JENKINS INSIDE DOCKER CONTAINER, INTEGRATION WITH GITHUB

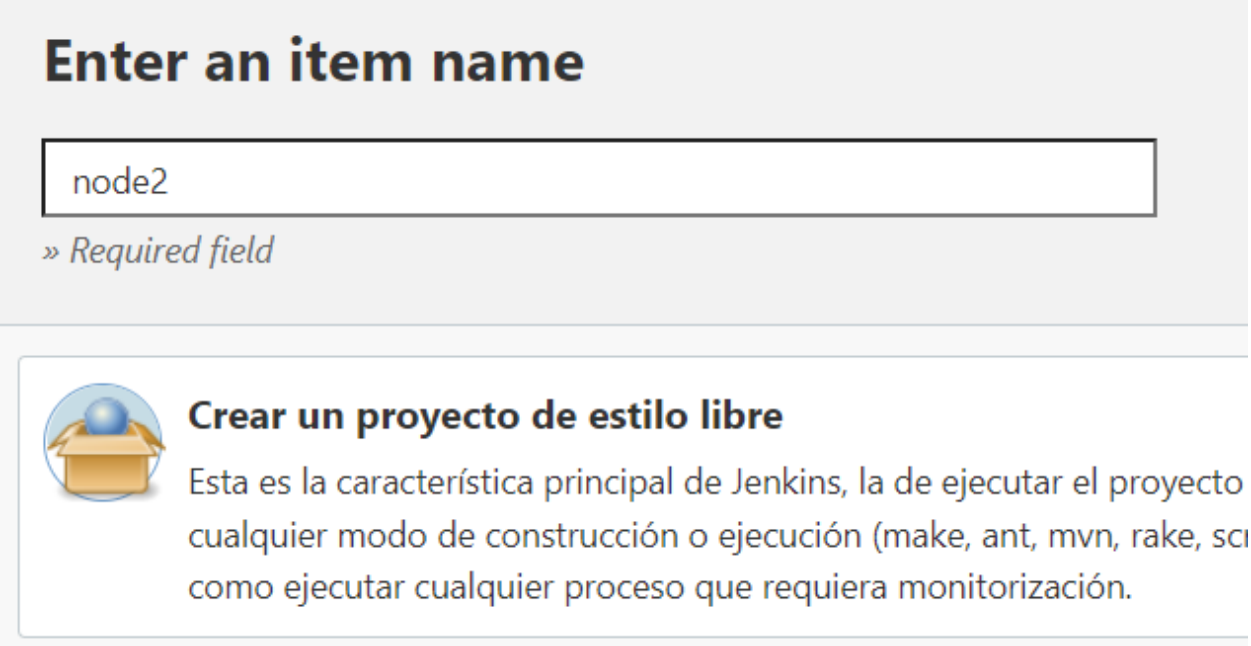
I am using an Ubuntu VM that has docker installed, inside a docker container is a Jenkins image running, in this tutorial I document the steps to integrate Jenkins and github. Jenkins will pull the repository after any commit tu any branch on the remote repository on github and it will execute a .js file with node after the pulling. My Ubuntu VM has the ip 192.168.0.29 and the user gustavo in the sudoers group

On ubuntu

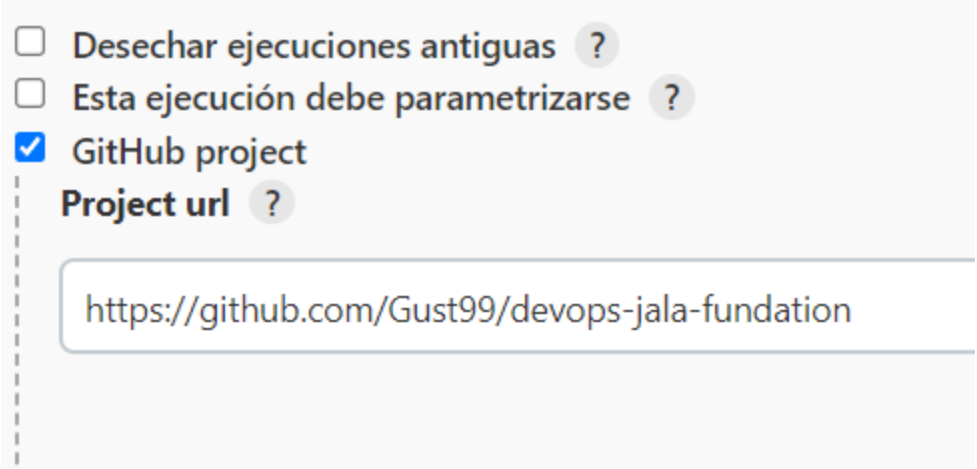
>sudo docker run -p 8080:8080 -p 50000:50000 -v jenkins_home:/var/jenkins_home jenkins/jenkins:lts-jdk11 (Run jenkins image in a container)

Now if I go to 192.168.0.29:8080 I can see the jenkins gui:

