

# Actividad Evaluable 3 - Docker - EJERCICIO 2 - Docker Compose

Tarea creada por Aida Montes Cabello para el módulo Despliegue de Aplicaciones Web.

## Actividad Evaluable 3 - Docker - EJERCICIO 2 - Docker Compose

Introducción

1. Archivo `compose.yaml`

2. Cómo cargar el archivo YAML en Docker Desktop

Funcionamiento FileBrowser

a) Acceso a FileBrowser

b) Navegación por FileBrowser

Carpetas y archivos creados

## Introducción

En este ejercicio utilizaremos `Docker Compose` a través de `Docker Desktop` para desplegar la aplicación `FileBrowser` cuya imagen está disponible en el repositorio oficial de Docker Hub: <https://hub.docker.com/r/hurlenko/filebrowser>. El despliegue se hará con la creación de un archivo `compose.yaml`, en el cual se definirán los servicios, puertos y volúmenes necesarios para el correcto funcionamiento de la aplicación.

## 1. Archivo `compose.yaml`

En este apartado, mostramos y explicamos el contenido del archivo `compose.yaml` que vamos a utilizar en este ejercicio.

```
services:
  filebrowser:
    image: hurlenko/filebrowser
    container_name: filebrowser
    ports:
      - "8080:8080"
    volumes:
      - ./data:/data
      - ./config:/config
    restart: unless-stopped
```

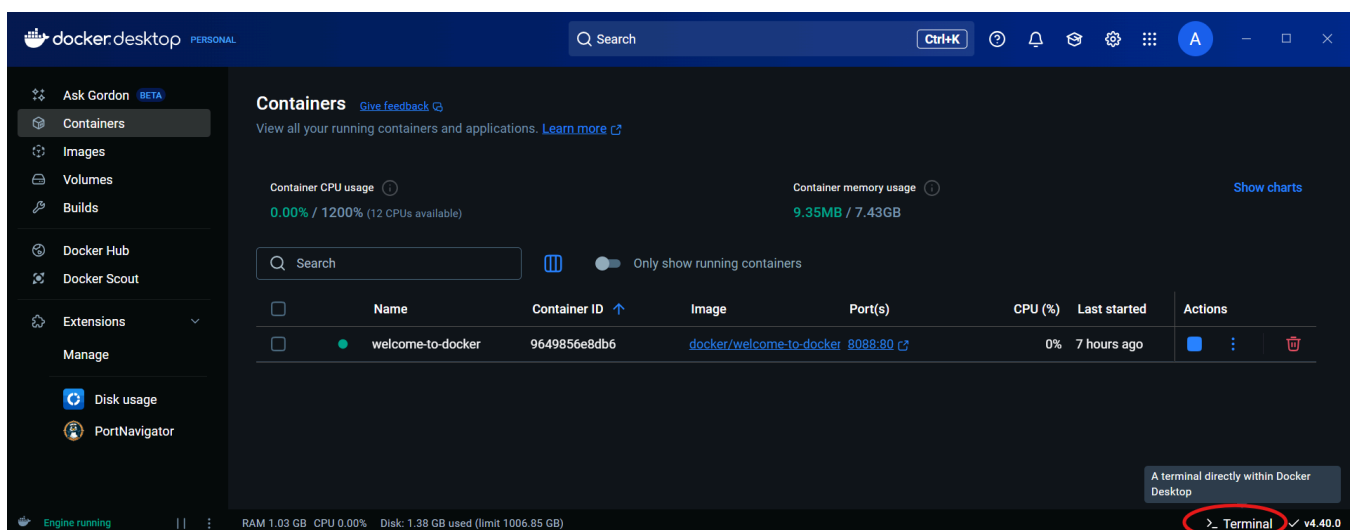
En este archivo se define un servicio que se llama `filebrowser` utilizando la imagen oficial `hurlenko/filebrowser`. A continuación, se explican los elementos de este archivo:

- `services:` : servicio que se desplegará: `filebrowser`.
- `image:` : imagen de Docker que se va a utilizar.
- `container_name:` : nombre del contenedor.

- **ports:** : abre el puerto 8080 del contenedor para poder acceder a la aplicación desde el navegador (`localhost:8080`).
- **volumes:** :
  - `./data:/data`: carpeta local (data) donde se almacenarán los archivos que se suban desde `FileBrowser`.
  - `./config:/config`: guarda la configuración de la aplicación.
- **restart: unless-stopped**: el contenedor se reinicia automáticamente.

