

Reporte Instalación Docker y PostgreSQL

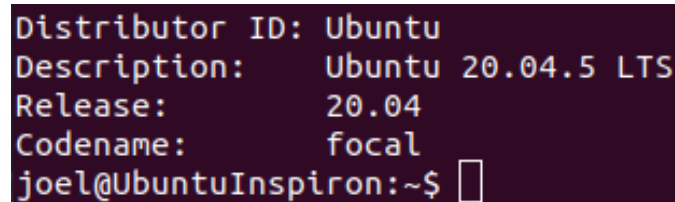
Integrante: Joel Haidd Reyes Cedillo

A continuación se darán las instrucciones que se siguieron paso a paso para la instalación del sistema operativo para contenedores *Docker*, el sistema manejador de bases de datos (SMBD) *PostgreSQL* y la herramienta de administración de bases de datos *DBeaver*, así como la forma de conectarse con esta última.

Especificaciones del equipo

A continuación se dan los detalles de las especificaciones del equipo que se usó para la instalación de todos los programas.

- Sistema Operativo: Linux
- Distribución: Ubuntu
- Versión de la instalación: 20.04 (Focal)
- Tiempo total de instalación: Aproximadamente 30 minutos

A terminal window with a dark background and light-colored text. It displays the output of the 'lsb_release -a' command, showing the following information: Distributor ID: Ubuntu, Description: Ubuntu 20.04.5 LTS, Release: 20.04, Codename: focal. The prompt at the bottom is 'joel@UbuntuInspiron:~\$' followed by a cursor.

```
Distributor ID: Ubuntu
Description:   Ubuntu 20.04.5 LTS
Release:      20.04
Codename:     focal
joel@UbuntuInspiron:~$
```

Figura 1: Version Ubuntu

Pasos de la instalación de Docker

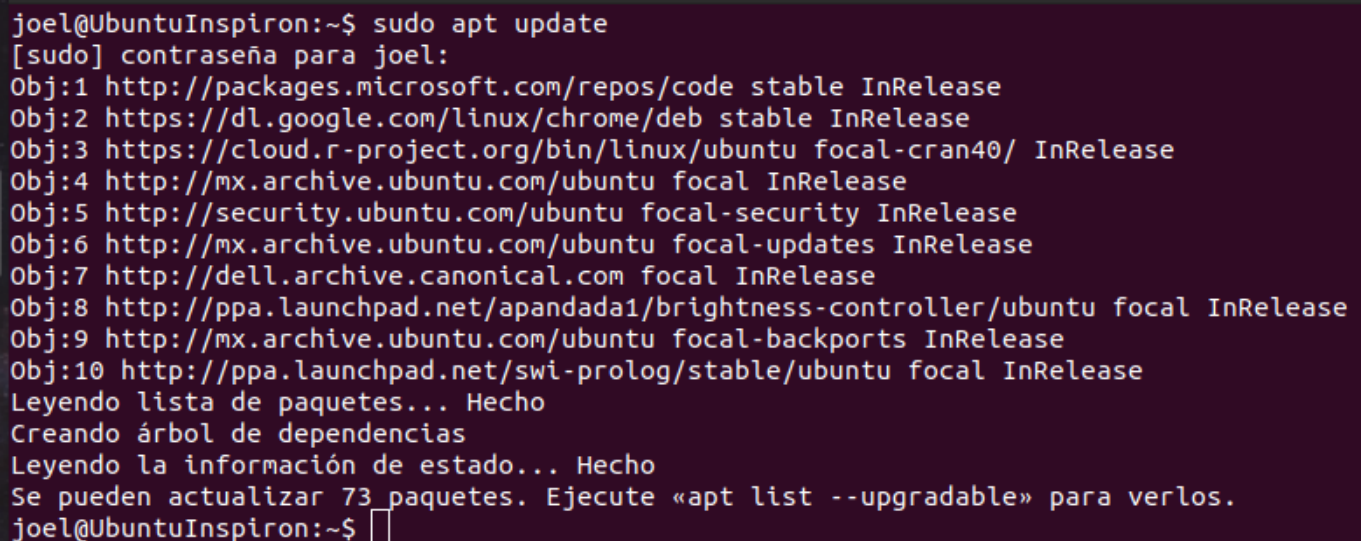
A continuación se dan las instrucciones que se siguieron para la instalación de Docker.

Nota: Para ejecutar los comandos es necesario que se ejecuten con la directiva **sudo** (y en caso de requerirlo dar la contraseña del equipo) para poder tener los permisos necesarios para instalar y descargar los programas.

1. Para comenzar se abre la terminal del sistema y se ejecuta el comando

```
1 $ sudo apt update
2
```

Esto permite actualizar la lista de paquetes existentes en el sistema.



```
joel@UbuntuInspiron:~$ sudo apt update
[sudo] contraseña para joel:
Obj:1 http://packages.microsoft.com/repos/code stable InRelease
Obj:2 https://dl.google.com/linux/chrome/deb stable InRelease
Obj:3 https://cloud.r-project.org/bin/linux/ubuntu focal-cran40/ InRelease
Obj:4 http://mx.archive.ubuntu.com/ubuntu focal InRelease
Obj:5 http://security.ubuntu.com/ubuntu focal-security InRelease
Obj:6 http://mx.archive.ubuntu.com/ubuntu focal-updates InRelease
Obj:7 http://dell.archive.canonical.com focal InRelease
Obj:8 http://ppa.launchpad.net/apandada1/brightness-controller/ubuntu focal InRelease
Obj:9 http://mx.archive.ubuntu.com/ubuntu focal-backports InRelease
Obj:10 http://ppa.launchpad.net/swi-prolog/stable/ubuntu focal InRelease
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se pueden actualizar 73 paquetes. Ejecute «apt list --upgradable» para verlos.
joel@UbuntuInspiron:~$
```

Figura 2: Ejecución del comando del paso 1

2. Una vez actualizados los paquetes se procede a ejecutar el comando

```
1 $ sudo apt install apt-transport-https ca-certificates curl software-  
properties-common  
2
```

Y cuando aparezca ¿Desea continuar? [S/n] \$ escribir S en la terminal y presionar Retorno para que comience la instalación.

