

Documentación de Coolify

Andrea Sofía Pais Dos Santos

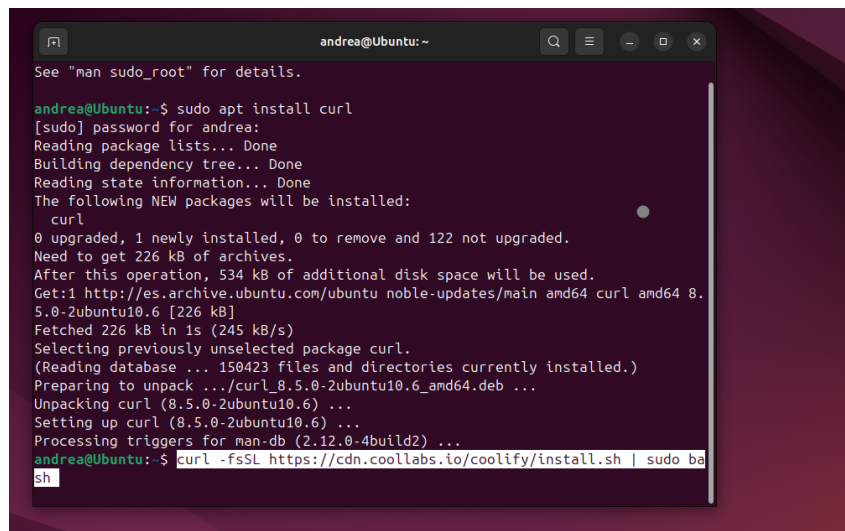
January 22, 2026

1 Índice

2. Instalación Coolify y Ngrok	página 3
3. Creación de un proyecto	página 4
4. Puente inverso	página 7
5. Despliegue continuo	página 7

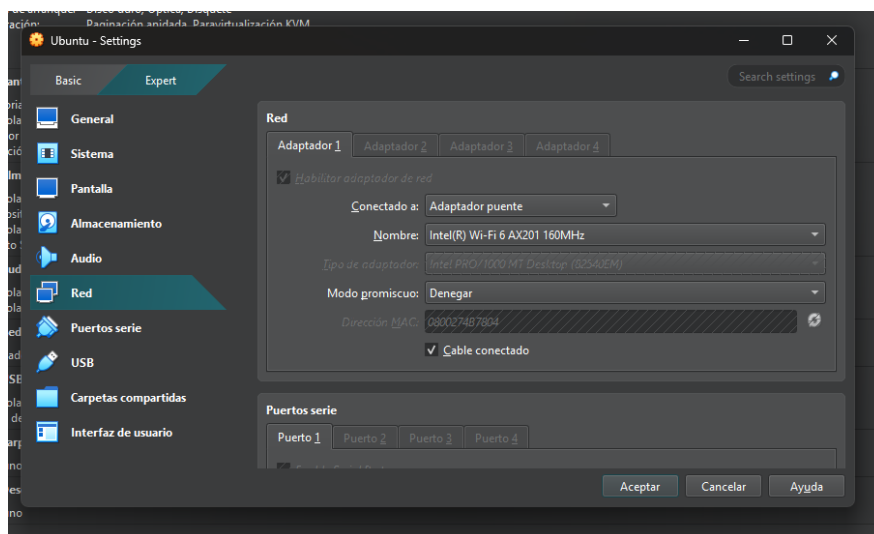
2 Instalación Coolify y Ngrok

En la máquina virtual de Ubuntu instalamos primero "curl" para poder luego instalar "coolify". Para instalar coolify buscamos coolify en el buscador y nos da un comando que nos permite instalar desde terminal.

