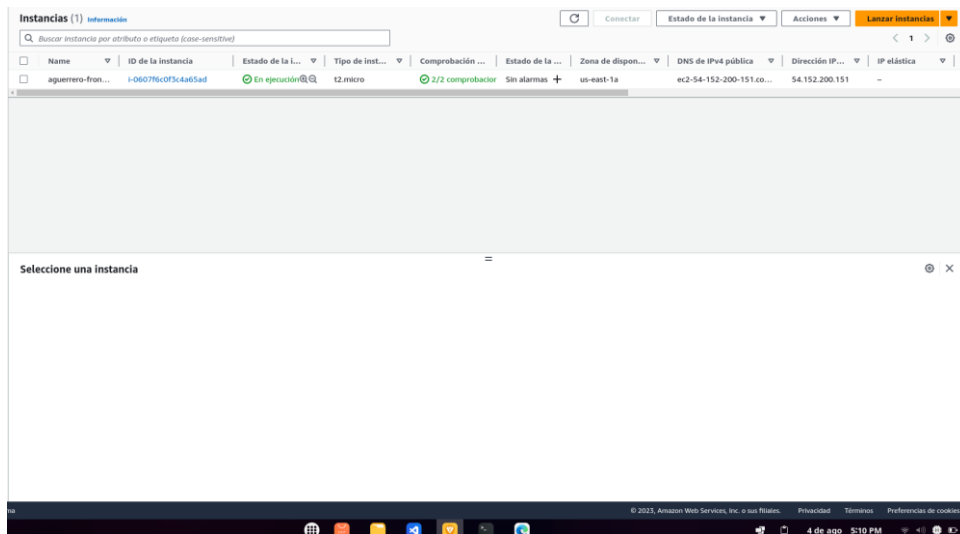


Creando instancia.



```
ubuntu@ip-172-31-81-131:~$ neofetch

      .-/+oossssoo+/-.
      `:+ssssssssssssss++:`
      -+ssssssssssssssyyssss+-
      .ossssssssssssssdMMMMyssso.
      /ssssssssshdmmNNmyNMMMhsssss/
      +ssssssssshmydMMMMMMNdddyssssss+
      /ssssssshNMMMyhhyyyhmNMMNhsssss/
      .sssssssdMMMhssssssshNMMMdssssss.
      +ssshhhyNMMNysssssssssyNMMMyssssss+
      ossyNMMMyMMhssssssssshmmhssssssso
      ossyNMMMyMMhssssssssshmmhssssssso
      +ssshhhyNMMNysssssssssyNMMMyssssss+
      .sssssssdMMMhssssssshNMMMdssssss.
      /ssssssshNMMMyhhyyyhdNMMNhsssss/
      +sssssssdmydMMMMMMNdddyssssss+
      /ssssssssshdmmNNmyNMMMhsssss/
      .ossssssssssssssdMMMMyssso.
      -+ssssssssssssssyyssss+-
      `:+ssssssssssssss++:`
      .-/+oossssoo+/-.

OS: Ubuntu 20.04.6 LTS x86_64
Host: HVM domU 4.11.amazon
Kernel: 5.15.0-1036-aws
Uptime: 1 min
Packages: 792 (dpkg), 6 (snap)
Shell: bash 5.0.17
Terminal: /dev/pts/0
CPU: Intel Xeon E5-2676 v3 (1) @ 2.400GHz
GPU: 00:02.0 Cirrus Logic GD 5446
Memory: 194MiB / 966MiB

ubuntu@ip-172-31-81-131:~$ node --version
v18.17.0
ubuntu@ip-172-31-81-131:~$
```

## Práctica de la Lección

Una vez ingresado a la consola de la instancia ejecutar los siguientes comandos:

- Listar el contenido del directorio actual

```
ubuntu@ip-172-31-81-131:~$ ls
app-angular  express
ubuntu@ip-172-31-81-131:~$
```

- Ejecutar comando que indica la ruta de directorio actual.

```
+ ubuntu@ip-172-31-81-131: ~
ubuntu@ip-172-31-81-131:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-81-131:~$
```

- Crear un directorio llamado workdir y cambiarse al directorio.

```
+ ubuntu@ip-172-31-81-131: ~/workdir
ubuntu@ip-172-31-81-131:~$ mkdir workdir
ubuntu@ip-172-31-81-131:~$ cd workdir/
ubuntu@ip-172-31-81-131:~/workdir$
```

- Crear un archivo llamado **welcome.txt** que tenga como contenido “Bienvenidos a Ubuntu”; despliega el contenido del archivo a través de línea de comandos.

```
+ ubuntu@ip-172-31-81-131: ~/workdir
GNU nano 4.8 welcome.txt
Bienvenidos a Ubuntu
+ ubuntu@ip-172-31-81-131: ~/workdir
ubuntu@ip-172-31-81-131:~/workdir$ cat welcome.txt
Bienvenidos a Ubuntu
ubuntu@ip-172-31-81-131:~/workdir$
```

- Ejecutar comando que instala la última versión de los paquetes

```
+ ubuntu@ip-172-31-81-131: ~/workdir
ubuntu@ip-172-31-81-131:~/workdir$ cat welcome.txt
Bienvenidos a Ubuntu
ubuntu@ip-172-31-81-131:~/workdir$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-backports InRelease
Hit:4 https://download.docker.com/linux/ubuntu focal InRelease
Hit:5 http://security.ubuntu.com/ubuntu focal-security InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
All packages are up to date.
ubuntu@ip-172-31-81-131:~/workdir$
```

## Creando aplicación de angular.

1. Crear un directorio de trabajo.

```
ubuntu@ip-172-31-81-131:~$ node --version
v18.17.0
ubuntu@ip-172-31-81-131:~$ mkdir app-angular
ubuntu@ip-172-31-81-131:~$ cd app-angular/
ubuntu@ip-172-31-81-131:~/app-angular$
```

2. Instalar Angular CLI

`npm install -g @angular/cli@latest`

```
ubuntu@ip-172-31-81-131:~/app-angular$ sudo npm install -g @angular/cli
added 240 packages in 11s
36 packages are looking for funding
  run `npm fund` for details
ubuntu@ip-172-31-81-131:~/app-angular$
```

3. Crear una nueva aplicación Angular

`ng new ejemplo --strict --style=scss --routing --skip-tests`

```
ubuntu@ip-172-31-81-131:~/app-angular$ ng new ejemplo --strict --style=scss --routing --skip-tests
Warning: This is a simple server for use in testing or debugging Angular applications
locally. It hasn't been reviewed for security issues.

Binding this server to an open connection can result in compromising your application or
computer. Using a different host than the "--host" flag might result in
websocket connection issues. You might need to use "--disable-host-check" if that's the
case.

? Generating browser application bundles (phase: building)...
[Progress bar]
```

4. Configurar la instancia EC2, seleccionar pestaña Seguridad y agregar las nuevas reglas

Detalles
**Seguridad**
Redes
Almacenamiento
Comprobaciones de estado
Monitoreo
Etiquetas

▼ Detalles de seguridad

Rol de IAM

-

ID del propietario

147750570083

Hora de lanzamiento

Fri Aug 04 20:

Grupos de seguridad

sg-071399a8893be581c (launch-wizard-7)

