

# MEMORIA

# SISTEMAS

# INFORMATICOS



## Índice

INTRODUCCIÓN.....	3
OBJETIVOS.....	3
Material per a realitzar la pràctica.....	4
TAREAS A REALIZAR.....	4
Comprobaciones antes de comenzar.....	4
ACTIVIDAD.....	4
.....	13
1. Preparar el Dockerfile.....	18
2. Crear la imagen de Docker.....	18
3. Probar la imagen.....	19
1. Preparar el Dockerfile.....	19
2. Crear la Imagen Docker.....	19
3. Probar la Imagen Docker.....	20
1. Preparar el Dockerfile.....	21
2. Crear el script start.sh.....	21
Paso 1: Crear la Red.....	22
Paso 2: Crear el Contenedor MySQL.....	22
Paso 3: Crear el Contenedor Wordpress.....	23
Paso 4: Instalar Wordpress.....	24
Migrar Contenedor MySQL de 5.6 a 5.7.....	25
Paso 1: Crear el archivo docker-compose.yml.....	26
CONCLUSIÓN:.....	31
WEBGRAFIA:.....	31

# INTRODUCCIÓN

En la actualidad digital, Windows se destaca como un pilar fundamental para la eficiencia informática. Este trabajo se enfoca en la práctica de Windows, explorando la instalación, administración y configuración del sistema operativo. Desde la elección de la versión adecuada hasta la configuración inicial, la instalación establece la base para un entorno informático eficiente. La administración, abordando la gestión de usuarios y políticas de seguridad, promueve la optimización operativa y la seguridad de la información. Asimismo, la configuración permite adaptar Windows a necesidades específicas, desde la interfaz hasta configuraciones avanzadas, facilitando una personalización flexible. En conjunto, este enfoque integral busca resaltar la importancia de estos procesos para la creación y mantenimiento de un entorno informático seguro y eficiente, aprovechando al máximo las funcionalidades de Windows.

## OBJETIVOS

- 1. Diferencias en Particiones entre UEFI y Legacy:** Identificar y analizar las variaciones en la configuración de particiones del disco al instalar un sistema operativo Windows sobre una UEFI en comparación con un sistema Legacy.
- 2. Relación con la Tabla de Particiones MBR y GPT:** Relacionar la respuesta anterior con las estructuras de las tablas de particiones MBR (Master Boot Record) y GPT (GUID Partition Table).
- 3. Verificación del Firmware de la BIOS:** Ejecutar el comando "msinfo32.exe" en una máquina virtual de Windows 11 o Windows 10 para localizar y comprobar la información sobre el firmware de la BIOS.

## Material per a realitzar la pràctica

### Indicar característiques principals de l'equip (benchmark):

Marca y model del Processador : 12th Gen Intel(R) Core(TM) i7-12700H 2.30 GHz

Capacitat de Memòria RAM : 16,0 GB

Tipus de dispositiu d'emmagatzematge i capacitat (GiB) : 1TB SSD

# TAREAS A REALIZAR

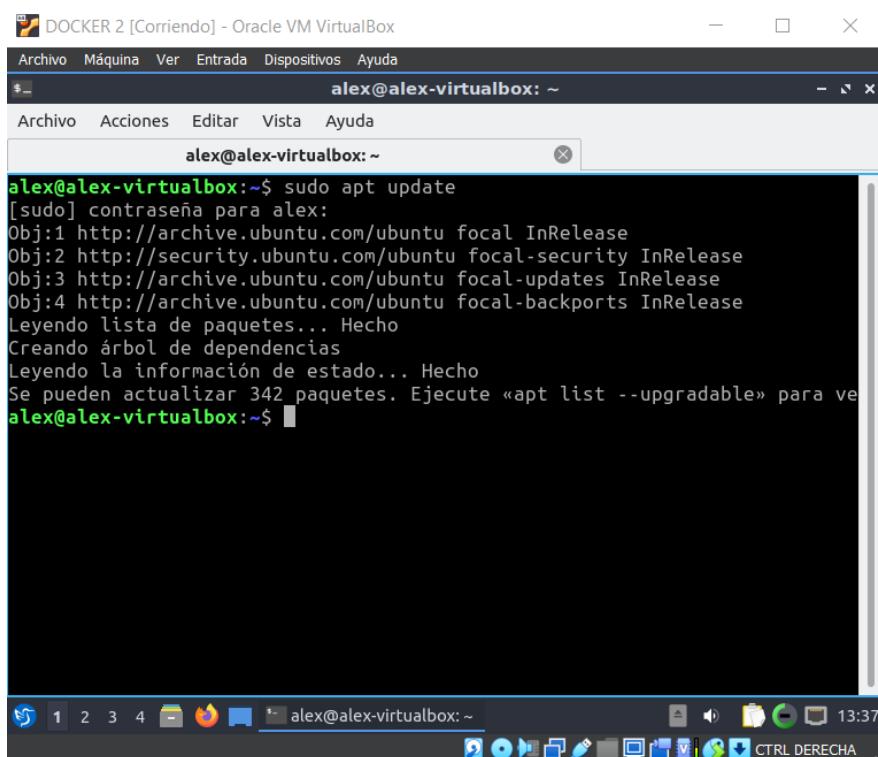
## Comprobaciones antes de comenzar

1. Comprueba que tu PC tenga instalado Virtualbox, en caso contrario avisar al profesor. Y acceder a los archivos en la carpeta compartida e instalarlo en una máquina virtual (MV) siguiendo la guía práctica.
2. Comprueba si puedes acceder al repositorio que contiene los programas para realizar la práctica, en caso contrario usa la memoria USB y/o accede a la carpeta de un compañero que no tenga ese problema.

## ACTIVIDAD

### 1. EJECUTAR EN LA MV LA INSTALACIÓN COMPLETA DE DOCKER

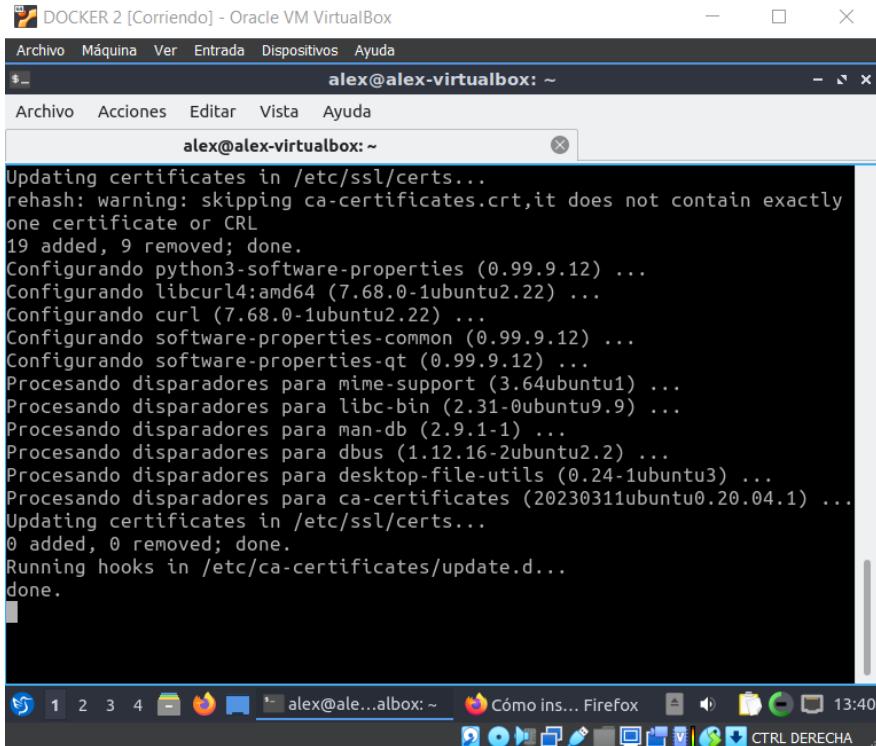
Para empezar la instalación lo primero que tenemos que hacer es actualizar todo con "sudo apt update".



The screenshot shows a terminal window titled "alex@alex-virtualbox: ~" running on a virtual machine. The window has a title bar with the name and status "DOCKER 2 [Corriendo] - Oracle VM VirtualBox". The menu bar includes "Archivo", "Máquina", "Ver", "Entrada", "Dispositivos", and "Ayuda". Below the menu is a toolbar with icons for "Archivo", "Acciones", "Editar", "Vista", and "Ayuda". The terminal itself displays the command "sudo apt update" being run, followed by the output of the command, which includes package lists from various Ubuntu repositories and a message indicating that 342 packages can be updated. The bottom of the screen shows the desktop environment's taskbar with various application icons and the system tray.

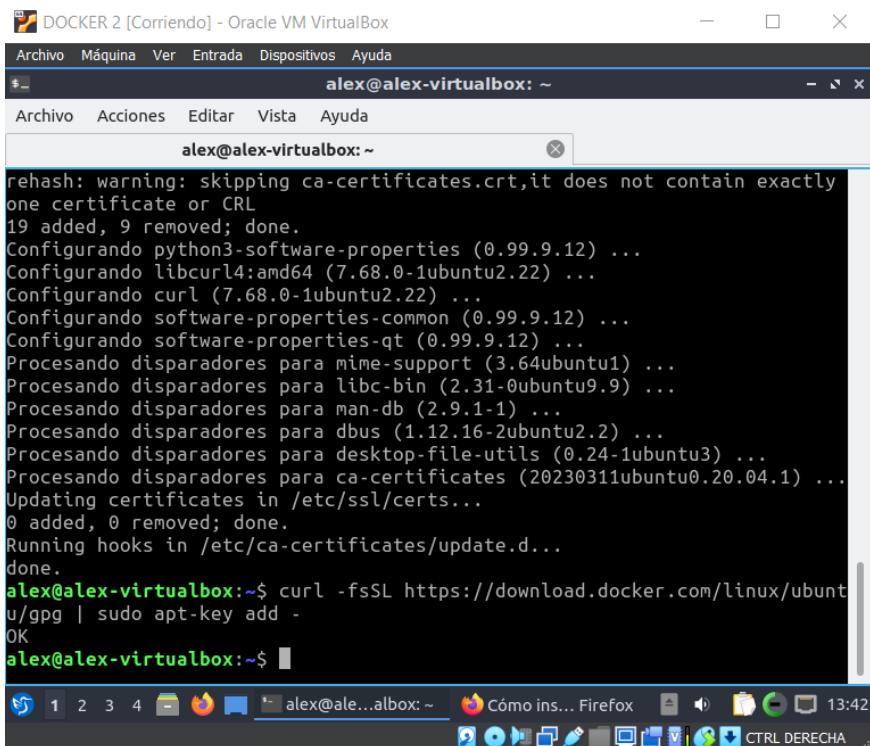
```
alex@alex-virtualbox:~$ sudo apt update
[sudo] contraseña para alex:
Obj:1 http://archive.ubuntu.com/ubuntu focal InRelease
Obj:2 http://security.ubuntu.com/ubuntu focal-security InRelease
Obj:3 http://archive.ubuntu.com/ubuntu focal-updates InRelease
Obj:4 http://archive.ubuntu.com/ubuntu focal-backports InRelease
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se pueden actualizar 342 paquetes. Ejecute «apt list --upgradable» para ver
alex@alex-virtualbox:~$
```

A continuación vamos a instalar algunos paquetes de requisitos previos que nos permitan usar paquetes a través de HTTPS, con el comando "sudo apt install apt-transport-https ca-certificates curl software-properties-common".



```
Updating certificates in /etc/ssl/certs...
rehash: warning: skipping ca-certificates.crt,it does not contain exactly
one certificate or CRL
19 added, 9 removed; done.
Configurando python3-software-properties (0.99.9.12) ...
Configurando libcurl4:amd64 (7.68.0-1ubuntu2.22) ...
Configurando curl (7.68.0-1ubuntu2.22) ...
Configurando software-properties-common (0.99.9.12) ...
Configurando software-properties-qt (0.99.9.12) ...
Procesando disparadores para mime-support (3.64ubuntu1) ...
Procesando disparadores para libc-bin (2.31-0ubuntu9.9) ...
Procesando disparadores para man-db (2.9.1-1) ...
Procesando disparadores para dbus (1.12.16-2ubuntu2.2) ...
Procesando disparadores para desktop-file-utils (0.24-1ubuntu3) ...
Procesando disparadores para ca-certificates (20230311ubuntu0.20.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
```

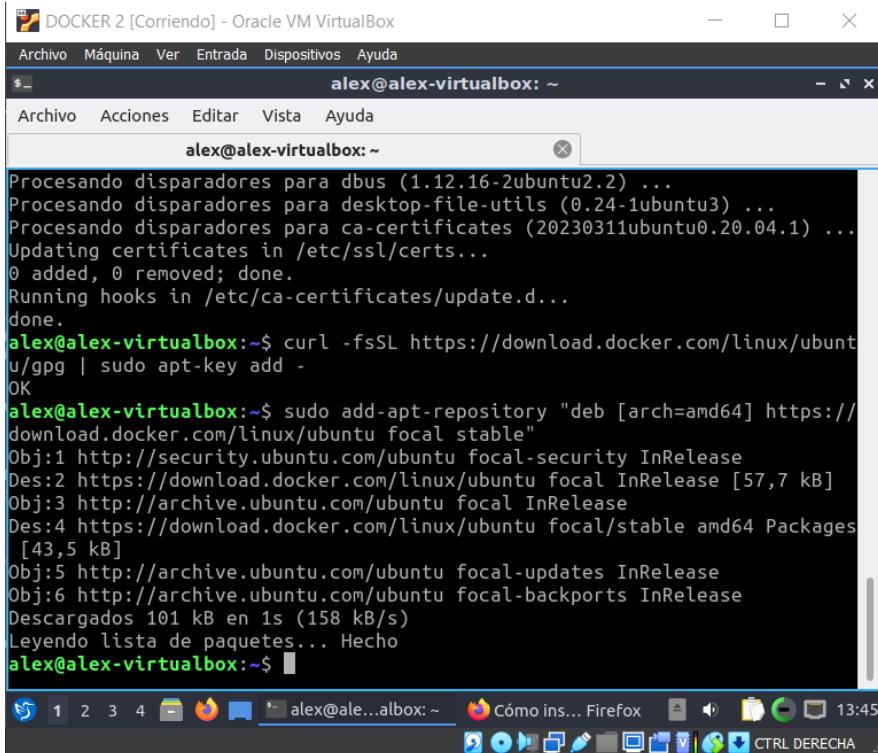
Luego hay que añadir la clave GPG , para el repositorio oficial de Docker. Usando el comando "curl -fsSL  
<https://download.docker.com/linux/ubuntu/gpg> | sudo apt-key add -"



```
rehash: warning: skipping ca-certificates.crt,it does not contain exactly
one certificate or CRL
19 added, 9 removed; done.
Configurando python3-software-properties (0.99.9.12) ...
Configurando libcurl4:amd64 (7.68.0-1ubuntu2.22) ...
Configurando curl (7.68.0-1ubuntu2.22) ...
Configurando software-properties-common (0.99.9.12) ...
Configurando software-properties-qt (0.99.9.12) ...
Procesando disparadores para mime-support (3.64ubuntu1) ...
Procesando disparadores para libc-bin (2.31-0ubuntu9.9) ...
Procesando disparadores para man-db (2.9.1-1) ...
Procesando disparadores para dbus (1.12.16-2ubuntu2.2) ...
Procesando disparadores para desktop-file-utils (0.24-1ubuntu3) ...
Procesando disparadores para ca-certificates (20230311ubuntu0.20.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
alex@alex-virtualbox:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
OK
alex@alex-virtualbox:~$
```

Agregamos el repositorio Docker a las fuentes de APT.