Este comando instala algunos paquetes de requisitos previos que permitan a apt a usar paquetes a través de HTTPS

```
joel@UbuntuInspiron:~$ sudo apt install apt-transport-https ca-certificates curl software-properties-common  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias  
Leyendo la información de estado... Hecho  
ca-certificates ya está en su versión más reciente (20211016ubuntu0.20.04.1).  
fijado ca-certificates como instalado manualmente.  
curl ya está en su versión más reciente (7.68.0-1ubuntu2.15).  
apt-transport-https ya está en su versión más reciente (2.0.9).  
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.  
chromium-codecs-ffmpeg-extra gir1.2-goa-1.0 gstreamer1.0-vaapi libfwupdplugin1 libgstreamer-plugins-bad1.0-0 libva-wayland2  
Utilice «sudo apt autoremove» para eliminarlos.  
Se instalarán los siguientes paquetes adicionales:  
python3-software-properties software-properties-gtk ubuntu-advantage-desktop-daemon ubuntu-advantage-tools  
Se instalarán los siguientes paquetes NUEVOS:  
ubuntu-advantage-desktop-daemon  
Se actualizarán los siguientes paquetes:  
python3-software-properties software-properties-common software-properties-gtk ubuntu-advantage-tools  
4 actualizados, 1 nuevos se instalarán, 0 para eliminar y 69 no actualizados.  
Se necesita descargar 297 kB de archivos.  
Se utilizarán 329 kB de espacio de disco adicional después de esta operación.  
¿Desea continuar? [S/n] S  
Des:1 http://mx.archive.ubuntu.com/ubuntu focal-updates/main amd64 ubuntu-advantage-tools amd64 27.13.5~20.04.1 [173 kB]  
Des:2 http://mx.archive.ubuntu.com/ubuntu focal-updates/main amd64 software-properties-common all 0.99.9.10 [10.4 kB]  
Des:3 http://mx.archive.ubuntu.com/ubuntu focal-updates/main amd64 ubuntu-advantage-desktop-daemon amd64 1.10~20.04.1 [23.5 kB]  
Des:4 http://mx.archive.ubuntu.com/ubuntu focal-updates/main amd64 software-properties-gtk all 0.99.9.10 [68.6 kB]  
Des:5 http://mx.archive.ubuntu.com/ubuntu focal-updates/main amd64 python3-software-properties all 0.99.9.10 [21.7 kB]  
Descargados 297 kB en 1s (322 kB/s)  
Preconfigurando paquetes ...  
(Leyendo la base de datos ... 284327 ficheros o directorios instalados actualmente.)  
Preparando para desempaquetar .../ubuntu-advantage-tools_27.13.5~20.04.1_amd64.deb ...  
Desempaquetando ubuntu-advantage-tools (27.13.5~20.04.1) sobre (27.10.1~20.04.1) ...  
Preparando para desempaquetar .../software-properties-common_0.99.9.10_all.deb ...  
Desempaquetando software-properties-common (0.99.9.10) sobre (0.99.9.8) ...  
Seleccionando el paquete ubuntu-advantage-desktop-daemon previamente no seleccionado.  
Preparando para desempaquetar .../ubuntu-advantage-desktop-daemon_1.10~20.04.1_amd64.deb ...  
Desempaquetando ubuntu-advantage-desktop-daemon (1.10~20.04.1) ...  
Preparando para desempaquetar .../software-properties-gtk_0.99.9.10_all.deb ...  
Desempaquetando software-properties-gtk (0.99.9.10) sobre (0.99.9.8) ...  
Preparando para desempaquetar .../python3-software-properties_0.99.9.10_all.deb ...
```

Figura 3: Ejecución del comando del paso 2

3. Ya que se han instalado los paquetes lo siguiente es añadir la clave de GPG para el repositorio de Docker con el comando:

```
1 $ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key  
add -  
2
```

Al terminar si aparece OK en la terminal significa que la clave se añadió exitosamente.

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

Figura 4: Ejecución del comando del paso 3

4. Ahora se procede a agregar el repositorio a las fuentes APT con el comando

```
1 $ sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/  
linux/ubuntu focal stable"  
2
```

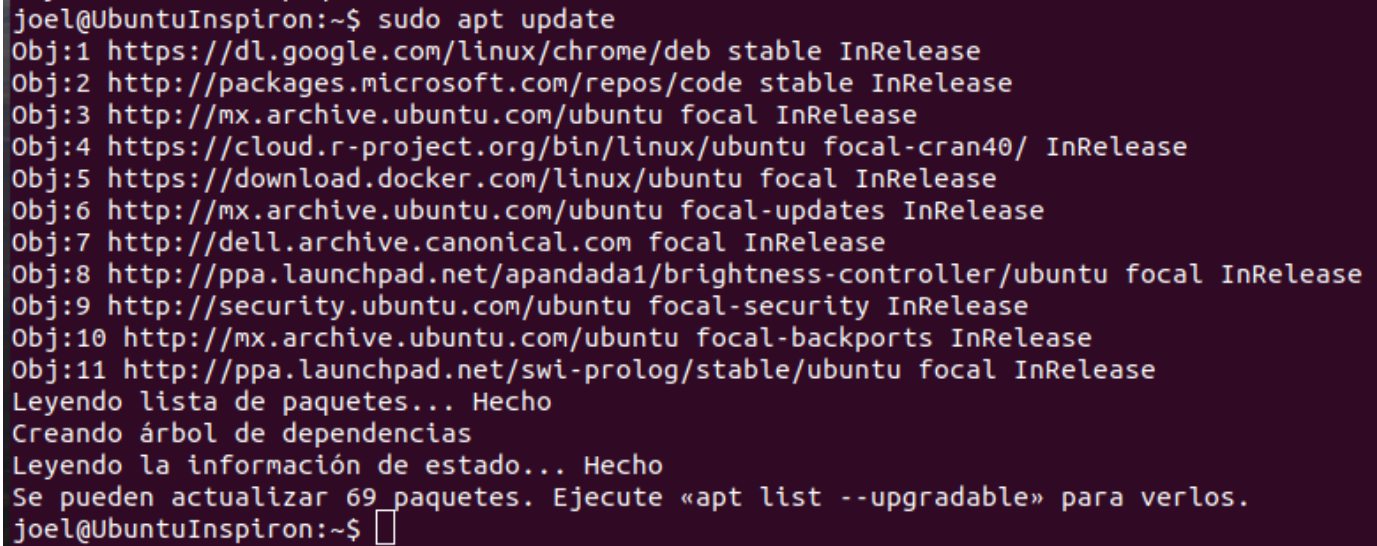
```
joel@UbuntuInspiron:~$ sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable"  
Obj:1 https://dl.google.com/linux/chrome/deb stable InRelease  
Obj:2 http://packages.microsoft.com/repos/code stable InRelease  
Des:3 https://download.docker.com/linux/ubuntu focal InRelease [57.7 kB]  
Obj:4 http://mx.archive.ubuntu.com/ubuntu focal InRelease  
Obj:5 http://security.ubuntu.com/ubuntu focal-security InRelease  
Obj:6 https://cloud.r-project.org/bin/linux/ubuntu focal-cran40/ InRelease  
Obj:7 http://mx.archive.ubuntu.com/ubuntu focal-updates InRelease  
Obj:8 http://ppa.launchpad.net/apandada1/brightness-controller/ubuntu focal InRelease  
Obj:9 http://dell.archive.canonical.com focal InRelease  
Obj:10 http://mx.archive.ubuntu.com/ubuntu focal-backports InRelease  
Des:11 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages [24.7 kB]  
Des:12 https://download.docker.com/linux/ubuntu focal/stable amd64 Contents (deb) [1 378 B]  
Obj:13 http://ppa.launchpad.net/swi-prolog/stable/ubuntu focal InRelease  
Descargados 83.8 kB en 2s (41.7 kB/s)  
Leyendo lista de paquetes... Hecho  
joel@UbuntuInspiron:~$
```

