3.14 Paperless-ngx. Gestión documental



Imagen obtenida de https://docs.paperless-ngx.com/

Esta herramienta sirve para...

la gestión documental. Al igual que vimos con PaperMerge, cuenta con OCR.

Web de proyecto y otros enlaces de interés

Web del proyecto: https://docs.paperless-ngx.com/

Repositorio en github: https://github.com/paperless-ngx/paperless-ngx

Copyright 2026

Despliegue

En esta ocasión vamos a variar ligeramente la forma de realizar el despliegue. Dentro del respositorio, la empresa desarrolladora ha creado un script que automatiza la instalación vía docker-compose y, dado que queremos simplicidad, haremos uso del mismo

> Cuidado con ejecutar cualquier script que encontréis en internet pues podría ser malicioso.

Vamos allá, accedemos al terminal y escribimos:

```
cd $HOME
mkdir paperless
cd paperless
```

y ahora si, lanzamos el script del siguiente modo

```
bash -c "$(curl -L https://raw.githubusercontent.com/paperless-ngx/paperless-ngx/main/install-paperless-ngx.sh)"
```

aparecerá un asistente que nos irá realizando una serie de preguntas para configurar el servicio y que tendrá un aspecto similar a este (remarco en un cuadrado rojo lo que he ido respondiendo):

Copyright 2026

```
pi@mediacenter:~/paperless $ bash -c "$(curl -L https://raw.githubusercontent.com/paperless-ngx/paperless-ngx/main/install-paperless-ngx.sh)"
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 11122  100 11122    0     0  37197      0 --:--:-- --:--:-- --:--:-- 37197
/usr/bin/docker-compose

##############################################
###   paperless-ngx docker installation   ###
##############################################

This script will download, configure and start paperless-ngx.

1. Application configuration
============================

The URL paperless will be available at. This is required if the
installation will be accessible via the web, otherwise can be left blank.
Example: https://paperless.example.com

URL []:

The port on which the paperless webserver will listen for incoming
connections.

Port [8000]:

Paperless requires you to configure the current time zone correctly.
Otherwise, the dates of your documents may appear off by one day,
depending on where you are on earth.
Example: Europe/Berlin
See here for a list: https://en.wikipedia.org/wiki/List_of_tz_database_time_zones

Current time zone [Europe/Madrid]:

Database backend: PostgreSQL, MariaDB, and SQLite are available. Use PostgreSQL
if unsure. If you're running on a low-power device such as Raspberry
Pi, use SQLite to save resources.

Database backend (postgres sqlite mariadb) [postgres]: sqlite

Paperless is able to use Apache Tika to support Office documents such as
Word, Excel, Powerpoint, and Libreoffice equivalents. This feature
requires more resources due to the required services.

Enable Apache Tika? (yes no) [no]: no
```

Copyright 2026

```
Specify the default language that most of your documents are written in.
Use ISO 639-2, (T) variant language codes:
https://www.loc.gov/standards/iso639-2/php/code_list.php
Common values: eng (English) deu (German) nld (Dutch) fra (French)
This can be a combination of multiple languages such as deu+eng

OCR language [eng]: es
```