▼ Reglas de entrada

Q

Filtrar reglas

Nombre	ID de la regla del grupo d...	Intervalo de pu...	Protocolo	Origen
-	sgr-004f8c451cb933c0b	80	TCP	0.0.0.0/0
-	sgr-072ba9f740ee7309d	443	TCP	0.0.0.0/0
-	sgr-078d03b4c56beeeaf	4200	TCP	0.0.0.0/0
-	sgr-0cf6f5d60385ee469	22	TCP	0.0.0.0/0

5. Iniciar el servidor para conocer ejecución de la herramienta

`ng serve --host 0.0.0.0`

6. Copiar la IP pública de la instancia y pegarla en el navegador Web, acceder a través del protocolo HTTP y agregar el puerto 4200 de angular

Resumen de instancia de i-0c9183ff8af577de6 (backend-linux)
Información

Conectar
Estado de la instancia
Acciones

ID de la instancia
i-0c9183ff8af577de6 (backend-linux)

Dirección IPv6
-

Tipo de nombre de anfitrión
Nombre de IP: ip-172-31-6-75.us-east-2.compute.internal

Responder al nombre DNS de recurso privado IPv4 (A)
Dirección IP asignada automáticamente
3.139.236.189 [IP pública]

Rol de IAM
-

IMDSv2
Optional

Dirección IPv4 pública
3.139.236.189 | dirección abierta

Estado de la instancia
En ejecución

Nombre DNS de IP privada (solo IPv4)
ip-172-31-6-75.us-east-2.compute.internal

Tipo de instancia
t2.micro

ID de VPC
vpc-06047a1607b56abb9

ID de subred
subnet-012d800bf62e1e35

Direcciones IPv4 privadas
172.31.6.75

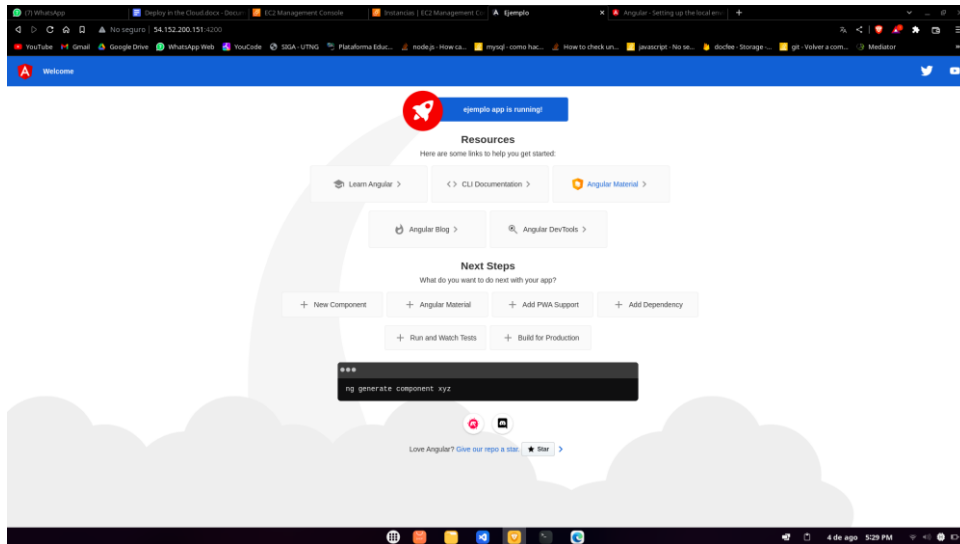
DNS de IPv4 pública
ec2-3-139-236-189.us-east-2.compute.amazonaws.com | dirección abierta

Direcciones IP elásticas
-

Hallazgo de AWS Compute Optimizer
Suscribirse a AWS Compute Optimizer para recibir recomendaciones.
Más información

Nombre del grupo de Auto Scaling
-

7. Abrir un navegador web e ingresar la dirección IP junto con su puerto



## Referencia

```
sudo apt update
```

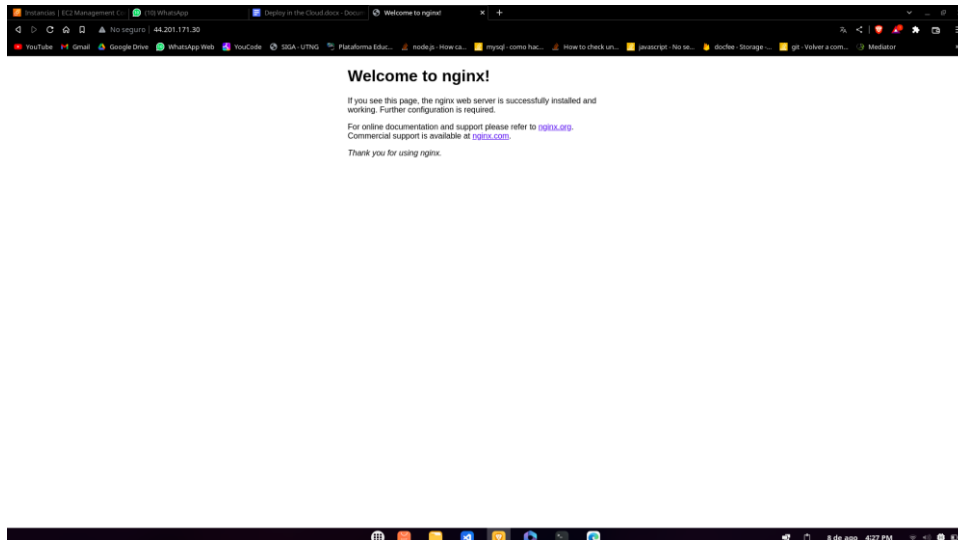
## Instalar el servidor Web Nginx con el comando

```
sudo apt install nginx
```

## Verificar el estado del servidor web

## systemctl status nginx

Verificar qué se levantó de manera correcta el servidor web, ingresa la IP Pública de EC2



Ahora vamos a construir el proyecto test, ingresar el comando

**ng build**

```
ubuntu@ip-172-31-81-131: ~/app-angular/ejemplo
ubuntu@ip-172-31-81-131:~/app-angular/ejemplo$ ng build
✓ Browser application bundle generation complete.
✓ Copying assets complete.
✓ Index html generation complete.

Initial Chunk Files | Names | Raw Size | Estimated Transfer Size
main.1bec130984be6491.js | main | 204.54 kB | 55.94 kB
polyfills.65f60585fbb1ab6c.js | polyfills | 33.01 kB | 10.68 kB
runtime.55dc05f423a7eca3.js | runtime | 892 bytes | 507 bytes
styles.ef46db3751d8e999.css | styles | 0 bytes | -
| Initial Total | 238.42 kB | 67.11 kB

Build at: 2023-08-08T22:31:54.573Z - Hash: ce0887e27ead5c87 - Time: 10012ms
ubuntu@ip-172-31-81-131:~/app-angular/ejemplo$
```

