



MANUAL DE INSTALAÇÃO DO MYZAP 2.0

AMBIENTE LOCAL – WINDOWS 10/WSL/UBUNTU

DANILO RUIZ DA SILVA

 (11) 95199-7553  (11) 93035-1407

Vs. 1.2

Sumário

1. Pré-Requisitos	3
2. Ativação do WSL (Windows Subsystem for Linux) no Windows 10	3
3. Instalação do Ubuntu 20.04 LTS.....	6
4. Instalação do Insomnia.....	9
5. Criação do projeto no Firebase para armazenamento das sessões.....	11
6. Instalação e configuração do MyZAP 2.0.....	21
7. Testando a API MyZAP 2.0.....	34
8. Créditos	44

1. Pré-Requisitos

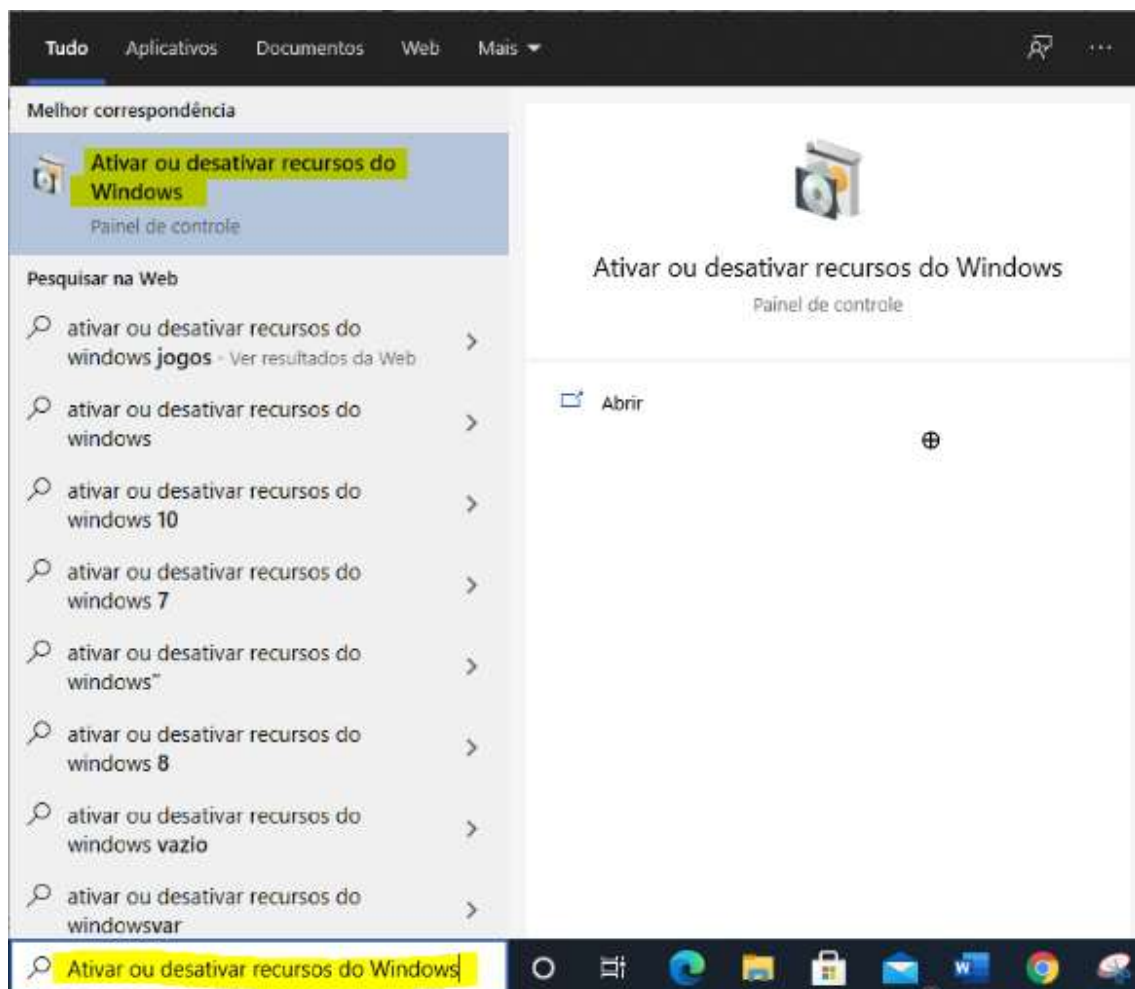
- ✓ Conexão com a internet;
- ✓ Acesso ao Git (usuário e senha) do repositório <https://github.com/billbarsch/myzap2.0>;
- ✓ Windows 10 64 bits

2. Ativação do WSL (Windows Subsystem for Linux) no Windows 10

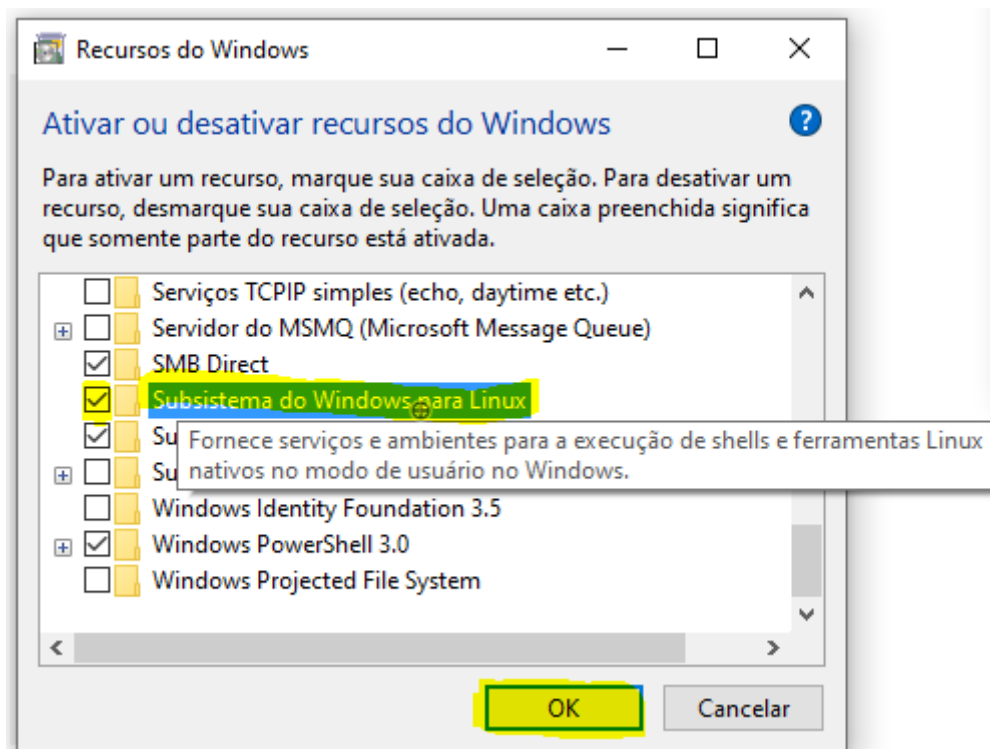
O WSL - Subistema Windows para Linux é um módulo do sistema operacional Windows 10 que visa a disponibilizar um ambiente Linux compatível no sistema da Microsoft, de forma que se possam executar programas (baseados em texto) nativos dos sistemas GNU/Linux dentro do próprio Windows sem a necessidade de emuladores ou do uso de máquinas virtuais. Vários distros Linux distintos podem, em princípio, ser instalados sobre este módulo.

Para a instalação do MyZAP 2.0, iremos utilizar o Ubuntu 20.04 LTS instalado diretamente no Windows 10 via WSL . Para isso necessitamos habilitar este recurso WSL do Windows de acordo com as instruções abaixo:

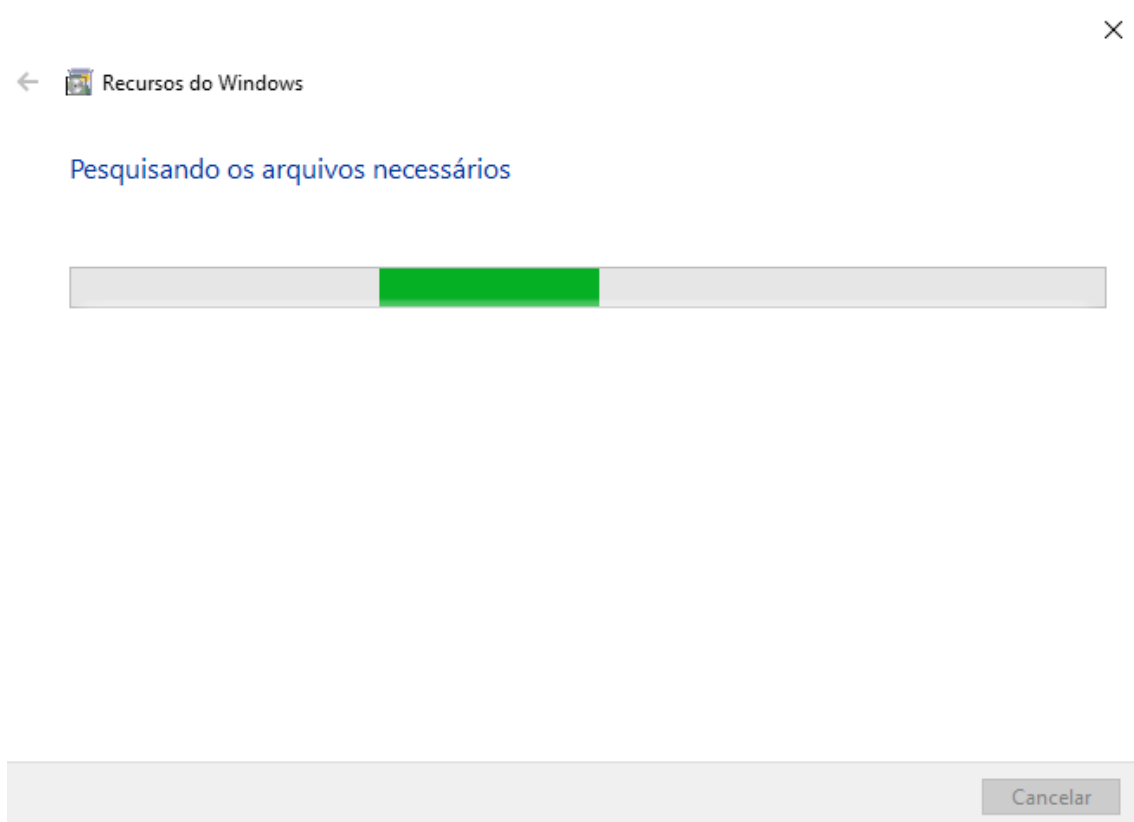
Na barra de pesquisa do Windows digite o texto “Ativar ou desativar recursos do Windows” e clique para abrir o painel onde iremos habilitar o recurso “Subsistema do Windows para Linux”.

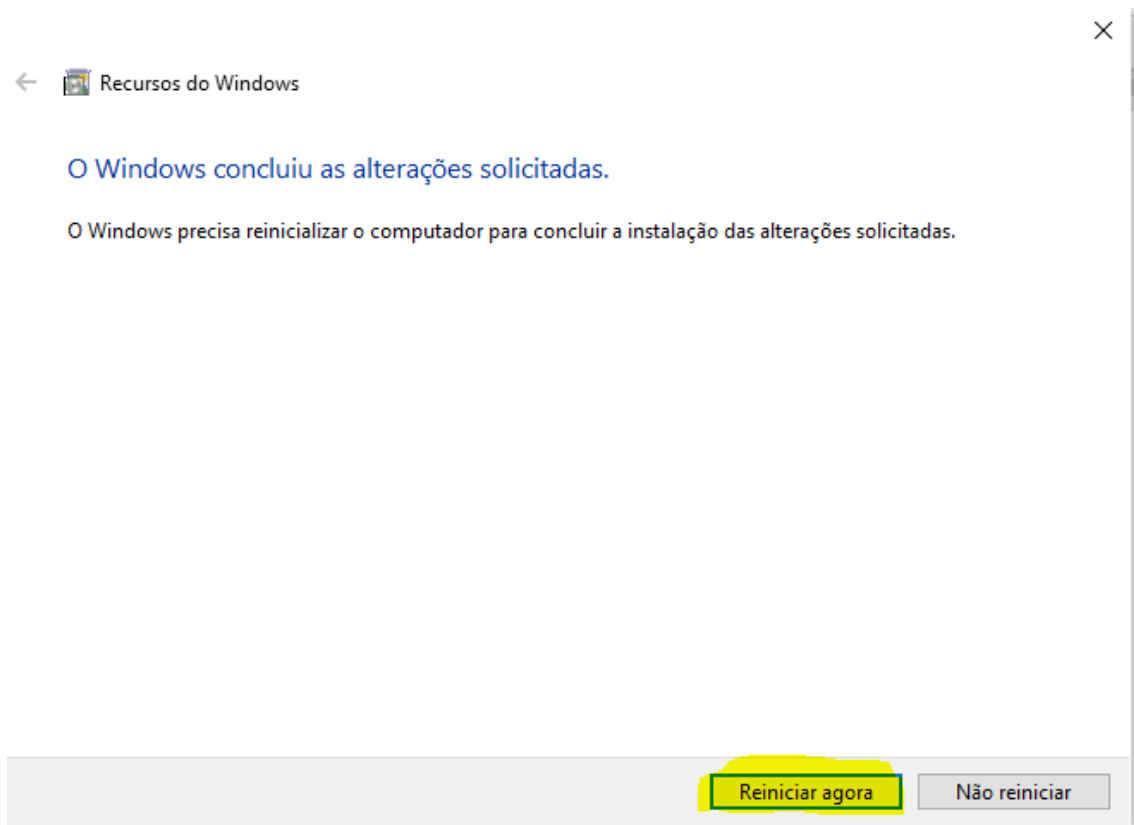


Após abrir o painel role o scroll até o final e selecione a opção “Subsistema do Windows para Linux”.



Clique na opção “OK”, após aplicar a habilitação do WSL clique na opção “Reiniciar agora” para reiniciar o computador. A ativação do recurso WSL estará ativada com sucesso.