usa "spa" en lugar de "es" (sin las comillas)

```
Specify the user id and group id you wish to run paperless as.
Paperless will also change ownership on the data, media and consume
folder to the specified values, so it's a good idea to supply the user id
and group id of your unix user account.
If unsure, leave default.

User ID [1000]:
Group ID [1000]:

2. Folder configuration
=======================

The target folder is used to store the configuration files of
paperless. You can move this folder around after installing paperless.
You will need this folder whenever you want to start, stop, update or
maintain your paperless instance.

Target folder [/home/pi/paperless/paperless-ngx]:

The consume folder is where paperless will search for new documents.
Point this to a folder where your scanner is able to put your scanned
documents.

CAUTION: You must specify an absolute path starting with / or a relative
path starting with ./ here. Examples:
  /mnt/consume
  ./consume

Consume folder [/home/pi/paperless/paperless-ngx/consume]:

The media folder is where paperless stores your documents.
Leave empty and docker will manage this folder for you.
Docker usually stores managed folders in /var/lib/docker/volumes.

CAUTION: If specified, you must specify an absolute path starting with /
or a relative path starting with ./ here.

Media folder []: /mnt/Expansion/docs
```

```
The data folder is where paperless stores other data, such as your
SQLite database, the search index and other data.
As with the media folder, leave empty to have this managed by docker.

CAUTION: If specified, you must specify an absolute path starting with /
or a relative path starting with ./ here.

Data folder []:

3. Login credentials
====================

Specify initial login credentials. You can change these later.
A mail address is required, however it is not used in paperless. You don't
need to provide an actual mail address.

Paperless username [pi]:
Paperless password:
Paperless password (again):
Email [pi@localhost]: pablorui[redacted]

Summary
=======

Target folder: /home/pi/paperless/paperless-ngx
Consume folder: /home/pi/paperless/paperless-ngx/consume
Media folder: /mnt/Expansion/docs
Data folder: Managed by docker

URL:
Port: 8000
Database: sqlite
Tika enabled: no
OCR language: es
User id: 1000
Group id: 1000

Paperless username: pi
Paperless email: pabloruiz[redacted]com

Press any key to install.
```

```
Press any key to install.

Installing paperless...

--2023-03-11 10:53:31--  https://raw.githubusercontent.com/paperless-ngx/paperless-ngx/main/docker/compose/docker-compose.sqlite.yml
Resolviendo raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.111.133, 185.199.109.133, 185.199.108.133, ...
Conectando con raw.githubusercontent.com (raw.githubusercontent.com)[185.199.111.133]:443... conectado.
Petici⌧n HTTP enviada, esperando respuesta... 200 OK
Longitud: 1840 (1,8K) [text/plain]
Grabando a: ⌧docker-compose.yml⌧

docker-compose.yml                                  100%[=============================================================================================================>]   1,80K  --.-KB/s    en 0s

2023-03-11 10:53:32 (6,20 MB/s) - ⌧docker-compose.yml⌧ guardado [1840/1840]

--2023-03-11 10:53:32--  https://raw.githubusercontent.com/paperless-ngx/paperless-ngx/main/docker/compose/.env
Resolviendo raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.108.133, 185.199.110.133, 185.199.111.133, ...
Conectando con raw.githubusercontent.com (raw.githubusercontent.com)[185.199.108.133]:443... conectado.
Petici⌧n HTTP enviada, esperando respuesta... 200 OK
Longitud: 31 [text/plain]
Grabando a: ⌧.env⌧

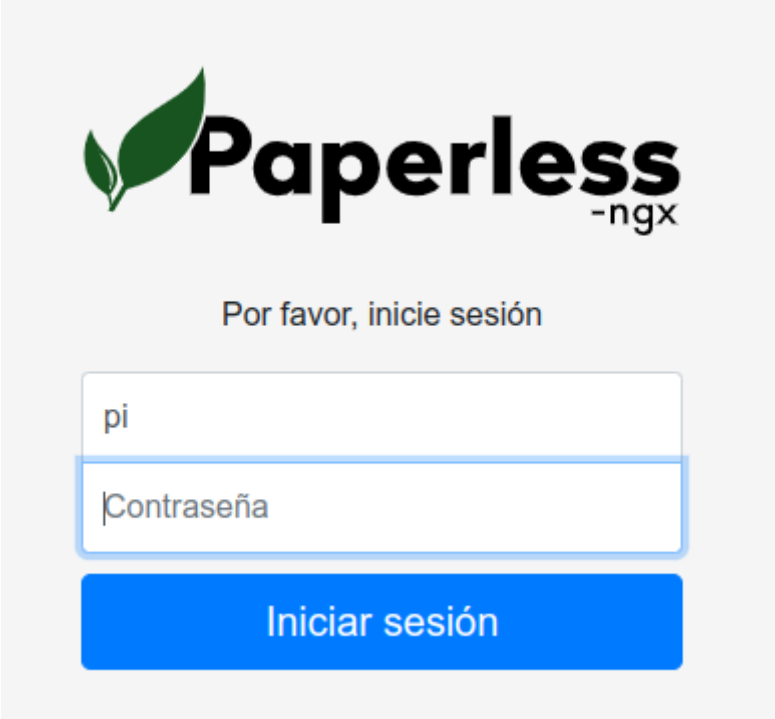
.env                                                100%[=============================================================================================================>]      31  --.-KB/s    en 0s

2023-03-11 10:53:32 (184 KB/s) - ⌧.env⌧ guardado [31/31]

Pulling broker    ... done
Pulling webserver ... done
Creating network "paperless_default" with the default driver
Creating volume "paperless_data" with default driver
Creating volume "paperless_redisdata" with default driver
Creating paperless_broker_1 ... done
Paperless-ngx docker container starting...
Installing languages...
Get:1 http://deb.debian.org/debian bullseye InRelease [116 kB]
Get:2 http://deb.debian.org/debian-security bullseye-security InRelease [48.4 kB]
Get:3 http://deb.debian.org/debian bullseye-updates InRelease [44.1 kB]
Get:4 http://deb.debian.org/debian bullseye/main arm64 Packages [8072 kB]
Get:5 http://deb.debian.org/debian-security bullseye-security/main arm64 Packages [230 kB]
Get:6 http://deb.debian.org/debian bullseye-updates/main arm64 Packages [12.0 kB]
Fetched 8522 kB in 8s (1126 kB/s)
Reading package lists... Done
Package tesseract-ocr-es not found! :(
Creating directory /usr/src/paperless/data/index
Creating directory /usr/src/paperless/media/documents
Creating directory /usr/src/paperless/media/documents/originals
Creating directory /usr/src/paperless/media/documents/thumbnails
Creating directory /tmp/paperless
Adjusting permissions of paperless files. This may take a while.
Waiting for Redis...
Connected to Redis broker.
Apply database migrations...
```

