

Paperless-AI

Overview

PaperlessAI is an AI companion service for Paperless-NGX that automates document classification. It monitors the Paperless document inbox and uses a local Ollama model to automatically assign tags, document types, correspondents, and titles to newly ingested documents — eliminating the need for manual categorisation.

It communicates with Ollama over the internal `ai-internal` network and with Paperless-NGX via its API over the external HTTPS URL.

Access

Type	URL	Notes
Internal	<code>https://paperlessai.home.local</code>	Authentik SSO required
External	<code>https://paperlessai.jeeves5454.ddns.net</code>	Authentik SSO + GeoBlock + CrowdSec

Configuration

Image: `clusterzx/paperless-ai:latest` **Compose project:** Paperless stack (managed via Portainer, separate from `ai-stack`)

Environment Variables

Variable	Value / Notes
<code>LLM_PROVIDER</code>	<code>ollama</code>
<code>OLLAMA_URL</code>	<code>http://ollama.home.local:11434</code>
<code>OLLAMA_MODEL</code>	<code>gemma4:12b</code>
<code>PAPERLESS_URL</code>	<code>https://paperless.jeeves5454.ddns.net</code>

Variable	Value / Notes
PAPERLESS_API_KEY	REDACTED
AUTO_MODE	true — processes documents automatically
AUTO_TAG	ai-processed — applied to every AI-classified document
NODE_ENV	production

AUTO_MODE=true means PaperlessAI polls the Paperless inbox on a schedule and processes new documents without any manual trigger. The ai-processed tag is applied to documents after classification, allowing easy filtering in Paperless.

Traefik Labels

```
# External route
traefik.http.routers.paperlessai-ext.rule: Host(`paperlessai.jeeves5454.ddns.net`)
traefik.http.routers.paperlessai-ext.entrypoints: websecure
traefik.http.routers.paperlessai-ext.tls.certresolver: letsencrypt
traefik.http.routers.paperlessai-ext.middlewares: plex-geoblock@file,crowdsec-bouncer@file,authentik-auth@docker
traefik.http.routers.paperlessai-ext.service: paperlessai-svc

# Internal route
traefik.http.routers.paperlessai-int.rule: Host(`paperlessai.home.local`)
traefik.http.routers.paperlessai-int.entrypoints: websecure
traefik.http.routers.paperlessai-int.tls.certresolver: step-ca
traefik.http.routers.paperlessai-int.middlewares: authentik-auth@docker
traefik.http.routers.paperlessai-int.service: paperlessai-svc

traefik.http.services.paperlessai-svc.loadbalancer.server.port: 3000
```

Note that Authentik middleware is applied on **both** internal and external routes.

Volumes / Bind Mounts

Host Path	Container Path	Purpose
/home/jeeves/docker/paperless/paperless-ai-data	/app/data	PaperlessAI configuration and state

Networks

PaperlessAI must be on the same network as Ollama to use `http://ollama.home.local:11434`. It reaches Paperless-NGX via its external HTTPS URL (outbound through Traefik).

Dependencies

- **Ollama** — must be running with the `gemma4:12b` model loaded for classification to work. Pull the model first:

```
docker exec -it ollama ollama pull gemma4:12b
```

- **Paperless-NGX** — must be accessible at `https://paperless.jeeves5454.ddns.net` with a valid API key
- Authentik for SSO on both routes

Notes / Gotchas

- If Ollama is down or the model isn't loaded, documents will queue and not be processed until Ollama is available again.
- The `AUTO_TAG` (`ai-processed`) helps track which documents were classified by AI vs manually. Review newly tagged documents periodically to validate AI accuracy.
- `gemma4:12b` (12 billion parameters) fits comfortably within the RTX 5080's 16 GB VRAM. If Ollama is under pressure from simultaneous inference requests, PaperlessAI classification may be slower.
- PaperlessAI connects to Paperless via the external URL (`https://paperless.jeeves5454.ddns.net`). Ensure this domain remains reachable from within the container network.

Last Updated: 2026-06-16

Revision #3

Created 2026-06-17 04:43:13 UTC by Admin

Updated 2026-06-17 16:55:02 UTC by Admin