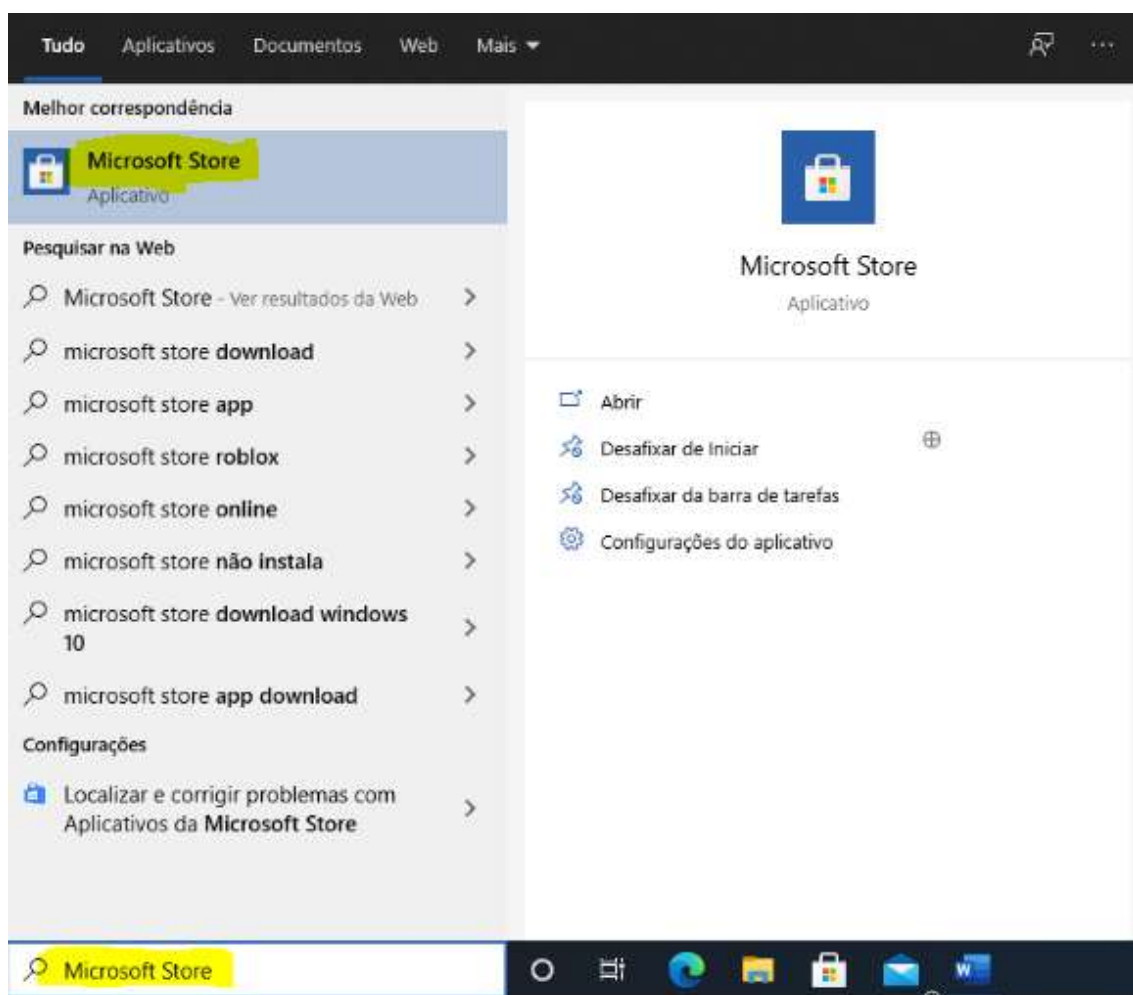




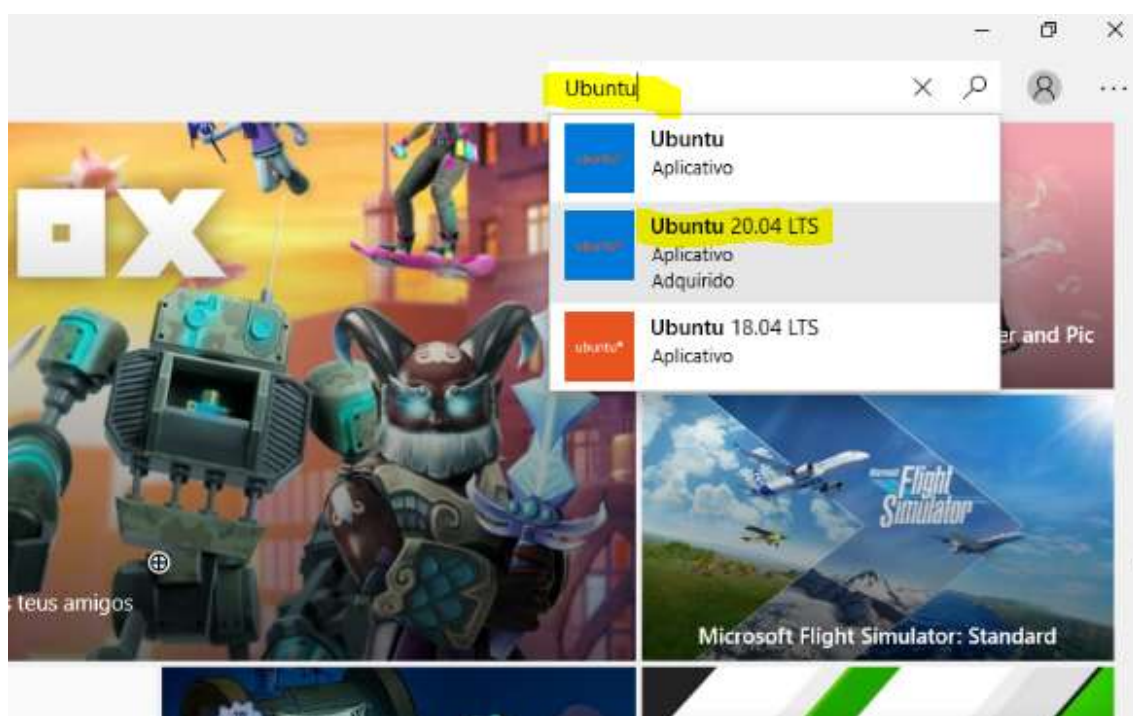
3. Instalação do Ubuntu 20.04 LTS

O Ubuntu 20.04 LTS é disponibilizado na própria loja de aplicativos da Microsoft (Microsoft Store), além de outras versões do Ubuntu e outras distros do Linux. Para realizar a instalação do Ubuntu basta seguir os seguintes passos:

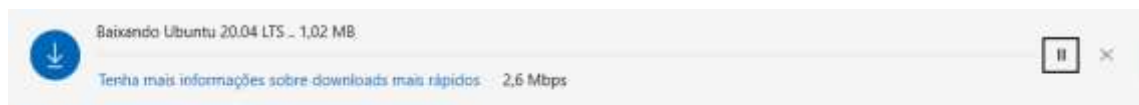
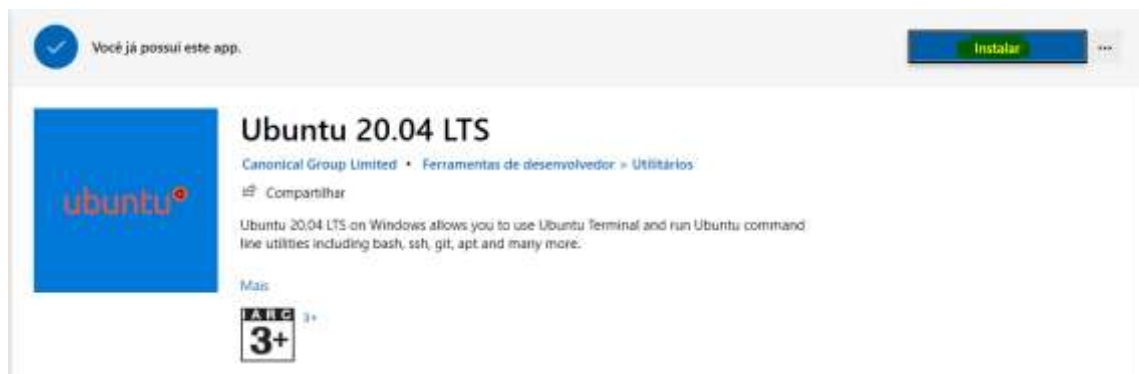
Na barra de pesquisa do Windows pesquise por “Microsoft Store”



Clique em “Microsoft Store” e na barra de pesquisa de aplicativos da loja digite “Ubuntu” e selecione “Ubuntu 20.04 LTS”.



Selecione a opção “Instalar” e aguarde realizar o download e a instalação do Ubuntu.



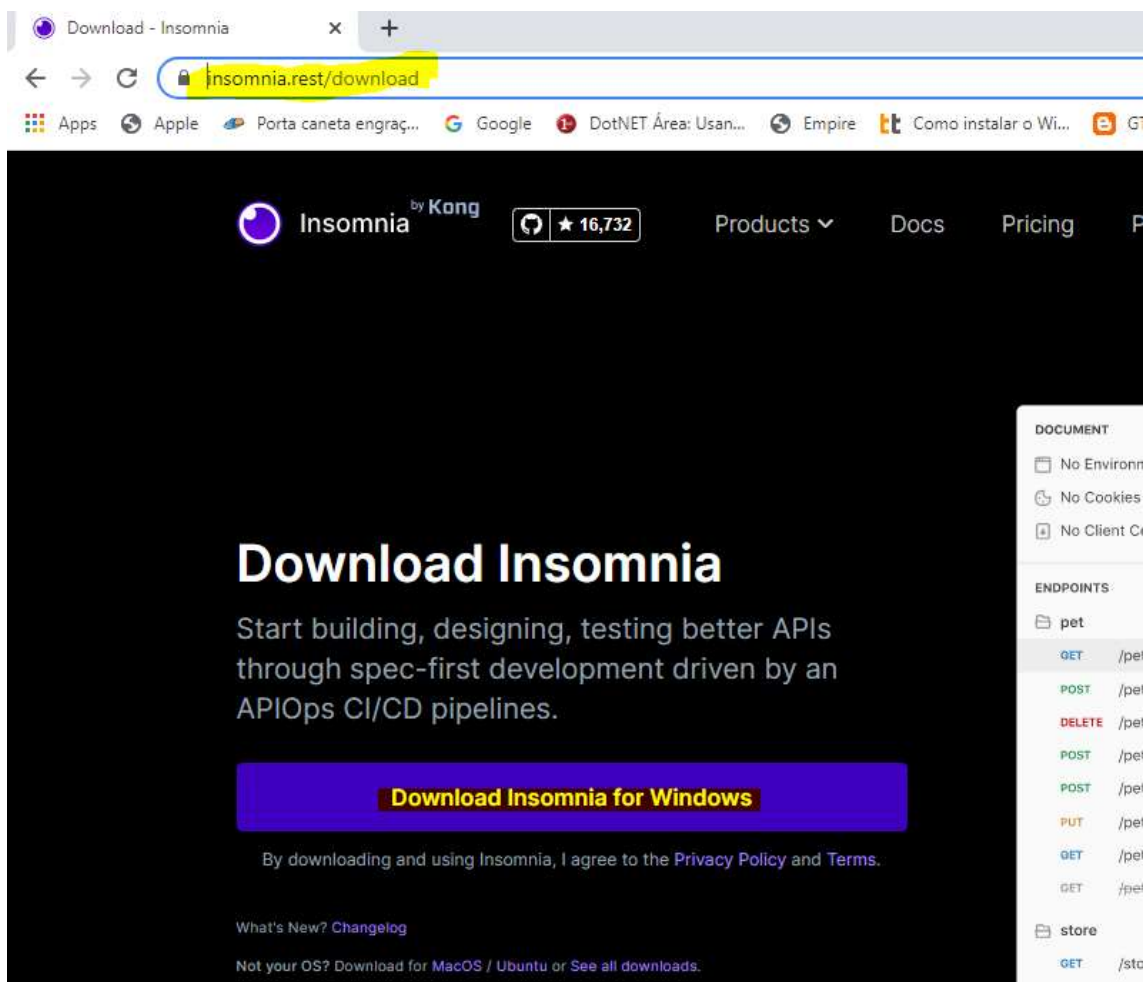
Finalizada a instalação do Ubuntu 20.04 LTS podemos fechar a loja de aplicativos clicando na opção “X”.



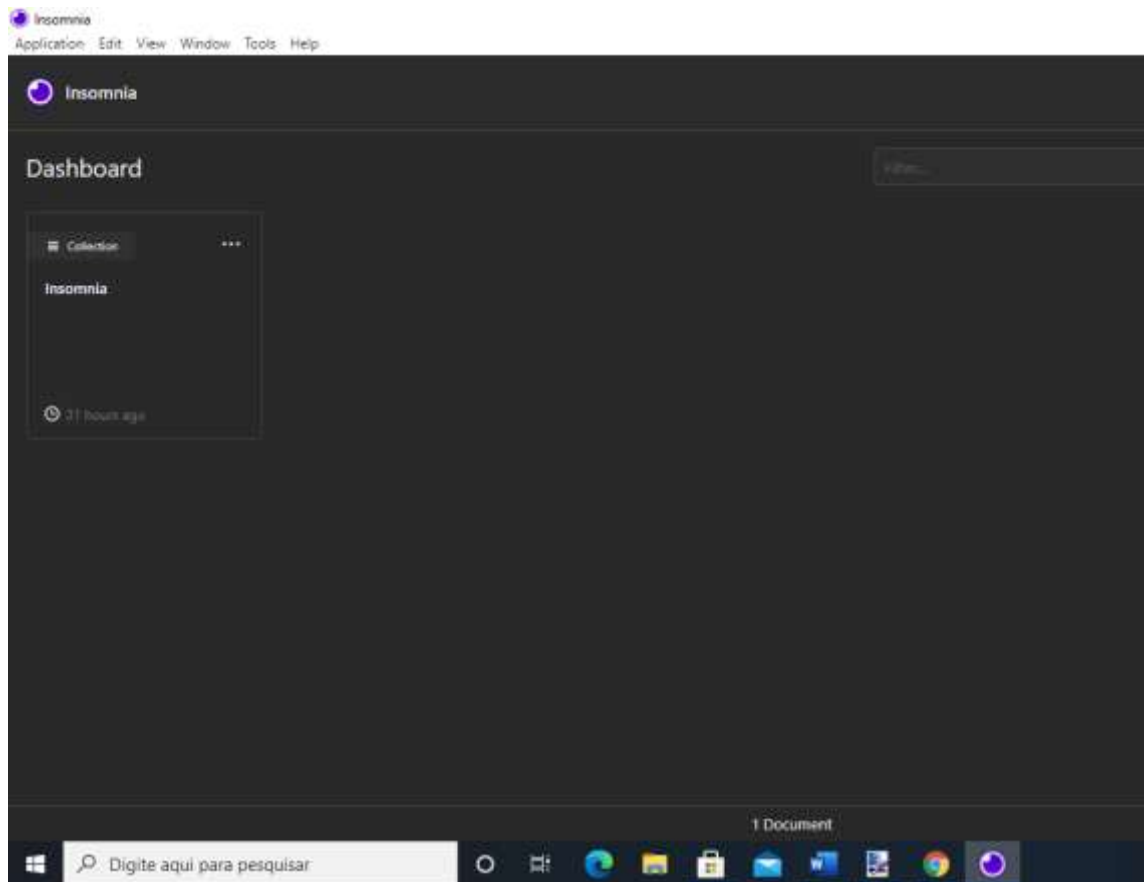
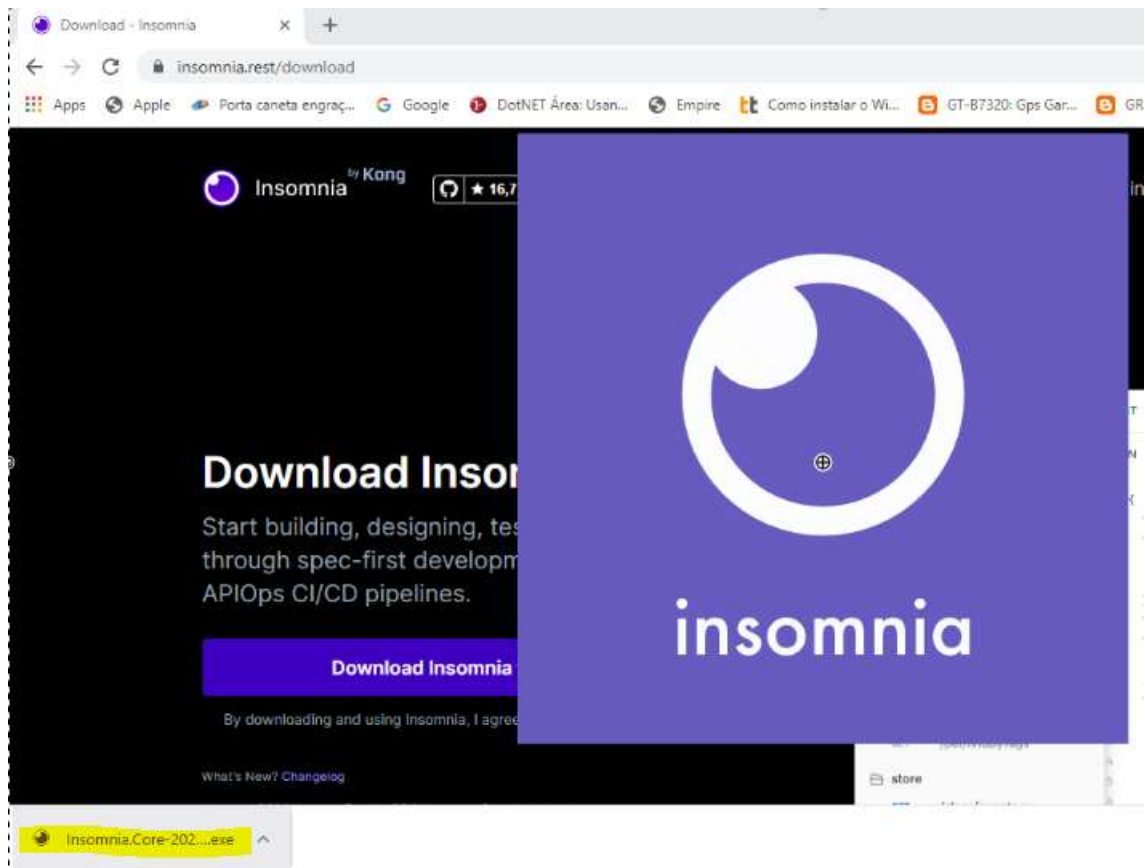
4. Instalação do Insomnia

Para realizar as chamadas das operações da API MyZAP 2.0 iremos utilizar a ferramenta Insomnia. Basta executar os passos abaixo para realizarmos a instalação da ferramenta:

Acessar a url <https://insomnia.rest/download> e selecionar a opção “Download Insomnia for Windows” para realizar o download de instalação da ferramenta.



Após realizar o download do arquivo “Insomnia.Core-2021.4.0.exe” (ou versão corrente), clicar duas vezes no arquivo para iniciar a instalação.