Figura 5: Ejecución del comando del paso 4

5. A continuación se vuelve a ejecutar el comando

```
1 $ sudo apt update
2
```

Para actualizar el paquete de base de datos con los paquetes de Docker

A terminal window with a dark purple background and light green text. The prompt is 'joel@UbuntuInspiron:~\$'. The command 'sudo apt update' has been executed. The output shows a list of 11 objects (repositories) being updated, including Google Chrome, Microsoft Code, Ubuntu focal, R-project, Docker, Canonical focal, Launchpad brightness-controller, Ubuntu focal-security, Ubuntu focal-backports, and Launchpad swi-prolog. It then says 'Leyendo lista de paquetes... Hecho', 'Creando árbol de dependencias', and 'Leyendo la información de estado... Hecho'. Finally, it states 'Se pueden actualizar 69 paquetes. Ejecute «apt list --upgradable» para verlos.' and returns to the prompt 'joel@UbuntuInspiron:~\$' with a cursor.

```
joel@UbuntuInspiron:~$ sudo apt update
Obj:1 https://dl.google.com/linux/chrome/deb stable InRelease
Obj:2 http://packages.microsoft.com/repos/code stable InRelease
Obj:3 http://mx.archive.ubuntu.com/ubuntu focal InRelease
Obj:4 https://cloud.r-project.org/bin/linux/ubuntu focal-cran40/ InRelease
Obj:5 https://download.docker.com/linux/ubuntu focal InRelease
Obj:6 http://mx.archive.ubuntu.com/ubuntu focal-updates InRelease
Obj:7 http://dell.archive.canonical.com focal InRelease
Obj:8 http://ppa.launchpad.net/apandada1/brightness-controller/ubuntu focal InRelease
Obj:9 http://security.ubuntu.com/ubuntu focal-security InRelease
Obj:10 http://mx.archive.ubuntu.com/ubuntu focal-backports InRelease
Obj:11 http://ppa.launchpad.net/swi-prolog/stable/ubuntu focal InRelease
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se pueden actualizar 69 paquetes. Ejecute «apt list --upgradable» para verlos.
joel@UbuntuInspiron:~$
```

Figura 6: Ejecución del comando del paso 5

6. Y ahora si se instala Docker con el comando

```
1 $ sudo apt install docker-ce
2
```

Y cuando aparezca ¿Desea continuar? [S/n] \$ escribir S y presionar Retorno para comenzar con la instalación.

Nota: Dependiendo la velocidad de internet que se tenga y la versión que se vaya a instalar la descarga puede tardar varios minutos.

```
joel@UbuntuInspiron:~$ sudo apt install docker-ce
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
 chromium-codecs-ffmpeg-extra gir1.2-goa-1.0 gstreamer1.0-vaapi libfwupdplugin1 libgstreamer-plugins-bad1.0-0 libva-wayland2
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes adicionales:
 containerd.io docker-buildx-plugin docker-ce-cli docker-ce-rootless-extras docker-compose-plugin docker-scan-plugin pigz slirp4netns
Paquetes sugeridos:
 aufs-tools cgroupfs-mount | cgroup-lite
Se instalarán los siguientes paquetes NUEVOS:
 containerd.io docker-buildx-plugin docker-ce docker-ce-cli docker-ce-rootless-extras docker-compose-plugin docker-scan-plugin pigz
 slirp4netns
0 actualizados, 9 nuevos se instalarán, 0 para eliminar y 69 no actualizados.
Se necesita descargar 112 MB de archivos.
Se utilizarán 401 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] S
Des:1 http://mx.archive.ubuntu.com/ubuntu focal/universe amd64 pigz amd64 2.4-1 [57.4 kB]
Des:2 https://download.docker.com/linux/ubuntu focal/stable amd64 containerd.io amd64 1.6.18-1 [28.2 MB]
Des:3 http://mx.archive.ubuntu.com/ubuntu focal/universe amd64 slirp4netns amd64 0.4.3-1 [74.3 kB]
Des:4 https://download.docker.com/linux/ubuntu focal/stable amd64 docker-buildx-plugin amd64 0.10.2-1-ubuntu.20.04-focal [25.9 MB]
Des:5 https://download.docker.com/linux/ubuntu focal/stable amd64 docker-ce-cli amd64 5:23.0.1-1-ubuntu.20.04-focal [13.2 MB]
Des:6 https://download.docker.com/linux/ubuntu focal/stable amd64 docker-ce amd64 5:23.0.1-1-ubuntu.20.04-focal [22.0 MB]
Des:7 https://download.docker.com/linux/ubuntu focal/stable amd64 docker-ce-rootless-extras amd64 5:23.0.1-1-ubuntu.20.04-focal [8 765 kB]
Des:8 https://download.docker.com/linux/ubuntu focal/stable amd64 docker-compose-plugin amd64 2.16.0-1-ubuntu.20.04-focal [10.2 MB]
Des:9 https://download.docker.com/linux/ubuntu focal/stable amd64 docker-scan-plugin amd64 0.23.0-ubuntu-focal [3 622 kB]
Descargados 112 MB en 11s (9 860 kB/s)
Seleccionando el paquete pigz previamente no seleccionado.
(Leyendo la base de datos ... 284416 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../0-pigz_2.4-1_amd64.deb ...
Desempaquetando pigz (2.4-1) ...
Seleccionando el paquete containerd.io previamente no seleccionado.
Preparando para desempaquetar .../1-containerd.io_1.6.18-1_amd64.deb ...
Desempaquetando containerd.io (1.6.18-1) ...
Seleccionando el paquete docker-buildx-plugin previamente no seleccionado.
Preparando para desempaquetar .../2-docker-buildx-plugin_0.10.2-1-ubuntu.20.04-focal_amd64.deb ...
Desempaquetando docker-buildx-plugin (0.10.2-1-ubuntu.20.04-focal) ...
```

Figura 7: Ejecución del comando del paso 6

7. Cuando se haya completado la descarga se ejecutará el comando

```
1 $ sudo systemctl status docker
2
```

Para comprobar que Docker se haya instalado y revisar algunos otros detalles como su versión y su estatus actual.

Si se quiere salir del comando presione **Ctrl-z** para terminar con su ejecución y continuar con la instalación.

```
joel@UbuntuInspiron:~$ 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 2023-02-25 19:07:18 CST; 32s ago
     TriggeredBy: ● docker.socket
       Docs: https://docs.docker.com
    Main PID: 11191 (dockerd)
      Tasks: 14
     Memory: 24.3M
    CGroup: /system.slice/docker.service
            └─11191 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

feb 25 19:07:17 UbuntuInspiron dockerd[11191]: time="2023-02-25T19:07:17.337067090-06:00" level=info msg="[core] [Channel #4 SubChannel #5] S
feb 25 19:07:17 UbuntuInspiron dockerd[11191]: time="2023-02-25T19:07:17.337125903-06:00" level=info msg="[core] [Channel #4] Channel Connect
feb 25 19:07:17 UbuntuInspiron dockerd[11191]: time="2023-02-25T19:07:17.810967703-06:00" level=info msg="Loading containers: start."
feb 25 19:07:18 UbuntuInspiron dockerd[11191]: time="2023-02-25T19:07:18.391897893-06:00" level=info msg="Default bridge (docker0) is assigne
feb 25 19:07:18 UbuntuInspiron dockerd[11191]: time="2023-02-25T19:07:18.569968557-06:00" level=info msg="Loading containers: done."
feb 25 19:07:18 UbuntuInspiron dockerd[11191]: time="2023-02-25T19:07:18.738971716-06:00" level=info msg="Docker daemon" commit=bc3805a graph
feb 25 19:07:18 UbuntuInspiron dockerd[11191]: time="2023-02-25T19:07:18.739151980-06:00" level=info msg="Daemon has completed initialization"
feb 25 19:07:18 UbuntuInspiron dockerd[11191]: time="2023-02-25T19:07:18.905561261-06:00" level=info msg="[core] [Server #7] Server created"
feb 25 19:07:18 UbuntuInspiron systemd[1]: Started Docker Application Container Engine.
feb 25 19:07:18 UbuntuInspiron dockerd[11191]: time="2023-02-25T19:07:18.913306712-06:00" level=info msg="API listen on /run/docker.sock"
lines 1-21/21 (END)
```