## 2. Cómo cargar el archivo YAML en Docker Desktop

Para cargar el archivo que hemos creado en el punto anterior, abriremos la terminal integrada en **Docker Desktop** pulsando en `>_ Terminal` que se encuentra abajo en la parte derecha.



Una vez abierta, nos tenemos que situar en la carpeta donde está el archivo:

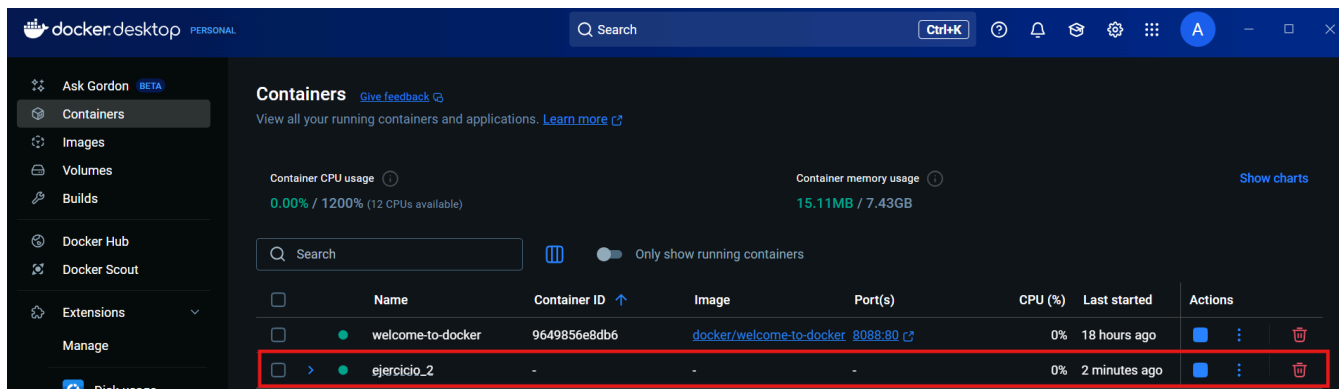
```
Terminal
PS C:\Users\Usuario> cd C:\Users\Usuario\Desktop\DAW\Ejercicio_2
PS C:\Users\Usuario\Desktop\DAW\Ejercicio_2> |
```

Tras acceder a la carpeta, ejecutamos el siguiente comando:

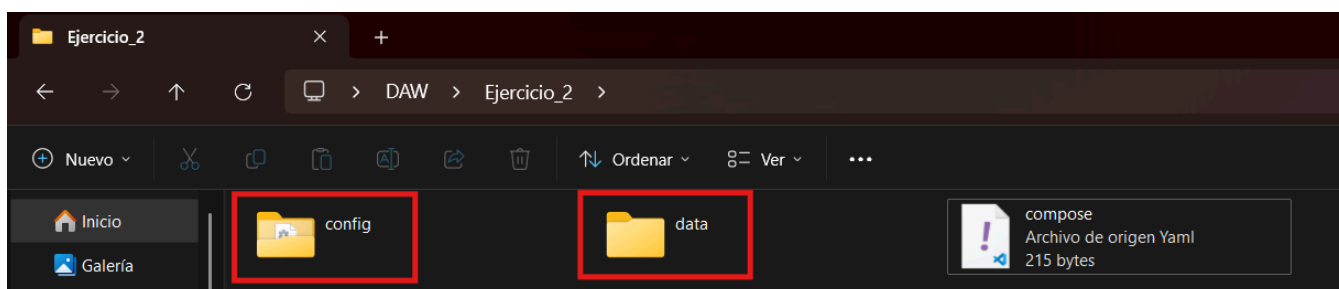
```
docker-compose up -d
```

```
Terminal
PS C:\Users\Usuario\Desktop\DAW\Ejercicio_2> docker-compose up -d
[+] Running 4/4
✓ filebrowser Pulled 10.9s
✓ bbe6e0b5af75 Pull complete 1.0s
✓ 7b276281f0bb Pull complete 3.3s
✓ 1f3e46996e29 Pull complete 1.4s
[+] Running 2/2
✓ Network ejercicio_2_default Created 0.1s
✓ Container filebrowser Started 0.6s
```

Comprobamos que se ha añadido como contenedor en la pestaña **Containers**.