5. Criação do projeto no Firebase para armazenamento das sessões.

Para armazenar as sessões do MyZAP 2.0 utilizando o Firebase, precisamos criar um projeto de Database seguindo as instruções abaixo:

Acessar a url <https://firebase.google.com>;

Realizar autenticação com sua conta do Google selecionando a opção “Fazer login” informando seu e-mail e senha;





The image shows a Google login page with the Google logo at the top. Below the logo, the text "Fazer login" (Login) is displayed, followed by "Prosseguir para o Google Developers" (Proceed to Google Developers). A text input field labeled "E-mail ou telefone" (Email or phone) contains the email address "daniloruiz2003@gmail.com". Below the input field, there is a link "Esqueceu seu e-mail?" (Forgot your email?). Further down, a message states "Não está no seu computador? Use o modo visitante para fazer login com privacidade. Saiba mais" (Not on your computer? Use visitor mode to log in with privacy. Learn more). At the bottom, there are two buttons: "Criar conta" (Create account) and "Próxima" (Next).

Google

Fazer login

Prosseguir para o Google Developers

E-mail ou telefone

daniloruiz2003@gmail.com

[Esqueceu seu e-mail?](#)

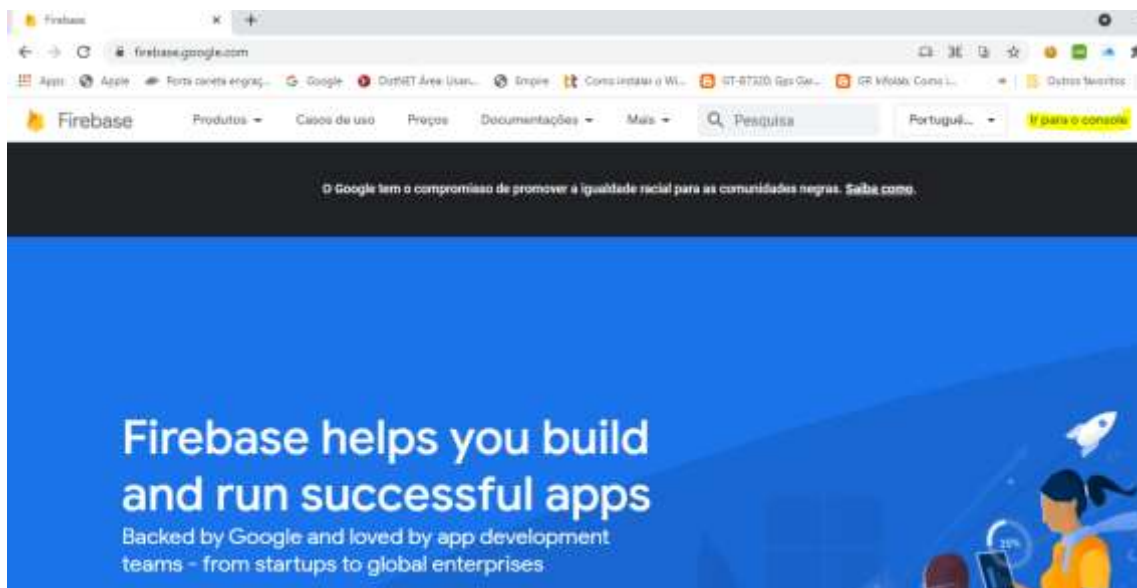
Não está no seu computador? Use o modo visitante para fazer login com privacidade. [Saiba mais](#)

[Criar conta](#)

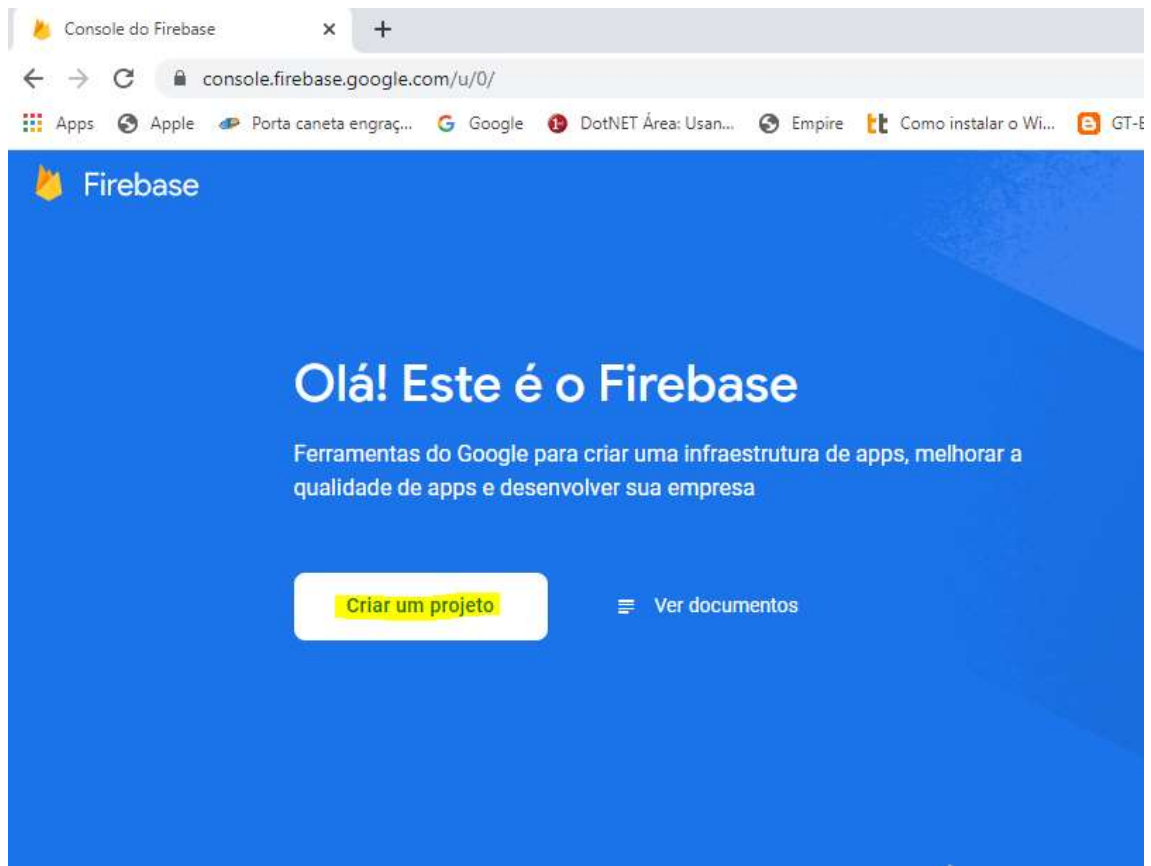
[Próxima](#)

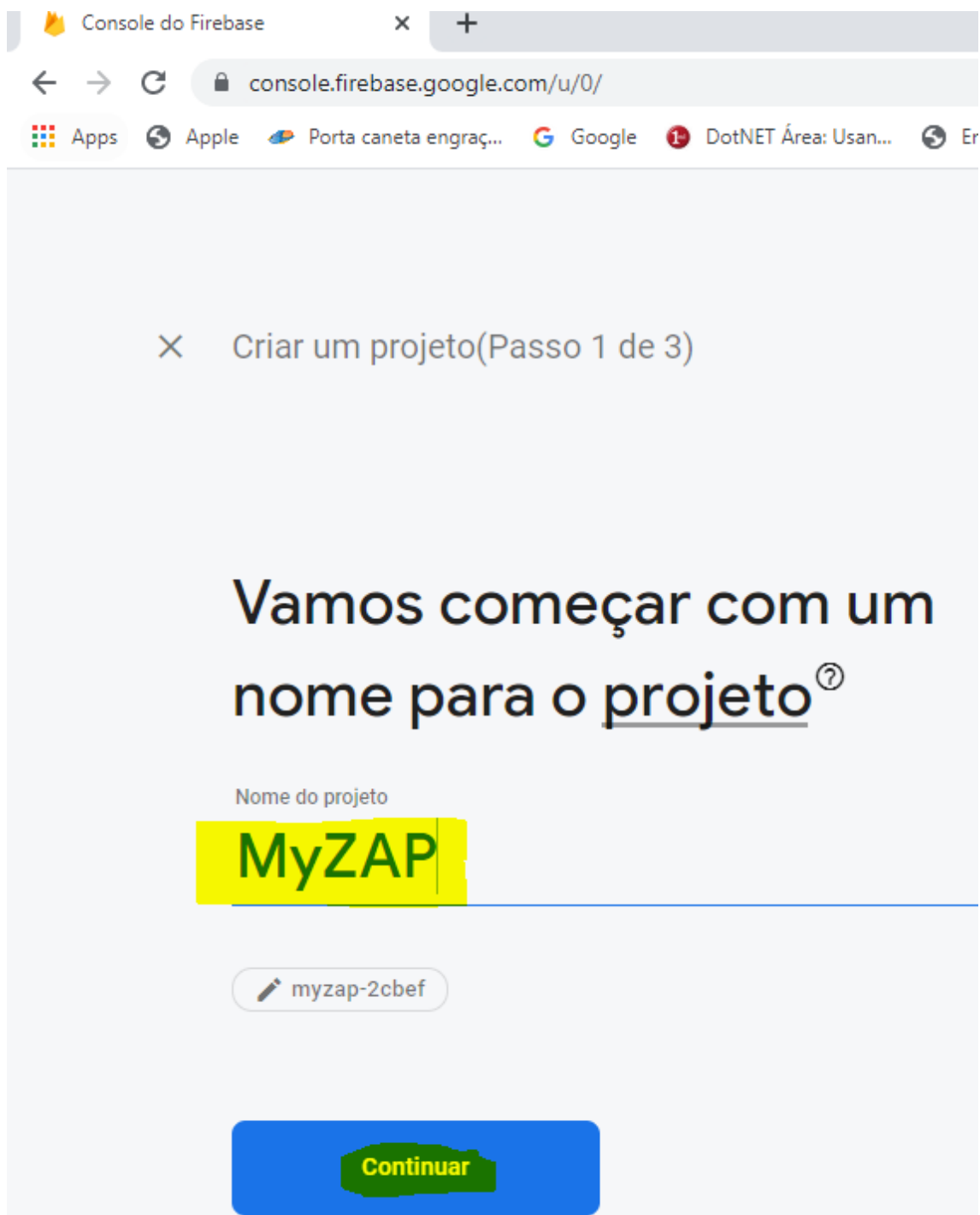
Realizar autenticação com sua conta do Google selecionando a opção “Fazer login” informando seu e-mail e senha;

Selecione a opção “Ir para o console”:



Clique na opção “Criar um projeto”, defina o nome “MyZAP” para o projeto e selecione a opção “Continuar”:





Desative a opção “Ativar o Google Analytics neste projeto” e selecione a opção “Criar projeto”.

Console do Firebase

console.firebase.google.com/u/0/

Apps Apple Porta caneta engraç... Google DotNET Área: Usan... Empire Como instalar o Wi... GT-B7320: G

× Criar um projeto(Passo 2 de 2)

O Google Analytics ativa:

× Teste A/B ?	× Usuários sem falhas ?
× Segmentação de usuários em produtos do Firebase ?	× Gatilhos do Cloud Functions com base em eventos ?
× Previsão do comportamento de usuários ?	× Geração de relatórios ilimitada gratuita ?


☒ Ativar o Google Analytics neste projeto
Recomendado

[Anterior](#) [Criar projeto](#)

Console do Firebase

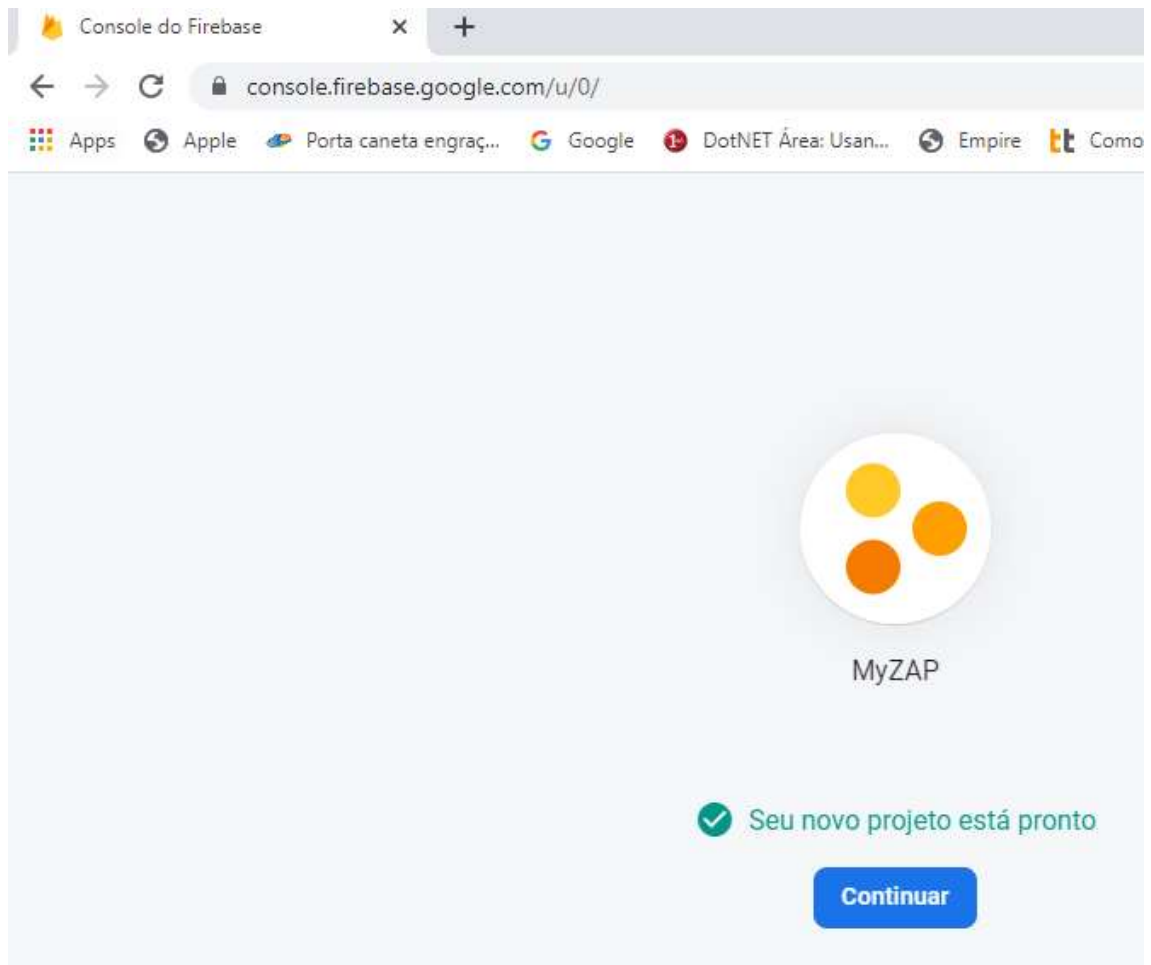
console.firebase.google.com/u/0/

Apps Apple Porta caneta engraç... Google DotNET Área: Usan... Empire Con

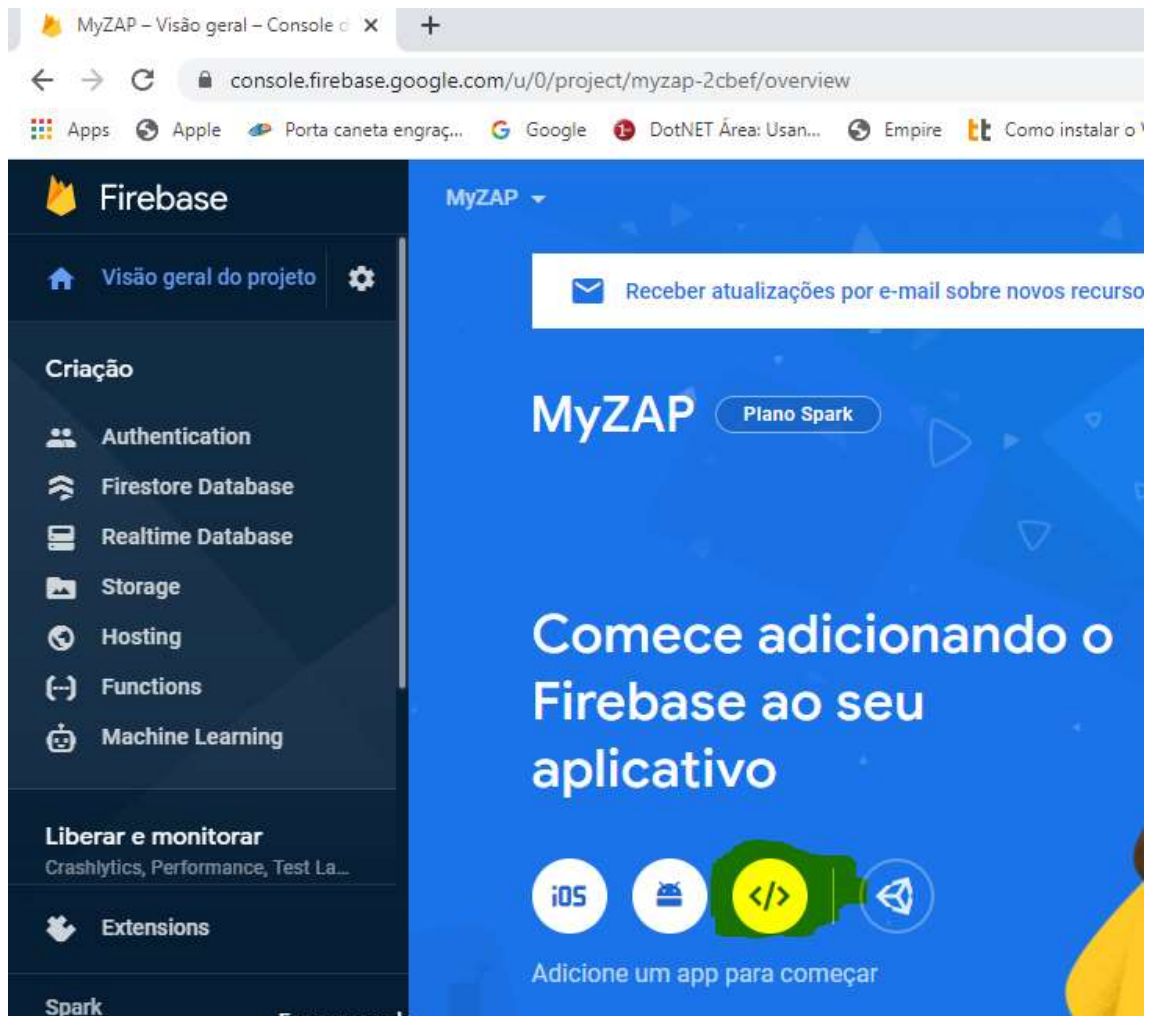


Criando seu projeto...

