DCC Sales Force Automation (SFA) - Developer Reference Handbook

Detailed Schema Documentation  
Version: 1.0

Date: 16/09/2025  
Prepared for: Development Team  
Prepared by: System Architect

Table of Contents

[1. Introduction 4](#_Toc209111778)

[2. System Overview 4](#_Toc209111779)

[3. Application Features List 4](#_Toc209111780)

[4. User Management 5](#_Toc209111781)

[Key Table: users 5](#_Toc209111782)

[5. Customer Management 5](#_Toc209111783)

[Tables 5](#_Toc209111784)

[Workflow Example 5](#_Toc209111785)

[6. Product & Inventory Management 6](#_Toc209111786)

[Tables 6](#_Toc209111787)

[Use Cases 6](#_Toc209111788)

[7. Promotions & Pricing 6](#_Toc209111789)

[Tables 6](#_Toc209111790)

[8. Sales Targets & Performance 6](#_Toc209111791)

[Tables 6](#_Toc209111792)

[9. Visits & Vehicles 6](#_Toc209111793)

[Tables 7](#_Toc209111794)

[10. Assets & Compliance 7](#_Toc209111795)

[Tables 7](#_Toc209111796)

[11. Workflows & Approvals 7](#_Toc209111797)

[Tables 7](#_Toc209111798)

[12. Customer Groups & Delivery Scheduling 7](#_Toc209111799)

[Tables 7](#_Toc209111800)

[13. WEB DASHBOARD UI (Desktop) 7](#_Toc209111801)

[MOBILE DASHBOARD UI (Field Sales App) 8](#_Toc209111802)

[14. User Roles 9](#_Toc209111803)

[Permissions and Visibility Matrix 10](#_Toc209111804)

[Role Management 10](#_Toc209111805)

[Mobile – Merchandiser / Technician View (Minimal, Task-Focused) 11](#_Toc209111806)

[Technician 11](#_Toc209111807)

[15. Customer Management 12](#_Toc209111808)

[New Customer Modal / Screen (Tabbed Form UI) 13](#_Toc209111809)

[16. Product & Inventory Management UI 16](#_Toc209111810)

[17. Promotions & Pricing 18](#_Toc209111811)

[18. Sales Targets & Performance 21](#_Toc209111812)

[19. Visit Logging & GPS Tracking 25](#_Toc209111813)

[Visit Outcomes 27](#_Toc209111814)

[20. Workflow & Multi-step Approvals 28](#_Toc209111815)

[21. Delivery Scheduling & Proof of Delivery (POD) 31](#_Toc209111816)

[22. Database Schema Reference (Full Details) 34](#_Toc209111817)

[Table: users 38](#_Toc209111818)

[Table: customers 38](#_Toc209111819)

[Table: products 39](#_Toc209111820)

[Table: product\_categories 39](#_Toc209111821)

[Table: product\_subcategories 40](#_Toc209111822)

[Table: batch\_lots 40](#_Toc209111823)

[Table: serial\_numbers 41](#_Toc209111824)

[Table: promotions 41](#_Toc209111825)

[Table: promotion\_parameters 42](#_Toc209111826)

[Table: sales\_targets 42](#_Toc209111827)

[Table: sales\_target\_assignments 43](#_Toc209111828)

[Table: bonuses 44](#_Toc209111829)

[Table: vehicles 44](#_Toc209111830)

[Table: visits 45](#_Toc209111831)

[Table: coolers 45](#_Toc209111832)

[Table: cooler\_inspections 46](#_Toc209111833)

[Table: approval\_workflows 46](#_Toc209111834)

[Table: workflow\_steps 48](#_Toc209111835)

[Table: customer\_documents 48](#_Toc209111836)

[Table: digital\_signatures 49](#_Toc209111837)

[Table: price\_history 50](#_Toc209111838)

[Table: customer\_groups 51](#_Toc209111839)

[Table: customer\_group\_members 52](#_Toc209111840)

[Table: delivery\_schedules 53](#_Toc209111841)

[Additional Recommendations 54](#_Toc209111842)

# 1. Introduction

This handbook serves as the primary developer reference for the Sales Force Automation (SFA) solution. It provides feature overviews, workflows, and detailed database schema to help developers implement, customize, and maintain the system effectively. The SFA solution is designed to enhance field sales operations, customer engagement, and performance monitoring.

# 2. System Overview

The SFA system automates sales operations by integrating customer management, product management, promotions, targets, workflows, and delivery scheduling. It ensures sales teams can operate efficiently while providing management with actionable insights.