También vamos a verificar que se han creado las carpetas locales (`./data`, `./config`) ya que hemos utilizado bind mounts. Para ello accedemos a la carpeta donde tenemos el `.yaml`.



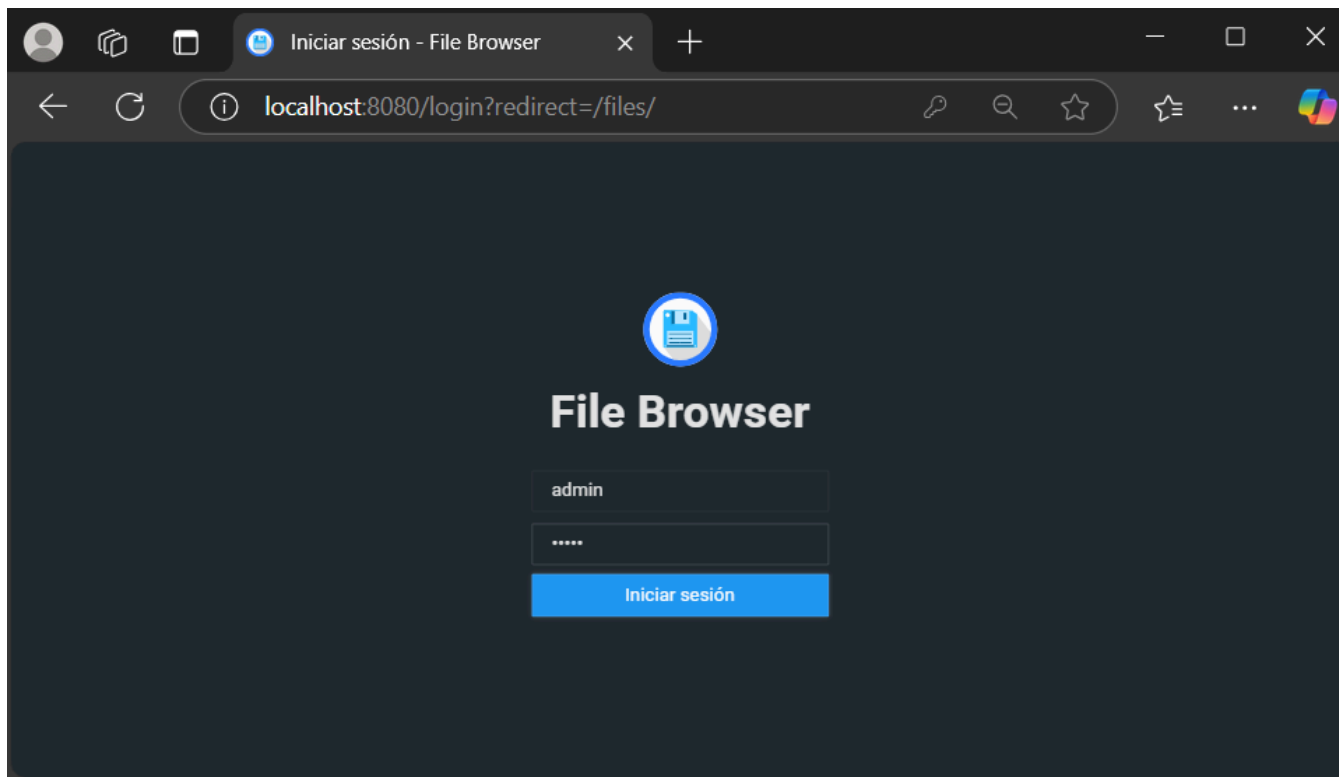
## Funcionamiento FileBrowser

FileBrowser es una aplicación web ligera la cual ofrece una interfaz gráfica para gestionar archivos y carpetas dentro del volumen `/data`. Permite subir archivos desde nuestro equipo local, crear o editarlos directamente desde la interfaz, descargar contenido, organizar carpetas y, en versiones avanzadas, configurar usuarios, roles y permisos. En resumen, facilita la gestión de archivos almacenados en un directorio del sistema de archivos del servidor, en nuestro caso, dentro del contenedor Docker.