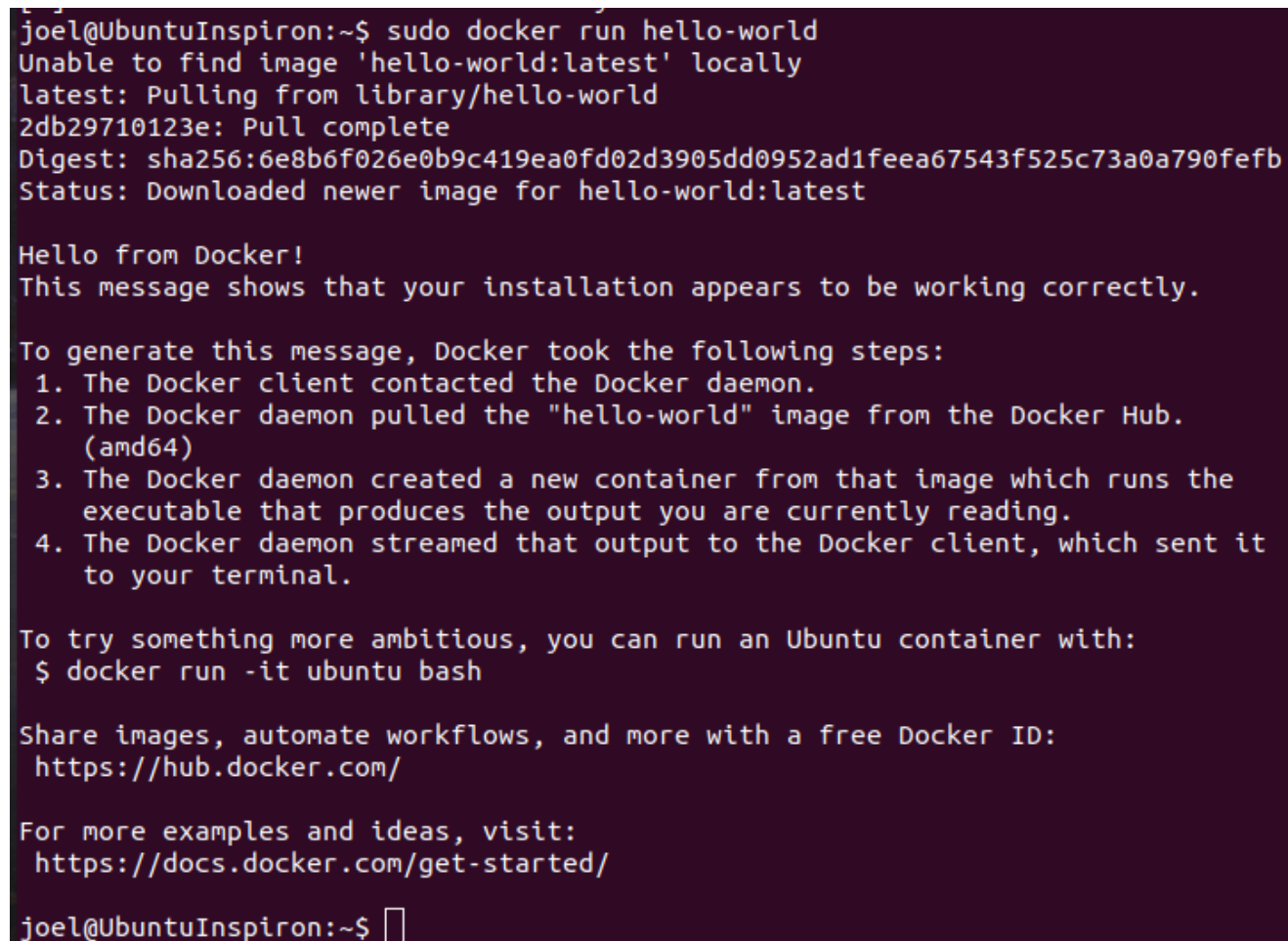
Figura 8: Ejecución del comando del paso 7

8. Finalmente se ejecuta el comando

```
1 $ sudo docker run hello-world
2
```

El cuál descarga y ejecuta una imagen de ejemplo para comprobar que se pueda trabajar con imágenes de Docker.

Si en la terminal aparece el mensaje `This message shows that your installation appears to be working coorrectly` o uno similar, entonces significa que la instalación fue exitosa y ya se puede utilizar Docker.



```
joel@UbuntuInspiron:~$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:6e8b6f026e0b9c419ea0fd02d3905dd0952ad1fee67543f525c73a0a790fefb
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

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

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

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

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

joel@UbuntuInspiron:~$
```

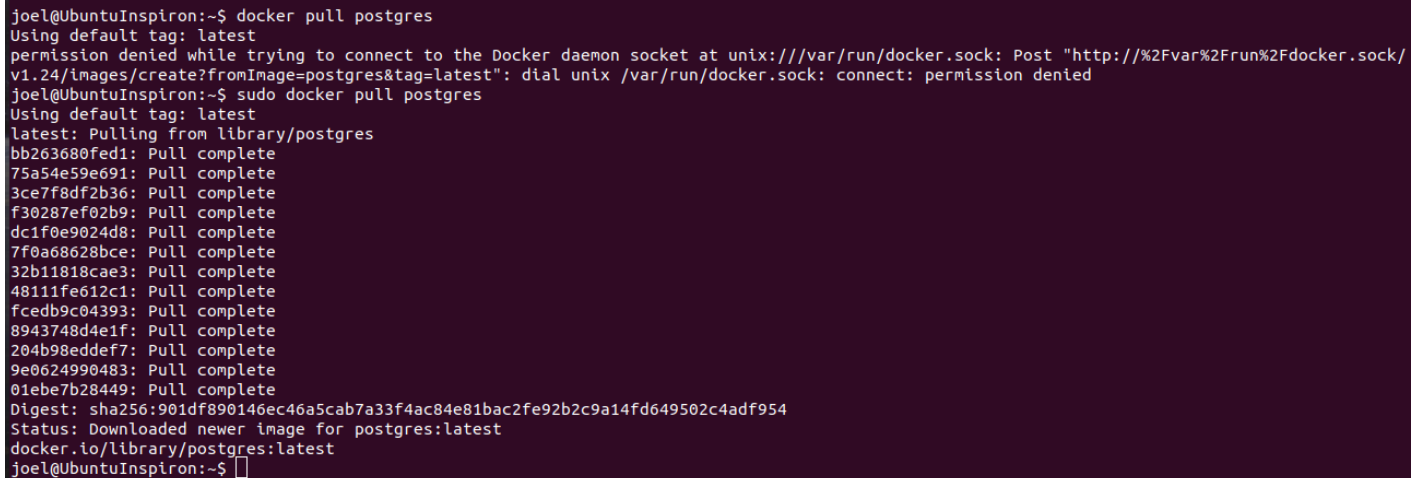
Figura 9: Ejecución del comando del paso 8

Pasos para usar PostgreSQL con Docker

A continuación se dan los pasos seguidos para usar PostgreSQL con Docker:

1. Primero hay que descargar la imagen de postgres, el cual Docker utilizara al momento de crear el contenedor, mediante el comando

```
1 $ sudo docker pull postgres
2
```



```
joel@UbuntuInspiron:~$ docker pull postgres
Using default tag: latest
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Post "http://%2Fvar%2Frun%2Fdocker.sock/v1.24/images/create?fromImage=postgres&tag=latest": dial unix /var/run/docker.sock: connect: permission denied
joel@UbuntuInspiron:~$ sudo docker pull postgres
Using default tag: latest
latest: Pulling from library/postgres
bb263680fed1: Pull complete
75a54e59e691: Pull complete
3ce7f8df2b36: Pull complete
f30287ef02b9: Pull complete
dc1f0e9024d8: Pull complete
7f0a68628bce: Pull complete
32b11818cae3: Pull complete
48111fe612c1: Pull complete
fcedb9c04393: Pull complete
8943748d4e1f: Pull complete
204b98eddef7: Pull complete
9e0624990483: Pull complete
01ebe7b28449: Pull complete
Digest: sha256:901df890146ec46a5cab7a33f4ac84e81bac2fe92b2c9a14fd649502c4adf954
Status: Downloaded newer image for postgres:latest
docker.io/library/postgres:latest
joel@UbuntuInspiron:~$
```

