WATERBIZ ERP · ARCHITECTURE v2

Mixed Fleet SaaS Architecture Guide

Company-owned vans and independent contractors inside one tenant

Official company accounting separated from internal contractor management

---

Revision notice. This guide replaces the two-model (company OR commission) design. Fleet treatment is now configured **per delivery team**, not per tenant. Every statement below was verified against the Laravel codebase; items marked EXISTS are already implemented, PARTIAL need extension, and MISSING are new build.

Generated 26 Nov 2026, 20:25 +04 · v2.0

1. Target architecture

One Laravel codebase, one database per tenant, and a shared operational engine. Inside a tenant, each delivery team carries its own commercial operating type, so a single water company can run owned vans and contractors side by side.

SaaS platform database tenants · domains · plans · licences · entitlements · audit TENANT DATABASE — one per water company Official accounting Admin / Accounts GL · VAT · AR · AP · payroll Internal operations Management / Supervisor vouchers · ledger · SOA Shared operations loading · orders · POS wallet · coupons · custody delivery_teams.operating_type per team company · contractor · company_managed Company fleet to official books Contractor fleet to contractor ledger

> **Governing principle**
>
> A vehicle's ownership never decides accounting. A company-owned van operated by a contractor still posts to the contractor sub-ledger. Classification lives on the `delivery_teams` record and is snapshotted onto documents.

Tenant isolation requirements

Surface	Requirement	Status
Database	One connection per tenant, no cross-tenant joins	**Missing**
Domain	Tenant resolved from verified host; client-supplied tenant id never trusted	**Missing**
API authentication	JWT scoped to the tenant that issued it	**Partial**
Users and permissions	Per-tenant staff list; no shared login	**Exists**
Storage and cache	Tenant-prefixed paths and cache keys	**Missing**
Queues and reports	Job payloads carry tenant context; exports scoped	**Missing**
Licence gate	Grace period with offline validation	**Partial**

2. Accounting boundary

Two accounting environments share one tenant database but never merge balances.

Official company accounting	Internal contractor management
Portals: /admin, /accounts	**Portals:** /management, /supervisor, /admin/contracts
General ledger and chart of accounts	Team issue vouchers
Customer invoices and VAT	Receiving and return documents
Supplier invoices and payments	Team stock balances
Payroll and employee costs	Collections and remittances
Fixed assets and depreciation	Charges and deductions
Loans, bank reconciliation, period lock	Contractor statement of account
Statutory reports, VAT Form 201	Team-wise performance

Existing control accounts EXISTS

Code	Account	Purpose
`1005`	Contractor Cash	Contractor-domain cash, separate from company bank/cash
`1150`	Contractor / Team Receivable	What teams owe the company
`2400`	Contractor Settlement Clearing	In-flight settlement between team and company

Boundary rules

Rule	Enforcement today
Issue voucher is an internal stock record, not company revenue	**No revenue journal created**
Contractor cash never mixes with company cash	**GL 1005 and contractor cash account**
Team expenses excluded from company profit and loss	**Filters on** `assigned_to_type`
No second customer invoice for contractor accounting	**Reports read existing invoices**
Team operating type frozen on each document	**Needs snapshot column**
Consolidated contractor accounting screen	**Service exists, route does not**

Audit finding

`ContractorAccountingService::dashboardData()` already computes the four control-account cards and per-team receivable balances, but no route or controller resolves it. The contractor accounting screen is therefore built but not reachable.

3. Fleet operating types

Replace the tenant-wide business model with a classification on each delivery team. A tenant may hold any mix of the three types at the same time.

Operating type	Typical arrangement	Accounting treatment
company	Company vehicle, salaried driver	Company operating expenses; no contractor ledger
contractor	Independent van operator	Contractor sub-ledger, vouchers, charges, statement
company_managed	Company-controlled delivery operation	Company expenses; contractor ledger not applicable

Illustrative mixed fleet

Van	Ownership	Operated by	Operating type	Ledger
Van 01	Company	Salaried driver	company	Company books
Van 02	Contractor	Contractor	contractor	Contractor ledger
Van 03	Company	Contractor	contractor	Contractor ledger
Van 04	Company	Salaried driver	company	Company books

Data model changes

Field	Where	Purpose	Status
`operating_type`	delivery_teams	Commercial treatment of the team	**Missing**
`settlement_enabled`	delivery_teams	Show contractor ledger and statement actions	**Missing**
`operating_type_snapshot`	vouchers and receipts	Preserve historical treatment after reclassification	**Missing**
`ownership_type`	vehicle record	Company, contractor or leased	**Missing**
`asset_id / vehicle_num`	delivery_teams	Current vehicle link	**Exists**
`delivery_type_id`	delivery_teams	Generic lookup - do not reuse as fleet switch	**Ambiguous**

> **Do not overload existing lookups**
>
> The `delivery_types` table is used both for *Company Van / Commissioned Van* in demo data and for *Bottles / Tankers* in other seeders. Introducing `operating_type` avoids reinterpreting production rows.

Reclassification rules

- Changing a team's operating type takes effect from a stated date.
- An open contractor balance must be settled or explicitly carried forward.
- Existing vouchers, receipts and statements keep their original treatment.
- Driver app capabilities follow the team's current type; history stays readable.

4. Contractor calculations

Three separate outputs, each with its own source of truth. They must never be collapsed into one number.

A. Stock ledger (quantity)

Opening stock + issued - returned - sold or consumed ± adjustment = closing stock. Already available through product ledger reporting built on issue vouchers and unloading returns.

