

Installation

Die Installation ist Dank Docker schnell und einfach erledigt.

Am besten zuerst einen neuen Ordner anlegen, in welchen dann eine docker-compose.yml sowie eine docker-compose.env angelegt werden.

Zuerst folgende Docker Compose Konfiguration anpassen und speichern.

```
services:
  paperless_redis:
    image: redis
    container_name: paperless_redis
    restart: unless-stopped
    volumes:
      - /pfad/zu/paperless-ngx/redis_data:/data
  paperless_db:
    image: mariadb
    container_name: paperless_db
    restart: unless-stopped
    volumes:
      - /pfad/zu/paperless-ngx/db_data:/var/lib/mysql
    environment:
      MARIADB_HOST: paperless
      MARIADB_DATABASE: paperless
      MARIADB_USER: paperless
      MARIADB_PASSWORD: paperless
      MARIADB_ROOT_PASSWORD: paperless
  paperless_gutenberg:
    image: docker.io/gutenberg/gutenberg
    container_name: paperless_gutenberg
    restart: unless-stopped
    # The gutenberg chromium route is used to convert .eml files. We do not
    # want to allow external content like tracking pixels or even javascript.
    command:
      - "gutenberg"
      - "--chromium-disable-javascript=true"
      - "--chromium-allow-list=file:///tmp/*"
```

```

paperless_tika:
  image: ghcr.io/paperless-ngx/tika:latest
  container_name: paperless_tika
  restart: unless-stopped
paperless_web:
  image: ghcr.io/paperless-ngx/paperless-ngx:latest
  container_name: paperless_web
  restart: unless-stopped
  depends_on:
    - paperless_db
    - paperless_redis
    - paperless_gutenberg
    - paperless_tika
  ports:
    - "8000:8000"
  volumes:
    - /pfad/zu/paperless-ngx/data:/usr/src/paperless/data
    - /pfad/zu/paperless-ngx/media:/usr/src/paperless/media
    - /pfad/zu/paperless-ngx/export:/usr/src/paperless/export
    - /pfad/zu/paperless-ngx/consume:/usr/src/paperless/consume
  env_file: /pfad/zu/paperless-ngx/docker-compose.env
  environment:
    PAPERLESS_REDIS: redis://paperless_redis:6379
    PAPERLESS_DBENGINE: mariadb
    PAPERLESS_DBHOST: paperless_db
    PAPERLESS_DBUSER: paperless # only needed if non-default username
    PAPERLESS_DBPASS: paperless # only needed if non-default password
    PAPERLESS_DBPORT: 3306
    PAPERLESS_TIKA_ENABLED: 1
    PAPERLESS_TIKA_GUTENBERG_ENDPOINT: http://paperless_gutenberg:3000
    PAPERLESS_TIKA_ENDPOINT: http://paperless_tika:9998

```

Anstelle einer MariaDB kann auch PostgreSQL oder SQLite (würde ich nicht empfehlen) verwendet werden. Weitere Informationen hierzu können in der offiziellen Dokumentation nachgelesen werden: [GitHub Paperless-ngx](#)

Nun kann die docker-compose.env erstellt werden, welche folgenden Inhalt erhält. Bitte auch hier ggf. die Einstellungen nach Wunsch anpassen z. B. die Sprache für die OCR-Erkennung.

```

# The UID and GID of the user used to run paperless in the container. Set this
# to your UID and GID on the host so that you have write access to the

```

```
# consumption directory.
USERMAP_UID=1000
USERMAP_GID=1000

# Additional languages to install for text recognition, separated by a
# whitespace. Note that this is
# different from PAPERLESS_OCR_LANGUAGE (default=eng), which defines the
# language used for OCR.
# The container installs English, German, Italian, Spanish and French by
# default.
# See https://packages.debian.org/search?keywords=tesseract-ocr-&searchon=names&suite=buster
# for available languages.
PAPERLESS_OCR_LANGUAGES=deu eng

#####
# Paperless-specific settings                                     #
#####

# All settings defined in the paperless.conf.example can be used here. The
# Docker setup does not use the configuration file.
# A few commonly adjusted settings are provided below.

# This is required if you will be exposing Paperless-ngx on a public domain
# (if doing so please consider security measures such as reverse proxy)
#PAPERLESS_URL=https://paperless.example.com

# Adjust this key if you plan to make paperless available publicly. It should
# be a very long sequence of random characters. You don't need to remember it.
#PAPERLESS_SECRET_KEY=change-me

# Use this variable to set a timezone for the Paperless Docker containers. If not specified,
# defaults to UTC.
PAPERLESS_TIME_ZONE=Europe/Berlin

# The default language to use for OCR. Set this to the language most of your
# documents are written in.
PAPERLESS_OCR_LANGUAGE=deu

# Set if accessing paperless via a domain subpath e.g. https://domain.com/PATHPREFIX and using
# a reverse-proxy like traefik or nginx
```

```
#PAPERLESS_FORCE_SCRIPT_NAME=/PATHPREFIX
#PAPERLESS_STATIC_URL=/PATHPREFIX/static/ # trailing slash required
```