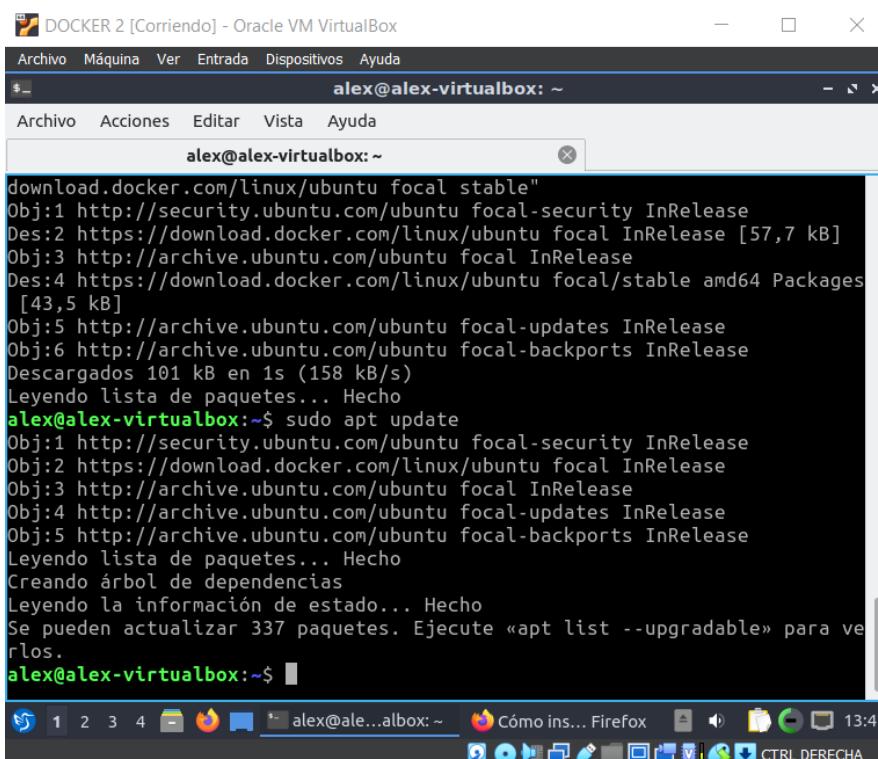
Usando el comando "sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable".



```
DOCKER 2 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
alex@alex-virtualbox: ~
Archivo Acciones Editar Vista Ayuda
alex@alex-virtualbox: ~

Procesando disparadores para dbus (1.12.16-2ubuntu2.2) ...
Procesando disparadores para desktop-file-utils (0.24-1ubuntu3) ...
Procesando disparadores para ca-certificates (20230311ubuntu0.20.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
alex@alex-virtualbox:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
OK
alex@alex-virtualbox:~$ sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable"
Obj:1 http://security.ubuntu.com/ubuntu focal-security InRelease
Des:2 https://download.docker.com/linux/ubuntu focal InRelease [57,7 kB]
Obj:3 http://archive.ubuntu.com/ubuntu focal InRelease
Des:4 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages [43,5 kB]
Obj:5 http://archive.ubuntu.com/ubuntu focal-updates InRelease
Obj:6 http://archive.ubuntu.com/ubuntu focal-backports InRelease
Descargados 101 kB en 1s (158 kB/s)
Leyendo lista de paquetes... Hecho
alex@alex-virtualbox:~$
```

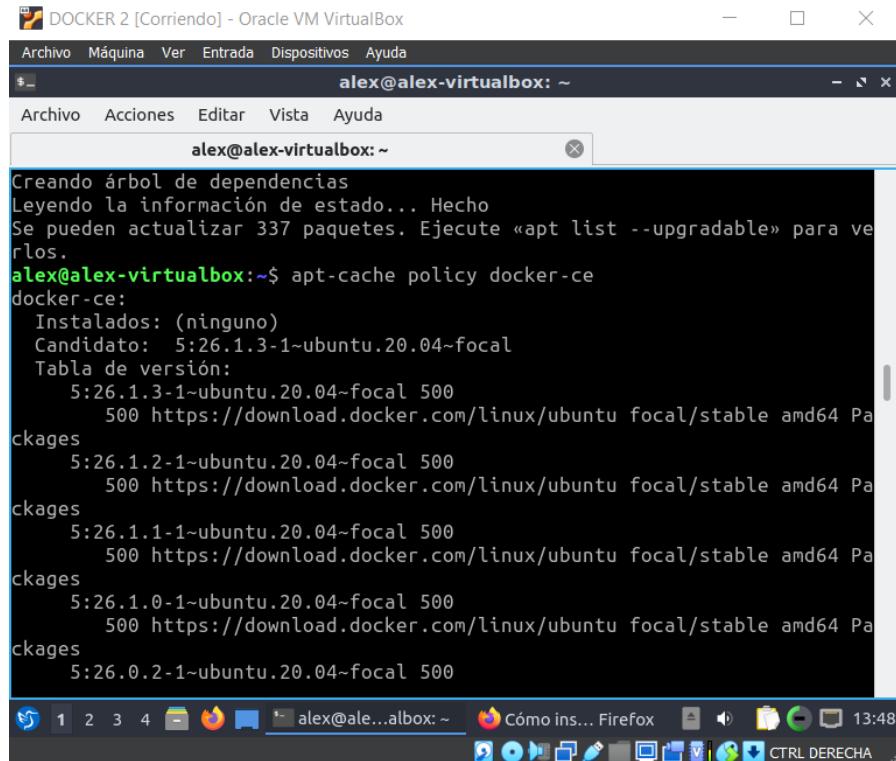
Ahora actualizamos el paquete de base de datos, con los paquetes de Docker del repositorio recien agregados. Con el comando "sudo apt update".



```
DOCKER 2 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
alex@alex-virtualbox: ~
Archivo Acciones Editar Vista Ayuda
alex@alex-virtualbox: ~

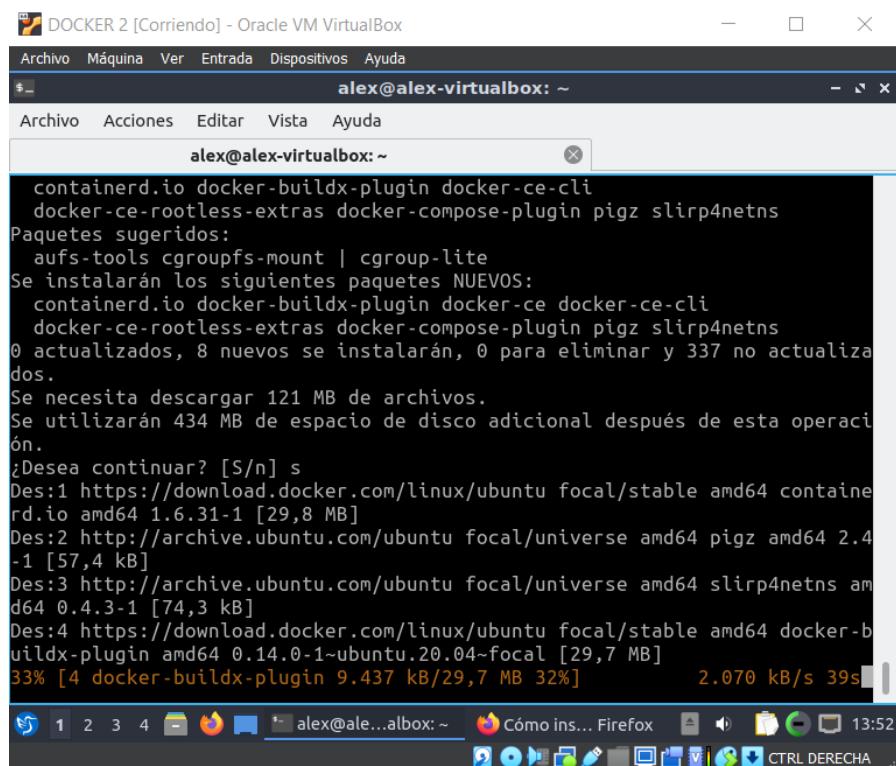
download.docker.com/linux/ubuntu focal stable"
Obj:1 http://security.ubuntu.com/ubuntu focal-security InRelease
Des:2 https://download.docker.com/linux/ubuntu focal InRelease [57,7 kB]
Obj:3 http://archive.ubuntu.com/ubuntu focal InRelease
Des:4 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages [43,5 kB]
Obj:5 http://archive.ubuntu.com/ubuntu focal-updates InRelease
Obj:6 http://archive.ubuntu.com/ubuntu focal-backports InRelease
Descargados 101 kB en 1s (158 kB/s)
Leyendo lista de paquetes... Hecho
alex@alex-virtualbox:~$ sudo apt update
Obj:1 http://security.ubuntu.com/ubuntu focal-security InRelease
Obj:2 https://download.docker.com/linux/ubuntu focal InRelease
Obj:3 http://archive.ubuntu.com/ubuntu focal InRelease
Obj:4 http://archive.ubuntu.com/ubuntu focal-updates InRelease
Obj:5 http://archive.ubuntu.com/ubuntu focal-backports InRelease
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se pueden actualizar 337 paquetes. Ejecute «apt list --upgradable» para verlos.
alex@alex-virtualbox:~$
```

Hay que asegurarse de que la instalación se realiza desde el directorio Docker y no desde el lubuntu.



```
alex@alex-virtualbox:~$ apt-cache policy docker-ce
docker-ce:
  Instalados: (ninguno)
  Candidato: 5:26.1.3-1~ubuntu.20.04~focal
  Tabla de versión:
    5:26.1.3-1~ubuntu.20.04~focal 500
      500 https://download.docker.com/linux/ubuntu focal/stable amd64 Pa
ckages
    5:26.1.2-1~ubuntu.20.04~focal 500
      500 https://download.docker.com/linux/ubuntu focal/stable amd64 Pa
ckages
    5:26.1.1-1~ubuntu.20.04~focal 500
      500 https://download.docker.com/linux/ubuntu focal/stable amd64 Pa
ckages
    5:26.1.0-1~ubuntu.20.04~focal 500
      500 https://download.docker.com/linux/ubuntu focal/stable amd64 Pa
ckages
    5:26.0.2-1~ubuntu.20.04~focal 500
```

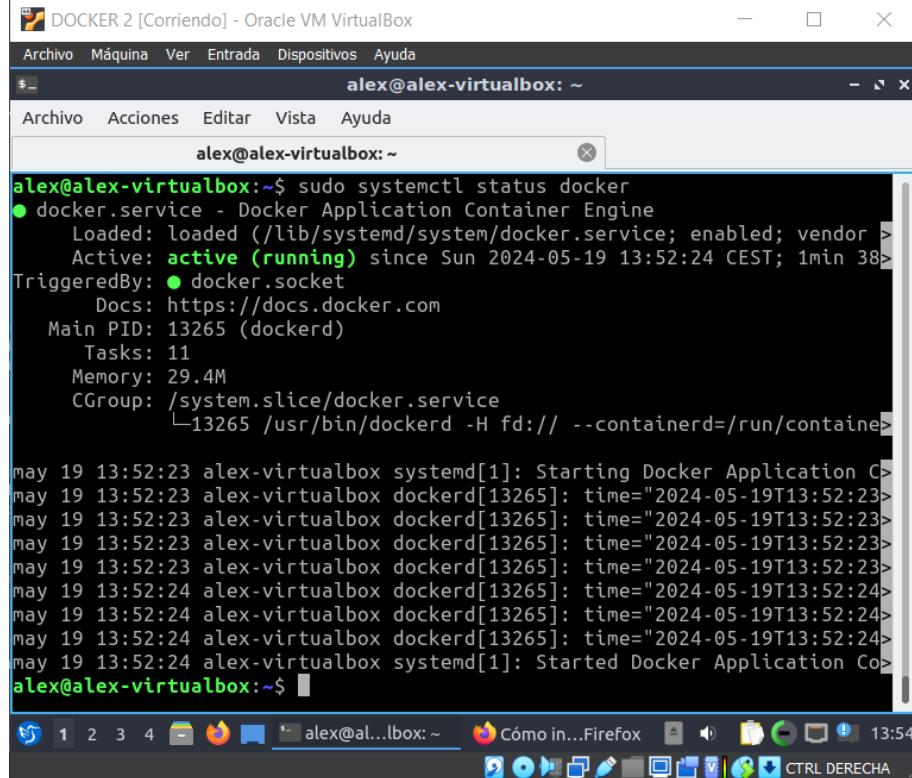
Ahora instalamos el Docker, con el comando "sudo apt install docker-ce".



```
alex@alex-virtualbox:~$ sudo apt install docker-ce
Paquetes sugeridos:
  aufs-tools cgroupfs-mount | cgroup-lite
Se instalarán los siguientes paquetes NUEVOS:
  containerd.io docker-buildx-plugin docker-ce-cli
  docker-ce-rootless-extras docker-compose-plugin pigz slirp4netns
0 actualizados, 8 nuevos se instalarán, 0 para eliminar y 337 no actualiza
dos.
Se necesita descargar 121 MB de archivos.
Se utilizarán 434 MB de espacio de disco adicional después de esta operaci
ón.
¿Desea continuar? [S/n] s
Des:1 https://download.docker.com/linux/ubuntu focal/stable amd64 containe
rd.io amd64 1:6.31-1 [29,8 MB]
Des:2 http://archive.ubuntu.com/ubuntu focal/universe amd64 pigz amd64 2.4
-1 [57,4 kB]
Des:3 http://archive.ubuntu.com/ubuntu focal/universe amd64 slirp4netns am
d64 0:4.3-1 [74,3 kB]
Des:4 https://download.docker.com/linux/ubuntu focal/stable amd64 docker-b
uildx-plugin amd64 0:14.0-1~ubuntu.20.04~focal [29,7 MB]
33% [4 docker-buildx-plugin 9.437 kB/29,7 MB 32%] 2.070 kB/s 39s
```

Despues de instalarlo, el demon se iniciará y el proceso se habilitará para ejecutarse en el inicio.

Comprobamos que funciona con el comando "sudo systemctl status docker".

