

3.10 PhotoPrism. Alternativa a las nubes de fotos



Imagen obtenida de <https://www.photoprism.app/>

Esta herramienta sirve para...

gestionar tu contenido multimedia de un modo avanzado. Haciendo uso de la demo <https://demo.photoprism.app/library/browse> podrás hacerte una idea de sus posibilidades.

Web de proyecto y otros enlaces de interés

Web: <https://www.photoprism.app/>

Repositorio: <https://github.com/photoprism/photoprism>

Puesta en marcha

Como en ocasiones anteriores vamos a hacer con docker-compose para ello accedemos al terminal y escribimos

```
cd $HOME
mkdir photoprism
cd photoprism
nano docker-compose.yml
```

y dentro del fichero copiaremos el siguiente contenido (adaptado del fichero visto en <https://dl.photoprism.app/docker/docker-compose.yml>):

```
version: '3.5'

services:
  photoprism:
    image: photoprism/photoprism:latest
    depends_on:
      - mariadb
    ## Don't enable automatic restarts until PhotoPrism has been properly configured and tested!
    ## If the service gets stuck in a restart loop, this points to a memory, filesystem, network, or database issue:
    ## https://docs.photoprism.app/getting-started/troubleshooting/#fatal-server-errors
    # restart: unless-stopped
    security_opt:
      - seccomp:unconfined
      - apparmor:unconfined
    ports:
      - "2342:2342" # HTTP port (host:container)
    environment:
      PHOTOPRISM_ADMIN_USER: "admin"           # superadmin username
      PHOTOPRISM_ADMIN_PASSWORD: "insecure"    # initial superadmin password (minimum 8 characters)
      PHOTOPRISM_AUTH_MODE: "password"         # authentication mode (public, password)
      PHOTOPRISM_SITE_URL: "http://photoprism.me:2342/" # server URL in the format
      "http(s)://domain.name(:port)/(path)"
      PHOTOPRISM_ORIGINALS_LIMIT: 5000         # file size limit for originals in MB (increase for high-res video)
      PHOTOPRISM_HTTP_COMPRESSION: "gzip"      # improves transfer speed and bandwidth utilization (none
or gzip)
      PHOTOPRISM_LOG_LEVEL: "info"            # log level: trace, debug, info, warning, error, fatal, or panic
```

```

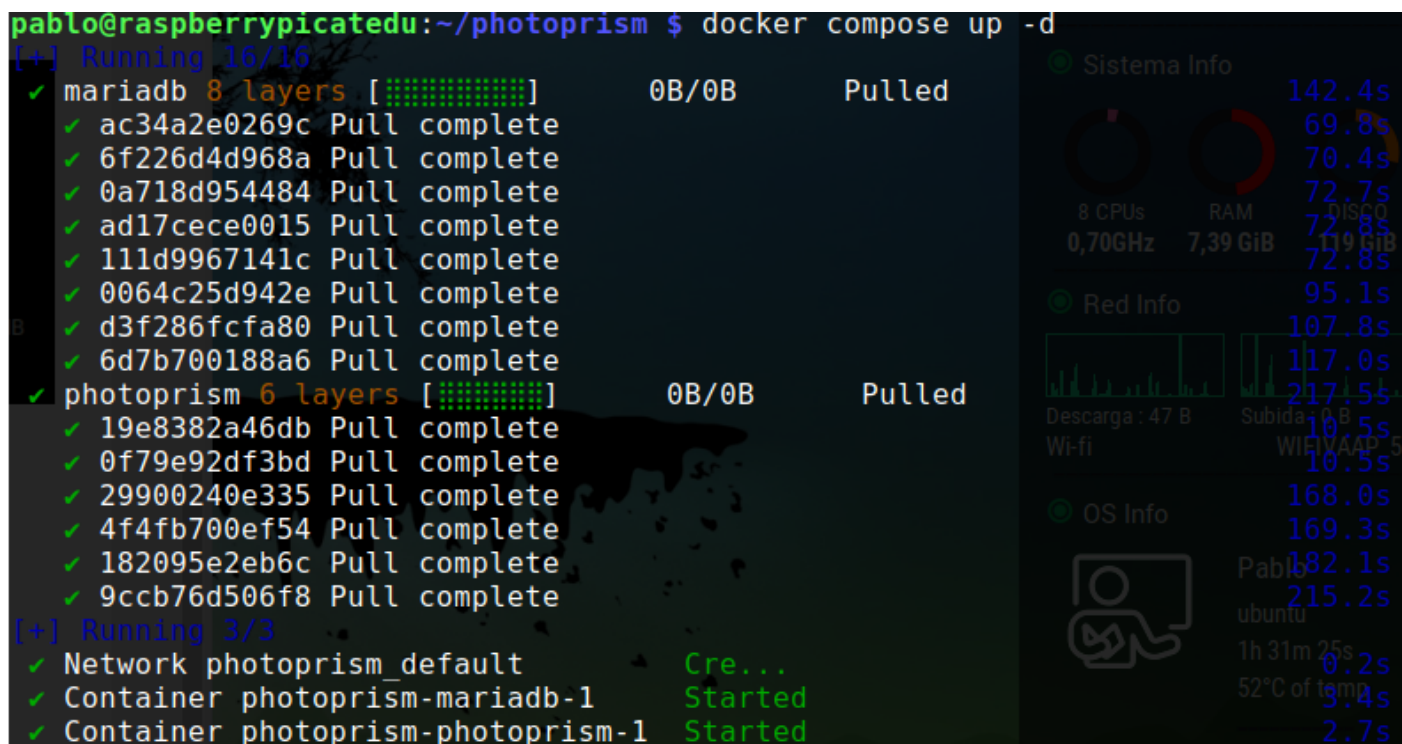
PHOTOPRISM_READONLY: "false"          # do not modify originals directory (reduced functionality)
PHOTOPRISM_EXPERIMENTAL: "false"       # enables experimental features
PHOTOPRISM_DISABLE_CHOWN: "false"      # disables updating storage permissions via chmod and
chown on startup
PHOTOPRISM_DISABLE_WEBDAV: "false"     # disables built-in WebDAV server
PHOTOPRISM_DISABLE_SETTINGS: "false"   # disables settings UI and API
PHOTOPRISM_DISABLE_TENSORFLOW: "false" # disables all features depending on TensorFlow
PHOTOPRISM_DISABLE_FACES: "false"      # disables face detection and recognition (requires
TensorFlow)
PHOTOPRISM_DISABLE_CLASSIFICATION: "false" # disables image classification (requires TensorFlow)
PHOTOPRISM_DISABLE_RAW: "false"        # disables indexing and conversion of RAW files
PHOTOPRISM_RAW_PRESETS: "false"        # enables applying user presets when converting RAW files
(reduces performance)
PHOTOPRISM_JPEG_QUALITY: 85             # a higher value increases the quality and file size of JPEG
images and thumbnails (25-100)
PHOTOPRISM_DETECT_NSFW: "false"        # automatically flags photos as private that MAY be offensive
(requires TensorFlow)
PHOTOPRISM_UPLOAD_NSFW: "true"         # allows uploads that MAY be offensive (no effect without
TensorFlow)
# PHOTOPRISM_DATABASE_DRIVER: "sqlite" # SQLite is an embedded database that doesn't require a
server
PHOTOPRISM_DATABASE_DRIVER: "mysql"    # use MariaDB 10.5+ or MySQL 8+ instead of SQLite for
improved performance
PHOTOPRISM_DATABASE_SERVER: "mariadb:3306" # MariaDB or MySQL database server (hostname:port)
PHOTOPRISM_DATABASE_NAME: "photoprism"   # MariaDB or MySQL database schema name
PHOTOPRISM_DATABASE_USER: "photoprism"   # MariaDB or MySQL database user name
PHOTOPRISM_DATABASE_PASSWORD: "insecure" # MariaDB or MySQL database user password
PHOTOPRISM_SITE_CAPTION: "AI-Powered Photos App"
PHOTOPRISM_SITE_DESCRIPTION: ""          # meta site description
PHOTOPRISM_SITE_AUTHOR: ""              # meta site author
working_dir: "/photoprism" # do not change or remove
## Storage Folders: "~" is a shortcut for your home directory, "." for the current directory
volumes:
# "/host/folder:/photoprism/folder"      # Example
- "~:/Pictures:/photoprism/originals"    # Original media files (DO NOT REMOVE)
# - "/example/family:/photoprism/originals/family" # *Additional* media folders can be mounted like this
# - "~:/Import:/photoprism/import"       # *Optional* base folder from which files can be imported to
originals
- "./storage:/photoprism/storage"        # *Writable* storage folder for cache, database, and sidecar
files (DO NOT REMOVE)

