Deployment & Operations Guide

Museum AR Smart Glass Guide — on-premises

HumanityAI'd — July 2026

CONFIDENTIAL — FOR MUSEUM IT & OPERATIONS

1. Deployment model

Everything runs on **one GPU server inside the museum network**. No component depends on the public internet after installation. Services are containerized (Docker) and start as a unit: FastAPI backend, PostgreSQL, Redis, nginx, Prometheus, Grafana, and the admin frontend.

2. Hardware sizing

Component	Minimum	Recommended	Notes
GPU	8 GB (recognition serving only)	**16 GB**	8 GB proven for serving; 16 GB gives head-room to generate narration on the same box without contention
System RAM	16 GB	32 GB	Postgres + Redis + services
CPU	6 cores	8+ cores	Recognition offload, ffmpeg
Storage	256 GB SSD	512 GB SSD	Images, FAISS index, audio, DB
Network	1 GbE, dedicated gallery SSID	1 GbE + ≥2 APs	Static server IP

GPU memory budget (measured): CLIP ViT-L/14 ≈ 2.8 GB; multilingual TTS model ≈ 3.4 GB. On an 8 GB card shared with other workloads these do not co-reside comfortably — hence narration generation is treated as an **offline/admin operation**, and 16 GB is recommended where the same box also generates audio.

3. Persistent data (must be backed up)

All state lives in named volumes so it survives container restarts and upgrades:

Data	Volume path (in container)	Contents
Database	Postgres volume	Exhibits, translations, sessions, analytics
Images	`/backend/app/static/images`	Enrolled exhibit photos
FAISS index	`/backend/app/database/faiss_indexes`	Visual recognition index + position map
Audio	`/backend/app/static/audio`	Generated narration (Opus)

Data	Volume path (in container)	Contents
Model cache	model-cache volume	CLIP/TTS weights (avoids re-download)

> Note: application paths derive from `/backend/app`. Volume mounts must target `/backend/app/...` (not `/backend/...`) or data is written to ephemeral container storage and lost on restart.

4. Install (outline)

1. Install Docker + NVIDIA container runtime; set nvidia as default runtime.
2. Provision secrets at install time (DB password, admin session secret, API keys) — never ship defaults.
3. Create the named volumes; place TLS certs.
4. Bring up the stack; run DB migrations (automatic at startup).
5. Verify `/health/ready` reports all components green; confirm GPU is detected.
6. Set the server's static IP as the glasses' API base URL.

5. Operating runbook

Enrol / update content — via the admin panel (see POC Playbook §6).

Regenerate narration (e.g., edited script): trigger generation for the exhibit; approve the new clip. On an 8 GB shared GPU, generate during off-peak hours. If recognition and TTS contend for memory, temporarily run recognition on CPU during a bulk regeneration, then restore GPU.

Backups: nightly snapshot of the Postgres volume + the images/audio/FAISS volumes. FAISS can be rebuilt from the database if the index is ever lost.

Monitoring: Grafana dashboards for request rate, latency, and GPU; alert on `/health/ready` failures and unhealthy containers.

Upgrades: image-only updates roll the affected service; environment/secret changes require a full stack re-deploy (env is baked at deploy time). Data volumes are external and are never touched by an upgrade.

6. Capacity & scaling

- One server ≈ one museum. A single 8 GB GPU host serves recognition for an estimated 25–40 concurrent active sessions.
- Scale-out path (roadmap v1.5): split the stateless API from a dedicated inference service so multiple GPU nodes serve one large site or a museum group.

7. Security & privacy posture

- No visitor imagery leaves the building; sessions are pseudonymous; a GDPR-style purge endpoint exists.
- Production hardening (JWT/RBAC, TLS + pinning on glasses, per-device tokens) is scheduled — see the Technical Due-Diligence Report. Until then, treat the deployment as a trusted-LAN pilot: isolate the gallery SSID, restrict the admin panel to staff VLAN.

8. Support

Tiered SLA (see Pricing & Packaging): response targets, remote diagnostics via the metrics stack, spare-glasses ratio, and optional managed content services.