

# TALLER FINAL: Docker y Kubernetes.

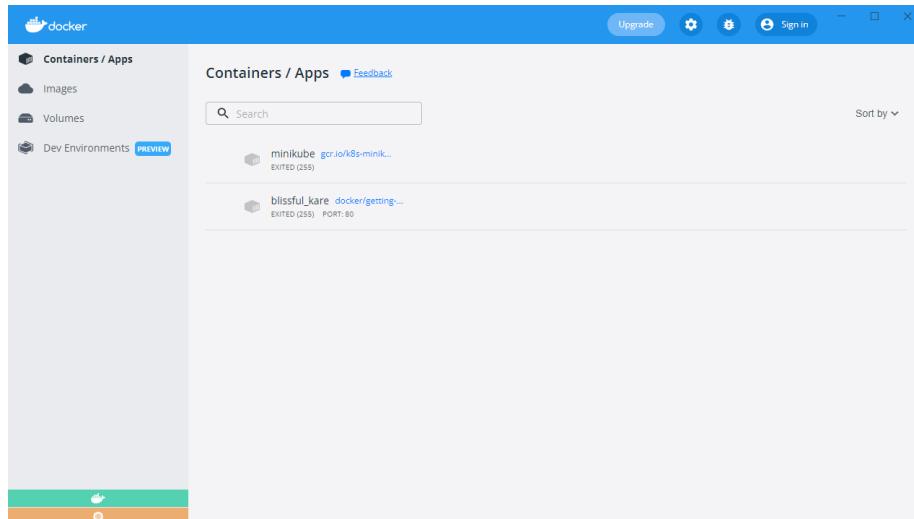
Ing. Est. Luis Felipe Narvaez Gomez. E-mail: [luis.narvaez@usantoto.edu.co](mailto:luis.narvaez@usantoto.edu.co). Cod: 2312660. Facultad de Ingeniería de Sistemas. USTA.

Verificamos la instalación de Docker en Windows, para esto invocamos la consola de comandos CMD en modo de Administrador.

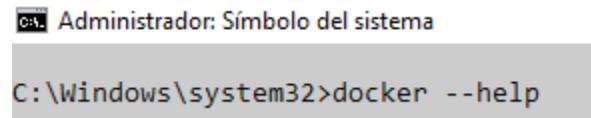


```
C:\Windows\system32>docker --version
Docker version 20.10.14, build a224086
C:\Windows\system32>
```

Una vez habiendo verificado que tenemos Docker en nuestro sistema operativo, podemos abrir a interfaz gráfica de usuario.



Ahora instalaremos las imágenes necesarias que vamos a necesitar como lo puede ser mysql en la versión 5.7 o superior, y php en versión 7.0 junto con apache, también en la especificada o en una más reciente. Estas instalaciones se harán mediante el uso del comando Docker pull, recuerde que puede hacer esto tanto en la versión de Docker Desktop y en la terminal de CMD con permisos de administrador, necesitara acceder al Hub de Docker para hacerse con las imágenes y con el comando necesario para escribir en la cmd. En caso de necesitar comandos auxiliares puede ver todas las herramientas de Docker con el siguiente comando.



```
C:\Windows\system32>docker --help
```

Para utilizar las imágenes por Docker Destop, debemos acceder a Docker Hub, para acceder al mismo primero debemos iniciar sesión. En la parte superior de la ventana de Docker Desktop tenemos un icono que nos permitirá hacer esto con las palabras “Sign In” esto abrirá una ventana en nuestro navegador predeterminado con la pagina donde podemos logearnos.

The screenshot shows two windows side-by-side. On the left is the Docker Desktop welcome screen, which asks for a Docker email address (luis.narvaez@usantoto.edu.co) and has a 'Continue' button. On the right is the Docker Hub search results page for 'https://hub.docker.com/search?q=mysql'. The search results list several official Docker images, including 'alpine', 'busybox', 'ubuntu', and 'python'. Each result shows the image name, official status, last update time, download count, star count, and supported architectures.

En cada paquete a descargar solo debemos escribir el nombre de la imagen que necesitamos y abrir la que nos interesa.

This screenshot shows the Docker Hub details page for the 'mysql' image. It features the MySQL logo, the image name 'mysql' with an official badge, and a brief description: 'MySQL is a widely used, open-source relational database management system (RDBMS)'. Below the description are download statistics (1B+), supported architectures (Linux, ARM 64, x86-64), and a 'Docker Official Image' badge. To the right, there's a terminal-like interface with the command 'docker pull mysql' and a 'Copy' button. Navigation tabs at the bottom include 'Description', 'Reviews', and 'Tags'.

```
C:\Windows\system32>docker pull mysql
Using default tag: latest
latest: Pulling from library/mysql
c1ad9731b2c7: Pull complete
54f6eb0ee84d: Pull complete
cffcf8691bc5: Pull complete
89a783b5ac8a: Pull complete
6a8393c7be5f: Pull complete
af768d0b181e: Pull complete
810d6aaaf54a: Pull complete
2e014a8ae4c9: Pull complete
a821425a3341: Pull complete
3a10c2652132: Pull complete
4419638feac4: Pull complete
681aedd97dfe: Pull complete
Digest: sha256:548da4c67fda71908f17c308b8ddb098acf5191d3d7694e56801c6a8b2072cc
Status: Downloaded newer image for mysql:latest
docker.io/library/mysql:latest

C:\Windows\system32>
```

The screenshot shows the Docker Hub interface for the 'php' image. It features a large icon of a ship, the text 'php DOCKER OFFICIAL IMAGE', and a brief description: 'While designed for web development, the PHP scripting language also provides general-purpose use.' Below this are filter options for '500M+' images, architecture ('Linux', 'mips64le', etc.), and a 'Docker Official Image' badge. A 'Copy and paste to pull this image' button contains the command 'docker pull php'. At the bottom, there's a 'View Available Tags' link.

Ahora verificamos tanto la versión como las imágenes que tenemos instaladas en nuestra computadora con el comando de images.

```
C:\Windows\system32>docker images
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
php                latest   ca1e1f0e01ef  18 hours ago  484MB
mysql              latest   65b636d5542b  20 hours ago  524MB
docker/getting-started    latest   cb90f98fd791  6 weeks ago   28.8MB
hubproxy.docker/internal:5000/docker/desktop-kubernetes   latest   74146699a088  2 months ago  347MB
k8s.gcr.io/kube-dns/kube-dns     v1.22.0    5344ca200e9e  3 months ago  1.47GB
hubproxy.docker/internal:5000/docker/desktop-kubernetes   v1.22.5    059e6cd8c7f8  5 months ago  128MB
k8s.gcr.io/kube-apiserver   v1.22.5    059e6cd8c7f8  5 months ago  128MB
hubproxy.docker/internal:5000/docker/desktop-kubernetes-controller-manager  v1.22.5    04185bc88e08  5 months ago  122MB
k8s.gcr.io/kube-controller-manager  v1.22.5    04185bc88e08  5 months ago  122MB
hubproxy.docker/internal:5000/docker/desktop-kubernetes-scheduler   v1.22.5    935db8fd2c52  5 months ago  52.7MB
hubproxy.docker/internal:5000/docker/desktop-kubernetes   v1.22.5    8974d672422d  5 months ago  104MB
k8s.gcr.io/kube-proxy    v1.22.5    8974d672422d  5 months ago  104MB
hubproxy.docker/internal:5000/docker/desktop-kubernetes-etcd   3.5.0-0    004811815584  11 months ago  295MB
k8s.gcr.io/etcd           3.5.0-0    004811815584  11 months ago  295MB
kubernetes/etcd-metrics-scrape   v1.0.7    7881fc6d5c50  11 months ago  34.4MB
hubproxy.docker/internal:5000/docker/desktop-kubernetes-coredns   v1.8.4    8d147537fb7d  12 months ago  47.6MB
k8s.gcr.io/coredns/coredns        v1.8.4    8d147537fb7d  12 months ago  47.6MB
hubproxy.docker/internal:5000/docker/desktop-vpnkit-controller   v2.0     8c23c8a0e676e  12 months ago  23MB
docker/desktop-vpnkit-controller  v2.0     99f89471f470  13 months ago  41.9MB
docker/desktop-storage-provisioner  v2.0     99f89471f470  13 months ago  41.9MB
hubproxy.docker/internal:5000/docker/desktop-storage-provisioner  3.5     ed210e3e4a5b  14 months ago  683kB
k8s.gcr.io/pause             3.5     ed210e3e4a5b  14 months ago  683kB

C:\Windows\system32>
```

A continuación, correremos las imágenes que vamos a utilizar en nuestra actividad, para esto es necesario utilizar el comando de run, seguido de las banderas especiales, estas banderas son:

1. La bandera (-p) es utilizada para indicar el puerto por el cual se escuchará la aplicación u el puerto del contenedor.
2. La bandera (--name) no proporciona el nombre del proceso o contenedor que se iniciara.
3. La bandera -V, indica la creación de una columna. Esto a su vez permite almacenar datos de la base de datos (BD) en una carpeta con el fin de salvar los datos en caso de que el contenedor se apague o se caiga.

La primera tuta ara referencia a la carpeta que se creara en nuestra computadora, normalmente siendo esta en el escritorio, la segunda hace referencia a la dirección del contenedor la cual esta asignada por defecto.

4. La bandera (-e) indica la contraseña la cual puede ser cualquiera.
5. La bandera (-d) se utiliza para que la imagen que vamos a correr lo haga en segundo plano.

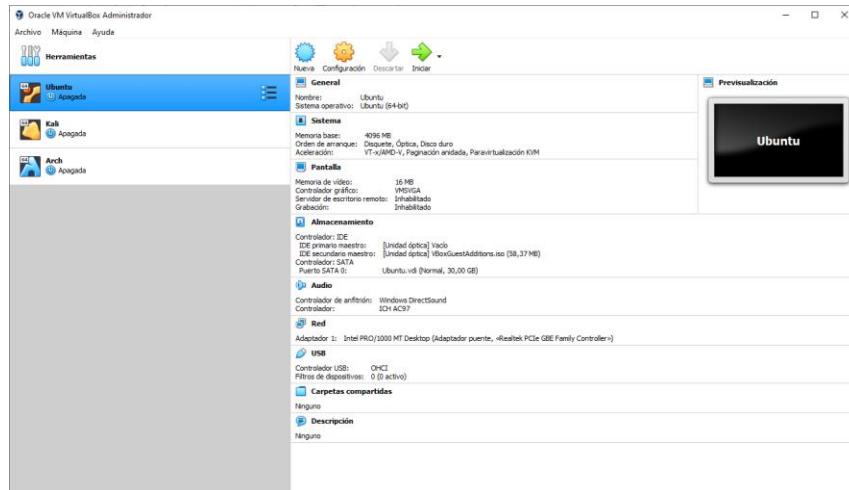
Primero iniciaremos corriendo la imagen de mysql con ayuda de las anteriores banderas.

```
C:\Windows\system32>docker login  
Authenticating with existing credentials...  
Login Succeeded  
  
C:\Windows\system32>
```

### **NOTA: Los comandos no me funcionaron en Windows asi que paso a la VM Ubuntu GNU Linux.**

Dependiendo del equipo en que vamos a trabajar, podemos desarrollar este taller en una VM (Virtual Machine) con sistema Linux, en una RaspBerry PI (Debian PI LINUX), en una maquina con Microsoft Windows desde base como sistema operativo o en su defecto una distribución de Linux, en mi caso trabajare con una VM.

Mi Sistema Operativo Base es Microsoft Windows 10 Home y el Software que utilizare para virtualizar una distribución de Linux será Virtual BOX.



En mi caso trabajare con la distribución de Ubuntu Linux. Antes de empezar a trabajar e iniciar la VM se debe confirmar que la misma tenga conexión a internet, la misma es recomendable que se la del Sistema Operativo base por configuración puente. Para que se de esto, en la configuración de redes de la VM se debe seleccionar el “Adaptador Puente” seguido de seleccionar el dispositivo o Driver mismo de uso que el de la Maquina Base. Para saber con qué dispositivo está trabajando la maquina Base, siendo en mi caso Microsoft Windows 10 Home, se debe ir al apartado de configuración, internet y red, estado, propiedades y en la parte final de la venta aparecerá la información que necesitamos.

## Configuración de IP

Asignación de IP: Automático (DHCP)

**Propiedades**

Velocidad de vínculo (recepción/transmisión): 1000/1000 (Mbps)

Dirección IPv6 del vínculo local: [REDACTED]

Dirección IPv4: [REDACTED]

Servidores DNS IPv4: [REDACTED]

Fabricante: Realtek

Descripción: Realtek PCIe GBE Family Controller

Versión del controlador: [REDACTED]

Dirección física (MAC): [REDACTED]

**Red**

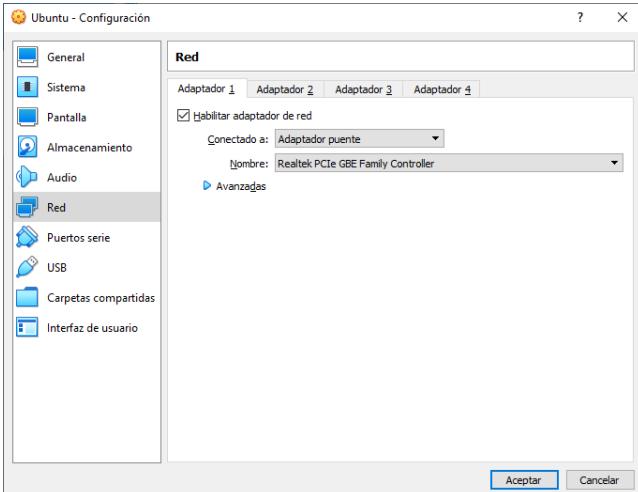
Habilitar adaptador de red

Conectado a: Adaptador puente

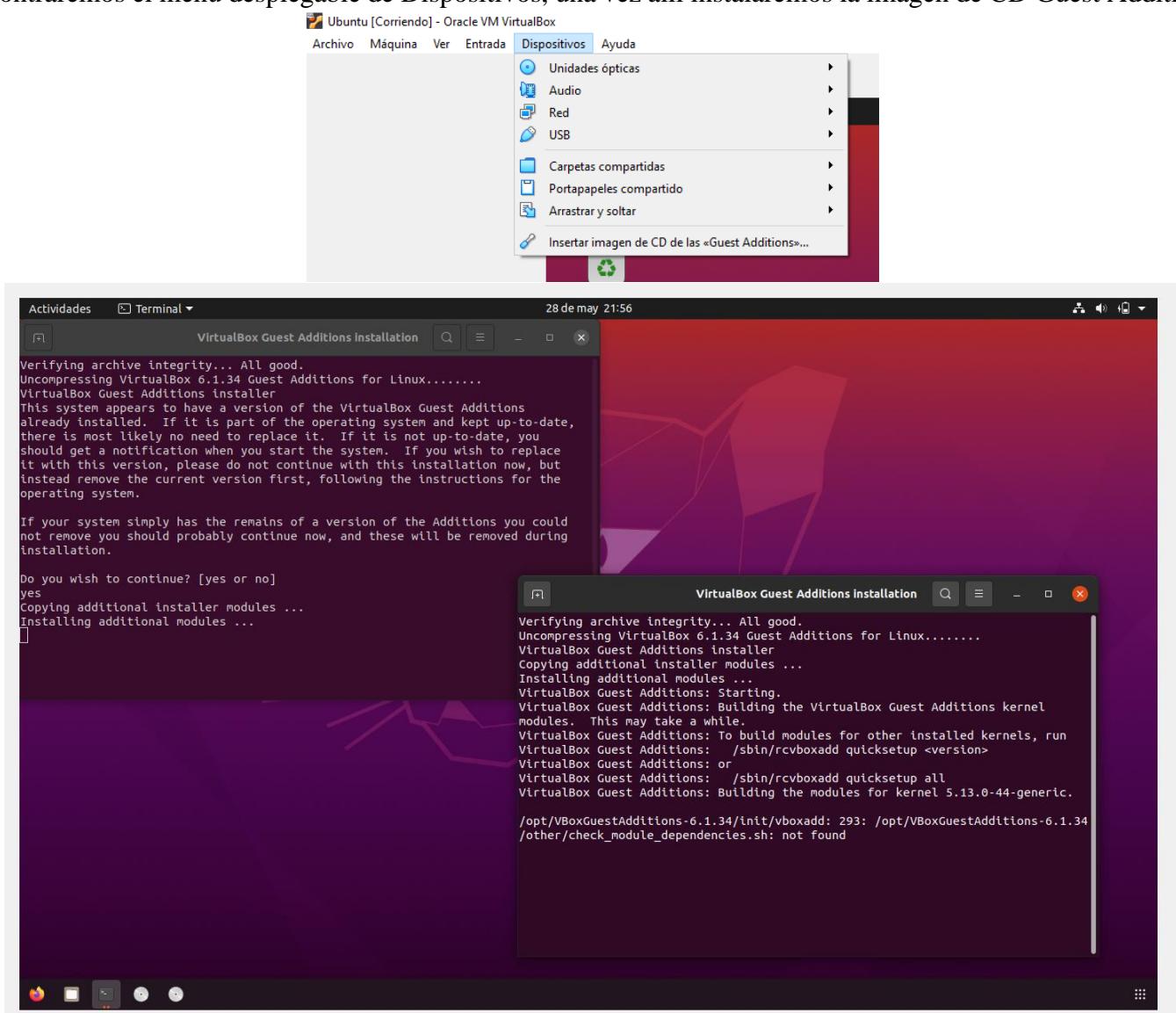
Nombre: Realtek PCIe GBE Family Controller

Aceptar Cancelar

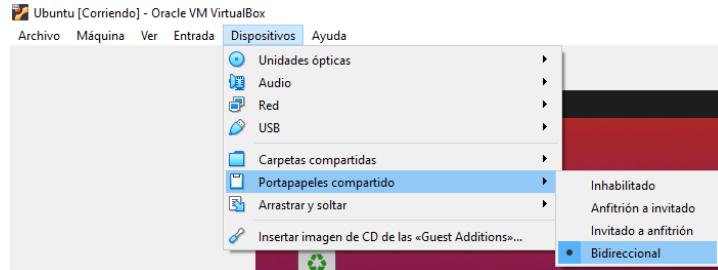
Copiar



Es recomendable que en la maquina virtual instalamos los discos necesarios para que se puedan comunicar el sistema operativo Base y el virtualizado. En virtual Box, en el menú de herramientas suma ves iniciada la máquina, encontraremos el menú desplegable de Dispositivos, una vez allí instalaremos la imagen de CD Guest Additions.



También habilitaremos, la opción de portapapeles compartido entre máquinas. Para esto en Virtual Box habilitaremos en la barra de menús, en la parte de Dispositivos, portapapeles compartido, la opción bidireccional.