```

```
## Database Server (recommended)
## see https://docs.photoprism.app/getting-started/faq/#should-i-use-sqlite-mariadb-or-mysql
mariadb:
  ## If MariaDB gets stuck in a restart loop, this points to a memory or filesystem issue:
  ## https://docs.photoprism.app/getting-started/troubleshooting/#fatal-server-errors
  restart: unless-stopped
  image: mariadb:10.10
  security_opt: # see https://github.com/MariaDB/mariadb-docker/issues/434#issuecomment-1136151239
    - seccomp:unconfined
    - apparmor:unconfined
  command: mysqld --innodb-buffer-pool-size=512M --transaction-isolation=READ-COMMITTED --character-set-
server=utf8mb4 --collation-server=utf8mb4_unicode_ci --max-connections=512 --innodb-rollback-on-
timeout=OFF --innodb-lock-wait-timeout=120
  ## Never store database files on an unreliable device such as a USB flash drive, an SD card, or a shared
network folder:
  volumes:
    - "./database:/var/lib/mysql" # DO NOT REMOVE
  environment:
    MARIADB_AUTO_UPGRADE: "1"
    MARIADB_INITDB_SKIP_TZINFO: "1"
    MARIADB_DATABASE: "photoprism"
    MARIADB_USER: "photoprism"
    MARIADB_PASSWORD: "insecure"
    MARIADB_ROOT_PASSWORD: "insecure"
```