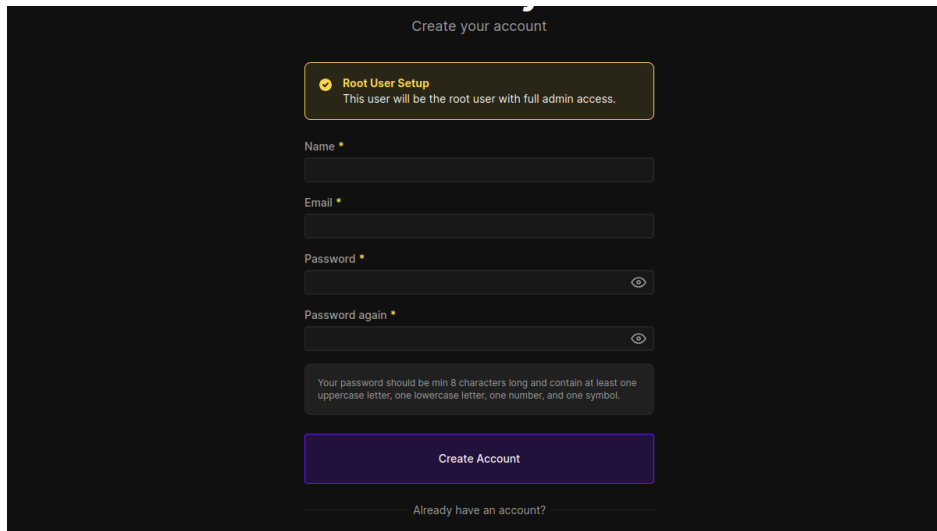


```
andrea@Ubuntu: ~  
See "man sudo_root" for details.  
  
andrea@Ubuntu:~$ sudo apt install curl  
[sudo] password for andrea:  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following NEW packages will be installed:  
  curl  
0 upgraded, 1 newly installed, 0 to remove and 122 not upgraded.  
Need to get 226 kB of archives.  
After this operation, 534 kB of additional disk space will be used.  
Get:1 http://es.archive.ubuntu.com/ubuntu noble-updates/main amd64 curl amd64 8.5.0-2ubuntu10.6 [226 kB]  
Fetched 226 kB in 1s (245 kB/s)  
Selecting previously unselected package curl.  
(Reading database ... 150423 files and directories currently installed.)  
Preparing to unpack .../curl_8.5.0-2ubuntu10.6_amd64.deb ...  
Unpacking curl (8.5.0-2ubuntu10.6) ...  
Setting up curl (8.5.0-2ubuntu10.6) ...  
Processing triggers for man-db (2.12.0-4build2) ...  
andrea@Ubuntu:~$ curl -fsSL https://cdn.coollabs.io/coolify/install.sh | sudo ba  
sh
```

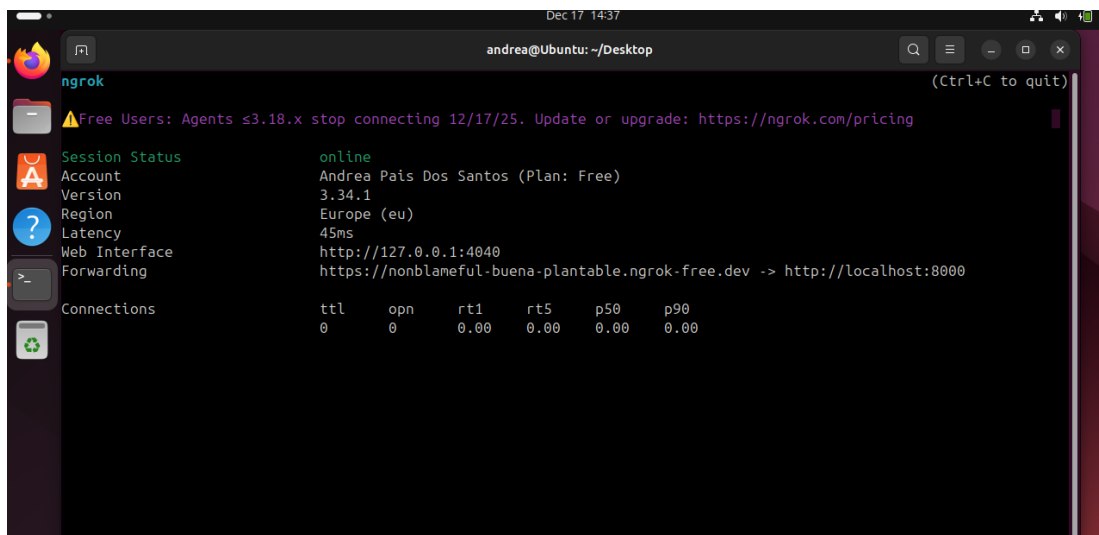
Al tener instalado "coolify" vamos a cambiar el adaptador de la máquina virtual a modo puente.