Iniciamos la máquina virtual y lo primero que debemos hacer es verificar que nuestro sistema se encuentre con sus dependencias actualizadas, para esto utilizamos nuestros comandos de Upgrade y Update.

```
ruiso@ruiso-VirtualBox:~$ sudo apt-get upgrade
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Calculando la actualización... Hecho
El paquete indicado a continuación se instaló de forma automática y ya no es necesario.
  libfwupdplugin1
Utilice «sudo apt autoremove» para eliminarlo.
Se actualizarán los siguientes paquetes:
  bolt
1 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 0 B/143 kB de archivos.
Se utilizarán 37,9 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] S
(Leyendo la base de datos ... 183035 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../bolt_0.9.1-2~ubuntu20.04.1_amd64.deb ...
Desempaquetando bolt (0.9.1-2~ubuntu20.04.1) sobre (0.8~ubuntu1) ...
Configurando bolt (0.9.1-2~ubuntu20.04.1) ...
bolt.service is a disabled or a static unit not running, not starting it.
Procesando disparadores para dbus (1.12.16~ubuntu2.2) ...
Procesando disparadores para man-db (2.9.1-1) ...
Procesando disparadores para man-db (2.9.1-1) ...
ruiso@ruiso-VirtualBox:~$
```

```
ruiso@ruiso-VirtualBox:~$ sudo apt-get update
Obj:1 http://co.archive.ubuntu.com/ubuntu focal InRelease
Des:2 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Des:3 http://co.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Des:4 http://co.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Des:5 http://security.ubuntu.com/ubuntu focal-security/main amd64 DEP-11 Metadata [40,7 kB]
Des:6 http://security.ubuntu.com/ubuntu focal-security/universe amd64 DEP-11 Metadata [66,3 kB]
Des:7 http://security.ubuntu.com/ubuntu focal-security/multiverse amd64 DEP-11 Metadata [2,464 B]
Des:8 http://co.archive.ubuntu.com/ubuntu focal-updates/main amd64 DEP-11 Metadata [278 kB]
Des:9 http://co.archive.ubuntu.com/ubuntu focal-updates/universe amd64 DEP-11 Metadata [390 kB]
Des:10 http://co.archive.ubuntu.com/ubuntu focal-updates/multiverse amd64 DEP-11 Metadata [940 B]
Des:11 http://co.archive.ubuntu.com/ubuntu focal-backports/main amd64 DEP-11 Metadata [7,964 B]
Des:12 http://co.archive.ubuntu.com/ubuntu focal-backports/universe amd64 DEP-11 Metadata [30,8 kB]
Descargados 1.153 kB en 2s (607 kB/s)
Leyendo lista de paquetes... Hecho
ruiso@ruiso-VirtualBox:~$
```

Ahora antes de instalar Docker necesitaremos de una paquetes previos y necesarios en nuestra VM para esto utilizaremos el comando:

**sudo apt-get install curl apt-transport-https ca-certificates software-properties-common**

```
ruiso@ruiso-VirtualBox:~$ sudo apt-get install curl apt-transport-https ca-certificates software-properties-common
[sudo] contraseña para ruiso:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
```

Este comando traduce:

1. apt-transport-https: permite que el administrador de paquetes transfiera datos a través de https
2. ca-certificates: permite que el navegador web y el sistema verifiquen los certificados de seguridad
3. curl: transfiere datos
4. software-properties-common: agrega scripts para administrar el software

Una vez instalados podemos agregar los repositorios de Docker, los cuales facilitaran el proceso de instalación y al mismo tiempo podremos utilizar el método de instalación oficial. Primero agregaremos la clave GPG con el siguiente comando.

**curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -**

```
ruiso@ruiso-VirtualBox:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
OK
ruiso@ruiso-VirtualBox:~$
```

Luego agregamos el repositorio.

**sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu \$(lsb\_release -cs) stable"**

```
ruiso@ruiso-VirtualBox:~$ sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"
Des:1 https://download.docker.com/linux/ubuntu focal InRelease [57,7 kB]
Des:2 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
```

Y por último actualizamos la información del repositorio

**sudo apt update**

```
ruiso@ruiso-VirtualBox:~$ sudo apt update
Obti:1 https://download.docker.com/linux/ubuntu focal InRelease
```

Puedes asegurarte de estar instalando desde el repositorio de Docker en lugar del repositorio predeterminado de Ubuntu con este comando, en el debemos observar que Docker-ce, no aparezca instalado.

## apt-cache policy docker-ce

```
ruiso@ruiso-VirtualBox:~$ apt-cache policy docker-ce
docker-ce:
  Instalados: (ninguno)
  Candidato: 5:20.10.16~3-0~ubuntu-focal
  Tabla de versión:
    5:20.10.16~3-0~ubuntu-focal 500
      500 https://download.docker.com/linux/ubuntu/focal/stable_amd64_Packages
```

Ahora instalamos propiamente Docker.

## sudo apt install docker-ce

```
ruiso@ruiso-VirtualBox:~$ sudo apt install docker-ce
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
```

Y verificamos el estado de Docker en nuestra VM.

## sudo systemctl status docker

```
ruiso@ruiso-VirtualBox:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2022-05-28 22:57:03 -05; 2min 11s ago
     TriggeredBy: ● docker.socket
 Main PID: 4332 (dockerd)
   Tasks: 7
    Memory: 28.2M
      CGroup: /system.slice/docker.service
             └─4332 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

may 28 22:57:01 ruiso-VirtualBox dockerd[4332]: time="2022-05-28T22:57:01.622096835-05:00" level=warning msg="Your kernel does not support CPU realtime scheduler"
may 28 22:57:01 ruiso-VirtualBox dockerd[4332]: time="2022-05-28T22:57:01.622547380-05:00" level=warning msg="Your kernel does not support cgroup blkio weight"
may 28 22:57:01 ruiso-VirtualBox dockerd[4332]: time="2022-05-28T22:57:01.622647356-05:00" level=warning msg="Your kernel does not support cgroup blkio weight_device"
may 28 22:57:01 ruiso-VirtualBox dockerd[4332]: time="2022-05-28T22:57:01.622870045-05:00" level=info msg="Loading containers: start."
may 28 22:57:02 ruiso-VirtualBox dockerd[4332]: time="2022-05-28T22:57:02.025732799-05:00" level=info msg="Default bridge (dockero) is assigned with an IP address 172.17.0.0/16. Daemon option --bip can be used"
may 28 22:57:02 ruiso-VirtualBox dockerd[4332]: time="2022-05-28T22:57:02.291125022-05:00" level=info msg="Loading containers: done."
may 28 22:57:02 ruiso-VirtualBox dockerd[4332]: time="2022-05-28T22:57:02.923623448-05:00" level=info msg="Docker daemon" commit=f756502 graphdriver(s)=overlay2 version=20.10.16
may 28 22:57:02 ruiso-VirtualBox dockerd[4332]: time="2022-05-28T22:57:02.924253137-05:00" level=info msg="Daemon has completed initialization"
may 28 22:57:03 ruiso-VirtualBox dockerd[4332]: time="2022-05-28T22:57:03.012710498-05:00" level=info msg="API listen on /run/docker.sock"
may 28 22:57:03 ruiso-VirtualBox systemd[1]: Started Docker Application Container Engine.
ESOC
```

Verificando la versión ...

```
ruiso@ruiso-VirtualBox:~$ docker --version
Docker version 20.10.16, build aa7e414
ruiso@ruiso-VirtualBox:~$
```

Instalamos las imágenes de mysql y php junto con apache.

```
ruiso@ruiso-VirtualBox:~$ sudo docker pull mysql
[sudo] contraseña para ruiso:
Using default tag: latest
latest: Pulling from library/mysql
c1ad9731b2c7: Pull complete
54f6eb0ee84d: Pull complete
cffcf8691bc5: Pull complete
89a783b5ac8a: Pull complete
6a8393c7be5f: Pull complete
af768dbb181e: Pull complete
810d6aaaf54a: Pull complete
ze014aabae4c9: Pull complete
a821425a3341: Pull complete
3a10c2652132: Pull complete
4419638bfeac4: Pull complete
681aeed97fe: Pull complete
Digest: sha256:548da4c67fd8a71908f17c308b8ddb098acf5191d3d7694e56801c6a8b2072cc
Status: Downloaded newer image for mysql:latest
docker.io/library/mysql:latest
ruiso@ruiso-VirtualBox:~$
```

```
ruiso@ruiso-VirtualBox:~$ sudo docker pull php
Using default tag: latest
latest: Pulling from library/php
42c077c10790: Pull complete
8934009a9160: Pull complete
5357ac116991: Pull complete
54ae63894b5a: Pull complete
72281f038a08: Pull complete
9fdb1b94317fe: Pull complete
00012d9e2ea5: Pull complete
2c220aff91be: Pull complete
48cfef9bf9b47: Pull complete
Digest: sha256:d7a3047dacb99c0bf739abef5113d3a3ead04cbffa0472d1c24f678251975a9f1
Status: Downloaded newer image for php:latest
docker.io/library/php:latest
ruiso@ruiso-VirtualBox:~$
```

Verificamos que las imágenes esten instaladas en la terminal.

```
ruiso@ruiso-VirtualBox:~$ sudo docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
php            latest    caa1ef0e01ef  20 hours ago  484MB
mysql          latest    65b636d5542b  23 hours ago  524MB
ruiso@ruiso-VirtualBox:~$
```

Ahora si a continuación, correremos las imágenes que vamos a utilizar en nuestra actividad, para esto es necesario utilizar el comando de run, seguido de las banderas especiales, estas banderas son:

1. La bandera (-p) es utilizada para indicar el puerto por el cual se escuchará la aplicación u el puerto del contenedor.
2. La bandera (--name) no proporciona el nombre del proceso o contenedor que se iniciara.
3. La bandera -V, indica la creación de una columna. Esto a su vez permite almacenar datos de la base de datos (BD) en una carpeta con el fin de salvar los datos en caso de que el contenedor se apague o se caiga.

La primera tuta ara referencia a la carpeta que se creara en nuestra computadora, normalmente siendo esta en el escritorio, la segunda hace referencia a la dirección del contenedor la cual esta asignada por defecto.

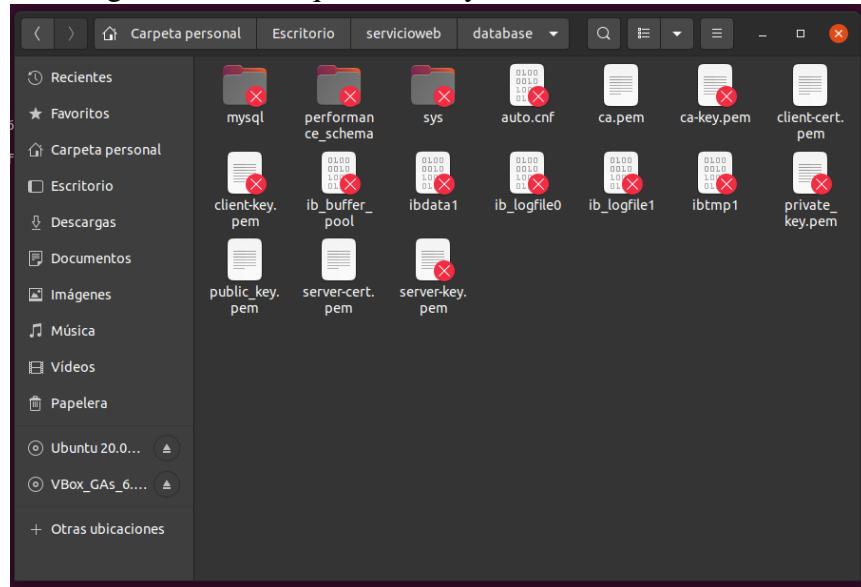
4. La bandera (-e) indica la contraseña la cual puede ser cualquiera.
5. La bandera (-d) se utiliza para que la imagen que vamos a correr lo haga en segundo plano.
6. La bandera (- -link), indica que se pueden comunicar entre el contenedor de php y la base de datos.

Primero iniciaremos corriendo la imagen de mysql con ayuda de las anteriores banderas.

```
sudo docker run -p 3307:3306 --name basedatos -v  
/home/ruiso/Escritorio/servicioweb/database:/var/lib/mysql -e MYSQL_ROOT_PASSWORD=password -  
d mysql:5.7
```

```
ruiso@ruiso-VirtualBox:~$ sudo docker run -p 3307:3306 --name basedatos -v /home/ruiso/Escritorio/servicioweb/database:/var/lib/mysql -e MYSQL_ROOT_PASSWORD=password -d mysql:5.7  
Unable to find image 'mysql:5.7' locally  
5.7: Pulling from library/mysql  
c1ad9731b2c7: Already exists  
54fe00ee84d: Already exists  
cfcfc8691bc5: Already exists  
89a783b5ac8: Already exists  
6a8393c7be5f: Already exists  
af768d0b181e: Already exists  
810d6aaaf54a: Already exists  
81fe6daf2395: Pull complete  
5ccf426818fd: Pull complete  
68bb3bb66054: Pull complete  
1b666c4f93df: Pull complete  
Digest: sha256:e99b2b8d5bca914ef31059858210f57b009c40375d647f0d4d65ecd01d6b1d5  
Status: Downloaded newer image for mysql:5.7  
09ed8013b16a0a2b56b503f5d8f31ef4b28127391b9ebe9160be8188edf7dc2a  
ruiso@ruiso-VirtualBox:~$
```

Tal y como escribimos el comando anterior, habremos creado en la dirección dada el árbol de ficheros en caso de que no existan en primer lugar, en caso de que no se hayan creado debemos hacerlo manualmente.



En caso de que tendreamos que por algún motivo reiniciar nuestra VM, apagar la maquina o continuar en un tiempo después la actividad, debemos contemplar que la imagen que creamos entrara en modo de STOP existira pero no funcionara a menos de que le digamos START nuevamente, esto se hace de la siguiente manera.

```
ruiso@ruiso-VirtualBox:~$ sudo docker ps -l  
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES  
09ed8013b16a mysql:5.7 "docker-entrypoint.s..." 36 hours ago Exited (0) 36 hours ago basedatos  
ruiso@ruiso-VirtualBox:~$ sudo docker ps  
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES  
ruiso@ruiso-VirtualBox:~$ sudo docker start 09ed  
09ed  
ruiso@ruiso-VirtualBox:~$ sudo docker ps  
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES  
09ed8013b16a mysql:5.7 "docker-entrypoint.s..." 36 hours ago Up 3 seconds 33060/tcp, 0.0.0.0:3307->3306/tcp, :::3307->3306/tcp basedatos  
ruiso@ruiso-VirtualBox:~$
```

Ahora, corremos php con banderas similares a las que utilizamos al correr mysql. Luego verificamos que los contenedores esten corriendo en simultaneo, debemos tener en cuenta que cada uno este en ejecución en puerto diferente para evitar problemas.

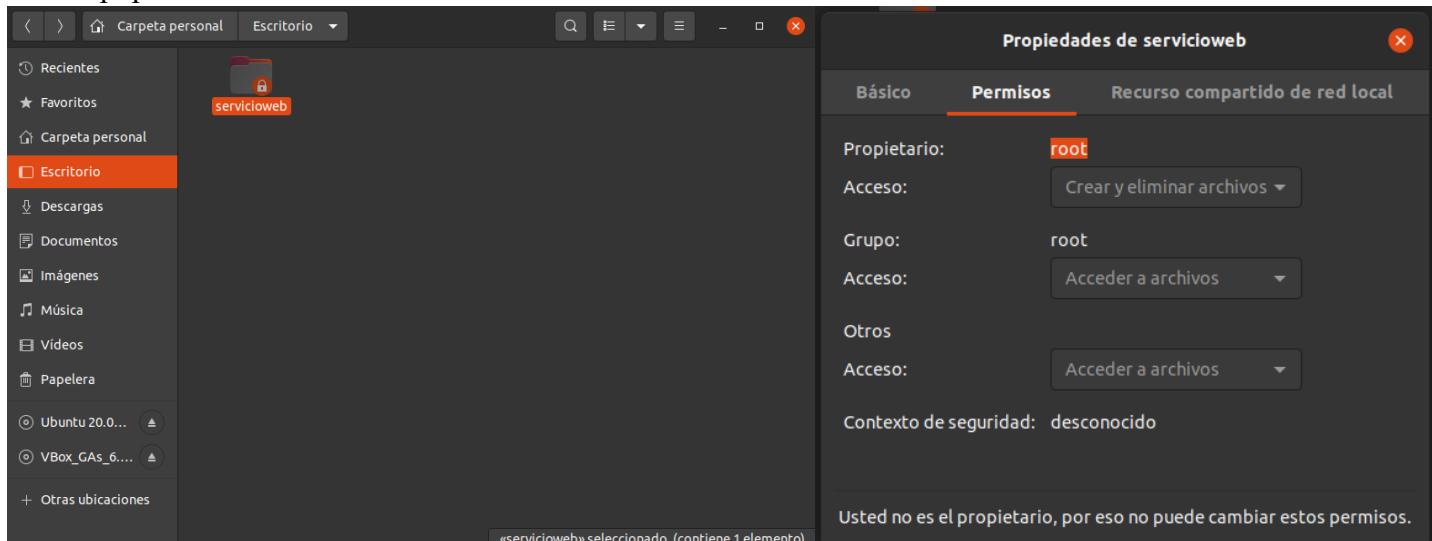
```
sudo docker run -p 1000:80 -v /home/ruiso/Escritorio/servicioweb:/var/www/html --name miservidorphp  
-d --link basedatos php:7.0-apache
```

```
ruiso@ruiso-VirtualBox:~$ sudo docker run -p 1000:80 -v /home/ruiso/Escritorio/servicioweb:/var/www/html  
--name miservidorphp -d --link basedatos php:7.0-apache  
Unable to find image 'php:7.0-apache' locally  
7.0-apache: Pulling from library/php  
177e7ef0df69: Pull complete  
9bf89f2eda24: Pull complete  
350207dcf1b7: Pull complete  
a8a33d96b4e7: Pull complete  
c0421d5b63d6: Pull complete  
f76e300fbe72: Pull complete  
af9ff1b9ce5b: Pull complete  
d9f072d61771: Pull complete  
37007e292198: Pull complete  
8ba923990f24: Pull complete  
98af8902979a: Pull complete  
f1548c2cd376: Pull complete  
e1062fd0605a: Pull complete  
Digest: sha256:1d34b2e491a02ba7a8d26478132015e197a5ffea37f0a93b42621d11cf042cc  
Status: Downloaded newer image for php:7.0-apache  
0f931837beaf66456844f4e1161d6f4265b574997de10bb3feeff415edd33de8  
ruiso@ruiso-VirtualBox:~$
```

Verificamos con Docker ps

```
ruiso@ruiso-VirtualBox: $ sudo docker ps  
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES  
0f931837beaf php:7.0-apache "docker-php-entrypoi..." About a minute ago Up About a minute 0.0.0.0:1000->80/tcp, ::1000->80/tcp miservidorphp  
09ed8013b16a mysql:5.7 "docker-entrypoint.s..." 37 hours ago Up 7 minutes 33060/tcp, 0.0.0.0:3307->3306/tcp, ::3307->3306/tcp basedatos  
ruiso@ruiso-VirtualBox:~$
```

Ahora dentro de la carpeta que hemos creado en el Escritorio, dentro crearemos un nuevo archivo php llamado “index.php”