Elaboración propia

Una vez terminemos (le costará un buen rato), como en ocasiones anteriores, accederemos al servicio a través del navegador escribiendo http://nuestra_ip_local:puerto En mi caso es http://192.168.0.201:8000

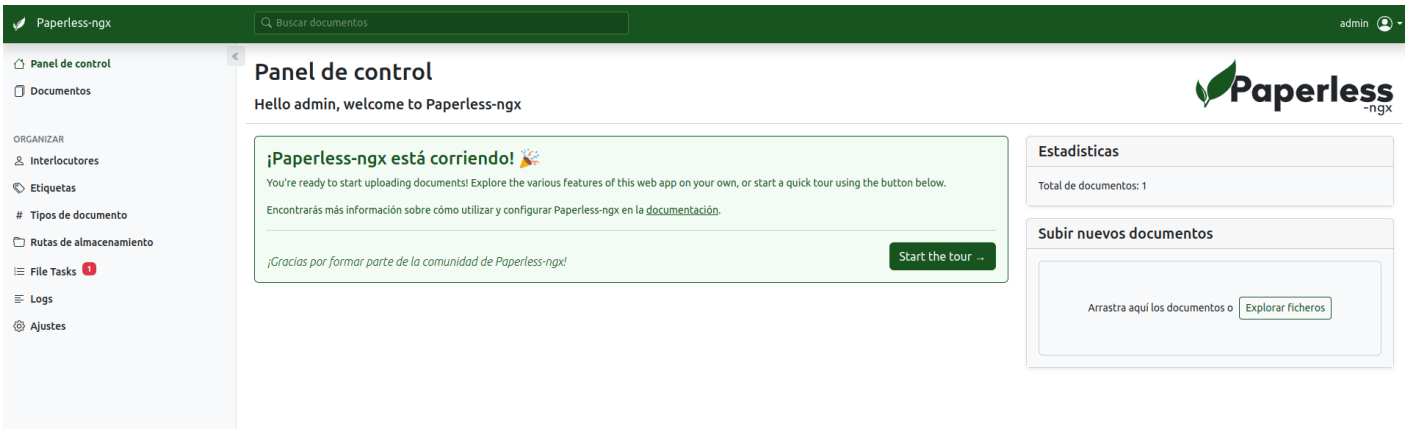


Elaboración propia

Si tienes problemas para acceder vuelve al terminal y escribe `docker-compose run --rm webserver createsuperuser` te pedirá que introduzcas un nuevo usuario y contraseña. A continuación tira el servicio con `docker-compose down` y cuando termine vuelve a levantarlo con `docker-compose up -d` . Con esto deberías poder acceder sin problema.