MyZAP



Adicione o Firebase ao seu aplicativo selecionando a opção “</>” abaixo:



Defina o apelido do app “MyZAP” e selecione a opção “Registrar app”:

MyZAP – Visão geral – Console d x +

← → ↻ console.firebase.google.com/u/0/project/myzap-2cbef/overview

Apps Apple Porta caneta engraç... Google DotNET Área: Usan... Empire Como instalar o Wi... GT-B7320

× Adicionar o Firebase ao seu app da Web

- 1 Registrar app**

Apelido do app ?

MyZAP

☐ Também configure o **Firebase Hosting** para este app. [Saiba mais](#)

A configuração do Hosting também pode ser feita depois. Comece a usar a qualquer momento sem pagar nada.

Registrar app
- 2 Adicionar o SDK do Firebase**

Essa etapa é muito importante, na próxima tela é exibido as informações de acesso ao Firebase que iremos configurar no arquivo de configuração do MyZAP mais adiante.

MyZAP – Visão geral – Console x +

← → ↺ console.firebase.google.com/u/0/project/myzap-2cbef/overview

Apps Apple Porta caneta engraç... Google DotNET Área: Usan... Empire Como instalar

```
<!-- The core Firebase JS SDK is always required and must
<script src="https://www.gstatic.com/firebasejs/8.7.0/fir

<!-- TODO: Add SDKs for Firebase products that you want t
https://firebase.google.com/docs/web/setup#available

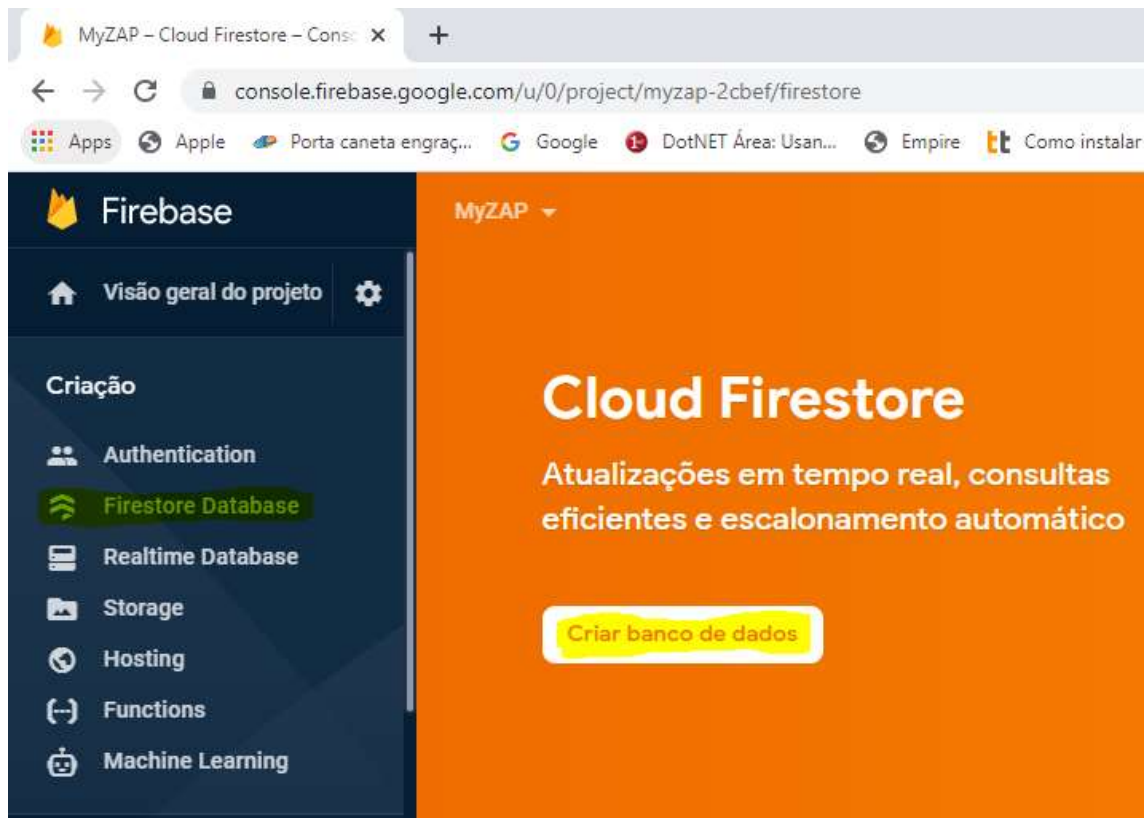
<script>
// Your web app's Firebase configuration
var firebaseConfig = {
  apiKey: "AIzaSyAHu4RoD4X9MHu8YpXUxaqYZf8XrYF42SY",
  authDomain: "myzap-2cbef.firebaseio.com",
  projectId: "myzap-2cbef",
  storageBucket: "myzap-2cbef.appspot.com",
  messagingSenderId: "938471440273",
  appId: "1:938471440273:web:856c5aa4982e0d96a62430"
};
// Initialize Firebase
firebase.initializeApp(firebaseConfig);
</script>
```

Saiba mais sobre o Firebase para Web: [Primeiros passos](#), [Referência da API W](#)

Você está usando npm e um bundler como Webpack ou Rollup? Confira o [SDK m](#) versão Beta.

Continuar no console

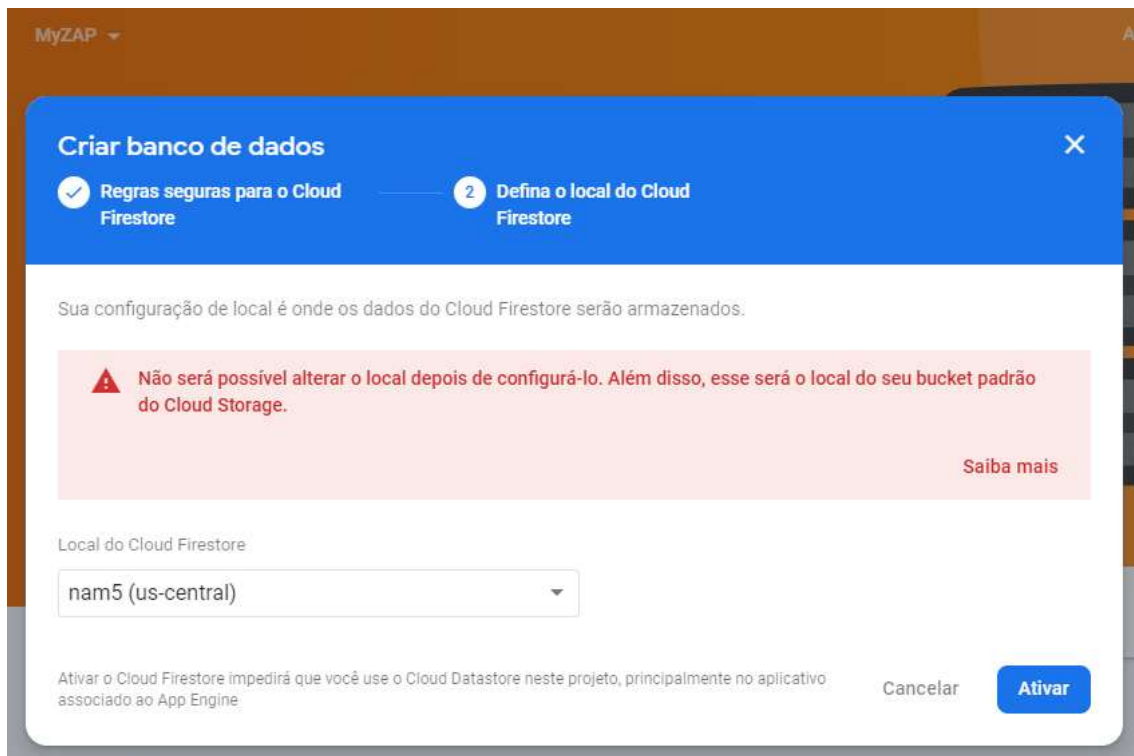
Selecione a opção “Firestore Database” e depois na opção “Criar banco de dados”.



Selecione a opção “Iniciar no modo produção” e depois na opção “Avançar”.



Selecione a opção “Ativar” e aguarde a criação do Database.



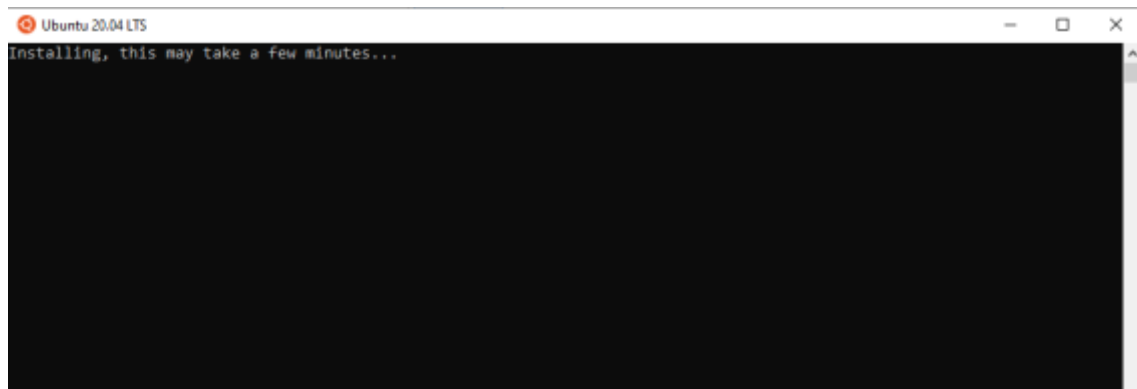
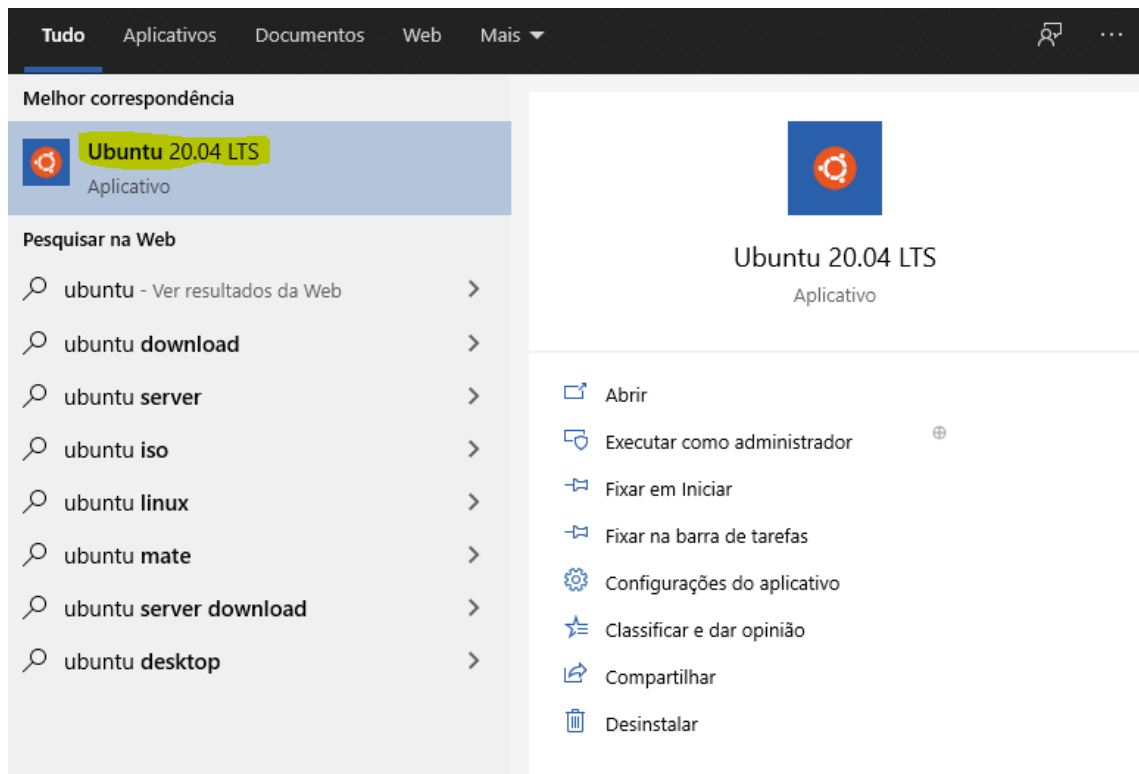
Selecione a opção “Regras” e edite a regra “allow read, write: if false;” para “allow read, write: if true;”. Após a edição da regra selecione a opção “Publicar” e estará finalizada a criação e configuração da base de dados.



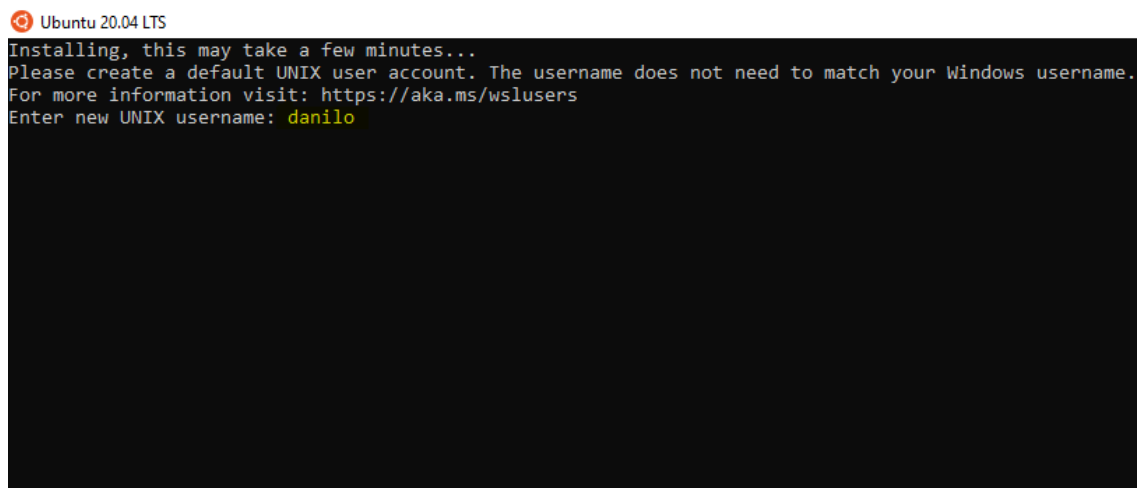
6. Instalação e configuração do MyZAP 2.0

Após a ativação do recurso WSL, as instalações do Ubuntu 20.04 LTS e do Insomnia e a criação da base de dados para armazenamento das sessões no Firebase, iremos instalar e configurar o MyZAP 2.0 de acordo com as instruções abaixo:

Abrir o Ubuntu 20.04 LTS digitando na barra de pesquisa do Windows 10 o texto “Ubuntu” e selecionando “Ubuntu 20.04 LTS”.



Defina um nome de usuário para o Linux na opção “Enter new UNIX username” e pressione “ENTER”. No caso do exemplo define o nome de usuário “danilo”.