Figura 10: Descarga de la imagen de PostgreSQL

2. Ya que se haya descargado la imagen de postgres, a continuación se creará un contenedor nuevo con el siguiente comando

```
1 $ sudo docker run -d --name postgres -e POSTGRES_PASSWORD=mysecretpassword  
-p 5432:5432 postgres  
2
```

El **nombre** del contenedor es el que va después de la directiva **--name** (en este caso se dejó como postgres)

La **contraseña** para acceder es la que está después de **POSTGRES_PASSWORD=** y se debe colocar una contraseña que se pueda recordar ya que en caso contrario no será posible acceder al contenedor. Finalmente el **puerto** de la máquina local y la del contenedor es el que se da después de la directiva **--p** (en este caso se dejó como 5432:5432 pero si ya hay un contenedor en ese puerto se debe escoger otro distinto).

```
joel@UbuntuInspiron:~$ sudo docker run -d --name postgres -e POSTGRES_PASSWORD=joelreyes -p 5432:5432 postgres  
b966d070aabdfca587c8bb4b77d429a12a06ca5f02d05b228371bd193384f01e  
joel@UbuntuInspiron:~$
```

Figura 11: Contenedor creado para la instalación

3. Y para verificar que el contenedor fue creado y se está ejecutando se comprueba con el comando

```
1 $ sudo docker ps -a  
2
```

Esto nos mostrará una lista con todos los contenedores que fueron creados en Docker junto a algunos detalles como su ID y su status actual

```
joel@UbuntuInspiron:~$ sudo docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
b966d070aabdfca587c8bb4b77d429a12a06ca5f02d05b228371bd193384f01e	postgres	"docker-entrypoint.s..."	27 seconds ago	Up 25 seconds	0.0.0.0:5432->5432/tcp, :::5432->5432/tcp
5dcd8bc18f86	hello-world	"/hello"	3 minutes ago	Exited (0) 3 minutes ago	
flamboyant_panini					

