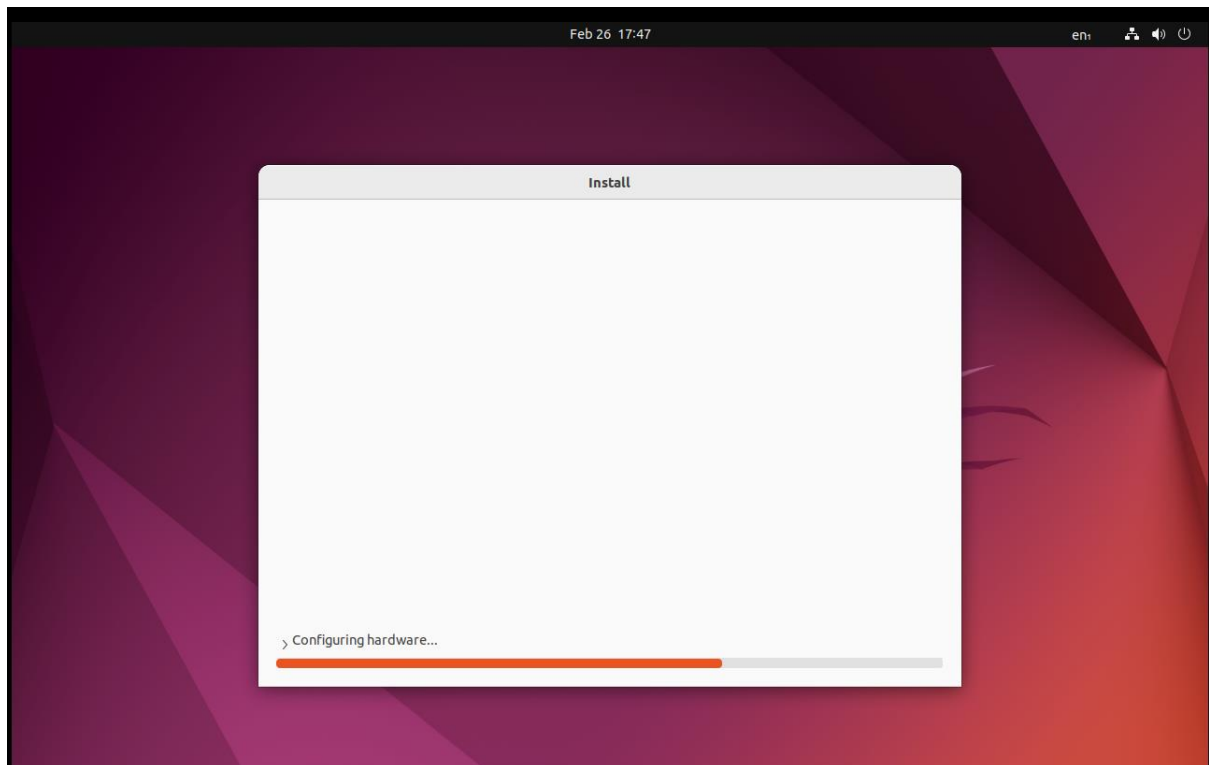


A-1 DEVOPS

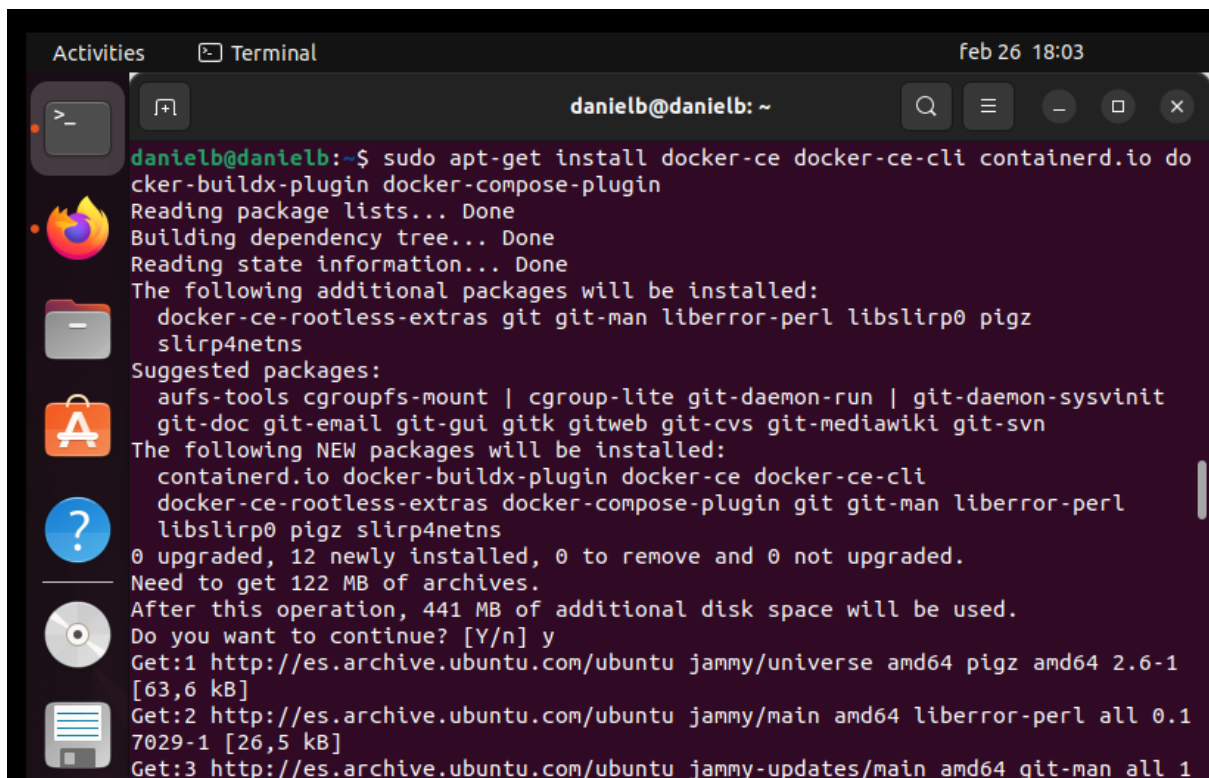
Daniel Barragues

INSTALACIÓN DE UBUNTU



Instalamos Ubuntu en una máquina virtual para la ejecución de la actividad.

INSTALACIÓN DE DOCKER



```
Activities Terminal feb 26 18:04
danielb@danielb: ~
Status: Downloaded newer image for hello-world:latest
Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

danielb@danielb:~$
```

CREACIÓN DEL PROYECTO DE PYTHON

Creamos el proyecto Python tal cual indican las instrucciones con un script Python que muestra nuestro nombre y un Dockerfile que se descargue la última imagen de Python disponible para ejecutar el script 'main.py'.

```
danielb@danielb:~/ProPython$ tree
.
├── Dockerfile
└── main.py

0 directories, 2 files
```

GENERAMOS LA IMAGEN CON NUESTRO NOMBRE

```
danielb@danielb:~/ProPython$ sudo docker build -t python-danielbarragues .
[+] Building 3.7s (4/6) docker:default
=> [internal] load build definition from Dockerfile 0.0s
=> => transferring dockerfile: 98B 0.0s
=> [internal] load metadata for docker.io/library/python:latest 2.3s
=> [internal] load .dockerignore 0.0s
=> => transferring context: 2B 0.0s
=> [internal] load build context 0.0s
=> => transferring context: 199B 0.0s
