

Pràctica 4: Aplicació d'accés a bases de dades utilitzant multiple contenidors Docker

1.- Objectius de l'activitat

Desplegament de dues aplicacions a partir de:

- Una màquina virtual creada amb **Vagrant**.
- Els codis **html** i **php** de l'aplicació.
- Contenidors **docker** amb els serveis i l'aplicació creats a partir de fitxers **Dockerfile** i **docker-compose.yml**.

L'activitat requereix:

- Entendre les especificacions de l'activitat
- Realització de les tasques demanades a l'activitat per desplegar una aplicació web amb contenidors **Docker**.
- Comprovació del correcte funcionament de l'aplicació.

Dins d'aquesta pràctica treballarem temes com:

- La creació de **volums** i de **carpetes compartides**:
 - Per compartir dades entre la màquina física i els contenidors
 - Permetre la permanència i transportabilitat de les dades dins de la base de dades.
 - Permetre poder fer canvis fàcilment de l'aplicació sense haver de crear i executar noves imatges.
- La creació de **xarxes** de contenidors
- El desplegament d'una aplicació que utilitza múltiples contenidors (en aquest cas 2) amb **docker compose**.
- La coordinació de múltiples contenidors per fer anar l'aplicació.
- La creació de fitxers **.yml** (o **.yaml**) per definir volums, contenidors, xarxes, serveis proporcionats pels contenidors, etc... que necessita l'eina **docker compose**.

2.- Creació de la màquina virtual

a) Crea una carpeta de nom **sm8act04** dins de la teva màquina física. Dins de la carpeta **sm8act04** crea una altra carpeta de nom **projectes** i un altre de nom **vm**.

b) Dins de la carpeta **vm**, copia el fitxer **Vagrantfile** que vas crear per fer l'activitat **sm8act03** i fes els següents canvis:

- El nom de la màquina serà **sm8act04_xxyyzz**. Recorda que **xxyyzz** són les 2 primeres lletres dels teus nom i cognoms.
- El nom de sistema de la màquina serà: **sm8act04-xxyyzz.fjeclot.local**.
- El port **80/tcp** de la màquina virtual s'exportarà al port **8000/tcp** de la màquina física.
- El port **443/tcp** de la màquina virtual s'exportarà al port **18443/tcp** de la màquina física.
- El port **3306/tcp** de la màquina virtual s'exportarà al port **33306/tcp** de la màquina física. Això significa que s'hauran de definir un tercer port virtual **PORT_VIR3**, un tercer port físic **PORT_FIS3**, i utilitzar-los dins d'una nova instrucció **config.vm.network**.

3. - Desplegament de l'aplicació amb Apache + MySQL

3.1- Fase de preparació del desplegament

a) Per desplegar l'aplicació utilitzarem:

- Un contenidor Docker amb:
 - Apache + PHP 8.3
 - L'extensió MySQLi de PHP.
- Un contenidor Docker amb un servidor MySQL amb usuaris i contrasenyes.
- Una xarxa de contenidors dockers per permetre la comunicació els 2 contenidors entre els dos contenidors, i la comunicació entre els contenidors i la màquina host.
- Volums per poder:
 - Compartir l'aplicació escrita en PHP i HTML que es trobarà a la carpeta **projectes** → **usuaris** → **app** de màquina host amb la carpeta **/var/www/html** del contenidor. Modificant l'aplicació en la màquina host, és modifica en el contenidor.
 - Compartir i fer accessible a través de la carpeta **/sql** del contenidor que executa el servidor MySQL un script SQL que es trobarà a la carpeta **projectes** → **usuaris** → **sql** de màquina host.
 - Compartir la base de dades entre la màquina host i el contenidor a través d'un volum permanent i transportable en la màquina host associat a la carpeta **/var/lib/mysql** del contenidor amb el servei **MySQL**.

b) Per poder fer el desplegament indicat a l'apartat anterior, primer de tot, accedeix a la màquina virtual i crea una carpeta de nom **projectes**, i a continuació una altra carpeta dins de **projectes** de nom **usuaris**.

c) Dins de la màquina virtual, accedeix a **usuaris** i crea:

- Una carpeta de nom **app**.
- Una carpeta de nom **sql**.
- Un fitxer de text de nom **Dockerfile**.
- Un fitxer de text de nom **docker-compose.yml**.

d) A continuació accedeix a **projectes** → **usuaris** → **app** i descarrega els fitxers amb els codis font de l'aplicació web:

- **app.css** → <https://raw.githubusercontent.com/dcolla2/sm8act04/refs/heads/main/app/app.css>
- **app.php** → <https://raw.githubusercontent.com/dcolla2/sm8act04/refs/heads/main/app/app.php>
- **config.php** → <https://raw.githubusercontent.com/dcolla2/sm8act04/refs/heads/main/app/config.php>
- **index.html** → <https://raw.githubusercontent.com/dcolla2/sm8act04/refs/heads/main/app/index.html>
- **library.php** → <https://raw.githubusercontent.com/dcolla2/sm8act04/refs/heads/main/app/library.php>

e) Dins de la màquina virtual, accedeix ara a la carpeta **projectes** → **usuaris** → **sql** i descarrega el següent script **.sql**:

- **bd.sql** → <https://raw.githubusercontent.com/dcolla2/sm8act04/refs/heads/main/sql/bd.sql>

f) Dins de la màquina virtual, accedeix ara a la carpeta **projectes** → **usuaris** i modifica el fitxer **docker-compose.yml**. Aquest fitxer ha d'utilitzar-se per fer les següents tasques:

- Posar en marxa un servei de nom **sMysql** que:
 - Utilitzi la imatge de contenidor oficial de **MySQL** que té el nom **mysql** i la versió **9.1**.
 - Crei a partir de la imatge descarregada un contenidor de nom **cMysql**.
 - Exporti el port **3306** del contenidor al port **33306** de la màquina virtual.
 - Utilitzant l'opció dels volums per compartir carpetes, comparteixi la carpeta ja existent **sql** de la màquina virtual amb un nou directori de nom **/sql** dins del contenidor. Recordeu que dins de la màquina virtual la carpeta **sql** està en el mateix directori que **docker-compose.yml**. La carpeta **/sql** de contenidor la crearà automàticament el programa **docker compose**.
 - Crei automàticament un volum dins de la màquina virtual de nom **vDades** que estarà associat a la carpeta **/var/lib/mysql** del contenidor.
 - Utilitza les següents variables d'entorn (environment):
 - **MYSQL_DATABASE** amb el nom de la base de dades de nom **'bdEmpresa'** (les cometes són importants) que utilitzarà l'aplicació i que es crearà amb el script **bd.sql**.
 - **MYSQL_USER** que valdrà **'Gestor'**. Això crearà un usuari de nom **'Gestor'** dins del servidor **MySQL** del contenidor amb tots els permisos necessaris.
 - **MYSQL_PASSWORD** que valdrà **'FjeClot2526#'** (aquest serà el password de **daw2**).
 - **MYSQL_ROOT_PASSWORD** que valdrà **'FjeClot2526@'**
 - Forma part d'una xarxa identificada dins de **docker-compose.yml** amb el nom **xEmpresa**.
- Posa en marxa un servei de nom **sWebApp** que:
 - Generi una imatge personalitzada amb **Apache + PHP + Extensió MySQLi** a partir d'un fitxer **Dockerfile**. El nom de la imatge serà **iwebapp** i la versió **1.0**.
 - Crei a partir de la imatge generada un contenidor de nom **cWebApp** amb **Apache + PHP + Extensió MySQLi**.
 - Que exporti el port **80** del contenidor al port **80** de la màquina virtual.
 - Que utilitzant volums comparteixi la carpeta **app** de la màquina virtual amb el directori **/var/www/html** del contenidor.
 - Depèn del servei **sMysql**. Si el servei **sMysql** no s'inicia tampoc s'inicia **sWebApp**.
 - Forma part d'una xarxa identificada dins de **docker-compose.yml** amb el nom **xEmpresa**.
- Ha de generar una xarxa identificada dins de **docker-compose.yml** amb el nom **xEmpresa** que:
 - dins de la màquina virtual tindrà el nom **Empresa**.
 - utilitzarà un driver de tipus **bridge**.
- Ha de generar un volum identificat dins de **docker-compose.yml** amb el nom **xvDades** que dins de la màquina virtual s'identificarà amb el nom **dades**.

Així doncs, el contingut de **docker-compose.yml** serà aquest:

```
services:
  sMysql:
    image: mysql:9.1
    container_name: cMysql
    ports:
      - "3306:3306"
    volumes:
      - ./sql:/sql # Això és una carpeta compartida entre host i contenidor
      - vDadesEmpresa:/var/lib/mysql # Això és un volum independent i transferible
    environment:
      MYSQL_DATABASE: 'bdEmpresa'
      MYSQL_USER: 'Gestor'
      MYSQL_PASSWORD: 'FjeClot2526#'
      MYSQL_ROOT_PASSWORD: 'FjeClot2526@'
    networks:
      - xEmpresa

  sWebApp:
    image: iwebapp:1.0
    build: .
    container_name: cWebApp
    ports:
      - "80:80"
    volumes:
      - ./app:/var/www/html # Una altra carpeta compartida entre host i contenidor
    depends_on:
      - sMysql
    networks:
      - xEmpresa

networks:
  xEmpresa:
    name: Empresa
    driver: bridge

volumes:
  vDadesEmpresa:
    name: DadesEmpresa
```

IMPORTANT:

- Els fitxers **.yml** utilitzen espais per crear sagnies (no es poden utilitzar tabuladors).
- Totes les sagnies han de tenir la mateixa quantitat d'espais.

g) El fitxer **docker-compose.yml** de l'apartat anterior necessita crear una imatge personalitzada **iwebapp:1.0**, i per tant, ens cal definir un fitxer Dockerfile amb la informació per crear-la. En el nostre cas, el contingut necessari per crear la imatge de contenidor que ens cal necessita:

- **Apache + PHP8.3.**
- La biblioteca amb l'**extensió MySQLi** de **PHP**.