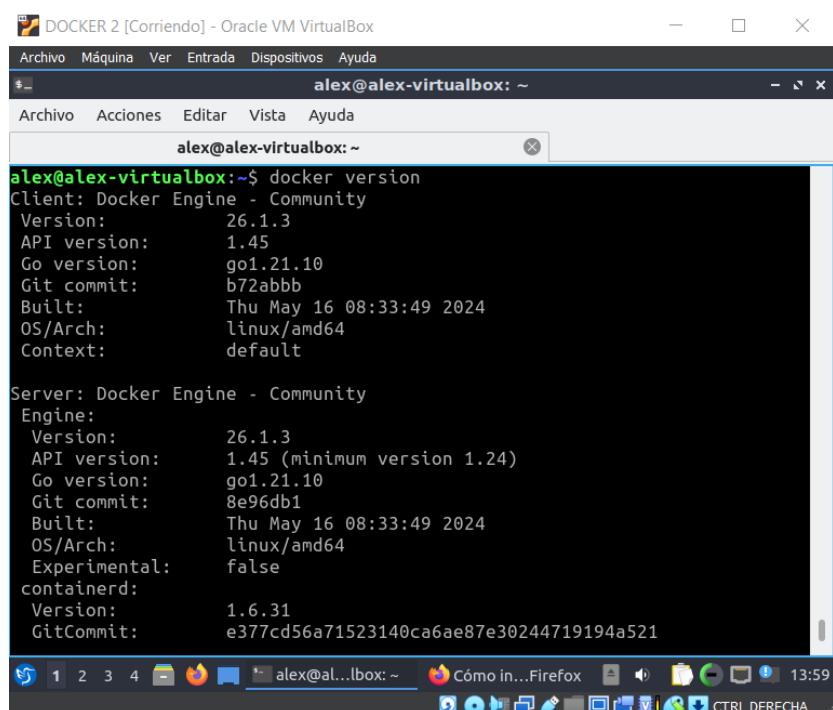


```
alex@alex-virtualbox:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor >
     Active: active (running) since Sun 2024-05-19 13:52:24 CEST; 1min 38s
   TriggeredBy: ● docker.socket
     Docs: https://docs.docker.com
   Main PID: 13265 (dockerd)
     Tasks: 11
    Memory: 29.4M
      CGroup: /system.slice/docker.service
              └─13265 /usr/bin/dockerd -H fd:// --containerd=/run/containerd.sock

may 19 13:52:23 alex-virtualbox systemd[1]: Starting Docker Application Co>
may 19 13:52:23 alex-virtualbox dockerd[13265]: time="2024-05-19T13:52:23Z" level=info msg="Docker daemon is starting up"
may 19 13:52:23 alex-virtualbox dockerd[13265]: time="2024-05-19T13:52:23Z" level=info msg="parsed config: &{GraphDriver:&{'Name':>
may 19 13:52:23 alex-virtualbox dockerd[13265]: time="2024-05-19T13:52:23Z" level=info msg="loading containerd"
may 19 13:52:24 alex-virtualbox dockerd[13265]: time="2024-05-19T13:52:24Z" level=info msg="containerd is running in a container, so the docker daemon will not run a container"
may 19 13:52:24 alex-virtualbox dockerd[13265]: time="2024-05-19T13:52:24Z" level=info msg="Docker daemon is ready to handle requests"
may 19 13:52:24 alex-virtualbox dockerd[13265]: time="2024-05-19T13:52:24Z" level=info msg="Listening for HTTP on /var/run/docker.sock"
may 19 13:52:24 alex-virtualbox dockerd[13265]: time="2024-05-19T13:52:24Z" level=info msg="Listening for GRPC on /var/run/docker.sock"
may 19 13:52:24 alex-virtualbox dockerd[13265]: time="2024-05-19T13:52:24Z" level=info msg="Docker daemon is ready to handle requests"
may 19 13:52:24 alex-virtualbox systemd[1]: Started Docker Application Co>
alex@alex-virtualbox:~$
```

Salimos de este estado, presionando la Q.

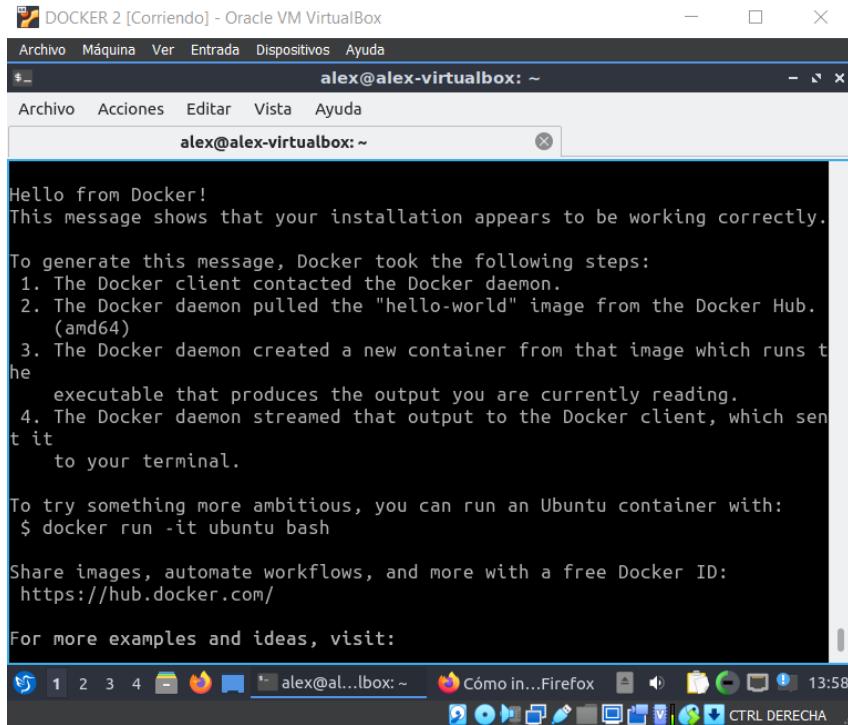
Para saber la version de Docker que usamos el comando de "docker version".



```
alex@alex-virtualbox:~$ docker version
Client: Docker Engine - Community
 Version:          26.1.3
 API version:      1.45
 Go version:       go1.21.10
 Git commit:       b72abbb
 Built:            Thu May 16 08:33:49 2024
 OS/Arch:          linux/amd64
 Context:          default

Server: Docker Engine - Community
Engine:
 Version:          26.1.3
 API version:      1.45 (minimum version 1.24)
 Go version:       go1.21.10
 Git commit:       8e96db1
 Built:            Thu May 16 08:33:49 2024
 OS/Arch:          linux/amd64
 Experimental:    false
 containerd:
 Version:          1.6.31
 GitCommit:        e377cd56a71523140ca6ae87e30244719194a521
```

Ahora usamos el comando de "docker run hello-world" para que nos muestre que la instalación se ha realizado correctamente.



```
DOCKER 2 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
$ alex@alex-virtualbox: ~
Archivo Acciones Editar Vista Ayuda
alex@alex-virtualbox: ~

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 t
he
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sen
t it
   to your terminal.

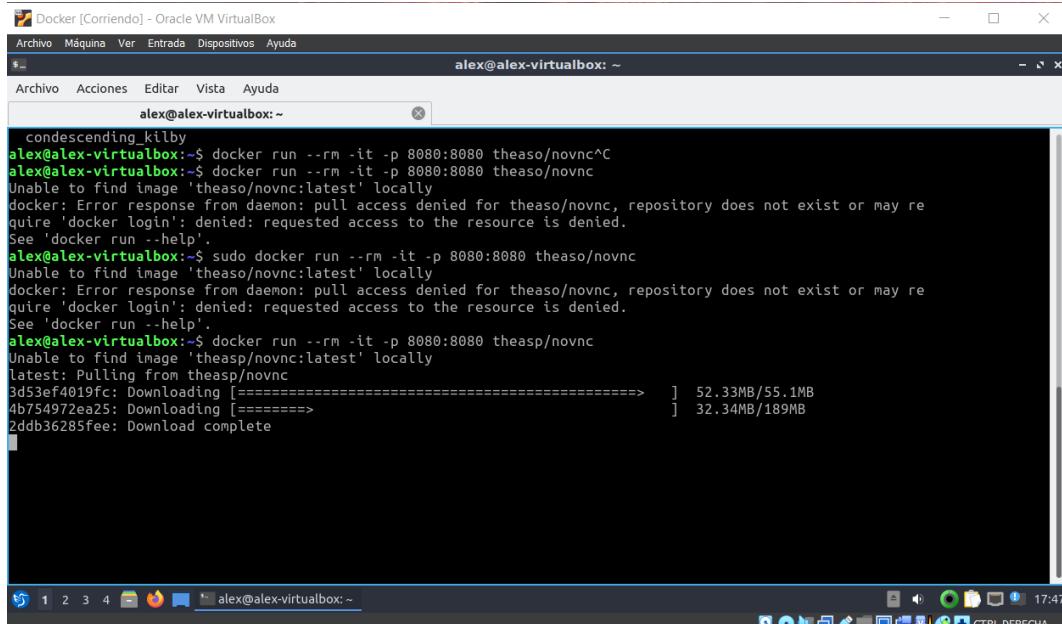
To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

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

For more examples and ideas, visit:
 1 2 3 4 alex@al...lbox: ~ Cómo in...Firefox 13:58
CTRL DERECHA
```

## 2. CAPTURA FINAL DEL NAVEGADOR DONDE SE OBSERVA QUE NoVNC ESTÁ FUNCIONANDO.

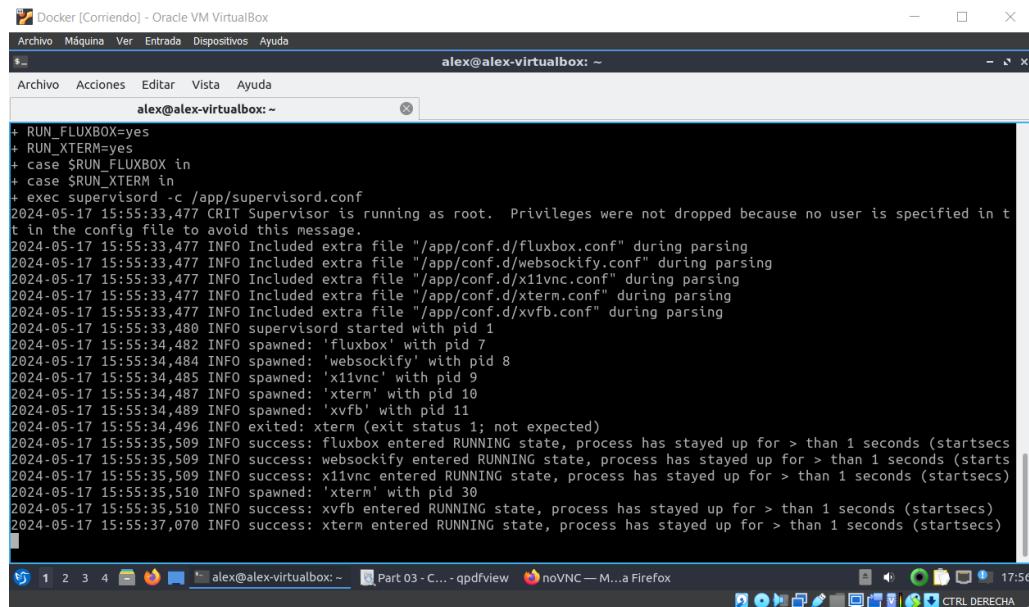
Para poder poner en marcha un servicio de VNC en conjunto con un cliente de NoVNC , necesitamos usar el comando "docker run --rm -it -p 8080:8080 theasp/novnc" , para poder crear el contenedor.



```
Docker [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
$ alex@alex-virtualbox: ~
Archivo Acciones Editar Vista Ayuda
alex@alex-virtualbox: ~

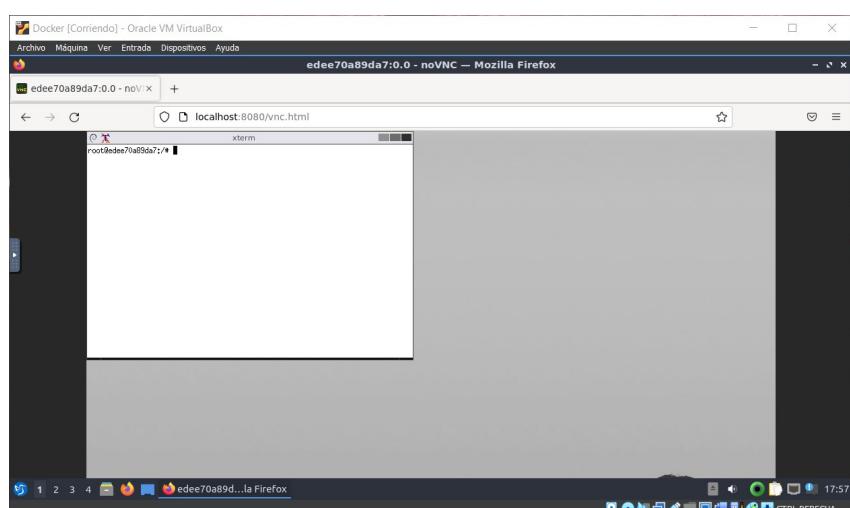
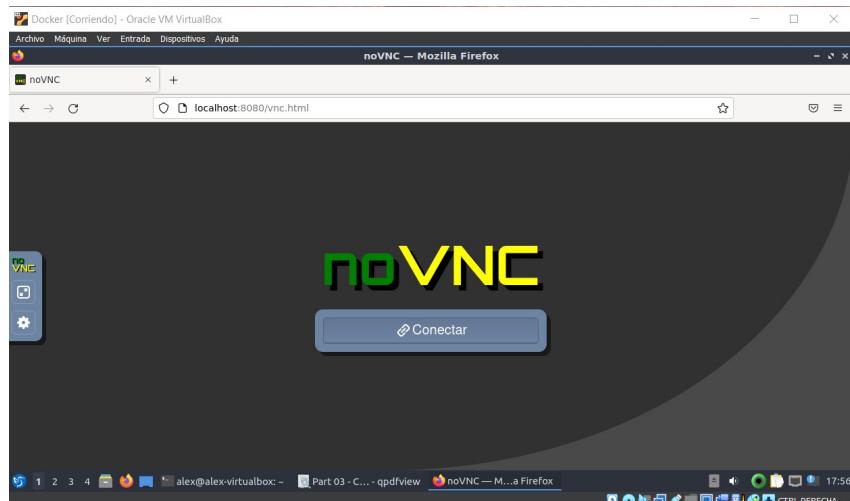
condescending_kilby
alex@alex-virtualbox:~$ docker run --rm -it -p 8080:8080 theaso/novnc^C
alex@alex-virtualbox:~$ docker run --rm -it -p 8080:8080 theaso/novnc
Unable to find image 'theaso/novnc:latest' locally
docker: Error response from daemon: pull access denied for theaso/novnc, repository does not exist or may re
quire 'docker login': denied: requested access to the resource is denied.
See 'docker run --help'.
alex@alex-virtualbox:~$ sudo docker run --rm -it -p 8080:8080 theaso/novnc
Unable to find image 'theaso/novnc:latest' locally
docker: Error response from daemon: pull access denied for theaso/novnc, repository does not exist or may re
quire 'docker login': denied: requested access to the resource is denied.
See 'docker run --help'.
alex@alex-virtualbox:~$ docker run --rm -it -p 8080:8080 theasp/novnc
Unable to find image 'theasp/novnc:latest' locally
latest: Pulling from theasp/novnc
3d53ef4019fc: Downloading [=====] 52.33MB/55.1MB
4b754972ea25: Downloading [=====] 32.34MB/189MB
2dd36285fee: Download complete
 1 2 3 4 alex@alex-virtualbox: ~ 17:47
CTRL DERECHA
```

Una vez se descarga, se pone en marcha.