# 3. Application Features List

|  |  |
| --- | --- |
| **Feature** | **Description** |
| Mobile Sales / Field Sales | Sales reps in the field can take orders, record visits, capture outlet/customer data via mobile app. |
| Distributor/Dealer Management | Manage relationships with distributors, dealers, chain stores. Assign pricing, discounts, promotions. |
| Route & Delivery Optimization | Optimize sales & delivery routes. GPS tracking of reps/vehicles. Map views, distance/time estimates. |
| Promotions & Discount Management | Define promotions (time‑based, product‑based, volume discounts), manage discount hierarchies. |
| Order & Invoice Management | Allow orders to be created via mobile/web, convert to invoices, track order status, manage returns. |
| Stock & Inventory Tracking | Monitor stock levels (warehouse, van stock, outlet stock), sync real‑time when possible. |
| Collection / Payments Tracking | Track collected payments, outstanding invoices, credit limits, reminders, settlements. |
| Reporting and Analytics | Dashboards & KPIs. Reports by region, rep, distributor, product, outlet. Trend analysis. |
| Real‑time Data Sync & Offline Capability | Offline working on mobile app; auto sync when connected. Real‑time data at HQ. |
| Workflow & Approvals | Discount approvals, credit approvals, order adjustments, supervisor validation workflows. |
| Notifications / Alerts | Alerts for low stock, overdue payments, promotions, tasks; push/SMS/email notifications. |
| User Roles & Permissions | Role-based access (Rep, Supervisor, Manager, Distributor, Admin). Control visibility & permissions. |
| Integration with ERP/Accounting | ERP sync (orders, invoices, taxes). Integration with e-commerce, finance systems. |
| Device Support | Mobile apps (Android/iOS), web portal for managers, optionally Windows app. |
| Map / GPS Functionality | Location tracking, route monitoring, delivery planning, geo-tagging of outlets. |
| B2B Ordering Portal | Portal/app for customers/distributors to place orders directly. |
| Audit / Log / History | Track all user activity, order changes, approvals for compliance & traceability. |

# List of Menus

|  |  |  |
| --- | --- | --- |
| User Management |  |  |
| SFA Settings |  |  |
| Reports |  |  |
| SFA |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# 4. User Management

The user management module handles system users, their roles, and activity tracking.

## Key Table: users

• id – Unique identifier  
• username – Login ID  
• is\_active – Active status  
• createdate/createdby – Audit fields  
• updatedate/updatedby – Audit fields  
• log\_inst – Log instance for concurrency control

The **Role Master UI** and the **Permissions Matrix** so your SFA system can scale, stay secure, and remain flexible as your org grows.

**1. Role Master UI (Web)**

This screen allows **Admins** to define roles, assign default access, and manage role hierarchy.

**🖥️ Role Master Screen Layout**

+--------------------------------------------------------------+

| ⚙️ Role Master |

+--------------------------------------------------------------+

| ⊕ Add Role | 🔍 Search Role | Filter: [ Active ✅ ▼ ] |

+--------------------------------------------------------------+

| Role Name | Level | Description | Status | ⚙️ |

|------------------|--------|--------------------------|--------|------|

| Admin / CEO | 5 | Full access | ✅ | ✏️ 🗑️ |

| Head Sales Mgr | 4 | Regional management | ✅ | ✏️ 🗑️ |

| Sales Coordinator| 3 | Mid-level coordinator | ✅ | ✏️ 🗑️ |

| Area Supervisor | 2 | Manages field teams | ✅ | ✏️ 🗑️ |

| Salesperson | 1 | Field sales user | ✅ | ✏️ 🗑️ |

| Merchandiser | 1 | Field merchandising | ✅ | ✏️ 🗑️ |

| Technician | 1 | Handles technical service| ✅ | ✏️ 🗑️ |

+--------------------------------------------------------------+

Add/Edit Role Modal

[ Add / Edit Role ]

Role Name: [ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ]

Level: [ 1–5 ▼ ] (higher = more power)

Description: [ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ]

Status: (•) Active ( ) Inactive

🛡️ Default Access Rights:

[☑️ View Only] [☑️ Create] [☐ Approve] [☑️ Edit] [☐ Delete]

[ Save Role ] [ Cancel ]

2. **Permissions Matrix (Role-Based Access Control)**

Define what **each role** can do across **modules**.

| **Module** | **Admin** | **Head Mgr** | **Coord.** | **Sup.** | **Sales** | **Merch** | **Tech** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 🎛️ Dashboard | Full | Full | Full | Team | Self | Self | Self |
| 👤 User Management | Full | View | View | ❌ | ❌ | ❌ | ❌ |
| 🧾 Orders | Full | View | Team | Team | Own | ❌ | ❌ |
| 🎯 Targets | Full | Full | Team | Assign | View | ❌ | ❌ |
| 🎁 Promotions | Full | Approve | Submit | View | View | ❌ | ❌ |
| 📍 Visits & GPS | View | View | Team | Team | Own | Own | Own |
| 🧾 Approvals Workflow | Approve | Approve | Approve | Partial | Request | ❌ | ❌ |
| 🚚 Delivery + POD | View | View | Assign | View | Confirm | View | Confirm |
| 📷 Assets / Compliance | View | View | View | Team | ❌ | Upload | Manage |