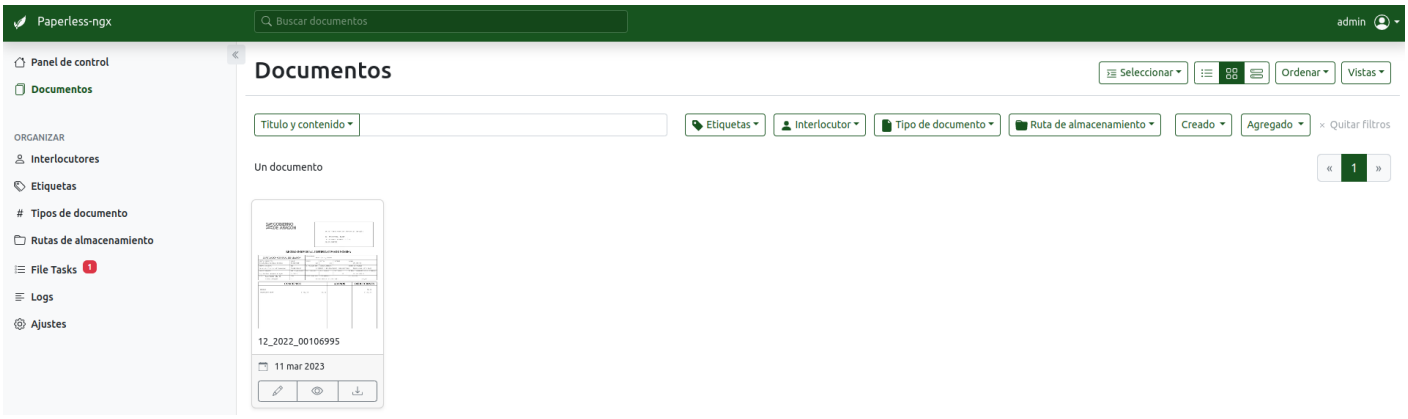
Funcionamiento

La primera vez que accedas verás un aspecto similar al siguiente:



Elaboración propia

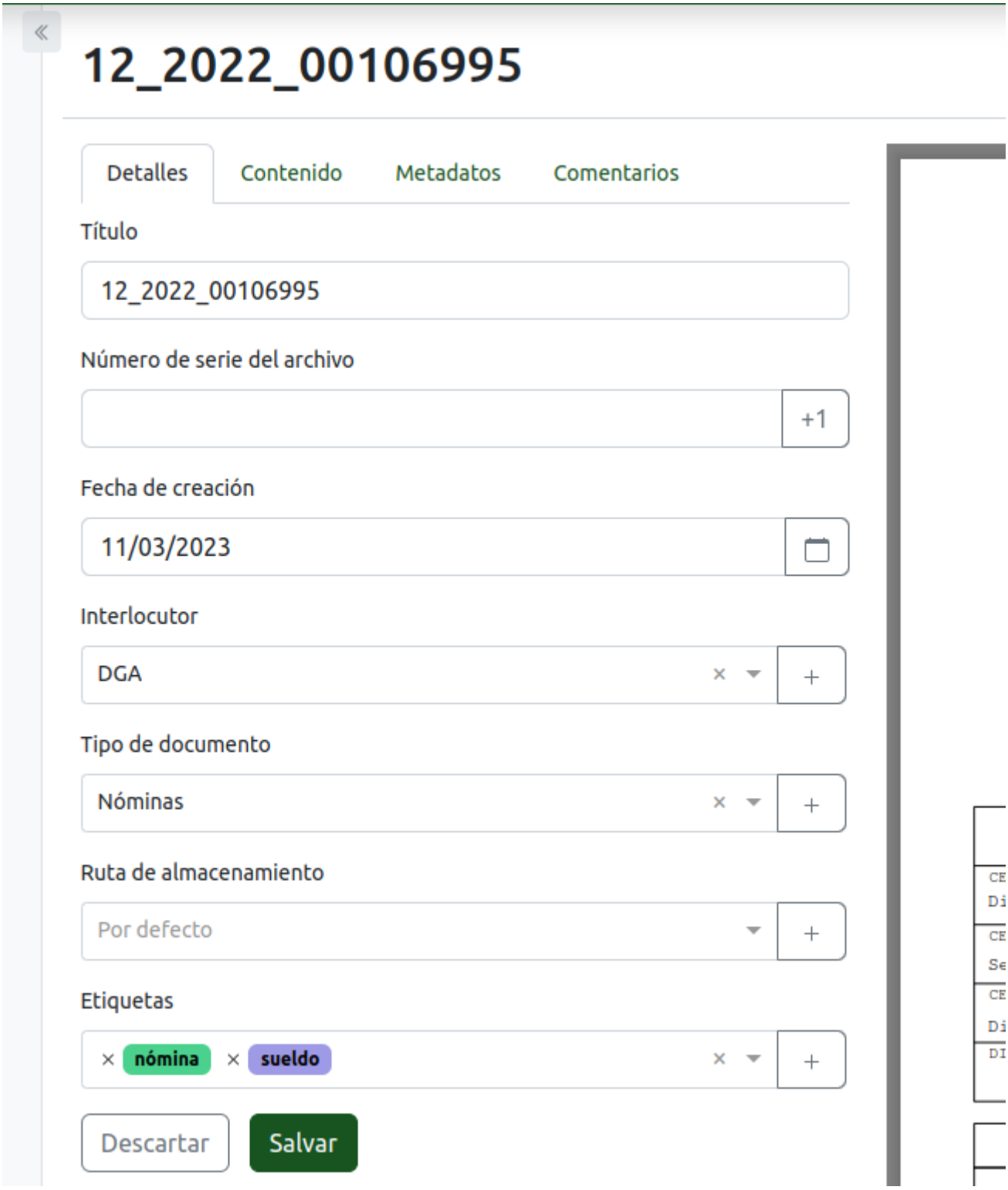
La herramienta tiene muchas posibilidades pero quizás lo mas interesante sea acceder a la sección documentos y allí arrastrar aquellos documentos que nos interese conservar (nóminas, facturas de la luz/gas/teléfono,...) Muchos de estos documentos están accesibles mientras somos clientes de una compañía pero dejan de estar a nuestro alcance cuando dejamos de ser clientes y, en ocasiones, nos pueden resultar de utilidad. En la imagen posterior he arrastrado una nómina