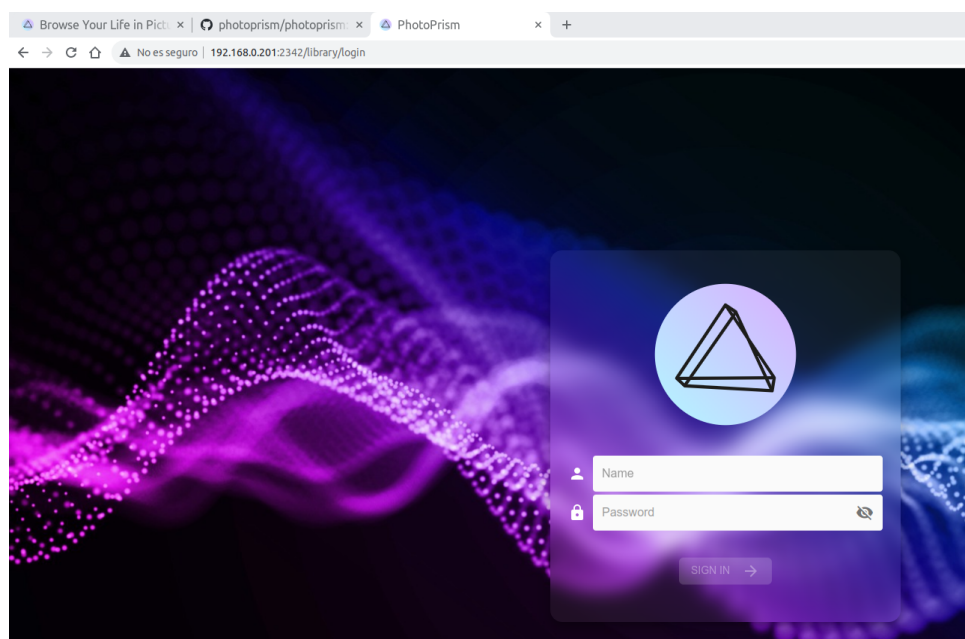
como en ocasiones anteriores, para guardar los cambios pulsaremos `control + x` y cuando nos pregunte aceptaremos. Una vez volvamos a estar en el terminal, escribiremos `docker compose up -d` para lanzar los servicios ubicados dentro del fichero docker-compose. Le va a costar un buen rato extraer las imágenes y empezar el despliegue, paciencia. El resultado será similar al siguiente:

```
pablo@raspberrypicadedu:~/photoprism $ docker compose up -d
[+] Running 16/16
✓ mariadb 8 layers [██████████] 0B/0B Pulled
✓ ac34a2e0269c Pull complete
✓ 6f226d4d968a Pull complete
✓ 0a718d954484 Pull complete
✓ ad17cece0015 Pull complete
✓ 111d9967141c Pull complete
✓ 0064c25d942e Pull complete
✓ d3f286fcfa80 Pull complete
✓ 6d7b700188a6 Pull complete
✓ photoprism 6 layers [██████████] 0B/0B Pulled
✓ 19e8382a46db Pull complete
✓ 0f79e92df3bd Pull complete
✓ 29900240e335 Pull complete
✓ 4f4fb700ef54 Pull complete
✓ 182095e2eb6c Pull complete
✓ 9ccb76d506f8 Pull complete
[+] Running 3/3
✓ Network photoprism_default Cre...
✓ Container photoprism-mariadb-1 Started
✓ Container photoprism-photoprism-1 Started
```

A screenshot of a terminal window on a Raspberry Pi. The terminal shows the command 'docker compose up -d' being executed. The output indicates that several Docker images are being pulled, including 'mariadb' and 'photoprism'. The 'photoprism' container is shown as 'Started'. On the right side of the terminal, there is a system information overlay showing various metrics like CPU usage, RAM, and disk space.

Elaboración propia

Se paciente, le cuesta un par de minutos arrancar. Si tras esa breve pausa accedemos al servicio como venimos haciendo, en este caso en el puerto 2342, veremos algo similar a:



Elaboración propia

El usuario y contraseña por defecto son `admin` y `insecure`. Fíjate que vienen establecidos en el fichero docker-compose.

Este servicio está al límite en cuanto a la capacidad de la Raspberry Pi 4 modelo B de 4 GB. Valora si la solución que hemos visto en el capítulo anterior es suficiente para ti.

Revision #10

Created 4 February 2023 10:07:33 by Pablo Ruiz

Updated 20 July 2023 17:49:35 by Pablo Ruiz