Defina uma senha em “New password” e pressione ENTER. No caso do exemplo defini como “danilo”. Informe a mesma senha novamente e pressione ENTER.

```
danilo@DESKTOP-JL5AQJB: ~  
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: danilo  
New password:  
Retype new password:  
passwd: password updated successfully  
Installation successful!  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 4.4.0-19041-Microsoft x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:        https://ubuntu.com/advantage  
  
System information as of Sat Jul  3 12:48:04 -03 2021  
  
System load:  0.52      IPv4 address for eth4:  172.20.240.1  
Usage of /home: unknown IPv4 address for eth5:  192.168.56.1  
Memory usage: 42%      IPv4 address for eth6:  192.168.38.1  
Swap usage:   0%        IPv4 address for eth7:  192.168.175.1  
Processes:    7         IPv4 address for wifi0: 192.168.0.109  
Users logged in: 0  
  
1 update can be applied immediately.  
To see these additional updates run: apt list --upgradable  
  
The list of available updates is more than a week old.  
To check for new updates run: sudo apt update  
  
This message is shown once a day. To disable it please create the  
/home/danilo/.hushlogin file.  
danilo@DESKTOP-JL5AQJB:~$
```

Copie e cole o comando abaixo no console do Ubuntu (basta copiar o comando abaixo e clicar como botão direito do mouse no console do Ubuntu e pressionar ENTER) para instalarmos as dependências. Informe a senha do usuário criada anteriormente:

```
sudo apt install -y curl nano git gconf-service libasound2 libatk1.0-0 libc6  
libcairo2 libcups2 libdbus-1-3 libexpat1 libfontconfig1 libgcc1 libgconf-2-4  
libgdk-pixbuf2.0-0 libglib2.0-0 libgtk-3-0 libnspr4 libpango-1.0-0  
libpangocairo-1.0-0 libstdc++6 libx11-6 libx11-xcb1 libxcb1 libxcomposite1  
libxcursor1 libxdamage1 libxext6 libxfixes3 libxi6 libxrandr2 libxrender1  
libxss1 libxtst6 ca-certificates fonts-liberation libappindicator1 libnss3  
lsb-release xdg-utils wget build-essential apt-transport-https libgbm-dev
```

```
This message is shown once a day. To disable it please create the  
/home/danilo/.hushlogin file.  
danilo@DESKTOP-JL5AQJB:~$ sudo apt install -y curl nano git gconf-service libasound2 libatk1.0-0 libc6 libcairo2 libcups2 libdbus-1-3 libexpat1 libfontconfig1 libgcc1  
libgconf-2-4 libgdk-pixbuf2.0-0 libglib2.0-0 libgtk-3-0 libnspr4 libpango-1.0-0 libpangocairo-1.0-0 libstdc++6 libx11-6 libx11-xcb1 libxcb1 libxcomposite1 libxcursor1  
libxdamage1 libxext6 libxfixes3 libxi6 libxrandr2 libxrender1 libxss1 libxtst6 ca-certificates fonts-liberation libappindicator1 libnss3 lsb-release xdg-utils wget  
build-essential apt-transport-https libgbm-dev  
[sudo] password for danilo:  
Sorry, try again.  
[sudo] password for danilo:  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
Note, selecting 'libgcc-s1' instead of 'libgcc1'  
Note, selecting 'apt' instead of 'apt-transport-https'  
Unable to locate package gconf-service  
Unable to locate package libgconf-2-4  
Unable to locate package libappindicator1  
danilo@DESKTOP-JL5AQJB:~$
```

Para instalarmos o ChromeDrive execute os comandos abaixo:

```
wget https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb
```



```

daniel@DESKTOP-JL5AQJB: ~$
Setting up libhttp-message-perl (6.22-1) ...
Setting up libhtml-form-perl (6.07-1) ...
Setting up libfile-mimeinfo-perl (0.29-1) ...
Setting up libhttp-negotiate-perl (6.01-1) ...
Setting up libhttp-cookies-perl (6.08-1) ...
Setting up libhtml-tree-perl (5.07-2) ...
Setting up libpango-1.0-0:amd64 (1.44.7-2ubuntu4) ...
Setting up libhtml-format-perl (2.12-1) ...
Setting up libnet-smtp-ssl-perl (1.04-1) ...
Setting up libmailtools-perl (2.21-1) ...
Setting up libgdk-pixbuf2.0-0:amd64 (2.48.0+dfsg-3ubuntu0.2) ...
Setting up libcups2:amd64 (2.3.1-9ubuntu1) ...
Setting up libhttp-daemon-perl (6.06-1) ...
Setting up libgdk-pixbuf2.0-bin (2.48.0+dfsg-3ubuntu0.2) ...
Setting up cpp (4:9.3.0-1ubuntu2) ...
Setting up libpangot2-1.0-0:amd64 (1.44.7-2ubuntu4) ...
Setting up libpangocairo-1.0-0:amd64 (1.44.7-2ubuntu4) ...
Setting up gtk-update-icon-cache (3.24.20-0ubuntu1) ...
Setting up x11-xserver-utils (7.7+8) ...
Setting up libsvg2-2:amd64 (2.48.9-1ubuntu0.20.04.1) ...
Setting up librsvg2-common:amd64 (2.48.9-1ubuntu0.20.04.1) ...
Setting up adwaita-icon-theme (3.36.1-2ubuntu0.20.04.2) ...
update-alternatives: using /usr/share/icons/Adwaita/cursor.theme to provide /usr/share/icons/default/index.theme (x-cursor-theme) in auto mode
Setting up liblup-protocol-https-perl (6.07-2ubuntu2) ...
Setting up libwww-perl (6.43-1) ...
Setting up humanity-icon-theme (0.6.15) ...
Setting up libxsl-parser-perl (2.46-1) ...
Setting up ubuntu-mono (19.04-0ubuntu3) ...
Setting up libxsl-twig-perl (1:3.58-2) ...
Setting up libnet-dbus-perl (1.2.0-1) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for mime-support (3.64ubuntu1) ...
Processing triggers for libgl1.0-0:amd64 (2.64.0-3-ubuntu20.04.3) ...
Setting up libgtk-3-0:amd64 (3.24.20-0ubuntu1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.2) ...
Setting up libgtk-3-bin (3.24.20-0ubuntu1) ...
Setting up google-chrome-stable (91.0.4472.114-1) ...
update-alternatives: using /usr/bin/google-chrome-stable to provide /usr/bin/x-www-browser (x-www-browser) in auto mode
update-alternatives: using /usr/bin/google-chrome-stable to provide /usr/bin/gnome-www-browser (gnome-www-browser) in auto mode
update-alternatives: using /usr/bin/google-chrome-stable to provide /usr/bin/google-chrome (google-chrome) in auto mode
Processing triggers for libgdk-pixbuf2.0-0:amd64 (2.48.0+dfsg-3ubuntu0.2) ...
daniel@DESKTOP-JL5AQJB: ~$

```

Após instalarmos o ChromeDrive vamos instalar o NodeJS executando as instruções abaixo:

Executar o comando:

```
cd ~
```

```

daniel@DESKTOP-JL5AQJB: ~$ cd ~
daniel@DESKTOP-JL5AQJB: ~$

```

Executar o comando:

```
curl -sL https://deb.nodesource.com/setup\_16.x -o nodesource_setup.sh
```

```

daniel@DESKTOP-JL5AQJB: ~$ curl -sL https://deb.nodesource.com/setup_16.x -o nodesource_setup.sh
daniel@DESKTOP-JL5AQJB: ~$

```

Executar o comando:

```
sudo bash nodesource_setup.sh
```

```

daniel@DESKTOP-JL5AQJB: ~$ sudo bash nodesource_setup.sh

## Installing the NodeSource Node.js 16.x repo...

## Populating apt-get cache...

+ apt-get update
Get:1 http://dl.google.com/linux/chrome/deb stable InRelease [1811 B]
Get:2 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:3 http://dl.google.com/linux/chrome/deb stable/main amd64 Packages [1100 B]
Get:4 http://security.ubuntu.com/ubuntu focal-security/main amd64 Packages [741 kB]
Get:5 http://security.ubuntu.com/ubuntu focal-security/main Translation-en [146 kB]
0% [Connecting to archive.ubuntu.com] [5 Translation-en 20.3 kB/146 kB 14%]

```

```
danilo@DESKTOP-JL5AQJB:~$  
Get:38 http://archive.ubuntu.com/ubuntu focal-backports/restricted amd64 c-n-f Metadata [116 B]  
Get:39 http://archive.ubuntu.com/ubuntu focal-backports/universe amd64 Packages [5788 B]  
Get:40 http://archive.ubuntu.com/ubuntu focal-backports/universe Translation-en [2888 B]  
Get:41 http://archive.ubuntu.com/ubuntu focal-backports/universe amd64 c-n-f Metadata [208 B]  
Get:42 http://archive.ubuntu.com/ubuntu focal-backports/multiverse amd64 c-n-f Metadata [116 B]  
Fetched 19.3 MB in 1min 5s (290 kB/s)  
Reading package lists... Done  
  
## Confirming "focal" is supported...  
  
$ curl -sIf -o /dev/null 'https://deb.nodesource.com/node_16.x/dists/focal/InRelease'  
  
## Adding the NodeSource signing key to your keyring...  
  
$ curl -s https://deb.nodesource.com/gpgkey/nodesource.gpg.key | gpg --dearmor | tee /usr/share/keyrings/nodesource.gpg >/dev/null  
  
## Creating apt sources list file for the NodeSource Node.js 16.x repo...  
  
$ echo "deb [signed-by=/usr/share/keyrings/nodesource.gpg] https://deb.nodesource.com/node_16.x focal main" > /etc/apt/sources.list.d/nodesource.list  
$ echo "deb-src [signed-by=/usr/share/keyrings/nodesource.gpg] https://deb.nodesource.com/node_16.x focal main" >> /etc/apt/sources.list.d/nodesource.list  
  
## Running 'apt-get update' for you...  
  
$ apt-get update  
Hit:1 http://dl.google.com/linux/chrome/deb stable InRelease  
Hit:2 http://security.ubuntu.com/ubuntu focal-security InRelease  
Hit:3 http://archive.ubuntu.com/ubuntu focal InRelease  
Hit:4 http://archive.ubuntu.com/ubuntu focal-updates InRelease  
Get:5 https://deb.nodesource.com/node_16.x focal InRelease [4583 B]  
Hit:6 http://archive.ubuntu.com/ubuntu focal-backports InRelease  
Get:7 https://deb.nodesource.com/node_16.x focal/main amd64 Packages [780 B]  
Fetched 5352 B in 4s (1104 B/s)  
Reading package lists... Done  
  
## Run "sudo apt-get install -y nodejs" to install Node.js 16.x and npm  
## You may also need development tools to build native addons:  
$ sudo apt-get install gcc g++ make  
  
## To install the Yarn package manager, run:  
$ curl -sL https://dl.yarnpkg.com/debian/pubkey.gpg | gpg --dearmor | sudo tee /usr/share/keyrings/yarnkey.gpg >/dev/null  
$ echo "deb [signed-by=/usr/share/keyrings/yarnkey.gpg] https://dl.yarnpkg.com/debian stable main" | sudo tee /etc/apt/sources.list.d/yarn.list  
$ sudo apt-get update && sudo apt-get install yarn
```

Executar o comando:

`sudo apt -y install nodejs`

```
danilo@DESKTOP-JL5AQJB:~$  
$ echo "deb [signed-by=/usr/share/keyrings/nodesource.gpg] https://deb.nodesource.com/node_16.x focal main" > /etc/apt/sources.list.d/nodesource.list  
$ echo "deb-src [signed-by=/usr/share/keyrings/nodesource.gpg] https://deb.nodesource.com/node_16.x focal main" >> /etc/apt/sources.list.d/nodesource.list  
  
## Running 'apt-get update' for you...  
  
$ apt-get update  
Hit:1 http://dl.google.com/linux/chrome/deb stable InRelease  
Hit:2 http://security.ubuntu.com/ubuntu focal-security InRelease  
Hit:3 http://archive.ubuntu.com/ubuntu focal InRelease  
Hit:4 http://archive.ubuntu.com/ubuntu focal-updates InRelease  
Get:5 https://deb.nodesource.com/node_16.x focal InRelease [4583 B]  
Hit:6 http://archive.ubuntu.com/ubuntu focal-backports InRelease  
Get:7 https://deb.nodesource.com/node_16.x focal/main amd64 Packages [780 B]  
Fetched 5352 B in 4s (1104 B/s)  
Reading package lists... Done  
  
## Run "sudo apt-get install -y nodejs" to install Node.js 16.x and npm  
## You may also need development tools to build native addons:  
$ sudo apt-get install gcc g++ make  
  
## To install the Yarn package manager, run:  
$ curl -sL https://dl.yarnpkg.com/debian/pubkey.gpg | gpg --dearmor | sudo tee /usr/share/keyrings/yarnkey.gpg >/dev/null  
$ echo "deb [signed-by=/usr/share/keyrings/yarnkey.gpg] https://dl.yarnpkg.com/debian stable main" | sudo tee /etc/apt/sources.list.d/yarn.list  
$ sudo apt-get update && sudo apt-get install yarn  
  
danilo@DESKTOP-JL5AQJB:~$ sudo apt -y install nodejs  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following NEW packages will be installed:  
  nodejs  
0 upgraded, 1 newly installed, 0 to remove and 17 not upgraded.  
Need to get 16.1 MB of archives.  
After this operation, 118 MB of additional disk space will be used.  
Get:1 https://deb.nodesource.com/node_16.x focal/main amd64 nodejs amd64 16.4.1-1deb-1nodesource [16.1 MB]  
11% [1 nodejs 16.1 MB/26.1 MB 71%]  
Fetched 16.1 MB in 1s (16.1 MB/s)  
debconf: delaying package configuration, since apt-utils is not installed  
Selecting previously unselected package nodejs.  
(Reading database ... 123456 files and directories currently installed.)  
Preparing to unpack .../nodejs_16.4.1-1deb-1nodesource_amd64.deb ...  
Unpacking nodejs (16.4.1-1deb-1nodesource) ...  
Setting up nodejs (16.4.1-1deb-1nodesource) ...  
nodejs service is enabled.
```

Para verificar a versão do node instalado basta executar o comando abaixo:

`node --version`

```
danilo@DESKTOP-JL5AQJB:~$ node --version  
v16.4.1  
danilo@DESKTOP-JL5AQJB:~$
```

Após a instalação do Node vamos iniciar a instalação do MyZAP 2.0 executando o comando abaixo:

`git clone https://github.com/billbarsch/myzap2.0`

Informe as credenciais de acesso ao repositório (email e senha).

```
danilo@DESKTOP-JL5AQJB:~$ git clone https://github.com/billbarsch/myzap2.0
Cloning into 'myzap2.0'...
Username for 'https://github.com': daniloruizdasilva@hotmail.com
Password for 'https://daniloruizdasilva@hotmail.com@github.com':
remote: Enumerating objects: 339, done.
remote: Counting objects: 100% (339/339), done.
remote: Compressing objects: 100% (229/229), done.
remote: Total 339 (delta 163), reused 193 (delta 83), pack-reused 0
Receiving objects: 100% (339/339), 398.74 KiB | 620.00 KiB/s, done.
Resolving deltas: 100% (163/163), done.
danilo@DESKTOP-JL5AQJB:~$
```

Entrar no diretório “myzap2.0” executando o comando abaixo:

```
cd myzap2.0
```

```
danilo@DESKTOP-JL5AQJB:~$ cd myzap2.0/
danilo@DESKTOP-JL5AQJB:~/myzap2.0$
```