En el punto anterior, hemos desplegado el contenedor, lo siguiente que tendremos que hacer es acceder a la aplicación desde el navegador mediante la dirección `http://localhost:8080`. Para el inicio de sesión, se utilizarán las credenciales predeterminadas: usuario `admin` y contraseña `admin`.

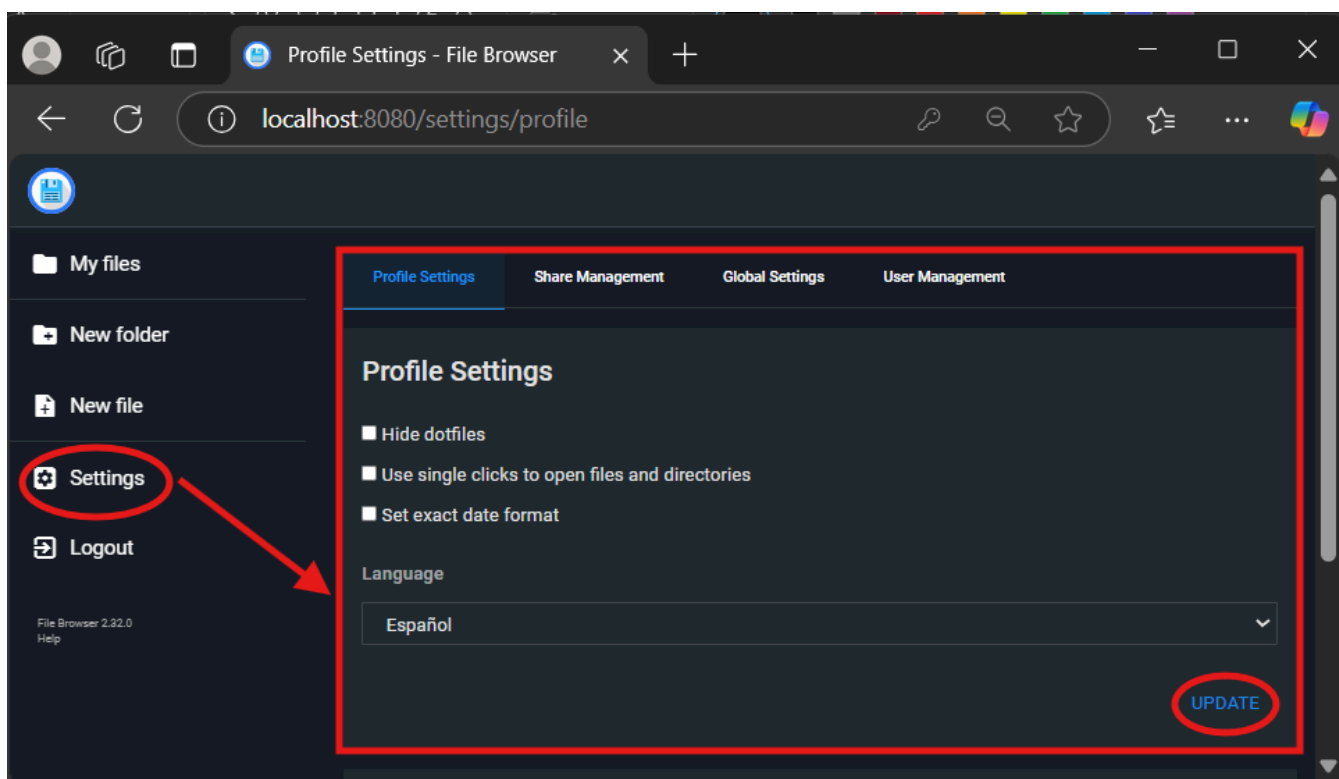
### a) Acceso a FileBrowser

Como bien hemos comentado, para entrar en la interfaz gráfica de la aplicación, vamos a nuestro navegador e ingresamos la dirección `http://localhost:8080` e introducimos el usuario y la contraseña.