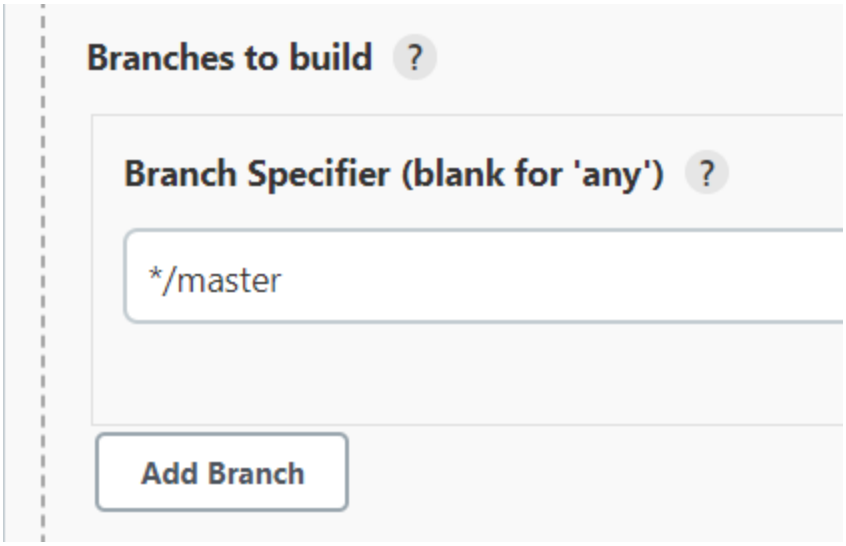
Then I create a Job

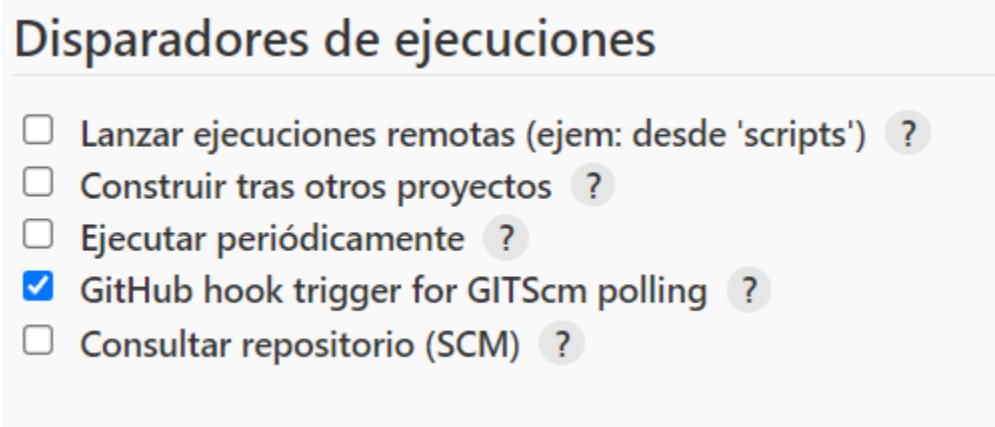


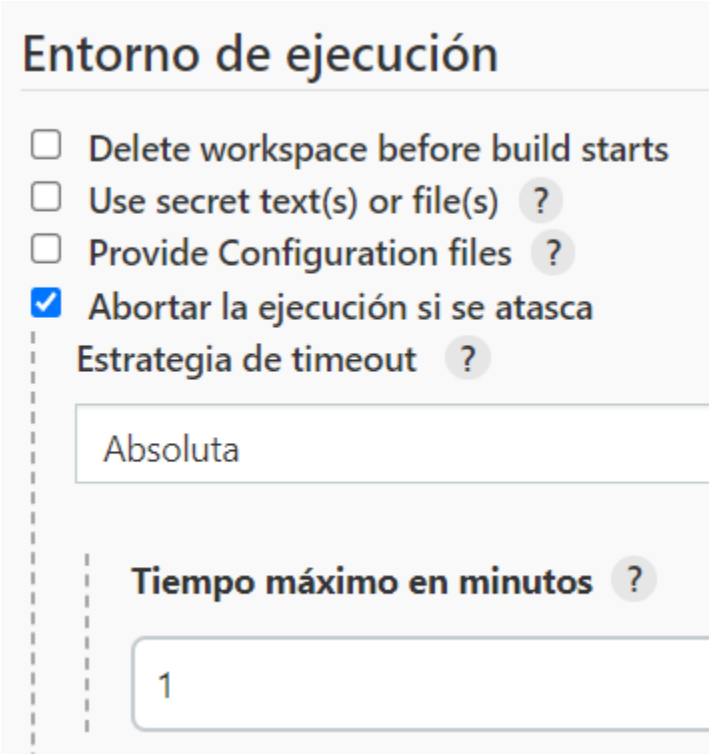
Then I configurate the Job

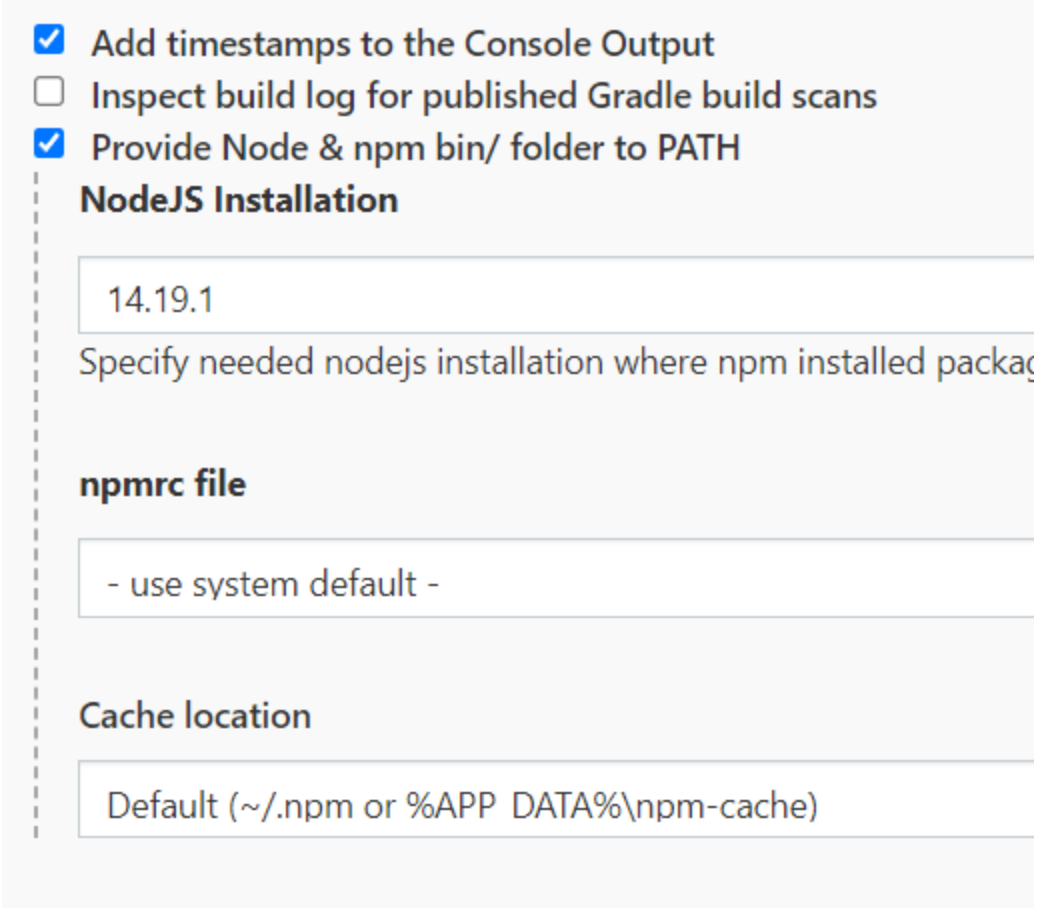


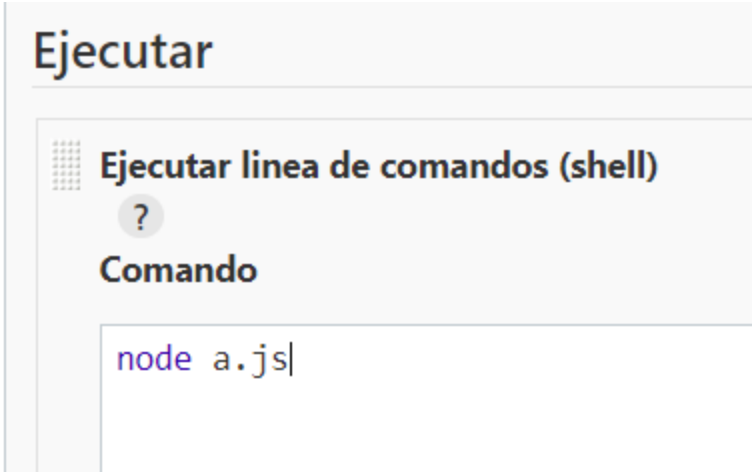

Configurar el origen del código fuente
Ninguno
Git
Repositories
Repository URL
https://github.com/Gust99/devops-jala-fundation.git
Credentials
- none -
Add


Branches to build
Branch Specifier (blank for 'any')
*/master
Add Branch

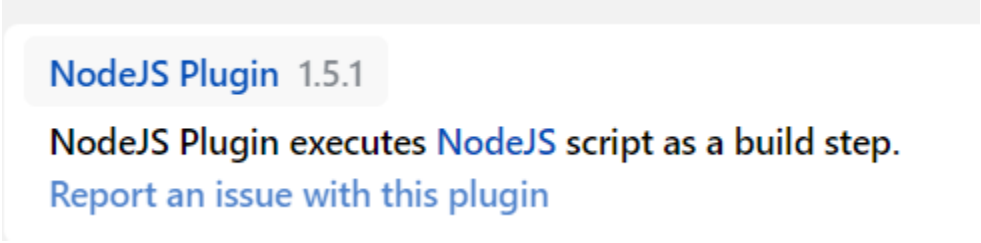

Disparadores de ejecuciones
Lanzar ejecuciones remotas (ejem: desde 'scripts')
Construir tras otros proyectos
Ejecutar periódicamente
GitHub hook trigger for GITScm polling
Consultar repositorio (SCM)