```
joel@UbuntuInspiron:~$
```

Figura 12: Status de los contenedores creados y ejecutados en Docker

Instalación y ejecución de DBeaver

Una vez que se ha creado el contenedor de postgres lo siguiente es instalar la herramienta de administración de bases de datos, en este caso se eligió DBeaver por lo que los pasos a continuación serán los que se siguieron para instalarlo en el equipo.

1. Primero hay que ir a la página oficial de descargas de DBeaver, la cuál se puede acceder con el siguiente enlace: <https://dbeaver.io/download/>
Y una vez ahí se elige la versión para el sistema operativo en el que se desea instalar (en este caso el paquete *Linux Debian package (installer)*)

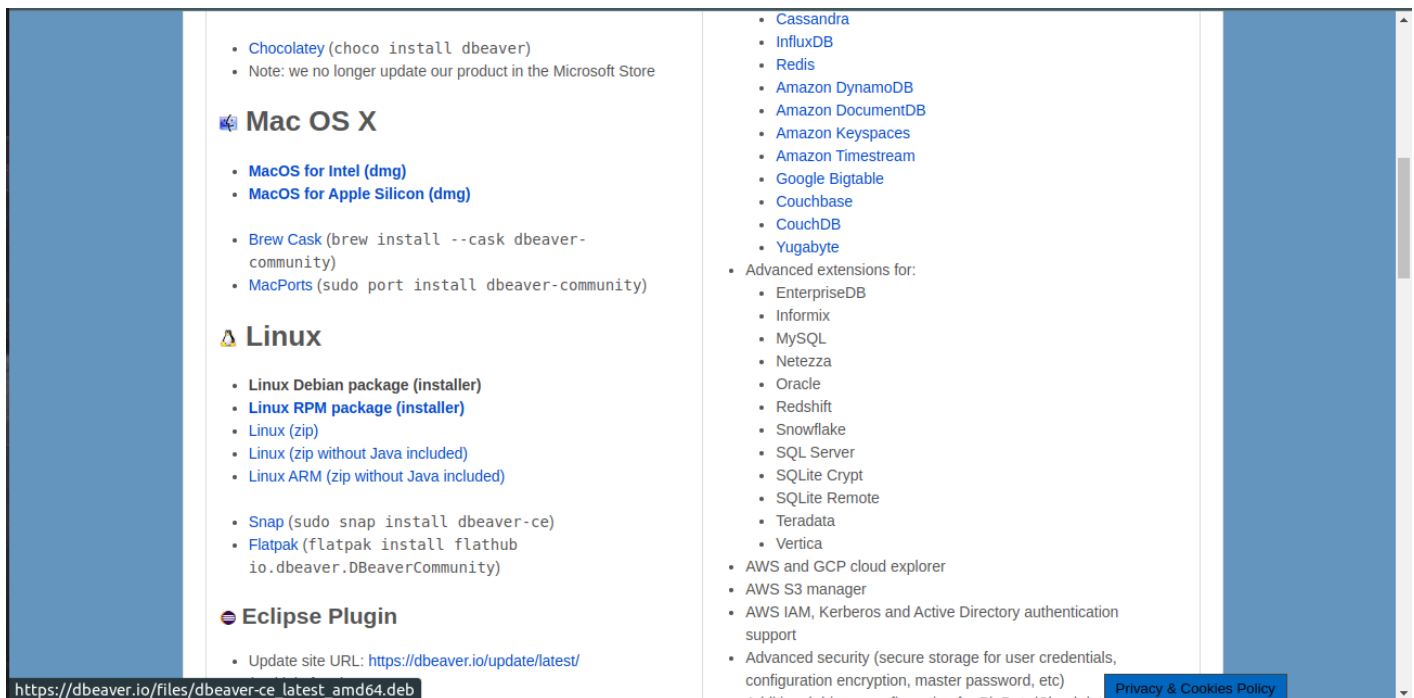


Figura 13: Página de descargas de DBeaver

2. Una vez que se ha descargado el archivo ir a la carpeta de descargas (o donde se guarden los archivos descargados) y ejecutarlo con la aplicación *Instalar software* (en el caso de Ubuntu) para que instale el programa

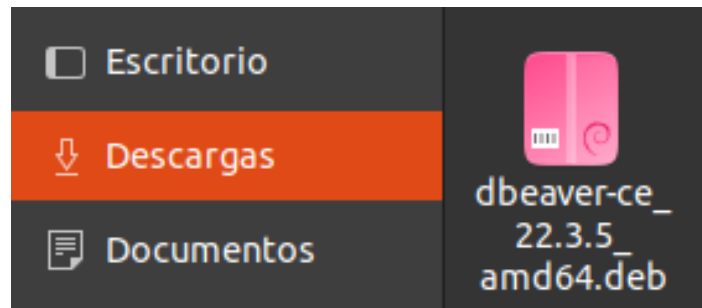


Figura 14: Archivo de descarga de DBeaver

3. Cuando se haya instalado el programa debe aparecer un programa en el equipo con el nombre de `dbeaver.ce` (o uno muy similar) con el logo del programa

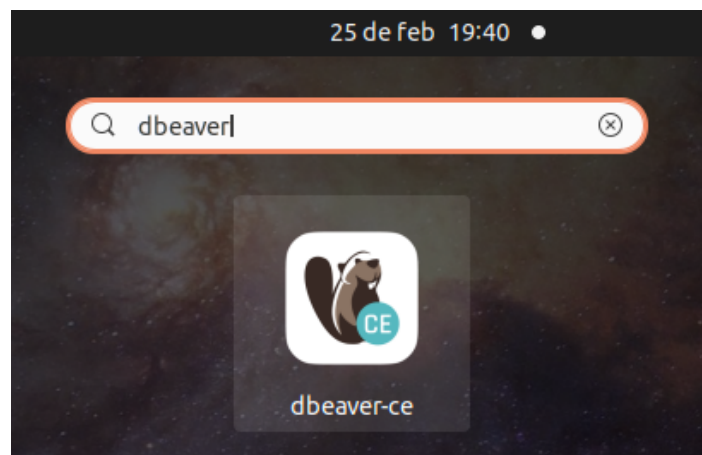


Figura 15: Aplicación de DBeaver en el equipo