Ingresar el comando para el listado de carpetas y archivos

**ls -l**

```
ubuntu@ip-172-31-81-131: ~/app-angular/ejemplo
ubuntu@ip-172-31-81-131:~/app-angular/ejemplo$ ls -l
total 476
-rw-r--r-- 1 root root 740 Aug 5 00:00 Dockerfile
-rw-r--r-- 1 ubuntu ubuntu 1061 Aug 4 23:21 README.md
-rw-r--r-- 1 ubuntu ubuntu 3468 Aug 4 23:21 angular.json
drwxrwxr-x 3 ubuntu ubuntu 4096 Aug 8 22:31 dist
drwxrwxr-x 560 ubuntu ubuntu 20480 Aug 4 23:22 node_modules
-rw-r--r-- 1 ubuntu ubuntu 426856 Aug 4 23:22 package-lock.json
-rw-r--r-- 1 ubuntu ubuntu 1038 Aug 4 23:21 package.json
drwxrwxr-x 4 ubuntu ubuntu 4096 Aug 4 23:21 src
-rw-r--r-- 1 ubuntu ubuntu 263 Aug 4 23:21 tsconfig.app.json
-rw-r--r-- 1 ubuntu ubuntu 901 Aug 4 23:21 tsconfig.json
-rw-r--r-- 1 ubuntu ubuntu 273 Aug 4 23:21 tsconfig.spec.json
ubuntu@ip-172-31-81-131:~/app-angular/ejemplo$
```

Para conocer la ruta de directorio actual ingresar comando

**pwd**

```
ubuntu@ip-172-31-81-131: ~/app-angular/ejemplo
ubuntu@ip-172-31-81-131:~/app-angular/ejemplo$ pwd
/home/ubuntu/app-angular/ejemplo
ubuntu@ip-172-31-81-131:~/app-angular/ejemplo$
```

Copiar la ruta y agregar la carpeta dist/test, por ejemplo:

`/home/ubuntu/workdir/test/dist/test`

Modificar el archivo de configuración de Nginx

`sudo nano /etc/nginx/sites-available/default`

```
ubuntu@ip-172-31-41-131: ~/app-angular/ejemplo
GNU nano 4.8 /etc/nginx/sites-available/default
#
# Self signed certs generated by the ssl-cert package
# Don't use them in a production server!
#
# include snippets/snakeoil.conf;

root /home/ubuntu/app-angular/ejemplo;

# Add index.php to the list if you are using PHP
index index.html index.htm index.nginx-debian.html;

server_name _;

location / {
    # First attempt to serve request as file, then
```

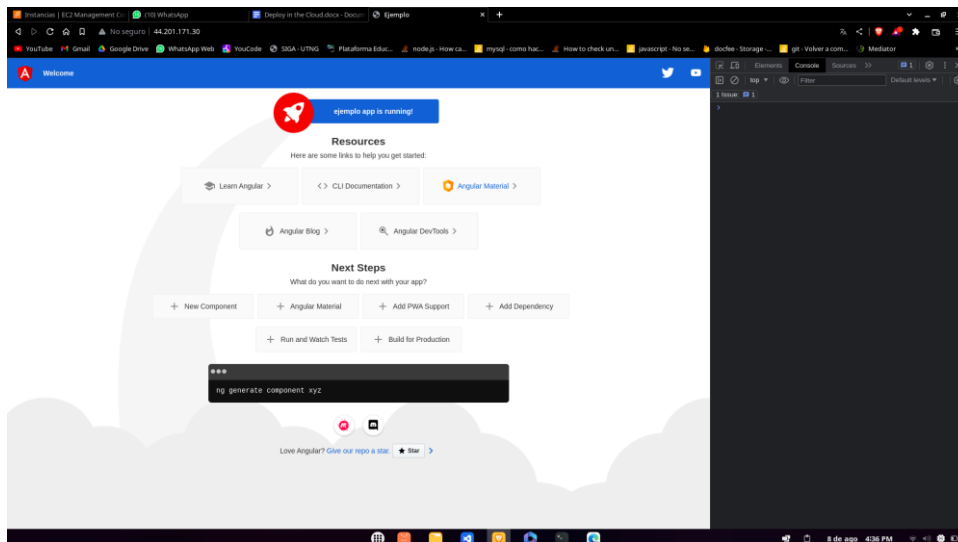
Modificar línea: root `/var/www/html`; Por ruta: `/home/ubuntu/workdir/test/dist/test`

Reiniciar servidor seb con comando

`sudo systemctl restart nginx`

```
ubuntu@ip-172-31-41-131: ~/app-angular/ejemplo
ubuntu@ip-172-31-41-131:~/app-angular/ejemplo$ sudo systemctl restart nginx
ubuntu@ip-172-31-41-131:~/app-angular/ejemplo$
```

Ingresar nuevamente la IP de la dirección pública de EC2 en su navegador





# Configuración HTTPS

Crear una carpeta llamada ssl

**mkdir ssl**

```
ubuntu@ip-172-31-81-131: ~/$ mkdir ssl
ubuntu@ip-172-31-81-131: ~/$ cd ssl/
ubuntu@ip-172-31-81-131: ~/ssl$
```

Instalar openssl

**sudo apt install openssl**

```
ubuntu@ip-172-31-81-131: ~/ssl$ sudo apt install openssl
Reading package lists... Done
Building dependency tree
Reading state information... Done
openssl is already the newest version (1.1.1f-1ubuntu2.19).
The following packages were automatically installed and are no longer required:
  gyp javascript-common libauthen-sasl-perl libbc-ares2 libdata-dump-perl libdrm-amdgpu libdrm-intel libdrm-nouveau libdrm-radeon libencode-locale-perl
  libfile-basedir-perl libfile-desktopentry-perl libfile-listing-perl libfile-mimeinfo-perl libfont-afm-perl libfontenc1 libgl1 libgl1-mesa-dri
  libglapi-mesa libglvnd0 libglx-mesa0 libglx0 libhtml-form-perl libhtml-format-perl libhtml-parser-perl libhtml-tagset-perl libhtml-tree-perl
  libhttp-cookies-perl libhttp-daemon-perl libhttp-date-perl libhttp-message-perl libhttp-negotiate-perl libice6 libio-html-perl libio-socket-ssl-perl
  libio-stringy-perl libipc-system-simple-perl libjs-inherits libjs-is-typedarray libjs-psl libjs-typedarray-to-buffer libllvm12 liblwp-mediatypes-perl
  liblwp-protocol-https-perl libmailtools-perl libnet-dbus-perl libnet-http-perl libnet-smtp-ssl-perl libnet-sni-perl libnode-dev libnode64 libpciaccess0
  libpython2-stdeb libpython2.7-minimal libpython2.7-stdlib libpython3-stdlib libpython3.9-minimal libpython3.9-stdlib libqt5core5a libqt5dbus5 libqt5gui5
  libqt5network5 libqt5opengl5 libqt5widgets5 libqt5xml5 libqtcore5 libqtgui5 libqtwidgets5 libqtxml5 libx11-xcb1 libxaw7 libxcb-dri2-0 libxcb-dri3-0
  libxcb-glx0 libxcb-present0 libxcb-randr0 libxcb-shape0 libxcb-shm0 libxcb-sync1 libxcb-xfixes0 libxcomposite1 libxcursor1 libxfixes3 libxft2
  libxi6 libxinerama1 libxkbfile1 libxml-parser-perl libxml-twig-perl libxml-xpathengine-perl libxmu6 libxrandr2 libxrender1 libxshmfence1 libxt6
  libxtst6 libxv1 libxxf86dga1 libxxf86vm1 mesa-vulkan-drivers nodejs-doc perl-openssl-defaults python-pkg-resources python2
  python2-minimal python2.7 python2.7-minimal x11-common x11-utils x11-xserver-utils xdg-utils
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
ubuntu@ip-172-31-81-131: ~/ssl$
```