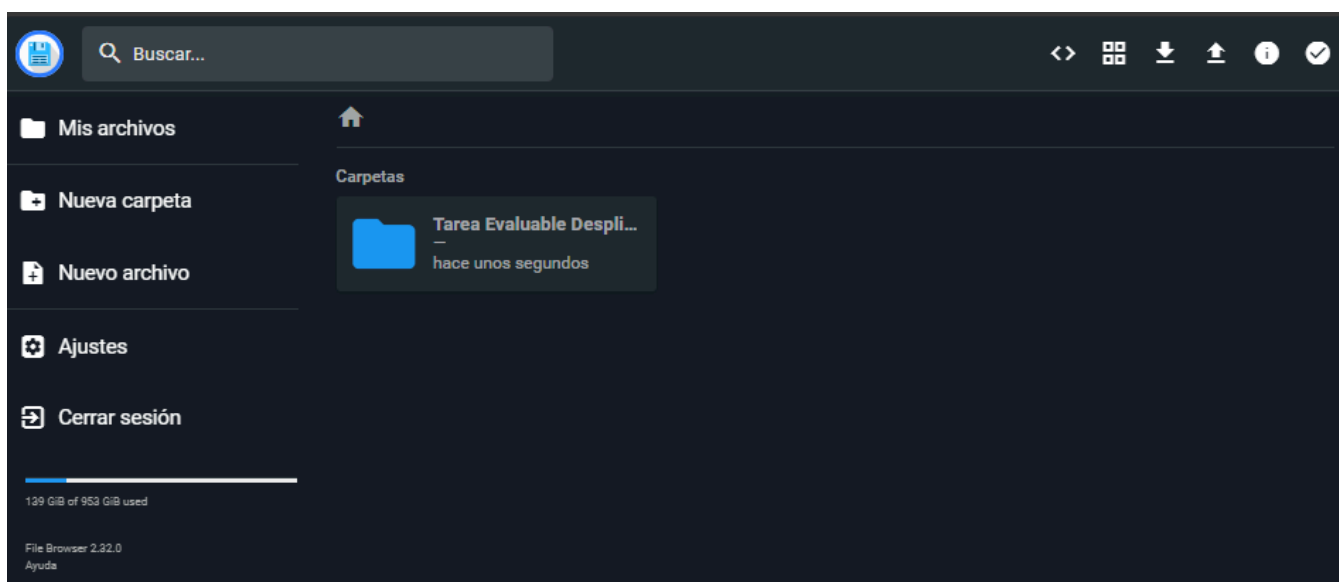
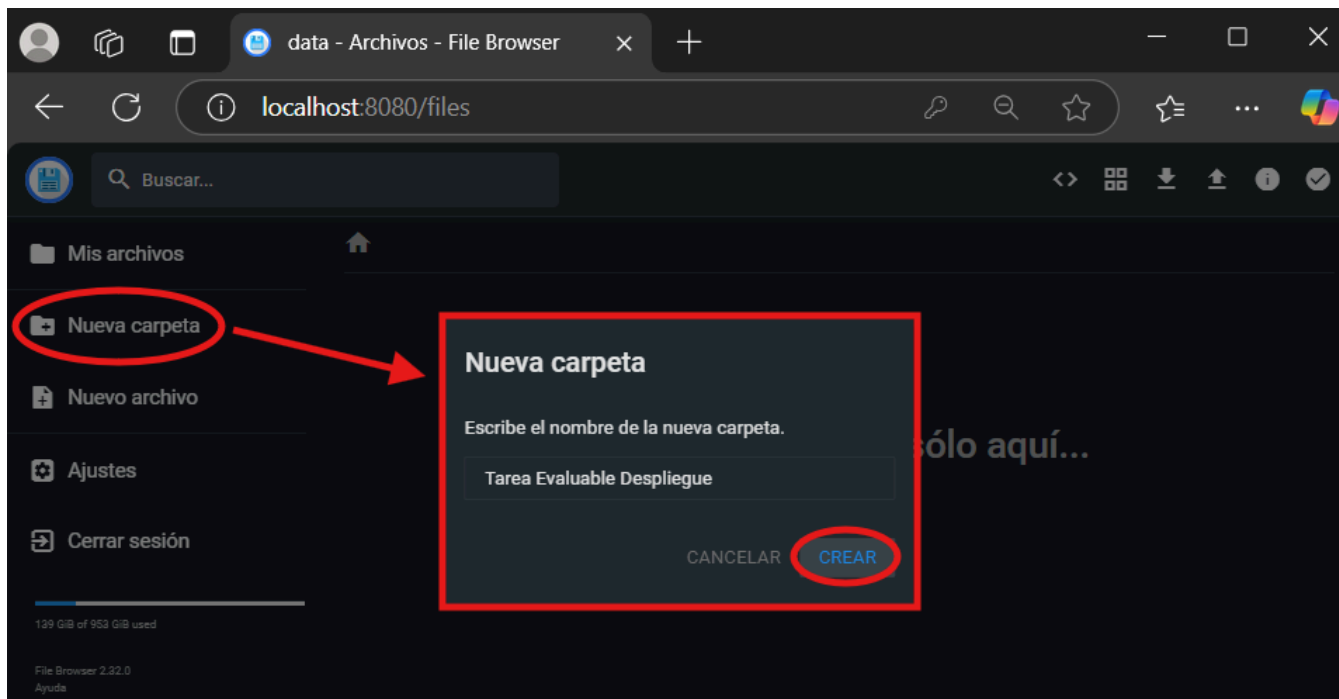


