

# Instal·lació d'un servidor Apache en un contenidor



Nom i Cognom: Carles Molina Espinós

Curs: 2CFGS DAW

# **Index**

Objectius.....	3
Enunciat.....	3
Ús bàsic de Docker.....	3
1. Descàrrega de la imatge oficial d'Ubuntu.....	3
2. Creació i execució d'un contenidor interactiu.....	3
3. Instal·lació de paquets dins del contenidor.....	3
3.2. Instal·lació nano.....	4
3.3. Reanomena el fitxer index.html que es troba a /var/www/html.....	4
4. Modificació de l'index.html.....	4
Creació d'imatges personalitzades a partir d'un contenidor.....	4
1. Creació d'una imatge a partir del contenidor.....	4
2. Creació d'un nou contenidor basat en la imatge personalitzada.....	4
3. Comprovació.....	5
Creació d'una imatge personalitzada amb Dockerfile.....	5
1. Creació del fitxer DockerFile.....	5
2. Construcció de la imatge.....	5
3. Execució del contenidor.....	5
Mapeig de volums.....	6
1. Creació d'un directori local.....	6
2. Execució d'un contenidor amb volum mapejat.....	6
Conclusions.....	6

# Objectius

- Generar aplicacions dintre de contenidors
- Definir fitxers de configuració de les imatges.
- Publicar ports per fer accessibles els serveis del contenidor

## Enunciat

### Ús bàsic de Docker

#### 1. Descàrrega de la imatge oficial d'Ubuntu

Executa la següent ordre per descarregar la imatge oficial d'Ubuntu:

```
docker pull ubuntu
```

#### 2. Creació i execució d'un contenidor interactiu

Aquesta ordre crea i inicia un contenidor anomenat `ubuntu_apache` en mode interactiu amb `bash` com a ordre per defecte.

```
docker run -it --name ubuntu_apache ubuntu /bin/bash
```

- `docker run`: diu a Docker que vols executar un nou contenidor.
- `-i`: manté la connexió interactiva (entrada de teclat).
- `-t`: t'obri una terminal simulada (TTY), com si estigueres treballant en una màquina real.
- `--name ubuntu_apache`: li poses el nom `ubuntu_apache` al contenidor, per a poder reconèixer-lo més fàcilment.
- `ubuntu`: és la imatge base del sistema operatiu que uses dins del contenidor, en este cas Ubuntu.
- `/bin/bash`: és el programa que s'executa dins del contenidor: una shell Bash, per a escriure comandes dins del sistema Ubuntu.

#### 3. Instal·lació de paquets dins del contenidor

Un cop dins del contenidor, executa les següents ordres:

##### 3.1. Instal·lació Apache

```
apt-get install -y apache2
```

### 3.2. Instal·lació nano

```
apt-get install nano
```

### 3.3. Reanomena el fitxer index.html que es troba a /var/www/html

```
mv /var/www/html/index.html /var/www/html/index2.html
```

Reanomenar el fitxer index.html per un altre nom a index2.html

### 4. Modificació de l'index.html

```
nano index.html /var/www/html
```

Aquestes ordres renomenen el fitxer index.html existent i creen un nou fitxer amb el missatge desitjat.

## Creació d'imatges personalitzades a partir d'un contenidor

### 1. Creació d'una imatge a partir del contenidor

```
docker commit ubuntu_apache alumne/ubuntu_apache
```

Aquesta ordre crea una nova imatge anomenada alumne/ubuntu\_apache a partir del contenidor en execu

### 2. Creació d'un nou contenidor basat en la imatge personalitzada

Aquesta ordre crea un contenidor anomenat ubuntu\_apache\_prova i publica el port 80 del contenidor al port local 4000.

```
docker run -d -p 4000:80 --name ubuntu_apache_prova alumne/ubuntu_apache
```