Y esto nos va a dar la ip que vamos a tener que usar para poder acceder a coolify desde fuera conectándonos desde "http://<laIp>:8000". Esto nos abre una ventana para crear una cuenta en "coolify".



Vamos a seguir más o menos los mismos pasos para instalar "ngrok". Accedemos en la máquina y obtenemos el comando para instalar "ngrok". Lo instalamos y configuramos el token que nos crea después de crear una cuenta. Con el comando "ngrok http 8000" nos ofrece un enlace que nos deja acceder a coolify desde esa dirección.



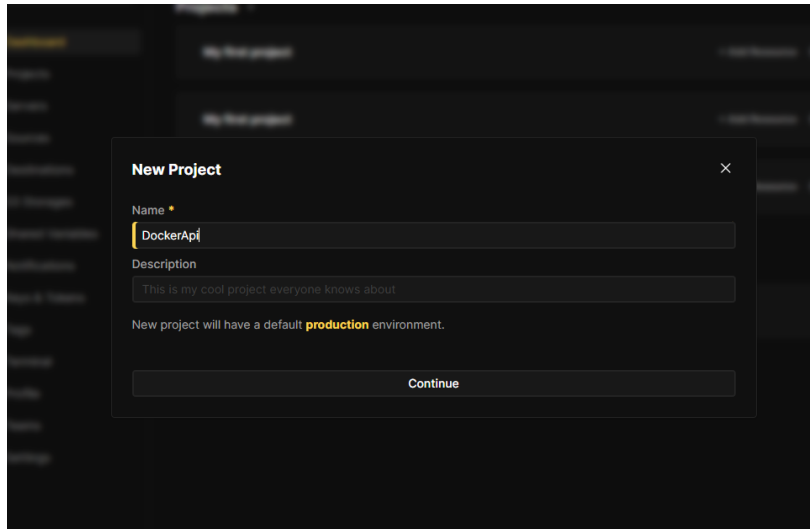
```
Dec 17 14:37 andrea@Ubuntu: ~/Desktop (Ctrl+C to quit)
ngrok
⚠ Free Users: Agents ≤3.18.x stop connecting 12/17/25. Update or upgrade: https://ngrok.com/pricing

Session Status      online
Account             Andrea Pais Dos Santos (Plan: Free)
Version             3.34.1
Region              Europe (eu)
Latency             45ms
Web Interface        http://127.0.0.1:4040
Forwarding           https://nonblameful-buena-plantable.ngrok-free.dev -> http://localhost:8000

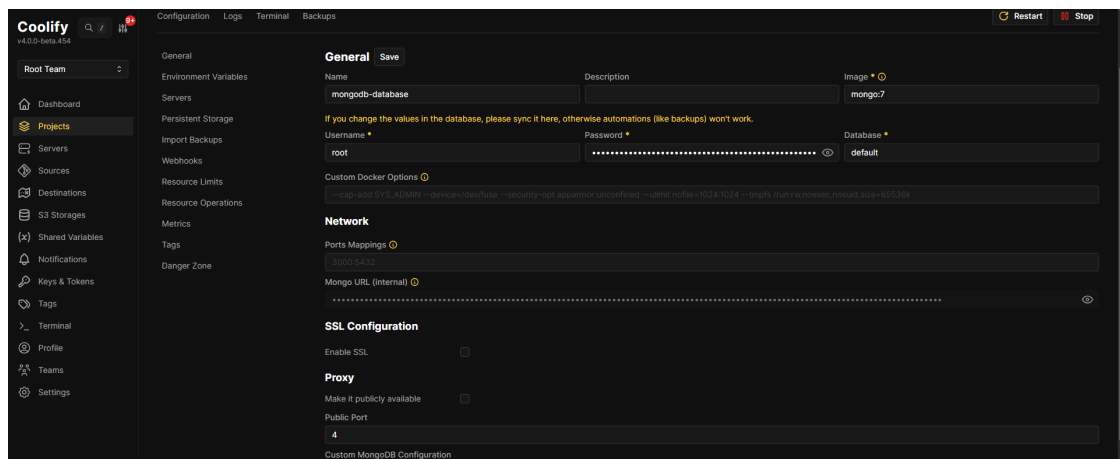
Connections
  ttl  opn  rt1  rt5  p50  p90
    0    0   0.00 0.00 0.00 0.00
```

