

EzyGas – Full Integrated App Manual (CTO Edition)

Confidential • Internal Engineering Reference • Version 1.0 • 2026-02-08

This manual defines the product requirements, system capabilities, security posture, scalability roadmap, and enforceable state logic for the EzyGas composite-cylinder ecosystem. It is the authoritative implementation blueprint for engineering, product, and operations.

Executive Summary

EzyGas is an asset-governed distribution network, not a courier app. The system's defensible core is **Composite Cylinder Lifecycle Intelligence**: each cylinder is a uniquely identifiable, stateful asset with compliance metadata, custody trails, fraud controls, and safety gates. B2C and B2B experiences are built on top of that asset layer, while the SLA engine converts delivery promises into quantifiable operational risk and automated escalation.

Table of Contents

- 1 1. Product Scope & Platform Surfaces
- 2 2. Roles, Identity & Access Control (RBAC)
- 3 3. Service Catalog & Commercial Logic (Home vs Premium)
- 4 4. Orders & SLA Engine (SLA as Code)
- 5 5. Composite Cylinder Intelligence (Lifecycle + Compliance)
- 6 6. Incoming Asset Validation (Swap Fraud Controls)
- 7 7. Driver App Offline-First Architecture & Scan Velocity
- 8 8. Inventory, Hubs & Chain-of-Custody
- 9 9. Dispatch, Routing, Zones & Telemetry
- 10 10. Payments, COD Reconciliation & Wallet Limits
- 11 11. Support, CRM (Zoho) & Background Sync
- 12 12. Data Model (Canonical Entities + Event Ledger)
- 13 13. Security Baseline & Auditability
- 14 14. Observability (KPIs, Logs, Tracing, Alerts)
- 15 15. Scalability Roadmap (Pilot → Governorate → Nation)
- 16 16. Acceptance Criteria & Definition of Done
- 17 Appendix A: Cylinder Lifecycle State Machine (Strict Transition Rules)
- 18 Appendix B: Offline Outbox & Idempotency Contract
- 19 Appendix C: API Domain Map (High-Level)

1. Product Scope & Platform Surfaces

System objective: deliver a governed composite LPG cylinder ecosystem with traceability, SLA reliability, and compliance-grade audit trails.

Platform surfaces:

- **B2C Consumer App** (Home / Premium) – order, track, pay, support, safety education.
- **B2B Portal** (Commercial) – multi-site, roles, approvals, standing orders, invoicing, compliance reports.
- **Driver App** – offline-first scanning, batch workflows, PoD, cash handling, incidents, safety checklists.
- **Admin/Ops Console** – dispatch, zoning, SLA monitoring, inventory, cylinder locks/maintenance, refunds/credits, analytics.

2. Roles, Identity & Access Control (RBAC)

Core roles: consumer, driver, admin, ops, hub, filling, system; plus B2B roles: site_user, site_manager, finance, ops_viewer.

RBAC requirements:

- All permissions are enforced **server-side**; clients are untrusted.
- High-risk actions require audit logs: pricing edits, cylinder locks, refunds, order overrides, KYC approvals.
- Sensitive data (phone, national ID, addresses) must be encrypted-at-rest and access-logged.

3. Service Catalog & Commercial Logic (Home vs Premium)

Catalog engine: a single definition powering B2C and B2B pricing, eligibility, and SLA promises.

Services: Swap/Exchange (default), New Cylinder acquisition/lease, Scheduled replenishment (B2B), Priority delivery (Premium).

Tiers:

- **Home:** standard SLA, basic support, mass-market pricing.
- **Premium:** priority dispatch, preferred time windows, proactive reminders, priority support, breach compensation rules.

4. Orders & SLA Engine (SLA as Code)

Order state machine: draft → placed → confirmed → assigned → picked_up → in_transit → arrived → delivered → closed; exceptions: cancelled, failed, rescheduled, returned, refunded.

SLA as a product primitive:

- Each order stores a computed **sla_target_at** timestamp.
- A continuous **risk_score** estimates breach probability (ETA vs SLA).
- Escalations: Risk > 70% notify dispatcher; Risk > 90% attempt reassignment; breach triggers auto-ticket + configured credit/refund.
- SLA analytics must segment by zone/hub/driver/time window with root-cause tags (stockout, traffic, customer delay).