✅ This can be stored in a simple config table (Role x Module x Action = TRUE/FALSE)

Use a **backend config table** like:

| role\_id | module\_name | can\_view | can\_create | can\_approve | can\_edit |

|---------|------------------|----------|------------|-------------|----------|

| 1 | Orders | 1 | 0 | 0 | 0 |

| 4 | Orders | 1 | 1 | 1 | 1 |

In UI logic, **disable buttons** or **hide sections** based on permissions.

**Optional Enhancements**

| **Feature** | **Description** |
| --- | --- |
| 🧩 Role Templates | Pre-built role setups for new orgs |
| 🧬 Role Cloning | Duplicate an existing role config easily |
| 📊 Role Usage Audit | Track which role accesses which modules |
| 🔄 Role Change Log | Audit trail for role edits & permissions |

Page: Role Master

╔══════════════════════════════════════════════════════════╗

║ ⚙️ ROLE MASTER ║

╠══════════════════════════════════════════════════════════╣

║ [+ Add Role] [🔍 Search Role \_\_\_\_\_\_\_\_\_\_\_] [Filter ▼] ║

╠══════════════════════════════════════════════════════════╣

║ Role Name | Level | Description | Status | ⚙️ Actions ║

║────────────────|───────|----------------------|--------|────────────║

║ Admin / CEO | 5 | Full access | ✅ | ✏️ 🗑️ ║

║ Head Sales Mgr | 4 | Regional mgmt | ✅ | ✏️ 🗑️ ║

║ Sales Coord. | 3 | Oversees teams | ✅ | ✏️ 🗑️ ║

║ Area Supv. | 2 | Manages field users | ✅ | ✏️ 🗑️ ║

║ Salesperson | 1 | Field sales user | ✅ | ✏️ 🗑️ ║

║ Merchandiser | 1 | Executes branding | ✅ | ✏️ 🗑️ ║

║ Technician | 1 | Handles equipment | ✅ | ✏️ 🗑️ ║

╚══════════════════════════════════════════════════════════╝

[Modal Pop-up] Add / Edit Role

╔══════════════════════════════╗

║ Add / Edit Role ║

╠══════════════════════════════╣

║ Role Name: [\_\_\_\_\_\_\_\_\_\_] ║

║ Level (1–5): [\_\_\_▼\_\_\_] ║

║ Description: [\_\_\_\_\_\_\_\_\_\_] ║

║ Status: (•) Active ( ) Inactive ║

╠══════════════════════════════╣

║ 🛡️ Default Access Rights: ║

║ [✓] View [✓] Create ║

║ [ ] Approve [✓] Edit ║

║ [ ] Delete ║

╠══════════════════════════════╣

║ [ Save Role ] [ Cancel ] ║

╚══════════════════════════════╝

# 5. Customer Management

Purpose: To manage customers, compliance, documents, and group associations.

## Tables

• customers – Core customer records  
• customer\_groups – Defines customer groupings  
• customer\_group\_members – Links customers to groups  
• customer\_documents – Stores compliance documents  
• digital\_signatures – Captures customer approvals

## Workflow Example

1. A new customer is registered by a salesperson.  
2. Customer is assigned to one or more groups.  
3. Required compliance documents are uploaded and verified.  
4. Customer signs digital agreements captured in the system.

# 6. Product & Inventory Management

This module manages product catalogs, categories, subcategories, batch and serial tracking.

## Tables

• products – Master product data  
• product\_categories – Defines product categories  
• product\_subcategories – Defines product subcategories  
• batch\_lots – Manages batch tracking  
• serial\_numbers – Manages serial number tracking

## Use Cases

• Create new products and categorize them  
• Assign batch numbers to products  
• Track serialized items for warranty and compliance  
• Integrate with promotions and pricing modules

# 7. Promotions & Pricing

Promotions and pricing rules drive sales incentives and flexible price management.

## Tables

• promotions – Promotion master  
• promotion\_parameters – Defines promotion rules  
• price\_history – Tracks historical price changes

# 8. Sales Targets & Performance

Sales targets ensure performance tracking at product, category, and salesperson level.

## Tables

• sales\_targets – Target definitions  
• sales\_target\_assignments – Assigns targets to salespeople  
• bonuses – Defines bonus structures

# 9. Visits & Vehicles

Tracks customer visits and manages vehicles assigned to field staff.

