**Core Databases — Enterprise Console Module Specification**

Supply & Value Chain–First Architecture • Laravel (Inertia + React + Tailwind)

Generated: September 30, 2025 08:14 AM

# Overview

Purpose  
A centralized catalog of enterprises and their managed data systems. It enables:  
1) Enterprise Directory — enumerate companies by Value/Supply‑Chain Stage (and optional Governance lens), with quick access to the company's default database.  
2) Company & System Provisioning — create the company record, choose the stage, enable application modules (procurement, accounting, sales, inventory, documents, compliance, etc.), and dispatch a Provisioner Job to stand up the tenant.  
3) Development Pipeline — track status, owner, ETA, progress %, milestones, and a log of notes/risks/blockers/decisions.

Scope  
This document defines: submodules & functions, API routes, and the PostgreSQL schema (tables, fields, indices, constraints) to integrate with Laravel 11 + Inertia/React.

# Submodules & Functions

## A) Enterprise Directory

List enterprises; filter by Value/Supply‑Chain Stage (1..8) and Governance lens; search by name/identifier; open default DB.

Key UI Actions:

• Filter by stage / governance lens

• Search

• Open Database

• View Company Profile (contacts, identifiers, locations)

Primary Tables:

• core.value\_chain\_stages

• core.governance\_lenses

• core.enterprises

• core.enterprise\_governance\_tags

• core.enterprise\_identifiers

• core.enterprise\_contacts

• core.enterprise\_locations

• core.enterprise\_db\_bindings

• core.vw\_enterprise\_default\_db (view)

Routes:

• GET /core/enterprises

• GET /core/enterprises/{id}

• GET /core/enterprises/{id}/open-database

## B) Company & System Provisioning

Create enterprise; choose stage; toggle modules; bind/create DB connections; run tenant migrations; seed defaults.

Key UI Actions:

• Create Enterprise

• Choose Stage

• Toggle Modules

• Submit Provisioning

Primary Tables:

• core.enterprises

• core.app\_modules

• core.enterprise\_enabled\_modules

• core.enterprise\_db\_bindings

Routes:

• GET /core/provision

• POST /core/provision

## C) Development Pipeline

Track status/owner/ETA/progress, milestones, and progress/issue logs per enterprise.

Key UI Actions:

• Update status/progress

• Add milestone

• Add pipeline log

Primary Tables:

• core.pipeline\_projects

• core.pipeline\_milestones

• core.pipeline\_logs

Routes:

• GET /core/pipeline

• GET /core/pipeline/{projectId}

• POST /core/pipeline

• POST /core/pipeline/{projectId}/milestones

• POST /core/pipeline/{projectId}/logs

• PATCH/POST /core/pipeline/{projectId}

# API Routes (Laravel + Inertia)

GET /core/enterprises # Directory index (filters: q, stage\_id, lens\_id)  
GET /core/enterprises/{id} # Company profile  
GET /core/enterprises/{id}/open-database # Resolve default DB → redirect to Database Management  
  
GET /core/provision # Creator form (stages, modules)  
POST /core/provision # Create enterprise + enabled modules; dispatch Provisioner  
  
GET /core/pipeline # Project list  
GET /core/pipeline/{projectId} # Project detail  
POST /core/pipeline # Create/attach project  
POST /core/pipeline/{projectId}/milestones # Upsert milestone  
POST /core/pipeline/{projectId}/logs # Add log entry  
PATCH/POST /core/pipeline/{projectId} # Update status/owner/ETA/progress

# Database Specification (MySql)

## core.value\_chain\_stages

|  |  |  |
| --- | --- | --- |
| Column | Type | Constraints / Notes |
| id | uuid PK | primary key |
| code | text | unique, not null (e.g., STAGE\_1) |
| name | text | not null (e.g., '1. Resource Extraction') |
| ordinal | int | not null, 1..8 |
| description | text | nullable |
| created\_at | timestamptz | not null |
| updated\_at | timestamptz | not null |

## core.governance\_lenses

|  |  |  |
| --- | --- | --- |
| Column | Type | Constraints / Notes |
| id | uuid PK | primary key |
| code | text | unique, not null |
| name | text | not null |
| description | text | nullable |
| created\_at | timestamptz | not null |
| updated\_at | timestamptz | not null |

## core.app\_modules

|  |  |  |
| --- | --- | --- |
| Column | Type | Constraints / Notes |
| id | uuid PK | primary key |
| key | text | unique, not null (e.g., procurement, accounting) |
| name | text | not null |
| description | text | nullable |
| is\_core | boolean | not null default false |
| created\_at | timestamptz | not null |
| updated\_at | timestamptz | not null |

## core.enterprises

|  |  |  |
| --- | --- | --- |
| Column | Type | Constraints / Notes |
| id | uuid PK | primary key |
| tenant\_id | uuid | nullable (if central multi-tenancy) |
| slug | text | unique, not null |
| legal\_name | text | not null |
| display\_name | text | nullable |
| stage\_id | uuid | fk → core.value\_chain\_stages(id) |
| status | text | not null default 'active' (active/archived/prospect) |
| website\_url | text | nullable |
| metadata | jsonb | not null default {} |
| created\_by | uuid | nullable users.id |
| created\_at | timestamptz | not null |
| updated\_at | timestamptz | not null |
| deleted\_at | timestamptz | nullable (soft delete) |