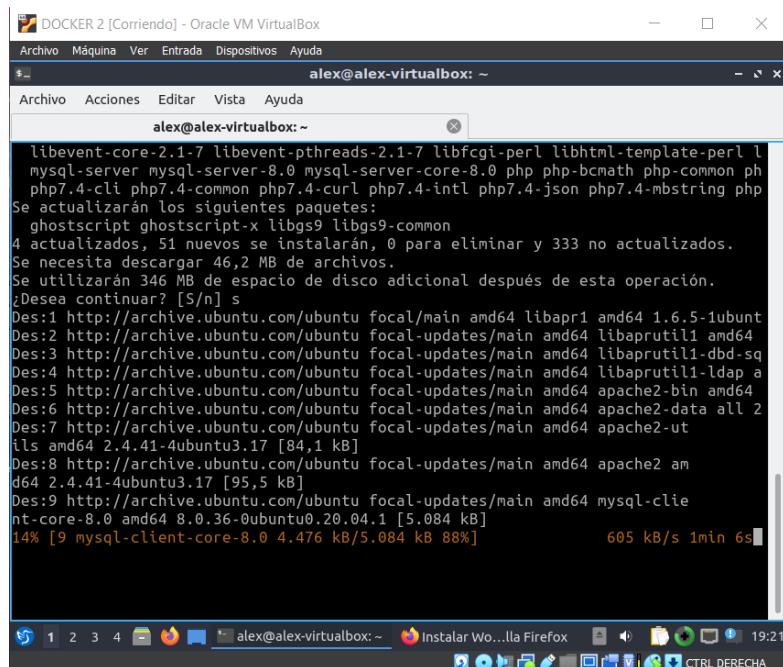
```
+ RUN_FLUXBOX=yes
+ RUN_XTERM=yes
+ case $RUN_FLUXBOX in
+ case $RUN_XTERM in
+ exec supervisord -c /app/supervisord.conf
2024-05-17 15:55:33,477 CRIT Supervisor is running as root. Privileges were not dropped because no user is specified in t
t in the config file to avoid this message.
2024-05-17 15:55:33,477 INFO Included extra file "/app/conf.d/fluxbox.conf" during parsing
2024-05-17 15:55:33,477 INFO Included extra file "/app/conf.d/websockify.conf" during parsing
2024-05-17 15:55:33,477 INFO Included extra file "/app/conf.d/x11vnc.conf" during parsing
2024-05-17 15:55:33,477 INFO Included extra file "/app/conf.d/xterm.conf" during parsing
2024-05-17 15:55:33,477 INFO Included extra file "/app/conf.d/xvfb.conf" during parsing
2024-05-17 15:55:33,480 INFO supervisor started with pid 1
2024-05-17 15:55:34,482 INFO spawned: 'fluxbox' with pid 7
2024-05-17 15:55:34,484 INFO spawned: 'websockify' with pid 8
2024-05-17 15:55:34,485 INFO spawned: 'x11vnc' with pid 9
2024-05-17 15:55:34,487 INFO spawned: 'xterm' with pid 10
2024-05-17 15:55:34,489 INFO spawned: 'xvfb' with pid 11
2024-05-17 15:55:34,496 INFO exited: xterm (exit status 1; not expected)
2024-05-17 15:55:35,509 INFO success: fluxbox entered RUNNING state, process has stayed up for > than 1 seconds (startsecs)
2024-05-17 15:55:35,509 INFO success: websockify entered RUNNING state, process has stayed up for > than 1 seconds (startsecs)
2024-05-17 15:55:35,509 INFO success: x11vnc entered RUNNING state, process has stayed up for > than 1 seconds (startsecs)
2024-05-17 15:55:35,510 INFO spawned: 'xterm' with pid 30
2024-05-17 15:55:35,510 INFO success: xvfb entered RUNNING state, process has stayed up for > than 1 seconds (startsecs)
2024-05-17 15:55:37,070 INFO success: xterm entered RUNNING state, process has stayed up for > than 1 seconds (startsecs)
```

Y después lo único que tenemos que hacer es dejar la consola abierta y copiar el url "http://localhost:8080/vnc.html" que es el que se encarga de iniciar el cliente.



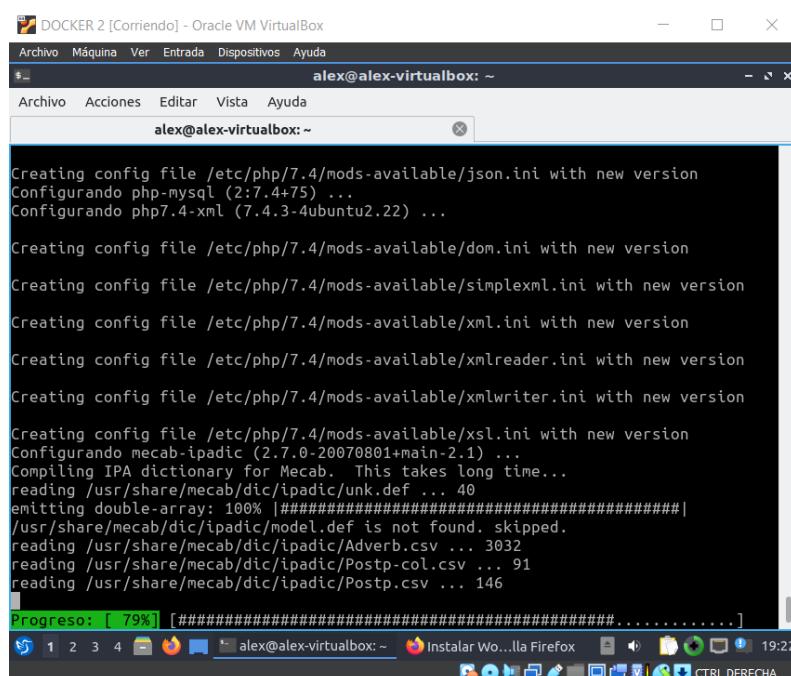
### 3. CAPTURA FINAL DEL NAVEGADOR DONDE SE OBSERVA QUE WORDPRESS ESTÁ FUNCIONANDO.

Para instalar WordPress lo primero que hay que hacer es instalar las dependencias. Con el comando: "sudo apt install apache2 \ghostscript \libapache2-mod-php \mysql-server \php \php-bcmath \php-curl \php-imagick \php-intl \php-json \php-mbstring \php-mysql \php-xml \php-zip"



```
DOCKER 2 [Corriendo] - Oracle VM VirtualBox
alex@alex-virtualbox: ~
```

```
libevent-core-2.1.7 libevent-pthreads-2.1.7 libfcgi-perl libhtml-template-perl lib
mysql-server mysql-server-8.0 mysql-server-core-8.0 php php-bcmath php-common ph
php7.4-cli php7.4-common php7.4-curl php7.4-intl php7.4-json php7.4-mbstring php
Se actualizan los siguientes paquetes:
ghostscript ghostscript-x libgs9 libgs9-common
4 actualizados, 51 nuevos se instalarán, 0 para eliminar y 333 no actualizados.
Se necesita descargar 46,2 MB de archivos.
Se utilizarán 346 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://archive.ubuntu.com/ubuntu focal/main amd64 libapr1 amd64 1.6.5-1ubunt
Des:2 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 libaprutil1 amd64
Des:3 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 libaprutil1-db-2
Des:4 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 libaprutil1-ldap a
Des:5 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 apache2-bin amd64
Des:6 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 apache2-data all 2
Des:7 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 apache2-ut
ils amd64 2.4.41-4ubuntu3.17 [84,1 kB]
Des:8 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 apache2 am
d64 2.4.41-4ubuntu3.17 [95,5 kB]
Des:9 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-clie
nt-core-8.0 amd64 8.0.36-0ubuntu0.20.04.1 [5,084 kB]
14% [9 mysql-client-core-8.0 4.476 kB/5.084 kB 88%] 605 kB/s 1min 6s
```



```
DOCKER 2 [Corriendo] - Oracle VM VirtualBox
alex@alex-virtualbox: ~
```

```
Creating config file /etc/php/7.4/mods-available/json.ini with new version
Configurando php-mysql (2:7.4+75) ...
Configurando php7.4-xml (7.4.3-4ubuntu2.22) ...

Creating config file /etc/php/7.4/mods-available/dom.ini with new version
Creating config file /etc/php/7.4/mods-available/simplexml.ini with new version
Creating config file /etc/php/7.4/mods-available/xml.ini with new version
Creating config file /etc/php/7.4/mods-available/xmlreader.ini with new version
Creating config file /etc/php/7.4/mods-available/xmlwriter.ini with new version

Creating config file /etc/php/7.4/mods-available/xsl.ini with new version
Configurando mecab-ipadic (2.7.0-20070801+main-2.1) ...
Compiling IPA dictionary for MeCab. This takes long time...
reading /usr/share/mecab/dic/ipadic/unk.def ... 40
emitting double-array: 100% |#####
/usr/share/mecab/dic/ipadic/model.def is not found. skipped.
reading /usr/share/mecab/dic/ipadic/Adverb.csv ... 3032
reading /usr/share/mecab/dic/ipadic/Postp-col.csv ... 91
reading /usr/share/mecab/dic/ipadic/Postp.csv ... 146
[Progresso: [ 79% ] #####.....]
```

Descargar los archivos de WordPress, crear primero un directorio de instalación. Con los comandos:

```
"sudo mkdir -p /srv/www
```

```
sudo chown www-data: /srv/www
```

```
curl https://wordpress.org/latest.tar.gz | sudo -u www-data tar zx -C /srv/www"
```

DOCKER 2 [Corriendo] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

alex@alex-virtualbox: ~

Archivo Acciones Editar Vista Ayuda

alex@alex-virtualbox: ~

```
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean.service → /lib/systemd/system/apache-htcacheclean.service.
Configurando php7.4 (7.4.3-4ubuntu2.22) ...
Configurando libapache2-mod-php (2:7.4+75) ...
Configurando php (2:7.4+75) ...
Procesando disparadores para ufw (0.36-6ubuntu1) ...
Procesando disparadores para systemd (245.4-4ubuntu3.17) ...
Procesando disparadores para man-db (2.9.1-1) ...
Procesando disparadores para fontconfig (2.13.1-2ubuntu3) ...
Procesando disparadores para libc-bin (2.31-0ubuntu9.9) ...
Procesando disparadores para php7.4-cli (7.4.3-4ubuntu2.22) ...
Procesando disparadores para libapache2-mod-php7.4 (7.4.3-4ubuntu2.22) ...
alex@alex-virtualbox:~$ sudo mkdir -p /srv/www
alex@alex-virtualbox:~$ sudo chown www-data: /srv/www
alex@alex-virtualbox:~$ curl https://wordpress.org/latest.tar.gz | sudo -u www-data tar zx -C /srv/www
% Total    % Received % Xferd  Average Speed   Time   Time     Time Current
          Dload  Upload   Total Spent   Left Speed
16 23.5M  16 3967k    0      0   692k      0  0:00:34  0:00:05  0:00:29  660k
19 23.5M  19 4800k    0      0   713k      0  0:00:33  0:00:06  0:00:27
23 23.5M  23 5679k    0      0   735k      0  0:00:32  0:00:07  0:00:25
100 23.5M 100 23.5M   0      0   878k      0  0:00:27  0:00:27  ---- 1267k
alex@alex-virtualbox:~$
```

Nos damos de alta en Apache, e instalamos nano.

Docker 2 [Corriendo] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

alex@alex-virtualbox: ~

alex@alex-virtualbox: ~

```
continents-cities-zh_CN.mo continents-cities-zh_TW.mo cs_CZ.mo cy.mo da_DK.mo de_DE.mo el.mo en_CA.mo eo.mo es_CL.mo es_ES.mo es_EE.mo et.mo eu.mo fa_IR.mo fi.mo fr_FR.mo gd.mo gl_ES.mo he_IL.mo hr.mo hu_HU.mo id_ID.mo it_IT.mo ja.mo jv_ID.mo ka_KR.mo ko_KR.mo lv_mo_mk_MK.mo ms_MY.mo my_MM.mo nb_NO.mo nl_NL.mo pl_PL.mo pt_BR.mo pt_PT.mo ro_RO.mo ru_RU.mo ru_UA.mo si_LK.mo sk_SK.mo sl_SI.mo sq.mo sr_RS.mo su_ID.mo sv_SE.mo sw_tw_LK.mo th.mo tr_TR.mo ug_CN.mo uk.mo ur.mo vi.mo zh_CN.mo zh_HK.mo zh_TW.mo
```

Configurando wordpress-l10n (5.3.2+dfsg1-1ubuntu1) ...

Procesando disparadores para mime-support (3.64ubuntu1) ...

Procesando disparadores para man-db (2.9.1-1) ...

Procesando disparadores para libapache2-mod-php7.4 (7.4.3-4ubuntu2.22) ...

Procesando disparadores para php7.4-cli (7.4.3-4ubuntu2.22) ...

```
alex@alex-virtualbox:~$ service apache2 start
alex@alex-virtualbox:~$ apt install nano
```

E: No se pudo abrir el fichero de bloqueo «/var/lib/dpkg/lock-frontend» - open (13 : Permiso denegado)

E: No se pudo obtener el bloqueo de la interfaz dpkg (/var/lib/dpkg/lock-frontend). ¿Es usted superusuario?

```
alex@alex-virtualbox:~$ sudo apt install nano
```

Leyendo lista de paquetes... Hecho

Creando árbol de dependencias

Leyendo la información de estado... Hecho

nano ya está en su versión más reciente (4.8-1ubuntu1).

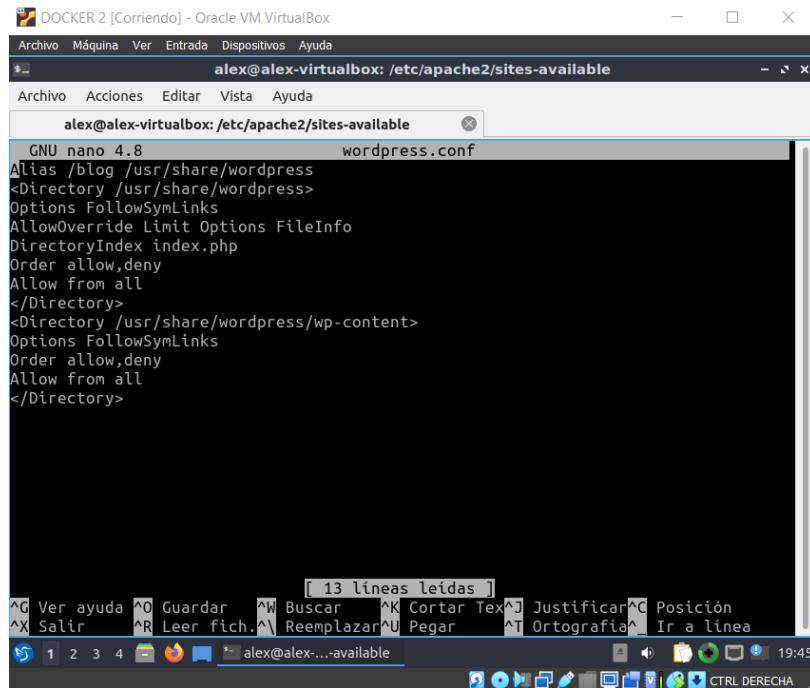
fijado nano como instalado manualmente.