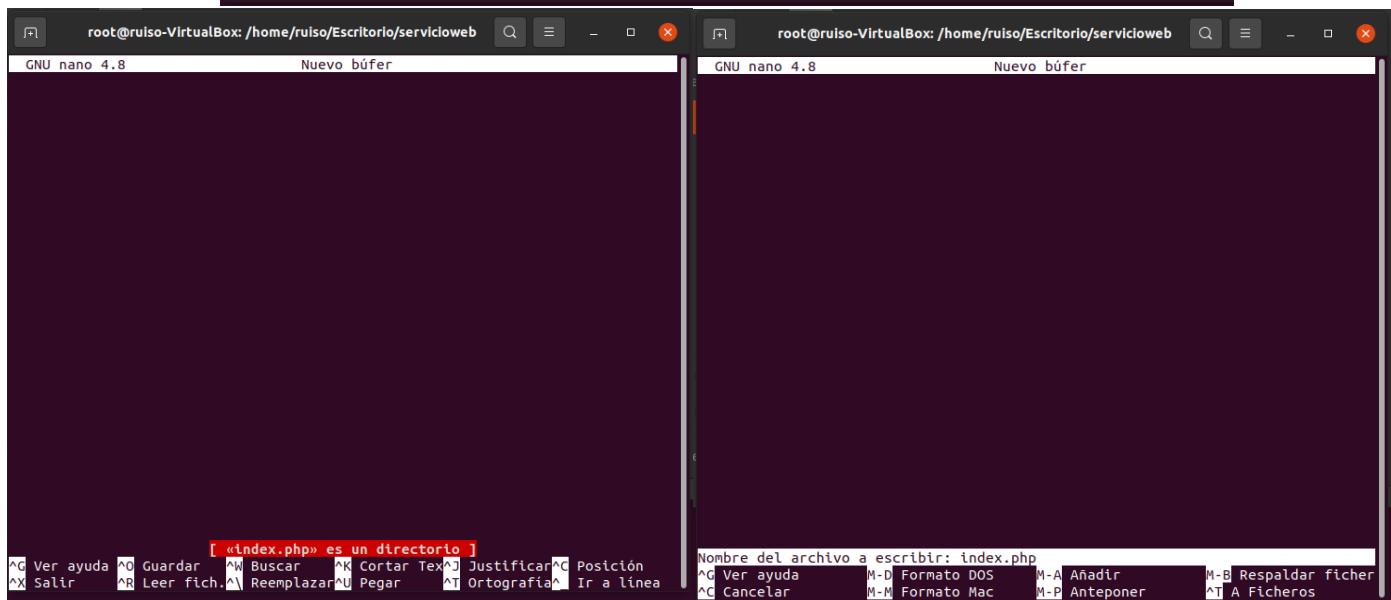
La carpeta a la que queremos acceder, se creo con permisos de administrador, por lo que no nos permitirá modificar, crear o leer un archivo o directorio sin los mismos permisos, estos nos dejan dos soluciones básicas, o creamos manualmente un nuevo directo a los cuales dirigir las imágenes que ya hemos creado o trabajamos con lo que tenemos, pero con una termina con los permisos de administrador. Elegiré la segunda opción, abriendo el ROOT desde una ruta raíz de la maquina y navegando hasta el directorio de interés, luego creamos y modificamos el archivo que necesitamos, “index.php”.

```

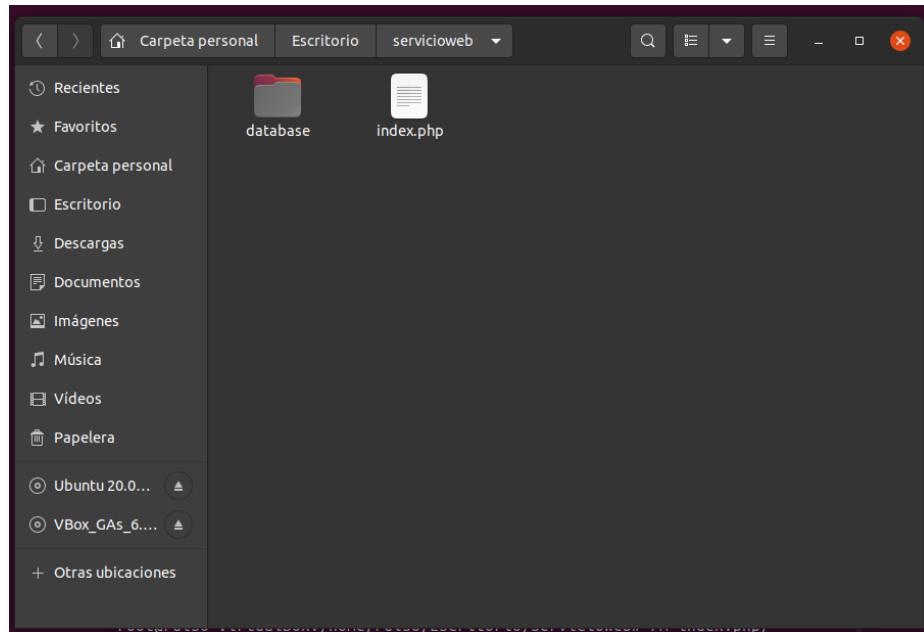
ruiso@ruiso-VirtualBox:~/Escritorio/servicioweb$ cd ..
ruiso@ruiso-VirtualBox:~/Escritorio$ sudo su -
[sudo] contraseña para ruiso:
root@ruiso-VirtualBox:~# ls
snap
root@ruiso-VirtualBox:~# cd ..
root@ruiso-VirtualBox:/# ls
bin dev lib libx32 mnt root snap sys var
boot etc lib32 lost+found opt run srv tmp
cdrom home lib64 media proc sbin swapfile usr
root@ruiso-VirtualBox:/# cd home
root@ruiso-VirtualBox:/home# ls
ruiso
root@ruiso-VirtualBox:/home# cd ruiso
root@ruiso-VirtualBox:/home/ruiso# ls
Descargas Escritorio Música Público
Documentos Imágenes Plantillas Videos
root@ruiso-VirtualBox:/home/ruiso# cd Escritorio/
root@ruiso-VirtualBox:/home/ruiso/Escritorio# ls
servicioweb
root@ruiso-VirtualBox:/home/ruiso/Escritorio# cd servicioweb/
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# mkdir index.php
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# 

root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# ls
database index.php
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# rm index.php/
rm: no se puede borrar 'index.php/': Es un directorio
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# rm -r index.php/
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# sudo nano index.php
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# 

```



Para guardar [CTRL]+[O], luego de escribir el nombre del archivo [ENTER], para salir [CTRL]+[X]. Como puede observar en el comando primero se creo un directorio o carpeta con el nombre que necesitamos, pero luego es arreglado con la eliminación de esta última y su reemplazo por un archivo de texto plano de extensión php.



Ahora vamos a modificar el archivo con ayuda de NANO de la siguiente manera

```
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# nano index.php
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb#
```

```
root@ruiso-VirtualBox: /home/ruiso/Escritorio/servicioweb
GNU nano 4.8           index.php           Modificado
<?php
phpinfo();
?>
```

Confirmamos que nuestro servidor de PHP este funcionando correctamente entrando a la ruta local de nuestra red, pero por el puerto de la imagen php que iniciamos, en este caso el puerto 1000.

Ubuntu [Corriendo] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

Actividades Navegador web Firefox 30 de may 12:58

phpinfo() x +

localhost:1000 ☆ ☰ ⌂ ⌂ ⌂

## PHP Version 7.0.33



|  |   |
|--|---|
| <b>System</b>                                  | Linux 0f931837beaf 5.13.0-44-generic #49~20.04.1-Ubuntu SMP Wed May 18 18:44:28 UTC 2022 x86_64   |
| <b>Build Date</b>                              | Dec 29 2018 06:50:15  |
| <b>Configure Command</b>                       | './configure' '--build=x86_64-linux-gnu' '--with-config-file-path=/usr/local/etc/php' '--with-config-file-scan-dir=/usr/local/etc/php/conf.d' '--enable-option-checking=fatal' '--with-mhash' '--enable-ftp' '--enable-mbstring' '--enable-mysqlind' '--with-curl' '--with-libedit' '--with-openssl' '--with-zlib' '--with-libdir=lib/x86_64-linux-gnu' '--with-apxs2' '--disable-cgi' 'build_alias=x86_64-linux-gnu' |
| <b>Server API</b>                              | Apache 2.0 Handler  |
| <b>Virtual Directory Support</b>               | disabled  |
| <b>Configuration File (php.ini) Path</b>       | /usr/local/etc/php  |
| <b>Loaded Configuration File</b>               | (none)  |
| <b>Scan this dir for additional .ini files</b> | /usr/local/etc/php/conf.d   |
| <b>Additional .ini files parsed</b>            | (none)  |
| <b>PHP API</b>                                 | 20151012  |
| <b>PHP Extension</b>                           | 20151012  |
| <b>Zend Extension</b>                          | 320151012   |
| <b>Zend Extension Build</b>                    | API320151012,NTS  |
| <b>PHP Extension Build</b>                     | API20151012,NTS   |
| <b>Debug Build</b>                             | no  |
| <b>Thread Safety</b>                           | disabled  |
| <b>Zend Signal Handling</b>                    | disabled  |
| <b>Zend Memory Manager</b>                     | enabled   |
| <b>Zend Multibyte Support</b>                  | provided by mbstring  |
| <b>IPv6 Support</b>                            | enabled   |
| <b>DTrace Support</b>                          | disabled  |
| <b>Registered PHP Streams</b>                  | https, ftps, compress.zlib, php, file, glob, data, http, ftp, phar  |
| <b>Registered Stream Socket Transports</b>     | tcp, udp, unix, udg, ssl, sslv2, tls, tlsv1.0, tlsv1.1, tlsv1.2   |
| <b>Registered Stream Filters</b>               | zlib.* , convert.iconv.* , string.rot13 , string.toupper , string.tolower , string.strip_tags , convert.* , consumed , dechunk  |

This program makes use of the Zend Scripting Language Engine:  
Zend Engine v3.0.0, Copyright (c) 1998-2017 Zend Technologies

**zend engine**

Ahora vamos a comprobar del mismo modo que nuestro servidore de mysql se encuentra funcionando correctamente, adicionalmente crearemos la base de datos como ROOT del sistema y de la imagen, seguidamente a eso insertaremos la clave, misma que destinamos en el momento de crear la imagen de MYSQL, en este caso sería password. Estando dentro de la imagen basta con utilizar los comandos en lenguaje SQL para MYSQL necesarios para crear la tabla de usuarios que podrán acceder a nuestro servidor y a su vez el usuario específicamente.

```

ruiso@ruiso-VirtualBox:~$ sudo docker exec -i -t basedatos /bin/bash
[sudo] contraseña para ruiso:
root@09ed8013b16a:/# ls
bin  dev          entrypoint.sh  home  lib64  mnt  proc  run  srv  tmp  var
boot docker-entrypoint-initdb.d  etc      lib   media  opt  root  sbin  sys  usr
root@09ed8013b16a:/# mysql -u root -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.7.38 MySQL Community Server (GPL)

Copyright (c) 2000, 2022, 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> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0.10 sec)

mysql> 

```

Dentro de la base de datos y como root de mysql, creamos la base de datos de ingreso para los usuarios (comandos SQL de MySQL).

```

mysql> USE usuarios;
Database changed
mysql> CREATE TABLE clientes (
    -> username VARCHAR(20) PRIMARY KEY NOT NULL,
    -> nombre VARCHAR(30),
    -> correo VARCHAR(50),
    -> contra VARCHAR(20)
    -> );
Query OK, 0 rows affected (0.14 sec)

mysql> SHOW TABLES;
+-----+
| Tables_in_usuarios |
+-----+
| clientes           |
+-----+
1 row in set (0.00 sec)

mysql> 
mysql> INSERT INTO clientes VALUES('ruiso','luis felipe','luis.narvaez@usantoto.edu.co','6789');
Query OK, 1 row affected (0.11 sec)

mysql> SELECT *
    -> FROM clientes;
+-----+-----+-----+
| username | nombre    | correo          | contra |
+-----+-----+-----+
| ruiso    | luis felipe | luis.narvaez@usantoto.edu.co | 6789   |
+-----+-----+-----+
1 row in set (0.00 sec)

mysql> 

```

Una vez tenemos nuestro usuario en la base de datos, vamos a editar nuestro archivo PHP 4, esto con el fin de comprobar la conexión de la base de datos.

```

root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# nano index.php
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# 

```

The terminal window shows the following PHP code:

```

root@ruiso-VirtualBox: /home/ruiso/Escritorio/servicioweb
GNU nano 4.8 index.php Modificado
<?php

//phpinfo();

//Create Connection
$conn = mysqli_connect("basedatos:3306","root","password","usuarios");

//Check Connection
if(!$conn){
    die("connection failed: ".mysqli_connect_error());
}
echo "Connected Successfully";

?>

```

The browser window shows the URL `localhost:1000/` and displays the following error message:

**Fatal error:** Uncaught Error: Call to undefined function mysqli\_connect() in `/var/www/html/index.php`:6 Stack trace: #0 {main} thrown in `/var/www/html/index.php` on line **6**

El error que tenemos es generado ya que la función de `mysqli_connect()` no se encuentra definida en el servidor que tenemos instalado en php, para tener la extensión y poder conectarse a php debemos agregarla, para esto hacemos lo siguiente:

```

ruiso@ruiso-VirtualBox:~/Escritorio/servicioweb$ ls
database index.php
ruiso@ruiso-VirtualBox:~/Escritorio/servicioweb$ 

```

a) Abrir el contenedor php por medio del comando `docker exec -i -t miservidorphp /bin/bash`

```

ruiso@ruiso-VirtualBox:~/Escritorio/servicioweb$ sudo docker exec -i -t miservidorphp /bin/bash
[sudo] contraseña para ruiso:
root@0f931837beaf:/var/www/html#

```

b) Ejecutamos el comando `cd /` y luego `ls` para listar los archivos.

```

root@0f931837beaf:/var/www/html# ls
database index.php
root@0f931837beaf:/var/www/html# cd /
root@0f931837beaf:/# ls
bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
root@0f931837beaf:/# 

```

c) Con el comando `docker-php-ext-install mysqli` instalamos la extensión.

```

root@0f931837beaf:/# docker-php-ext-install mysqli
Configuring for:
PHP Api Version:      20151012
Zend Module Api No:   20151012
Zend Extension Api No: 320151012
checking for grep that handles long lines and -e... /bin/grep
checking for egrep... /bin/grep -E

```

```
find . -name '*.la' -o -name '*.a' | xargs rm -f  
find . -name '*.so' | xargs rm -f  
find . -name .libs -a -type d|xargs rm -rf  
rm -f libphp.la      modules/* libs/*  
root@0f931837beaf:/#
```

- d) Una vez instalada la extensión tenemos que copiar la dirección en la que se guardó este archivo, el archivo se llama mysqli.os.

```
more information, such as the ca(1) and ca.so(8) manual pages.  
-----
```

```
Build complete.  
Don't forget to run 'make test'.
```

```
Installing shared extensions:      /usr/local/lib/php/extensions/no-debug-non-zts-20151012/  
Installing header files:         /usr/local/include/php/  
find -name \*.acno -o -name \*.acda | xargs rm -f
```

- e) Copiamos la ruta que aparece después de: "Installing shared extensions: "

</usr/local/lib/php/extensions/no-debug-non-zts-20151012>

- f) Nos ubicamos en la ruta usr/local/etc/php por medio del comando cd y luego damos ls con el fin listar los archivos que se encuentran.

```
root@0f931837beaf:/#  
root@0f931837beaf:/# cd /usr/local/lib/php/extensions  
root@0f931837beaf:/usr/local/lib/php/extensions# ls  
no-debug-non-zts-20151012  
root@0f931837beaf:/usr/local/lib/php/extensions# cd no-debug-non-zts-20151012/  
root@0f931837beaf:/usr/local/lib/php/extensions/no-debug-non-zts-20151012# ls  
mysqli.so  opcache.so  
root@0f931837beaf:/usr/local/lib/php/extensions/no-debug-non-zts-20151012#
```

- g) Debemos abrir el archivo php.ini-development por medio del comando nano, pero para hacer uso de este primero debemos instalarlo por medio del comando aptget update y luego apt-get install nano

```
root@0f931837beaf:/usr/local/lib/php/extensions/no-debug-non-zts-20151012#  
root@0f931837beaf:/usr/local/lib/php/extensions/no-debug-non-zts-20151012# cd /  
root@0f931837beaf:# cd usr/local/etc/php  
root@0f931837beaf:/usr/local/etc/php# ls  
conf.d  php.ini-development  php.ini-production  
root@0f931837beaf:/usr/local/etc/php#  
  
root@0f931837beaf:/usr/local/etc/php# nano php.ini-development  
bash: nano: command not found  
root@0f931837beaf:/usr/local/etc/php# apt-get install nano  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
E: Unable to locate package nano  
root@0f931837beaf:/usr/local/etc/php# apt-get update  
Get:1 http://security.debian.org/debian-security stretch/updates InRelease [53.0 kB]  
Get:2 http://security.debian.org/debian-security stretch/updates/main amd64 Packages [770 kB]  
Ign:3 http://deb.debian.org/debian stretch InRelease  
Get:4 http://deb.debian.org/debian stretch-updates InRelease [93.6 kB]  
Get:5 http://deb.debian.org/debian stretch Release [118 kB]  
Get:6 http://deb.debian.org/debian stretch Release.gpg [3177 B]  
Get:7 http://deb.debian.org/debian stretch/main amd64 Packages [7080 kB]  
Fetched 8117 kB in 3s (2417 kB/s)  
Reading package lists... Done  
root@0f931837beaf:/usr/local/etc/php#
```

```

root@0f931837beaf:/usr/local/etc/php#
root@0f931837beaf:/usr/local/etc/php# apt-get install nano
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  spell
The following NEW packages will be installed:
  nano
0 upgraded, 1 newly installed, 0 to remove and 71 not upgraded.
Need to get 485 kB of archives.
After this operation, 2092 kB of additional disk space will be used.
Get:1 http://deb.debian.org/debian stretch/main amd64 nano amd64 2.7.4-1 [485 kB]
Fetched 485 kB in 0s (1041 kB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package nano.
(Reading database ... 13099 files and directories currently installed.)
Preparing to unpack .../nano_2.7.4-1_amd64.deb ...
Unpacking nano (2.7.4-1) ...
Setting up nano (2.7.4-1) ...
update-alternatives: using /bin/nano to provide /usr/bin/editor (editor) in auto mode
update-alternatives: warning: skip creation of /usr/share/man/man1/editor.1.gz because associated file /usr/share/man/man1/nano.1.gz (of link group editor) doesn't exist
update-alternatives: using /bin/nano to provide /usr/bin/pico (pico) in auto mode
update-alternatives: warning: skip creation of /usr/share/man/man1/pico.1.gz because associated file /usr/share/man/man1/nano.1.gz (of link group pico) doesn't exist
root@0f931837beaf:/usr/local/etc/php#

