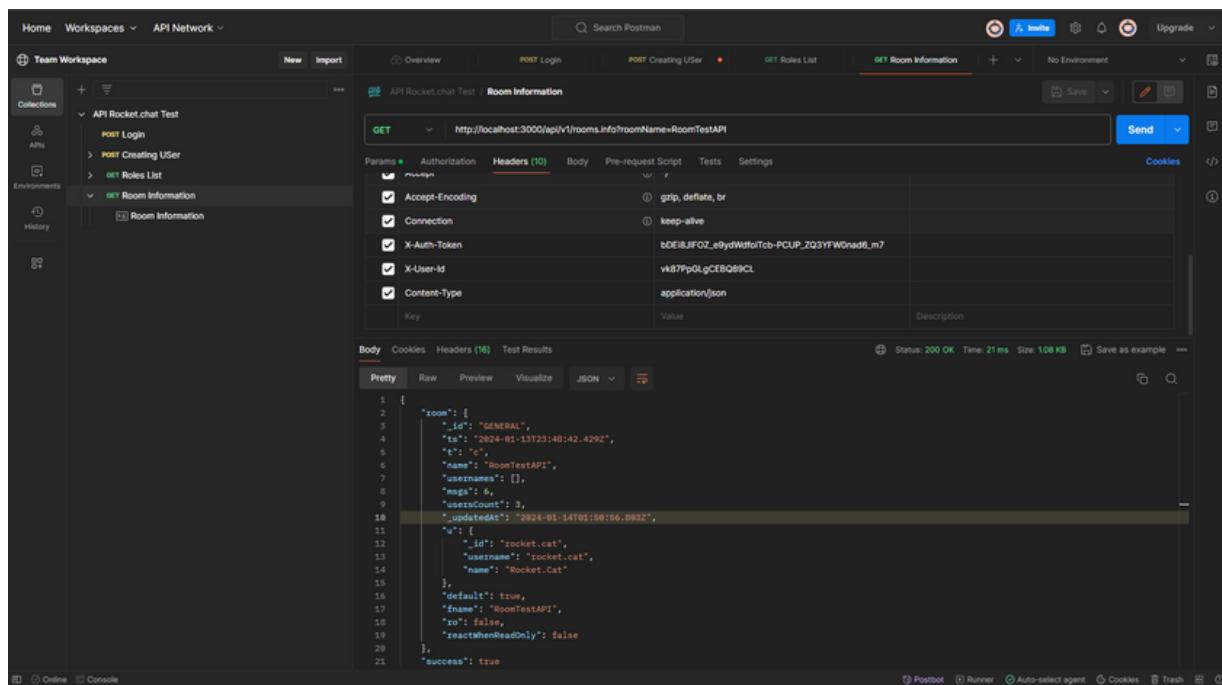
```
curl -H "X-Auth-Token: bDEi8JlFOZ_e9ydWdfoiTcb-PCUP_ZQ3YFW0nad6_m7" \-H  
"X-User-Id: aobEdbYhXfu5hkeqG" \http://localhost:3000/api/v1/rooms.info?  
roomName=RoomTestAPI
```

The response :



Postman screenshot showing the results of a GET request to `/api/v1/rooms.info?roomName=RoomTestAPI`. The response body is as follows:

```
1 {  
2   "room": [  
3     {"_id": "GENERAL",  
4      "ts": "2024-01-13T23:48:42.429Z",  
5      "t": "+",  
6      "name": "RoomTestAPI",  
7      "username": [],  
8      "msgs": 1,  
9      "usersCount": 1,  
10     "updatedAt": "2024-01-14T01:08:56.083Z",  
11     "u": [  
12       {"_id": "rocket.cat",  
13          "username": "rocket.cat",  
14          "name": "Rocket.Cat"  
15        ],  
16        "default": true,  
17        "fname": "RoomTestAPI",  
18        "ro": false,  
19        "reactWhenReadOnly": false  
20     ],  
21   "success": true
```



Postman screenshot showing the results of a GET request to `/api/v1/rooms.info?roomName=RoomTestAPI` with additional headers. The response body is identical to the previous screenshot.

Headers added:

- Accept-Encoding: gzip, deflate, br
- Connection: keep-alive
- X-Auth-Token: bDEi8JlFOZ_e9ydWdfoiTcb-PCUP_ZQ3YFW0nad6_m7
- X-User-Id: aobEdbYhXfu5hkeqG
- Content-Type: application/json

Response body:

```
1 {  
2   "room": [  
3     {"_id": "GENERAL",  
4      "ts": "2024-01-13T23:48:42.429Z",  
5      "t": "+",  
6      "name": "RoomTestAPI",  
7      "username": [],  
8      "msgs": 1,  
9      "usersCount": 1,  
10     "updatedAt": "2024-01-14T01:08:56.083Z",  
11     "u": [  
12       {"_id": "rocket.cat",  
13          "username": "rocket.cat",  
14          "name": "Rocket.Cat"  
15        ],  
16        "default": true,  
17        "fname": "RoomTestAPI",  
18        "ro": false,  
19        "reactWhenReadOnly": false  
20     ],  
21   "success": true
```

13 - To obtain the list of all permissions, I used the following command:

```
curl -H "X-Auth-Token: bDEi8JlFOZ_e9ydWdfoiTcb-PCUP_ZQ3YFW0nad6_m7" \
-H "X-User-Id: vk87PpGLgCEBQ89CL" \
http://localhost:3000/api/v1/roles.list
```

The response :

The screenshot shows the Postman interface with a collection named "API Rocket.chat Test". Under this collection, there are three items: "POST Login", "GET Roles List", and "GET Room Information". The "GET Roles List" item is selected. The "Headers" tab is active, showing the following headers:

Key	Value	Description
Cache-Control	no-cache	
Postman-Token	<calculated when request is sent>	
Host	<calculated when request is sent>	
User-Agent	PostmanRuntime/7.36.0	
Accept	*/*	
Accept-Encoding	gzip, deflate, br	
Connection	keep-alive	
X-Auth-Token	bDEi8JlFOZ_e9ydWdfoiTcb-PCUP_ZQ3YFW0nad6_m7	
X-User-Id	vk87PpGLgCEBQ89CL	

The "Body" tab shows the JSON response:

```
1  {
2   "roles": [
3     {
4       "_id": "admin",
5       "scope": "users",
6       "description": "Admin",
7       "mandatory2fa": false,
8       "name": "admin",
9       "protected": true
10    },
11    {
12      "_id": "moderator",
13      "scope": "subscribers"
14    }
15  ]
```

Note : All the response will be included on a txt file