0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 333 no actualizados.

```
alex@alex-virtualbox:~$
```

Buscamos el archivo "/etc/apache2/sites-available/wordpress.conf" y le escribimos el contenido que queremos mediante:

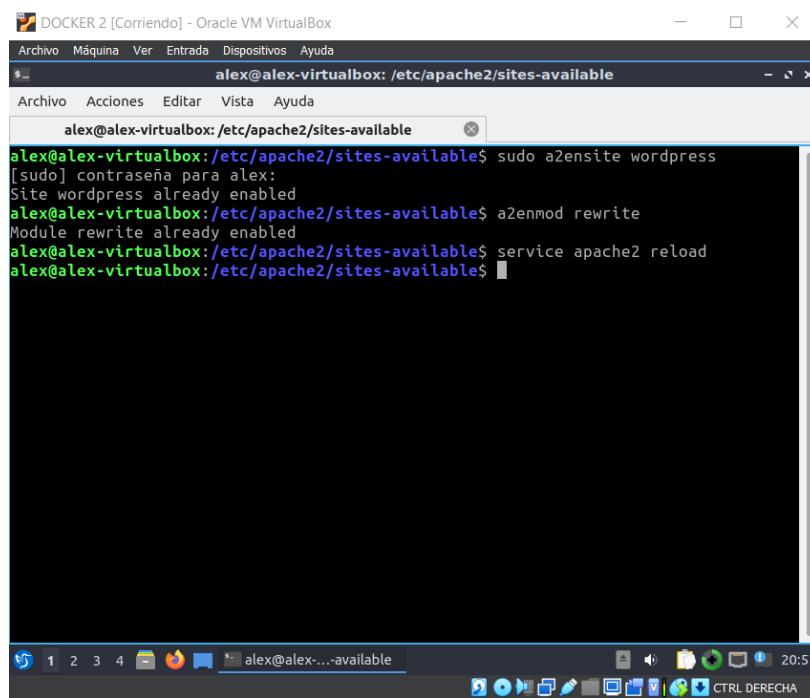
"sudo nano /wordpress.conf "



The screenshot shows a terminal window titled "alex@alex-virtualbox: /etc/apache2/sites-available". The command "sudo nano /etc/apache2/sites-available" has been run, and the file "wordpress.conf" is open in the nano editor. The content of the file is as follows:

```
GNU nano 4.8          wordpress.conf
Alias /blog /usr/share/wordpress
<Directory /usr/share/wordpress>
Options FollowSymLinks
AllowOverride Limit Options FileInfo
DirectoryIndex index.php
Order allow,deny
Allow from all
</Directory>
<Directory /usr/share/wordpress/wp-content>
Options FollowSymLinks
Order allow,deny
Allow from all
</Directory>
```

The terminal window also displays the message "[ 13 líneas leidas ]" (13 lines read). At the bottom of the terminal, there is a menu bar with options like Ver ayuda, Guardar, Buscar, Cortar Texto, Justificar, Posición, Salir, Leer fich., Reemplazar, Pegar, Ortografía, Ir a línea, and a status bar showing the time as 19:45.

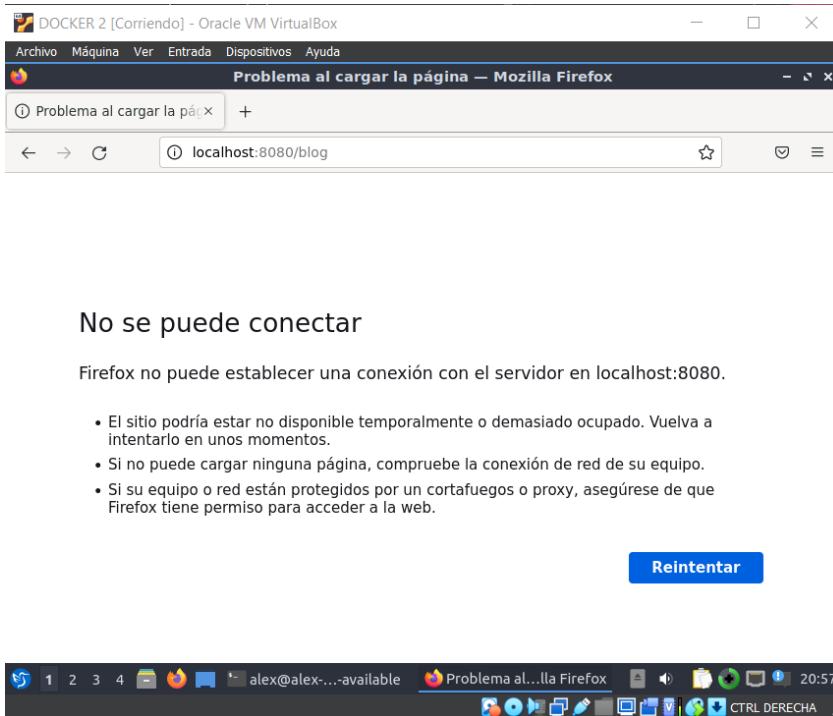


The screenshot shows a terminal window titled "alex@alex-virtualbox: /etc/apache2/sites-available". The user runs several commands to manage the Apache site:

```
alex@alex-virtualbox:/etc/apache2/sites-available$ sudo a2ensite wordpress
[sudo] contraseña para alex:
Site wordpress already enabled
alex@alex-virtualbox:/etc/apache2/sites-available$ a2enmod rewrite
Module rewrite already enabled
alex@alex-virtualbox:/etc/apache2/sites-available$ service apache2 reload
alex@alex-virtualbox:/etc/apache2/sites-available$
```

The terminal window has a menu bar with Archivo, Máquina, Ver, Entrada, Dispositivos, Ayuda, and a title bar showing "alex@alex-virtualbox: /etc/apache2/sites-available". The status bar at the bottom shows the time as 20:57.

Ya podemos verificar el enlace:"<http://localhost:8080/blog>" pero veremos que da un error porque todavia no hemos configurado el fichero.



Ahora vamos a preparar el servicio de mySQL y lo tenemos que poner en marcha con el comando:"`service mysql start`"

A screenshot of a terminal window titled "alex@alex-virtualbox: /etc/apache2/sites-available". The terminal shows the following command execution:

```
alex@alex-virtualbox:~/etc/apache2/sites-available$ sudo a2ensite wordpress
[sudo] contraseña para alex:
Site wordpress already enabled
alex@alex-virtualbox:~/etc/apache2/sites-available$ a2enmod rewrite
Module rewrite already enabled
alex@alex-virtualbox:~/etc/apache2/sites-available$ service apache2 reload
alex@alex-virtualbox:~/etc/apache2/sites-available$ service mysql start
alex@alex-virtualbox:~/etc/apache2/sites-available$
```

The terminal is part of a desktop environment, as indicated by the taskbar at the bottom.

Despues tenemos ejecutar el comando para generar una password de root de mySQL Server de forma segura , el comando es:"mysql\_secure\_installation "

```
alex@alex-virtualbox: ~
$ mysql_secure_installation

-- MySQL secure installation -- Version 1.52.0

Normally, root should only be allowed to connect from 'localhost'. This ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) : y
Success.

By default, MySQL comes with a database named 'test' that anyone can access. This is also intended only for testing, and should be removed before moving into a production environment.

Remove test database and access to it? (Press y|Y for Yes, any other key for No) :
n

... skipping.

Reloading the privilege tables will ensure that all changes made so far will take effect immediately.

Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y
Success.

All done!
alex@alex-virtualbox:~$
```

Después accedemos a la base de datos del cliente de MySQL con el siguiente comando:"mysql -u root -p "-

```
alex@alex-virtualbox: ~
$ mysql -u root -p
Enter password:
ERROR 1698 (28000): Access denied for user 'root'@'localhost'
alex@alex-virtualbox: ~$ sudo mysql -u root -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 19
Server version: 8.0.36-Ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

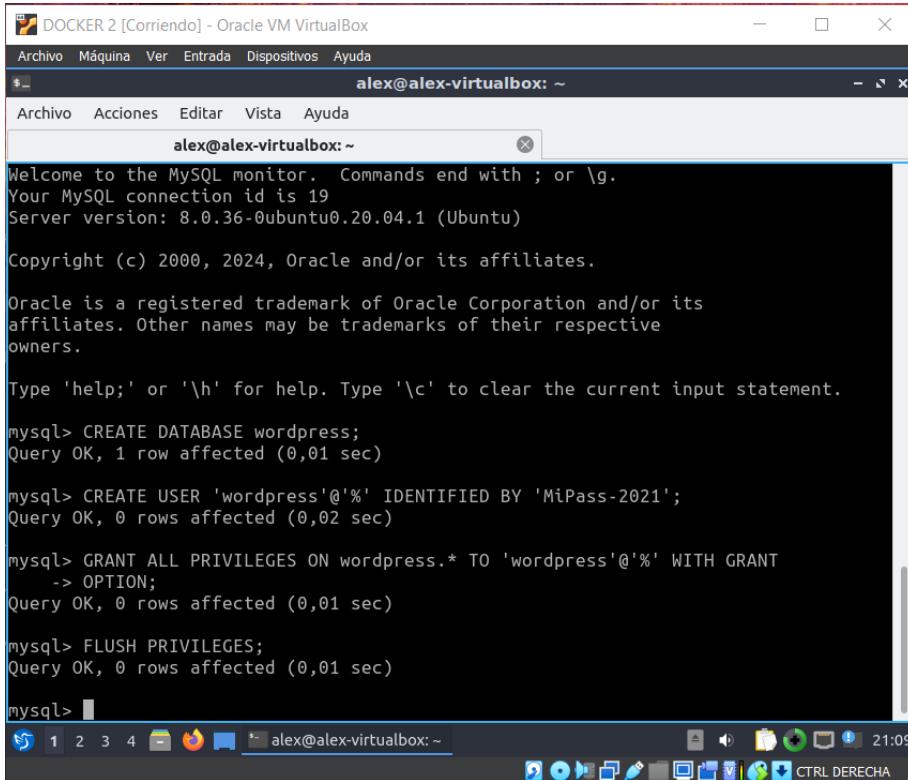
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql>
```

Una vez dentro ya podemos darle indicaciones a MySQL.

Primero, vamos a crear una base de datos.

Segundo, creamos el usuario "wordpress" con contraseña "MiPass-2021" y le damos permisos.

Tercero, propagamos los permisos.



```
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 19
Server version: 8.0.36-Ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE wordpress;
Query OK, 1 row affected (0,01 sec)

mysql> CREATE USER 'wordpress'@'%' IDENTIFIED BY 'MiPass-2021';
Query OK, 0 rows affected (0,02 sec)

mysql> GRANT ALL PRIVILEGES ON wordpress.* TO 'wordpress'@'%' WITH GRANT
    -> OPTION;
Query OK, 0 rows affected (0,01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,01 sec)

mysql>
```

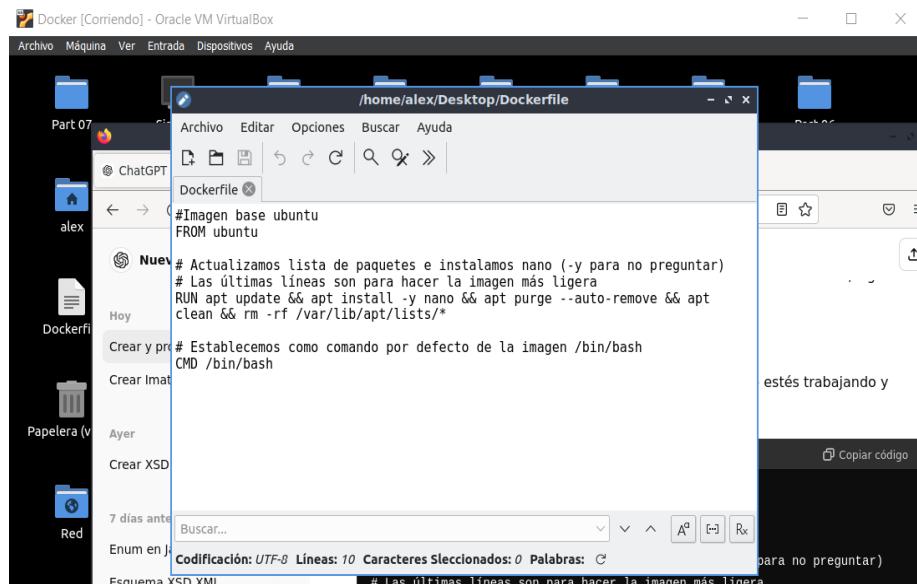
Vamos a configurar WordPress , cogemos el fichero "/etc/wordpress/config-localhost.php" y lo editamos.

