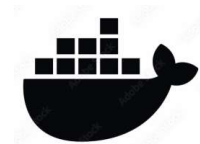


MOODLE SOBRE LAMP



ÍNDICE

Moodle	2
Despliegue de Moodle sobre LAMP	2
Preparación del entorno	2
Repositorio GitHub sincronizado con GitHub Desktop	2
Carpeta Moodle en el repositorio	3
Estructura del proyecto	3
Archivo docker-compose.yml	3
Archivo Dockerfile	4
Construcción e Inicio de Contenedores	5
Moodle en el navegador	6
TROUBLESHOOTING	8



Moodle

Moodle (CMS) es un sistema de gestión de aprendizaje, gratuito y de código abierto escrito en PHP y distribuido bajo la licencia Pública General GNU. Sirve para crear espacios de enseñanza online y administrar, distribuir y controlar todas las actividades de formación no presencial de una entidad educativa u organización.

Despliegue de Moodle sobre LAMP

En este manual, vamos a desplegar Moodle sobre una infraestructura Docker utilizando Docker Compose y un Dockerfile, integrando un servidor LAMP.

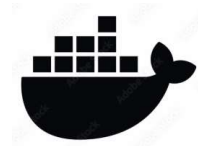
Preparación del entorno

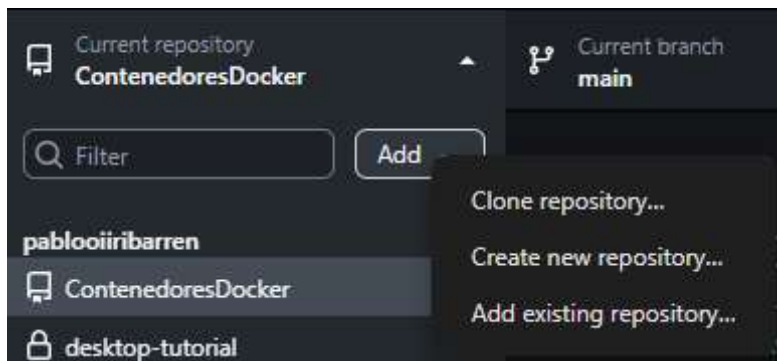
Deberemos seguir los siguientes pasos para tener el entorno listo para que funcione todo correctamente.

Como vamos a trabajar con Docker deberemos tener instalados Docker y Docker Compose.

Repositorio GitHub sincronizado con GitHub Desktop

Como vamos a guardar todos los cambios del proyecto en un repositorio de GitHub, deberemos **clonar** el repositorio que hemos creado nuevo en GitHub Desktop para comenzar a hacer cambios en VSCode. (Asegurarse que la carpeta en la que trabajamos en VSCode es la misma con la que realizaremos los 'commits' en GitHub Desktop).



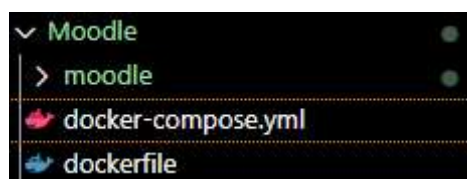


Carpeta Moodle en el repositorio

Crearemos una carpeta dentro de la propia carpeta 'Moodle' para guardar más adelante los archivos de instalación de Moodle.

Estructura del proyecto

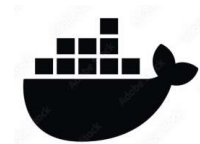
En mi caso utilizo VSCode como IDE, por lo que crearemos una nueva carpeta dentro de nuestro proyecto llamada 'Moodle'. Dentro de esta carpeta crearemos dos archivos principales que son los que van a hacer que funcione toda la infraestructura del proyecto: 'docker-compose.yml' y 'dockerfile'.



Archivo docker-compose.yml

Primero, configuraremos este archivo donde se define la infraestructura para Moodle: el servicio DB con el contenedor de MySQL y el servicio de Moodle con el contenedor de datos de Moodle.

Deberá tener una forma parecida a esta:

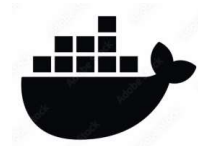


```
Moodle > docker-compose.yml
1
2  version: '3.8'
3
4  services:
5    db:
6      image: mariadb:latest
7      container_name: moodle_db
8      environment:
9        MYSQL_DATABASE: moodle
10       MYSQL_USER: moodle_user
11       MYSQL_PASSWORD: moodle_password
12       MYSQL_ROOT_PASSWORD: root_password
13     ports:
14       - "3306:3306"
15     volumes:
16       - db_data:/var/lib/mysql
17
18   moodle:
19     build:
20       context: .
21       dockerfile: Dockerfile
22     container_name: moodle_web
23     environment:
24       MOODLE_DB_HOST: db
25       MOODLE_DB_USER: moodle_user
26       MOODLE_DB_PASSWORD: moodle_password
27       MOODLE_DB_NAME: moodle
28     ports:
29       - "8080:80"
30     volumes:
31       - moodle_data:/var/www/html
32
33   volumes:
34     db_data:
35     moodle_data:
36
37   networks:
38     moodle_net:
39     driver: bridge
```

Archivo Dockerfile

Después, en el archivo Dockerfile definiremos la configuración del contenedor web. Se configura PHP y Apache para Moodle.