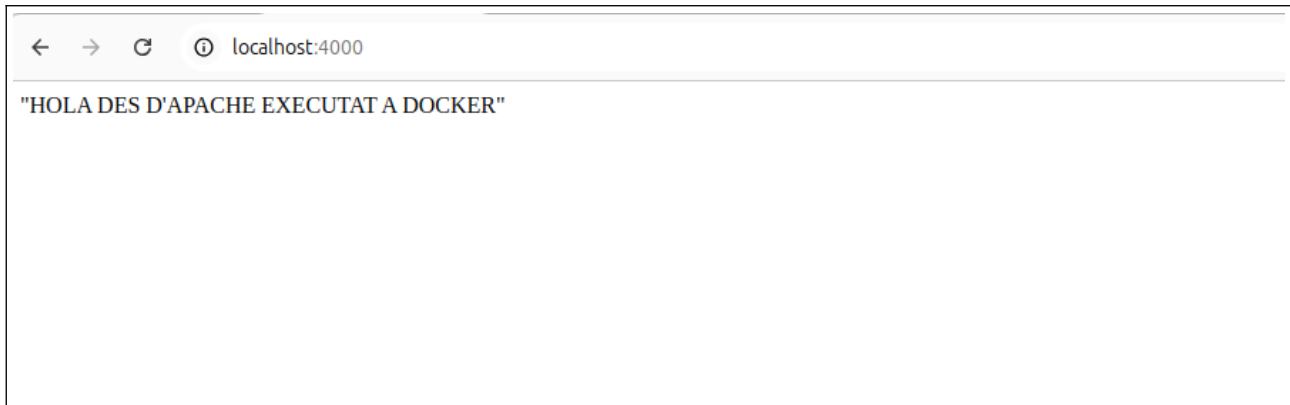
#### Desglossament:

- -p 4000:80: mapeja el port **80** del contenidor (on Apache sol escoltar) al port **4000** del teu ordinador.
- Així, pots visitar <http://localhost:4000> per veure el servidor Apache del contenidor.
- --name ubuntu\_apache\_prova: li poses nom al contenidor, en este cas ubuntu\_apache\_prova.
- alumne/ubuntu\_apache: és la **imatge Docker** que vas a usar. En este cas, pareix que és una imatge personalitzada feta per un usuari o alumne, i que segurament **ja té Apache instal·lat**.

### 3.Comprovació

http://localhost:4000

Si tot està bé, veuràs la pàgina d'inici del servidor Apache, o el que haja preparat la imatge a `alumne/ubuntu_apache`.



## Creació d'una imatge personalitzada amb Dockerfile

### 1. Creació del fitxer DockerFile

Crea un fitxer Dockerfile amb el següent contingut:

```
FROM ubuntu

# Instal·la Apache i nano
RUN apt update && apt install -y apache2 nano

# Escriu un fitxer HTML bàsic
RUN echo "HOLA DES D'APACHE EXECUTAT A DOCKER" > /var/www/html/index.html

# Inicia Apache en primer pla
CMD ["apachectl", "-D", "FOREGROUND"]

# Exposa el port 80 per a HTTP
EXPOSE 80
```

### 2. Construcció de la imatge

```
sudo docker build -t alumne/ubuntu_apache_dockerfile /home/carmol/Documentos/DAW/docker/
```

Aquesta ordre construeix la imatge amb el nom `alumne/ubuntu_apache_dockerfile`.

### 3. Execució del contenidor

```
sudo docker run -d -p 4000:80 --name ubuntu_apache_docker alumne/ubuntu_apache_dockerfile
```

# Mapeig de volums

## 1. Creació d'un directori local

```
echo "HOLA DES D'APACHE EXECUTAT AMB MAPEIG DE VOLUMS" > ~/web/index.html
```

## 2. Execució d'un contenidor amb volum mapejat

```
docker run -d -p 4000:80 --name ubuntu_apache_mapping -v ~/web:/var/www/html alumne/ubuntu_apache
```

Aquesta ordre mapeja el directori ~/web en DocumentRoot d'Apache.



## Conclusions

Amb aquesta pràctica s'ha pogut comprovar la potència i flexibilitat que ofereix Docker per a la creació i gestió de contenidors. Hem après a:

- Utilitzar Docker per desplegar serveis de manera aïllada i repetible.
- Crear imatges personalitzades a partir de contenidors existents amb docker commit.
- Automatitzar la creació d'imatges mitjançant fitxers Dockerfile.
- Fer accessibles serveis del contenidor mitjançant el mapeig de ports.
- Treballar amb volums per mantenir persistència i flexibilitat en el contingut servit per Apache.