

# Creación de escenarios con Docker Compose

## Parte 1 - Wordpress + MariaDB

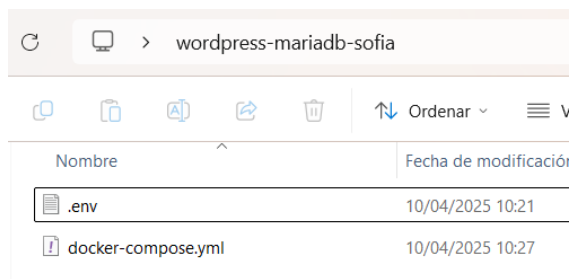
Wordpress es un CMS (Content Management System) que permite crear páginas web sin necesidad de programar. Para su despliegue, necesita apoyarse en una base de datos MySQL, como por ejemplo MariaDB:

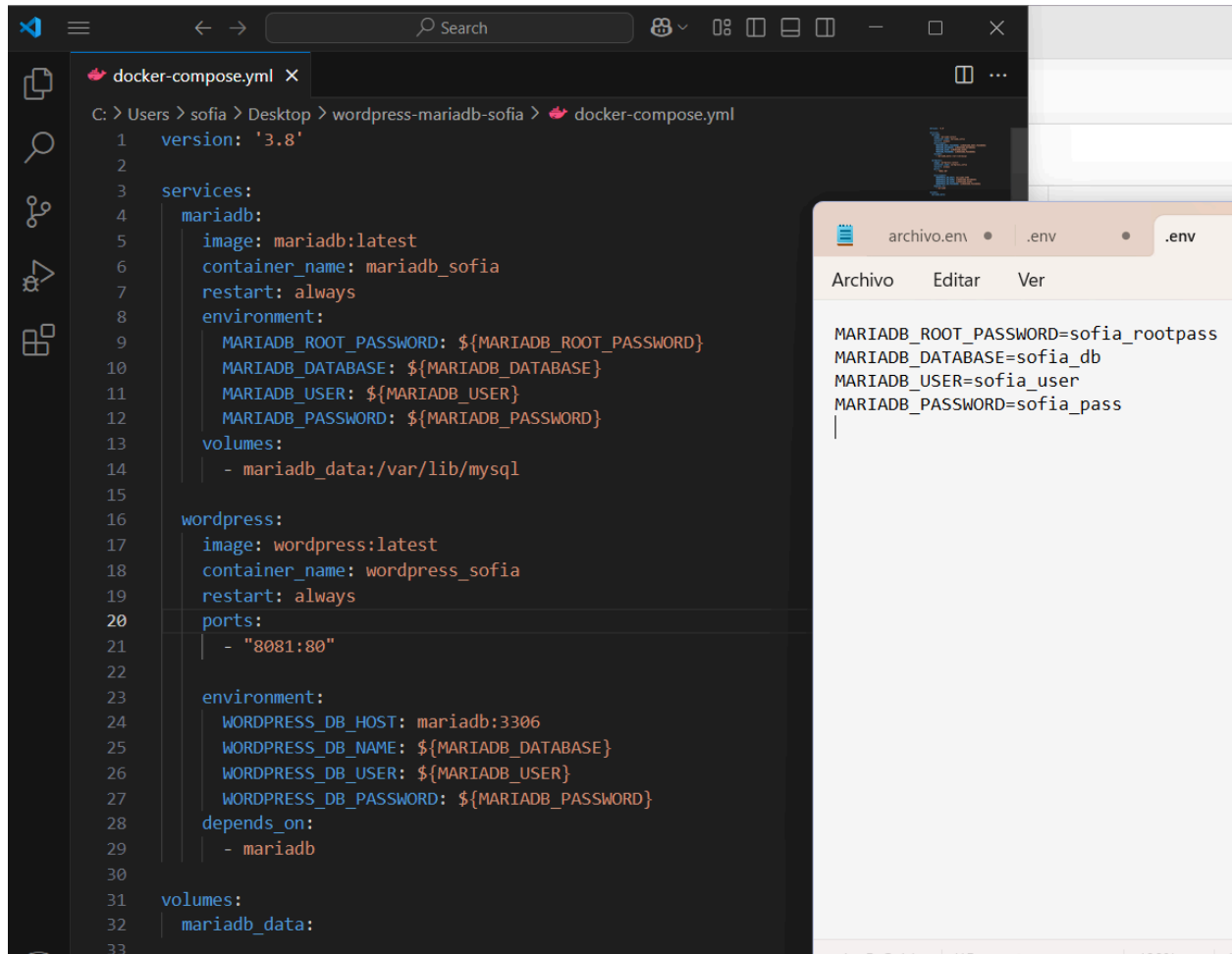
El ejercicio consiste en crear un escenario que implemente Wordpress y MariaDB.