Entorno de ejecución
Delete workspace before build starts
Use secret text(s) or file(s)
Provide Configuration files
Abortar la ejecución si se atasca
Estrategia de timeout
Absoluta
Tiempo máximo en minutos
1

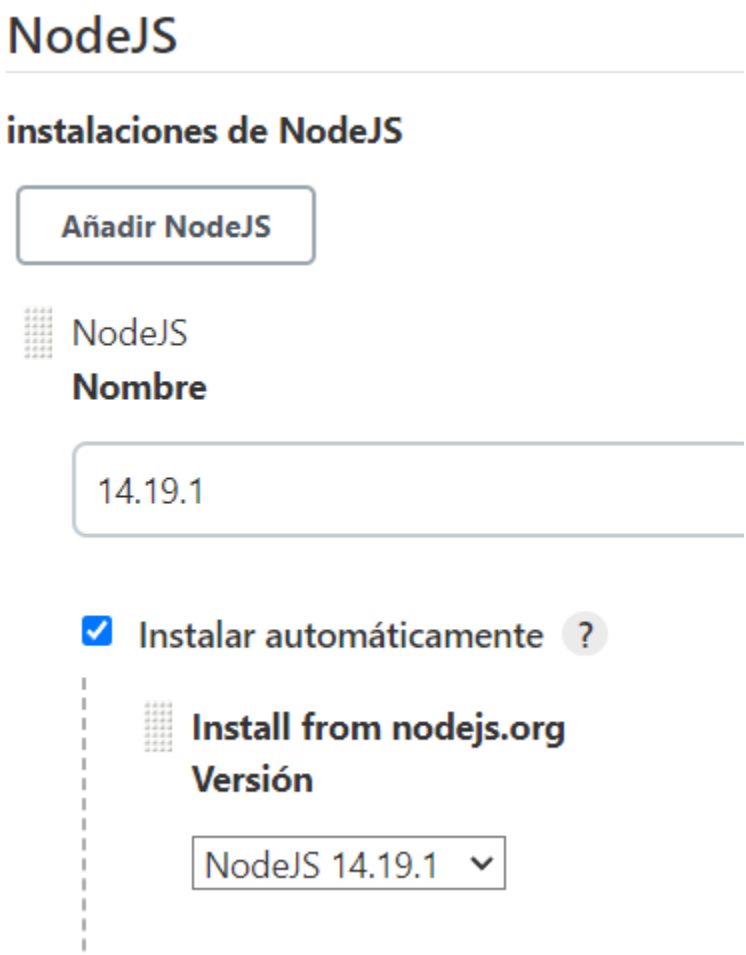




In order to execute .js files with node I installed a node plugin for jenkins



Configurations for the plugin



Use of ngrok

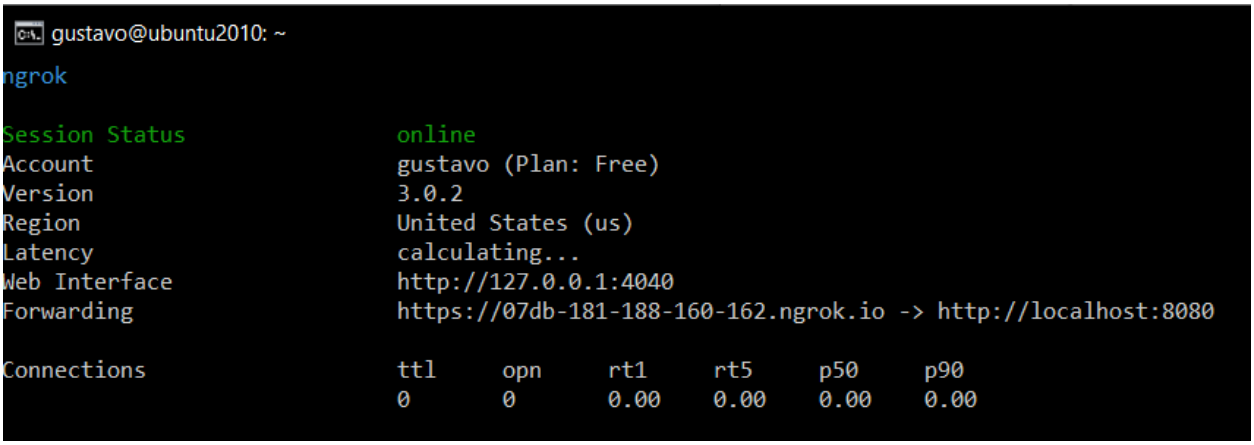
Ngrok will let my Ubuntu VM to be accessible from Internet, I need this to let github communicate with my local jenkins server

>sudo tar xvzf ~/Downloads/ngrok-v3-stable-linux-amd64.tgz -C /usr/local/bin

>curl -s https://ngrok-agent.s3.amazonaws.com/ngrok.asc | sudo tee /etc/apt/trusted.gpg.d/ngrok.asc >/dev/null &&
echo "deb https://ngrok-agent.s3.amazonaws.com buster main" | sudo tee /etc/apt/sources.list.d/ngrok.list &&
sudo apt update && sudo apt install ngrok