```

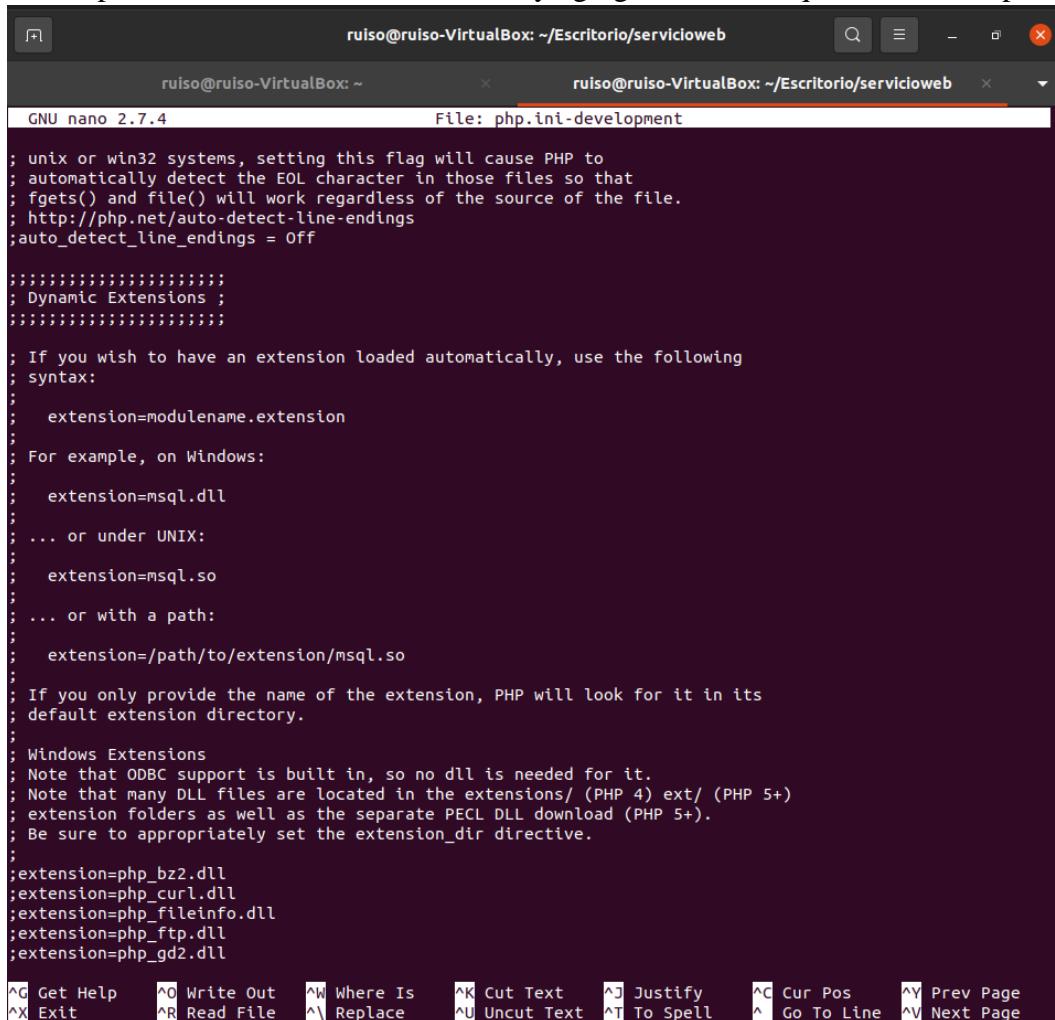
- h) Ahora si ejecutamos el comando nano php.ini-development que abrirá el archivo.

```

root@0f931837beaf:/usr/local/etc/php#
root@0f931837beaf:/usr/local/etc/php# ls
conf.d  php.ini-development  php.ini-production
root@0f931837beaf:/usr/local/etc/php# nano php.ini-development [ENTER]

```

- i) Buscamos el apartado de extensiones dinámicas y agregamos la ruta que teníamos copiada.



```

GNU nano 2.7.4                               File: php.ini-development

; unix or win32 systems, setting this flag will cause PHP to
; automatically detect the EOL character in those files so that
; fgets() and file() will work regardless of the source of the file.
; http://php.net/auto-detect-line-endings
;auto_detect_line_endings = Off

;;;;;;;;;;;;;;;;;;;
; Dynamic Extensions ;
;;;;;;;;;;;;;;;;;;

; If you wish to have an extension loaded automatically, use the following
; syntax:
;
; extension=modulename.extension
;
; For example, on Windows:
;
; extension=msql.dll
;
; ... or under UNIX:
;
; extension=msql.so
;
; ... or with a path:
;
; extension=/path/to/extension/msql.so

; If you only provide the name of the extension, PHP will look for it in its
; default extension directory.

; Windows Extensions
; Note that ODBC support is built in, so no dll is needed for it.
; Note that many DLL files are located in the extensions/ (PHP 4) ext/ (PHP 5+)
; extension folders as well as the separate PECL DLL download (PHP 5+).
; Be sure to appropriately set the extension_dir directive.

;extension=php_bz2.dll
;extension=php_curl.dll
;extension=php_fileinfo.dll
;extension=php_ftp.dll
;extension=php_gd2.dll

^G Get Help      ^W Where Is      ^K Cut Text      ^J Justify      ^C Cur Pos      ^Y Prev Page
^X Exit          ^R Read File     ^\ Replace      ^U Uncut Text    ^T To Spell     ^^ Go To Line   ^V Next Page

```

j) Para salir y guardar los cambios presionamos: control + x, y, Enter. Cambiamos de la imagen 1 a la 2.

```
;;
; ... or with a path:
;
; extension=/path/to/extension/msql.so
;
```

Cambiamos a ...

```
;;
; ... or with a path:
;
extension=/usr/local/lib/php/extensions/no-debug-non-zts-20151012/msqli.os
;
; If you only provide the name of the extension, PHP will look for it in its
; default extension directory.
;
```

[CTRL]+[O]

[ENTER]

[CTRL]+[X]

k) Realizamos los pasos de los apartados i) y j) después de abrir el archivo php.ini.production con el comando nano.

```
root@0f931837beaf:/usr/local/etc/php#
root@0f931837beaf:/usr/local/etc/php# ls
conf.d  php.ini-development  php.ini-production
root@0f931837beaf:/usr/local/etc/php# nano php.ini-production
```

[ENTER]

```
;;;;;;
; Dynamic Extensions ;
;;;;;;

; If you wish to have an extension loaded automatically, use the following
; syntax:
;

; extension=modulename.extension
;

; For example, on Windows:
;

; extension=msql.dll
;

; ... or under UNIX:
;

; extension=msql.so
;

; ... or with a path:
;

; extension=/path/to/extension/msql.so
;

; If you only provide the name of the extension, PHP will look for it in its
; default extension directory.
;
```

Cambiamos a la siguiente dirección ...

```
; ... or with a path:  
;  
extension=/usr/local/lib/php/extensions/no-debug-non-zts-20151012/msqli.os  
;  
; If you only provide the name of the extension, PHP will look for it in its  
; default extension directory.  
;
```

[CTRL]+[O]

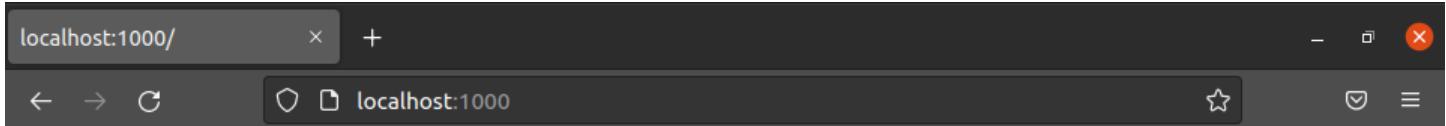
[ENTER]

[CTRL]+[X]

- l) Reiniciamos el servicio con el comando docker restart miservidorphp
- m) Verificamos que los dos contenedores estén corriendo con el comando docker ps

```
ruiso@ruiso-VirtualBox:~/Escritorio/servicioweb$ sudo docker ps
[sudo] contraseña para ruiso:
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
0f931837beaf php:7.0-apache "docker-php-entrypoi..." 5 hours ago Up 5 hours 0.0.0.0:1000->80/tcp, :::1000->80/tcp miservidorphp
09ed8013b16a mysql:5.7 "docker-entrypoint.s..." 41 hours ago Up 5 hours 33060/tcp, 0.0.0.0:3307->3306/tcp, :::3307->3306/tcp basedatos
ruiso@ruiso-VirtualBox:~/Escritorio/servicioweb$ sudo docker restart miservidorphp
miservidorphp
ruiso@ruiso-VirtualBox:~/Escritorio/servicioweb$ sudo docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
0f931837beaf php:7.0-apache "docker-php-entrypoi..." 5 hours ago Up 5 seconds 0.0.0.0:1000->80/tcp, :::1000->80/tcp miservidorphp
09ed8013b16a mysql:5.7 "docker-entrypoint.s..." 41 hours ago Up 5 hours 33060/tcp, 0.0.0.0:3307->3306/tcp, :::3307->3306/tcp basedatos
ruiso@ruiso-VirtualBox:~/Escritorio/servicioweb$
```

- n) En el navegador recargamos la ruta localhost:1000 y observamos que aparece el mensaje de conexión exitosa.



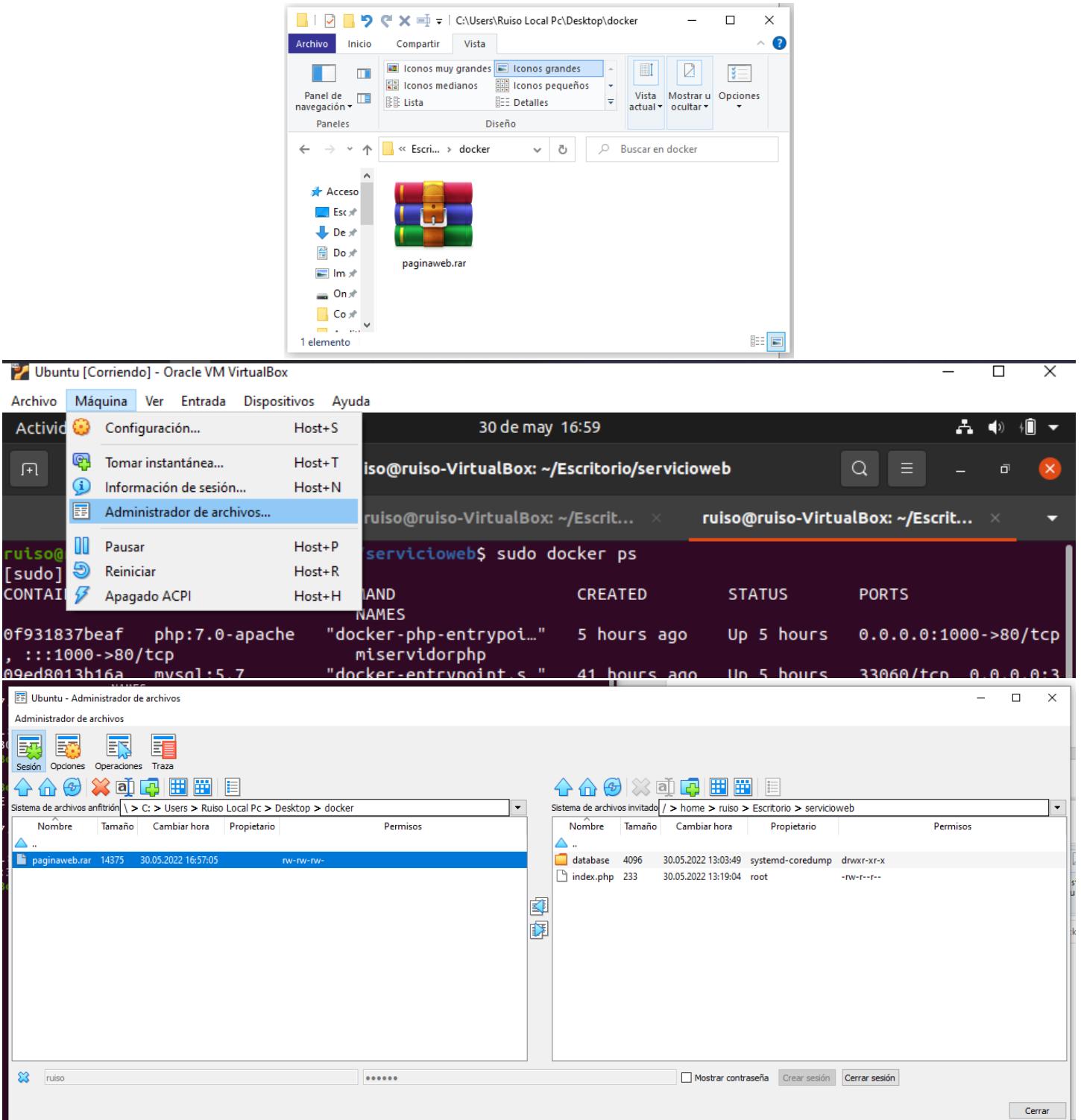
Para razones prácticas de la realización de este taller, se reemplazara el archivo index.php con los archivos descomprimidos del campus virtual de la Universidad Santo Tomas seccional Tunja para la materia de Devops.

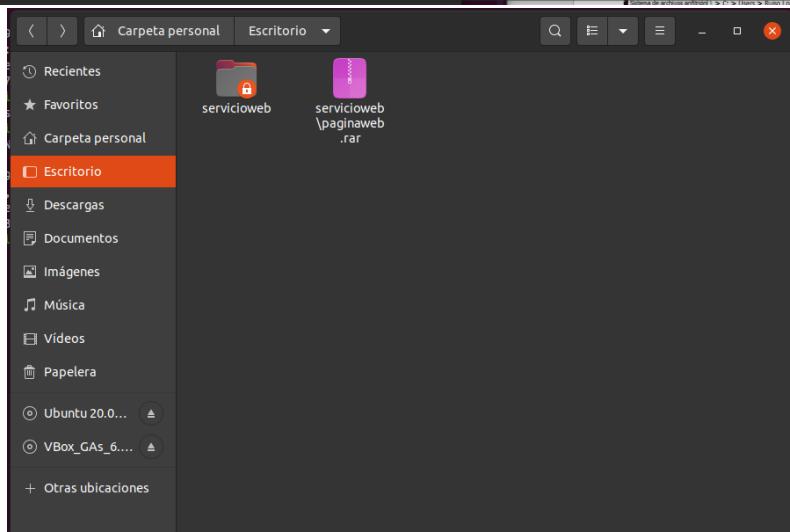
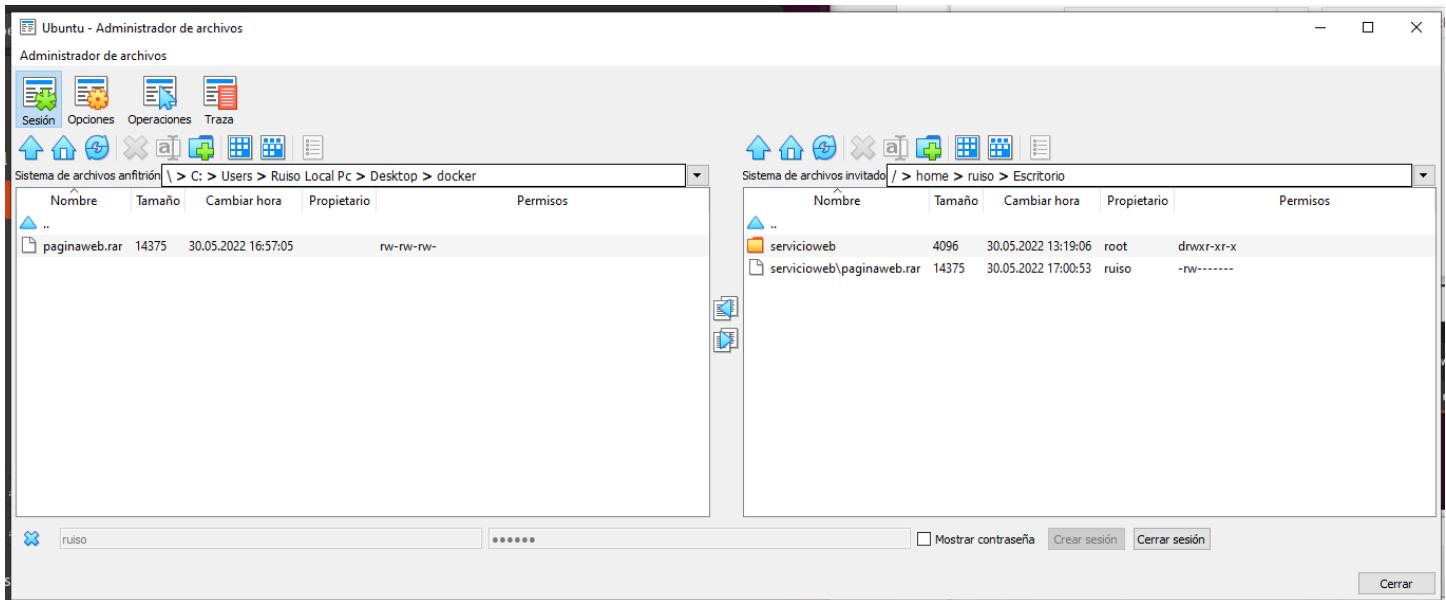
A screenshot of a learning management system interface. At the top, a green bar says '¿Qué tanto aprendí?'. Below it, there are two completed tasks: 'paginaweb' and 'Taller Final Tercer Corte', each marked with a green checkmark. In the center, there's a box labeled 'TAREA' containing 'Taller Tercer Corte'. At the bottom, a file named 'paginaweb.rar' is shown, along with a 'Mostrar todo' button and a close 'x' button.

Debido a que descargamos el archivo del Campus Virtual en nuestro Sistema Operativo de Base, debemos trasladarlo hasta la Maquina Virtual, esto lo hacemos con ayuda de la herramienta de Virtual Box de Administrador de archivos, ahí crearemos una sección de la máquina virtual.

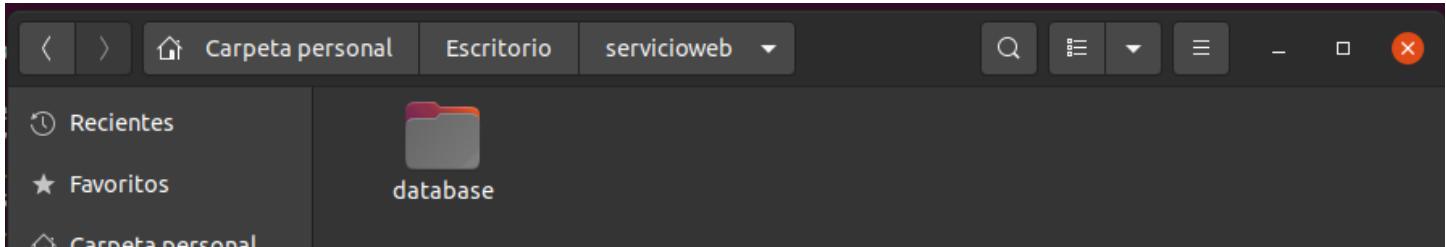
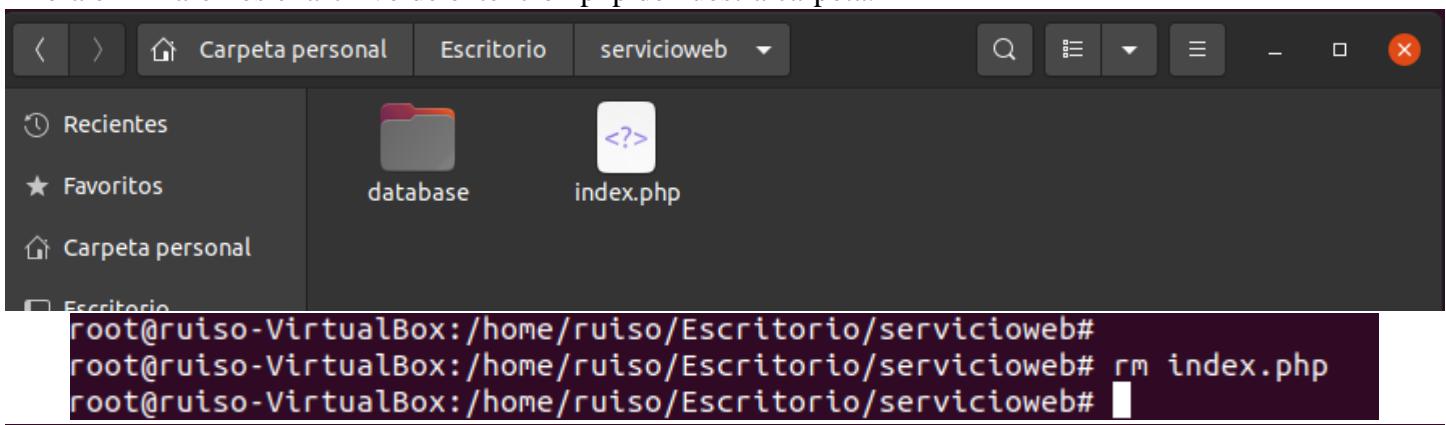
En la parte izquierda estará el árbol de archivos de nuestra maquina base y en el lado derecho la de nuestra maquina virtual, para traspasar archivos de lado a lado, basta con ubicar las direcciones o ubicaciones de cada

máquina, seleccionar el objeto a traspasar de nuestro interés y mandarlo de un lado a otro con ayuda del sentido de las flechas del centro de la ventana del Administrador de archivos.





