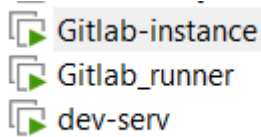


Rendu

Les 3 Machines virtuelles dont nous avons besoin pour lancer le projet

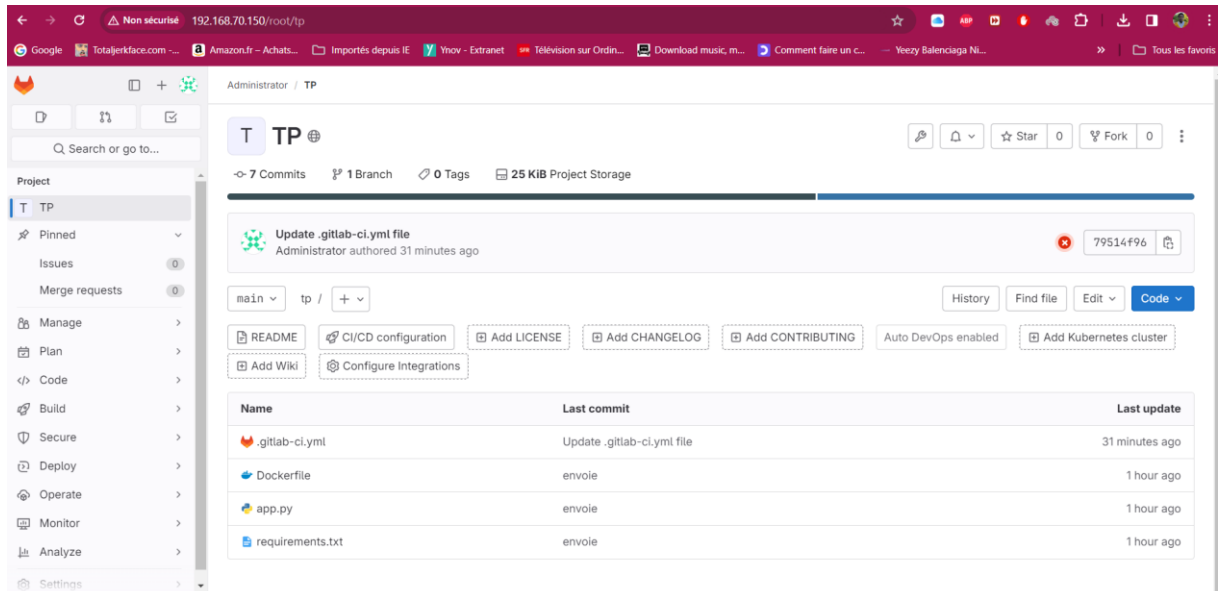


- Gitlab-Instance avec l'adresse IP que vous retrouverez bien sur le web

```
gitlab-instance x  gitlab_runner x  dev-serv x

klein@klein:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens3: <BRIDGE,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:87:10:ad brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.70.150/24 metric 100 brd 192.168.70.255 scope global dynamic ens3
        valid_lft 1318sec preferred_lft 1318sec
    inet6 fe80::20c:29ff:fe87:10ad/64 scope link
        valid_lft forever preferred_lft forever
klein@klein:~$
```

Vous pouvez bien voir la meme adresse Ip que notre Gitlab Instance



Creation de l'image sur Docker

<input type="checkbox"/>	kleinmb3446/tp b8acf460f94a	latest	In use	2 hours ago	224.26 MB	▶	⋮	🗑
<input type="checkbox"/>	tp b8acf460f94a	latest	In use	2 hours ago	224.26 MB	▶	⋮	🗑

Création de l'image Docker file

```
C:\>docker buildx build -t tp C:\Users\Klein\Documents\gitlab-instance\tp
2024/03/18 12:05:44 http2: server: error reading preface from client //./pipe/docker_engine: file has already been closed
[+] Building 67.9s (12/12) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 839B
=> [internal] load metadata for docker.io/library/python:3.10-alpine
=> [auth] library/python:pull token for registry-1.docker.io
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [1/6] FROM docker.io/library/python:3.10-alpine@sha256:d75c7712b6161e93efa287df5fceedeae2ef773f44a6b71efc6e42fec4666d
=> => resolve docker.io/library/python:3.10-alpine@sha256:d75c7712b6161e93efa287df5fceedeae2ef773f44a6b71efc6e42fec4666d
=> => sha256:c3cdf40b8bda8e4ca4be0f5fa7f1d128907271efcbc72cbfc7c8b0f939ec25ea 619.60kB / 619.60kB
=> => sha256:e364c76f7f9cc89a05dadb79c49384527e89bfad476079d9f86cfeee8fbc4f8b 12.21MB / 12.21MB
=> => sha256:d75c7712b6161e93efa287df5fceedeae2ef773f44a6b71efc6e42fec4666d 1.65kB / 1.65kB
=> => sha256:6f87e65712a96b84e61479040bc3f136b1244510e9a3f7b3ec970f31683661d8 1.37kB / 1.37kB
=> => sha256:63966ae4848813a95d7f7dbac0e5c4b5d1dab1833036f23ecf6f77765926af10 6.27kB / 6.27kB
=> => sha256:4abcf20661432fb2d719aaf90656f55c287f8ca915dc1c92ec14ff61e67fbaf8 3.41MB / 3.41MB
=> => sha256:00a2c001a0af4c6f1f281a2e431410e9e5b47480033eaea15999bfdc558219a3 239B / 239B
=> => sha256:77b5492a4815804d074ff11a54ae3c5c3bbd6c6e2f9ba5f95f62e1f567f1e7bf 3.08MB / 3.08MB
=> => extracting sha256:4abcf20661432fb2d719aaf90656f55c287f8ca915dc1c92ec14ff61e67fbaf8
=> => extracting sha256:c3cdf40b8bda8e4ca4be0f5fa7f1d128907271efcbc72cbfc7c8b0f939ec25ea
=> => extracting sha256:e364c76f7f9cc89a05dadb79c49384527e89bfad476079d9f86cfeee8fbc4f8b
=> => extracting sha256:00a2c001a0af4c6f1f281a2e431410e9e5b47480033eaea15999bfdc558219a3
=> => extracting sha256:77b5492a4815804d074ff11a54ae3c5c3bbd6c6e2f9ba5f95f62e1f567f1e7bf
=> [internal] load build context
=> => transferring context: 1.45kB
=> [2/6] WORKDIR /code
=> [3/6] RUN apk add --no-cache gcc musl-dev linux-headers
=> [4/6] COPY requirements.txt requirements.txt
=> [5/6] RUN pip install -r requirements.txt
=> [6/6] COPY . .
=> exporting to image
=> => exporting layers
=> => writing image sha256:b8acf460f94abedabee862f737a781248202763309088b0f864ad9c6fab6a6b
=> => naming to docker.io/library/tp
```