>ngrok config add-authtoken <token>

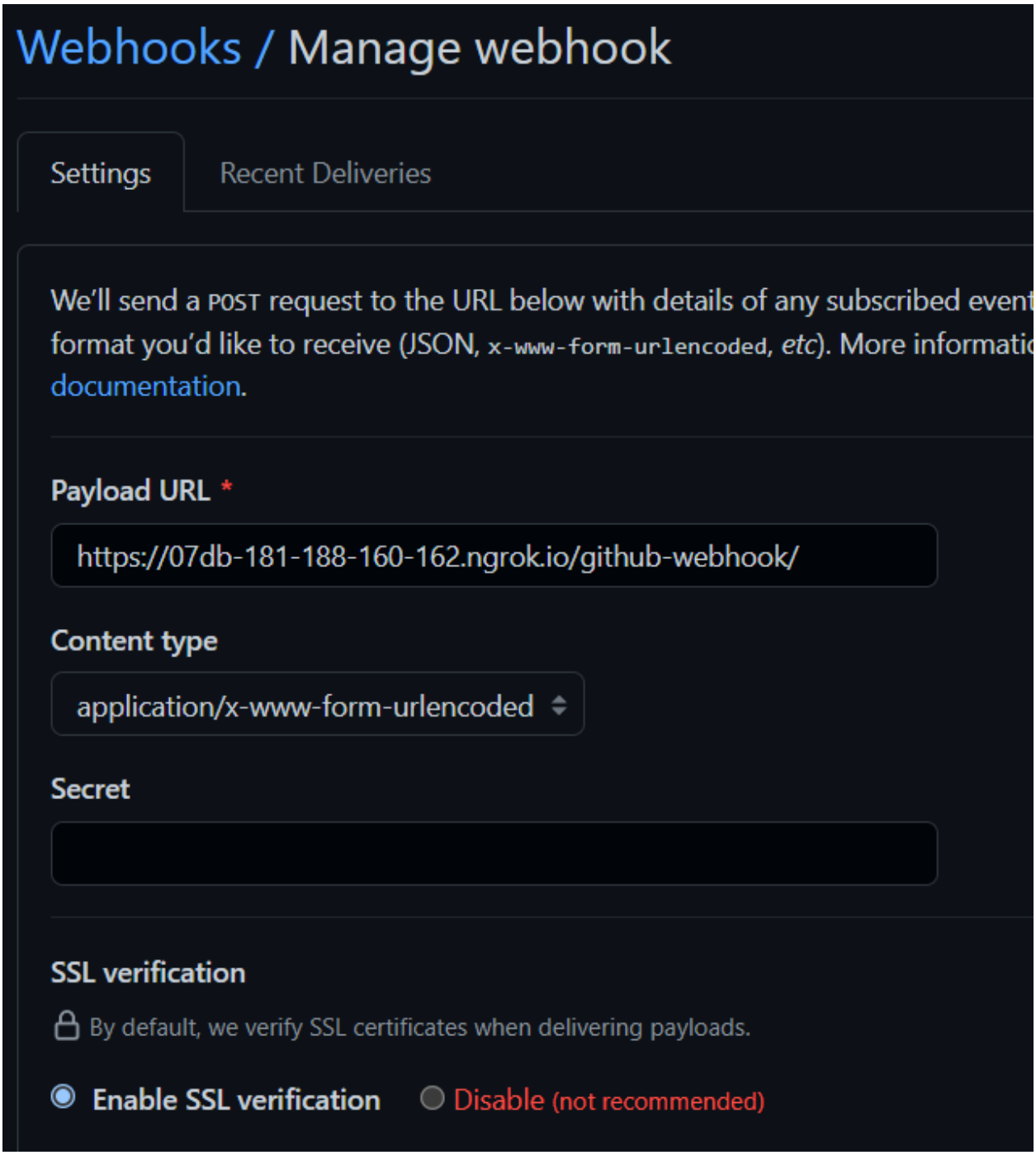
>ngrok http 8080

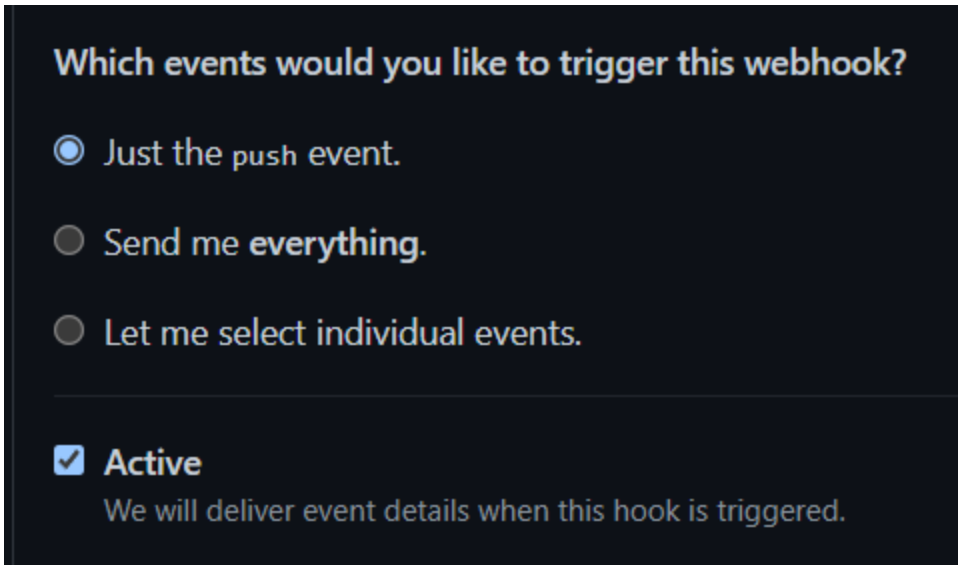


Now the forwarding url represents the url of my local Ubuntu VM, is the url that will let github connect to my jenkins server

Github webhooks

Inside github in settings/webhooks I configurated a webhook, the payload url field has the ngrok url and /github-webhook/


Webhooks / Manage webhook
Settings
Recent Deliveries
We'll send a POST request to the URL below with details of any subscribed event
format you'd like to receive (JSON, x-www-form-urlencoded, etc). More informatic