## Tables

• visits – Records customer visits  
• vehicles – Manages vehicles assigned to salespersons

# 10. Assets & Compliance

Manages assets provided to customers (e.g., coolers) and ensures compliance checks.

## Tables

• coolers – Coolers installed at customer sites  
• cooler\_inspections – Regular inspection records  
• customer\_documents – Compliance docs  
• digital\_signatures – E-signature verification

# 11. Workflows & Approvals

Approval workflows manage multi-step approvals for pricing, discounts, and promotions.

## Tables

• approval\_workflows – Workflow definition  
• workflow\_steps – Steps in a workflow process

# 12. Customer Groups & Delivery Scheduling

Groups customers for segmentation and manages delivery planning.

## Tables

• customer\_groups – Defines groups  
• customer\_group\_members – Group membership  
• delivery\_schedules – Delivery assignments and tracking

# 13. WEB DASHBOARD UI (Desktop)

+-----------------------------------------------------------------------------------+

| Logo | 🧑 Profile | 🔔 Notifications | ⚙️ Settings |

+-----------------------------------------------------------------------------------+

| 📊 Dashboard | 👤 Customers | 📦 Products | 🎯 Targets | 🚚 Delivery | ⚙️ Admin |

| 📍 Visits | 💰 Promotions | 🧾 Approvals | 🧊 Assets | 📋 Compliance |

|-----------------------------------------------------------------------------------|

| DASHBOARD SUMMARY |

|-----------------------------------------------------------------------------------|

| 🧍 Sales Today | 🎯 Target Progress | 🚚 Deliveries | 🔄 Pending Approvals |

| ₹12,500 | 80% of ₹50,000 | 12 Deliveries | 3 Waiting for Approval |

|-----------------------------------------------------------------------------------|

| 🔍 Quick Actions: [ + New Customer ] [ + Order ] [ Start Visit ] |

|-----------------------------------------------------------------------------------|

| 📍 Upcoming Visits | 📦 Inventory Alerts | 💡 Promotions Active |

| - ABC Stores, 11 AM | - Low stock: Cola 330ml | - Diwali Offer (ends Fri) |

+-----------------------------------------------------------------------------------+

## MOBILE DASHBOARD UI (Field Sales App)

+------------------------------------------------+

| ☰ Menu Dashboard 🔔 |

+------------------------------------------------+

| 🧍 ₹12,500 🎯 80% Target 🚚 12 Dels |

+------------------------------------------------+

| 📍 Next Visit: ABC Stores – 11:00 AM |

+------------------------------------------------+

| 🔘 Quick Actions: |

| [+ Customer] [+ Order] [Start Visit] |

+------------------------------------------------+

| 📦 Low Stock Alert: Cola 330ml |

| 💡 Active Promo: Diwali Discount |

+------------------------------------------------+

| Footer Nav: 📊 Home | 👤 Cust | 📦 Prod | 🧾 Orders |

+------------------------------------------------+

# 14. User Roles

1️⃣ Admin / CEO – sees & controls everything

2️⃣ Head Sales Manager – oversees entire sales org

3️⃣ Sales Coordinator – manages Area Supervisors & supports sales tracking

4️⃣ Area Sales Supervisor – leads a field sales team

5️⃣ Salesperson – handles daily visits, orders, and customer interactions

6️⃣ Merchandiser – handles in-store branding, visibility, planograms

7️⃣ Technician – handles cooler maintenance, hardware, installations

A diagram of a company

AI-generated content may be incorrect.

## Permissions and Visibility Matrix

| **Role** | **View Own** | **View Team** | **Approve** | **View Assets** | **View Branding** |
| --- | --- | --- | --- | --- | --- |
| **Admin / CEO** | **✅** | **✅ All** | **✅ All** | **✅** | **✅** |
| **Head Sales Mgr** | **✅** | **✅ All** | **✅** | **✅** | **✅** |
| **Sales Coordinator** | **✅** | **✅** | **✅** | **✅** | **✅** |
| **Area Supervisor** | **✅** | **✅** | **Partial** | **✅** | **✅** |
| **Salesperson** | **✅** | **❌** | **❌** | **❌** | **❌** |
| **Merchandiser** | **✅** | **❌** | **❌** | **❌** | **✅** |
| **Technician** | **✅** | **❌** | **❌** | **✅ (Coolers)** | **❌** |

