
Instalacion Docker

Primero actualizamos y agregamos los paquetes necesarios de Docker:

```
sudo apt-get update
```

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

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

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

Docker CE (CE es la versión gratuita de Docker):

```
sudo apt-get update
```

```
sudo apt-get install docker-ce
```

```
docker --version
```

```
Docker version 18.09.1, build 4c52b90
```

Instalar la ultima versión de Docker-Compose en Linux

```
sudo curl -L "https://github.com/docker/compose/releases/download/1.23.2/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
```

```
sudo chmod +x /usr/local/bin/docker-compose
```

```
docker-compose --version
```

```
docker-compose version 1.23.2, build 1110ad01
```

Instalar y configurar LaraDock

```
git clone https://github.com/Laradock/laradock.git
```

```
cd laradock
```

```
cp env-example .env
```

Editamos el archivo .env

nano .env

DOCKER_HOST_IP=127.0.0.1

Generar contenedores para Laravel

sudo docker-compose up -d nginx mysql phpmyadmin

sudo docker-compose exec workspace bash

Accedemos a nuestro espacio de trabajo

docker-compose exec workspace bash

Instalamos Laravel dentro de nuestro espacio de trabajo.

Lo hacemos de la forma tradicional, luego creamos el .env, ejecutamos: artisan key:generate

root@0bd8859e9750:/var/www# cp .env.example .env

root@0bd8859e9750:/var/www# artisan key:generate

root@0bd8859e9750:/var/www# exit

\$ cd ..

\$ cd blog/

\$ sudo chmod -R 777 storage bootstrap/cache

Configurar el contenedor Nginx para que apunte hacia nuestro proyecto

Editamos de nuevo el archivo .env de laradock y

colocamos el nombre de la carpeta del proyecto en la variable APP_CODE_PATH_HOST=nameproject

Reiniciamos los contenedores:

docker-compose down

docker-compose up -d nginx mysql

Configurar el contenedor Mysql:

Volvemos al archivo .env de Laradock y editamos MYSQL_DATABASE y MYSQL_ROOT_PASSWORD

```
MYSQL_DATABASE=chat
MYSQL_USER=root
MYSQL_PASSWORD=c0ntr0ll3r
MYSQL_PORT=3306
MYSQL_ROOT_PASSWORD=root
```

Luego vamos al archivo my.cnf de mysql de la siguiente forma:

```
nano laradock/mysql/my.cnf
```

Agregamos la última línea del archivo:

```
default_authentication_plugin=mysql_native_password
```

Ahora ingresamos al contenedor de Mysql, le damos los permisos al usuario y creamos la db:

docker-compose exec mysql bash
mysql -u root -p
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'root';
ALTER USER 'root'@'%' IDENTIFIED WITH mysql_native_password BY 'root';
ALTER USER 'default'@'%' IDENTIFIED WITH mysql_native_password BY 'secret';
create database dockerapp;

- Configurar el .env de Laravel

```
DB_CONNECTION=mysql
DB_HOST=mysql
DB_PORT=3306
DB_DATABASE=chat
DB_USERNAME=root
DB_PASSWORD=root
```

Reiniciamos los contenedores:

`docker-compose down`

`docker-compose up -d nginx mysql`

`docker-compose exec workspace bash`

`artisan migrate`