## b) Navegación por FileBrowser

Una vez dentro de la aplicación lo primero que vamos a hacer es cambiar el idioma a español, para ello iremos al icono de **Settings** que se encuentra en el menú izquierdo.

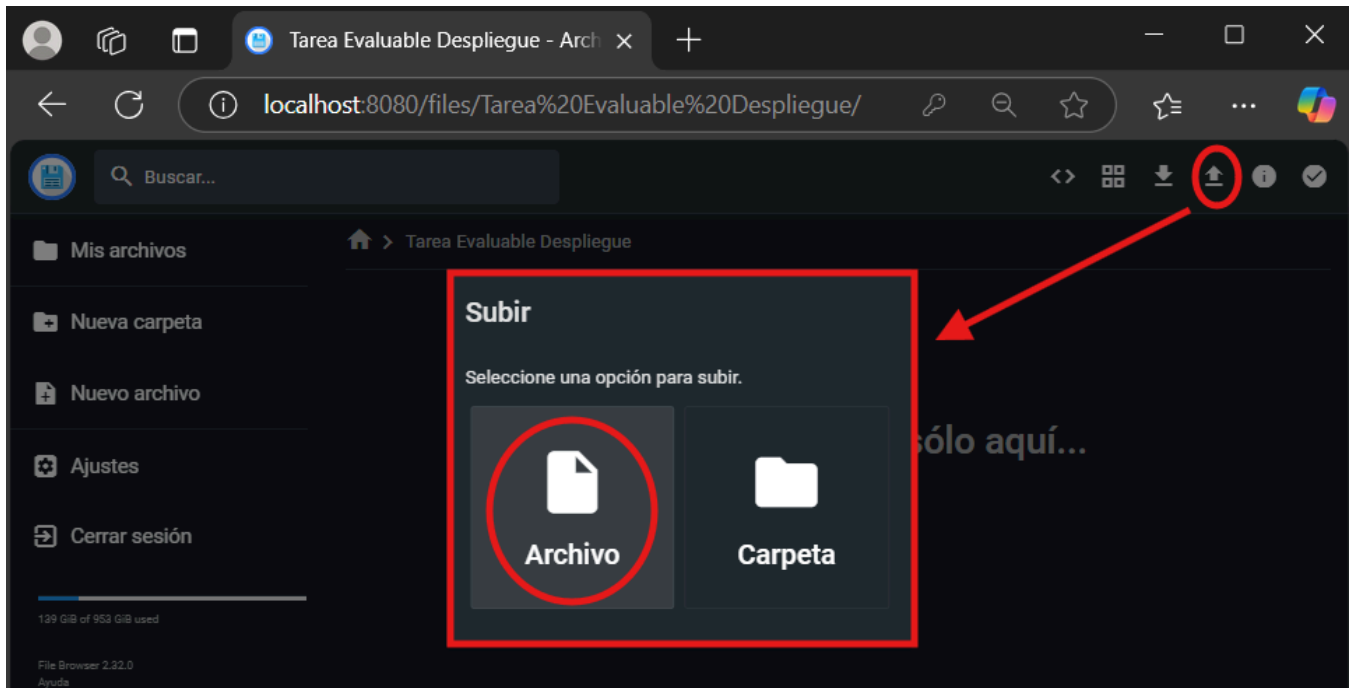


Después de cambiar el idioma vamos a realizar alguna tarea, como por ejemplo la creación de una carpeta y la subida de un archivo. Para la creación de una carpeta lo que tendremos es que acceder a la pestaña **Nueva carpeta** situada en el menú izquierdo.