Em seguida executar o comando:

```
npm install --allow-root --unsafe-perm=true
```

```
danilo@DESKTOP-JL5AQJB:~/myzap2.0$ ls -l
total 81632
-rw-r--r-- 1 danilo danilo 83565736 Jun 16 23:09 google-chrome-stable_current_amd64.deb
drwxr-xr-x 1 danilo danilo 512 Jul 3 13:15 myzap2.0
-rw-r--r-- 1 danilo danilo 13764 Jul 3 13:04 nodesource_setup.sh
danilo@DESKTOP-JL5AQJB:~$ cd myzap2.0/
danilo@DESKTOP-JL5AQJB:~/myzap2.0$ npm install --allow-root --unsafe-perm=true
[.....] \ reify:debug: http fetch GET 200 https://registry.npmjs.org/debug/-/debug-4.3.1.tgz 3312ms (cache miss)
```

```
danilo@DESKTOP-JL5AQJB:~/myzap2.0$ ls -l
total 81632
-rw-r--r-- 1 danilo danilo 83565736 Jun 16 23:09 google-chrome-stable_current_amd64.deb
drwxr-xr-x 1 danilo danilo 512 Jul 3 13:15 myzap2.0
-rw-r--r-- 1 danilo danilo 13764 Jul 3 13:04 nodesource_setup.sh
danilo@DESKTOP-JL5AQJB:~$ cd myzap2.0/
danilo@DESKTOP-JL5AQJB:~/myzap2.0$ npm install --allow-root --unsafe-perm=true
npm WARN deprecated request-promise@4.2.6: request-promise has been deprecated because it extends the now deprecated request/issues/3142
npm WARN deprecated har-validator@5.1.5: this library is no longer supported
npm WARN deprecated request@2.88.2: request has been deprecated, see https://github.com/request/request/issues/3142
added 585 packages, and audited 586 packages in 1m
64 packages are looking for funding
  run `npm fund` for details
2 vulnerabilities (1 moderate, 1 high)
To address all issues, run:
  npm audit fix
Run `npm audit` for details.
npm notice
npm notice New minor version of npm available! 7.10.1 -> 7.19.1
npm notice Changelog: https://github.com/npm/cli/releases/tag/v7.19.1
npm notice Run `npm install -g npm@7.19.1` to update!
danilo@DESKTOP-JL5AQJB:~/myzap2.0$
```

Crie um arquivo .env em branco executando o comando abaixo:

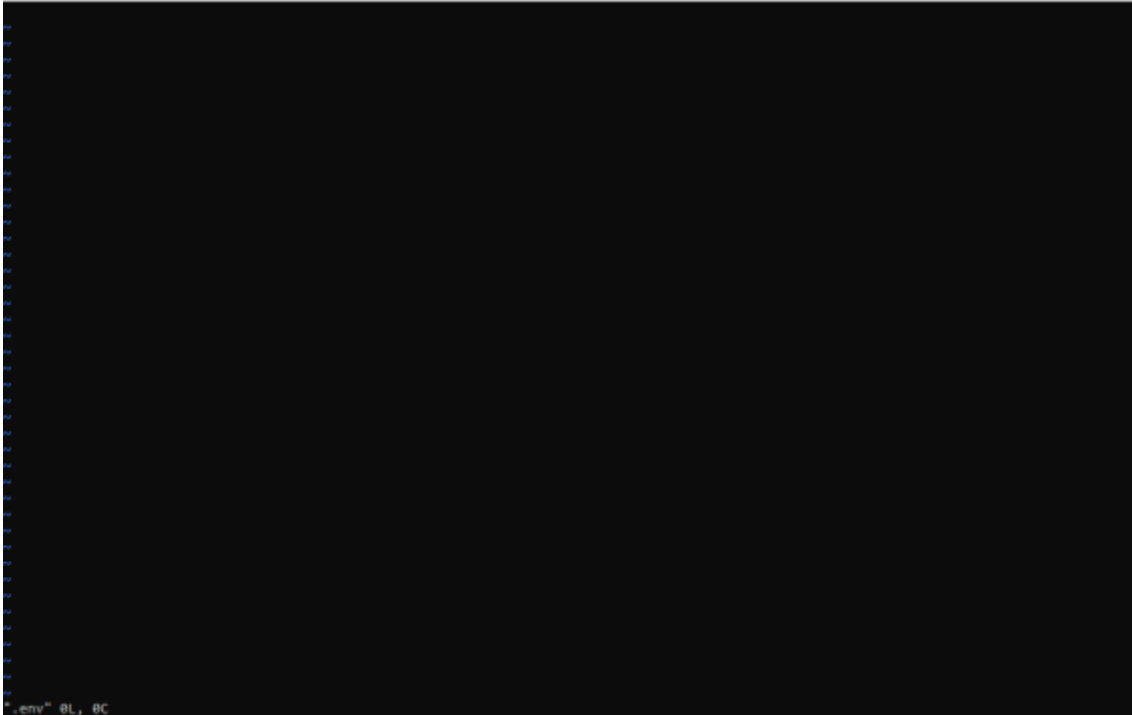
```
touch .env
```

```
danilo@DESKTOP-JL5AQJB: ~/myzap2.0
danilo@DESKTOP-JL5AQJB:~/myzap2.0$ touch .env
danilo@DESKTOP-JL5AQJB:~/myzap2.0$
```

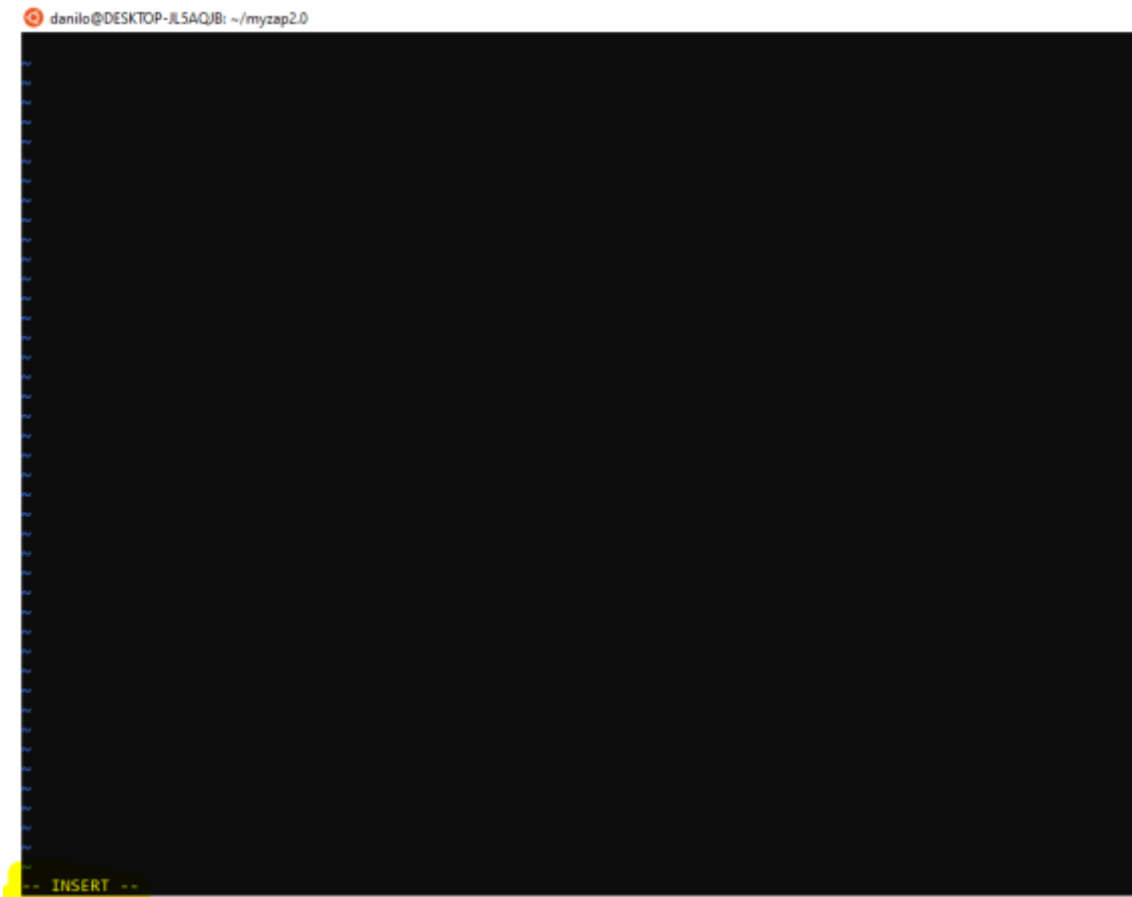
Edite o arquivo “.env” com o comando:

`vi .env`

```
danilo@DESKTOP-JL5AQJB: ~/myzap2.0
```



Aperte a tecla “INSERT” do teclado para entrar no modo edição e editarmos o arquivo “.env”:



Copie o texto abaixo, mas antes defina um valor para o parâmetro “TOKEN” que desejar. No caso do exemplo defini o valor “TOKENMYZAP” que será utilizado nas requisições.

Também atualize os parâmetros “API_KEY”, “AUTH_DOMAIN”, “PROJECT_ID”, “STORAGE_BUCKET”, “MESSAGING_SENDER_ID” e “APP_ID” de acordo com as credenciais suas de acesso ao Firebase descritos no capítulo 05.

```
<script>
  // Your web app's Firebase configuration
  var firebaseConfig = {
    apiKey: "AIzaSyAHu4RoD4X9MHu8YpXUxaqYZf8XrYF42SY",
    authDomain: "myzap-2cbef.firebaseio.com",
    projectId: "myzap-2cbef",
    storageBucket: "myzap-2cbef.appspot.com",
    messagingSenderId: "938471440273",
    appId: "1:938471440273:web:856c5aa4982e0d96a62430"
  };
  // Initialize Firebase
  firebase.initializeApp(firebaseConfig);
</script>
```

Neste manual estamos utilizando a engine WPPCONNECT (parâmetro ENGINE = 2) para a execução da API. Caso desejarem podem testar as outras engines também (WhatsApp-Web-JS ou VENOM) que o MyZAP suporta. Maiores informações das engines podem ser encontradas nos links conforme abaixo:

WhatsApp-Web-JS - <https://github.com/pedroslopez/whatsapp-web.js>

WPPCONNECT – <https://wppconnect-team.github.io>

VENOM – <https://github.com/orkestral/venom>

(Após editar os parâmetros no texto abaixo copiar todo o texto para adicionarmos no arquivo .env criado em branco).

```
/*#####  
#####
```

```
# File: .env                                     #  
# Project: MyZap2.0                             #  
# Created Date: 2021-06-21 12:52:13             #  
# Author: Eduardo Policarpo                     #  
# Last Modified: 2021-06-21 12:59:26            #  
# Modified By: Eduardo Policarpo                #
```

```
#####  
#####*/
```

Defina o numero da porta a ser usada pela API.

PORT=**3333**

Defina o HOST aqui caso voce utilize uma VPS deve ser colocado o IP da VPS, ou caso vc tenha,

um nome de dominio ja apontado para sua VPS deve-se colocar o nome de dominio

Exemplos:

HOST=204.202.54.2 => IP da VPS

HOST=api.myzap.com.br => caso dominio apontado para VPS

HOST=localhost => caso esteja usando na sua propria maquina local

HOST=localhost

ENGINE: setar o valor correspondente a ENGINE que deseja utilizar
1 - para WhatsApp-Web-JS / 2 - para WPPCONNECT / 3 - para VENOM
ENGINE=2

Caso utilize certificado SSL defina a variavel HTTPS=1 e informe o diretorio nas variaveis

HTTPS=0

SSL_KEY_PATH=/etc/letsencrypt/live/yourdomain.net/privkey.pem

SSL_CERT_PATH=/etc/letsencrypt/live/yourdomain.net/cert.pem

TOKEN => Aqui voce define uma chave de segurança para Uso da API,

só conseguira conectar e criar sessões informando o codigo definido Aqui

TOKEN=

#firebase database config

Nesta sessão voce deve configurar os seus dados de autenticação da conta firebase

API_KEY=

AUTH_DOMAIN=

PROJECT_ID=

STORAGE_BUCKET=

MESSAGING_SENDER_ID=

APP_ID=

As Sessoes criadas na API são salvas no banco de dados GOOGLE firebase

Caso queira que ao iniciar a API todas as sessões salvas sejam inicializadas automaticamente

START_ALL_SESSIONS=true


```
danilo@DESKTOP-JL5AQJB: ~/myzap2.0
#####

# Defina o número da porta a ser usada pela API.
PORT=1111

# Defina o HOST aqui, caso você utilize uma VPS deve ser colocada o IP da VPS, ou caso vc tenha
# um nome de domínio já apontado para sua VPS deve-se colocar o nome do domínio
# Exemplos:
# HOST=194.282.54.2 => IP da VPS
# HOST=api.myzap.com.br => caso domínio apontado para VPS
# HOST=localhost => caso esteja usando na sua propria maquina local

HOST=localhost

# ENGINE: usar o valor correspondente a ENGINE que deseja utilizar
# 1 - para whatsapp-web-2S / 2 - para WPPCONNECT / 3 - para VENDOR
ENGINE=2

# Caso utilize certificado SSL defina a variavel HTTPS=1 e informe o diretório das variaveis
HTTPS=0
# SSL_KEY_PATH=/etc/letsencrypt/live/yourdomain.net/privatekey.pem
# SSL_CERT_PATH=/etc/letsencrypt/live/yourdomain.net/cert.pem

# TOKEN => Aqui você define uma chave de segurança para uso da API,
# só consegue conectar e criar sessões informando o token definido aqui
TOKEN=MYZAP

# Firebase database config
# Nesta opção você deve configurar os seus dados de autenticação da conta Firebase
API_KEY=AIzaSyAHu4Ro04X9MHu8YpXUxaqVZf8XrYF42SY
AUTH_DOMAIN=myzap-2cbef.firebaseio.com
PROJECT_ID=myzap-2cbef
STORAGE_BUCKET=myzap-2cbef.appspot.com
MESSAGING_SENDER_ID=938471440273
APP_ID=1:1:938471440273:web:856c5aa4982e0d96a62430

# As Sessões criadas na API são salvas no Banco de Dados Google Firebase
# Caso queira que ao iniciar a API todas as sessões salvas sejam inicializadas automaticamente
START_ALL_SESSIONS=true

-- INSERT --
```

Para salvarmos e fecharmos nosso arquivo “.env” pressione a tecla “ESC” e depois digite “:wq!” e pressione ENTER.

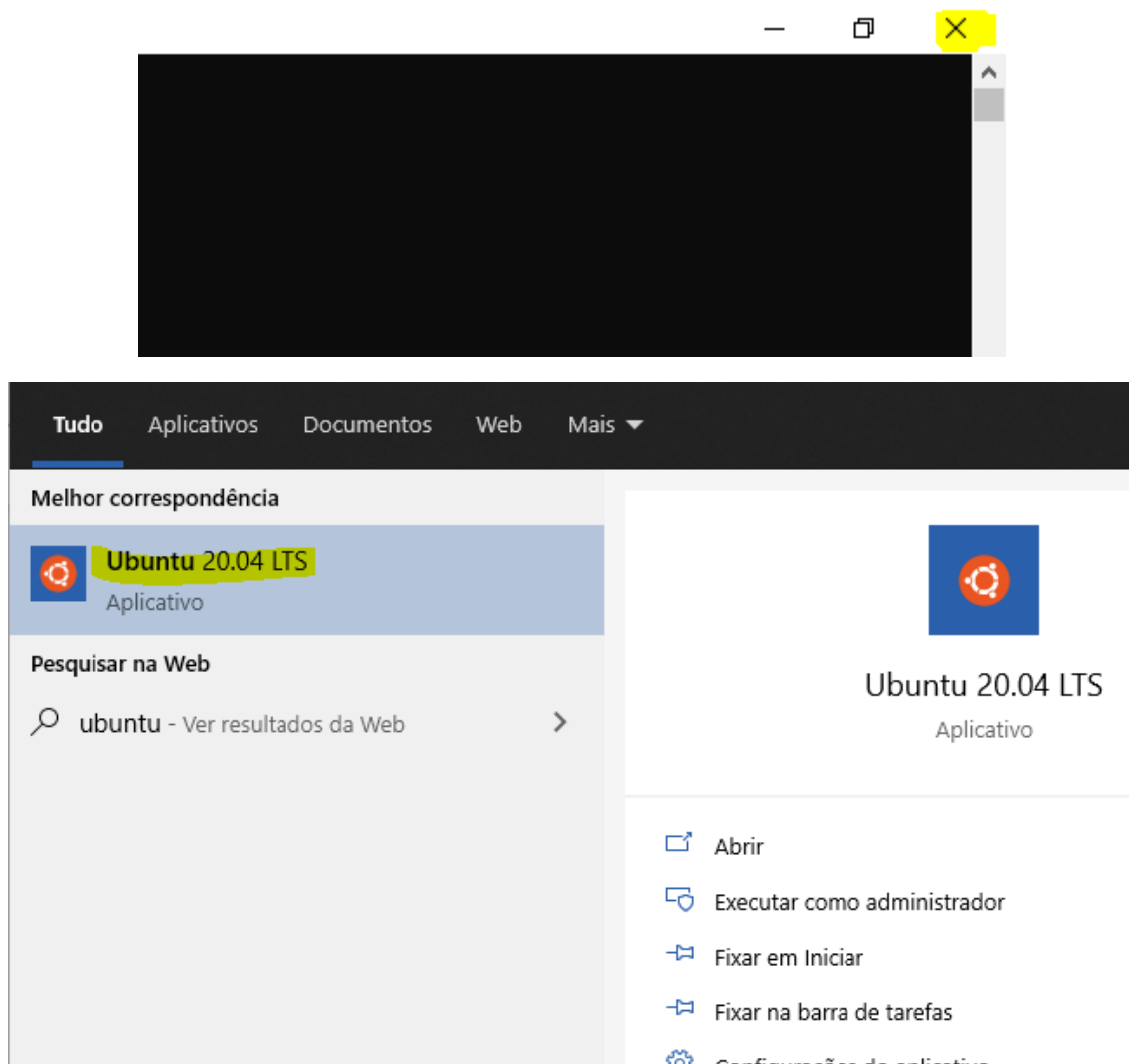
```
MESSAGING_SENDER_ID=938471440273
APP_ID=1:1:938471440273:web:856c5aa4982e0d96a62430

# As Sessões criadas na API são salvas no banco de dados
# Caso queira que ao iniciar a API todas as sessões salvas sejam inicializadas automaticamente
START_ALL_SESSIONS=true

:wq!
```

```
danilo@DESKTOP-JL5AQJB: ~/myzap2.0
danilo@DESKTOP-JL5AQJB:~/myzap2.0$ touch .env
danilo@DESKTOP-JL5AQJB:~/myzap2.0$ vi .env
danilo@DESKTOP-JL5AQJB:~/myzap2.0$
```

Para garantir a leitura correta das variáveis do arquivo .env feche o Ubuntu selecionando a opção “X” e abre ele novamente.