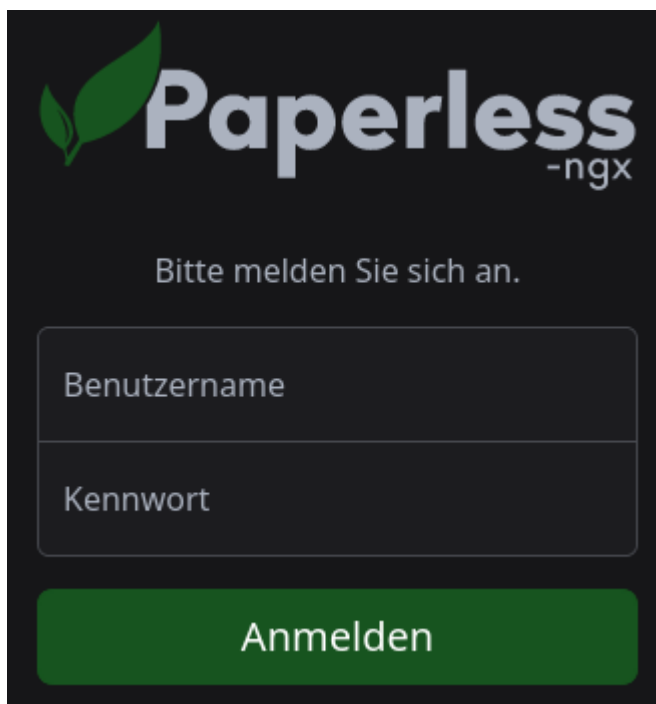
Nachdem die beiden Dateien erstellt und angepasst worden sind, können die Container erstellt und gestartet werden.

Um die Docker Compose Konfiguration auszuführen, kann am besten in das Verzeichnis der YAML Datei gewechselt werden. Danach wird je nach gewählter Installation `sudo docker-compose up -d` oder `sudo docker compose up -d` (keine Bindestrich zwischen docker und compose) eingegeben, um die Standard Konfiguration `docker-compose.yml` zu starten. Compose erstellt dann die gewünschten Container mit den angegebenen Optionen. Sollten die Container bereits mit dieser Compose Konfiguration erstellt worden sein, so werden die Container in dieser neu erstellt, dessen Konfiguration geändert wurde.

Der erste Start dauert mehrere Sekunden, da die Datenbanken, Verzeichnisse usw. automatisch initialisiert werden. Diese Zeit kann genutzt werden, um auf der Kommandozeile einen Superadmin zu erstellen. Hierfür bitte folgenden Befehl ausführen (ggf. den Namen des Containers anpassen).

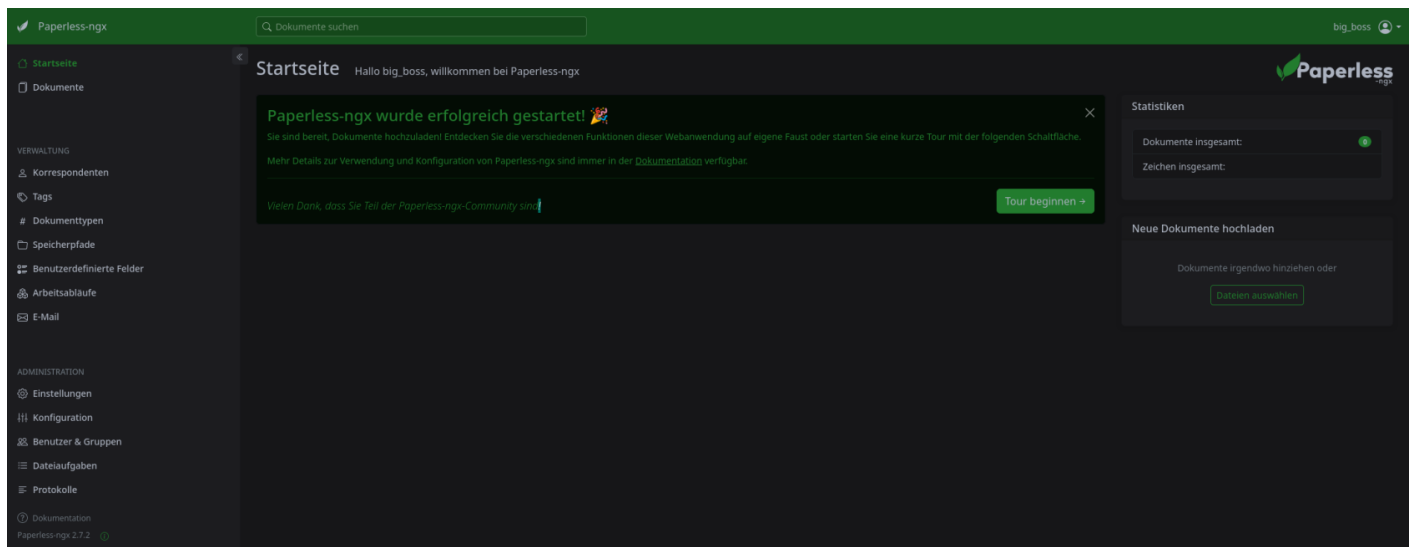
```
sudo docker compose exec -it paperless_web python3 manage.py createsuperuser
```

Anschließend steht die Weboberfläche unter <http://localhost:8000> zur Verfügung und der soeben erstellte Superuser kann zum Login verwendet werden.



Nach dem erfolgreichen Login wird eine Willkommensseite angezeigt.

Hier wird eine *Tour* vorgeschlagen, welche die wichtigsten Funktionen kurz zeigt und erklärt. Diese ist unbedingt zu empfehlen, wenn Paperless vorher noch nie verwendet wurde.



Version #2

Erstellt: 27 April 2024 20:11:03 von Marcel

Zuletzt aktualisiert: 29 April 2024 18:28:05 von Marcel