**Pista:** Para este uso, MariaDB necesita las siguientes variables de entorno:

- **MARIADB\_ROOT\_PASSWORD**
- **MARIADB\_DATABASE**
- **MARIADB\_USER**
- **MARIADB\_PASSWORD**

1. Crear la carpeta para meter los archivos .env y .yml:





## 2. Abrir terminal y ejecutar Docker Compose:

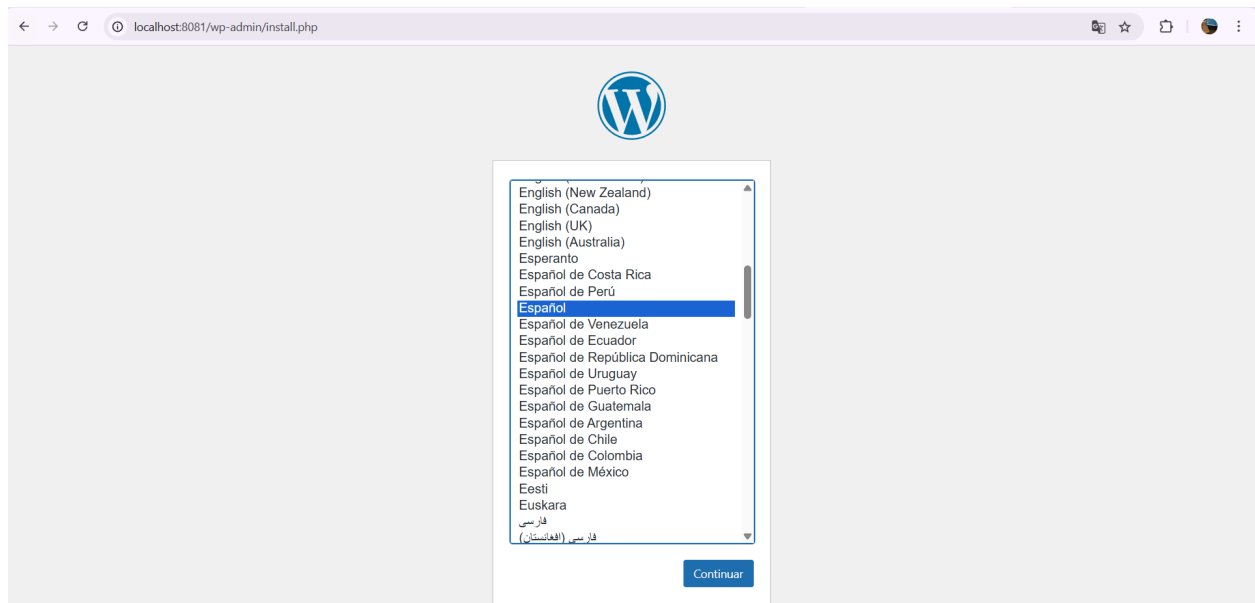
```
C:\Users\sofia>cd "C:\Users\sofia\Desktop\wordpress-mariadb-sofia"

C:\Users\sofia\Desktop\wordpress-mariadb-sofia>docker-compose up -d
time="2025-04-10T10:22:56+02:00" level=warning msg="C:\Users\sofia\Desktop\wordpress-mariadb-sofia\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 3/4
✔ Network wordpress-mariadb-sofia_default Created 0.4s
✔ Volume "wordpress-mariadb-sofia_mariadb_data" Created 0.0s
✔ Container mariadb_sofia Started 0.9s
- Container wordpress_sofia Starting 1.0s
Error response from daemon: driver failed programming external connectivity on endpoint wordpress_sofia (b4da528d74e088a280e78287d4f2c0f80721fad42688b23d67b2ea674a12f60c): Bind for 0.0.0.0:8080 failed: port is already allocated

C:\Users\sofia\Desktop\wordpress-mariadb-sofia>docker-compose down
time="2025-04-10T10:27:15+02:00" level=warning msg="C:\Users\sofia\Desktop\wordpress-mariadb-sofia\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 3/3
✔ Container wordpress_sofia Removed 0.1s
✔ Container mariadb_sofia Removed 0.8s
✔ Network wordpress-mariadb-sofia_default Removed 1.0s

C:\Users\sofia\Desktop\wordpress-mariadb-sofia>docker-compose up -d
time="2025-04-10T10:27:18+02:00" level=warning msg="C:\Users\sofia\Desktop\wordpress-mariadb-sofia\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 3/3
✔ Network wordpress-mariadb-sofia_default Created 0.4s
✔ Container mariadb_sofia Started 1.0s
✔ Container wordpress_sofia Started 1.5s
```

3. Comprobar que funciona desde el navegador:



Nota: adminer es una interfaz web para gestionar la base de datos. En esta práctica no es obligatorio incluirlo, aunque sí recomendable.

## Parte 2 - Redmine + PostgreSQL

Redmine, aunque también se catalogue como CMS en Docker Hub, está enfocado en la gestión de proyectos.

Este ejercicio consiste en realizar un despliegue similar al anterior, utilizando en este caso la aplicación web ‘Redmine’ junto con el sistema de gestión de base de datos Postgres:

Ten en cuenta lo siguiente a la hora de definir el escenario:

- Utiliza parámetros para las variables de entorno
- Asigna nombres a los contenedores
- Crea un volumen vinculado a la carpeta que indica la documentación oficial para cada contenedor.
- Utiliza la versión 15 de Postgres.