Conocer versión de openssl

**openssl version**

```
ubuntu@ip-172-31-81-131: ~/ssl$ openssl version
OpenSSL 1.1.1f  31 Mar 2020
ubuntu@ip-172-31-81-131: ~/ssl$
```

Generar una clave privada llamada **clave-privada.key**

**openssl genpkey -algorithm RSA -out clave-privada.key**

```
ubuntu@ip-172-31-81-131: ~/ssl$ openssl genpkey -algorithm RSA -out clave-privada.key
.....+++++
.....+++++
ubuntu@ip-172-31-81-131: ~/ssl$ ls
clave-privada.key
ubuntu@ip-172-31-81-131: ~/ssl$
```

Escribir comando para verificar qué se creo

Generar un certificado en base a la clave privada recién creada, responde a las interrogantes como en imagen de ejemplo

**openssl req -new -key clave-privada.key -out cert.csr**

```
ubuntu@ip-172-31-81-131: ~$ openssl req -new -key clave-privada.key -out cert.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:MX
State or Province Name (full name) [Some-State]:GTO
Locality Name (eg, city) []:DH
Organization Name (eg, company) [Internet Widgits Pty Ltd]:UTNG
Organizational Unit Name (eg, section) []:TIC'S
Common Name (e.g. server FQDN or YOUR name) []:DESWEB
Email Address []:alejandro02262@gmail.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:linux123
An optional company name []:UTNG
ubuntu@ip-172-31-81-131: ~$
```

Generar nuevamente certificado firmado

**openssl x509 -req -days 365 -in /home/ubuntu/workdir/ssl/cert.csr -signkey /home/ubuntu/workdir/ssl/clave-privada.key -out /home/ubuntu/workdir/ssl/cert-auto.crt**

```
ubuntu@ip-172-31-81-131: ~$ openssl x509 -req -days 365 -in /home/ubuntu/ssl/cert.csr -signkey /home/ubuntu/ssl/clave-privada.key -out /home/ubuntu/ssl/cert-auto.crt
Signature ok
subject=C = MX, ST = GTO, L = DH, O = UTNG, OU = TIC'S, CN = DESWEB, emailAddress = alejandro02262@gmail.com
Getting Private key
ubuntu@ip-172-31-81-131: ~$
```

Unir ambos certificados para mayor seguridad

**cat cert-auto.crt clave-privada.key > cert-completo.crt**

```
-rw-rw-r-- 1 ubuntu ubuntu 1281 Aug  8 19:02 cert-auto.crt
-rw-rw-r-- 1 ubuntu ubuntu 2989 Aug  8 19:03 cert-completo.crt
-rw-rw-r-- 1 ubuntu ubuntu 1098 Aug  8 19:00 cert.csr
-rw----- 1 ubuntu ubuntu 1708 Aug  8 18:58 clave-privada.key
```

```
ubuntu@ip-172-31-81-131: ~$ cat cert-auto.crt clave-privada.key > cert-completo.crt
ubuntu@ip-172-31-81-131: ~$
```

Modificar archivo en servidor web Nginx

**sudo nano /etc/nginx/sites-available/default**

```
GNU nano 4.8 /etc/nginx/sites-available/default
# Please see /usr/share/doc/nginx-doc/examples/ for more detailed examples.
##

# Default server configuration
#
server {
    listen 80 default_server;
    listen [::]:80 default_server;

    # SSL configuration
    #
    listen 443 ssl default_server;
    ssl_certificate /home/ubuntu/ssl/cert-completo.crt;
    ssl_certificate_key /home/ubuntu/ssl/clave-privada.key;
    # listen [::]:443 ssl default_server;
    #
```

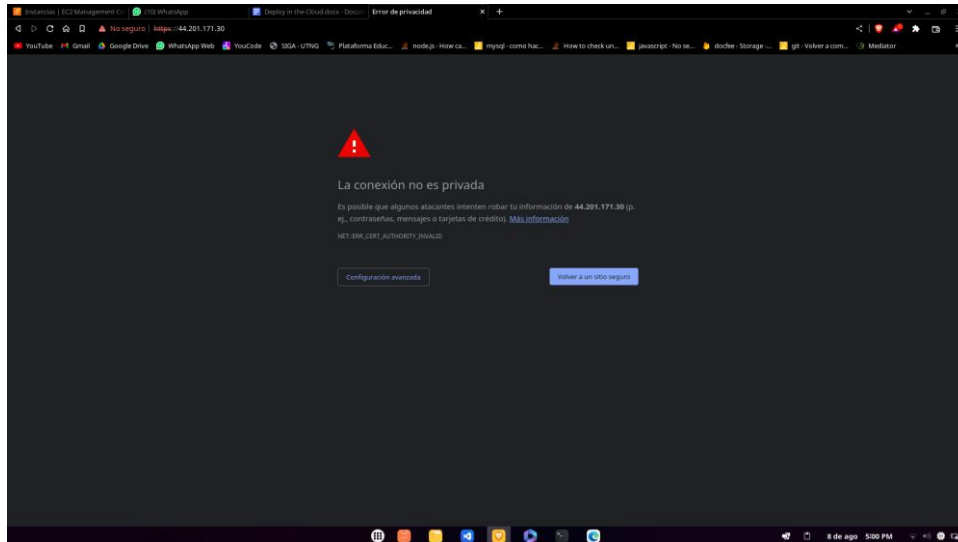
CTRL + S Guardar

CTRL + X Salir

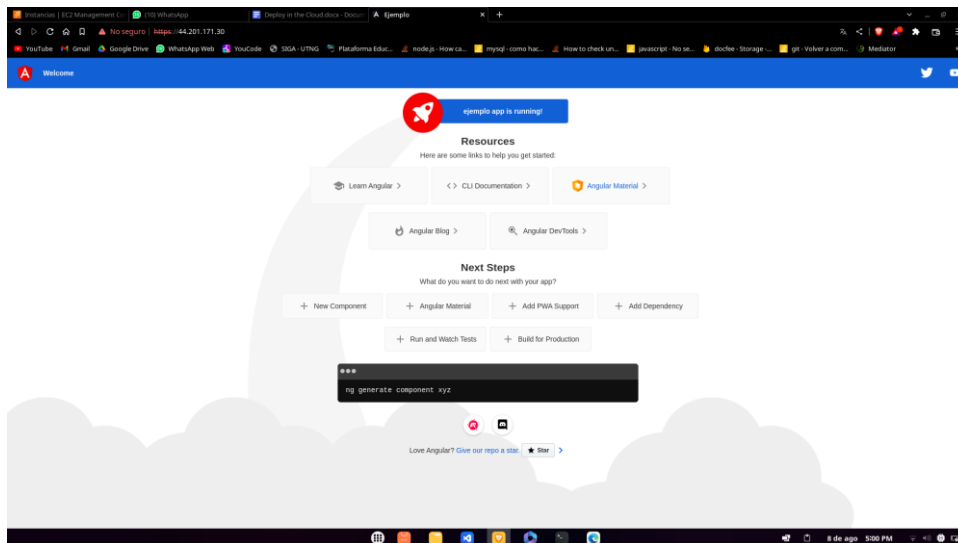
Reiniciar nuevamente el servidor

**sudo systemctl restart nginx**

Abrir navegador web con la direccion: <https://44.201.171.30/>



Seleccionar configuración avanzada | Continuar a 18.233.156.132 (no seguro)



