

Database Structure Documentation (Prisma + PostgreSQL)

1. Purpose of This Document

This document provides a **clear and structured explanation of the database design** for the Inventory, Billing, and Stock Management System. It is prepared in a **client-facing, professional format** to help stakeholders understand:

- What data is stored in the system
 - How tables relate to each other
 - Which business processes use which data
 - How APIs interact with the database
-

2. System Overview

The database supports the following business capabilities:

- Multi-location inventory management
- Role-based user access
- Product and barcode management
- Stock tracking with full audit trail
- Sales invoicing and taxation
- Automatic reorder tracking

The system is designed to be **scalable, auditable, and GST/tax compliant**.

3. Entity Relationship (ER) Overview

3.1 High-Level ER Relationships

- A **User** can be assigned to multiple **Locations** (many-to-many)
 - A **Location** contains many **Products** (via ProductLocation)
 - A **Product** is globally unique by barcode and can exist in multiple locations
 - An **Invoice** belongs to one location and is created by one user
 - An **Invoice** contains multiple **InvoiceItems**
 - Every stock change is recorded in **StockLedger** for audit purposes
 - **ReorderQueue** tracks products that fall below reorder levels
-

4. Table-Level Documentation

4.1 User

Stores system users such as administrators, managers, and staff.

Column	Type	Description
id	UUID	Unique user identifier
name	String	Full name
email	String	Login email (unique)
passwordHash	String	Encrypted password
role	Enum	ADMIN, MANAGER, STAFF
isActive	Boolean	Active status
createdAt	DateTime	Created timestamp
updatedAt	DateTime	Updated timestamp

Used for authentication, authorization, and audit tracking.

4.2 Location

Represents physical stores or warehouses.

Column	Type	Description
id	UUID	Location identifier
code	String	Unique location code
name	String	Location name
address	String	Address
isActive	Boolean	Status
createdAt	DateTime	Created timestamp
updatedAt	DateTime	Updated timestamp

Used for stock separation, billing, and reporting.

4.3 Product

Global master record of products identified uniquely by barcode.

Column	Type	Description
id	UUID	Product identifier

Column	Type	Description
productCode	String	Internal product code
barcode	String	Globally unique barcode
name	String	Product name
categoryId	UUID	Category reference
supplierId	UUID	Supplier reference
isActive	Boolean	Status

Used across inventory, billing, and stock audit processes.

4.4 ProductLocation

Maintains product-specific data per location.

Column	Type	Description
productId	UUID	Product reference
locationId	UUID	Location reference
purchasePrice	Decimal	Cost price
sellingPrice	Decimal	Selling price
taxId	UUID	Applied tax
currentStock	Decimal	Available quantity
rol	Decimal	Reorder level
eoq	Decimal	Economic order quantity

Used for inventory valuation, pricing, and reorder logic.

4.5 Invoice & InvoiceItem

Represents sales transactions and their line items.

Invoice stores header-level information, while InvoiceItem stores product-level details.

Used for billing, accounting, and tax reporting.

4.6 StockLedger

Immutable audit log of every stock movement.

Tracks: - Stock before and after change - Reason for change (sale, adjustment, check-in) - User who performed the action

Critical for audits and compliance.

4.7 ReorderQueue

Tracks products that fall below reorder levels.

Used for procurement alerts and automation workflows.

5. API-Level Mapping

This section explains how APIs interact with the database tables.

5.1 User & Access APIs

API	Tables Used	Purpose
POST /auth/login	User	Authentication
POST /users	User	Create user
POST /users/assign-location	UserLocation	Assign location

5.2 Product & Inventory APIs

API	Tables Used	Purpose
POST /products	Product	Create product
POST /products/location	ProductLocation	Add product to location
GET /products/barcode	Product	Barcode scan lookup
GET /inventory	ProductLocation	Stock view

5.3 Billing APIs

API	Tables Used	Purpose
POST /invoices	Invoice, InvoiceItem	Create invoice
POST /invoices/pay	Invoice	Mark invoice paid
GET /invoices	Invoice	Invoice listing

StockLedger is updated automatically during invoice creation.

5.4 Stock & Audit APIs

API	Tables Used	Purpose
POST /stock/checkin	StockLedger, ProductLocation	Stock addition
POST /stock/adjust	StockLedger	Manual adjustment
GET /stock/ledger	StockLedger	Audit report

5.5 Reorder APIs

API	Tables Used	Purpose
GET /reorder/queue	ReorderQueue	Low stock products
POST /reorder/close	ReorderQueue	Close reorder

6. Key Design Benefits

- Clear separation of global and location-specific data
- Full stock traceability
- Barcode-driven product identification
- Ready for GST and multi-tax scenarios
- Scalable to multiple stores and warehouses