B. Financial ledger (money owed)

Entry	Effect on team balance	Source
Opening balance	Carried forward	Prior statement
Stock issued	Debit	Issue voucher
Stock returned	Credit	Unloading return
Stock payment	Credit	Stock payment receipt
Charge	Debit	Non-stock charge
Approved adjustment	Debit or credit	Adjustment entry

C. Earnings report (operational performance)

Retail sales - wholesale stock cost - operating charges = contractor earnings. This explains how the contractor performed; it is not the settlement balance.

Settlement balance ≠ Earnings (two different reports, two different purposes)

Worked example

Item	Quantity	Rate	Amount
Stock issued to team	200	3.00	600.00
Retail sales	200	5.00	1,000.00
Gross difference			400.00
Operating charges			50.00
Contractor earnings			**350.00**
Balance payable to company			**600.00** less payments received

Illustrative only: assumes all stock sold, no returns, no wallet or coupon settlement.

5. Module matrix by operating type

The modules are shared. What differs is the accounting treatment and the screens a team may open.

Module	Company fleet	Contractor fleet
Loading sheets	**Yes**	**Yes**
Issue vouchers	Internal stock record	Contractor stock debit
Receiving / returns	Internal stock record	Contractor stock credit
Customer orders and delivery	**Yes**	**Yes**
Customer invoices and VAT	**Yes**	**Yes**
Delivery driver app	**Yes**	**Yes**
Wallet and coupon books	**Yes**	**Yes**
Bottle and dispenser custody	**Yes**	**Yes**
Company GL and statutory reports	**Yes**	**Yes**
Company expenses	**Yes**	Not as contractor charges
Team charges and deductions	Not applicable	**Yes**
Stock payments and collections	Not applicable	**Yes**
Contractor statement of account	Not applicable	**Yes**
Fleet performance reports	**Yes**	**Yes**

Portals and their scope

Portal	Scope	Fleet awareness
/admin	Full ERP, official accounting plus contractor screens	Shows both; actions gated per team
/accounts	Official accounting only	No contractor settlement actions
/management	Internal operations: vouchers, receipts, charges, statements	Contractor teams only for settlement
/cashier	POS and counter sales	Company sales
/delivery	Driver workspace	Capabilities follow team type
/supervisor	Sales group oversight, read only	Contractor settlement visibility

6. SaaS foundation status

Capability	Current state	Required work
Single codebase	**Exists**	None
Platform database	**Exists (SQLite)**	Move to production-grade store; add entitlements
Company and plan records	**Exists**	Link plans to feature entitlements
Licence verification	**Middleware exists, not applied**	Apply to tenant routes; add grace policy
Feature gating middleware	**Exists, unused**	Repurpose for entitlements, not fleet type
Tenant database isolation	**Missing**	Connection resolver, no tenant id column
Domain to tenant resolution	**Missing**	Verified host lookup, cached
Provisioning	**Env bundle download only**	Create database, migrate, seed, issue licence
Backup and restore per tenant	**Missing**	Per-tenant dump and restore runbook
Platform audit log	**Missing**	Provisioning, plan changes, impersonation

Configuration after revision

```
APP_MODE=tenant_instance
TENANT_ID=tenant_001
DB_DATABASE=waterbiz_tenant_001
FLEET_MODEL=hybrid
SAAS_LICENSE_KEY=...
```

The fleet model is chosen at `/install` and can be changed in `Admin → Settings → Fleet Model` or with `php artisan noor:fleet`. All three write the same database key, so they cannot drift.

`APP_BUSINESS_MODEL` no longer selects the fleet model. It remains only as a legacy mirror for tenants that have no per-team classification yet.

Additional requirements before commercial launch

Requirement	Detail
Tenant customisation	Products, prices, zones, VAT settings, currency, operating rules
Feature subscriptions	Accounting, inventory, contractor management, HR sold as entitlements
Data ownership	Authorised export and backup on request
Audit history	Vouchers, stock adjustments, user actions, configuration changes
Release management	Controlled migrations and rollback across tenants
Licence resilience	Grace period so a licence service outage never locks a tenant out

7. Implementation order

Phase	Scope	Runtime change
0	Gap analysis, corrected architecture and docs approval	**None**
1	Install switch (Company Owned / Hybrid), settings switch, route gating, `noor:fleet`	**Shipped**
2	SaaS foundation: tenant store, connection resolver, provisioning, entitlements, licence gate	**Next**
3	Mixed fleet: operating type per team, vehicle ownership, history snapshots	**Yes**
4	Contractor accounting: expose the existing service, receiving voucher flow, stock reconciliation report	**Yes**
5	Accounting boundary: posting policy, duplicate-posting tests, team expense isolation checks	**Yes**
6	Mobile: capability flags per team in driver and supervisor APIs	**Yes**
7	Isolation and financial reconciliation testing, licence expiry, backup restore	**Yes**

8. Open decisions

Decision	Options	Recommendation
Vehicle master	Extend assets, or new vehicles table with assignment history	Separate vehicles table; assets stay the financial register
Delivery types lookup	Reuse as fleet type, or add operating type	Add operating type; leave lookup untouched
Earnings report cadence	Per month, or per voucher cycle	Per month, aligned to statement period
Type change policy	Any date with settlement cutoff, or period boundary only	Any date with explicit cut-off and carry forward
Tenant isolation	Database per tenant, or shared database with tenant scopes	Database per tenant for financial separation
Company-managed delivery	Stock documents, or expense tracking only	Stock documents, no contractor ledger

Do not do

- Rename existing tables such as issue vouchers; it breaks reports, APIs and history.
- Create a second customer invoice for contractor accounting.
- Gate the contractor ledger by a tenant-wide flag instead of the team's operating type.
- Rewrite historical vouchers when a team is reclassified.
- Authorise runtime changes before this document is approved.