Ahora eliminaremos el archivo de extencion php de nuestra carpeta.



Ahora moveremos el comprimido a la carpeta de nuestro inter  s.

```

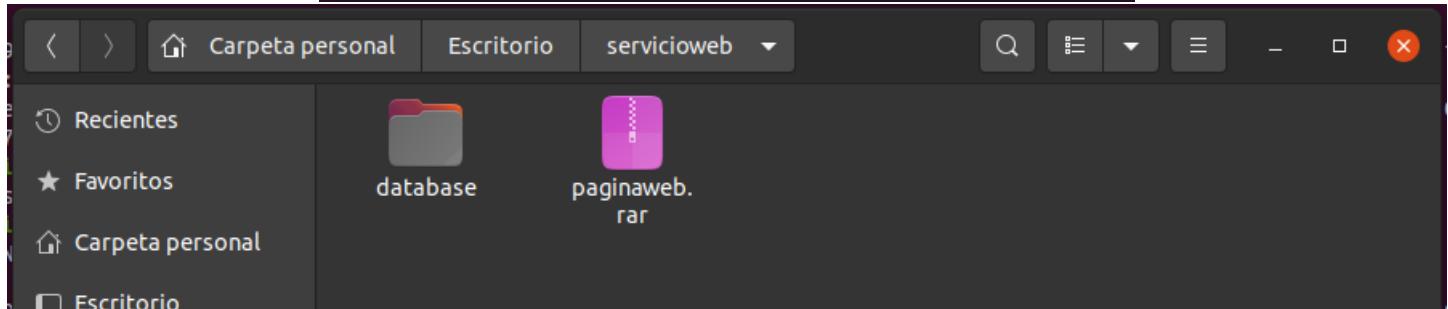
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb#
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# ls
database
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# cd ..
root@ruiso-VirtualBox:/home/ruiso/Escritorio# ls
servicioweb 'servicioweb\paginaweb.rar'
root@ruiso-VirtualBox:/home/ruiso/Escritorio#



root@ruiso-VirtualBox:/home/ruiso/Escritorio#
root@ruiso-VirtualBox:/home/ruiso/Escritorio# ls
paginaweb.rar servicioweb
root@ruiso-VirtualBox:/home/ruiso/Escritorio# mv paginaweb.rar /home/ruiso/Escritorio/servicioweb
root@ruiso-VirtualBox:/home/ruiso/Escritorio#



root@ruiso-VirtualBox:/home/ruiso/Escritorio#
root@ruiso-VirtualBox:/home/ruiso/Escritorio# ls
servicioweb
root@ruiso-VirtualBox:/home/ruiso/Escritorio# cd servicio/
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicio# ls
database paginaweb.rar
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicio#
```



Para descomprimir el archivo debemos instalar el complemento de rar para Linux.

```

ruiso@ruiso-VirtualBox:~/Escritorio/servicioweb$ sudo apt-get install rar
[sudo] contraseña para ruiso:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
El paquete indicado a continuación se instaló de forma automática y ya no es necesario.
  libfwupdplugin1
Utilice «sudo apt autoremove» para eliminarlo.
Paquetes sugeridos:
  unrar
Se instalarán los siguientes paquetes NUEVOS:
  rar
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 2 no actualizados.
Se necesita descargar 295 kB de archivos.
Se utilizarán 799 kB de espacio de disco adicional después de esta operación.
Des:1 http://co.archive.ubuntu.com/ubuntu focal/multiverse amd64 rar amd64 2:5.5.0-1build1 [295 kB]
Descargados 295 kB en 1s (251 kB/s)
Seleccionando el paquete rar previamente no seleccionado.
(Leyendo la base de datos ... 184230 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../rar_2%3a5.5.0-1build1_amd64.deb ...
Desempaquetando rar (2:5.5.0-1build1) ...
Configurando rar (2:5.5.0-1build1) ...
Procesando disparadores para man-db (2.9.1-1) ...
ruiso@ruiso-VirtualBox:~/Escritorio/servicioweb$
```

```

ruiso@ruiso-VirtualBox:~/Escritorio/servicioweb$ sudo apt-get install unrar
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
El paquete indicado a continuación se instaló de forma automática y ya no es necesario.
  libfwupdplugin1
Utilice «sudo apt autoremove» para eliminarlo.
Se instalarán los siguientes paquetes NUEVOS:
  unrar
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 2 no actualizados.
Se necesita descargar 113 kB de archivos.
Se utilizarán 406 kB de espacio de disco adicional después de esta operación.
Des:1 http://co.archive.ubuntu.com/ubuntu focal/multiverse amd64 unrar amd64 1:5.6.6-2build1 [113 kB]
Descargados 113 kB en 1s (119 kB/s)
Seleccionando el paquete unrar previamente no seleccionado.
(Leyendo la base de datos ... 184244 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../unrar_1%3a5.6.6-2build1_amd64.deb ...
Desempaquetando unrar (1:5.6.6-2build1) ...
Configurando unrar (1:5.6.6-2build1) ...
update-alternatives: utilizando /usr/bin/unrar-nonfree para proveer /usr/bin/unrar (unrar) en modo automático
Procesando disparadores para man-db (2.9.1-1) ...
ruiso@ruiso-VirtualBox:~/Escritorio/servicioweb$ 

```

Ahora podemos descomprimir el archivo.

```

root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb#
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# ls
database paginaweb paginaweb.rar
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# rm -r paginaweb
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# unrar x paginaweb.rar /home/ruiso/Escritorio/se
rvicioweb

```

```
UNRAR 5.61 beta 1 freeware      Copyright (c) 1993-2018 Alexander Roshal
```

Extracting from paginaweb.rar

```

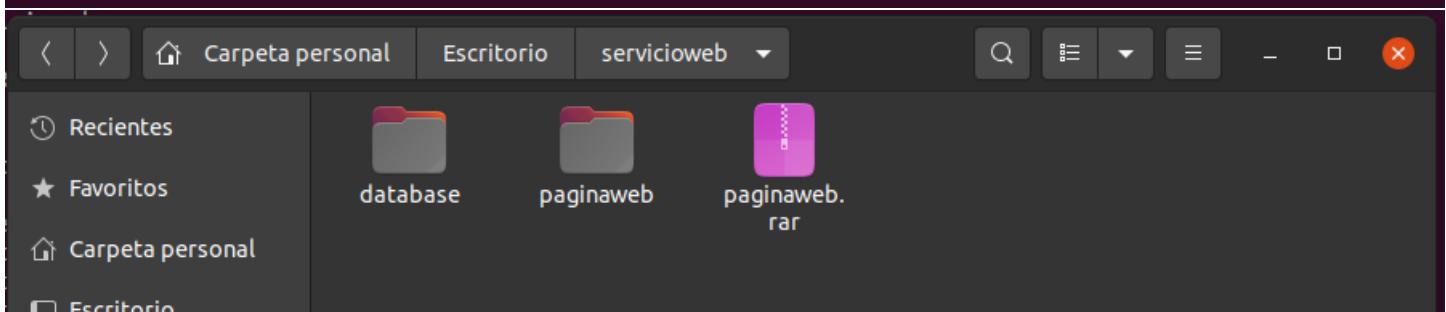
Creating   /home/ruiso/Escritorio/servicioweb/paginaweb          OK
Extracting  /home/ruiso/Escritorio/servicioweb/paginaweb/conexion.php  OK
Extracting  /home/ruiso/Escritorio/servicioweb/paginaweb/data_maintenance_isometric.svg  OK
Extracting  /home/ruiso/Escritorio/servicioweb/paginaweb/estiloIndex.css  OK
Extracting  /home/ruiso/Escritorio/servicioweb/paginaweb/estiloRegistro.css  OK
Extracting  /home/ruiso/Escritorio/servicioweb/paginaweb/estilosesion.css  OK
Extracting  /home/ruiso/Escritorio/servicioweb/paginaweb/index.html  OK
Extracting  /home/ruiso/Escritorio/servicioweb/paginaweb/registro.html  OK
Extracting  /home/ruiso/Escritorio/servicioweb/paginaweb/registro.php  OK
Extracting  /home/ruiso/Escritorio/servicioweb/paginaweb/sesion.html  OK
Extracting  /home/ruiso/Escritorio/servicioweb/paginaweb/undraw_male_avatar_323b.svg  OK
Extracting  /home/ruiso/Escritorio/servicioweb/paginaweb/undraw_sign_in_e6hj.svg  OK
All OK

```

```

root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# ls
database paginaweb paginaweb.rar
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# 

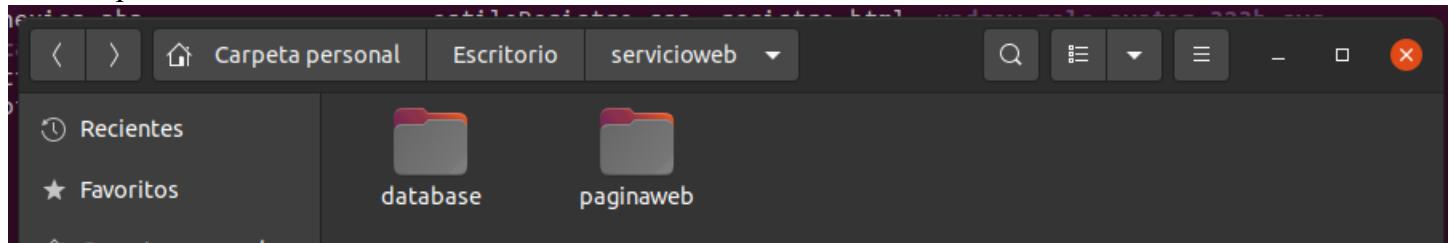
```



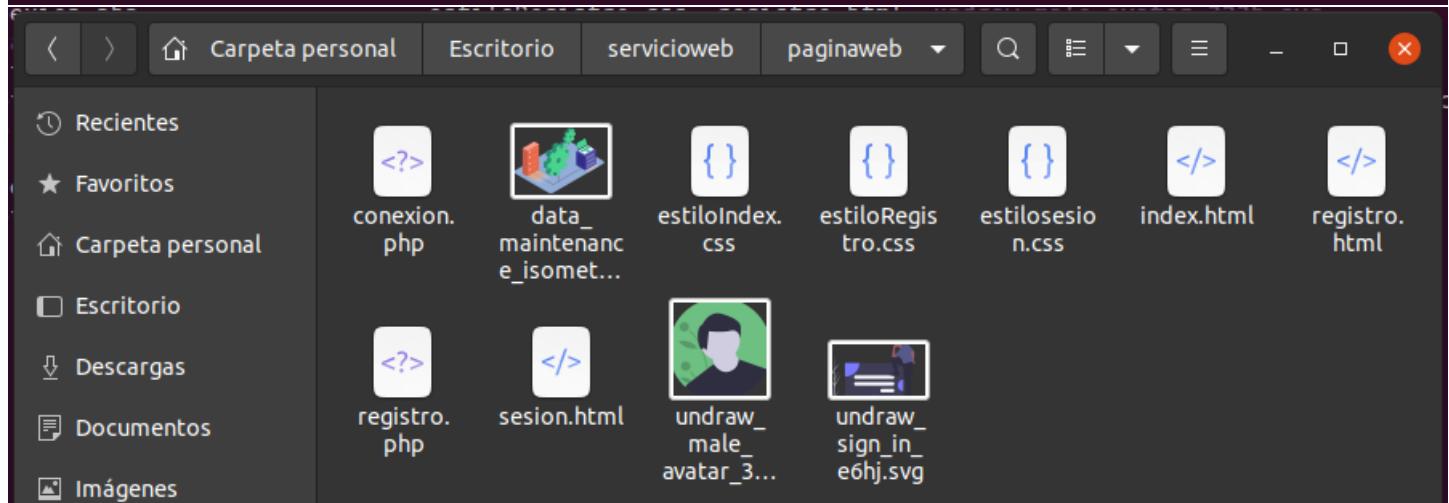
Eliminamos de paso el rar.

```
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# ls
database paginaweb paginaweb.rar
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# rm paginaweb.rar
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# ls
database paginaweb
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb#
```

Movemos todos los archivos dentro de la carpeta “paginaweb” por fuera de esta para que queden al mismo nivel de la carpeta “database”.



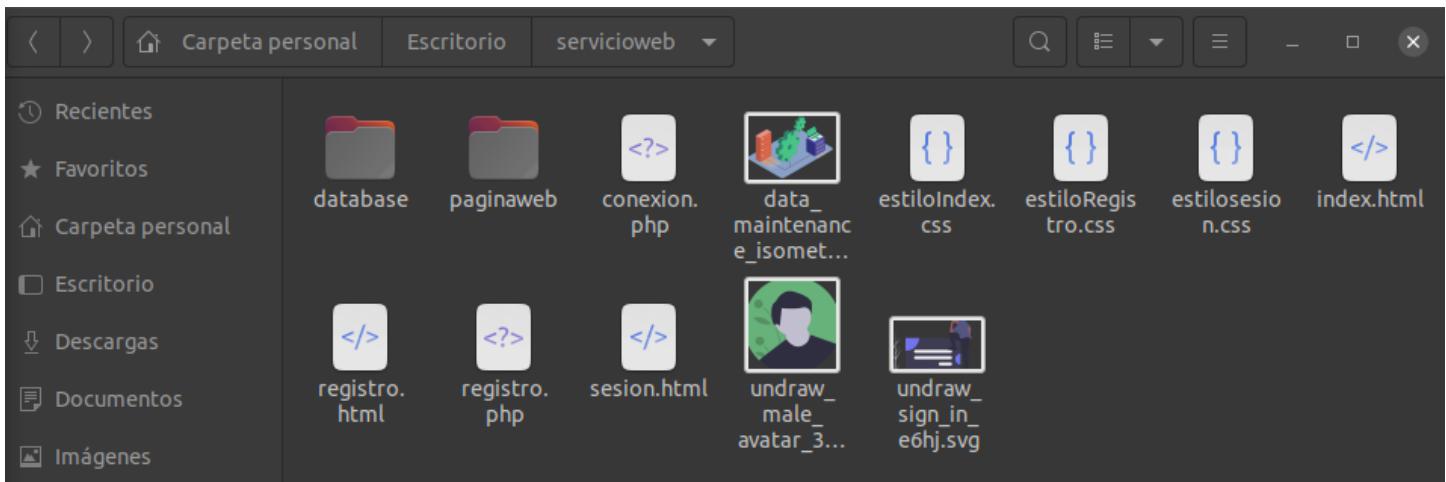
```
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# ls
database paginaweb
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# cd paginaweb/
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb/paginaweb# ls
connexion.php           estiloRegistro.css  registro.html  undraw_male_avatar_323b.svg
data_maintenance_isometric.svg  estilosesion.css   registro.php   undraw_sign_in_e6hj.svg
estiloIndex.css          index.html        sesion.html
```



Para mover los archivos debemos utilizar el siguiente comando

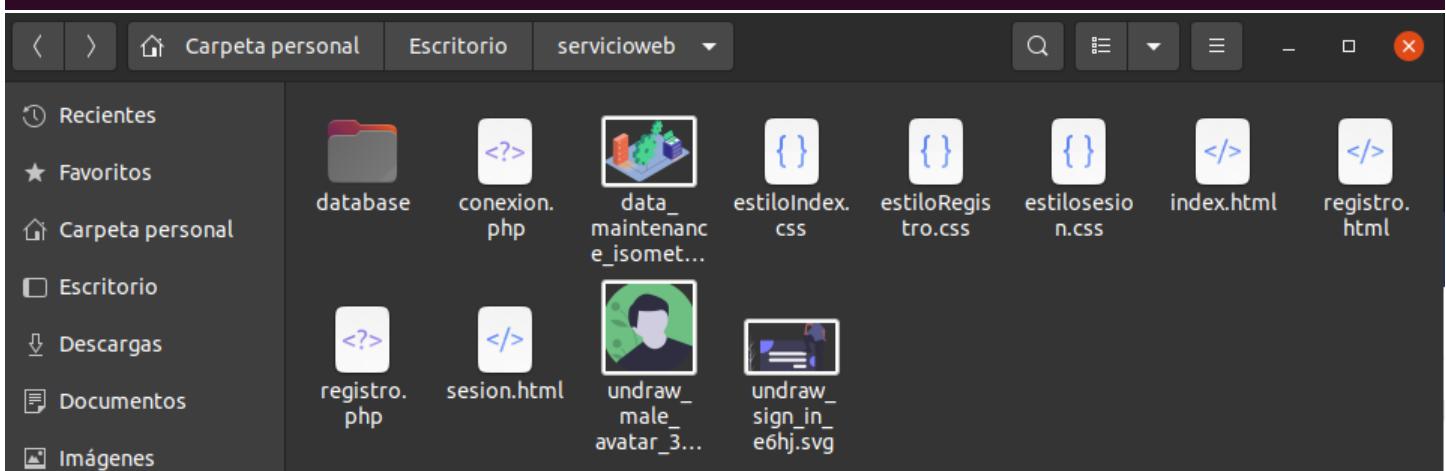
```
mv nombreArchivo.extencion ruta/donde/se/va/a/mover
```

```
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb/paginaweb# ls
registro.html  registro.php  sesion.html  undraw_male_avatar_323b.svg  undraw_sign_in_e6hj.svg
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb/paginaweb# mv registro.html /home/ruiso/Escritorio/servicioweb
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb/paginaweb# mv registro.php /home/ruiso/Escritorio/servicioweb
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb/paginaweb# mv undraw_male_avatar_323b.svg /home/ruiso/Escritorio/servicioweb
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb/paginaweb# mv undraw_sign_in_e6hj.svg /home/ruiso/Escritorio/servicioweb
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb/paginaweb# mv sesion.html /home/ruiso/Escritorio/servicioweb
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb/paginaweb#
```



Eliminamos la carpeta ahora vacía.

```
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb/paginaweb# ls
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb/paginaweb# cd ..
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# ls
conexion.php          estiloRegistro.css  registro.html           undraw_sign_in_e6hj.svg
database              estilosesion.css   registro.php          undraw_male_avatar_323b.svg
data_maintenance_isometric.svg  index.html      sesion.html
estiloIndex.css        paginaweb       undraw_male_avatar_3...
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# rm -r paginaweb/
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb#
```



Ahora abriremos el archivo denominado “conexión.php” y modificamos los parámetros de conexión de la función “mysql\_connect”.

```
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# ls
conexion.php          estiloIndex.css  index.html      sesion.html
database              estiloRegistro.css  registro.html  undraw_male_avatar_323b.svg
data_maintenance_isometric.svg  estilosesion.css  registro.php  undraw_sign_in_e6hj.svg
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# nano conexion.php
```