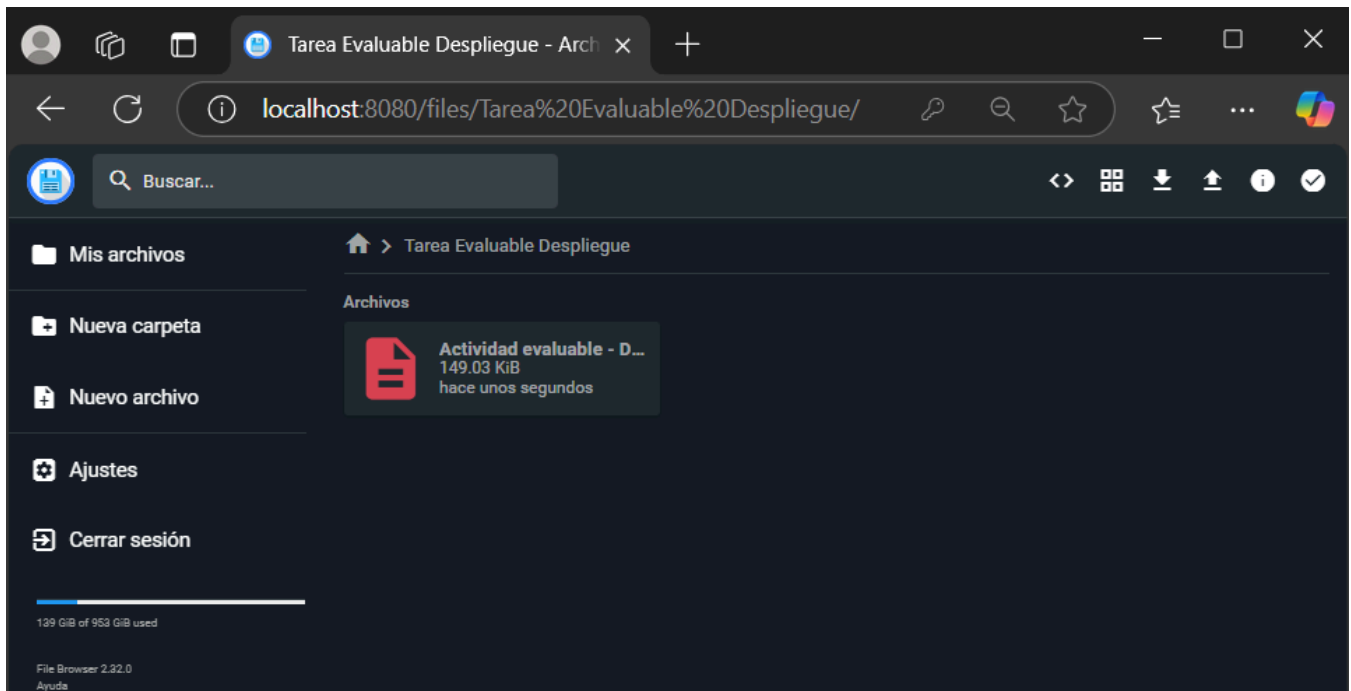


A continuación vamos a subir el archivo PDF de la tarea dentro de la carpeta que hemos creado previamente, para ello, accedemos a dicha carpeta haciendo doble clic sobre ella.

Una vez dentro, pulsamos sobre el icono de flecha hacia arriba, que se encuentra en la parte superior derecha. Seleccionamos **Archivo** y buscamos, seleccionamos el PDF de la tarea ( **Actividad Evaluable 3 - Docker** ) y lo subimos.



Ahora nos aparecerá el archivo dentro de la carpeta que hemos creado.