## Instalar MySQL

Crear una nueva instancia donde se deposite la base de datos.



```
ubuntu@ip-172-31-93-176:~$ sudo systemctl start mysql.service
ubuntu@ip-172-31-93-176:~$ sudo systemctl status mysql
● mysql.service - MySQL Community Server
   Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2023-08-08 23:35:57 UTC; 26s ago
     Main PID: 2552 (mysqld)
    Status: "Server is operational"
       Tasks: 38 (limit: 1141)
      Memory: 355.5M
        CGroup: /system.slice/mysql.service
                └─2552 /usr/sbin/mysqld

Aug 08 23:35:56 ip-172-31-93-176 systemd[1]: Starting MySQL Community Server...
Aug 08 23:35:57 ip-172-31-93-176 systemd[1]: Started MySQL Community Server.
ubuntu@ip-172-31-93-176:~$
```

```
ubuntu@ip-172-31-93-176:~$ sudo mysql_secure_installation

Securing the MySQL server deployment.

Connecting to MySQL using a blank password.

VALIDATE PASSWORD COMPONENT can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No: y

There are three levels of password validation policy:

LOW      Length >= 8
MEDIUM  Length >= 8, numeric, mixed case, and special characters
STRONG Length >= 8, numeric, mixed case, special characters and dictionary file

Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: 0

Skipping password set for root as authentication with auth_socket is used by default.
If you would like to use password authentication instead, this can be done with the "ALTER USER" command.
See https://dev.mysql.com/doc/refman/8.0/en/alter-user.html#alter-user-password-management for more information.

By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : Y
Success.

Normally, root should only be allowed to connect from
```

Crear la base de datos web\_integral;

```
ubuntu@ip-172-31-93-176:~$ mysql -u aguerrero -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.33-0ubuntu0.20.04.4 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

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

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

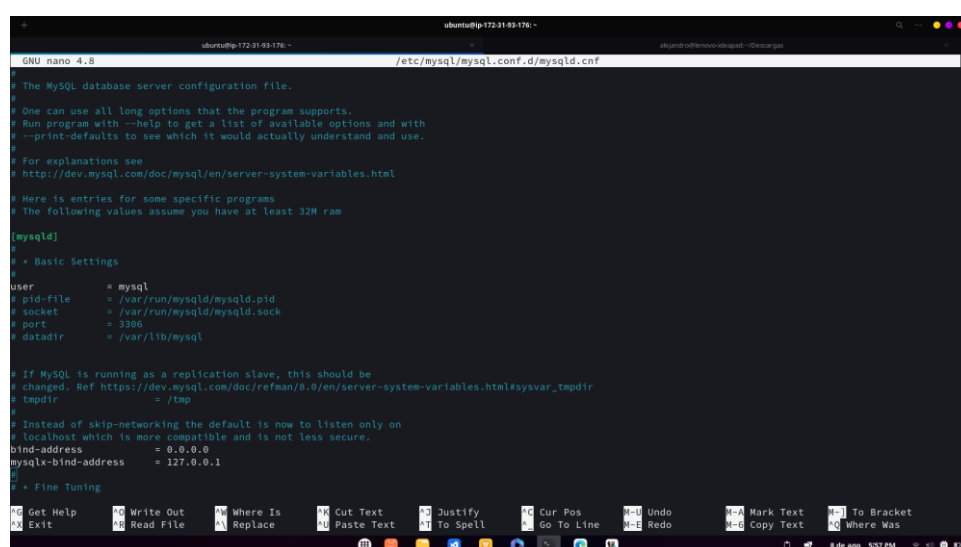
mysql> CREATE DATABASE web_integral;
Query OK, 1 row affected (0.05 sec)

mysql> use web_integral;
Database changed
mysql>
```

Creando tabla de usuario.

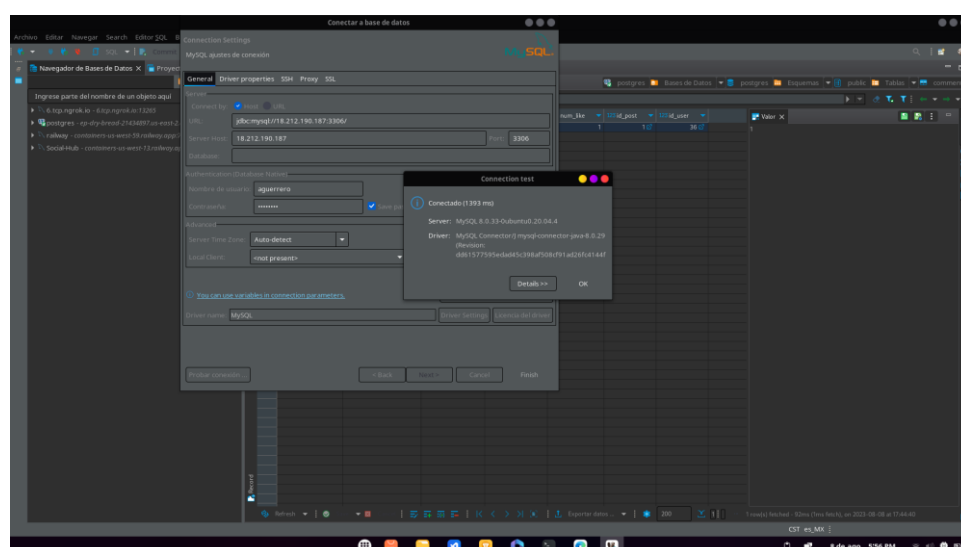
```
mysql> CREATE TABLE tbl_usuario (  
->     Username VARCHAR(30) PRIMARY KEY,  
->     Password VARCHAR(250) NOT NULL,  
->     Role VARCHAR(20) NOT NULL  
-> );  
Query OK, 0 rows affected (0.04 sec)  
  
mysql> 
```