=> [1/2] FROM docker.io/library/python:latest@sha256:e83d1f4d0c735c7a54fc9dae3cca8c58473e3b3de08fcb7ba3d342ee75cfc09d 1.3s
=> => resolve docker.io/library/python:latest@sha256:e83d1f4d0c735c7a54fc9dae3cca8c58473e3b3de08fcb7ba3d342ee75cfc09d 0.0s
=> => sha256:a3aef03c6c1029222ff035a72a0fcaddba1fd7a26abdf2491ace8eb3c654471e 7.11kB / 7.11kB 0.0s
=> => sha256:7bb465c2914923b08ae03b7fc67b92a1ef9b09c4c1eb9d6711b22ee6bbb46a00 24.12MB / 49.55MB 1.3s
=> => sha256:2b9b41aaa3c52ab268b47da303015b94ced04a1eb02e58860e58b283404974f4 6.29MB / 24.05MB 1.3s
=> => sha256:49b40be4436eff6fe463f6977159dc727df37cabe65ade75c75c1caa3cb0a234 0B / 64.14MB 1.3s
=> => sha256:e83d1f4d0c735c7a54fc9dae3cca8c58473e3b3de08fcb7ba3d342ee75cfc09d 2.14kB / 2.14kB 0.0s
=> => sha256:35eff340c0acd837b7962f77ee4b8869385dd6fe7d3928375a08f0a3bdd18beb 2.01kB / 2.01kB 0.0s
```

EJECUTAMOS LA IMAGEN EN DOCKER

```
danielb@danielb:~/ProPython$ sudo docker run python-danielbarrague
;Soy Daniel Barragues de 2 de DAM!
Un pequeno loop como extra0
Un pequeno loop como extra1
Un pequeno loop como extra2
Un pequeno loop como extra3
danielb@danielb:~/ProPython$
```

(En el script he hecho un pequeño loop como algo extra para mostrar)

```
GNU nano 6.2
#!/usr/bin/env python3
print(";Soy Daniel Barragues de 2 de DAM!")

x = 0
for x in range(0,4):
    print("Un pequeno loop como extra" + str(x))
    x+=1
```