**4. En esta unidad has de realizar el primer caso práctico (imagen con nano) y subirlo a Docker Hub. Después, demuestra con alguna captura de las partes finales, que se han realizado en el 3 o 4 caso práctico.**

## PARTE 1: Creant imatge Ubuntu amb nano

### 1. Preparar el Dockerfile

Crea un archivo llamado Dockerfile en el directorio donde estés trabajando y copia el siguiente contenido en él:

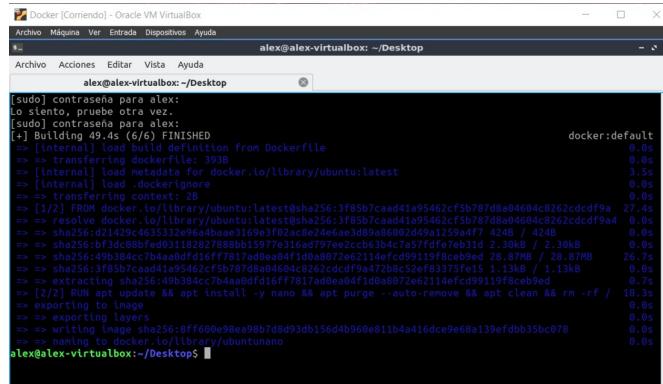


```
#Imagen base ubuntu
FROM ubuntu
```

### 2. Crear la imagen de Docker

Abre una terminal y navega al directorio donde se encuentra tu Dockerfile. Luego ejecuta el siguiente comando para crear la imagen:

Con el comando :"docker build -t ubuntunano ./"



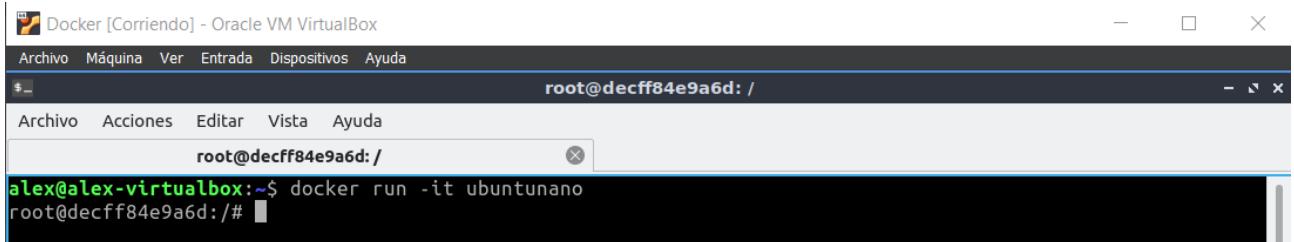
```
[sudo] contraseña para alex:
alex@alex-virtualbox:~/Desktop
```

```
[sudo] contraseña para alex:
[*] Building 49.ds (6/6) FINISHED
==> [internal] load build definition from Dockerfile
==> [internal] transferring dockerfile: 393B
==> [internal] load metadata for docker.io/library/ubuntu:latest
==> [internal] load .dockerrigmore
==> [internal] load context
==> [internal] resolve context
==> [1/2] FROM docker.io/library/ubuntu:latest@sha256:3f85b7caad41a95462cf5b787d8a04604c8262cdcdf9a
==> [2/2] RUN apt update && apt install -y nano && apt purge --auto-remove && apt clean && rm -rf /var/lib/apt/lists/*
# las últimas líneas son para hacer la imagen más ligera
alex@alex-virtualbox:~/Desktop
```

### 3. Probar la imagen

Una vez creada la imagen, puedes crear y ejecutar un contenedor basado en ella con el siguiente comando:

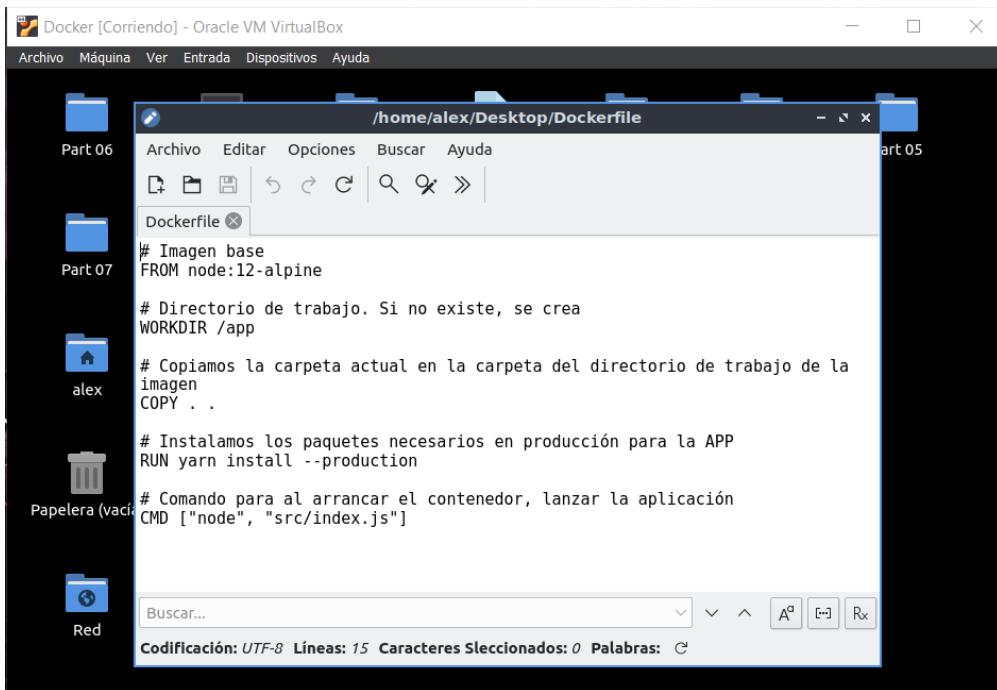
```
docker run -it ubuntunano
```



## PARTE 2: Creant imatge amb APP en Node

### 1. Preparar el Dockerfile

Crear un archivo llamado Dockerfile en el directorio del proyecto con el siguiente contenido:



### 2. Crear la Imagen Docker

Con el Dockerfile preparado, abrir una terminal y navegar al directorio donde se encuentra el Dockerfile. Ejecutar el siguiente comando para crear la imagen Docker:

```

Docker [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
alex@alex-virtualbox: ~/Desktop
Archivo Acciones Editar Vista Ayuda
alex@alex-virtualbox: ~/Desktop
alex@alex-virtualbox: ~/Desktop
=> [internal] load .dockerrigore          0.0s
=> => transferring context: 28          0.0s
=> [1/4] FROM docker.io/library/node:12-alpine@sha256:d4b15b3d48f42059a15bd659be60 19.7s
=> => resolve docker.io/library/node:12-alpine@sha256:d4b15b3d48f42059a15bd659be60a 0.0s
=> => sha256:4517380049fc3c9aacceae7764fcf3500354b0ac8a47e4af835b5b 1.16kB / 1.16kB 0.0s
=> => sha256:bb6d28039b8cec9aa8d9032f9aa640a792a60c2cb1644691627bf0 6.58kB / 6.58kB 0.0s
=> => sha256:df9b9388f04ad6279a7410b85cedfdb2208c0a003da7ab5613af7 2.81MB / 2.81MB 4.0s
=> => sha256:3bf6d738020517f4622814e8ae7cab4e4f872c 24.91MB / 24.91MB 18.2s
=> => sha256:7939e601ee5e4737cf7fdb6d1dfe31ca4c2697109290462f694710 2.36MB / 2.36MB 5.7s
=> => sha256:d4b15b3d48f42059a15bd659be60afe21762aae9d6cbea6f124440 1.43kB / 1.43kB 0.0s
=> => extracting sha256:df9b9388f04ad6279a7410b85cedfdb2208c0a003da7ab5613af710791 0.1s
=> => sha256:31f0fb9de071269230cb0f786012ae4e81d26e489b1fe922e57b5201e6 451B / 451B 4.4s
=> => extracting sha256:3bf6d738020517f4622814e8c21db4b4aaa78ae7cab4e4f872c11696886 1.3s
=> => extracting sha256:7939e601ee5e4737cf7fdb6d1dfe31ca4c2697109290462f69471076145 0.1s
=> => extracting sha256:31f0fb9de071269230cb0f786012ae4e81d26e489b1fe922e57b5201e6b 0.0s
=> [internal] load build context          0.1s
=> => transferring context: 8.58MB      0.1s
=> [2/4] WORKDIR /app                   0.2s
=> [3/4] COPY . .                      0.0s
=> [4/4] RUN yarn install --production 0.6s
=> => exporting to image               0.1s
=> => exporting layers                0.1s
=> => writing image sha256:a8a403cb0303aca42fab14d97b178c0c73a4d91b1fe84e1e9b4e2011 0.0s
=> => naming to docker.io/library/sampledocker 0.0s
alex@alex-virtualbox:~/Desktop$ 
```

### 3. Probar la Imagen Docker

Para ejecutar la aplicación en un contenedor Docker y exponerla en el puerto 3000 de la máquina host, ejecutar el siguiente comando:

```

Docker [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
alex@alex-virtualbox: ~/Desktop
Archivo Acciones Editar Vista Ayuda
alex@alex-virtualbox: ~/Desktop
alex@alex-virtualbox: ~/Desktop
=> [1/4] FROM docker.io/library/node:12-alpine@sha256:d4b15b3d48f42059a15bd659be60 19.7s
=> => resolve docker.io/library/node:12-alpine@sha256:d4b15b3d48f42059a15bd659be60a 0.0s
=> => sha256:4517380049fc3c9aacceae7764fcf3500354b0ac8a47e4af835b5b 1.16kB / 1.16kB 0.0s
=> => sha256:bb6d28039b8cec9aa8d9032f9aa640a792a60c2cb1644691627bf0 6.58kB / 6.58kB 0.0s
=> => sha256:df9b9388f04ad6279a7410b85cedfdb2208c0a003da7ab5613af7 2.81MB / 2.81MB 4.0s
=> => sha256:3bf6d738020517f4622814e8c21db4b4aaa78ae7cab4e4f872c 24.91MB / 24.91MB 18.2s
=> => sha256:7939e601ee5e4737cf7fdb6d1dfe31ca4c2697109290462f694710 2.36MB / 2.36MB 5.7s
=> => sha256:d4b15b3d48f42059a15bd659be60afe21762aae9d6cbea6f124440 1.43kB / 1.43kB 0.0s
=> => extracting sha256:df9b9388f04ad6279a7410b85cedfdb2208c0a003da7ab5613af710791 0.1s
=> => sha256:31f0fb9de071269230cb0f786012ae4e81d26e489b1fe922e57b5201e6 451B / 451B 4.4s
=> => extracting sha256:3bf6d738020517f4622814e8c21db4b4aaa78ae7cab4e4f872c11696886 1.3s
=> => extracting sha256:7939e601ee5e4737cf7fdb6d1dfe31ca4c2697109290462f69471076145 0.1s
=> => extracting sha256:31f0fb9de071269230cb0f786012ae4e81d26e489b1fe922e57b5201e6b 0.0s
=> [internal] load build context          0.1s
=> => transferring context: 8.58MB      0.1s
=> [2/4] WORKDIR /app                   0.2s
=> [3/4] COPY . .                      0.0s
=> [4/4] RUN yarn install --production 0.6s
=> => exporting to image               0.1s
=> => exporting layers                0.1s
=> => writing image sha256:a8a403cb0303aca42fab14d97b178c0c73a4d91b1fe84e1e9b4e2011 0.0s
=> => naming to docker.io/library/sampledocker 0.0s
alex@alex-virtualbox:~/Desktop$ docker run -dp 3000:3000 sampledocker
53668d66297b4da166e5538e5cbe5b15d880f81d1541c8d765e5b95e1459cd7a
alex@alex-virtualbox:~/Desktop$ 
```

Una vez ejecutado este comando, la aplicación debería estar disponible en <http://localhost:3000>.



## No se puede conectar

Firefox no puede establecer una conexión con el servidor en localhost:3000.

- El sitio podría estar no disponible temporalmente o demasiado ocupado. Vuelva a intentarlo en unos momentos.
- Si no puede cargar ninguna página, compruebe la conexión de red de su equipo.
- Si su equipo o red están protegidos por un cortafuegos o proxy, asegúrese de que Firefox tiene permiso para acceder a la web.

[Reintentar](#)

## PARTE 3:Apache2 amb PHP usant Alpine

### 1. Preparar el Dockerfile

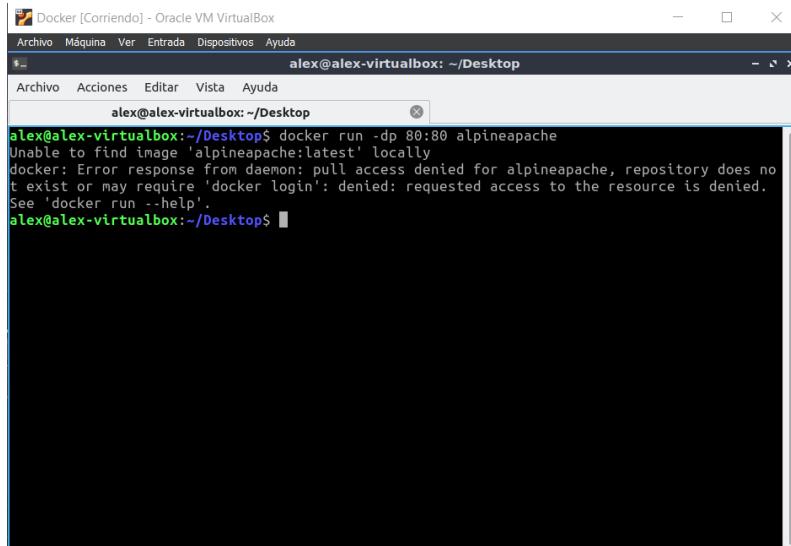
Primero, debemos crear un archivo llamado Dockerfile con el siguiente contenido:

```
FROM alpine
MAINTAINER Sergi <sergi.profesor@gmail.com>
#Actualizamos e instalamos paquetes con APK para Alpine
RUN apk update && apk add apache2 php php-apache2 opensrc tar
#Copiamos script para lanzar Apache 2
ADD ./start.sh /start.sh
#Descargamos un ejemplo de <?php phpinfo(); ?> por enseñar como bajar algo de Internet
#Podria haber sido simplemente
#RUN echo "<?php phpinfo(); ?>" > /var/www/localhost/htdocs/index.php
ADD https://gist.githubusercontent.com/SyntaxC4/5648247/raw/9427715663879c309f2e36e19b7f378ba7364907/info.php /var/www/localhost/htdocs/index.php

# Si quisieramos algo como Wordpress haríamos
#ADD http://wordpress.org/latest.tar.gz /var/www/localhost/htdocs/
wordpress.tar.gz
#DLM tar xvzf /var/www/localhost/htdocs/wordpress.tar.gz -f /var/www/localhost/htdocs/wordpress
Buscar...
Codificación: UTF-8 Líneas: 29 Caracteres Seleccionados: 0 Palabras: 0
```

## 2. Crear el script start.sh

El siguiente paso es crear un script start.sh que contendrá los comandos necesarios para iniciar Apache. Este archivo debe estar en el mismo directorio que el Dockerfile.



A screenshot of a terminal window titled "Docker [Corriendo] - Oracle VM VirtualBox". The window has a menu bar with "Archivo", "Máquina", "Ver", "Entrada", "Dispositivos", and "Ayuda". Below the menu is a toolbar with "Archivo", "Acciones", "Editar", "Vista", and "Ayuda". The main area shows a command-line session:

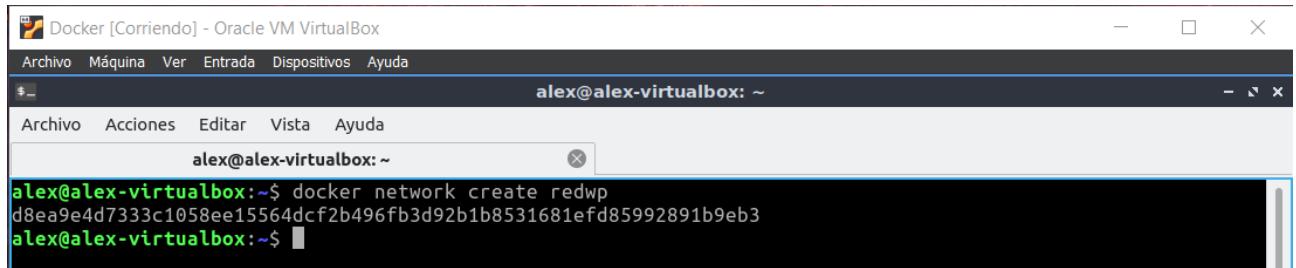
```
alex@alex-virtualbox:~/Desktop$ docker run -dp 80:80 alpineapache
Unable to find image 'alpineapache:latest' locally
docker: Error response from daemon: pull access denied for alpineapache, repository does not exist or may require 'docker login': denied: requested access to the resource is denied.
See 'docker run --help'.
alex@alex-virtualbox:~/Desktop$
```

Como me esta dando todo el rato el mismo error y no se solucionarlo, no puedo seguir con la practica.