## Configurar base de datos para acceso remoto



## Instalar [DBeaver](#) para acceder a la BD

pasar los parámetros de conexión



## Práctica de la lección

- Crear la base de datos de aplicación de ejemplo.

```
+ ubuntu@ip-172-31-93-176:~$ mysql -u aguerrero -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.33-0ubuntu0.20.04.4 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

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

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

mysql> CREATE DATABASE web_integral;
Query OK, 1 row affected (0.05 sec)

mysql> use web_integral;
Database changed
mysql>

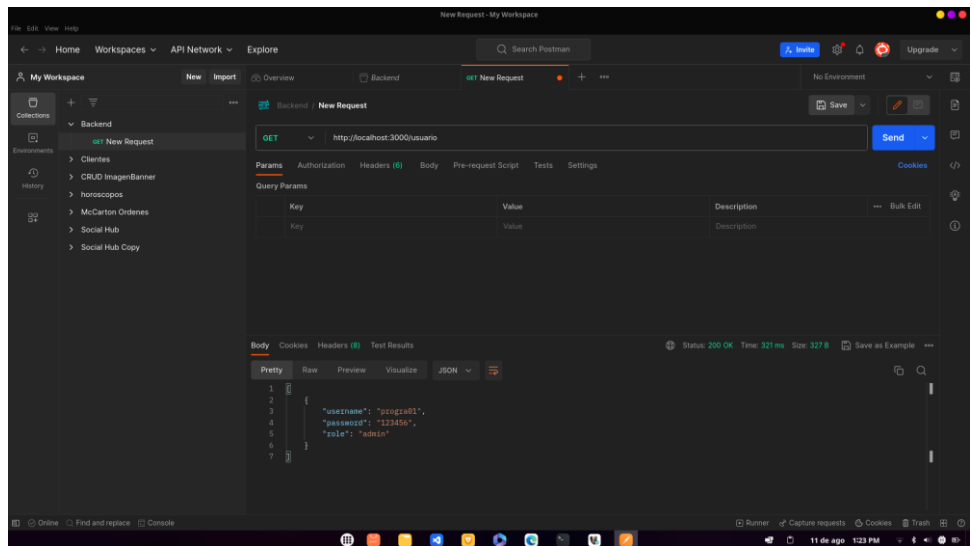
mysql> CREATE TABLE tbl_usuario (
  ->     Username VARCHAR(30) PRIMARY KEY,
  ->     Password VARCHAR(250) NOT NULL,
  ->     Role VARCHAR(20) NOT NULL
  -> );
Query OK, 0 rows affected (0.04 sec)

mysql>
```

- Montar el backend

### Configurando backend

- Verificar funcionalidad con Postman





## Creando contenedor Docker-Angular

```
ubuntu@ip-172-31-81-131: ~/app-angular/ ejemplo
GNU nano 4.8 Dockerfile
# Etapa 1: Compila y construye codebase de Angular
# Utilizar la última imagen oficial de Node
FROM node:latest as build

# Establecer el directorio de trabajo
WORKDIR /usr/local/app

# Agregar el código fuente al directorio app
COPY ./ /usr/local/app/

# Instalar las dependencias
RUN npm install

# Generar y correr la aplicación
RUN npm run build

# Instalar la herramienta Web  with nginx

# Uso official nginx image as the base image
# Utilizar la imagen oficial nginx como imagen base
FROM nginx:latest

AG Get Help      AR Write Out      AW Where Is      AK Cut Text      AJ Justify      AC Cur Pos      MU Undo
AX Exit          AR Read File     AL Replace       AU Paste Text    AT To Spell     AL Go To Line   ME Redo
```

Ejecutando imagen.

The screenshot shows a terminal window with the following content:

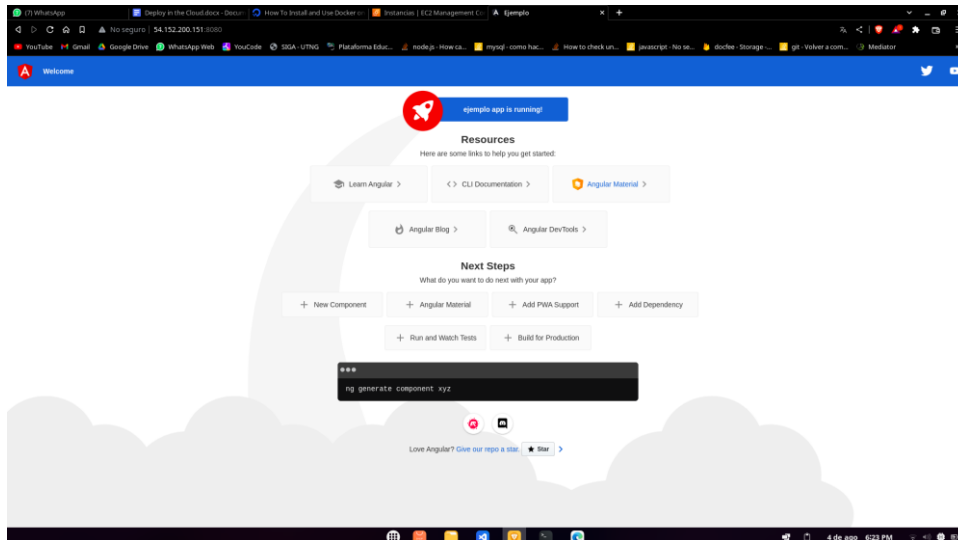
```

ubuntu@ip-172-31-81-131: ~/app-angular/example
$
ubuntu@ip-172-31-81-131: ~/app-angular/example$ docker run -d -p 8080:80 ejsample:1.0.1
bfe3bb98e2e59b26cd0b0147e436e149cf6d6e958ab82ac31fd6a8afad

```

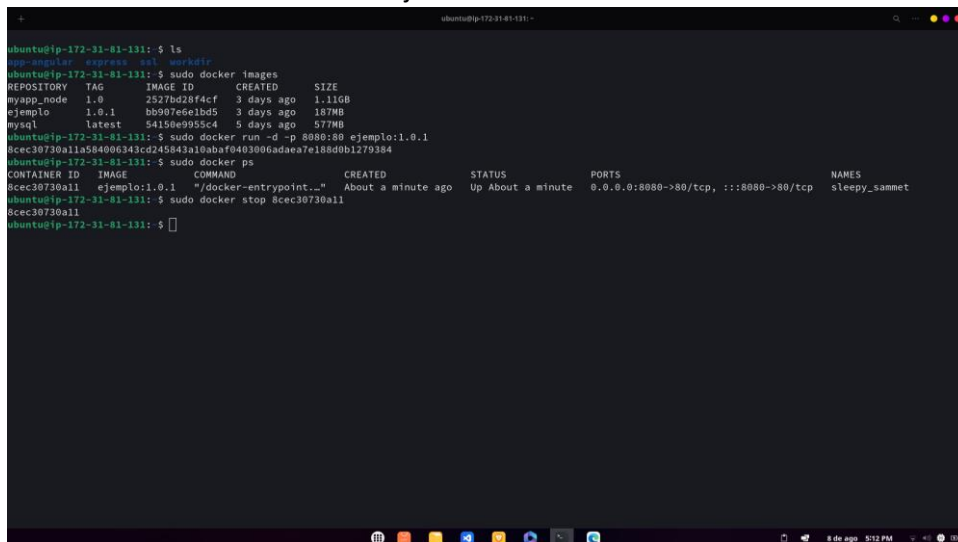
The terminal output shows the command `docker run -d -p 8080:80 ejsample:1.0.1` being executed, resulting in a long alphanumeric string representing the container ID.