3 Creación de un proyecto

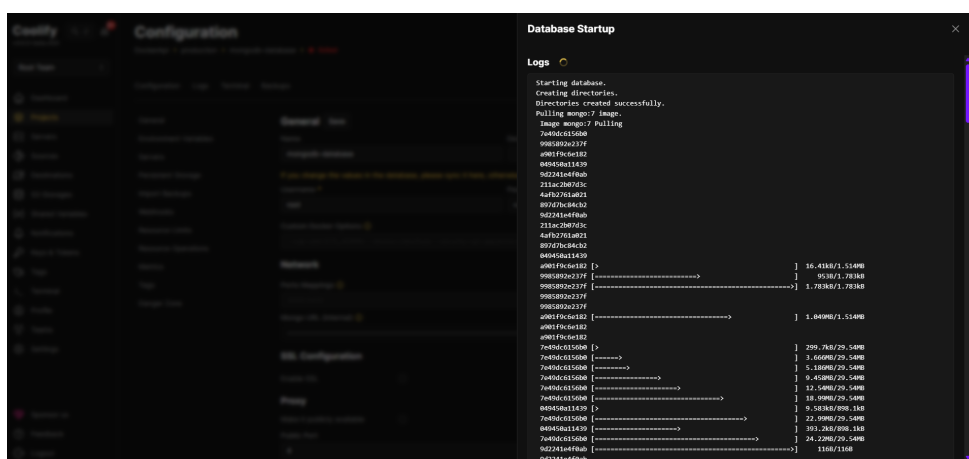
Dentro de "coolify" vamos a crear un proyecto nuevo y le ponemos el nombre que queramos.



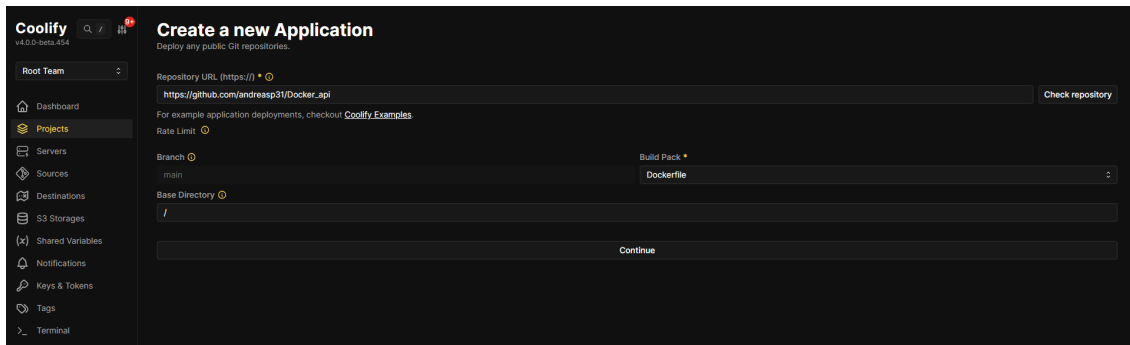
Vamos a añadir un nuevo recurso que va a ser la base de datos mongo, tenemos que hacer que sea público por lo que ponemos el puerto que consideremos y guardamos los cambios.