**Pista:** Para este uso, Postgres necesita las siguientes variables de entorno:

- POSTGRES\_PASSWORD
- POSTGRES\_USER
- POSTGRES\_DB

Una vez esté desplegado, puedes iniciar sesión con las siguientes credenciales:

- User: admin
- Password: admin

1. Crear la carpeta para meter los archivos .env y .yaml:

The image shows a file explorer window for the directory 'redmine-postgresql-sofia'. It contains two files: '.env' and 'docker-compose.yml', both modified on 10/04/2025 at 10:41 and 10:43 respectively.

Below the file explorer is a code editor showing the content of 'docker-compose.yml'.

```
1 version: '3.8'
2
3 services:
4   postgres:
5     image: postgres:15
6     container_name: postgres_sofia
7     restart: always
8     environment:
9       POSTGRES_PASSWORD: ${POSTGRES_PASSWORD}
10      POSTGRES_USER: ${POSTGRES_USER}
11      POSTGRES_DB: ${POSTGRES_DB}
12     volumes:
13       - postgres_data:/var/lib/postgresql/data
14
15   redmine:
16     image: redmine:latest
17     container_name: redmine_sofia
18     restart: always
19     ports:
20       - "3000:3000"
21     environment:
22       REDMINE_DB_ADAPTER: postgresql
23       REDMINE_DB_HOST: postgres:5432
24       REDMINE_DB_NAME: ${POSTGRES_DB}
25       REDMINE_DB_USER: ${POSTGRES_USER}
26       REDMINE_DB_PASSWORD: ${POSTGRES_PASSWORD}
27     depends_on:
28       - postgres
29
30 volumes:
31   postgres_data:
```

Overlaid on the code editor is a small window showing the content of the '.env' file:

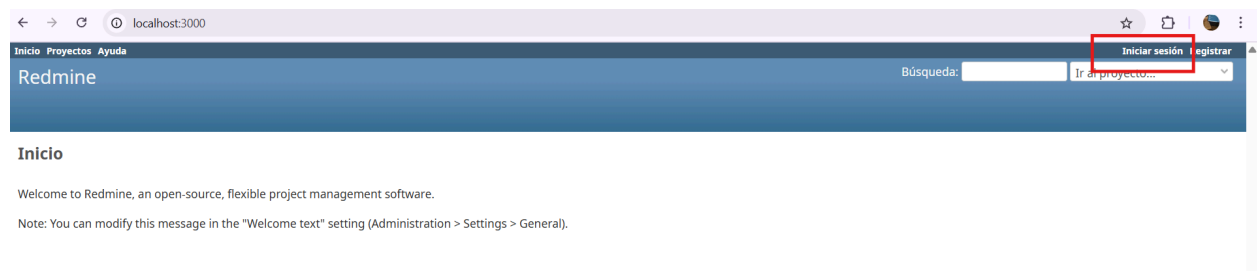
```
POSTGRES_PASSWORD=sofia_pgpass
POSTGRES_USER=sofia_pguser
POSTGRES_DB=sofia_redmine_db
```

The status bar at the bottom right of the code editor indicates 'Ln 4, Col 1' and '87 caracteres'.

## 2. Abrir terminal y ejecutar Docker Compose:

```
PS C:\Users\sofia> cd "C:\Users\sofia\Desktop\redmine-postgresql-sofia"
PS C:\Users\sofia\Desktop\redmine-postgresql-sofia> docker-compose up -d
time="2025-04-10T10:43:36+02:00" level=warning msg="The \"POSTGRES_PASSWORD\" variable is not set. Defaulting to a blank string."
time="2025-04-10T10:43:36+02:00" level=warning msg="The \"POSTGRES_USER\" variable is not set. Defaulting to a blank string."
time="2025-04-10T10:43:36+02:00" level=warning msg="The \"POSTGRES_DB\" variable is not set. Defaulting to a blank string."
time="2025-04-10T10:43:36+02:00" level=warning msg="The \"POSTGRES_DB\" variable is not set. Defaulting to a blank string."
time="2025-04-10T10:43:36+02:00" level=warning msg="The \"POSTGRES_USER\" variable is not set. Defaulting to a blank string."
time="2025-04-10T10:43:36+02:00" level=warning msg="The \"POSTGRES_PASSWORD\" variable is not set. Defaulting to a blank string."
time="2025-04-10T10:43:36+02:00" level=warning msg="C:\\Users\\sofia\\Desktop\\redmine-postgresql-sofia\\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 2/2
✔ Container redmine_sofia   Running      0.0s
✔ Container postgres_sofia Started       0.0s
```

## 3. Acceder a Redmine en el navegador: <http://localhost:3000>



## 4. Iniciar sesión:

**Identificador**

**Contraseña** [¿Olvidaste la contraseña?](#)