documentation.
Payload URL *
https://07db-181-188-160-162.ngrok.io/github-webhook/
Content type
application/x-www-form-urlencoded
Secret
SSL verification
By default, we verify SSL certificates when delivering payloads.
Enable SSL verification
Disable (not recommended)

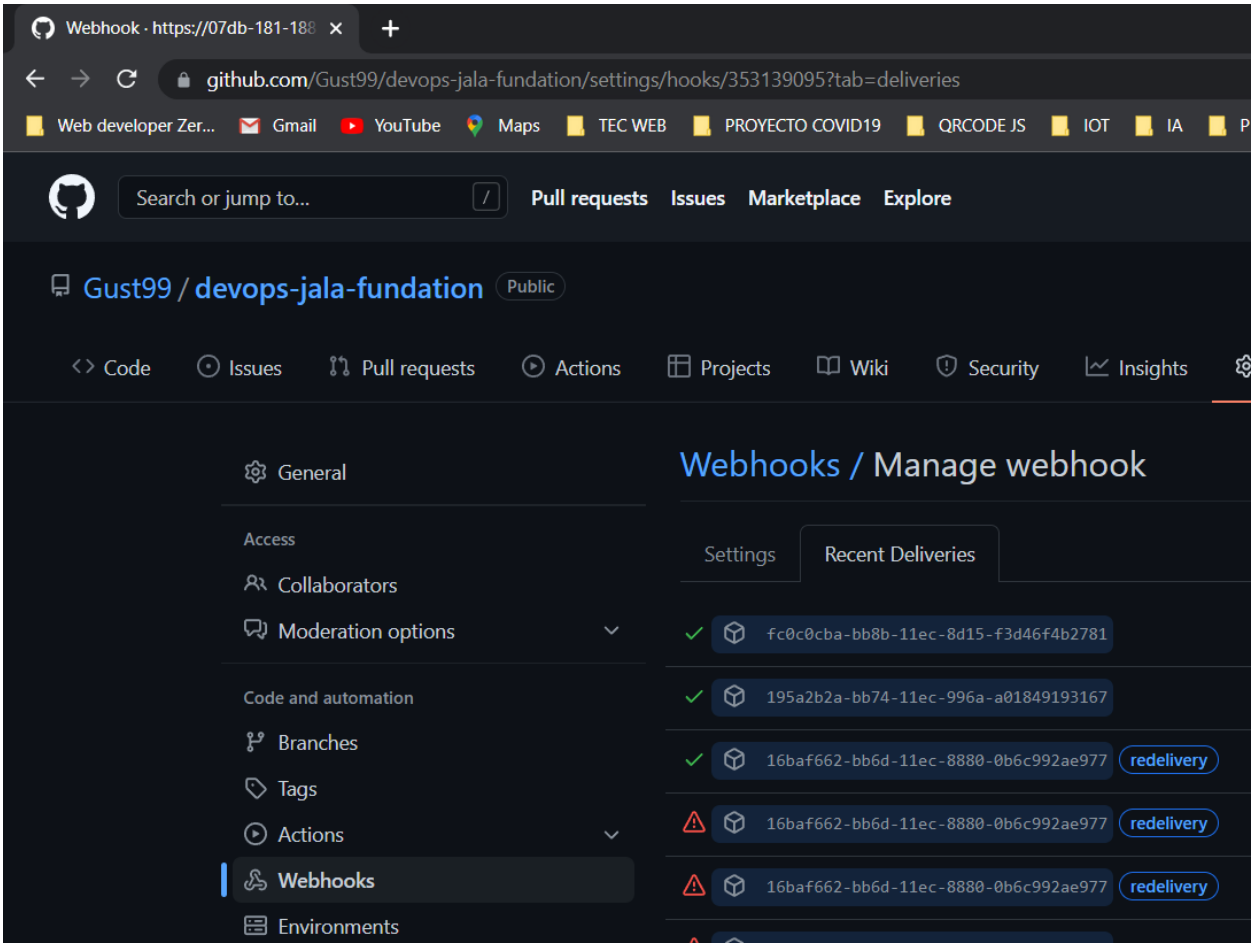


After that if from my windows I push a commit to master branch then github should send the repository to my local jenkins ubuntu server and jenkins should execute the a.js file, this file prints the first 30 fibonacci numbers.