4. Al abrir el programa lo primero que aparecerá son dos ventanas, y una de ellas nos dirá que seleccionemos el sistema manejador de base de datos, en este caso se seleccionó PostgreSQL porque es con el que se va a trabajar.

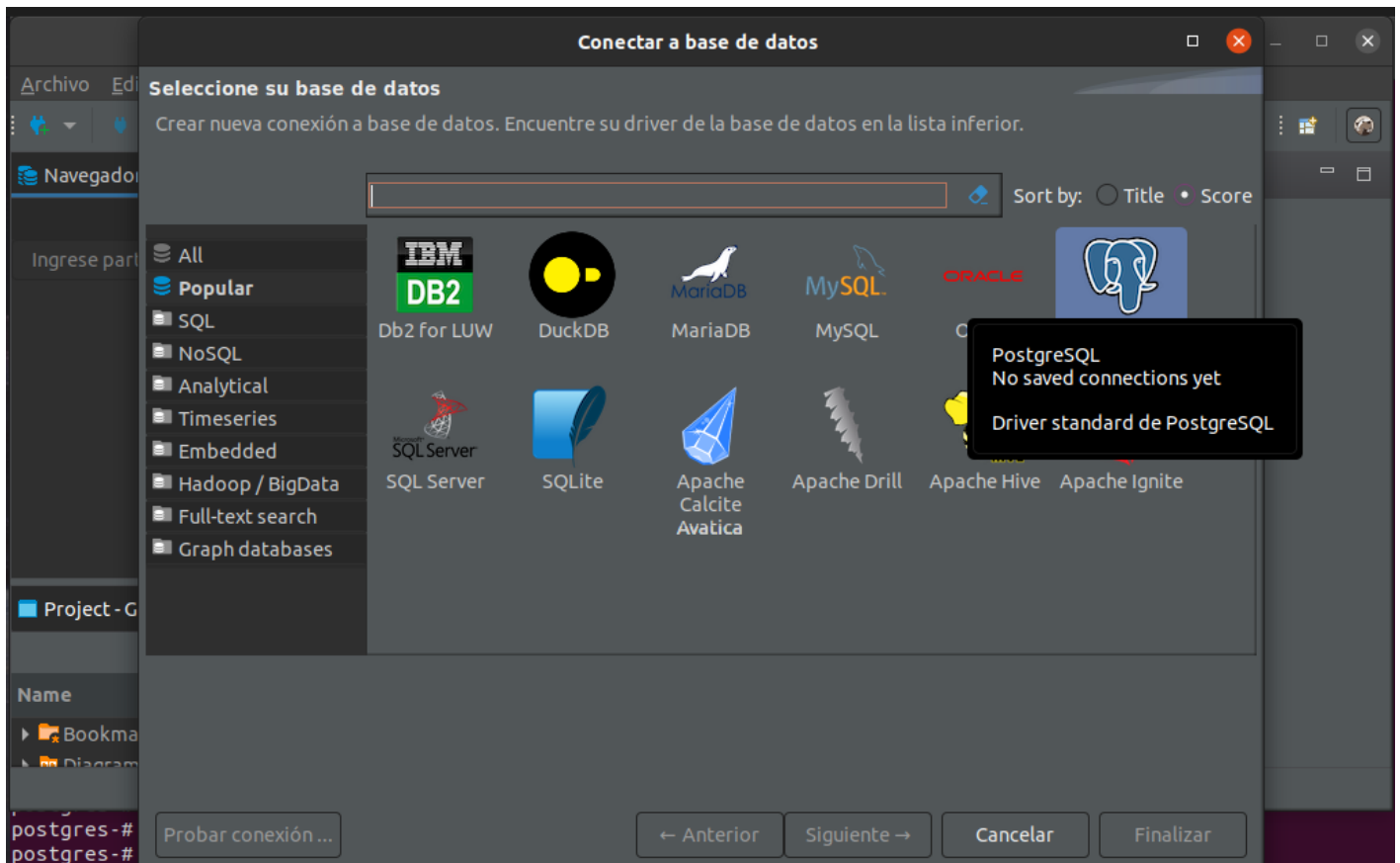


Figura 16: Ventana inicial de DBeaver

5. Una vez seleccionado el SMBD aparecerá otra ventana donde debemos colocar los datos para conectarnos a la base de datos, en este caso los mismos datos que se colocaron al crear el contenedor en Docker (puerto, nombre de usuario, contraseña y el nombre de la base de datos).

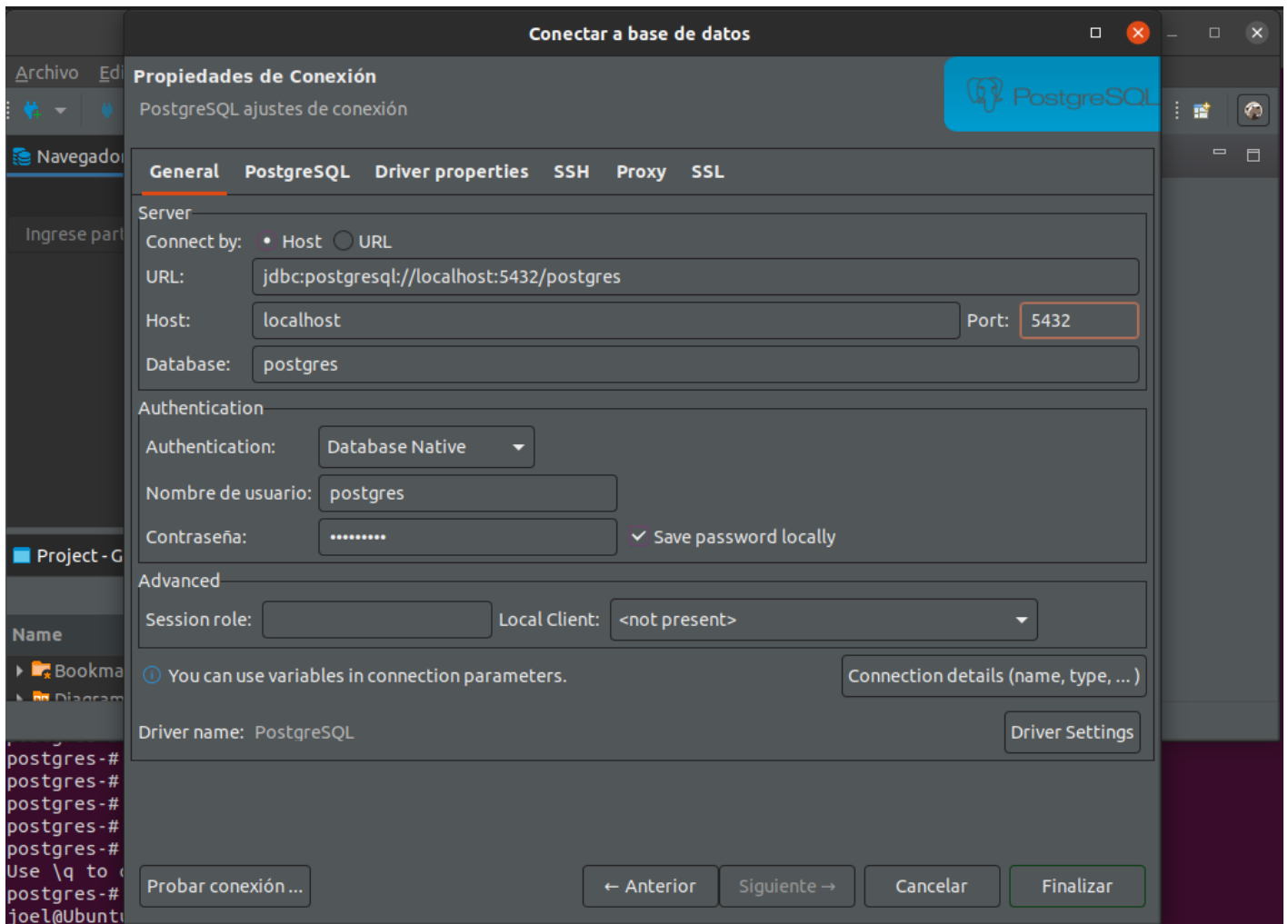


Figura 17: Ventana Conectar a base de datos

6. Cuando ya se hayan colocado los datos se debe dar clic en el botón *Probar conexión...* situado en la esquina inferior izquierda de la ventana, y al hacerlo aparecerá una ventana que pedirá descargar unos archivos extra necesarios para trabajar con PostgreSQL, por lo que le damos en el botón *Download* para comenzar con la descarga

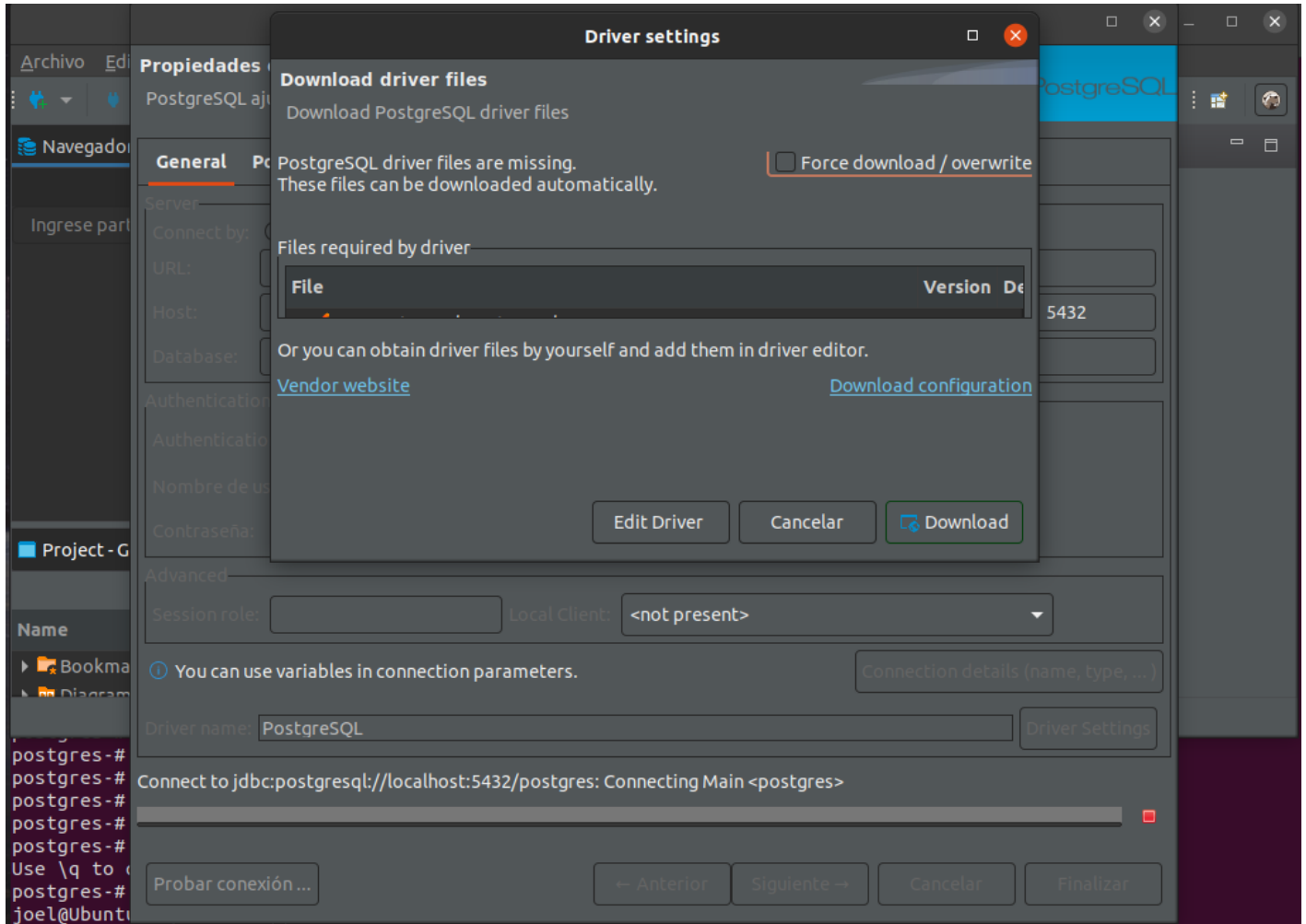


Figura 18: Ventana Descargar Drivers

7. Al terminar la descarga aparecerá una ventana con los detalles de conexión, y si aparece el mensaje *Conectado* significa que la conexión con el contenedor de Docker fue exitosa, por lo que se presiona el botón *Aceptar* para continuar, y después en el botón *Finalizar* de la ventana *Conectar a base de datos*.

