

Ejercicio 2 - Docker Compose

Explorar la imagen de la aplicación FileBrowser

Captura de pantalla y documento donde se vea el fichero dockercompose yaml que has creado.

Captura de pantalla donde se vean los volúmenes/carpetas donde se han almacenado los datos

Captura de pantalla donde se vea la aplicación funcionando, sube algún fichero, cambia el lenguaje a español...

Explicar brevemente cómo funciona esta aplicación y qué hace.

Ejercicio 2 - Docker Compose

Explorar la imagen de la aplicación FileBrowser

```
docker run -d --name filebrowser -p 80:8080 hurlenko/filebrowser
```

To run as current user and to map custom volume locations use:

```
docker run -d \
  --name filebrowser \
  --user $(id -u):$(id -g) \
  -p 8080:8080 \
  -v /DATA_DIR:/data \
  -v /CONFIG_DIR:/config \
  -e FB_BASEURL=/filebrowser \
  hurlenko/filebrowser
```

docker-compose

Minimal `docker-compose.yml` may look like this:


```
version: "3"

services:
  filebrowser:
    image: hurlenko/filebrowser
    user: "${UID}:${GID}"
    ports:
      - 443:8080
    volumes:
      - /DATA_DIR:/data
      - /CONFIG_DIR:/config
    environment:
      - FB_BASEURL=/filebrowser
    restart: always
```

Ports description

- `8080` - default filebrowser port

Supported environment variables

The environment variables are prefixed by `FB_` followed by the option name in caps. So to set "database" via an env variable, you should set `FB_DATABASE`. The list of available options can be [found here](#) .

Supported volumes

- `/data` - Data directory to browse
- `/config` - `filebrowser.db` location

Attaching multiple directories

Attaching multiple directories

If you want to attach multiple directories you need to mount them as subdirectories of the data directory inside of the container (`/data` by default):

```
docker run \  
  -v /path/to/music:/data/music \  
  -v /path/to/movies:/data/movies \  
  -v /path/to/photos:/data/photos \  
  hurlenko/filebrowser
```

Building

```
git clone https://github.com/hurlenko/filebrowser-docker  
cd filebrowser-docker  
docker build -t hurlenko/filebrowser .
```

Esto son los datos de los que vamos a sacar la información para realizar

De aquí debemos sacar la información:

- nombre de la imagen: hurlenko/filebrowser
- puerto : 8080
- volúmenes: /data y /config

Captura de pantalla y documento donde se vea el fichero dockercompose yaml que has creado.

```
D: > USER > Desktop > Estudios 24-25 > DAW > Seg
1  version: "3"
2
3  ▶Run All Services
4  services:
5      ▶Run Service
6      filebrowser:
7          image: hurlenko/filebrowser
8          container_name: filebrowser
9          ports:
10             - "8080:8080"
11          volumes:
12             - ./data:/data
13             - ./config:/config
14          restart: unless-stopped
```

A continuación creamos en un editor de texto el archivo .yaml con el que utilizaremos para lanzar el contenedor

Descripción:

- versión - Indica la versión del esquema Docker Compose
- filebrowser - Nombre del servicio
- image - es la imagen de Docker que usamos
- container_name - nombre que le damos al contenedor
- ports- indicas que puerto del contenedor se debe conectar a que puerto de la máquina
- volumes - Creas el vínculo entre carpetas locales y rutas dentro del contenedor con esto estamos realizando **bind-mount**
- restart: unless-stopped - reinicia este contenedor automáticamente si se apaga, a no ser que lo apague yo

A continuación en Git-Bash escribimos el comando `docker compose up -d` para lanzar el contenedor por comandos

```
docker compose up -d
[+] Running 1/4
✔ filebrowser Pulled                                1.2s
✔ ibed60e4e729 Pull complete                        0.5s
✔ if3e4090e29 Pull complete                        0.7s
✔ 7b2752081f0b Pull complete                        1.3s
[+] Running 2/2
✔ Network ejercicio2_default Created                0.0s
✔ Container filebrowser Started                     0.6s
[+] Running 0/0
```

Captura de pantalla donde se vean los volúmenes/carpetas donde se han almacenado los datos

Comprobamos que realmente se ven las carpetas donde se almacenan los datos

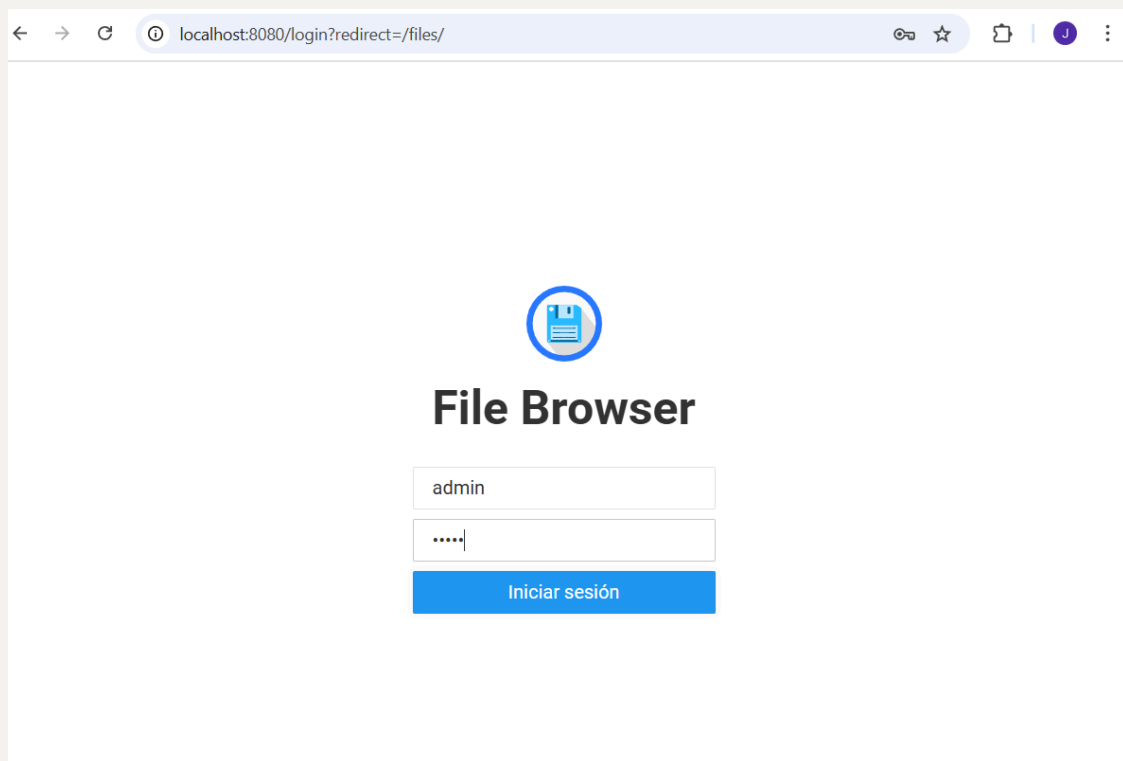
Nombre	Fecha
config	07/04,
data	07/04,
README.assets	07/04,
compose.yaml	07/04,
README.md	07/04,