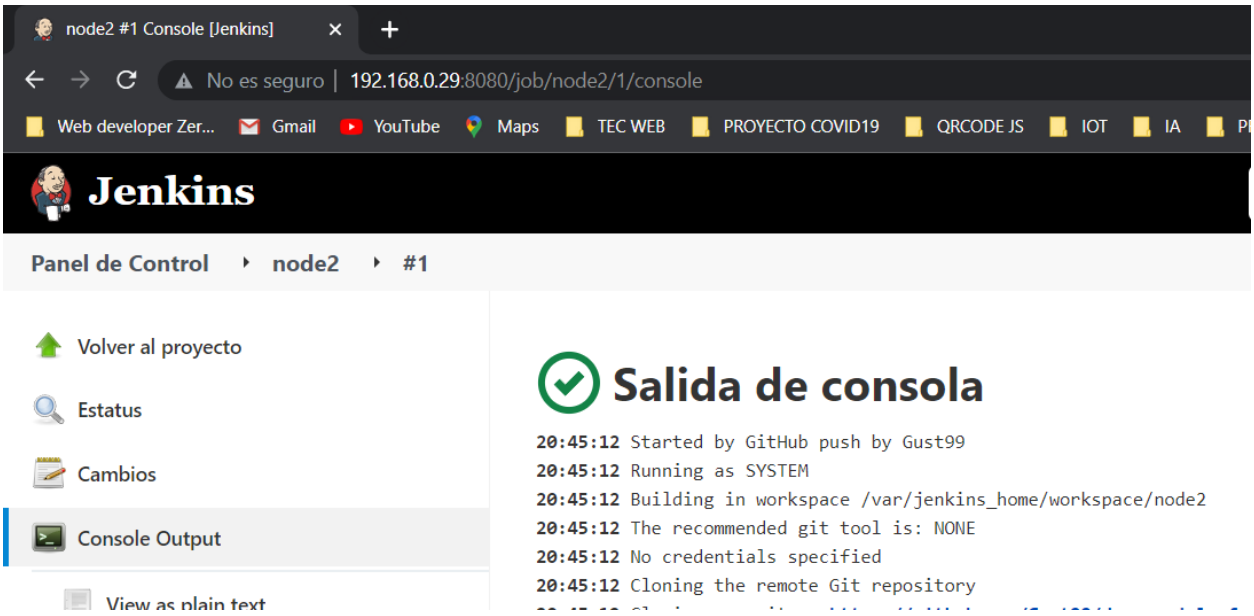
From windows push to github master branch