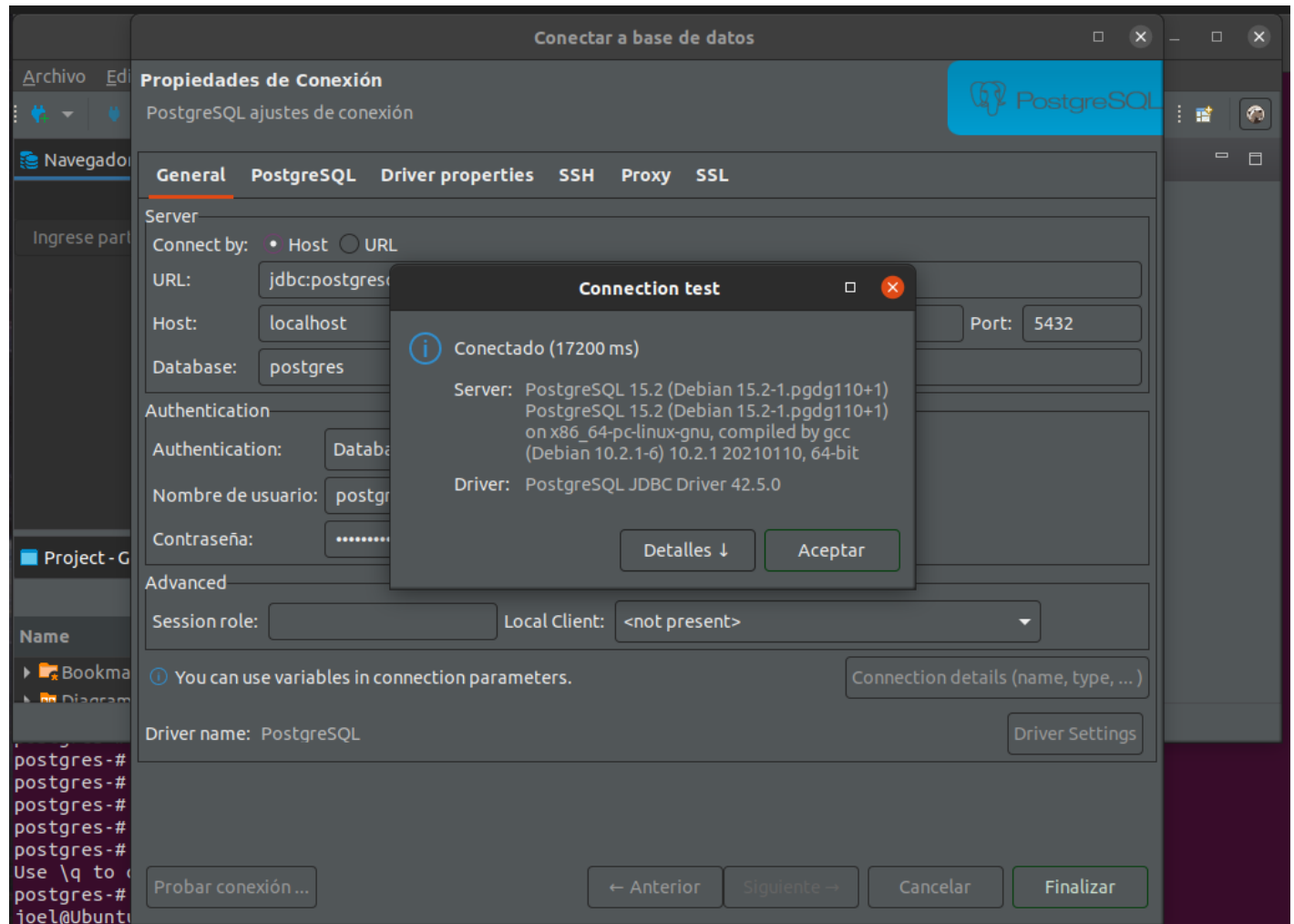


Figura 19: Ventana Test conexión

8. Finalmente aparecerá la ventana principal y en el lado derecho la base de datos que se creó en Docker, por lo tanto con esto la instalación y conexión a la base de datos terminó de forma exitosa.

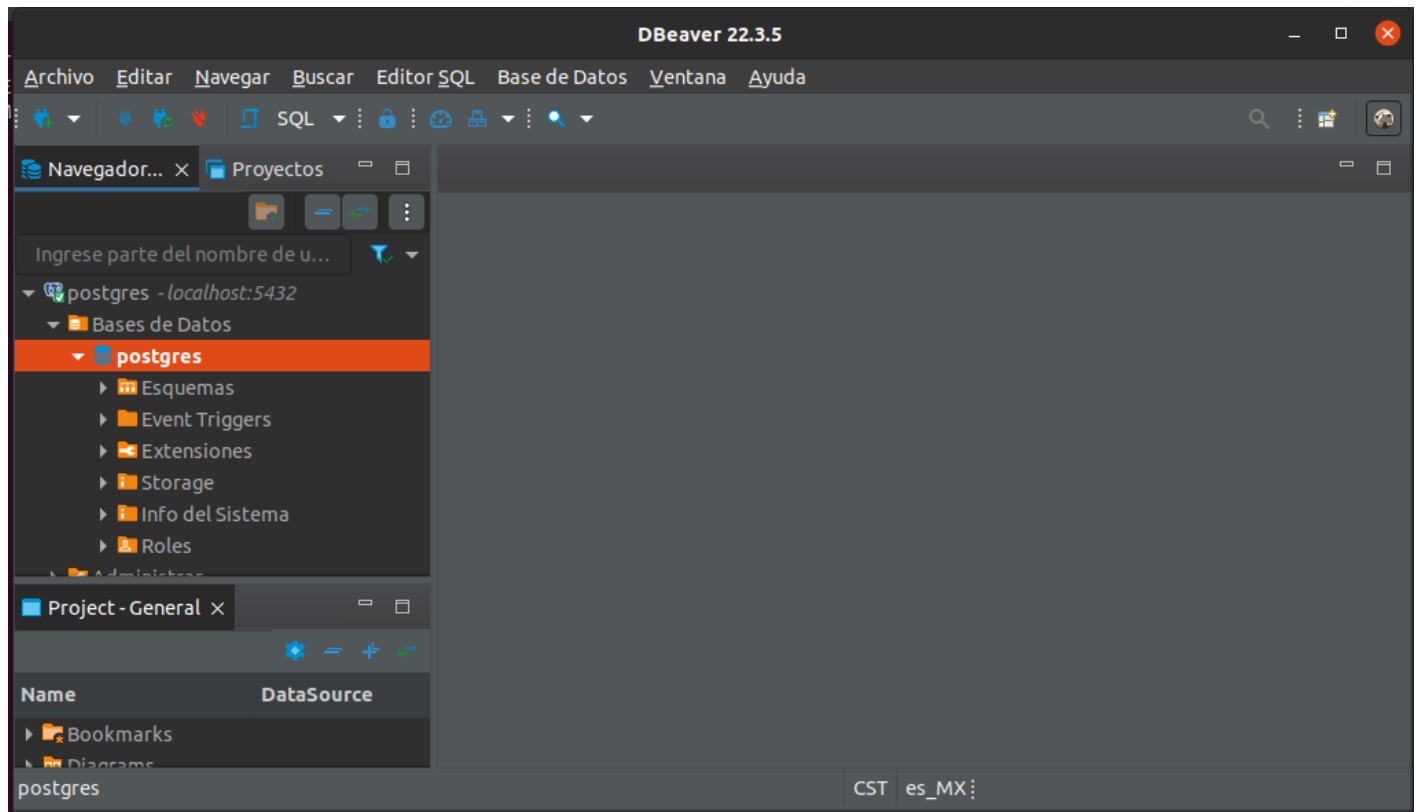


Figura 20: Ventana Principal Base de Datos

Comentarios y problemas surgidos en la instalación

El proceso de instalación de manera general ocurrió sin ningún problema mayor, los únicos inconvenientes que surgieron fue que algunos comandos no fueron ejecutados con la directiva **sudo** y eso impidió que el sistema diera permiso para instalar y correr Docker, pero gracias a la aclaración dada por el ayudante de laboratorio se pudo solucionar fácilmente.

Para concluir quiero comentar que la decisión de usar DBeaver sobre pgAdmin4 fue debido a que, si bien pgAdmin4 está hecho para poder usarse de mejor manera con PostgreSQL, DBeaver también permite usar otros SMD o hasta el uso de bases de datos NoSQL, y dado que en el futuro es probable que me vea usando otros SMD aparte de PostgreSQL es por eso que escogí instalar DBeaver en mi equipo para familiarizarme más con él y usarlo a largo plazo.