## Role Management

**+---------------------------------------------------------------------------------+**

**| 📋 User Role & Hierarchy Management |**

**+---------------------------------------------------------------------------------+**

**| 🔍 Search Users | ⊕ Add New User | 🔄 Sync Roles | ⎘ Export |**

**+---------------------------------------------------------------------------------+**

**| Name | Role | Reports To | Team Count | 🔧 |**

**|--------------|------------------|-----------------------|------------|-------------|**

**| John A. | Admin / CEO | — | All | [Edit] |**

**| Priya S. | Head Sales Mgr | John A. | 3 | [Edit] |**

**| Ravi K. | Sales Coord. | Priya S. | 7 | [Edit] |**

**| Meena P. | Area Sup. | Ravi K. | 12 | [Edit] |**

**| Akash M. | Salesperson | Meena P. | — | [Edit] |**

**| Sneha B. | Merchandiser | Meena P. | — | [Edit] |**

**| Suraj R. | Technician | Meena P. | — | [Edit] |**

**+---------------------------------------------------------------------------------+**

## Mobile – Merchandiser / Technician View (Minimal, Task-Focused)

**+----------------------------+**

**| Role: Merchandiser |**

**+----------------------------+**

**| 📍 Today’s Stores (3) |**

**| 🧾 Tasks: |**

**| - Update branding photos |**

**| - Submit shelf visibility |**

**| - Planogram compliance |**

**| 📸 [ Upload Photo ] |**

**+----------------------------+**

## Technician

**+-----------------------------+**

**| Role: Technician |**

**+-----------------------------+**

**| 🧊 Assigned Coolers (5) |**

**| 📅 Today’s Visits (2) |**

**| ⚙️ Tasks: |**

**| - Fix leak @ ABC Café |**

**| - Replace unit @ XYZ Store |**

**| 📎 Upload Service Report |**

**+-----------------------------+**

# 15. Customer Management

**Customer Management screen** for your **Sales Force Automation system**, focusing on:

* **Customer Onboarding**
* **Document Upload (KYC)**
* **Digital Signatures**
* **Customer Groups/Segmentation**

Customer Management – Web UI Wireframe

+--------------------------------------------------------------------------------+

| Logo | Dashboard | Customers | Products | Visits | Orders | [Profile] |

+--------------------------------------------------------------------------------+

| 📂 Customer Management |

|--------------------------------------------------------------------------------|

| 🔍 Search | ⊕ New Customer | ⛃ Import CSV | ⎘ Download Data |

|--------------------------------------------------------------------------------|

| Name | Group | Status | Created On | Actions (👁️ ✏️ 🗑️) |

|------------|-----------|------------|------------|-----------------------------|

| ABC Stores | Retail | ✅ Active | 10-Aug-25 | [View] [Edit] [Delete] |

| XYZ Café | HORECA | 🟡 Pending | 12-Aug-25 | [View] [Edit] [Delete] |

|--------------------------------------------------------------------------------|

| ⬅️ Prev | Page 1 of 5 | ➡️ Next |

+--------------------------------------------------------------------------------+