## Probando en la nube.

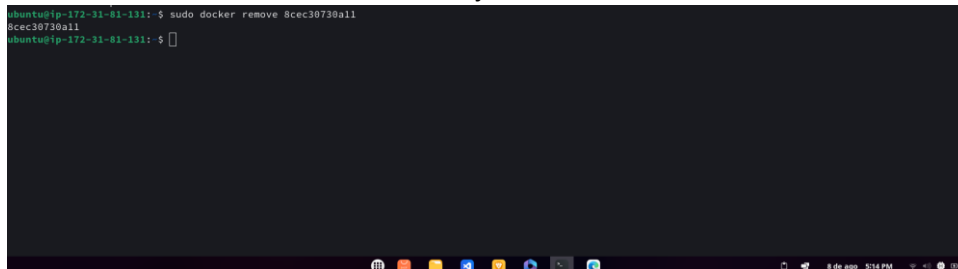


# Práctica de la sesión

- Para contenedor recién ejecutado.



- Eliminar el contenedor recién ejecutado



- Eliminar la imagen

```
+ ubuntu@ip-172-31-81-131: ~
ubuntu@ip-172-31-81-131:~$ sudo docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
myapp_node    1.0       2527bd28f4cf   3 days ago    1.11GB
ejemplo       1.0.1     bb907e6e1bd5   3 days ago    187MB
mysql         latest    54150e9955c4   5 days ago    577MB
ubuntu@ip-172-31-81-131:~$ sudo docker rmi bb907e6e1bd5
Untagged: ejemplo:1.0.1
Deleted: sha256:bb907e6e1bd591f46eae2f4ded4268d74f676dbd9e1845f9eb2e6d88aba6f6ff
ubuntu@ip-172-31-81-131:~$
```

- Listar todas las imágenes

```
ubuntu@ip-172-31-81-131:~$ sudo docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
myapp_node    1.0       2527bd28f4cf   3 days ago    1.11GB
mysql         latest    54150e9955c4   5 days ago    577MB
ubuntu@ip-172-31-81-131:~$
```

## Docker NodeJs

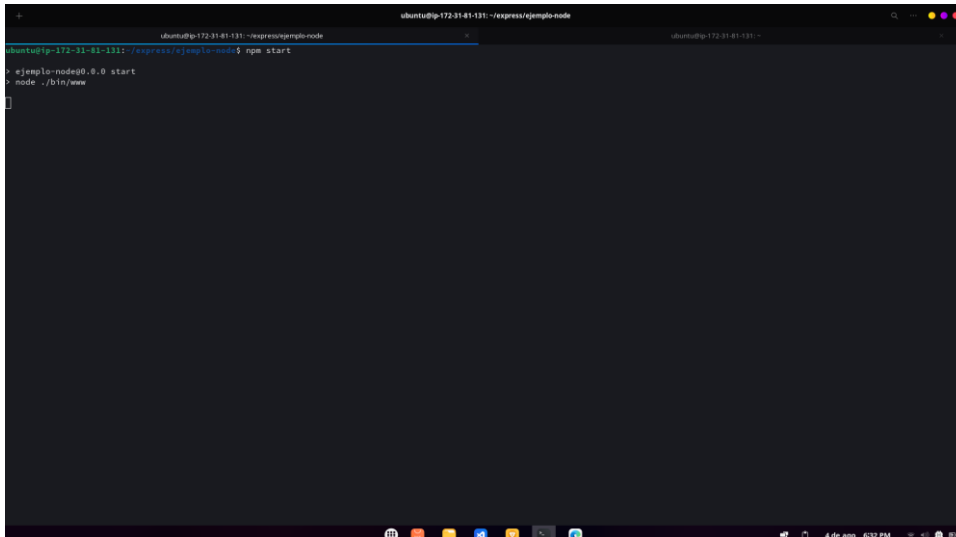
Ubicar un directorio de trabajo para alojar la aplicación.

Instalar express-generator, para mayor referencia véase [AQUI](#)

`npm install express-generator`

Crear el proyecto

`express ejemplo-node`



Ingresar la URL <http://localhost:3000>

Parar el servidor Node

Ingresar `Ctrl+C`

## Creando el contenedor Docker para Node

Crear un archivo **Dockerfile** en el directorio raíz del proyecto e ingresar las siguientes instrucciones

```
FROM node:latest as build
WORKDIR /usr/local/app
COPY package*.json /usr/local/app/
RUN npm install
COPY . /usr/local/app/
CMD [ "npm", "start" ]
```

```
+
ubuntu@ip-172-31-81-131: ~/express/ejemplo-node
GNU nano 4.8
FROM node:latest as build

WORKDIR /usr/local/app

COPY package*.json /usr/local/app/

RUN npm install

COPY ./ /usr/local/app/

CMD [ "npm", "start" ]
```

Crear la imagen

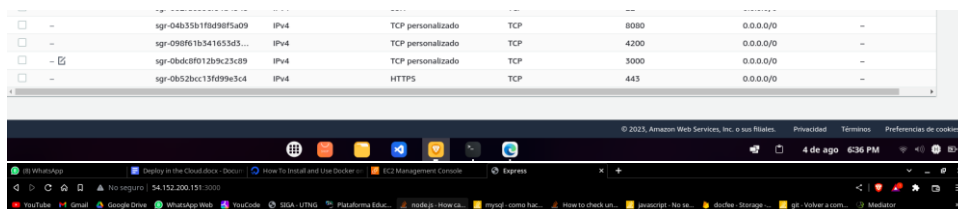
```
sudo docker build -t myapp_node:1.0 .
```

Ejecutar el contenedor

```
sudo docker run -p 3000:3000 myapp_node:1.0
```

```
ubuntu@ip-172-31-81-131:~/express/ejemplo-node$ sudo docker run -p 3000:3000 myapp_node:1.0
ejemplo-node@0.0.0 start
node ./bin/www
```

Agregando regla en aws



# Docker MySQL

Extraer la imagen de mysql

**docker pull mysql:latest**

```
ubuntu@ip-172-31-81-131:~$ sudo docker pull mysql:latest
latest: Pulling from library/mysql
49bb46380f8c: Pull complete
aab3066bbf8f: Pull complete
d6eef8c26cf9: Pull complete
0e908b1dcba2: Pull complete
480c3912a2fd: Pull complete
ef90fc42d4db: Pull complete
a2ffc585c753: Pull complete
e2ef842ff3d6: Pull complete
c6c990e874d7: Pull complete
a554d403eafe: Pull complete
Digest: sha256:6a5dbd2810e36048669639811461f27fee48d4e22039e5d31f4273a20d542f6
Status: Downloaded newer image for mysql:latest
docker.io/library/mysql:latest
ubuntu@ip-172-31-81-131:~$
```

Ejecutar el siguiente comando donde creará automáticamente la imagen con una contraseña y el volumen para almacenar la información

**docker run --name mysql-demo -e MYSQL\_ROOT\_PASSWORD=demo -d -v mysql-demo:/var/lib/mysql mysql:latest**

**docker ps**

```
ubuntu@ip-172-31-81-131:~$ sudo docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                    NAMES
0d5f3067fea    mysql:latest   "docker-entrypoint.s..." 13 seconds ago Up 11 seconds   3306/tcp, 33060/tcp      mysql-demo
fb1665d1156d   myapp_node:1.0 "docker-entrypoint.s..." 5 minutes ago  Up 5 minutes   0.0.0.0:3000->3000/tcp, :::3000->3000/tcp  pe
062bfe5bb98e   ejemplo:1.0.1  "/docker-entrypoint.s..." 19 minutes ago Up 19 minutes   0.0.0.0:8080->80/tcp, :::8080->80/tcp      ga
llant_benz
```