5. Composite Cylinder Intelligence (Lifecycle + Compliance)

Cylinder is the core asset: every composite cylinder has a unique identity (QR mandatory; RFID roadmap), compliance metadata, custody trails, and enforced states.

Mandatory fields: qrCode, size, batchId, status, certifiedUntil, lastInspection, cycleCount, tareWeightNominal, tareWeightActual (last known).

Fraud & safety controls: duplicate scan detection, geo/time anomaly detection, locked status enforcement, expired certification forcing inspection.

6. Incoming Asset Validation (Swap Fraud Controls)

Problem: customer attempts to swap steel/competitor cylinder for an EzyGas composite cylinder.

Requirement: model the returned physical object as an **Incoming Asset** that must be validated before swap completion.

Workflow:

- 1) Driver scans returned cylinder QR (if present).
- 2) Driver sets asset type: ezygas_composite / competitor_composite / steel / unknown.
- 3) System policy decides: accept, accept_with_deposit, reject, or escalate (ops override with audit).
- 4) If non-EzyGas asset and customer expects EzyGas composite output → deposit required or block.

7. Driver App Offline-First Architecture & Scan Velocity

Offline-first is mandatory: basements, elevators, and remote compounds have zero signal. Driver workflows must continue uninterrupted.

Local database: SQLite/WatermelonDB; all scans and PoD events are written to a local **outbox** immediately.

Batch scan mode: scan-scan-scan → submit; no per-scan API waits. UI requires Flash toggle + explicit Zoom + low-light optimization.

Local blacklist cache: stolen/unsafe/expired cylinders must be cached locally to allow red-alert validation while offline.

Sync contract: background sync retries with exponential backoff. Every event includes idempotency keys to prevent duplicates.

8. Inventory, Hubs & Chain-of-Custody

Inventory nodes: hub/warehouse, distributor depot, vehicles (mobile nodes), B2B sites, customer custody.

Ledger-based custody: transfers require scans and generate immutable events (hub→vehicle→customer→return).

Reconciliation: daily hub audits, shrinkage reporting, mismatch exception queue, and forced investigation for repeated anomalies.

9. Dispatch, Routing, Zones & Telemetry

Zoning: polygon zones mapped to hubs; drivers assigned to zones with capacity constraints (stops/weight/cylinder mix).

Dispatch modes: manual, recommended, auto (phase 3).

Telemetry: driver location pings (battery-friendly cadence); geofence events: arrived/picked_up/delivered/return.

Cost note: maps APIs can be expensive. Roadmap supports Mapbox for UI styling; OSRM/OpenStreetMap routing later for hosted routing economics.

10. Payments, COD Reconciliation & Wallet Limits

Payment methods (phased): COD first (with strict controls), then online/wallet, then B2B invoicing/credit limits.

COD risk controls: implement cash ledger (append-only). Do not store mutable balances without ledger source-of-truth.

Wallet limit enforcement: if driver cash > configured threshold (e.g., 5,000 EGP) → block acceptance of new dispatch until hub deposit; allow current job completion; show ops dashboard cash-risk alerts.

11. Support, CRM (Zoho) & Background Sync

Tickets: auto-generated for SLA breaches, failed deliveries, safety incidents, payment disputes.

Zoho integration: asynchronous only. Use job queue (e.g., BullMQ + Redis) with retries and dead-letter handling. Zoho downtime must not impact ordering, scanning, dispatch, or delivery.

Customer success: NPS/CSAT capture; complaint resolution time tracked as KPI.

12. Data Model (Canonical Entities + Event Ledger)

Canonical entities: users, user_profiles, drivers, cylinders, orders, deliveries, inventory_nodes, inventory_ledger, payments/cash_ledger, tickets, audit_logs.

Append-only events table: (event_id, entity_type, entity_id, event_type, payload JSONB, actor_id, actor_role, occurred_at, received_at, geo).

Why events: auditability, regulator exports, anomaly detection, and scalable read-model generation for dashboards.

13. Security Baseline & Auditability

Baseline: OWASP-aligned; rate limits on auth/OTP/order/scan endpoints; secure sessions; server-side RBAC; encryption at rest for PII.

Audit scope: pricing edits, refunds, cylinder locks/unlocks, order overrides, KYC approvals, cash reconciliation actions.