**5. Este ejercicio implica la configuración de un CMS Wordpress utilizando Docker. Crearemos una red Docker para conectar dos contenedores: uno para MySQL y otro para Wordpress.**

### Paso 1: Crear la Red

Primero, creamos una red llamada "redwp" que utilizarán los contenedores:



A screenshot of a terminal window titled "Docker [Corriendo] - Oracle VM VirtualBox". The window has a menu bar with "Archivo", "Máquina", "Ver", "Entrada", "Dispositivos", and "Ayuda". Below the menu is a toolbar with "Archivo", "Acciones", "Editar", "Vista", and "Ayuda". The main area shows a command-line session:

```
alex@alex-virtualbox:~$ docker network create redwp
d8ea9e4d7333c1058ee15564dcf2b496fb3d92b1b8531681efd85992891b9eb3
alex@alex-virtualbox:~$
```

## Paso 2: Crear el Contenedor MySQL

A continuación, creamos un contenedor para MySQL. Utilizaremos la imagen de MySQL 5.6 y configuraremos varias variables de entorno:

```
name nuestromysql: Nombra el contenedor nuestromysql.  
--network redwp: Conecta el contenedor a la red redwp.  
-v /home/alex/mysqldata:/var/lib/mysql: Mapea el directorio local  
/home/alex/mysqldata al directorio /var/lib/mysql dentro del  
contenedor.  
-e MYSQL_ROOT_PASSWORD=cefireroot: Establece la contraseña del  
usuario root de MySQL.  
-e MYSQL_USER=cefireuser -e MYSQL_PASSWORD=cefirepass: Crea un  
usuario de MySQL llamado cefireuser con la contraseña cefirepass.  
-e MYSQL_DATABASE=cefiredb: Crea una base de datos llamada  
cefiredb.  
-d mysql:5.6: Usa la imagen mysql:5.6.
```

## Paso 3: Crear el Contenedor Wordpress

Ahora creamos un contenedor para Wordpress usando la imagen de Wordpress oficial:

```
--name nuestrowp: Nombra el contenedor nuestrowp.  
--network redwp: Conecta el contenedor a la red redwp.  
-p 8080:80: Mapea el puerto 80 del contenedor al puerto 8080 del  
host.  
-d wordpress: Usa la imagen wordpress.
```

A screenshot of a terminal window titled "Docker [Corriendo] - Oracle VM VirtualBox". The window has a dark theme. At the top, there's a menu bar with "Archivo", "Máquina", "Ver", "Entrada", "Dispositivos", and "Ayuda". Below the menu is a toolbar with "alex@alex-virtualbox: ~". The main area shows a command-line session:

```
alex@alex-virtualbox:~$ docker network create redwp
d8ea9e4d7333c1058ee15564dcf2b496fb3d92b1b8531681efd85992891b9eb3
alex@alex-virtualbox:~$ docker run --name nuestromysql --network redwp -v /home/sergi/mysqldata:/var/lib/mysql -e MYSQL_ROOT_PASSWORD=cefireroot -e MYSQL_USER=cefireuser -e MYSQL_PASSWORD=cefirepass -e MYSQL_DATABASE=cefiiredb -d mysql:5.6
Unable to find image 'mysql:5.6' locally
5.6: Pulling from library/mysql
35b2232c987e: Pull complete
fc55c00e48f2: Pull complete
0030405130e3: Pull complete
e1fef7f6a8d1: Pull complete
1c76272398bb: Pull complete
f57e698171b6: Pull complete
f5b825b269c0: Pull complete
dcbaaf686073: Pull complete
27bbfeb886d1: Pull complete
6f70cc868145: Pull complete
1f6637f4600d: Pull complete
Digest: sha256:20575ecebe6216036d25dab5903808211f1e9ba63dc7825ac20cb975e34cfcae
Status: Downloaded newer image for mysql:5.6
ae49ca08adbfe47fd6b8dad0a629cf9ea3b40c83966d78d28ec5f2e21a0fcc49
alex@alex-virtualbox:~$
```

## Paso 4: Instalar Wordpress

Accede a <http://localhost:8080> en tu navegador para iniciar la instalación de Wordpress. Usa las siguientes configuraciones:

- **Nombre de la base de datos:** cefiiredb
- **Nombre de usuario:** cefireuser
- **Contraseña:** cefirepass
- **Servidor de la base de datos:** nuestromysql
- **Prefijo de tabla:** (deja el valor predeterminado)

Sigue los pasos de instalación para completar la configuración de Wordpress.

Docker [Corriendo] - Oracle VM VirtualBox

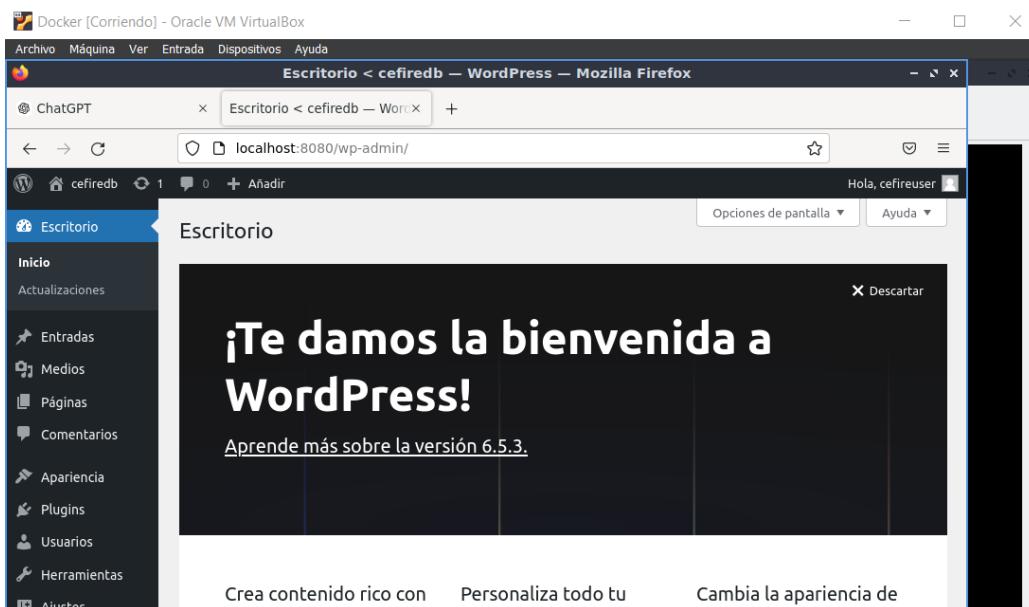
Archivo Máquina Ver Entrada Dispositivos Ayuda

alex@alex-virtualbox: ~

alex@alex-virtualbox:~\$ docker run --name nuestrowp --network redwp -p 8080:80 -d wordpress

```
Unable to find image 'wordpress:latest' locally
latest: Pulling from library/wordpress
09f376ebb190: Pull complete
76afc86551: Pull complete
ceed4541c527: Pull complete
9ec84be954b0: Pull complete
ff0e278869f9: Pull complete
1693466e4cc6: Pull complete
57c8d94a4882: Pull complete
43af3fe8136a: Pull complete
ddef75e08a5d: Pull complete
a4ba0bdbbf0a: Pull complete
29e89bcc69515: Pull complete
bdad27815722: Pull complete
dbbbc9a88332: Pull complete
ca4ce74a758b: Pull complete
b8447b6adfa2: Pull complete
6384f9388f98: Pull complete
ad26f7163064: Pull complete
4b2fdddf63d23: Pull complete
5967a1599e89: Pull complete
cabeccef25649: Pull complete
c9f0fb308c3: Pull complete
```

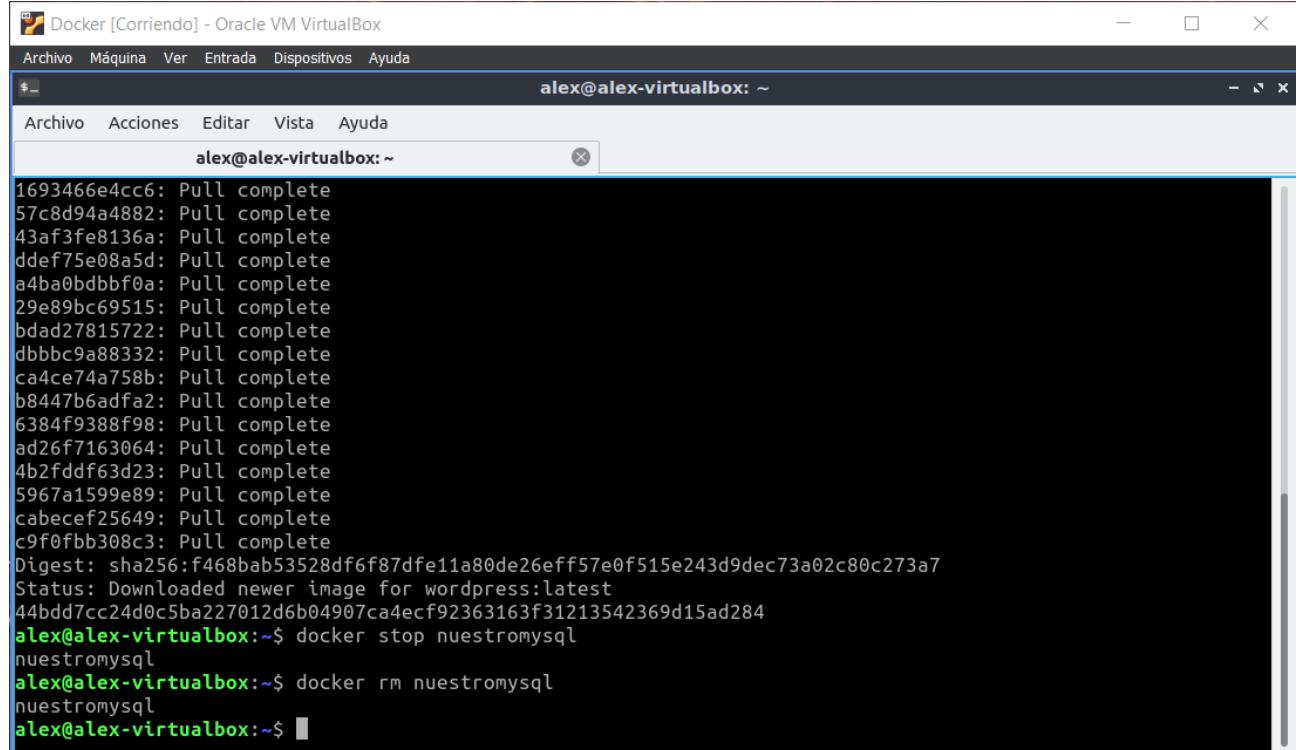
Aquí vemos como se nos abre el WordPress



## Migrar Contenedor MySQL de 5.6 a 5.7

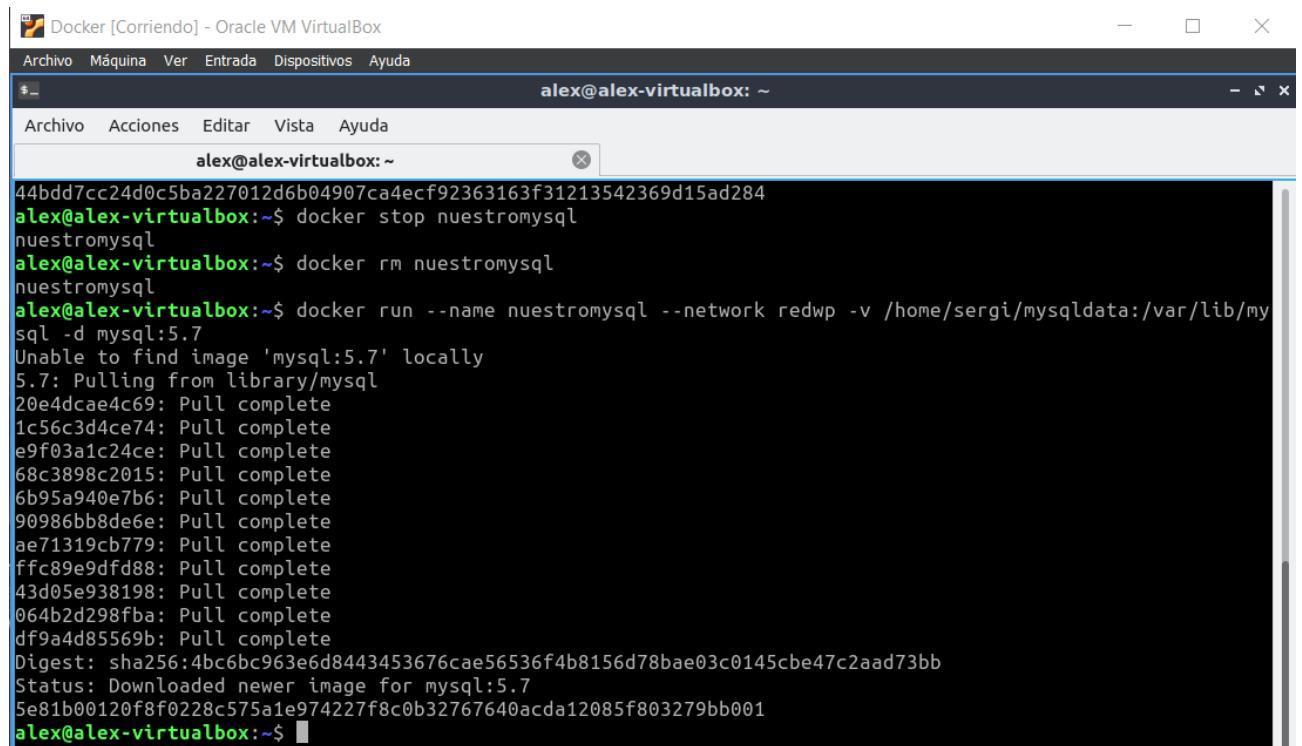
Para migrar de MySQL 5.6 a 5.7, sigue estos pasos:

1. Detén y elimina el contenedor `nuestromysql`:



```
alex@alex-virtualbox: ~
$_
alex@alex-virtualbox: ~
1693466e4cc6: Pull complete
57c8d94a4882: Pull complete
43af3fe8136a: Pull complete
ddef75e08a5d: Pull complete
a4ba0bdbbf0a: Pull complete
29e89bc69515: Pull complete
bdad27815722: Pull complete
dbbbc9a88332: Pull complete
ca4ce74a758b: Pull complete
b8447b6adfa2: Pull complete
6384f9388f98: Pull complete
ad26f7163064: Pull complete
4b2fddf63d23: Pull complete
5967a1599e89: Pull complete
cabecef25649: Pull complete
c9f0fbb308c3: Pull complete
Digest: sha256:f468bab53528df6f87dfe11a80de26eff57e0f515e243d9dec73a02c80c273a7
Status: Downloaded newer image for wordpress:latest
44bdd7cc24d0c5ba227012d6b04907ca4ecf92363163f31213542369d15ad284
alex@alex-virtualbox:~$ docker stop nuestromysql
nuestromysql
alex@alex-virtualbox:~$ docker rm nuestromysql
nuestromysql
alex@alex-virtualbox:~$
```

