

**Technical Assessment**

**Rocket.Chat Enterprise Solutions Team**

**Deploy your own Rocket.Chat Server Instance**

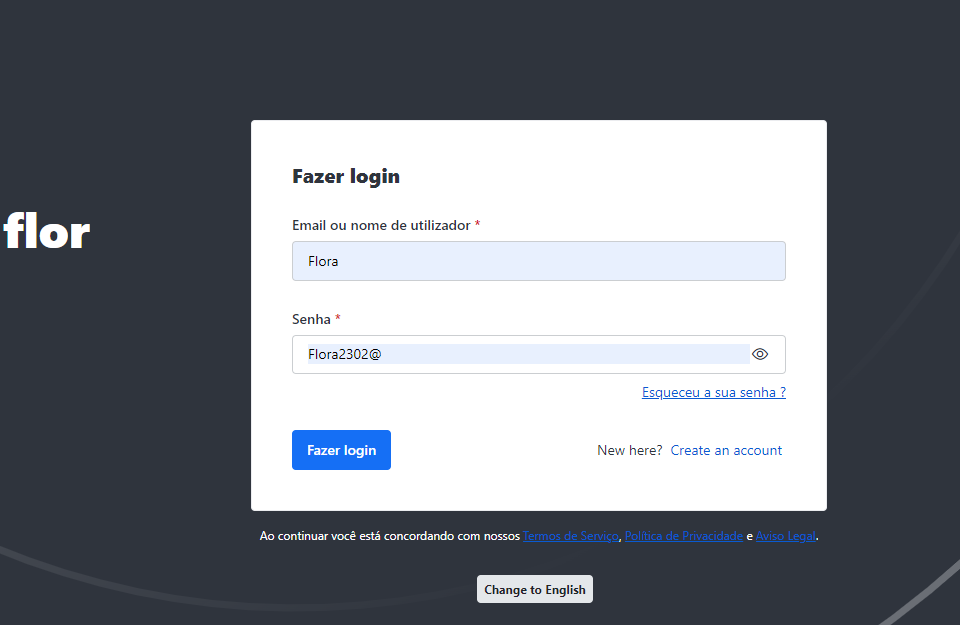
**Mandatory:** Install and set up your Rocket.Chat environment and provide us the URL and a user/password so we can access it. You can use any of the supported installation methods we have (further info on https://docs.rocket.chat/ and https://github.com/RocketChat/Rocket.Chat). There's no restriction at all, but we'll evaluate how challenging your deployment choice was. Please also don't forget to document the steps you did, explaining the installation process. You must also share this summary with us. Hint: there are tools capable of exposing your localhost through a secure connection.

**I took the free saas trial, because my computer is having some problems, but the intention was to install docker to carry out the installation.**

**In this case I followed the steps, as it is very intuitive and created my space with the name flor**

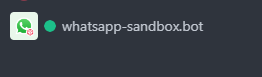
**access information:**

[**https://flor.rocket.chat/home**](https://flor.rocket.chat/home)

****

**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 went to the app - marketplace information and selected WhatsApp sandbox (available for the version I was using) and click install**



**So, a few more simple steps, the bot itself guides you, get the API key on 360  
Then put this api key in the app settings**

**this is just a test environment**

**Optional:** Set a reverse proxy, a domain, and an SSL certificate so your server can use HTTPS protocol.

**Important:** There is no need to buy a domain for this, as this is optional. If you have a domain to spare, feel free to use it, but we don't want you to spend money on this test. �

I don't have my own domain to test

**Rocket.Chat APIs Test**

After creating your Rocket.Chat server, run the following API queries/updates, and provide us the results. We need only the request/responses for each operation (you can share with us screenshots or the actual files/collections). Note that the API calls should be done to your server.

The documentation is available at https://docs.rocket.chat/.

1. Create a new user via an API endpoint

2. Get the room information via an API endpoint

3. Get a list of all user roles in the system via an API endpoint

**Important:** The results of the tests must be published in a GitHub repository named “Rocket.Chat\_ES\_YourFirstName\_YourLastName”

docs.rocket.chat

**Welcome to Rocket.Chat Docs**

Here you can find all the docs about Rocket.Chat server and client.