De la manera que treballarem:

- No caldrà indicar el directori de treball del contenidor i tampoc ni tampoc copiar l'aplicació dins de la imatge perquè treballarem amb **volums** i fitxers **docker-compose.yml**, i això ens permet ser més flexibles a l'hora de crear imatges.
- Tampoc ens caldrà indicar els ports exposats i publicats perquè també es poden definir dins del fitxer **docker-compose.yml**

Així doncs, el **Dockerfile** només cal que tingui dues línies:

```
FROM php:8.3-apache
RUN docker-php-ext-install mysqli
```

La primera línia serveix per afegir el servidor Apache, el llenguatge PHP, i el mòdul de PHP per Apache. La segona línia afegeix l'extensió MySQLi de PHP que permet accedir a bases de dades i taules gestionades amb el servidor MySQL utilitzant un programa escrit en PHP.

- h) A continuació, modifica el fitxer **bd.sql** de manera que quan s'executi:
- Utilitzi la base de dades definida dins de **docker-compose.yml**.
 - La taula amb les dades de l'aplicació tindrà el nom **Treballadors**.
 - La clau primària serà la columna **codi**.
- i) Finalment, dins del directori **app**, s'haurà de modificar el fitxer **config.php** tenint en compte que:
- **\$host** ha de ser igual al nom del servei que proporciona MySQL definit dins de **docker-compose.yml**.
 - **\$user** és l'usuari de la base de dades l'indicat a **docker-compose.yml**.
 - **\$passwd** és la contrasenya indicada a **docker-compose.yml**.
 - **\$bd** és la base de dades indicada a **bd.sql** i també a **docker-compose.yml**.
 - **\$taula** és la taula indicada a **bd.sql**.

3.2 – Fase de desplegament

a) Ara ja podem desplegar l'aplicació. Dins de la màquina virtual, accedeix a la carpeta **usuaris** i desplega l'aplicació amb l'ordre **docker compose** executant:

docker compose up -d

b) Comprova que s'han creat i funcionen les imatges, els contenidors, les xarxes i els volums executant:

docker images → Comprova les imatges creades

docker compose ps -a → Comprova els contenidors creats i el seu estat de funcionament (Status → Up)

docker volume ls → Comprova els volums creats (però no les carpetes compartides) → Comprova que s'ha creat el volum amb permanència i transferible de nom **DadesEmpresa**.

docker networks ls → Comprova les xarxes creades. Comprova que s'ha creat la xarxa **Empresa**.

c) Accedeix a l'interior del contenidor amb el servidor MySQL i afegeix la taula **Treballadors** (que encara no existeix) i les seves dades a la taula **bdEmpresa** executant el script **bd.sql**. Realitza els següents passos:

- Accedeix al contenidor **cMySQL** executant:
docker exec -it cMysql bash
- Dins del contenidor accedeix a la carpeta **/sql** executant:
cd /sql
- Executa el script **bd.sql** que ara també es troba dins **/sql**:
mysql -u Gestor -p -e 'source ./bd.sql'
- Escribeix la contrasenya de l'usuari **Gestor** indicada a **docker-compose.yml**.
- Comprova que la taula **usuaris** i les seves dades s'han creat. Executa:
mysql -u Gestor -p -e 'use bdEmpresa;select * from Treballadors;'
- Surts del contenidor. Executa:
exit

3.2 – Fase de comprovacions

a) Accedeix a l'aplicació amb el navegador des de la màquina física a partir de l'adreça **IP** de la màquina física i del port **8000/tcp** creat a la **màquina física** i que és redirecciona cap al port **80/tcp** dins de la **màquina virtual**. Com que el port **80/tcp** de la màquina virtual és redirecciona al port **80/tcp** del contenidor **cWebApp**, el resultat és que ens estarem connectant a l'aplicació dins del contenidor amb l'aplicació. Comprova que l'aplicació mostra tots els registres de la taula **Treballadors** de la base de dades **bdEmpresa**.

b) Atura l'aplicació i comprova que deixa de funcionar (Status → Exited). Executa:

docker compose stop

c) Inicia l'aplicació i comprova que l'aplicació funciona (Status → Exited). Executa:

docker compose start

Lliurament de l'activitat

a) Comprovacions:

- Inicia l'aplicació i mostra que:
 - Els contenidors es posen en marxa
 - L'aplicació funciona i mostra el contingut de la base de dades.
- Atura l'aplicació i mostra que:
 - Els contenidors estant aturats (Exited).
 - L'aplicació no funciona.
- Crea un fitxer de nom **afegeix.sql** dins del directori **sql** de la màquina virtual que permeti afegir un nou treballador a la taula **Treballadors** i amb l'aplicació funcionant torna a comprovar que es mostra el nou contingut

b) Data de lliurament: Fins el **dimecres 28-1-26** a les **19.10h** per obtenir el **100%** de la nota (es pot aprofitar per fer el lliurament també el dia **26-1-26** fins a les **20.05h**).