Após abrir novamente o Ubuntu 20.04 LTS execute o comando para entrarmos no diretório do MyZAP 2.0:

```
cd myzap2.0
```

Para iniciarmos o MyZAP 2.0 execute o comando:

```
node index.js
```

[illegible]

Caso abrir um painel para habilitar o acesso a porta 3333 permita em todas as redes.

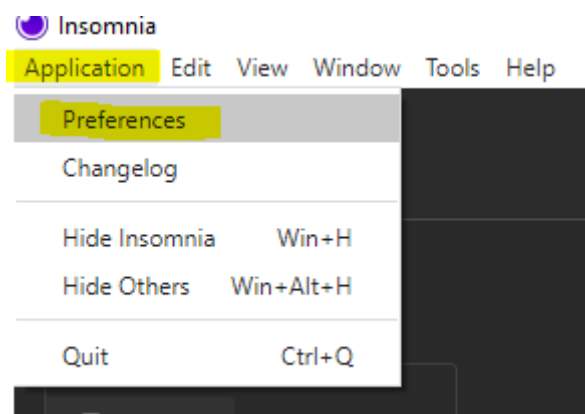
Caso desejar manter os processos ativos a cada reinicialização do servidor execute o comando abaixo informando a senha do usuário criado anteriormente:

```
sudo npm install -y pm2 -g pm2 start index.js pm2 startup
```

7. Testando a API MyZAP 2.0

Para testarmos a API do MyZAP 2.0 vamos abrir o Insomnia e importar a Collection “Insomnia MyZAP VS 2 0 2021-07-03.yaml”.

Selecione a opção “Application” e depois a opção “Preferences”.



Na opção “Data” selecione a opção “Import Data” e depois “From File”:

Insomnia Preferences – v2021.4.0 ⓘ

General

Data

Themes

Keyboard

Account

Plugins

Import format will be automatically detected (Insomnia, Postman v2, HAR, Curl)

Don't see your format here? [Add Your Own](#).

Export Data ▼

Import Data ▼

Create Run Button

* Tip: You can also paste

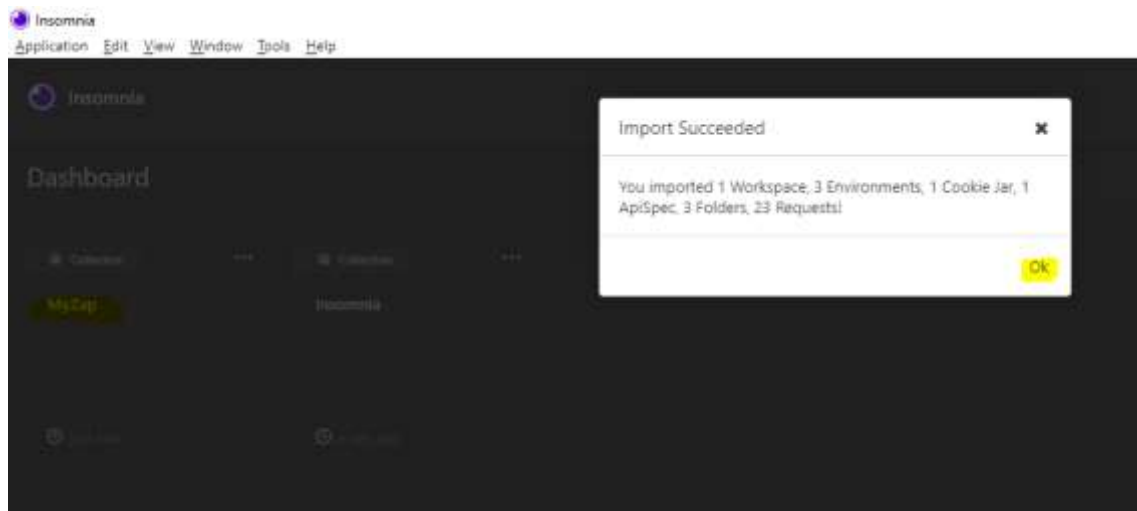
CHOOSE IMPORT TYPE

From File

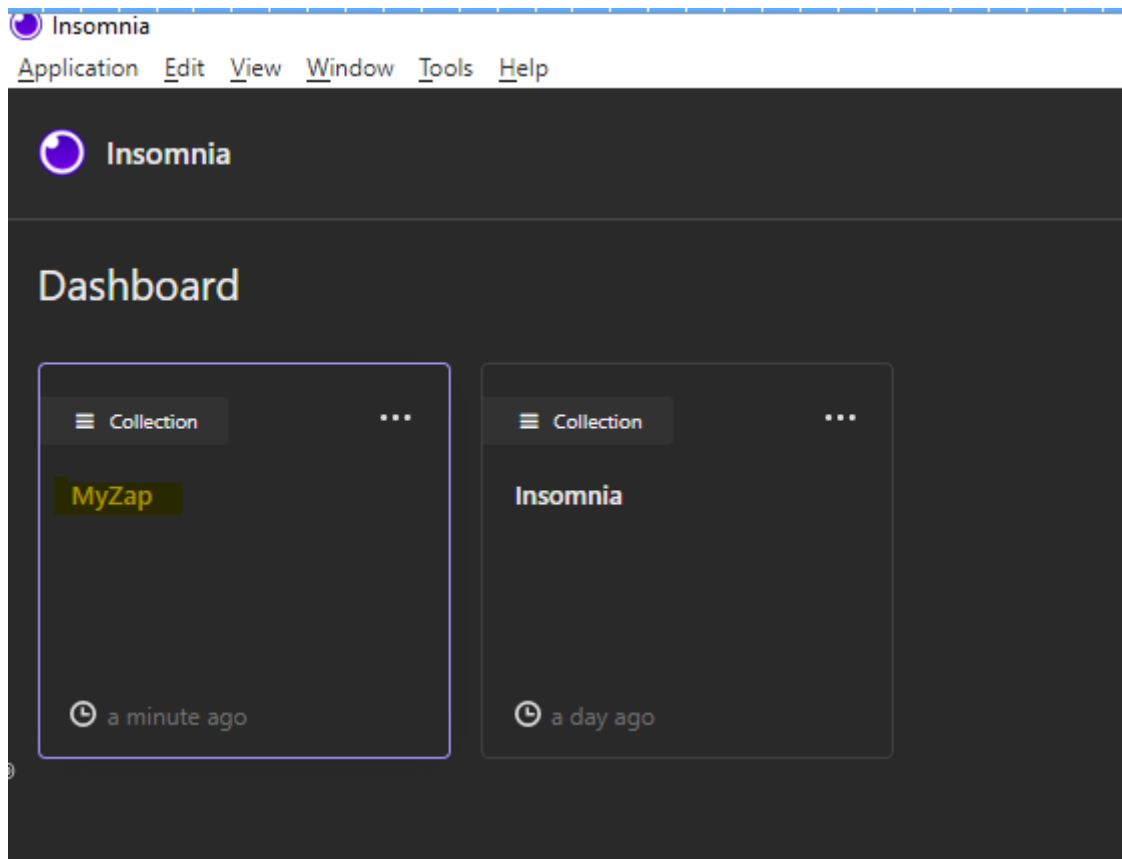
From URL

From Clipboard

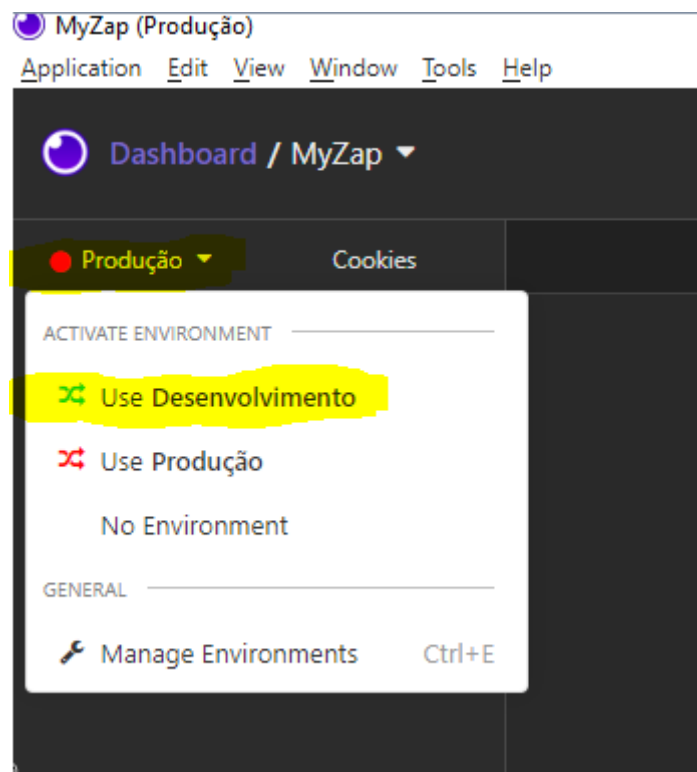
Selecione o arquivo “Insomnia_MyZAP_VS_2_0_2021-07-03.yaml” no seu computador e depois a opção “OK”



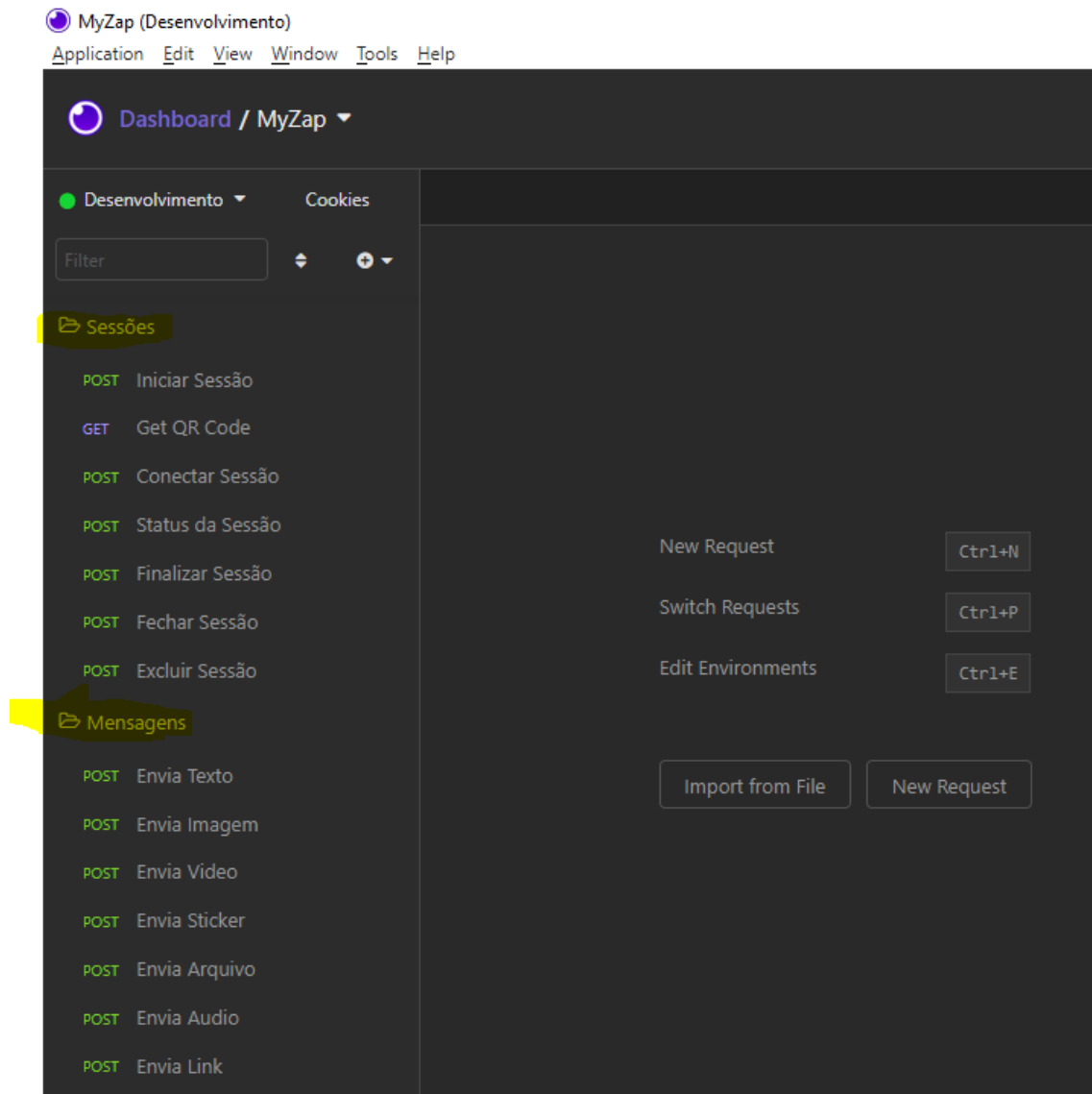
Selecione a Collection “MyZap” no Dashboard:



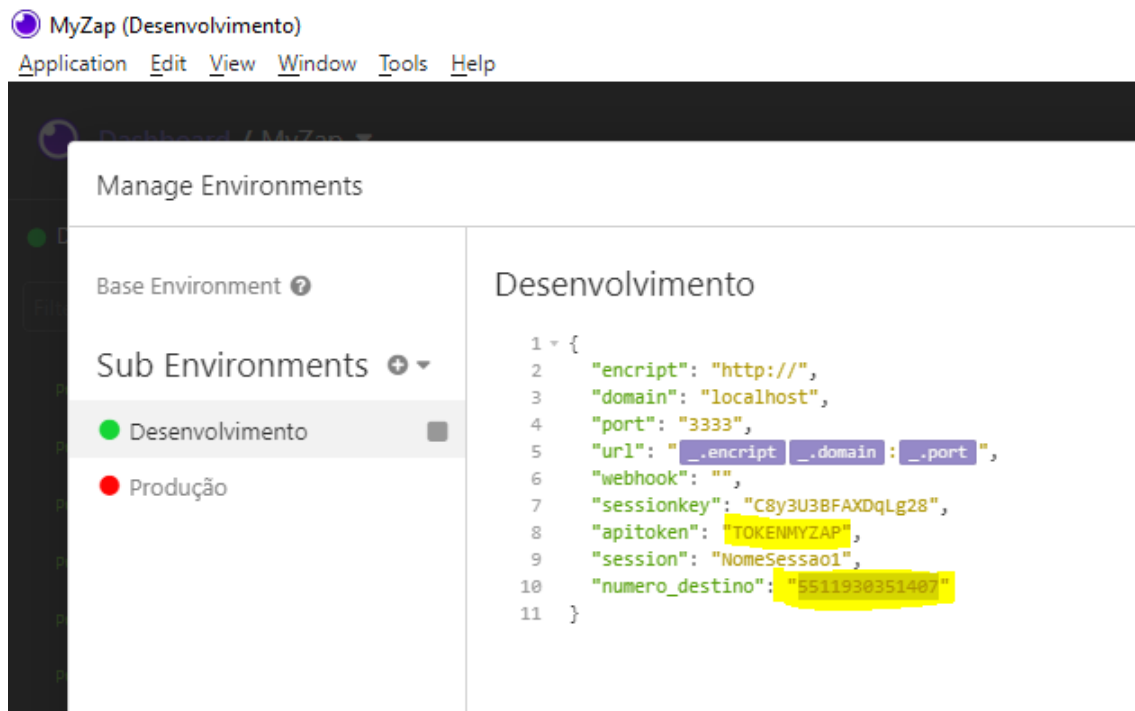
Selecione a combobox “Produção” e selecione a opção “Use Desenvolvimento”. Clique nas pastas “Sessões”.



Selecione as pastas “Sessões”, “Mensagens” e “Gerais” para expandir os itens e visualizar as operações da Collection da API.