2. Crea un nuevo contenedor usando MySQL 5.7, mapeando los mismos datos:

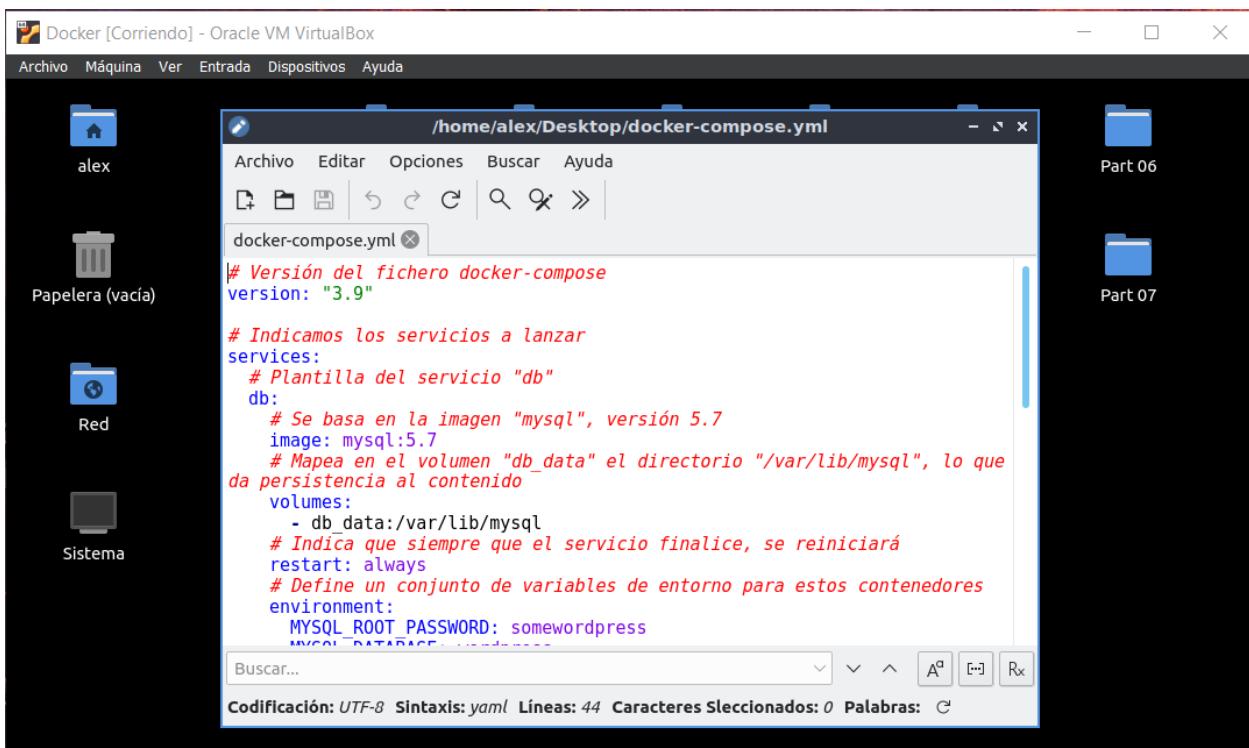


```
alex@alex-virtualbox: ~
$_
alex@alex-virtualbox: ~
44bdd7cc24d0c5ba227012d6b04907ca4ecf92363163f31213542369d15ad284
alex@alex-virtualbox:~$ docker stop nuestromysql
nuestromysql
alex@alex-virtualbox:~$ docker rm nuestromysql
nuestromysql
alex@alex-virtualbox:~$ docker run --name nuestromysql --network redwp -v /home/sergi/mysqldata:/var/lib/mysql
sql -d mysql:5.7
Unable to find image 'mysql:5.7' locally
5.7: Pulling from library/mysql
20e4dcae4c69: Pull complete
1c56c3d4ce74: Pull complete
e9f03a1c24ce: Pull complete
68c3898c2015: Pull complete
6b95a940e7b6: Pull complete
90986bb8de6e: Pull complete
ae71319cb779: Pull complete
ffc89e9df88: Pull complete
43d05e938198: Pull complete
064b2d298fba: Pull complete
df9a4d85569b: Pull complete
Digest: sha256:4bc6bc963e6d8443453676cae56536f4b8156d78bae03c0145cbe47c2aad73bb
Status: Downloaded newer image for mysql:5.7
5e81b00120f8f0228c575a1e974227f8c0b32767640acda12085f803279bb001
alex@alex-virtualbox:~$
```

6. En este caso práctico pondremos en marcha el popular CMS WordPress. Para ello utilizaremos un archivo "docker-compose.yml" comentado que nos pondrá en marcha dos contenedores: el primero utilizando "Apache+PHP", junto con una versión instalada de WordPress, mientras que el segundo contendrá un servidor de base de datos MySQL. Este ejemplo es similar al propuesto como ejemplo durante el contenido de la unidad anterior.

## Paso 1: Crear el archivo docker-compose.yml

Crea un archivo llamado docker-compose.yml con el siguiente contenido:



```
# Versión del fichero docker-compose
version: "3.9"

# Indicamos los servicios a lanzar
services:
  db:
    # Plantilla del servicio "db"
    image: mysql:5.7
    # Mapea en el volumen "db_data" el directorio "/var/lib/mysql", lo que
    # da persistencia al contenido
    volumes:
      - db_data:/var/lib/mysql
    # Indica que siempre que el servicio finalice, se reiniciará
    restart: always
    # Define un conjunto de variables de entorno para estos contenedores
    environment:
      MYSQL_ROOT_PASSWORD: somewordpress
      MYSQL_DATABASE: wordpress
      MYSQL_USER: wordpress
      MYSQL_PASSWORD: wordpress
```

## Paso 2: Poner en marcha el sistema

Ubícate en el directorio donde has guardado el archivo docker-compose.yml y ejecuta el siguiente comando para iniciar los contenedores en segundo plano:

```

Docker [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
alex@alex-virtualbox: ~/Desktop
Archivo Acciones Editar Vista Ayuda
alex@alex-virtualbox: ~/Desktop

Procesando disparadores para man-db (2.9.1-1) ...
Procesando disparadores para systemd (245.4-4ubuntu3.20) ...
Procesando disparadores para linux-image-5.15.0-107-generic (5.15.0-107.117-20.04.1) ...
/etc/kernel/postinst.d/dkms:
 * dkms: running auto installation service for kernel 5.15.0-107-generic
...done.
/etc/kernel/postinst.d/intramfs-tools:
update-intramfs: Generating /boot/initrd.img-5.15.0-107-generic
/etc/kernel/postinst.d/vboxadd:
run-parts: failed to exec /etc/kernel/postinst.d/vboxadd: Exec format error
dpkg: error al procesar el paquete linux-image-5.15.0-107-generic (--configure):
el subproceso instalador paquete linux-image-5.15.0-107-generic script post-installation devolvió el código de salida de error 1
Se detiene el trabajo de procesar:
linux-image-5.15.0-107-generic
E: Sub-process /usr/bin/dpkg returned an error code (1)
alex@alex-virtualbox:~/Desktop$ docker-compose up -d
ERROR: Version in "./docker-compose.yml" is unsupported. You might be seeing this error because you're using the wrong Compose file version. Either specify a supported version (e.g "2.2" or "3.3") and place your service definitions under the 'services' key, or omit the 'version' key and place your service definitions at the root of the file to use version 1.
For more on the Compose file format versions, see https://docs.docker.com/compose/compose-file/
alex@alex-virtualbox:~/Desktop$ 
```

Como me esta dando todo el rato el mismo error y no se solucionarlo, no puedo seguir con la practica.

**PRACTICA 7:Para realizar el ejercicio descrito en el documento sobre el desarrollo de una aplicación en PHP utilizando Visual Studio Code en un contenedor Docker, sigue estos pasos:**

**Paso 1:Instalación de Visual Studio Code y Docker:**

**Descarga e instala Visual Studio Code**

Documentation for Visual Studio Code — Mozilla Firefox

code\_1.89.1-1715060508\_amd64.deb  
Completada — 97.1 MB

Version 1.89 is now available

Thanks for downloading VS Code for Linux!

Download not starting? Try this [direct download link](#).  
Please take a few seconds and help us improve ... [click to take survey](#).

GETTING STARTED

- VS Code in Action
- Top Extensions
- First Steps
- Keyboard Shortcuts
- Downloads
- Privacy

Subscribe

Ask questions

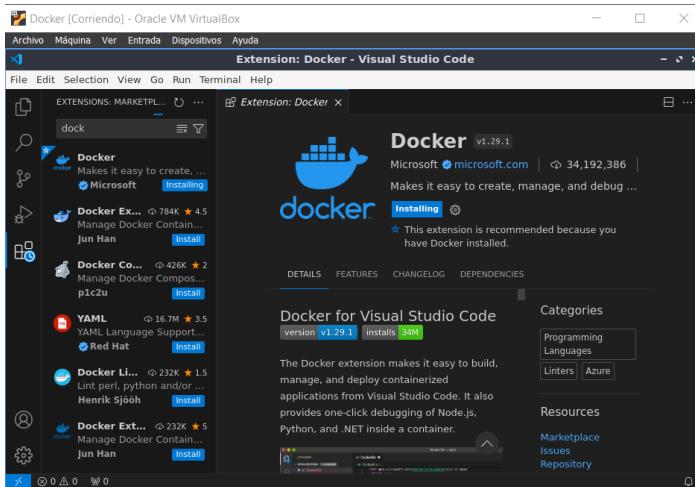
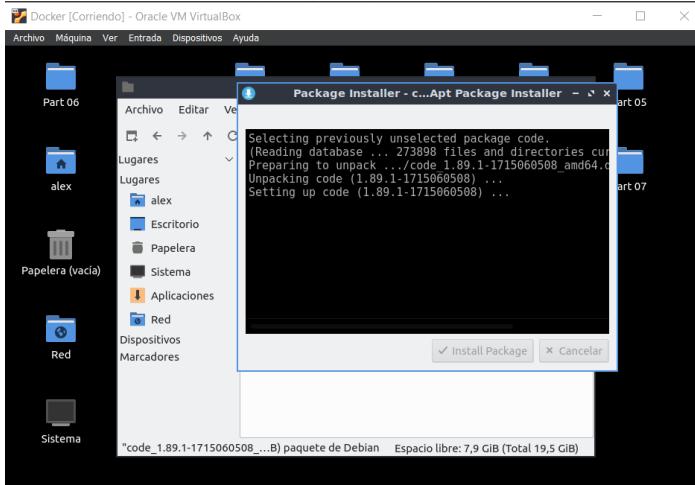
Follow @code

Request features

## Getting Started

Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages and runtimes (such as C++, C#, Java, Python, PHP, Go, .NET). Begin your journey with VS Code with

- Subscribe
- Ask questions
- Follow @code
- Request features



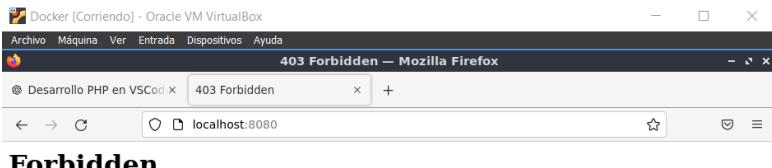
## Paso 2: Instalación de Plugins para Docker en Visual Studio Code:

- Abre Visual Studio Code.
- Ve a la extensión de Visual Studio Code (ícono de cuadrados en la barra lateral izquierda) y busca e instala los siguientes plugins:  
-Docker

```
alex@alex-virtualbox:~$ docker run -d --name servidordesarrollo -p 8080:80 php:7.2-apache
Unable to find image 'php:7.2-apache' locally
7.2-apache: Pulling from library/php
7ec7bd162b2: Pull complete
db606474d60c: Pull complete
efb30f0cd8e0: Pull complete
3bb2e091594: Pull complete
4c761b44e2c: Pull complete
c2199db96575: Pull complete
1b9a9381eeab: Pull complete
fd07bbc59d34: Pull complete
72b73ab27698: Pull complete
983308fafd96: Pull complete
6c13f026e6da: Pull complete
e5e6cd163689: Pull complete
5c5516e56582: Pull complete
154729f6ba86: Pull complete
Digest: sha256:4dc0f015acf8c2f0df69295ae822e49f5ad5fe849725847f15aa0e5802b55f8
Status: Downloaded newer image for php:7.2-apache
lace0ed75f4d15c144a191c0a5b9fe937ae3c197a921924d2901ca4a3f53e0ba
alex@alex-virtualbox:~$
```

### Paso 3: Creación y ejecución de un contenedor PHP:

- Abre una terminal y ejecuta el siguiente comando para crear un contenedor Docker que ejecutará un servidor Apache con PHP:  
"docker run -d --name servidordesarrollo -p 8080:80 php:7.2-apache"
- Verifica que el contenedor esté funcionando accediendo a "http://localhost:8080" en tu navegador. Deberías ver una página de Apache por defecto.



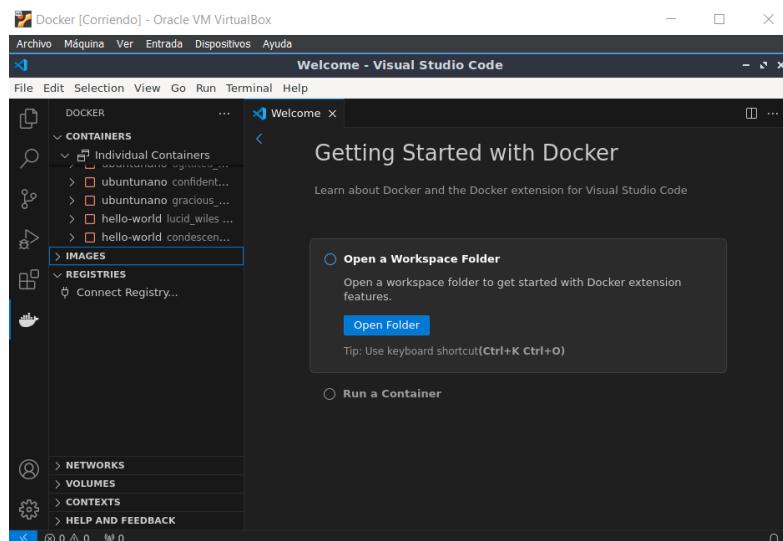
### Forbidden

You don't have permission to access this resource.

Apache/2.4.38 (Debian) Server at localhost Port 8080

### Paso 4: Uso de Docker en Visual Studio Code:

- Abre Visual Studio Code y haz clic en el ícono de Docker (una ballena) en la barra lateral izquierda.
- Deberías ver el contenedor servidordesarrollo listado bajo la sección "Containers".



Como me esta dando todo el rato el mismo error y no se solucionarlo, no puedo seguir con la practica.

## **PROBLEMAS:**

He tenido un montón de problemas en el cmd, con el tema de que me dan errores que no entiendo el porqué, y no he encontrado la solución buscandola en los PDF ni en internet.

## **CONCLUSIÓN:**

Hay que tener un control muy bueno de los conocimientos previos, para poder hacer este trabajo.

## **WEBGRAFIA:**

Los PDF de Oscar.