## core.enterprise\_governance\_tags

|  |  |  |
| --- | --- | --- |
| Column | Type | Constraints / Notes |
| enterprise\_id | uuid | pk part, fk → core.enterprises(id) on delete cascade |
| lens\_id | uuid | pk part, fk → core.governance\_lenses(id) on delete restrict |

## core.enterprise\_identifiers

|  |  |  |
| --- | --- | --- |
| Column | Type | Constraints / Notes |
| id | uuid PK | primary key |
| enterprise\_id | uuid | fk → core.enterprises(id) on delete cascade |
| id\_type | text | not null (SEC, TIN, DTI, …) |
| id\_value | text | not null |
| verified\_at | timestamptz | nullable |
| unique |  | (enterprise\_id, id\_type) |

## core.enterprise\_contacts

|  |  |  |
| --- | --- | --- |
| Column | Type | Constraints / Notes |
| id | uuid PK | primary key |
| enterprise\_id | uuid | fk → core.enterprises(id) on delete cascade |
| name | text | nullable |
| role | text | nullable |
| email | text | nullable |
| phone | text | nullable |
| is\_primary | boolean | not null default false |
| created\_at | timestamptz | not null |
| updated\_at | timestamptz | not null |

## core.enterprise\_locations

|  |  |  |
| --- | --- | --- |
| Column | Type | Constraints / Notes |
| id | uuid PK | primary key |
| enterprise\_id | uuid | fk → core.enterprises(id) on delete cascade |
| label | text | not null (e.g., HQ, Plant A) |
| address\_line1/2, city, region, country, postal\_code | text | nullable |
| geo | jsonb | nullable {lat,lng} |
| created\_at | timestamptz | not null |
| updated\_at | timestamptz | not null |

## core.enterprise\_enabled\_modules

|  |  |  |
| --- | --- | --- |
| Column | Type | Constraints / Notes |
| enterprise\_id | uuid | pk part, fk → core.enterprises(id) on delete cascade |
| module\_id | uuid | pk part, fk → core.app\_modules(id) on delete restrict |
| enabled\_at | timestamptz | not null default now() |
| enabled\_by | uuid | nullable users.id |
| config | jsonb | not null default {} |

## core.enterprise\_db\_bindings

|  |  |  |
| --- | --- | --- |
| Column | Type | Constraints / Notes |
| id | uuid PK | primary key |
| enterprise\_id | uuid | fk → core.enterprises(id) on delete cascade |
| db\_connection\_id | uuid | fk → core.db\_connections(id) on delete restrict |
| role | text | not null default 'primary' (primary/reporting/etl/archive) |
| is\_default | boolean | not null default false |
| created\_at | timestamptz | not null default now() |
| unique |  | (enterprise\_id) where is\_default = true |

## core.pipeline\_projects

|  |  |  |
| --- | --- | --- |
| Column | Type | Constraints / Notes |
| id | uuid PK | primary key |
| enterprise\_id | uuid | fk → core.enterprises(id) on delete cascade |
| name | text | not null default 'Core System Deployment' |
| status | text | not null default planned (planned/in\_progress/blocked/done) |
| owner\_user\_id | uuid | nullable users.id |
| eta | date | nullable |
| progress\_pct | int | not null default 0 (0..100) |
| notes | text | nullable |
| created\_at | timestamptz | not null |
| updated\_at | timestamptz | not null |

## core.pipeline\_milestones

|  |  |  |
| --- | --- | --- |
| Column | Type | Constraints / Notes |
| id | uuid PK | primary key |
| project\_id | uuid | fk → core.pipeline\_projects(id) on delete cascade |
| name | text | not null |
| planned\_at | date | nullable |
| started\_at | timestamptz | nullable |
| completed\_at | timestamptz | nullable |
| status | text | not null default planned (planned/in\_progress/done/blocked) |
| owner\_user\_id | uuid | nullable users.id |
| notes | text | nullable |

## core.pipeline\_logs

|  |  |  |
| --- | --- | --- |
| Column | Type | Constraints / Notes |
| id | uuid PK | primary key |
| project\_id | uuid | fk → core.pipeline\_projects(id) on delete cascade |
| log\_type | text | not null default note (note/risk/decision/blocker/change) |
| severity | text | nullable (low/medium/high/critical) |
| message | text | not null |
| created\_by | uuid | nullable users.id |
| created\_at | timestamptz | not null default now() |

# Integration Notes

Laravel Integration Notes  
• Create schema: DB::statement('CREATE SCHEMA IF NOT EXISTS core');  
• Eloquent models: Enterprise, ValueChainStage, GovernanceLens, AppModule, EnterpriseEnabledModule, EnterpriseDbBinding, PipelineProject, PipelineMilestone, PipelineLog, EnterpriseIdentifier, EnterpriseContact, EnterpriseLocation.  
• Key relationships:  
 Enterprise->stage(), ->modules() (belongsToMany via core.enterprise\_enabled\_modules), ->dbBindings(), ->defaultDb(), ->pipeline().  
• Provisioner Job flow:  
 1) Create enterprise + enabled modules  
 2) Create/bind db\_connections in DB Management  
 3) Create enterprise\_db\_bindings (mark one is\_default)  
 4) Run tenant migrations for selected modules  
 5) Open a pipeline\_project + initial milestones