## New Customer Modal / Screen (Tabbed Form UI)

**[New Customer Onboarding]**

**Tabs: [🧍 Basic Info] [📑 Documents] [🖋 Digital Signature] [👥 Grouping]**

**------------------------------------------------------**

**🧍 Basic Info Tab:**

**------------------------------------------------------**

**Customer Name: [\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_]**

**Contact Person: [\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_]**

**Phone Number: [\_\_\_\_\_\_] Email: [\_\_\_\_\_\_]**

**Customer Type: ( ) Retail ( ) HORECA ( ) Modern Trade**

**Geo-Location: [📍 Pick from Map]**

**[Next >]**

**------------------------------------------------------**

**📑 Documents Tab:**

**------------------------------------------------------**

**Upload Business License: [Choose File] ✅**

**Upload PAN / GST Docs: [Choose File] ❌**

**Upload Store Photo: [Choose File] ✅**

**[< Back] [Next >]**

**------------------------------------------------------**

**🖋 Digital Signature Tab:**

**------------------------------------------------------**

**Customer Signature: [ ✍️ Draw on Canvas ]**

**[📎 Upload Image]**

**Salesperson Signature: [ ✍️ Draw or Upload ]**

**[< Back] [Next >]**

**------------------------------------------------------**

**👥 Grouping Tab:**

**------------------------------------------------------**

**Assign Group: [Retail, Gold Tier ▼]**

**Assigned By: [Salesperson Name]**

**Route/Beat: [Monday – Route A ▼]**

**[Save] [Cancel]**

**Customer Management – Mobile UI Wireframe**

**+--------------------------------------+**

**| ☰ Menu Customers 🔍 |**

**+--------------------------------------+**

**| ⊕ Add New Customer |**

**|--------------------------------------|**

**| ABC Stores ✅ Active |**

**| Retail | Created: 10-Aug |**

**| [👁️ View] [✏️ Edit] |**

**|--------------------------------------|**

**| XYZ Café 🟡 Pending |**

**| HORECA | Created: 12-Aug |**

**+--------------------------------------+**

**+ Floating Action Button (FAB): ⊕ +**

**Add Customer (Step-based Form)**

**Step 1/4: Basic Info**

**---------------------**

**Name: [\_\_\_\_\_\_\_\_\_\_]**

**Phone: [\_\_\_\_\_\_\_\_\_\_]**

**Type: [Retail ▼]**

**[Next >]**

**Step 2/4: Upload Docs**

**---------------------**

**[ + Upload License ]**

**[ + Upload PAN Card ]**

**[Next >]**

**Step 3/4: Signature**

**---------------------**

**[ ✍️ Draw Signature ]**

**[ Upload Photo ]**

**Step 4/4: Group & Route**

**---------------------**

**Group: [Gold ▼]**

**Route: [Mon - Route A ▼]**

**[Submit]**

# 16. Product & Inventory Management UI

This module includes:

* Product Catalog (with images, categories, units)
* Batch/Lot/Serial Number tracking
* Stock Levels (by location or salesperson)
* Integration with Orders and Promotions

**Web UI Wireframe – Product & Inventory**

**+----------------------------------------------------------------------------------+**

**| 📦 Product & Inventory Management |**

**+----------------------------------------------------------------------------------+**

**| 🔍 Search Product | ⊕ Add Product | ⎘ Import Excel | ⎘ Export List |**

**+----------------------------------------------------------------------------------+**

**| Image | Product Name | SKU | Category | Stock | Serial/Batch | 🔧 |**

**|-------|------------------|----------|------------|-------|---------------|--------|**

**| 🖼️ | Cola 330ml Can | SKU-001 | Beverages | 320 | BATCH-001 | ✏️ 🗑️ |**

**| 🖼️ | Energy Drink 500ml| SKU-002 | Beverages | 150 | SN-90829 | ✏️ 🗑️ |**

**| 🖼️ | Chips 150g | SKU-010 | Snacks | 700 | — | ✏️ 🗑️ |**

**+----------------------------------------------------------------------------------+**

**🔘 Filters: [ Category ▼ ] [ In Stock ✅ ] [ Serial/Batch ▼ ]**

**Add Product Modal / Form**

**[ Add New Product ]**

**Product Name: [\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_]**

**SKU Code: [\_\_\_\_\_\_\_\_\_\_]**

**Category: [ Snacks, Beverages, etc. ▼ ]**

**Unit: [ Carton, Bottle, Can ▼ ]**

**Image: [📎 Upload Image]**

**Supports: [x] Serial Number [x] Batch Number**

**Initial Stock: [\_\_\_\_\_ units] [📍 Select Warehouse ▼]**

**[Save Product] [Cancel]**

**Mobile UI – Product List (Field App)**

**+--------------------------------------------------+**

**| ☰ Menu Products 🔍 |**

**+--------------------------------------------------+**

**| Cola 330ml Can | SKU: 001 | Stock: 320 |**

**| [Image] | Batch: BATCH-001 |**

**|--------------------------------------------------|**

**| Energy Drink 500ml| SKU: 002 | SN: SN-90829 |**

**| [Image] | Stock: 150 |**

**+--------------------------------------------------+**

**[ ⊕ Add Product ] (Admin/Supervisor only)**

**Role-Based Access (Permissions)**

| **Role** | **View Stock** | **Edit Product** | **Add Product** | **See Batches** |
| --- | --- | --- | --- | --- |
| **Admin / CEO** | **✅** | **✅** | **✅** | **✅** |
| **Head Sales Manager** | **✅** | **❌** | **❌** | **✅** |
| **Area Sales Supervisor** | **✅** | **✅ (local)** | **✅ (local)** | **✅** |
| **Salesperson** | **✅ (assigned)** | **❌** | **❌** | **Read-Only** |
| **Merchandiser** | **✅ (read-only)** | **❌** | **❌** | **❌** |
| **Technician** | **❌** | **❌** | **❌** | **❌** |

# 17. Promotions & Pricing

**📌 Core Features:**

* **Manage Promotions (SKU-based, time-based, customer-segment-based)**
* **Flexible Pricing Engine (price lists, discounts, special rates)**
* **Visibility by Role**
* **Approval Flows (e.g., for custom discounts)**