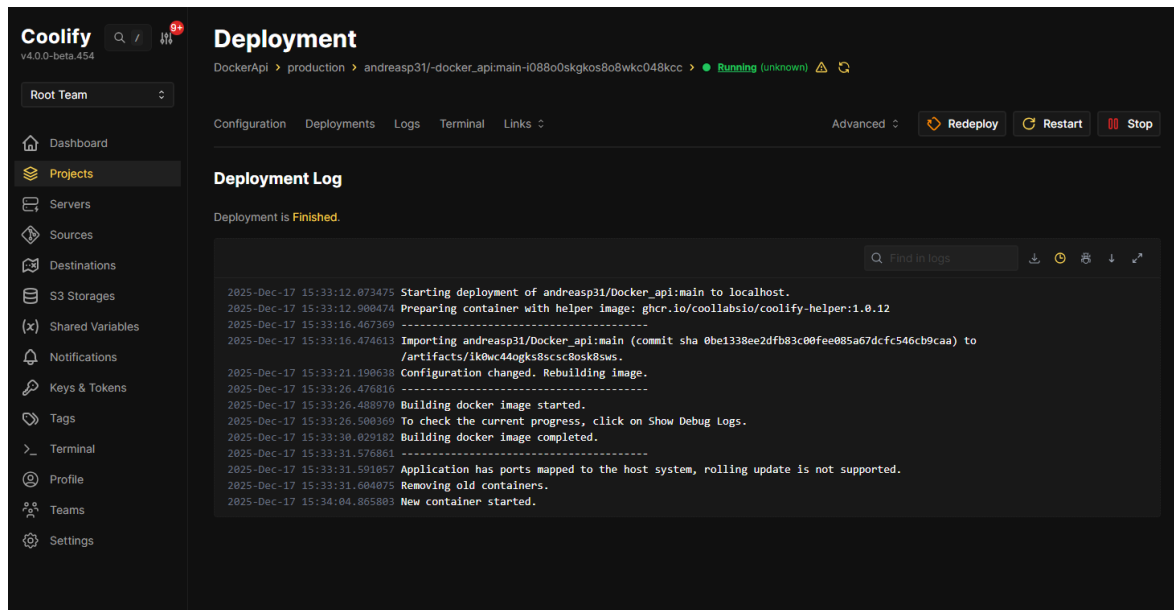
Para ver que está correcto podemos arrancar la base de datos que hemos creado.



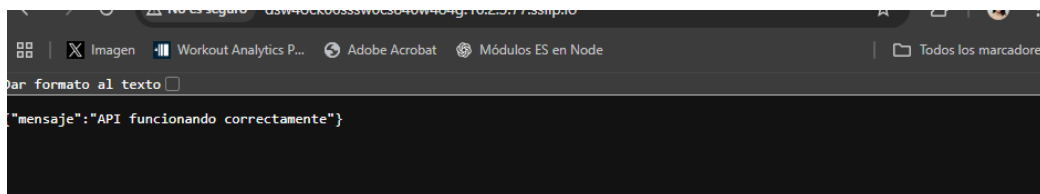
Ahora tenemos que añadir un nuevo recurso que va a ser el repositorio local de la práctica anterior de docker. Añadimos el enlace al repositorio, seleccionamos en que rama estamos y donde está nuestro archivo "Dockerfile" y el nombre que le tenemos.



Si todo está correcto, guardamos todo y lo ejecutamos.



Para comprobar que funciona la api, accedemos al dominio que nos crea al arrancarlo pero tenemos que cambiar la ip dentro del dominio y poner la ip de nuestra máquina virtual, lo guardamos y la abrimos en otra página. Si nos sale este estilo de mensaje es que se ha conectado correctamente.