Cambiar de esto

```
GNU nano 4.8                                     conexion.php
<?php
$nombre = $_POST['nombre'];
$pass = $_POST['password'];

$con = mysqli_connect("localhost","root","","usuarios");
$query = "select * from clientes where username='".$nombre."' and contra='".$pass."'";
```

A esto

```

GNU nano 4.8                               connexion.php                                Modificado
<?php
$nombre = $_POST['nombre'];
$pass = $_POST['password'];

$con = mysqli_connect("basedatos:3306","root","","usuarios");
$query = "select * from clientes where username='".$nombre."' and contra='".$pass."'";
if($con){
    $result = mysqli_query($con, $query);

    if ($result->num_rows > 0) {
        // output data of each row
        while($row = $result->fetch_assoc()) {
            $nombre = $row['nombre'];
            $username = $row['username'];
            $password = $row['contra'];
            $correo = $row['correo'];

            echo '<!DOCTYPE html>
<html lang="es">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=1, initial-scale=1.0">
    <title>LoyoSteve Inc. | Sesion</title>
    <link rel="stylesheet" href="estilosesion.css">
    <link rel="shortcut icon" href="undraw_male_avatar_323b.svg" type="image/x-icon">
</head>

<body>
    <header>
        <div class="contentheader">
            
            <dir></dir>
            <nav>
                <ul class="menu">
                    <li><a href="index.html">Inicio</a></li>
                    <li><a href="index.html">Portafolio</a></li>
                    <li><a href="index.html">Servicios</a></li>
                    <li><a href="index.html">Contactanos</a></li>
                </ul>
            </nav>
        </div>
    </header>
</body>

```

AG Ver ayuda   ^O Guardar   ^W Buscar   ^K Cortar Texto ^J Justificar   ^C Posición   M-U Deshacer  
 AX Salir   ^R Leer fich.   ^\ Reemplazar   ^U Pegar   ^T Ortografía   ^ I Ir a línea   M-E Rehacer

[CTRL]+O → [ENTER] → [CTRL]+[X]. Ahora abrimos el documento denominado “registro.php” y cambiamos los parámetros de conexión de la función “mysql\_connect”.

```

root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# ls
connexion.php           estiloIndex.css   index.html    sesion.html
database               estiloRegistro.css registro.html undraw_male_avatar_323b.svg
data_maintenance_isometric.svg estilosesion.css registro.php undraw_sign_in_e6hj.svg
root@ruiso-VirtualBox:/home/ruiso/Escritorio/servicioweb# nano registro.php

```

Cambiamos de esto

```

$con = mysqli_connect("localhost","root","","usuarios");
$query = "insert into clientes values('$username', '$nombre', '$correo', '$contra')";

if($con){

```

A esto

```

GNU nano 4.8                               registro.php                                Modificado
<?php

$nombre = $_POST['nombre'];
$correo = $_POST['correo'];
$username = $_POST['username'];
$contra = $_POST['contra'];

$con = mysqli_connect("basedatos:3306","root","password","usuarios");
$query = "insert into clientes values('$username', '$nombre','$correo','$contra')";

if($con){
    $result = mysqli_query($con, $query);
    if($result){
        echo'<script type="text/javascript">
            alert("Registro Correcto");
            window.location.href="index.html";
        </script>';
    }else{
        echo'<script type="text/javascript">
            alert("Algo salio mal :/");
            window.location.href="registro.html";
        </script>';
    }
}else{
    echo "no conectado bro..";
}

?>

```

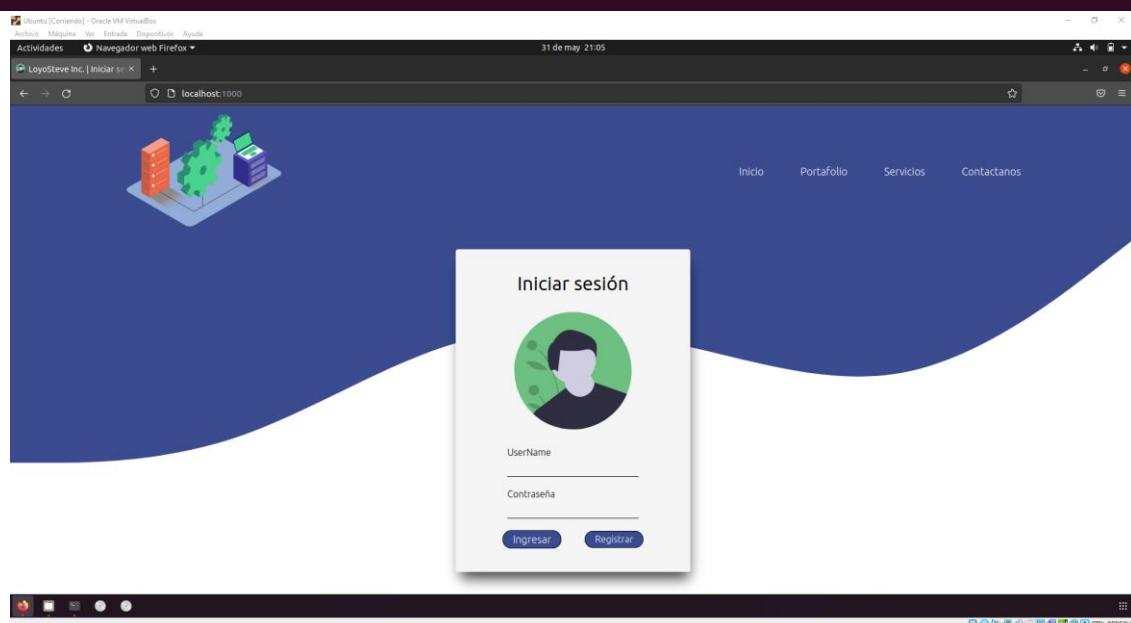
[CTRL]+[O] → [ENTER] → [CTRL]+[X]

Reiniciamos los servicios que estamos utilizando y observamos dentro de un navegador con ruta “localhost:1000” que tenemos una página de acceso, en ella podemos probar los usuarios que especificamos en nuestra base de datos.

```

ruiso@ruiso-VirtualBox:~$ sudo docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
0f931837beaf php:7.0-apache "docker-php-entrypoi..." 33 hours ago Exited (255) 27 hours ago 0.0.0.0:1000->80/tcp, :::1000->80/tcp miservidorphp
09ed8013b16a mysql:5.7 "docker-entrypoint.s..." 2 days ago Exited (255) 27 hours ago 33060/tcp, 0.0.0.0:3307->3306/tcp, :::3307->3306/tcp basedatos
ruiso@ruiso-VirtualBox:~$ 
ruiso@ruiso-VirtualBox:~$ sudo docker start 09ed
09ed
ruiso@ruiso-VirtualBox:~$ sudo docker start 0f9
0f9
ruiso@ruiso-VirtualBox:~$ sudo docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
0f931837beaf php:7.0-apache "docker-php-entrypoi..." 33 hours ago Up 5 seconds 0.0.0.0:1000->80/tcp, :::1000->80/tcp miservidorphp
09ed8013b16a mysql:5.7 "docker-entrypoint.s..." 2 days ago Up 13 seconds 33060/tcp, 0.0.0.0:3307->3306/tcp, :::3307->3306/tcp basedatos
ruiso@ruiso-VirtualBox:~$ 

```



# Register

Nombre\*

BrowserUser

Correo\*

browseruser@mozilla.com

UserName\*

mozilla

Contraseña\*

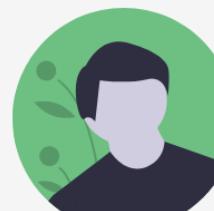
\*\*\*\*\*



**Registrar**

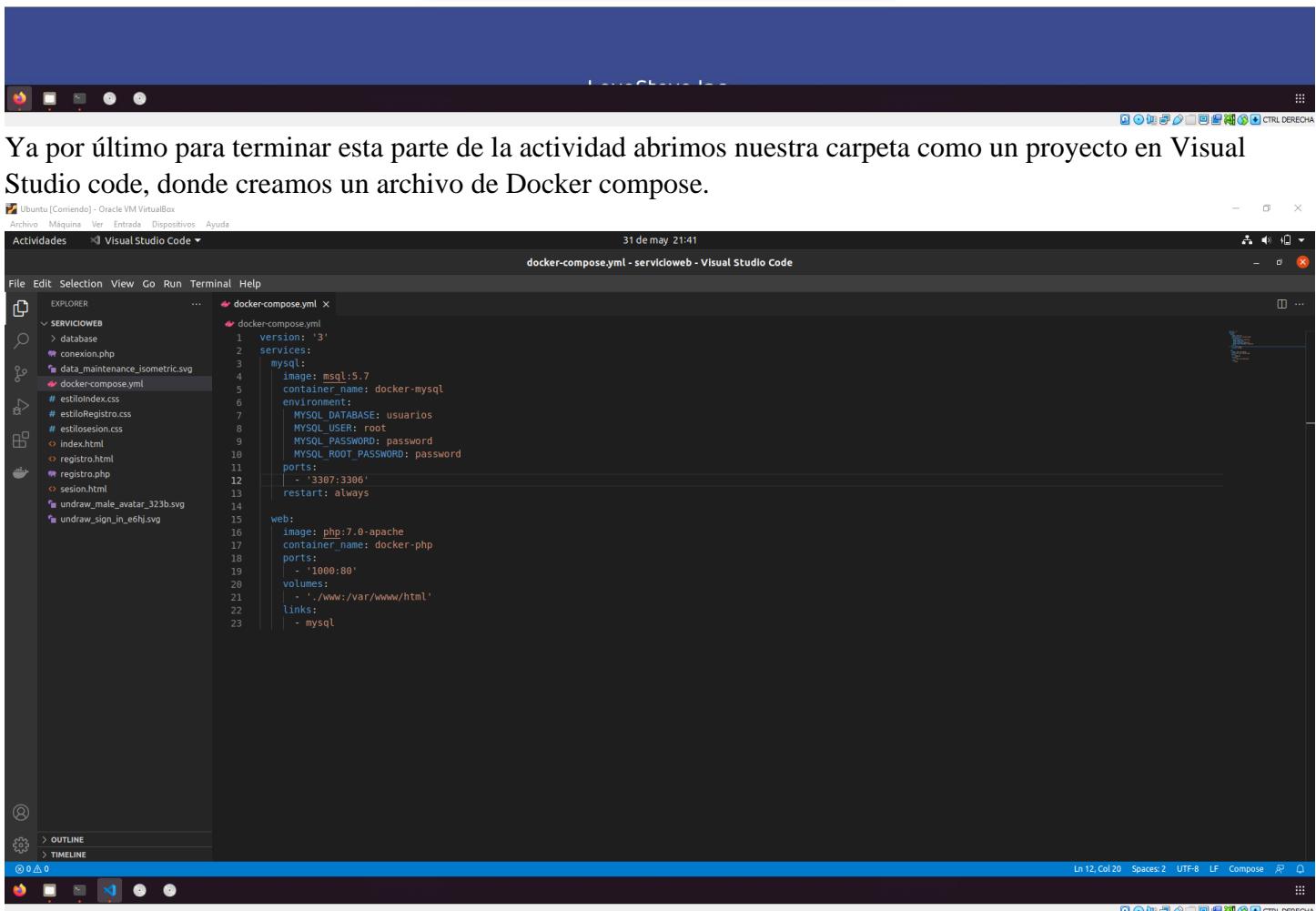
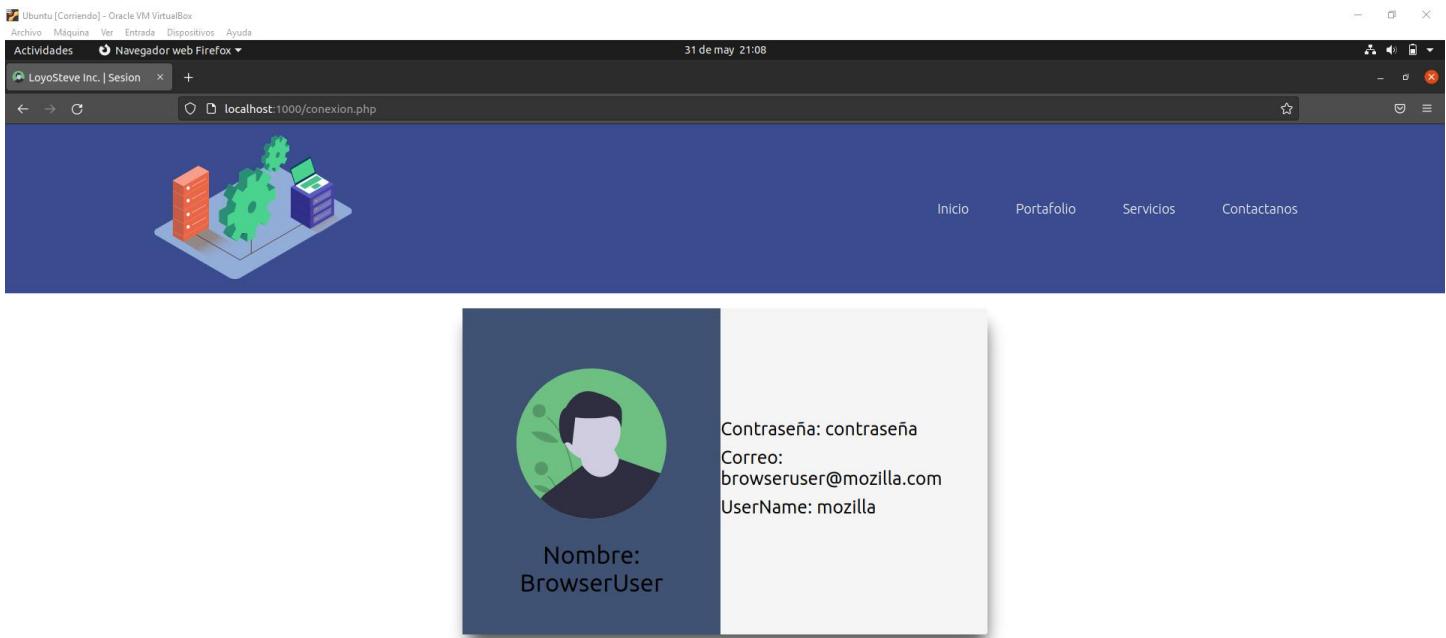


# Iniciar sesión

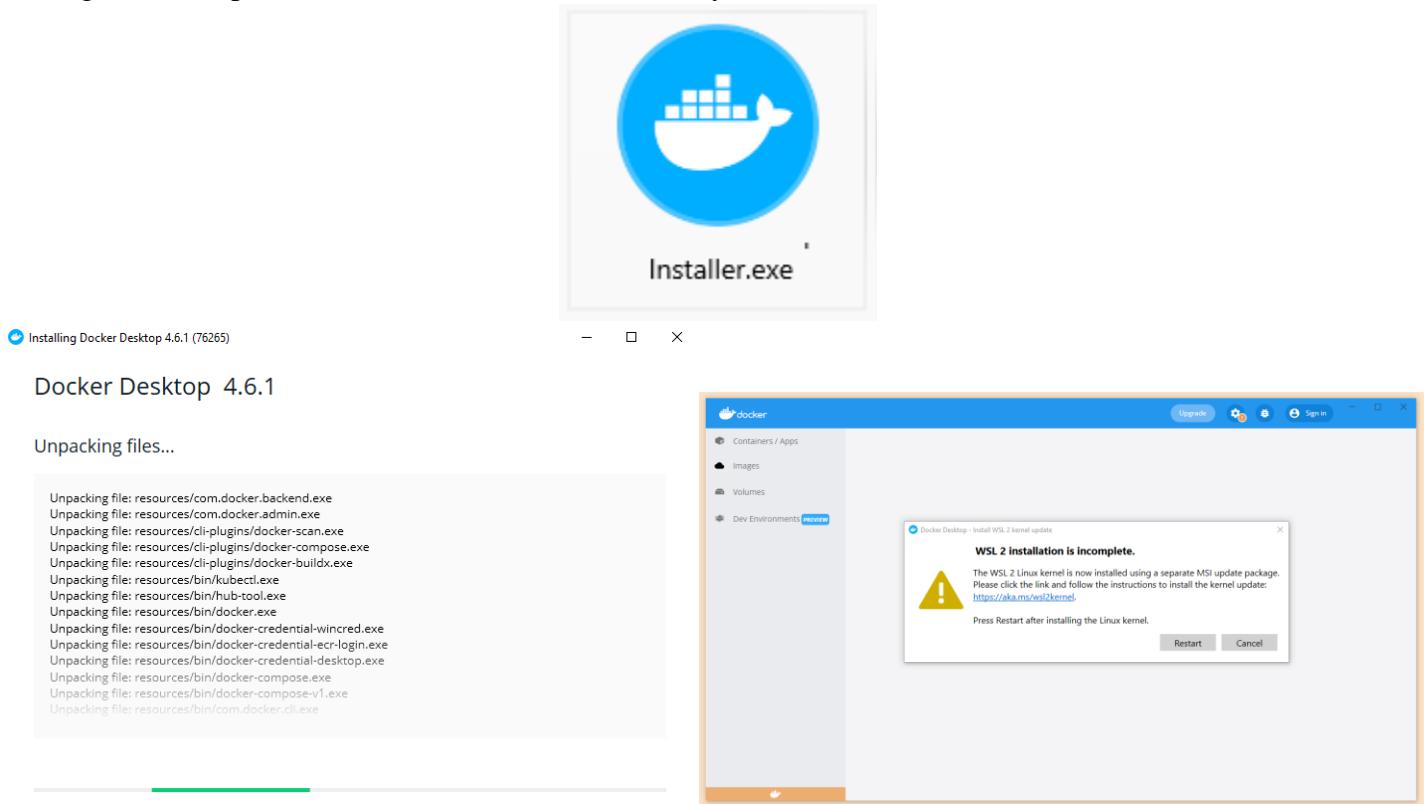


UserName

Contraseña

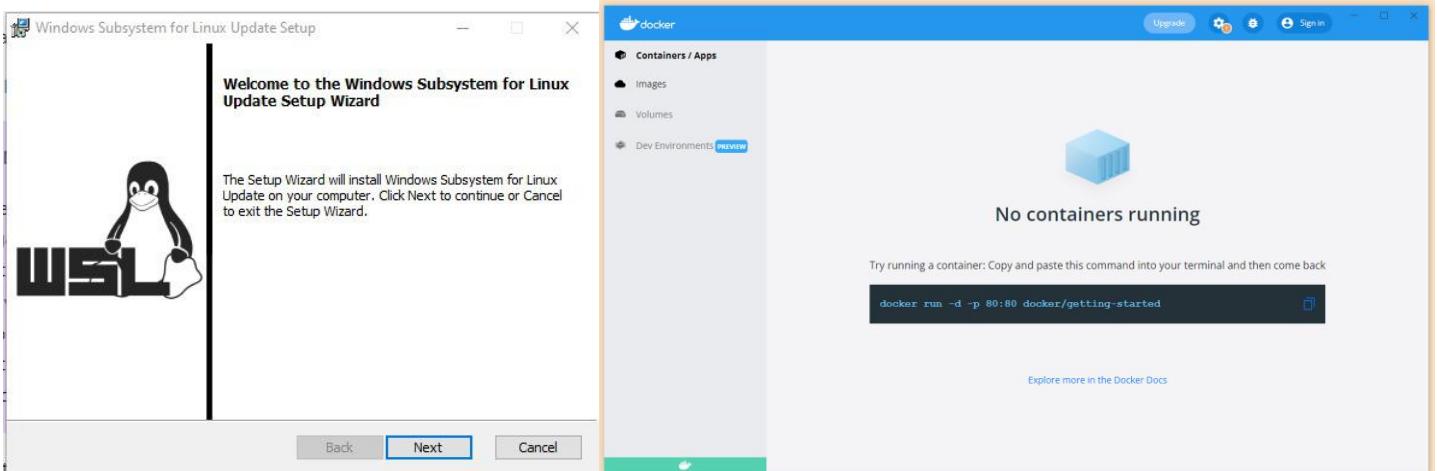


Ahora trabajaremos con Docker, pero en el sistema Operativo Windows. La Actividad que vamos a realizar Necesita que nosotros tengamos instalado Docker y habilitado el Kubernetes. Primero Debemos instalar y descargar Docker para Windows, en caso de no tenerlo ya.