Deberá tener una serie de instrucciones parecidas a estas:



```

Moodle > dockerfile > ...
1
2 FROM php:8.1-apache
3
4 #Instalar exstensiones necesarias para Moodle
5 RUN apt-get update && apt-get install -y \
6     libzip-dev unzip mariadb-client \
7     && docker-php-ext-install zip mysqli pdo_mysql
8
9 #Instalar la tarea Cron de Unix que funcione Moodle correctamente
10 RUN apt-get update && apt-get install -y cron
11 CMD ["service", "cron", "start"]
12
13 #Habilitar módulos de Apache
14 RUN a2enmod rewrite
15
16 COPY ./moodle /var/www/html
17
18 #Permisos correspondientes
19 RUN chown -R www-data:www-data /var/www/html
20
21 #Exponer puerto 80
22 EXPOSE 80

```

Construcción e Inicio de Contenedores

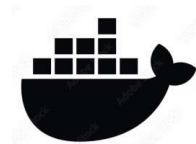
Una vez configurados los archivos de Docker, ejecutaremos el siguiente comando desde el directorio Moodle para construir los contenedores: **docker-compose build**

```

PS C:\Users\Pablo\Documents\GitHub\ContenedoresDocker\Moodle> docker-compose build
time="2024-11-30T13:31:59+01:00" level=warning msg="C:\\Users\\Pablo\\Documents\\GitHub\\ContenedoresDocker\\
\Moodle\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to av
oid potential confusion"
[+] Building 6.8s (5/10)                                docker:desktop-linux
=> => transferring dockerfile: 564B                                0.0s
=> [moodle internal] load metadata for docker.io/library/php:8.1-apache        1.6s
=> [moodle internal] load .dockerignore                                          0.0s
=> => transferring context: 2B                                                0.0s
=> CACHED [moodle 1/6] FROM docker.io/library/php:8.1-apache@sha256:e3d4b282a162873e05dfa6df3429e456 0.0s
=> => resolve docker.io/library/php:8.1-apache@sha256:e3d4b282a162873e05dfa6df3429e45605049095a5a468 0.0s
=> [moodle internal] load build context                                          2.2s
=> => transferring context: 2.77MB                                           2.1s
=> [moodle 2/6] RUN apt-get update && apt-get install -y libzip-dev unzip mariadb-client && 5.1s
=> => # Get:4 http://deb.debian.org/debian bookworm/main amd64 Packages [8789 kB]
=> => # Get:5 http://deb.debian.org/debian bookworm-updates/main amd64 Packages [2712 B]
=> => # Get:6 http://deb.debian.org/debian-security bookworm-security/main amd64 Packages [206 kB]

```

Después, levantaremos los contenedores en segundo plano: **docker-compose up -d**



```
PS C:\Users\Pablo\Documents\GitHub\ContenedoresDocker\Moodle> docker-compose up -d
time="2024-11-30T13:35:17+01:00" level=warning msg="C:\\Users\\Pablo\\Documents\\GitHub\\ContenedoresDocker\\
\\Moodle\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to av
oid potential confusion"
[+] Running 2/2
 ✓ Container moodle_db      Running      0.0s
 ✓ Container moodle_web     Started      4.2s
```

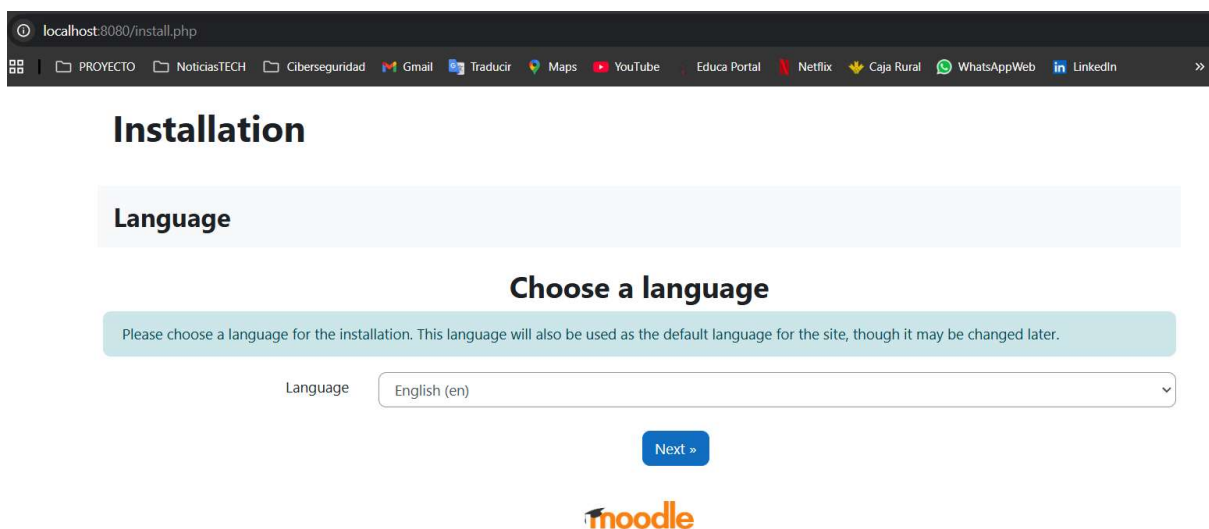
Para verificar que los contenedores están ejecutándose correctamente ejecutamos:

`docker ps`

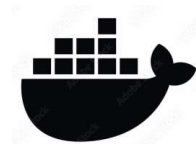
```
PS C:\Users\Pablo\Documents\GitHub\ContenedoresDocker\Moodle> docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
54d7045f3aa4   mariadb:late  "docker-entrypoint.s..." 2 hours ago   Up 2 hours   0.0.0.0:3306->3306/tcp
moodle_db
2e864f6677df   mysql        "docker-entrypoint.s..." 4 days ago   Up 5 seconds  3306/tcp, 33060/tcp, 0
.0.0.0:3308->3308/tcp
roleplayinggame-database
```

Moodle en el navegador

Accederemos a Moodle poniendo en la barra del navegador: <http://localhost:8080>



The screenshot shows a web browser window with the address bar displaying 'localhost:8080/install.php'. The browser's address bar and tabs are visible at the top. The main content area is titled 'Installation' and 'Language'. Below this, there is a section titled 'Choose a language' with a message: 'Please choose a language for the installation. This language will also be used as the default language for the site, though it may be changed later.' A dropdown menu labeled 'Language' is set to 'English (en)'. A 'Next >' button is located below the dropdown. The Moodle logo is visible at the bottom of the page.



localhost:8080/install.php

Si su portal es accesible desde internet, y desde una red interna (llamada intranet), entonces use la dirección pública aquí.

Si la dirección actual no es correcta, por favor, cambie el URL en la barra de búsqueda de su navegador y reinicie la instalación.

Directorio de Moodle
Ruta completa del directorio que contiene el código de Moodle.

Directorio de Datos
Un directorio en el que Moodle puede guardar los archivos subidos por usuarios.

En este directorio el usuario del servidor web (por lo general 'nobody', 'apache' o 'www-data') debe poder leer y escribir.

No debe poderse acceder a esta carpeta directamente a través de la web.

Si el directorio no existe, el instalador tratará de crearlo.

Dirección Web

Directorio de Moodle

Directorio de Datos

[< Anterior](#) [Siguiente >](#)



Aquí nos pide que confirmemos las rutas de configuración.

localhost:8080/install.php

Instalación

Base de datos

Seleccione el controlador de la base de datos

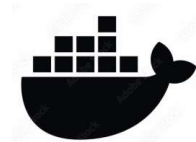
Moodle soporta varios tipos de servidores de base de datos. Por favor, póngase en contacto con el administrador del servidor si no sabe qué tipo usar.

Tipo

[< Anterior](#) [Siguiente >](#)



```
PS C:\Users\Pablo\Documents\GitHub\ContenedoresDocker\Moodle> docker exec -it moodle_web bash
root@08b7274d6d3b:/var/www/html# ls -ld /var/www/moodledata
ls: cannot access '/var/www/moodledata': No such file or directory
root@08b7274d6d3b:/var/www/html# mkdir -p /var/www/moodledata
root@08b7274d6d3b:/var/www/html# chown -R www-data:www-data /var/www/moodledata
root@08b7274d6d3b:/var/www/html# chmod -R 777 /var/www/moodledata
root@08b7274d6d3b:/var/www/html#
```



TROUBLESHOOTING

- Dar los permisos adecuados a las carpetas donde alojamos los archivos

```
PS C:\Users\Pablo\Documents\GitHub\ContenedoresDocker\Moodle> docker exec -it moodle_web bash
root@2aeade647705:/var/www/html# chmod -R www-data:www-data /var/www
chmod: invalid mode: 'www-data:www-data'
Try 'chmod --help' for more information.
root@2aeade647705:/var/www/html# chown -R www-data:www-data /var/www
root@2aeade647705:/var/www/html# chmod -R 777 /var/www
root@2aeade647705:/var/www/html#
```

Dirección Web	<input type="text" value="http://localhost:8080"/>
Directorio de Moodle	<input type="text" value="/var/www/html"/>
Directorio de Datos	<input type="text" value="/var/www/moodledata"/>

El directorio padre (/var/www) no tiene permisos de escritura. El directorio de los datos (/var/www/moodledata) no puede ser creado por el instalador.