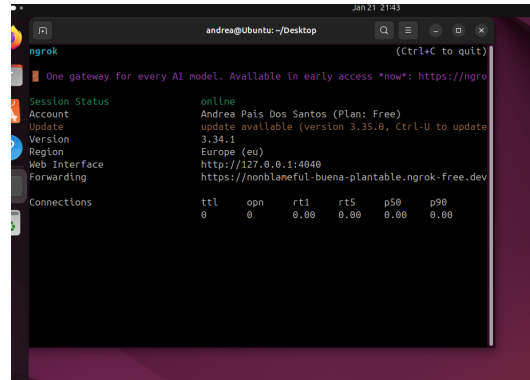


4 Puente inverso

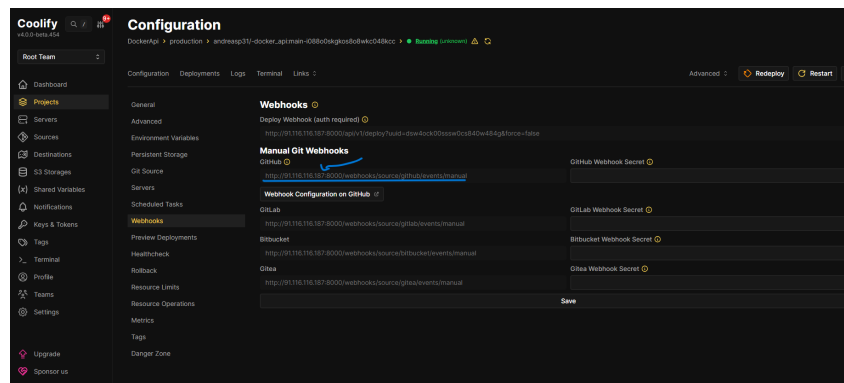
Hacer que cualquiera desde fuera se pudiera conectar desde un dispositivo externo. Me ha dado problema y no he conseguido realizar el puente.

5 Despliegue continuo

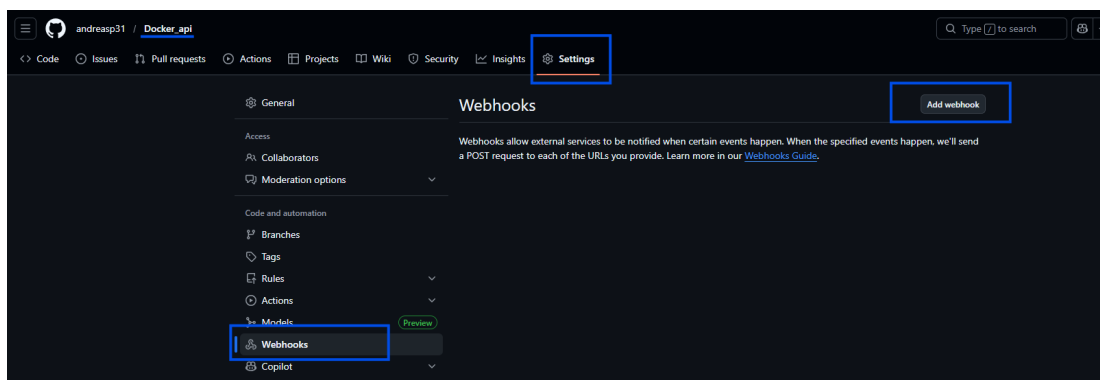
Implementar webhooks para que en el repositorio de GitHub reciba notificaciones en tiempo real cuando se realice eventos en GitHub. Para que el código se actualice solo sin tener que hacer "Redeploy" manualmente, hay que implementar webhooks de Github. Como en la máquina virtual no podemos acceder desde internet directamente, hay que usar el enlace de ngrok para usarlo de puente.