En caso de que salga el anterior error, en vez de dar clic en el botón de RESTART iremos al link de Pasos de instalación manual para versiones anteriores de WSL | Microsoft Docs en el que seleccionaremos en el paso 1 la opción “Paquete de actualización del kernel de Linux en WSL 2 para máquinas x64”, este descargara un archivo de extensión MSI que ejecutaremos. Cuando terminamos de instalar la actualización, podemos picar el botón de RESTART del anterior cuadro de dialogo.

A screenshot of a Microsoft Edge browser window. The address bar shows the URL: https://docs.microsoft.com/es-mx/windows/wsl/install-manual#step-4---download-the-linux-kernel-update-package. The main content area displays 'Paso 4: Descarga del paquete de actualización del kernel de Linux'. It contains two numbered steps: 1. Descargue la versión más reciente: a bullet point 'Paquete de actualización del kernel de Linux en WSL 2 para máquinas x64' followed by a note about ARM64 users. 2. Ejecuta el paquete de actualización que descargaste en el paso anterior. (Haga doble clic para ejecutarlo. Se le pedirán permisos elevados. Seleccione "Sí" para aprobar esta instalación.). Below this, a note says: 'Una vez completada la instalación, vaya al paso siguiente: configuración de WSL 2 como versión predeterminada al instalar nuevas distribuciones de Linux. (Omítala este paso si quiere que las nuevas instalaciones de Linux se establezcan en WSL 1.)'. Another note below it says: 'Para obtener más información, consulta el artículo cambios en la actualización del kernel de Linux en WSL2', disponible en el blog de la línea de comandos de Windows. At the bottom, 'Paso 5: Definición de WSL 2 como versión predeterminada' is shown with a PowerShell command: 'wsl --set-default-version 2'. A small icon of a terminal window with a checkmark is next to the text.



Para instalar Minikube debemos dirigirnos al enlace <https://minikube.sigs.k8s.io/docs/start/> en el encontrará la sección que le permitirá elegir el sistema operativo en que lo quiere instalar y los pasos que debemos seguir para instalar la herramienta. En este caso lo instalaremos en el Sistema Operativo de Microsoft Windows Home Single.

Seleccionamos el sistema operativo en el que queremos instalar.

Operating system

[Linux](#) [macOS](#) [Windows](#)

Architecture

[x86-64](#)

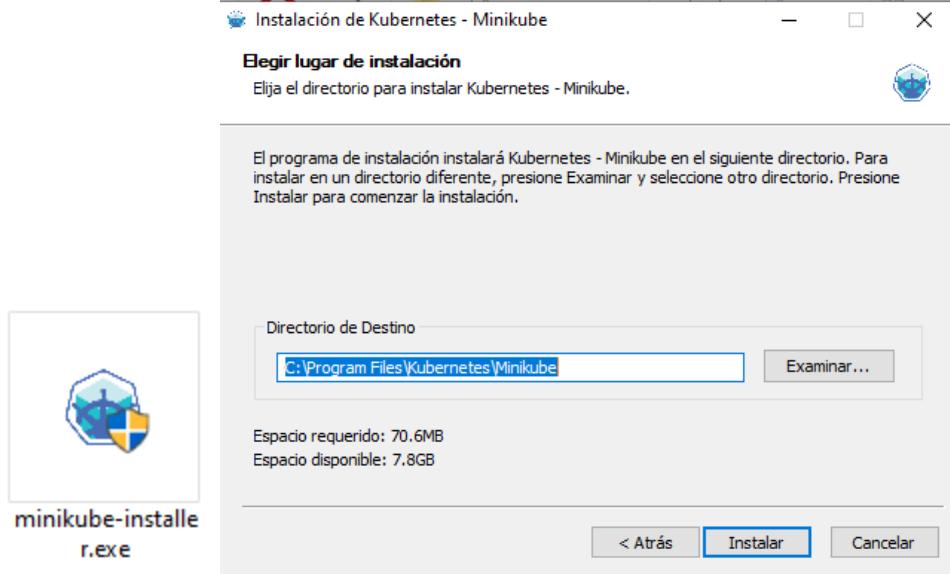
Release type

[Stable](#) [Beta](#)

Installer type

[.exe download](#) [Windows Package Manager](#) [Chocolatey](#)

Descargaremos la última versión por medio del enlace en texto, aunque también se puede utilizar comando de power shell para esta tarea.



Una vez tenemos Minikube en nuestro sistema, ejecutaremos la consola de Power shell de windows pero con permisos de Ani ministrador. En el utilizaremos el siguiente comando en morado. Con este comando agregaremos los binarios en el archivo PATH.

```
$oldPath = [Environment]::GetEnvironmentVariable('Path', [EnvironmentVariableTarget]::Machine)
if ($oldPath.Split(';') -inotcontains 'C:\minikube'){
    [Environment]::SetEnvironmentVariable('Path', ${'{}0';C:\minikube' -f $oldPath),
    [EnvironmentVariableTarget]::Machine)
}
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Windows\system32> $oldPath = [Environment]::GetEnvironmentVariable('Path', [EnvironmentVariableTarget]::Machine)
>> if ($oldPath.Split(';') -inotcontains 'C:\minikube'){
>>     [Environment]::SetEnvironmentVariable('Path', ${'{}0';C:\minikube' -f $oldPath}, [EnvironmentVariableTarget]::Machine)
>> }
```

El único cambio que debemos hacer a este comando es el cambio de la ruta de instalación de Minikube en el sistema, ya que el comando está hecho para una ruta por defecto de instalación y en mi caso específico decidí mover la instalación a la siguiente ruta D:\SOFTWARE\Minikube\

```
$oldPath = [Environment]::GetEnvironmentVariable('Path', [EnvironmentVariableTarget]::Machine)
if ($oldPath.Split(';') -inotcontains 'D:\SOFTWARE\Kubernetes\Minikube'){
    [Environment]::SetEnvironmentVariable('Path', ${'{}0';D:\SOFTWARE\Kubernetes\Minikube' -f
$oldPath), [EnvironmentVariableTarget]::Machine)
}
```

Una vez hecho lo anterior podemos iniciar un cluster, para esto podemos iniciar tanto la consola de CMD de Widows o la Power Shell, pero en cualquiera de los dos casos debemos iniciarla como administradores, en caso de que tengamos un sistema operativo basado en Linux debemos recordar que el modo administrador es el ROOT. La primera vez que se ejecute el cluster, se descargaran las diferentes dependencias para que este funcione, así como todas las utilidades que ofrece minikube como en la siguiente imagen.

```

PS C:\Windows\system32> minikube start
* minikube v1.25.2 en Microsoft Windows 10 Home Single Language 10.0.19044 Build 19044
* Controlador docker seleccionado automáticamente. Otras opciones: hyperv, virtualbox, ssh
* Starting control plane node minikube in cluster minikube
* Pulling base image ...
  > gcr.io/k8s-minikube/kicbase: 379.06 MiB / 379.06 MiB 100.00% 13.28 MiB p
* Creando docker container (CPUs=2, Memory=4000MB) ...E0501 17:39:03.969043 21824 kic.go:267] icacls failed applying permissions - error - [File(<nil>)], output - [archivo procesado: C:\Users\Ruiso Local Pc\.minikube\machines\minikube\id_rsa]
Se procesaron correctamente 1 archivos; error al procesar 0 archivos

* Preparando Kubernetes v1.23.3 en Docker 20.10.12...
- kubelet.housekeeping-interval=5m
- Generando certificados y llaves
- Iniciando plano de control
- Configurando reglas RBAC...
* Verifying Kubernetes components...
- Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Complementos habilitados: storage-provisioner, default-storageclass
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
PS C:\Windows\system32>

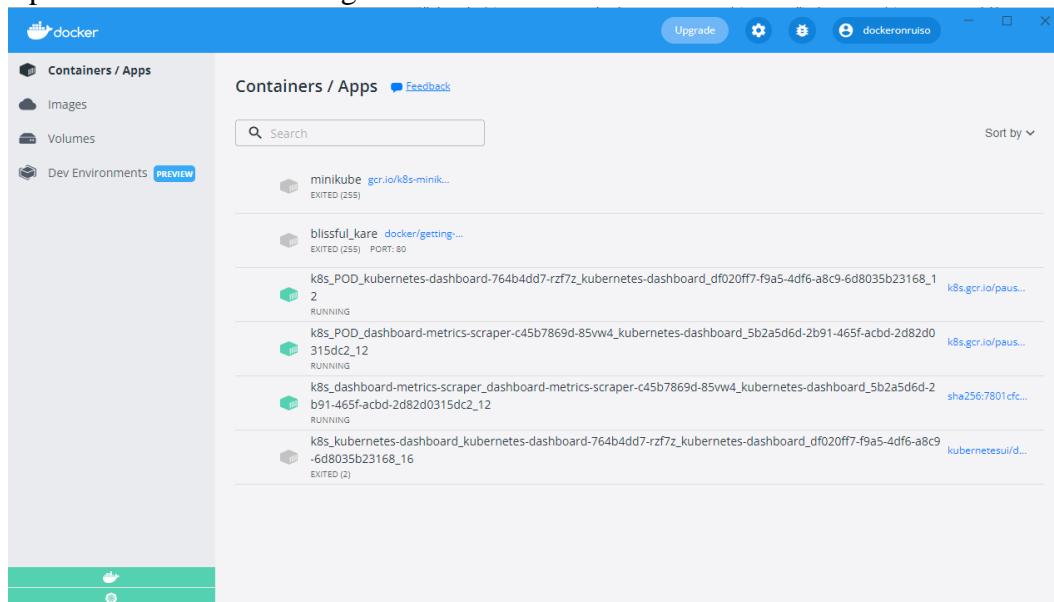
```

En caso de que no se inicie el comando de “minikube Start” pudde revisar que no exista ya un cluster creado a partir de este comando por lo que puede utilizar el comando “minikube delete” el cual eliminara todos los cluster previamente creados en el equipo. En caso de que este no sea el problema puede consultar la página de controladores para obtener ayuda en la configuración del contenedor, la cual sea compatible con su máquina; el enlace es el siguiente Drivers | minikube (k8s.io). Una vez tenemos KUBECTL instalado podemos utilizarlo para abrir el cluster que hemos creado con el comando “kubectl get po -A”.

| PS C:\Windows\system32> kubectl get po -A |                                  |       |         |               |       |
|---|----------------------------------|-------|---------|---------------|-------|
| NAMESPACE                                 | NAME                             | READY | STATUS  | RESTARTS      | AGE   |
| kube-system                               | coredns-64897985d-gkh7h          | 1/1   | Running | 0             | 7m56s |
| kube-system                               | etcd-minikube                    | 1/1   | Running | 0             | 8m7s  |
| kube-system                               | kube-apiserver-minikube          | 1/1   | Running | 0             | 8m7s  |
| kube-system                               | kube-controller-manager-minikube | 1/1   | Running | 0             | 8m7s  |
| kube-system                               | kube-proxy-pz6ds                 | 1/1   | Running | 0             | 7m57s |
| kube-system                               | kube-scheduler-minikube          | 1/1   | Running | 0             | 8m7s  |
| kube-system                               | storage-provisioner              | 1/1   | Running | 1 (7m34s ago) | 8m4s  |

De forma alternativa, minikube puede descargar la versión más adecuada de Kubectl en caso de que este no se encuentre instalado al utilizar el comando “minikube kubectl -- get po -A”. Otra manera seria crear un alias en la configuración de Shell con el comando “alias kubectl="minikube kubectl -"”.

Incialmente, es posible que algunos de los servicios como el aprovisionador y almacenamiento de la información, aun no estén en ejecución. Esta es una condición norma durante la aparición del cluster y se resolverá momentáneamente. Ahora en bien, en caso de que esto no sea la primera vez que corremos Minikube podemos observar un comportamiento similar al siguiente:

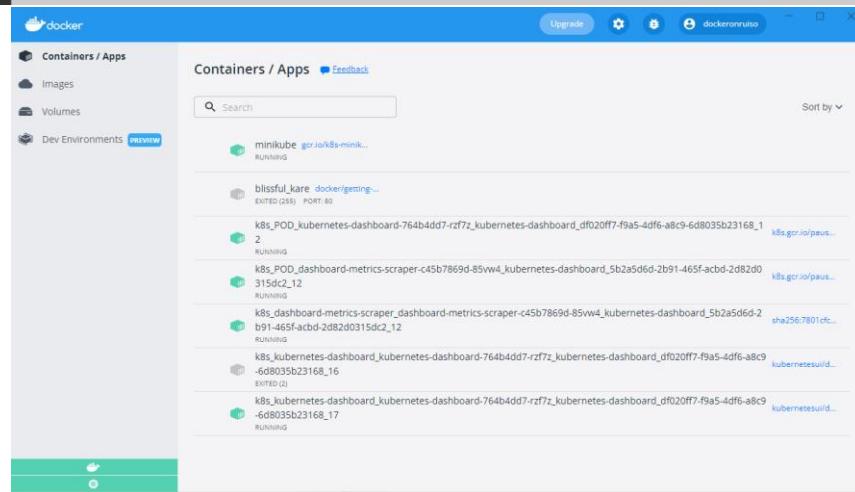


```
C:\Windows\system32>kubectl version
Client Version: version.Info{Major:"1", Minor:"23", GitVersion:"v1.23.6", GitCommit:"ad3338546da947756e8a88aa6822e9c11e7eac22", GitTreeState:"clean", BuildDate:"2022-04-14T08:49:13Z", GoVersion:"go1.17.9", Compiler:"gc", Platform:"windows/amd64"}
Unable to connect to the server: dial tcp 127.0.0.1:49365: connectex: No se puede establecer una conexión ya que el equipo de destino denegó expresamente dicha conexión.
```

```
C:\Windows\system32>_
```

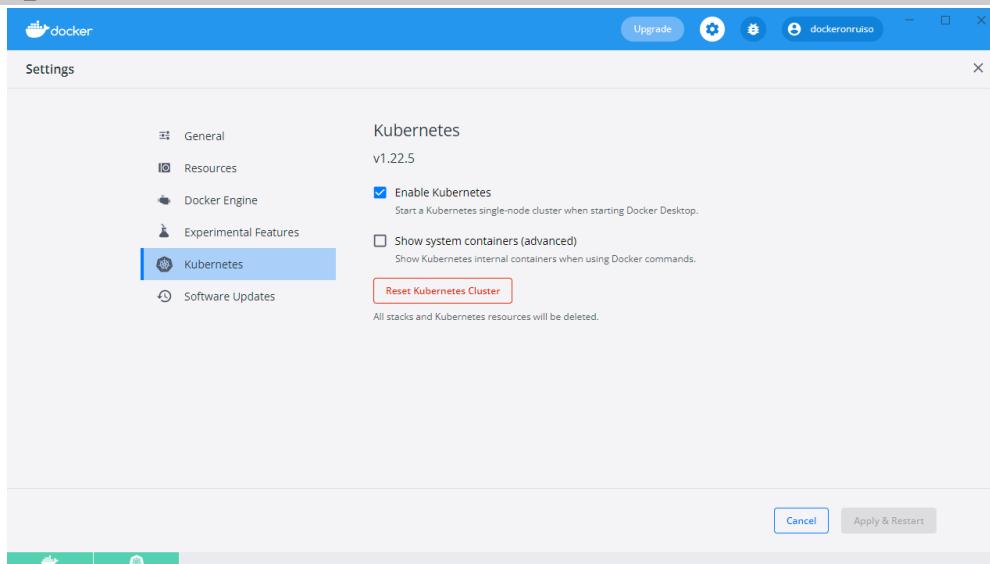
```
C:\Windows\system32>minikube start
* minikube v1.25.2 en Microsoft Windows 10 Home Single Language 10.0.19044 Build 19044
* Using the docker driver based on existing profile
* Starting control plane node minikube in cluster minikube
* Pulling base image ...
* Restarting existing docker container for "minikube" ...
* Preparando Kubernetes v1.23.3 en Docker 20.10.12...
- kubelet.housekeeping-interval=5m
* Verifying Kubernetes components...
- Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Complementos habilitados: storage-provisioner, default-storageclass
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
```

```
C:\Windows\system32>_
```



```
C:\Windows\system32>minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured
```

```
C:\Windows\system32>_
```



Ahora podemos ver los nodos actuales que tenemos en nuestro KUBECTL

```
C:\Windows\system32>kubectl get nodes
NAME      STATUS    ROLES     AGE      VERSION
minikube  Ready     control-plane, master   21d      v1.23.3
```

```
C:\Windows\system32>_
```

```
C:\Windows\system32>minikube dashboard
* Habilitando dashboard
  - Using image kubernetesui/dashboard:v2.3.1
  - Using image kubernetesui/metrics-scraper:v1.0.7
* Verifying dashboard health ...
* Launching proxy ...
* Verifying proxy health ...
* Opening http://127.0.0.1:57914/api/v1/namespaces/kubernetes-dashboard/services/http:kubernetes-dashboard:/proxy/ in your default browser...
```

The screenshot shows the Kubernetes Dashboard interface. On the left, a sidebar lists various resources: Workloads (Cron Jobs, Daemon Sets, Deployments, Jobs, Pods, Replica Sets, Replication Controllers, Stateful Sets), Service (Ingresses, Services), Config and Storage (Config Maps, Persistent Volume Claims, Secrets), and Cluster (Cluster Role Bindings, Cluster Roles, Namespaces, Network Policies). The main area displays the 'Workload Status' with four green circles representing the count of Running workloads: Daemon Sets (1), Deployments (1), Pods (3), and Replica Sets (1). Below this, detailed tables show the configurations for each type of workload.

| Name             | Namespace | Images       | Labels | Pods  | Created     |
|------------------|-----------|--------------|--------|-------|-------------|
| nginx-deployment | default   | nginx:alpine | -      | 1 / 1 | 20.days.ago |

| Name             | Namespace | Images       | Labels | Pods  | Created     |
|------------------|-----------|--------------|--------|-------|-------------|
| nginx-deployment | default   | nginx:alpine | -      | 2 / 2 | 20.days.ago |

| Name       | Namespace | Images | Labels | Node | Status | Restarts | CPU Usage (cores) | Memory Usage (bytes) | Created |
|------------|-----------|--------|--------|------|--------|----------|-------------------|----------------------|---------|
| app: noinx |           |        |        |      |        |          |                   |                      |         |