Elaboración propia

La potencia de la herramienta reside en que gracias a OCR y el sistema de etiquetas que incluye podemos ir almacenando documentos de modo que en el futuro resulte muy fácil encontrar la documentación que nos interese. Así, si en el ejemplo anterior, pulsamos en el lápiz que aparece justo del documento subido a modo de ejemplo podremos configurar una serie de parámetros:

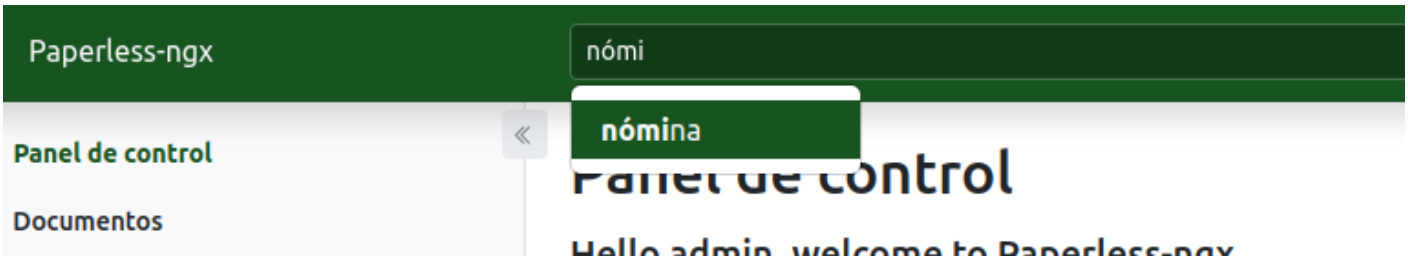
Copyright 2026



Elaboración propia

Una vez clasificados los documentos podemos utilizar el buscador superior o el ubicado en documentos par localizar aquella información que nos interese.

Copyright 2026



Elaboración propia

> Este servicio es también bastante exigente para el hardware que incorpora una Raspberry Pi modelo 4. De todos modos, recuerda que lo que estamos haciendo con docker y docker-compose sobre la Raspberry Pi puedes replicarlo en cualquier otro sistema operativo que permita su ejecución y con un hardware mas potente.

Revision #6
Created 2023-02-14 21:24:14 CET by Pablo Ruiz
Updated 2023-03-11 11:56:31 CET by Pablo Ruiz

Copyright 2026