**Web UI – Promotions Dashboard**

**+--------------------------------------------------------------------------------+**

**| 🎯 Promotions & Pricing Management |**

**+--------------------------------------------------------------------------------+**

**| 🔍 Search Promo | ⊕ Create Promotion | ⎘ Export | 📁 View Price History |**

**+--------------------------------------------------------------------------------+**

**| Name | Type | Validity | Target Group | Status | 🔧 |**

**|-----------------|----------|------------------|---------------|--------|-------|**

**| Festive Offer | Flat % | 01-10 to 10-10 | Retailers | ✅ | ✏️ 🗑️ |**

**| Combo Deal | BOGO | 15-09 to 30-09 | All | ✅ | ✏️ 🗑️ |**

**| Weekend Special | Cash Off| 21-09 to 22-09 | HORECA | 🟡 | ✏️ 🗑️ |**

**+--------------------------------------------------------------------------------+**

**Filters: [ Promo Type ▼ ] [ Target Group ▼ ] [ Active ✅ ]**

**[ 📊 View Pricing Matrix ]**

**Create Promotion – Form (Multi-Step)**

**Step 1: Promotion Details**

* **Promotion Name: [\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_]**
* **Type: ( ) Flat % ( ) Cash Off ( ) BOGO ( ) Custom**
* **Valid From / To: [Date ▼] to [Date ▼]**
* **Target Group: [Retailers, HORECA, etc. ▼]**

**Step 2: Apply To Products**

* **Select Products:**

**[☑️ Cola 330ml] [☐ Chips 100g] [☑️ Energy Drink 500ml]**

**Discount/Offer:**

* **If % → "10% Off"**
* **If BOGO → "Buy 1 Get 1 Free"**

**Step 3: Approval**

* **Requires Approval: ✅**
* **Approver Role: [Sales Coordinator ▼]**
* **Comments: [\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_]**

**[Save Promotion] [Cancel]**

**Mobile UI – Promotions (Salesperson View)**

**+--------------------------------------------------+**

**| 🎯 Active Promotions |**

**+--------------------------------------------------+**

**| Festive Offer - 10% Off |**

**| ✔️ Cola 330ml | ✔️ Energy Drink 500ml |**

**| 📅 Valid till: 10-Oct |**

**+--------------------------------------------------+**

**| Combo Deal – BOGO |**

**| ✔️ Chips 150g |**

**| 📅 Valid: 15–30 Sept |**

**Price Management Module**

**Accessible from sidebar or tab within Promotions**

**Price List Matrix (Web)**

| **Product** | **Base Price** | **Promo Price** | **Start Date** | **End Date** | **Group** |
| --- | --- | --- | --- | --- | --- |
| **Cola 330ml** | **₹30.00** | **₹27.00** | **01-10** | **10-10** | **Retailers** |
| **Chips 150g** | **₹20.00** | **₹20.00** | **—** | **—** | **All** |
| **Energy Drink** | **₹50.00** | **₹45.00** | **01-10** | **10-10** | **HORECA** |

**Role-Based Permissions**

| **Role** | **View Promos** | **Create** | **Approve** | **Price Edit** |
| --- | --- | --- | --- | --- |
| **Admin / CEO** | **✅** | **✅** | **✅** | **✅** |
| **Head Sales Manager** | **✅** | **✅** | **✅** | **✅** |
| **Sales Coordinator** | **✅** | **✅** | **✅** | **Partial** |
| **Area Supervisor** | **✅** | **✅** | **❌** | **❌** |
| **Salesperson** | **✅ (assigned only)** | **❌** | **❌** | **❌** |

# 18. Sales Targets & Performance

**Goal:**

Set, track, and manage targets at **any level**: Salesperson, Supervisor, Region, Product Line, or Time Period.

**Who Uses This Module?**

| **Role** | **What They Do Here** |
| --- | --- |
| Admin / CEO | Set org-wide targets, view national performance |
| Head Sales Manager | Assign targets regionally |
| Sales Coordinator | Manage targets for supervisors & teams |
| Area Sales Supervisor | View team performance, assign to salespersons |
| Salesperson | View own targets vs. achievements |

Web UI – Sales Target Management

+--------------------------------------------------------------------------------------+

| 📊 Sales Targets & Performance |

+--------------------------------------------------------------------------------------+

| ⊕ Create Target | 📅 Select Month: [ Sept 2025 ▼ ] | ⎘ Export Report | 🔍 Search |

+--------------------------------------------------------------------------------------+

| Employee | Role | Target (₹) | Achieved (₹) | % | 🔘 Status |

|------------------|--------------------|-------------|--------------|----|-------------|

| Akash M. | Salesperson | ₹ 150,000 | ₹ 142,000 | 95%| 🟡 Near |

| Meena P. | Area Supervisor | ₹ 800,000 | ₹ 920,000 | 115%| ✅ Met |

| Ravi K. | Sales Coordinator | ₹ 2,500,000 | ₹ 2,200,000 | 88%| 🔴 Below |

+--------------------------------------------------------------------------------------+