Fraud detection: geo/time anomalies, duplicate scan attempts, repeated “unknown incoming asset” patterns, driver route anomalies.

14. Observability (KPIs, Logs, Tracing, Alerts)

Logs: structured JSON with correlation IDs across request chains.

Metrics: on-time rate, breach rate, ETA error, scan failure rate, stockout rate, cash risk, ticket volume, resolution time.

Tracing: order placement → dispatch → delivery; SLA risk recalculation pipeline.

Alerts: SLA breach spike, hub stockout, scan anomaly spike, sync backlog growth, Zoho job failures.

15. Scalability Roadmap (Pilot → Governorate → Nation)

Phase 1 (Pilot): modular monolith with domain modules + event ledger + background jobs (SLA monitor, CRM sync, reconciliation).

Phase 2 (Multi-governorate): carve out dispatch/telemetry/billing services as hotspots; introduce message bus if event volume demands it.

Phase 3 (National, 1M+ cylinders): event-driven backbone with partitioned data by region/hub; materialized views for ops dashboards; streamed anomaly detection.

16. Acceptance Criteria & Definition of Done

Definition of Done: secure + auditable + observable + SLA-aware + resilient under edge cases.

Edge cases: customer no-show, wrong pin, cylinder locked/maintenance, scan mismatch, partial B2B delivery, offline delivery completion with later sync.

Resilience: graceful degradation (maps down, CRM down, payment down), DB backups + restore tests.

Appendix A: Cylinder Lifecycle State Machine (Strict Transition Rules)

The cylinder lifecycle is the enforceable heart of EzyGas. Transitions are restricted by role, compliance validity, custody requirements, and explicit proofs (scan + PoD).

From	To	Event	Allowed Roles	Hard Requirements
maintenance	inspection	START_INSPECTION	hub/admin	physical possession at node; audit logged
inspection	empty	PASS_INSPECTION	hub/admin	inspection record; certifiedUntil optional update
empty	filled	FILL	filling/admin/system	filling node; increments cycle_count; optional tare_weight
filled	reserved	RESERVE_FOR_ORDER	ops/admin/system	orderId required; prevents double assignment
reserved	in_transit	LOAD_FOR_DELIVERY	hub/ops/admin	orderId must match; custody to driver node
in_transit	delivered	DELIVER	driver/admin/system	PoD reference required; orderId match; scan validated
delivered	returned	RETURN	driver/hub/admin	custody event; return nodeId required
*	locked	LOCK	admin/system	reason required; blocks all trade states
locked	inspection	UNLOCK_TO_INSPECTION	admin	audit + nodeId; clears lock reason

Reference Implementation Notes (XState)

- The server must be the final authority for transitions.
- Driver app can execute transitions offline, but they are stored as outbox events.
- Server validates each outbox event against the state machine, with idempotency checks.
- Illegal transitions are rejected and surfaced as exceptions for ops review.

Appendix B: Offline Outbox & Idempotency Contract

Objective: allow drivers to scan and complete deliveries with zero connectivity while preserving auditability and correctness.

Local tables: outbox_events, local_blacklist_cache, local_jobs, local_media_queue (PoD photos).

Outbox event fields: event_id (UUID), device_id, actor_id, actor_role, entity_type, entity_id, event_type, payload, occurred_at, geo, synced_at (nullable).

Idempotency rules: server treats event_id as unique. Replays are safe. Events are processed in occurred_at order per cylinderId; conflicts create a reconciliation ticket.

Local blacklist cache: sync a compact dataset keyed by qrCode/hash with status=locked/decommissioned/expired. Validation must be O(1) and work offline.

Appendix C: API Domain Map (High-Level)

Consumer

- Profile & addresses
- Catalog/pricing quote
- Place order
- Live tracking
- History/invoices
- Support/tickets

Driver

- Availability
- Dispatch jobs
- Location pings
- Batch scans
- PoD submission
- Cash ledger updates
- Incident reporting

Ops/Admin

- Order queue + SLA dashboard
- Dispatch assign/override
- Zones/hubs
- Inventory transfers
- Cylinder lock/maintenance
- Refund/credit approvals
- Analytics exports

System

- Event ingestion
- SLA risk scoring jobs
- Zoho sync jobs
- Anomaly detection jobs
- Notifications (SMS/Push/WhatsApp roadmap)