Nombre	re
filebrowser.db	07,

Podemos comprobar que efectivamente tenemos **data y config**

A continuación en el navegador escribimos localhost:8080 y entraremos en el panel de login de **FileBrowser**

Captura de pantalla donde se vea la aplicación funcionando, sube algún fichero, cambia el lenguaje a español...



Cambiamos el idioma a español

[Profile Settings](#)[Share Management](#)[Global Settings](#)[User Management](#)

Profile Settings

☐ Hide dotfiles

☐ Use single clicks to open files and directories

☐ Set exact date format

Language

Español

▼

UPDATE

Creo un archivo en mi ordenador y a continuación en la flecha de subir de la aplicación exploro por mi ordenador hacia la ubicación del archivo y la subo

Subir

Seleccione una opción para subir.



Archivo



Carpeta

←

→

🔄

📁 localhost:8080/files

🔍

Buscar...

📁 Mis archivos

🏠

➕ Nueva carpeta

📄 Nuevo archivo

⚙️ Ajustes

🚪 Cerrar sesión

Archivos

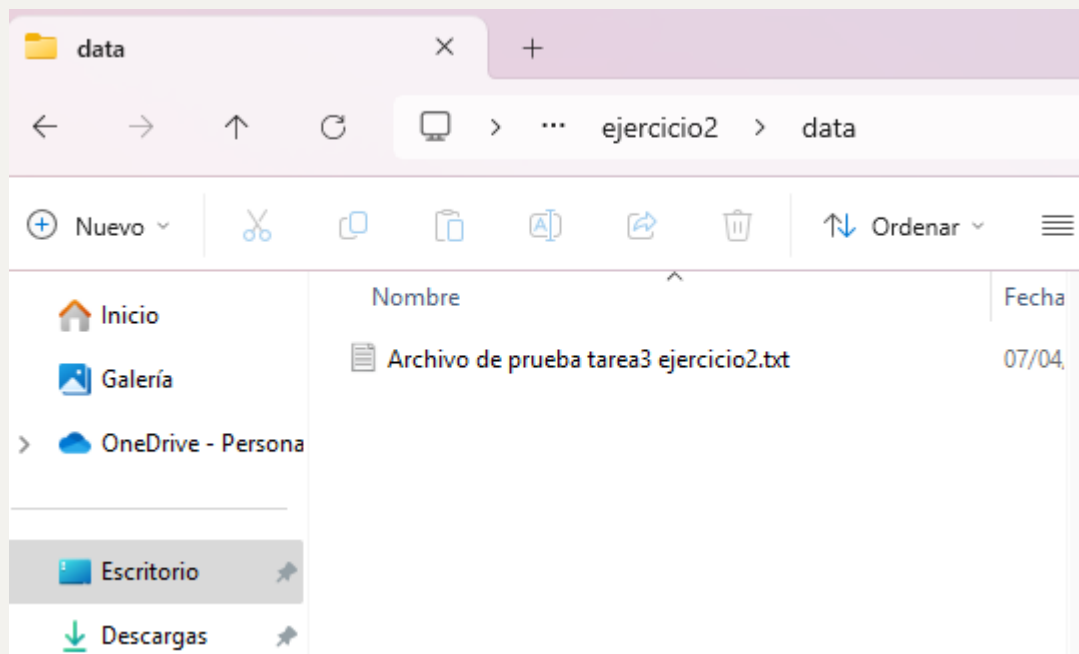
Archivo de prueba tarea3 ...
69 B
hace unos segundos

260 GiB of 520 GiB used

File Browser 2.32.0

Ayuda

Por último desde mi ordenador nos dirigimos a data para comprobar que efectivamente el archivo se ha guardado en data.



Explicar brevemente cómo funciona esta aplicación y qué hace.

Esta aplicación que está creada con docker lo que permite no tenerla instalada en nuestro equipo sino acceder a ella desde el navegador, nos permite manejar archivos de nuestro equipo desde el navegador, lo que nos da muchas posibilidades, ya que esto nos permite añadir, borrar o editar archivos desde cualquier otro dispositivo que esté conectado a internet.

Con la configuración adecuada podríamos acceder a archivos subidos a FileBrowser desde nuestro móvil, en esta práctica los archivos se almacenan en carpeta local gracias al uso de bind-mounts.