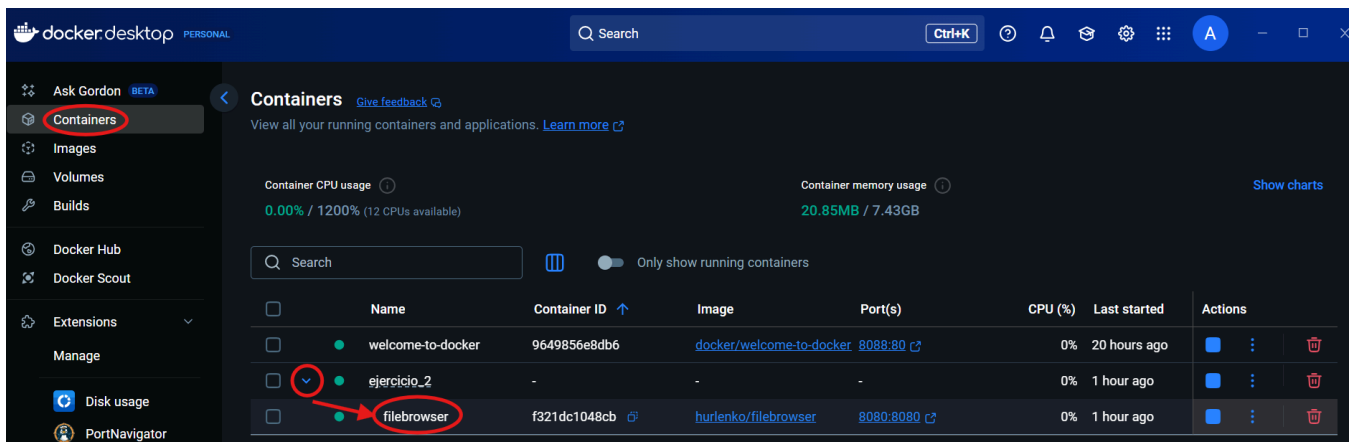
## Carpetas y archivos creados

Estamos usando bind-mounts, estamos mapeando carpetas locales, ya que en nuestro `compose.yaml` hemos puesto:

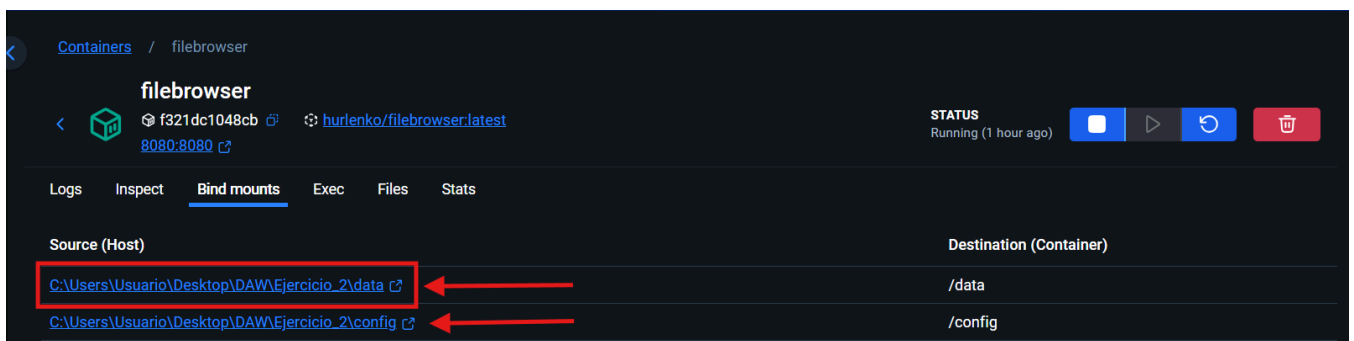
```
volumes:
  - ./data:/data
  - ./config:/config
```

Por lo tanto, las carpetas `data` y `config` están dentro de la carpeta donde se encuentra el archivo `yaml`. A su vez, el archivo que hemos subido desde FileBrowser (`Actividad Evaluable - Docker`) aparecerá en la carpeta `data`.

Para verlo más gráficamente, lo que vamos a hacer es acceder al contenedor de `filebrowser` para ver las rutas de estas carpetas. Para ello iremos a `Containers` y apretaremos sobre la flecha que aparece a la izquierda de `ejercicio_2` para desplegar el contenedor. Una vez desplegado, pulsamos sobre él.



Dentro, seleccionamos la pestaña `Bind mounts`, y aquí podemos observar los paths en las que se montan los datos.



Por último, vamos a acceder en nuestro caso a la ruta de la carpeta `data` para poder ver la carpeta creada desde FileBrowser y el archivo PDF que hemos subido.