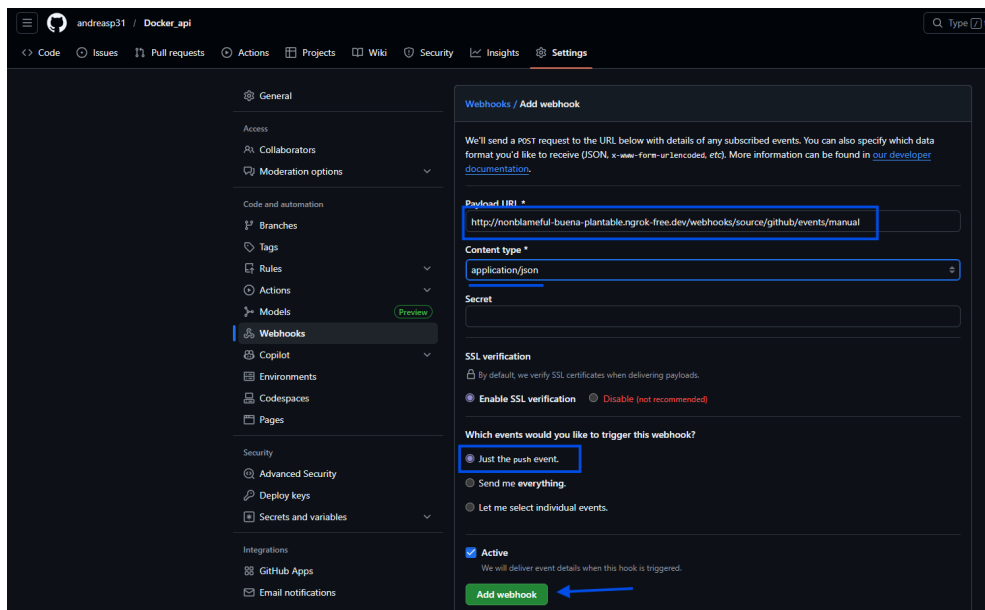
Primero, vamos a la pestaña de nuestra aplicación en Coolify y en el menú de la izquierda seleccionamos Webhooks. Copiamos la URL que nos da para GitHub.



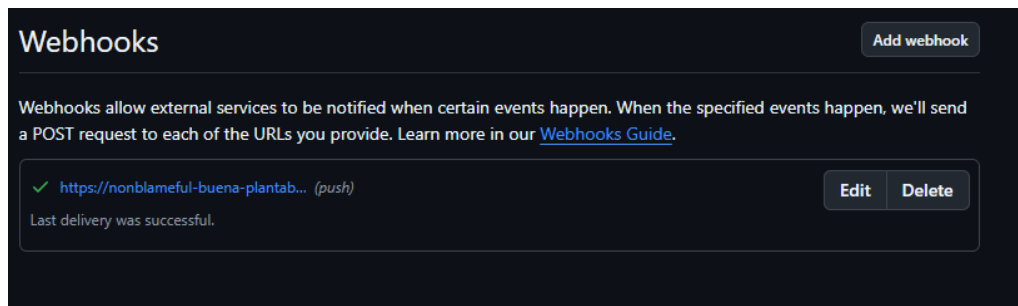
Luego, en el repositorio de GitHub, en los Settings > Webhooks y añadimos uno nuevo. En "Payload URL" pegamos la dirección de ngrok pero le añadimos al final la ruta que copiamos de Coolify. Ponemos el "Content type" en application/json, también seleccionar la opción para que funcione cuando haya un push y guardamos.



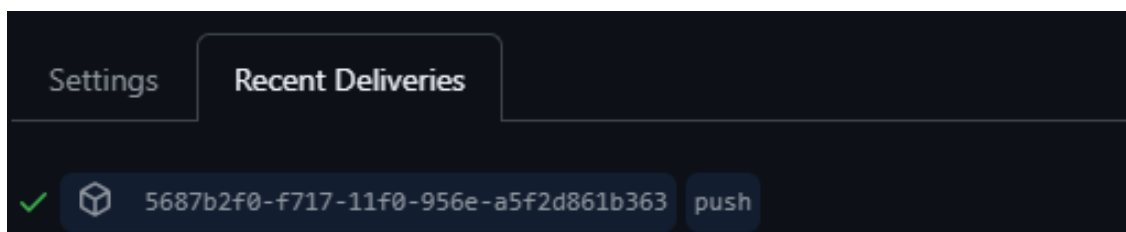
Si todo está correcto el Webhook funciona al aparecer un tick verde.



Ahora hay que realizar un push cambiando cualquier cosa, un comentario o algo.



Luego se puede ver que el push se ha realizado.



Al final vemos en Coolify dentro de el proyecto de nuestra API en el apartado de "Deployments" y vemos que se han realizado los pushes.

