Marqib — Architecture and data flow

Version 1.0 · September 2026 · Provided to prospective pilot customers for vendor due diligence. Items marked **(planned)** are not built yet; everything else describes the product as it runs today.

1. What Marqib does

Marqib reads a regulated firm's end-of-day data (clients, trades, orders, transactions, logins, bank-detail changes). It runs deterministic trade-surveillance and transaction-monitoring rules over the whole book and turns each alert into a case file for the compliance team:

- the reasons, with the evidence rows each reason rests on;
- linked parties;
- a regulatory deadline clock;
- a recommendation;
- a draft STR narrative and goAML XML.

A named analyst proposes a decision and a named MLRO approves or returns it. Nothing is filed automatically.

2. Components

Component	Technology	Role
Application	Next.js (Node.js) web application	Screens, API, rule engine, case workflow
Database	PostgreSQL	Customer data tables, cases, append-only audit log
Rule engine	SQL + TypeScript, thresholds in one configuration file	Deterministic detection
Narrative layer (optional)	Pluggable HTTP client to a model endpoint the firm chooses	Rewrites reasons and STR draft from collected evidence
Exports	Server-side rendering	Evidence pack (with SHA-256), goAML XML, MI report

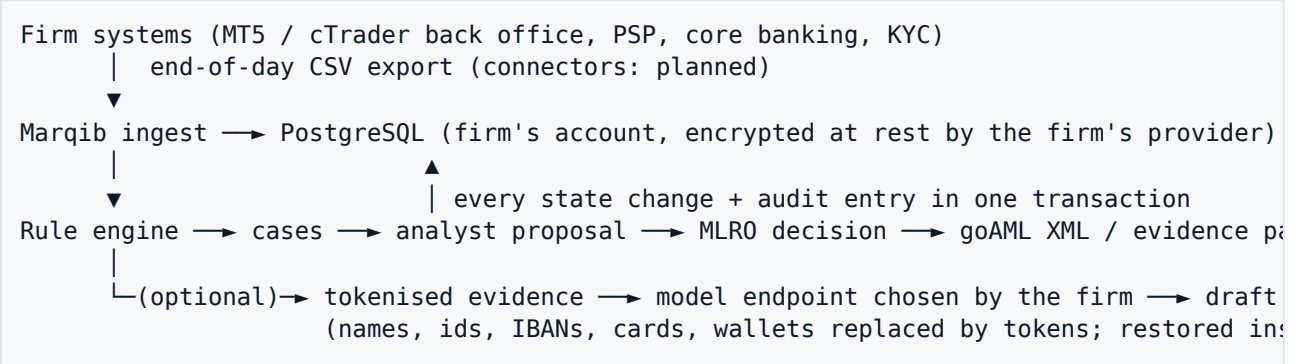
There are no background agents with outbound access, no telemetry to the vendor, and no third-party scripts inside the application.

3. Deployment models

1. **Firm-hosted (production target).** The application and database run in the firm's own cloud account (AWS, Azure or Google Cloud, in the region the firm chooses) or on its own servers. Customer data never leaves that environment. Container packaging for this model is **(planned)**: a Docker Compose / container image with migration scripts, delivered with the first production licence.

2. **Dedicated pilot instance.** An instance run by Marqib for a single firm, in the firm's region. It holds only that firm's data and is deleted at the end of the pilot unless a licence follows. Deletion is confirmed in writing.
3. **Public sandbox (marqib.com).** This is not for customer production data. It runs synthetic books by default. Each visitor gets a separate database schema, deleted after 7 days. If a visitor uploads their own files, identifiers are tokenised in the browser before upload (see section 6).

4. Data flow (firm-hosted)



5. Data categories processed

- **Client master:** internal id, name, type, country, beneficial-owner reference, occupation, expected turnover, onboarding date.
- **Trades and orders:** id, client id, time, instrument, side, quantity, price, status, introducing partner.
- **Transactions:** id, client id, time, kind, amount, currency, channel, counterparty name and reference (IBAN, card or wallet), status.
- **Optional:**
  - Logins: device id, IP, country, success.
  - Bank-detail changes.
- **Generated:** cases, reasons, decisions, notes, audit log, reports.

No special-category data is required. Identity documents and images are never ingested.

6. Tokenisation

- **Narrative layer.** Before any call to a model, client names, client ids, IBANs, card numbers and wallet addresses are replaced by tokens. The mapping stays in the application's memory for the request and is never sent to the model. The audit log records the provider, model, prompt hash and number of tokens replaced.
- **Sandbox upload.** The visitor's browser replaces the following with salted SHA-256 tokens before anything is uploaded:
  - names, word by word;
  - client and UBO ids;
  - IBAN, card and wallet references;
  - device ids;

- IP addresses.

The salt stays in the visitor's browser. Marqib never receives the original values.

7. Integrity and audit

- An append-only audit log records every action with actor, time and detail. Each entry carries the hash of the previous one, and the chain is re-verified when the log is viewed.
- A case change and its audit entry are committed in the same database transaction. Decisions lock the case row, so two reviewers cannot decide the same case concurrently.
- Each evidence pack carries a SHA-256 of the case and its trail.

8. Access control

- Named users with roles `analyst` and `mlro`. Maker-checker is enforced server-side: nobody approves their own proposal.
- Sessions are HMAC-signed cookies (HttpOnly, SameSite) that expire after 12 hours.
- Sign-in attempts are rate-limited, and cross-site state-changing requests are refused.
- **(planned)** SAML/OIDC single sign-on, MFA, server-side session revocation.

9. Network

- **Firm-hosted:** inbound HTTPS from the firm's own network only (placement behind the firm's VPN or identity-aware proxy is recommended). Outbound traffic goes only to the model endpoint, if the firm enables one.
- **Hosted services:** TLS 1.2+ everywhere, served from Vercel with the database on Neon PostgreSQL (see the sub-processor list).