Crearemos un POD el cual ejecutara un contenedor basado en la imagen que es provista por el mismo Docker.

```
C:\Windows\system32>kubectl create deployment hello-node --image=k8s.gcr.io/echoserver:1.4
deployment.apps/hello-node created
```

```
C:\Windows\system32>_
```

```
C:\Windows\system32>kubectl get deployments
NAME        READY   UP-TO-DATE   AVAILABLE   AGE
hello-node  1/1     1           1           50s
nginx-deployment  2/2     2           2           20d
```

```
C:\Windows\system32>_
```

```
C:\Windows\system32>kubectl get pods
NAME                           READY   STATUS    RESTARTS   AGE
hello-node-6b89d599b9-zlnz1  1/1     Running   0          77s
nginx-deployment-66c9c7669-5k7gp  1/1     Running   1 (18m ago)  20d
nginx-deployment-66c9c7669-rc4x7  1/1     Running   1 (18m ago)  20d
nginx-deployment-hj65p         1/1     Running   1 (18m ago)  20d
```

```
C:\Windows\system32>_
```

```

Administrator: Simbolo del sistema
C:\Windows\system32>kubectl get events
LAST SEEN   TYPE      REASON          OBJECT                                MESSAGE
11s Normal   Scheduled  pod/hello-node-6b89d599b9-zlnz1  MESSAGE
11s Normal   Pulling   pod/hello-node-6b89d599b9-zlnz1  Successfully assigned default/hello-node-6b89d599b9-zlnz1 to minikube
11s Normal   Pulled    pod/hello-node-6b89d599b9-zlnz1  Pulling image "k8s.gcr.io/echoserver:1.4"
10s Normal   Created   pod/hello-node-6b89d599b9-zlnz1  Successfully pulled image "k8s.gcr.io/echoserver:1.4" in 9.334366s
10s Normal   Started   pod/hello-node-6b89d599b9-zlnz1  Created container echoserver
10s Normal   SuccessfulCreate  replicaset/hello-node-6b89d599b9-zlnz1  Started container echoserver
11s Normal   ScalingReplicaSet deployment/hello-node  Created pod: hello-node-6b89d599b9-zlnz1
11s Normal   Starting   node/minikube  Scaled up replica set hello-node-6b89d599b9-zlnz1 to 1
18m Normal   NodeHasSufficientMemory  node/minikube  Starting kubelet.
18m Normal   NodeHasNoDiskPressure   node/minikube  Node minikube status is now: NodeHasSufficientMemory
18m Normal   NodeHasSufficientPID   node/minikube  Node minikube status is now: NodeHasNoDiskPressure
18m Normal   NodeIsSatisfiableEnforced node/minikube  Node minikube status is now: NodeHasSufficientPID
18m Normal   Starting   node/minikube  Updated Node Allocatable limit across pods
18m Normal   RegisteredNode  node/minikube  Node minikube event: Registered Node minikube in Controller
20d Normal   Scheduled  pod/nginx-deployment-66c9c7669-5k7gp  Successfully assigned default/nginx-deployment-66c9c7669-5k7gp to minikube
20d Normal   Pulled    pod/nginx-deployment-66c9c7669-5k7gp  Contained image "nginx:alpine" already present on machine
20d Normal   Created   pod/nginx-deployment-66c9c7669-5k7gp  Created container nginx
20d Normal   Started   pod/nginx-deployment-66c9c7669-5k7gp  Started container nginx
18m Normal   SandboxChanged  pod/nginx-deployment-66c9c7669-5k7gp  Pod sandbox changed, it will be killed and re-created.
18m Normal   Pulled    pod/nginx-deployment-66c9c7669-5k7gp  Container image "nginx:alpine" already present on machine
18m Normal   Created   pod/nginx-deployment-66c9c7669-5k7gp  Created container nginx
18m Normal   Started   pod/nginx-deployment-66c9c7669-5k7gp  Started container nginx
20d Normal   Scheduled  pod/nginx-deployment-66c9c7669-9tc2  Successfully assigned default/nginx-deployment-66c9c7669-9tc2 to minikube
20d Normal   Pulled    pod/nginx-deployment-66c9c7669-9tc2  Contained image "nginx:alpine" already present on machine
20d Normal   Created   pod/nginx-deployment-66c9c7669-9tc2  Created container nginx
20d Normal   Started   pod/nginx-deployment-66c9c7669-9tc2  Started container nginx
18m Normal   SandboxChanged  pod/nginx-deployment-66c9c7669-9tc2  Pod sandbox changed, it will be killed and re-created.
18m Normal   Failed    pod/nginx-deployment-66c9c7669-rc47  Error: services have not yet been read at least once, cannot construct envvars
18m Normal   Created   pod/nginx-deployment-66c9c7669-rc47  Created container nginx
18m Normal   Started   pod/nginx-deployment-66c9c7669-rc47  Started container nginx
20d Normal   SuccessfulCreate  replicaset/nginx-deployment-66c9c7669  Started container nginx
20d Normal   SuccessfulCreate  replicaset/nginx-deployment-66c9c7669  Created pod: nginx-deployment-66c9c7669-rc47
20d Normal   SuccessfulCreate  replicaset/nginx-deployment-66c9c7669  Created pod: nginx-deployment-66c9c7669-9tc2
20d Normal   Scheduled  pod/nginx-deployment-hj65p  Successfully assigned default/nginx-deployment-hj65p to minikube
20d Normal   Pulled    pod/nginx-deployment-hj65p  Container image "nginx:alpine" already present on machine
20d Normal   Created   pod/nginx-deployment-hj65p  Created container nginx
20d Normal   Started   pod/nginx-deployment-hj65p  Started container nginx
18m Normal   SandboxChanged  pod/nginx-deployment-hj65p  Pod sandbox changed, it will be killed and re-created.
18m Normal   Pulled    pod/nginx-deployment-hj65p  Container image "nginx:alpine" already present on machine
18m Normal   Created   pod/nginx-deployment-hj65p  Created container nginx
18m Normal   Started   pod/nginx-deployment-hj65p  Started container nginx
20d Normal   ScalingReplicaSet deployment/nginx-deployment  Scaled up replica set nginx-deployment-66c9c7669 to 2
20d Normal   SuccessfulCreate  daemonset/nginx-deployment  Created pod: nginx-deployment-hj65p
20d Normal   Killed    pod/nginx  Stopping container nginx
20d Normal   Scheduled  pod/nginx  Successfully assigned default/nginx to minikube
20d Normal   Pulled    pod/nginx  Container image "nginx:alpine" already present on machine
20d Normal   Created   pod/nginx  Created container nginx
20d Normal   Started   pod/nginx  Started container nginx
20d Normal   Killing   pod/nginx  Stopping container nginx

C:\Windows\system32>
Administrator: Simbolo del sistema
C:\Windows\system32>kubectl config view
apiVersion: v1
clusters:
- cluster:
  - certificate-authority-data: DATA+OMITTED
    server: https://kubernetes.docker.internal:6443
    name: docker-desktop
- cluster:
  - certificate-authority: C:\Users\Ruiso Local Pc\.minikube\ca.crt
  - extensions:
  - extension:
    last-update: Mon, 30 May 2022 18:56:28 -05
    provider: minikube.sigs.k8s.io
    version: v1.25.2
    name: cluster-info
    server: https://127.0.0.1:60528
    name: minikube
  contexts:
  - context:
    cluster: docker-desktop
    user: docker-desktop
    name: docker-desktop
  - context:
    cluster: minikube
    extensions:
    - extension:
      last-update: Mon, 30 May 2022 18:56:28 -05
      provider: minikube.sigs.k8s.io
      version: v1.25.2
      name: context_info
      namespace: default
      user: minikube
      name: minikube
  current-context: minikube
kind: Config
preferences: {}
users:
- user:
  - client-certificate-data: REDACTED
    client-key-data: REDACTED
  - name: minikube
  user:
    client-certificate: C:\Users\Ruiso Local Pc\.minikube\profiles\minikube\client.crt
    client-key: C:\Users\Ruiso Local Pc\.minikube\profiles\minikube\client.key

C:\Windows\system32>

```

Ahora crearemos un servicio

```
C:\Windows\system32>kubectl expose deployment hello-node --type=LoadBalancer --port=8080
service/hello-node exposed
```

```
C:\Windows\system32>
```

```
C:\Windows\system32>kubectl get services
NAME      TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)      AGE
hello-node LoadBalancer  10.96.205.144 <pending>    8080:31496/TCP 28s
kubernetes ClusterIP   10.96.0.1     <none>       443/TCP     21d
```

```
C:\Windows\system32>_
```

```
C:\Windows\system32>minikube service hello-node
* Starting tunnel for service hello-node.
* Opening service default/hello-node in default browser...
! Porque estás usando controlador Docker en windows, la terminal debe abrirse para ejecutarlo.
```

[CTRL]+[C]. Primero habilitemos las extensiones soportadas actualmente en Windows 10 Home.

```
C:\Windows\system32>minikube addons list
```

| ADDON NAME                  | PROFILE  | STATUS    | MAINTAINER                       |
|-----------------------------|----------|-----------|----------------------------------|
| ambassador                  | minikube | disabled  | third-party (ambassador)         |
| auto-pause                  | minikube | disabled  | google                           |
| csi-hostpath-driver         | minikube | disabled  | kubernetes                       |
| dashboard                   | minikube | enabled ↗ | kubernetes                       |
| default-storageclass        | minikube | enabled ↗ | kubernetes                       |
| efk                         | minikube | disabled  | third-party (elastic)            |
| freshpod                    | minikube | disabled  | google                           |
| gcp-auth                    | minikube | disabled  | google                           |
| gvisor                      | minikube | disabled  | google                           |
| helm-tiller                 | minikube | disabled  | third-party (helm)               |
| ingress                     | minikube | disabled  | unknown (third-party)            |
| ingress-dns                 | minikube | disabled  | google                           |
| istio                       | minikube | disabled  | third-party (istio)              |
| istio-provisioner           | minikube | disabled  | third-party (istio)              |
| kong                        | minikube | disabled  | third-party (Kong HQ)            |
| kubevirt                    | minikube | disabled  | third-party (kubevirt)           |
| logviewer                   | minikube | disabled  | unknown (third-party)            |
| metallb                     | minikube | disabled  | third-party (metallb)            |
| metrics-server              | minikube | disabled  | kubernetes                       |
| nvidia-driver-installer     | minikube | disabled  | google                           |
| nvidia-gpu-device-plugin    | minikube | disabled  | third-party (nvidia)             |
| olm                         | minikube | disabled  | third-party (operator framework) |
| pod-security-policy         | minikube | disabled  | unknown (third-party)            |
| portainer                   | minikube | disabled  | portainer.io                     |
| registry                    | minikube | disabled  | google                           |
| registry-aliases            | minikube | disabled  | unknown (third-party)            |
| registry-creds              | minikube | disabled  | third-party (upmc enterprises)   |
| storage-provisioner         | minikube | enabled ↗ | google                           |
| storage-provisioner-gluster | minikube | disabled  | unknown (third-party)            |
| volumesnapshots             | minikube | disabled  | kubernetes                       |

```
C:\Windows\system32>
```

```
C:\Windows\system32>minikube addons enable metrics-server
- Using image k8s.gcr.io/metrics-server/metrics-server:v0.4.2
* The 'metrics-server' addon is enabled
```

```
C:\Windows\system32>
```

```
C:\Windows\system32>kubectl get pod,svc -n kube-system
NAME                               READY   STATUS    RESTARTS   AGE
pod/coredns-64897985d-hgk6r      1/1     Running   2 (33m ago)  21d
pod/etcfd-minikube               1/1     Running   2 (33m ago)  21d
pod/kube-apiserver-minikube     1/1     Running   2 (33m ago)  21d
pod/kube-controller-manager-minikube 1/1     Running   2 (33m ago)  21d
pod/kube-proxy-4kgsh             1/1     Running   2 (33m ago)  21d
pod/kube-scheduler-minikube     1/1     Running   2 (33m ago)  21d
pod/metrics-server-6b76bd68b6-c52mt 1/1     Running   0          115s
pod/storage-provisioner         1/1     Running   6 (32m ago)  21d

NAME           TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
service/kube-dns  ClusterIP  10.96.0.10    <none>        53/UDP,53/TCP,9153/TCP  21d
service/metrics-server ClusterIP  10.101.21.70  <none>        443/TCP    114s

C:\Windows\system32>

C:\Windows\system32>minikube addons disable metrics-server
* El complemento "metrics-server" está desactivado

C:\Windows\system32>
```

Ahora podemos limpiar los recursos creados en el cluster.

```
C:\Windows\system32>kubectl delete service hello-node
service "hello-node" deleted

C:\Windows\system32>kubectl delete deployment hello-node
deployment.apps "hello-node" deleted

C:\Windows\system32>
```

Creemos otro tipo de contenedor.

```
C:\Windows\system32>kubectl get nodes
NAME      STATUS  ROLES      AGE  VERSION
minikube  Ready   control-plane,master  21d  v1.23.3

C:\Windows\system32>kubectl apply -f https://k8s.io/examples/application/guestbook/redis-leader-deployment.yaml
deployment.apps/redis-leader created

C:\Windows\system32>kubectl get pods
NAME                               READY   STATUS    RESTARTS   AGE
nginx-deployment-66c9c7669-5k7gp  1/1     Running   1 (43m ago)  20d
nginx-deployment-66c9c7669-rc4x7  1/1     Running   1 (43m ago)  21d
nginx-deployment-hj65p            1/1     Running   1 (43m ago)  20d
redis-leader-766465cd9c-54gdz   0/1     ContainerCreating   0          12s

C:\Windows\system32>kubectl apply -f https://k8s.io/examples/application/guestbook/redis-follower-service.yaml
service/redis-follower created

C:\Windows\system32>kubectl get service
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
kubernetes  ClusterIP  10.96.0.1    <none>        443/TCP      21d
redis-follower ClusterIP  10.97.159.106  <none>        6379/TCP    5s
redis-leader   ClusterIP  10.106.235.90  <none>        6379/TCP    84s

C:\Windows\system32>
```

```
clusterip      clusterip   10.106.235.90  <none>        80/TCP       84s
C:\Windows\system32>kubectl apply -f https://k8s.io/examples/application/guestbook/frontend-deployment.yaml
deployment.apps/frontend created

C:\Windows\system32>kubectl get service
NAME           TYPE      CLUSTER-IP    EXTERNAL-IP   PORT(S)      AGE
kubernetes     ClusterIP  10.96.0.1    <none>        443/TCP     21d
redis-follower ClusterIP  10.97.159.106 <none>        6379/TCP    56s
redis-leader   ClusterIP  10.106.235.90 <none>        6379/TCP    2m15s

C:\Windows\system32>kubectl get pods -l app=guestbook -l tier=frontend
NAME          READY   STATUS             RESTARTS   AGE
frontend-57df59b89c-2lh42  0/1    ContainerCreating   0          20s
frontend-57df59b89c-vdbgml  0/1    ContainerCreating   0          20s
frontend-57df59b89c-w2dp5  0/1    ContainerCreating   0          20s

C:\Windows\system32>
C:\Windows\system32>kubectl apply -f https://k8s.io/examples/application/guestbook/frontend-service.yaml
service/frontend created

C:\Windows\system32>kubectl get service
NAME           TYPE      CLUSTER-IP    EXTERNAL-IP   PORT(S)      AGE
frontend        ClusterIP  10.105.248.145 <none>        80/TCP       6s
kubernetes     ClusterIP  10.96.0.1    <none>        443/TCP     21d
redis-follower ClusterIP  10.97.159.106 <none>        6379/TCP    119s
redis-leader   ClusterIP  10.106.235.90 <none>        6379/TCP    3m18s
```

```
C:\Windows\system32>kubectl port-forward svc/frontend 8080:80
Forwarding from 127.0.0.1:8080 -> 80
Forwarding from [::1]:8080 -> 80

Guestbook
Messages
Submit
Guestbook
Messages
Submit
hallo

C:\Windows\system32>kubectl port-forward svc/frontend 8080:80
Forwarding from 127.0.0.1:8080 -> 80
Forwarding from [::1]:8080 -> 80
Handling connection for 8080
Handling connection for 8080
Handling connection for 8080
Handling connection for 8080
```

```
C:\Windows\system32>kubectl apply -f https://k8s.io/examples/application/guestbook/frontend-service.yaml
service/frontend created
```

```
C:\Windows\system32>kubectl get service
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
frontend   ClusterIP  10.105.248.145  <none>        80/TCP       6s
kubernetes ClusterIP  10.96.0.1     <none>        443/TCP      21d
redis-follower ClusterIP  10.97.159.106  <none>        6379/TCP     119s
redis-leader   ClusterIP  10.106.235.90   <none>        6379/TCP     3m18s
```

```
C:\Windows\system32>kubectl port-forward svc/frontend 8080:80
Forwarding from 127.0.0.1:8080 -> 80
Forwarding from [::1]:8080 -> 80
Handling connection for 8080
Handling connection for 8080
Handling connection for 8080
Handling connection for 8080
```

```
C:\Windows\system32>kubectl scale deployment frontend --replicas=3
deployment.apps/frontend scaled
```

```
C:\Windows\system32>kubectl get pods
NAME                  READY   STATUS    RESTARTS   AGE
frontend-57df59b89c-2lh42  1/1    Running   0          36m
frontend-57df59b89c-vdbgsm 1/1    Running   0          36m
frontend-57df59b89c-w2dp5  1/1    Running   0          36m
nginx-deployment-66c9c7669-5k7gp 1/1    Running   1 (91m ago)  21d
nginx-deployment-66c9c7669-rc4x7  1/1    Running   1 (91m ago)  21d
nginx-deployment-hj65p       1/1    Running   1 (91m ago)  21d
redis-follower-84fcc94dfc-8rqg8 1/1    Running   0          37m
redis-follower-84fcc94dfc-qh7jf 1/1    Running   0          37m
redis-leader-766465cd9c-54gdz  1/1    Running   0          47m
```

```
C:\Windows\system32>_
```

```
C:\Windows\system32>kubectl scale deployment frontend --replicas=2
deployment.apps/frontend scaled
```

```
C:\Windows\system32>kubectl get pods
NAME                  READY   STATUS    RESTARTS   AGE
frontend-57df59b89c-vdbgsm 1/1    Running   0          39m
frontend-57df59b89c-w2dp5  1/1    Running   0          39m
nginx-deployment-66c9c7669-5k7gp 1/1    Running   1 (94m ago)  21d
nginx-deployment-66c9c7669-rc4x7  1/1    Running   1 (94m ago)  21d
nginx-deployment-hj65p       1/1    Running   1 (94m ago)  21d
redis-follower-84fcc94dfc-8rqg8 1/1    Running   0          41m
redis-follower-84fcc94dfc-qh7jf 1/1    Running   0          41m
redis-leader-766465cd9c-54gdz  1/1    Running   0          51m
```

```
C:\Windows\system32>_
```

```
C:\Windows\system32>kubectl delete deployment -l app=redis
deployment.apps "redis-follower" deleted
deployment.apps "redis-leader" deleted
```

```
C:\Windows\system32>kubectl delete service -l app=redis
service "redis-follower" deleted
service "redis-leader" deleted
```

```
C:\Windows\system32>kubectl delete deployment frontend
deployment.apps "frontend" deleted
```

```
C:\Windows\system32>kubectl delete service frontend
service "frontend" deleted
```

```
C:\Windows\system32>kubectl get pods
NAME                  READY   STATUS    RESTARTS   AGE
nginx-deployment-66c9c7669-5k7gp   1/1     Running   1 (97m ago)   21d
nginx-deployment-66c9c7669-rc4x7   1/1     Running   1 (97m ago)   21d
nginx-deployment-hj65p            1/1     Running   1 (97m ago)   21d
```

```
C:\Windows\system32>_
```