Precisamos fazer a alteração na variáveis da Collection informando o número de celular de destino que deseja realizar os testes. Para isso clica na combobox “Desenvolvimento” e depois a opção “Manage Environments”.

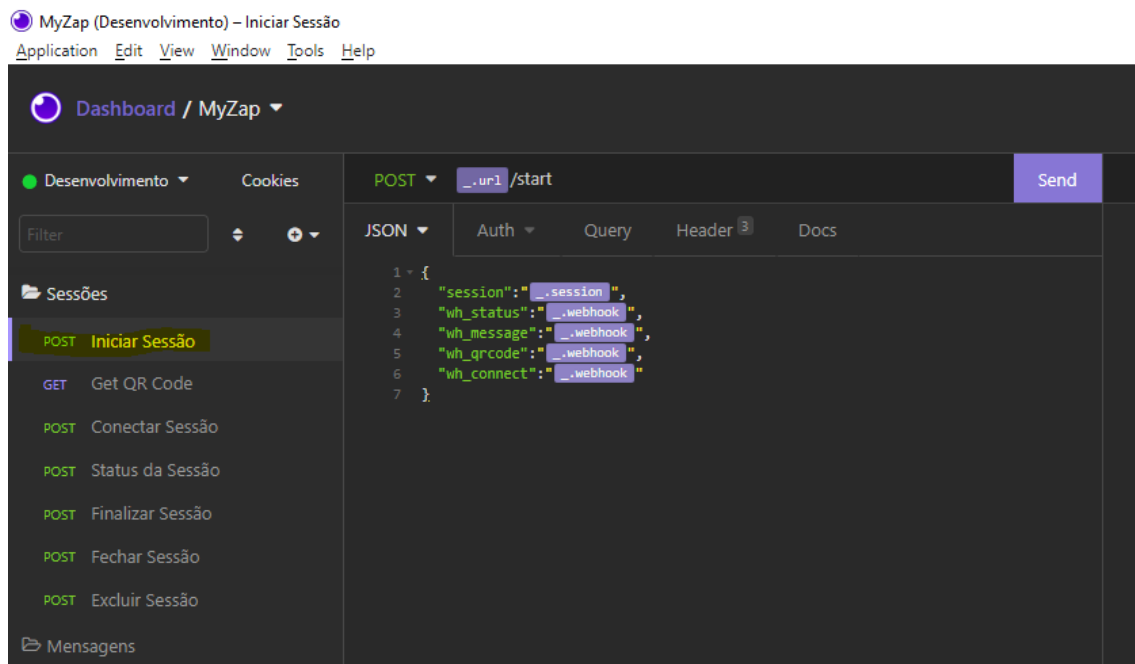


Após informar o celular de destino selecione a opção “Done” para confirmar a alteração.

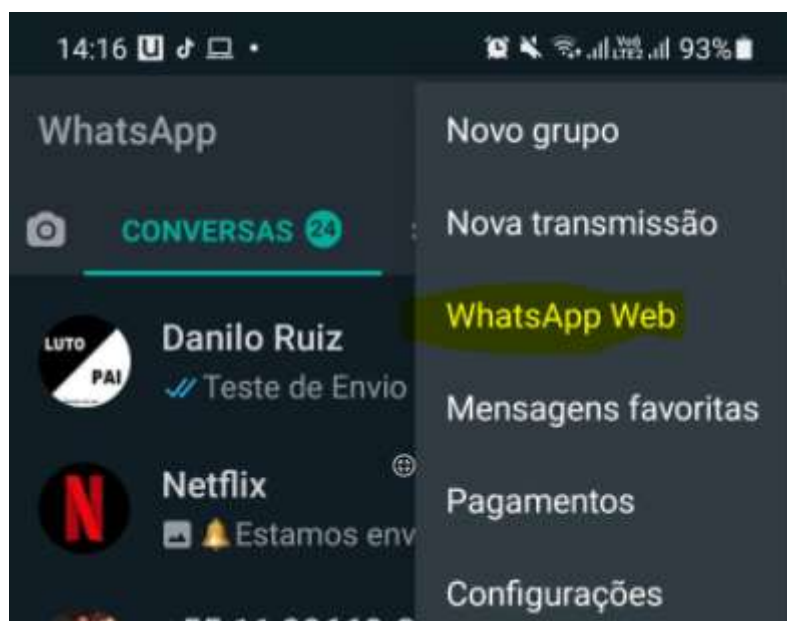
O conteúdo do parâmetro “apitoken” deve ser o mesmo valor configurado no parâmetro “TOKEN” do arquivo “.env” configurado no MyZAP. Caso informar um outro valor diferente de “TOKENMYZAP” no arquivo .env informar o mesmo valor para o parâmetro “apitoken” da Collection.

O conteúdo do parâmetro “domain” está definido como “localhost” por se tratar da instalação em ambiente local, Para quem instalar em um VPS por exemplo deve definir o valor deste parâmetro com o IP do servidor (Ex: 198.50.206.137).

Realizada as configurações vamos executar a operação “Iniciar Sessão” para realizar a leitura do QRCode no terminal do Ubuntu 20.04 LTS.

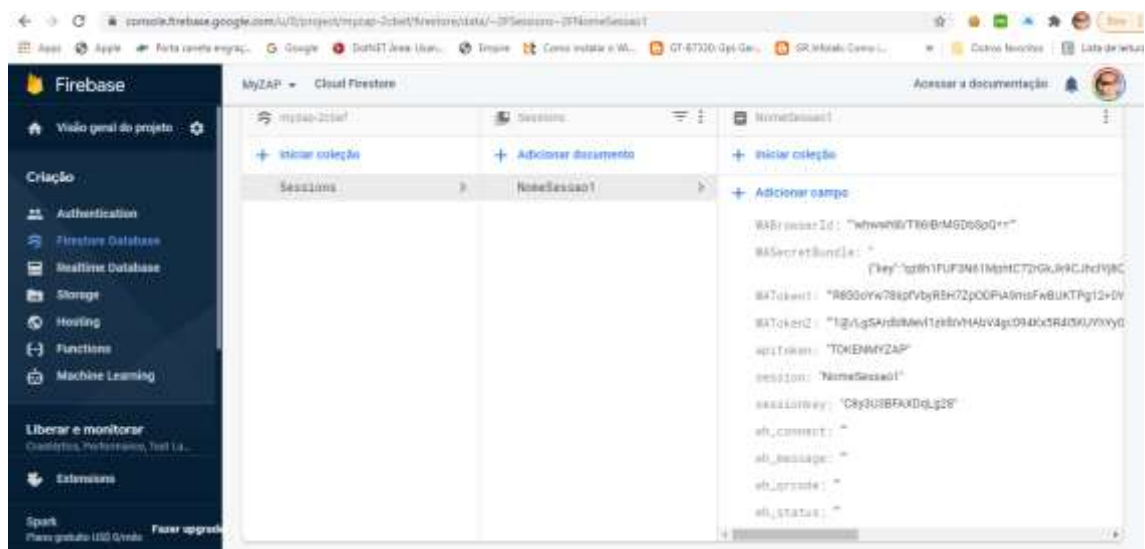


Após clicar na operação “Iniciar Sessão” e depois na opção “Send”, pegar o celular, ir na opção “WhatsApp Web” e depois na opção “Conectar um aparelho” lendo o QRCode do terminal do Ubuntu.

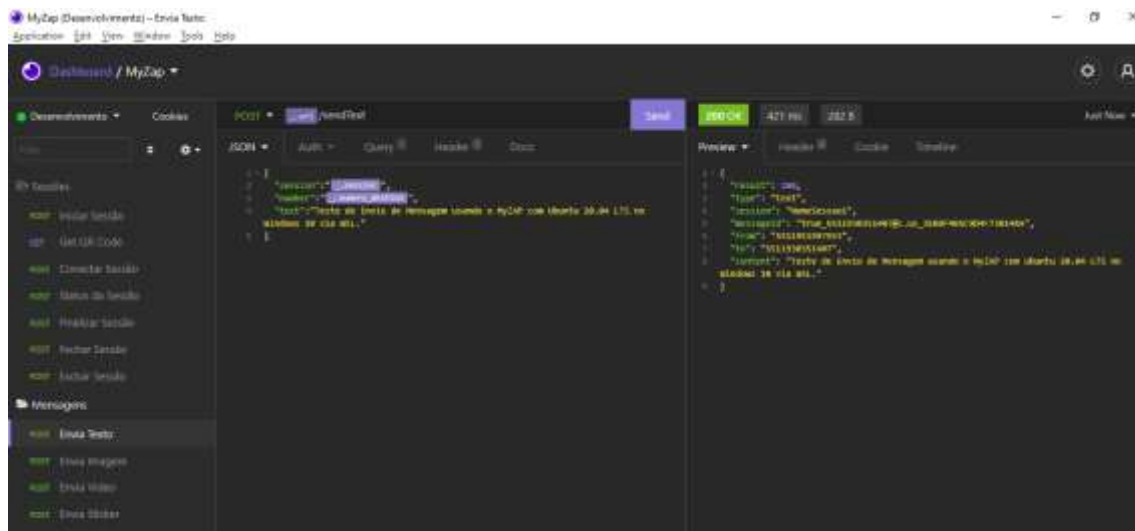
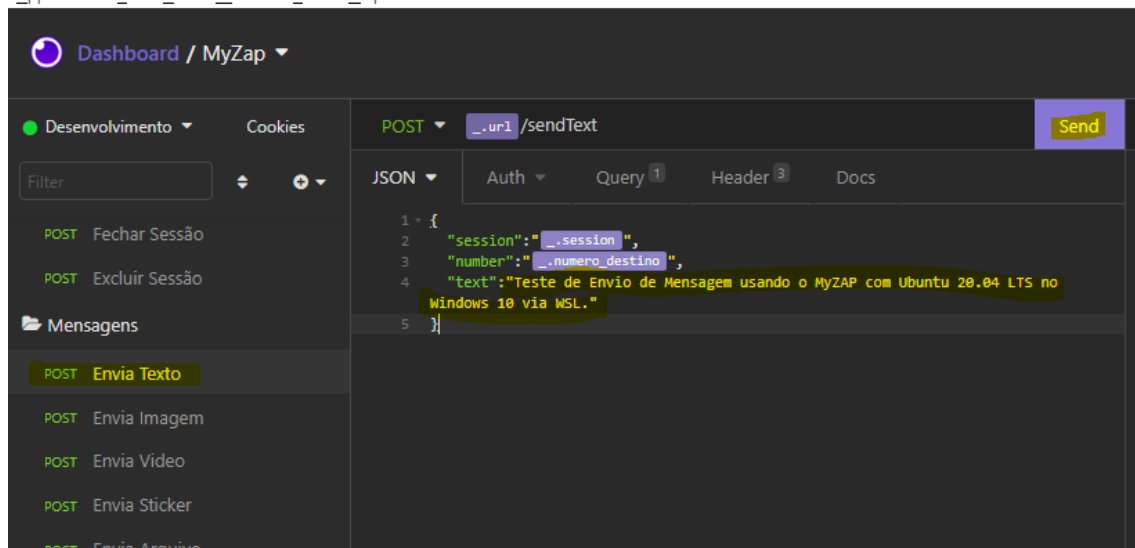


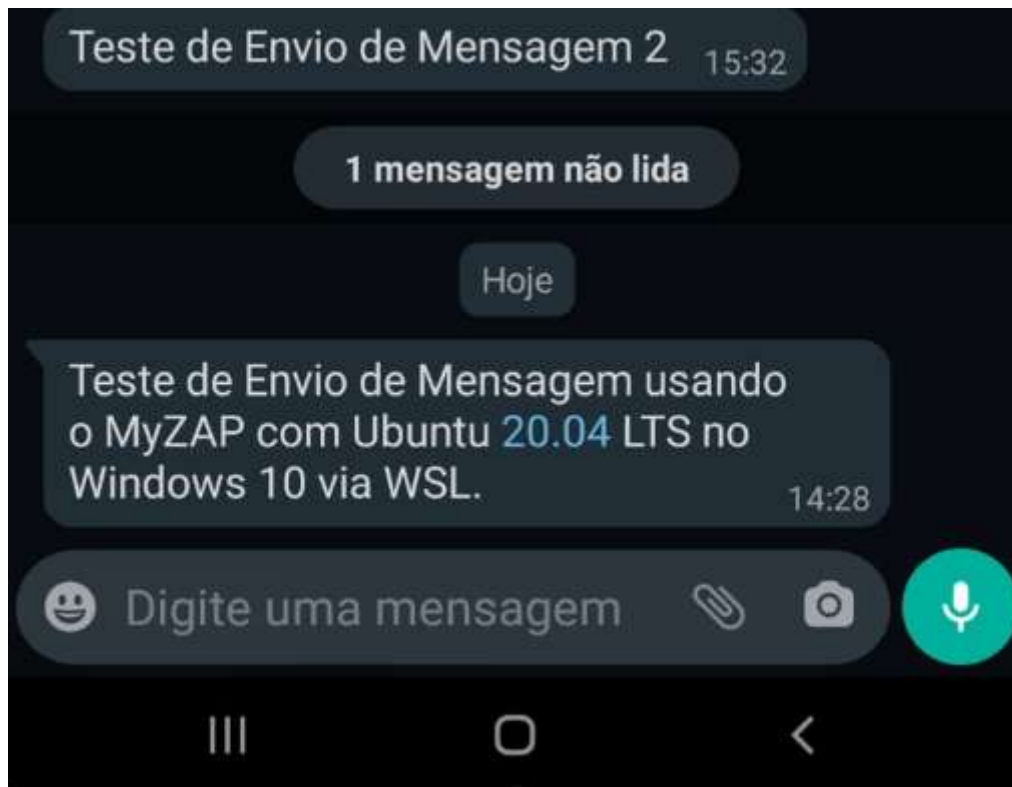


Se irmos no painel do Firebase iremos ver a sessão gravada no nosso banco de dados.

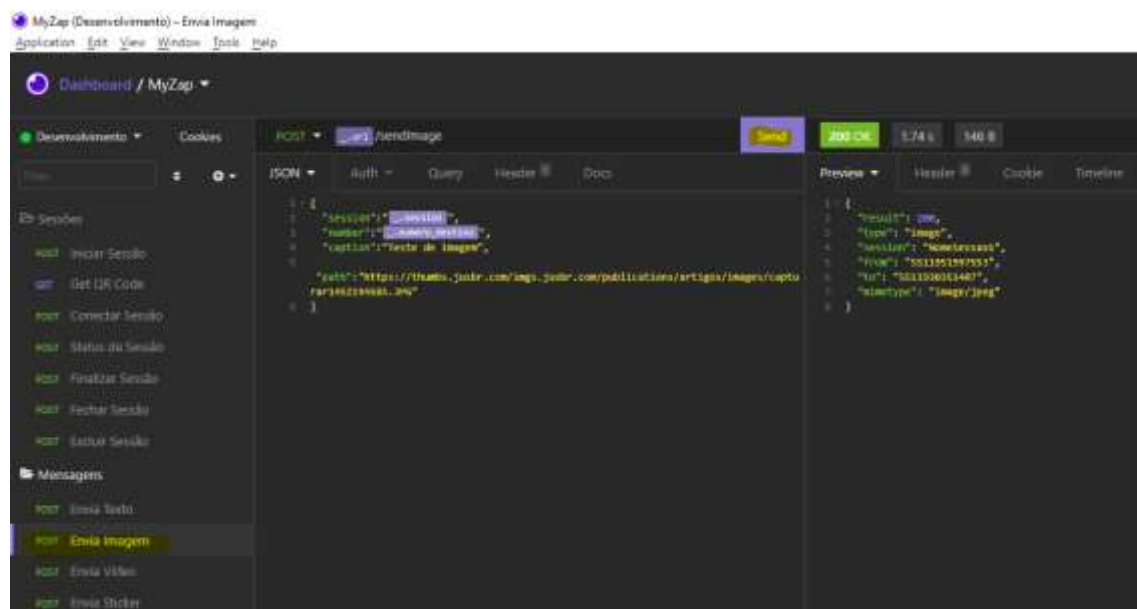


Vamos executar um envio de um texto para o celular de destino. Para isso basta executarmos a operação “Enviar texto” da pasta “Mensagens”:





Vamos executar outro teste enviando uma foto, para isso basta executarmos a operação “Envia imagem”:





8. Créditos

Este material só foi possível graças a iniciativa, ao grande trabalho e contribuições dos colaboradores:

- ✓ Bill Barsch - (63) 99215-8117
- ✓ Eduardo Policarpo (Desenvolvedor Oficial) - (43) 99661-1437
- ✓ Lindolfo Júnior - (61) 9977-1425

e também de todos os outros participantes do grupo.

Link Oficial do Github da versão estável do projeto:

<https://github.com/billbarsch/myzap2.0>