

GUIA PAS A PAS DE DOCKER

El primer pas consisteix en crear el fitxer **docker-compose.yml**, el qual s'encarregarà de definir i desplegar tres contenidors: un servidor **Apache amb PHP**, una base de dades **MySQL** i una interfície d'administració mitjançant **Adminer**.

L'objectiu d'aquesta infraestructura és implementar una aplicació web per al registre i gestió d'incidències.

Per començar, es defineixen els serveis dins de l'apartat **services**: del fitxer **docker-compose.yml**, on cada servei representa un contenidor.

El primer servei, anomenat **web**, correspon al contenidor que allotja **Apache i PHP**. Aquest es construeix mitjançant la directiva **build**, la qual inclou el **context** —que indica la ubicació del directori arrel del projecte— i el nom del fitxer **Dockerfile** que s'utilitzarà per generar la imatge del contenidor.

A continuació, es defineixen els **ports** a través de l'apartat **ports**, on el valor a l'esquerra indica el port del sistema host que s'utilitzarà per accedir al servei mitjançant el navegador, i el valor a la dreta especifica el port exposat pel servidor **Apache** dins del contenidor.

L'apartat **volumes** permet mapar el directori **/src** del projecte local al directori per defecte que el servidor **Apache** utilitza per servir els fitxers web, assegurant així que la pàgina web s'executi correctament amb el contingut desitjat.

Finalment, es fa servir la directiva **depends_on** per establir la dependència del servei web respecte al servei **db**, que és el contenidor de **MySQL**, garantint que aquest últim estigui disponible abans d'iniciar el servidor web.

```
services:
  web:
    #La imatge php:apache no conté els drivers de mysql, hem de fer una imatge a mida
    #image: php:apache
    build:
      context: ./images
      dockerfile: Dockerfile_php
    ports:
      - "8080:80"
    volumes:
      - ./src:/var/www/html
    depends_on:
      - db
```

Una vegada fet això, el següent pas serà crear el contenidor de **MySQL**, anomenat **db**.

Es defineix la directiva **image:**, on es especifica la imatge que es vol descarregar. En aquest cas, utilitzarem **mysql:9.3**.

A continuació, es troba l'apartat **environment:**, on es defineixen els paràmetres principals per configurar la base de dades (no es definirà el nom de la base de dades, ja que es crearà mitjançant l'**script** que es troba més endavant, a l'apartat de **volumes**). El primer valor que es defineix és **MYSQL_ROOT_PASSWORD**, el qual no modificarem ja que no el farem servir. Després es defineix **MYSQL_USER**, que serà l'usuari amb el qual accedirem a la base de dades. Finalment, es defineix **MYSQL_PASSWORD**, que és la contrasenya de l'usuari creat anteriorment. Dins de l'apartat **environment** també es troba **LANG**, que no es modifica.

Per últim, en l'apartat **volumes** de **db**, es defineixen dues línies. La primera, **./db_data:/var/lib/mysql**, on la ruta de la dreta és la ruta per defecte dins del contenidor, i la de l'esquerra correspon a la ruta en el nostre projecte, la qual s'utilitza per emmagatzemar totes les dades de la base de dades. La segona línia, **./db_init:/docker-entrypoint-initdb.d**, fa referència a la ruta on es troba l'script per crear la base de dades.

```
db:
# Tota la informació d'aquesta imatge està https://dockerhub.com/ /mysql
image: mysql:9.3
environment:
  MYSQL_ROOT_PASSWORD: passwordDeRootQueNoShaDeFerServirMai
  # És millor no crear la BBDD aquí, ja que no pots controlar la codificació
  # de caràcters i aleshores donarà problemes amb accents i caràcters especials
  # La BBDD es crearà a l'inici del contenidor amb els script d'inicialització
  # MYSQL_DATABASE: persones
  MYSQL_USER: usuari
  MYSQL_PASSWORD: paraula_de_pas
  LANG: C.UTF-8
# El mysql no s'exposa a l'exterior
# L'aplicació web hi accedirà per la xarxa interna de docker
# ports:
#   - "3306:3306"

# La carpeta de mysql ha d'estar al .gitignore per no pujar-la al repositori
volumes:
  - ./db_data:/var/lib/mysql
  - ./db_init:/docker-entrypoint-initdb.d
```

Una vegada fet el **MYSQL**, passarem al **Adminer**.

Per començar, definirem el contenidor **adminer**, com es va fer anteriorment. Un cop fet això, afegirem l'apartat **image** amb el paquet adminer. Seguidament, definirem l'apartat **ports**:, on per accedir-hi de manera local haurem d'utilitzar el port que assignem, que serà el port de l'esquerra, ja que el de la dreta és el port per defecte que utilitza **adminer**. Per últim, afegirem el paràmetre **depends_on**: i definirem el contenidor de **MYSQL**, que es **db**.

```
# Aquesta imatge és un client web per gestionar el mysql via web
# està disponible a http://localhost:8081
# i les credencials són les del mysql (per que "simplement" es connecta al mysql)
adminer:
  image: adminer
  ports:
    - "8081:8080"
  depends_on:
    - db
```

Després d'això, el fitxer **docker-compose.yml** hauria de crear l'estructura de directoris, on tindrem la carpeta **Apache/PHP**, que en el nostre cas és **./images**. A continuació, tenim les dues carpetes per a **MYSQL**. La primera és la carpeta on es guarden les dades de la base de dades, que ha d'estar buida; en el nostre cas és **./db_data**. La segona carpeta conté l'**script** per crear la base de dades, que en el nostre cas és **./db_init**.

A continuació, tenim el contenidor d'**adminer**, però aquí no cal crear cap carpeta, ja que només s'utilitzarà per veure i modificar les taules de la base de dades. En el nostre cas, utilitzarem el **port :8081**. Finalment, per accedir a la nostra pàgina web de manera local, utilitzarem el **port :8080**, que es el que hem definit en el contenidor.