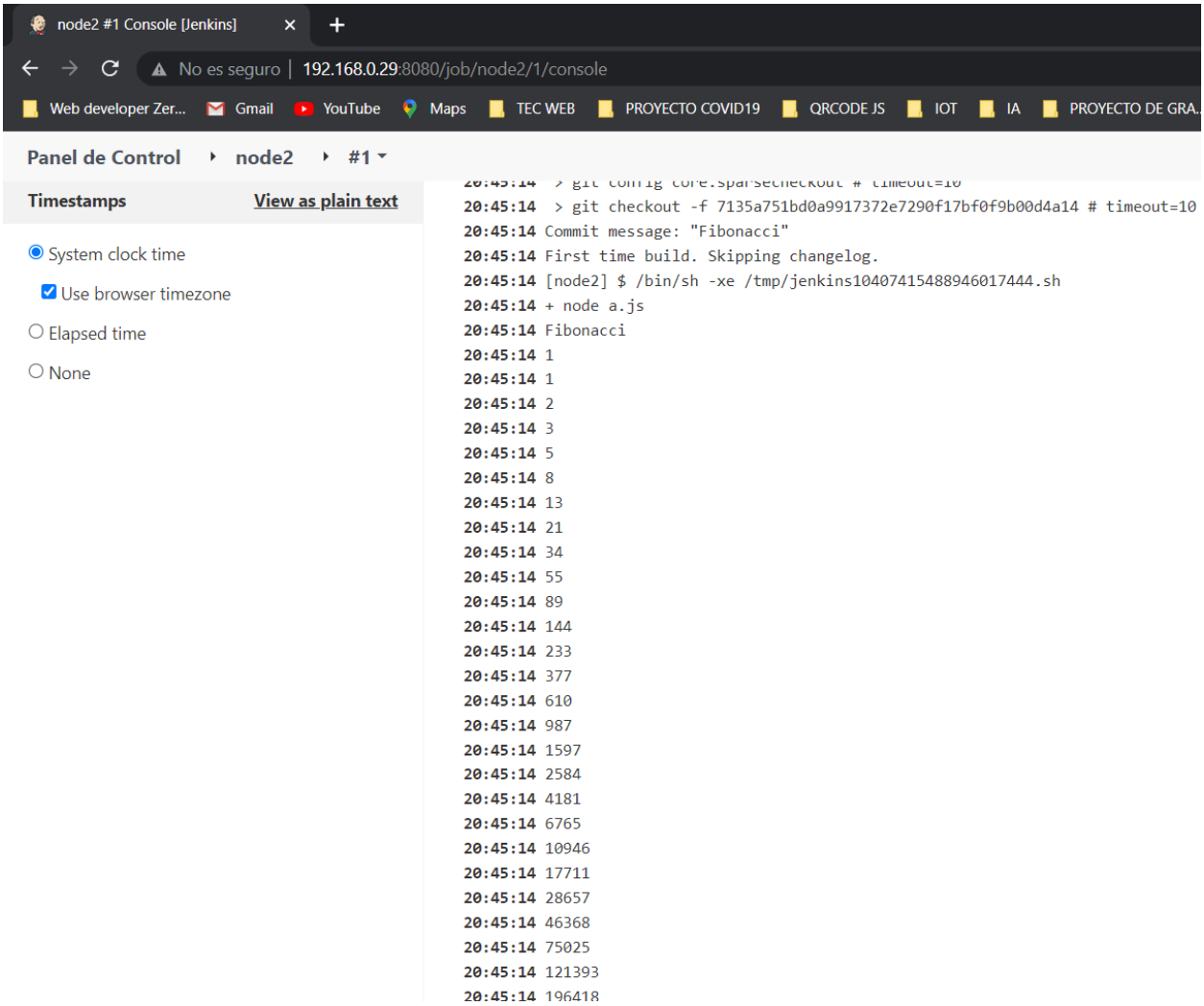
```
MINGW64:/c/Users/Usuario/Desktop/BOOTCAMP/DEVOPS/vagrant
   08835ce..e029b9c  master -> master

Usuario@LAPTOP-902IJ86V MINGW64 ~/Desktop/BOOTCAMP/DEVOPS/vagrant (master)
$ nano a.js

Usuario@LAPTOP-902IJ86V MINGW64 ~/Desktop/BOOTCAMP/DEVOPS/vagrant (master)
$ git add .

Usuario@LAPTOP-902IJ86V MINGW64 ~/Desktop/BOOTCAMP/DEVOPS/vagrant (master)
$ git commit -m "Fibonacci"
[master 7135a75] Fibonacci
 1 file changed, 1 insertion(+), 1 deletion(-)

Usuario@LAPTOP-902IJ86V MINGW64 ~/Desktop/BOOTCAMP/DEVOPS/vagrant (master)
$ git push origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 16 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 288 bytes | 96.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/Gust99/devops-jala-fundation.git
   e029b9c..7135a75  master -> master
```

Github will send the repository to my local jenkins server



Then from jenkins inside the Job I should see the output


node2 #1 Console [Jenkins]
No es seguro | 192.168.0.29:8080/job/node2/1/console
Web developer Zer... Gmail YouTube Maps TEC WEB PROYECTO COVID19 QRCODE JS IOT IA
Jenkins
Panel de Control node2 #1
Volver al proyecto
Estatus
Cambios
Console Output
View as plain text
Salida de consola
20:45:12 Started by GitHub push by Gust99
20:45:12 Running as SYSTEM
20:45:12 Building in workspace /var/jenkins_home/workspace/node2
20:45:12 The recommended git tool is: NONE
20:45:12 No credentials specified
20:45:12 Cloning the remote Git repository


node2 #1 Console [Jenkins]
No es seguro | 192.168.0.29:8080/job/node2/1/console
Web developer Zer... Gmail YouTube Maps TEC WEB PROYECTO COVID19 QRCODE JS IOT IA PROYECTO DE GRA...
Panel de Control node2 #1
Timestamps
View as plain text
System clock time
Use browser timezone
Elapsed time
None
20:45:14 > git checkout -f 7135a751bd0a9917372e7290f17bf0f9b00d4a14 # timeout=10
20:45:14 Commit message: "Fibonacci"
20:45:14 First time build. Skipping changelog.
20:45:14 [node2] $ /bin/sh -xe /tmp/jenkins10407415488946017444.sh
20:45:14 + node a.js
20:45:14 Fibonacci
20:45:14 1
20:45:14 1
20:45:14 2
20:45:14 3
20:45:14 5
20:45:14 8
20:45:14 13
20:45:14 21
20:45:14 34
20:45:14 55
20:45:14 89
20:45:14 144
20:45:14 233
20:45:14 377
20:45:14 610
20:45:14 987
20:45:14 1597
20:45:14 2584
20:45:14 4181
20:45:14 6765
20:45:14 10946
20:45:14 17711
20:45:14 28657
20:45:14 46368
20:45:14 75025
20:45:14 121393
20:45:14 196418

```
20:45:14 17711
20:45:14 28657
20:45:14 46368
20:45:14 75025
20:45:14 121393
20:45:14 196418
20:45:14 317811
20:45:14 514229
20:45:14 832040
20:45:14 Finished: SUCCESS
```

I used the repository in wich the devops classes homework has to be uploaded, in the image below you can see the a.js file that jenkins executes after a push to master branch on github