Practica

EJERCICIO: Montar un servidor con Express

C:\Users\PM Andrés Henao\Documents\U\Doc de procesos\API\_304\auth\_Service>npm install express

added 57 packages, and audited 58 packages in 1s

7 packages are looking for funding

run `npm fund` for details

C:\Users\PM Andrés Henao\Documents\U\Doc de procesos\API\_304\auth\_Service>npm install ts-node

added 19 packages, and audited 77 packages in 13s

7 packages are looking for funding

run `npm fund` for details

found 0 vulnerabilities

C:\Users\PM Andrés Henao\Documents\U\Doc de procesos\API\_304\auth\_Service>npm install ts-node-dev

added 43 packages, and audited 120 packages in 5s

14 packages are looking for funding

run `npm fund` for details

found 0 vulnerabilities

C:\Users\PM Andrés Henao\Documents\U\Doc de procesos\API\_304\auth\_Service>npm install -g typescript

added 1 package in 786ms

Cuando instalamos el Xamp el instala apache por defecto en puerto 80.

Protocolo HTTP:

GET

POST

PATCH

PUT

DELETE

Para crear un contenedor en Docker: primero necesito un archivo DockerFile

Donde lo primero que se especifica es el sistema que se va a utilizar, utilizando la palabra FROM

PASOS PARA CREAR UN CONTENEDOR DOCKER:

1. Imagen (mongo, Ubuntu, pyton, node, etc)
   1. Para el proyecto base de datos no SQL y con mongo
   2. Tenemos redes, volúmenes, imágenes,etc

Docker image ls

Docker volumen ls

Docker container ls

Docker ps / listar contenedores activos(running)

PARA CREAR UN CONTENEDOR USO:

1. Docker run + nombre de imagen
2. Docker -d nginx

DOCKER RUN hello-world / Valida si existe y la utiliza si no la descarga de donde yo indique

docker run -d --name contenedordos hello-world // Libera la consola

docker rmn -f contenedor contenedordos beautiful\_vaughan condescending\_cartwright gallant\_moore / eliminar contenedores

Contenedores vivos mientras existan procesos corriendo o ejecutándose dentro de ellos

docker run -it centos bash // iniciar un proceso

ejm: docker run -d centos ping 8.8.8.8

ejp 2:docker exec -it 78 bash

Para validar que dos servidores se ven utilizo ping mas ip del otro server

Para saber si un puerto este activo utilizo un telnet + ip + puerto

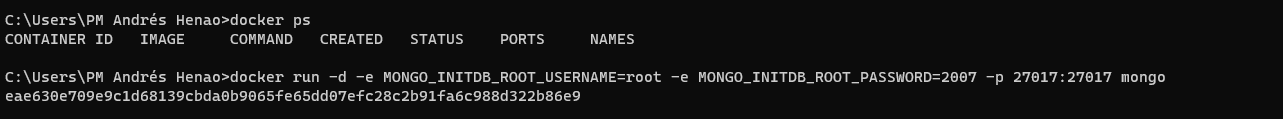
Para exponer un puerto con la bandera

docker run -d -p 27017:27017 contenedor\_test mongo

el primer puerto es el de mi maquina: y el segundo es el del Docker

Texto

Descripción generada automáticamente



No se recomienda que el usuario root sea con el que se conecten las aplicaciones

Texto

Descripción generada automáticamente

Para acceder a un contenedor

**Docker exec -ti + nombre contenedor + bash**

Docker exec -ti o -it(-t para una terminal y -i para que sea interactiva) contenedor\_test ls

Para obtener el nombre de la maquina = Uname -a

Para saber quien soy whoami

Para acceder a la base de datos:

Mongo -u Usuario

Texto

Descripción generada automáticamente

Para crear la base de datos

Show dbs me las lista

Use MongoBd

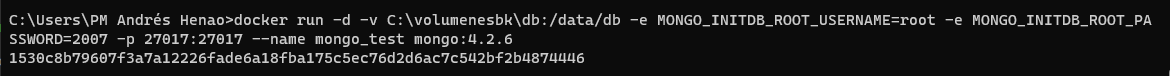
Switch to db MongoBd

db.createUser({user:"userBD",pwd:"userBD", roles:["readWrite"]})

db.getUsers() obtener usuarios

crear colecciones en studio 3d

db.getCollection**(**"usersprueba"**)**.insertOne**({**name**:** "AndresHenao"**})**



**REDES**

Todos los Docker quedan en la red bridge por defecto

Para inspeccionar una red

docker network inspect 10

Para modificar el archivo host en gnix:

vim /etc/hosts

ip +nombre

172.17.0.2 angry\_einstein

Para crear una red en Docker

docker network create red1 -d bridge

Para crear un Docker que quede atado a la red que creamos:

docker run -d --network red1 nginx

para agregar un contenedor a una red se utiliza:

docker network connect red1 contenedor

COMO CREAR NUESTRA PROPIA IMAGEN

Lo primero es crear un archivo Docker file

Luego Docker build -t imagendocker .

docker build -t otraimagen .

docker run -d -P otraimagen

Para subir una imagen a Docker:

Docker tag + nombre o id + nombredeUsuarioenDocker/nombreQueQuieroPonerAImagen

Luego hacemos un push

Docker push nombredeUsuarioenDocker/nombrequediALaImagen

DOCKER-COMPOSE

Esta compuesto por servicios