Instalar cliente de MySQL

**sudo apt install mysql-client -y**

```
After this operation, 75.6 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-client-core-8.0 amd64 8.0.33-0ubuntu0.20.04.4 [5176 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/main amd64 mysql-common all 5.8+1.0.5ubuntu2 [7496 B]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-client-8.0 amd64 8.0.33-0ubuntu0.20.04.4 [22.0 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-client all 8.0.33-0ubuntu0.20.04.4 [9356 B]
Fetched 5215 kB in 0s (11.4 MB/s)
Selecting previously unselected package mysql-client-core-8.0.
(Reading database ... 72211 files and directories currently installed.)
Preparing to unpack .../mysql-client-core-8.0_8.0.33-0ubuntu0.20.04.4_amd64.deb ...
Unpacking mysql-client-core-8.0 (8.0.33-0ubuntu0.20.04.4) ...
Selecting previously unselected package mysql-client-8.0.
Preparing to unpack .../mysql-client-8.0_8.0.33-0ubuntu0.20.04.4_amd64.deb ...
Unpacking mysql-client-8.0 (8.0.33-0ubuntu0.20.04.4) ...
Selecting previously unselected package mysql-client.
Preparing to unpack .../mysql-client_8.0.33-0ubuntu0.20.04.4_all.deb ...
Unpacking mysql-client (8.0.33-0ubuntu0.20.04.4) ...
Setting up mysql-common (5.8+1.0.5ubuntu2) ...
update-alternatives: using /etc/mysql/my.cnf.fallback to provide /etc/mysql/my.cnf (my.cnf) in auto mode
Setting up mysql-client-core-8.0 (8.0.33-0ubuntu0.20.04.4) ...
Setting up mysql-client-8.0 (8.0.33-0ubuntu0.20.04.4) ...
Setting up mysql-client (8.0.33-0ubuntu0.20.04.4) ...
Processing triggers for man-db (2.9.1-1) ...
```

Habilitar la extracción de instalación y versión de MySQL

**which mysql**

**mysql --version**

```
After this operation, 75.6 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-client-core-8.0 amd64 8.0.33-0ubuntu0.20.04.4 [5176 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/main amd64 mysql-common all 5.8+1.0.5ubuntu2 [7496 B]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-client-8.0 amd64 8.0.33-0ubuntu0.20.04.4 [22.0 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-client all 8.0.33-0ubuntu0.20.04.4 [9356 B]
Fetched 5215 kB in 0s (11.4 MB/s)
Selecting previously unselected package mysql-client-core-8.0.
(Reading database ... 72211 files and directories currently installed.)
Preparing to unpack .../mysql-client-core-8.0_8.0.33-0ubuntu0.20.04.4_amd64.deb ...
Unpacking mysql-client-core-8.0 (8.0.33-0ubuntu0.20.04.4) ...
Selecting previously unselected package mysql-common.
Preparing to unpack .../mysql-common_5.8+1.0.5ubuntu2_all.deb ...
Unpacking mysql-common (5.8+1.0.5ubuntu2) ...
Selecting previously unselected package mysql-client-8.0.
Preparing to unpack .../mysql-client-8.0_8.0.33-0ubuntu0.20.04.4_amd64.deb ...
Unpacking mysql-client-8.0 (8.0.33-0ubuntu0.20.04.4) ...
Selecting previously unselected package mysql-client.
Preparing to unpack .../mysql-client-8.0.33-0ubuntu0.20.04.4_all.deb ...
Unpacking mysql-client (8.0.33-0ubuntu0.20.04.4) ...
Setting up mysql-common (5.8+1.0.5ubuntu2) ...
update-alternatives: using /etc/mysql/my.cnf.fallback to provide /etc/mysql/my.cnf (my.cnf) in auto mode
Setting up mysql-client-core-8.0 (8.0.33-0ubuntu0.20.04.4) ...
Setting up mysql-client-8.0 (8.0.33-0ubuntu0.20.04.4) ...
Setting up mysql-client (8.0.33-0ubuntu0.20.04.4) ...
Processing triggers for man-db (2.9.1-1) ...
ubuntu@ip-172-31-81-131:~$ which mysql
/usr/bin/mysql
ubuntu@ip-172-31-81-131:~$ mysql --version
mysql Ver 8.0.33-0ubuntu0.20.04.4 for Linux on x86_64 ((Ubuntu))
ubuntu@ip-172-31-81-131:~$
```

## Establecer comunicaci3n

### mysql -u root -p

```
ubuntu@ip-172-31-81-131:~$ mysql -u root -p
Enter password:
ERROR 2002 (HY000): Can't connect to local MySQL server through socket '/var/run/mysqld/mysqld.sock' (2)
```

El comando de inspecci3n ayuda a asignar una direcci3n IP a la instancia del servidor MySQL:

### docker inspect -f

'{{range NetworkSettings.Networks}}{{.IPAddress}}{{end}}' mysql-demo

```
ubuntu@ip-172-31-81-131:~$ mysql -h 172.17.0.4 -P 3306 --protocol=tcp -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.1.0 MySQL Community Server - GPL

Copyright (c) 2000, 2023, 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>
```

Proporcionar la direcci3n IP anterior en la opci3n de host del cliente, con el n3mero de puerto predeterminado y el tipo de protocolo como TCP:

### mysql -h 172.17.0.2 -P 3306 --protocol=tcp -u root -p

```
ubuntu@ip-172-31-81-131: ~$ mysql -h 172.17.0.4 -P 3306 --protocol=tcp -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.1.0 MySQL Community Server - GPL

Copyright (c) 2000, 2023, 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>
```

**Ingresar el siguiente comando para mostrar las base de datos**  
**show databases;**

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0.14 sec)

mysql>
```

**Crear una base de datos**

**create database automatiza;**

```
mysql> create database automatiza;
Query OK, 1 row affected (0.03 sec)

mysql>
```

**Cambiarse a la base de datos**

**use automatiza;**



Crear una tabla llamada usuarios con los siguientes campos

Campo	Tipo de datos	Indices
username	varchar(30) PK	
password	varchar(20)	
email	varchar(80)	UNIQUE
telefono	varchar(10)	UNIQUE

```
mysql> create database automatiza;
Query OK, 1 row affected (0.03 sec)

mysql> use automatiza;
Database changed
mysql> CREATE TABLE usuarios (
  ->     username VARCHAR(30) PRIMARY KEY,
  ->     password VARCHAR(20),
  ->     email VARCHAR(80) UNIQUE,
  ->     telefono VARCHAR(10) UNIQUE
  -> );
Query OK, 0 rows affected (0.05 sec)

mysql> 
```

Ejecutar comando quit para salir de cliente mysql

**quit**

Para el contenedor

**sudo docker stop mysql-demo**

```
ubuntu@ip-172-31-81-131:~$ sudo docker stop mysql-demo
mysql-demo
ubuntu@ip-172-31-81-131:~$ 
```

**Eliminar el contenedor**

**sudo docker rm mysql-demo**

**Creando un contenedor de red**

**docker network create network-mysql**

**Modificar archivo para crear contenedor y agregar comando**

**--network network-mysql \**