[📊 View by Product ▼] [📍 View by Region ▼] [🧑 View by User ▼]

**Create Sales Target – Form**

**[ Create New Sales Target ]**

**🎯 Target For: [ Salesperson ▼ ]**

**Name: [Akash M.]**

**Period: [ Sept 2025 ▼ ]**

**Type: ( ) Value Based ( ) Volume Based**

**Breakdown:**

**---------------------------------------------**

**Product | Target (Units or ₹)**

**--------------------|-----------------------**

**Cola 330ml | 5,000 units**

**Chips 150g | 3,000 units**

**Energy Drink 500ml | ₹ 50,000**

**---------------------------------------------**

**Total Target Value: ₹ 150,000**

**[Assign Target] [Cancel]**

**Mobile UI – Salesperson Target Dashboard**

**+---------------------------------------------+**

**| 🎯 My Target – September 2025 |**

**+---------------------------------------------+**

**| 🎯 Total Target: ₹150,000 |**

**| ✅ Achieved: ₹142,000 (95%) |**

**| 🗓️ Days Left: 5 |**

**| 🔄 Last Updated: Today |**

**+---------------------------------------------+**

**| 📊 Product-wise: |**

**| - Cola 330ml: 4800 / 5000 units |**

**| - Chips 150g: 3100 / 3000 units ✅ |**

**| - Energy Drink: ₹48,000 / ₹50,000 |**

**📈 Analytics Dashboard (Web)**

**+---------------------------------------------------------------+**

**| 📊 Target vs Achievement Report (By Region, Product, User) |**

**+---------------------------------------------------------------+**

**| 🔍 Filter: [Region ▼] [Month ▼] [Product ▼] [User ▼] |**

**+---------------------------------------------------------------+**

**| 📍 East Region - September 2025 |**

**| Target: ₹5,000,000 | Achieved: ₹4,800,000 | %: 96% |**

**|---------------------------------------------------------------|**

**| 🧊 Top Products Underperforming: Energy Drink, Chips |**

**| 🧑 Top Performers: Meena P. (115%), Akash M. (95%) |**

**+---------------------------------------------------------------+**

**[ ⎘ Download Excel ] [ 📈 View Charts ]**

**Role-Based Access**

| **Role** | **View Targets** | **Assign Targets** | **Approve Bonuses** |
| --- | --- | --- | --- |
| **Admin / CEO** | **✅ Org-wide** | **✅** | **✅** |
| **Head Sales Manager** | **✅ Regionally** | **✅** | **✅** |
| **Sales Coordinator** | **✅ Team** | **✅** | **Partial (if delegated)** |
| **Area Supervisor** | **✅ Own + Team** | **✅ (only Salesperson level)** | **❌** |
| **Salesperson** | **✅ Own Only** | **❌** | **❌** |

**Bonuses & Rewards (Optional Module)**

* **Based on target achievement %**
* **Trigger rules:**
  + **90%+ = Bronze Bonus**
  + **100%+ = Silver Bonus**
  + **120%+ = Gold Bonus**
* **Visible in salesperson’s app (“🎉 You earned ₹3,000 this month!”)**

# 19. Visit Logging & GPS Tracking

**Objective:**

Enable **salespeople**, **merchandisers**, and **technicians** to log visits, capture location data, plan routes, and provide proof of interaction.

**Who Uses It?**

| **Role** | **What They Do Here** |
| --- | --- |
| Salesperson | Check-in/check-out at outlets, log meeting outcomes |
| Merchandiser | Log branding work, take photos |
| Technician | Log service visits, attach service reports |
| Area Supervisor | Monitor real-time visit status of team |
| Sales Coordinator | Review logs, productivity, and route deviations |
| Admin / CEO | View high-level reports and heatmaps |

Web UI – Visit Monitoring Dashboard (Manager View)

+-------------------------------------------------------------------------------------+

| 📍 Visit Tracking – Live Dashboard |

+-------------------------------------------------------------------------------------+

| 📅 Date: [ 18-Sept-2025 ▼ ] | 🔍 Search User | 📍 Map View | 📈 Analytics View |

+-------------------------------------------------------------------------------------+

| Salesperson | Check-in Time | Outlet | Status | Distance | 📸 | 🔧 |

|-------------|----------------|---------------|----------|----------|-----|-------------|

| Akash M. | 10:15 AM | ABC Retailers | ✅ Done | 2.3 km | 🖼️ | ✏️ Log |

| Sneha B. | 11:02 AM | XYZ Store | ⏳ Ongoing| 1.8 km | | |

| Suraj R. | 09:40 AM | Café Coolers | ✅ Done | 3.5 km | 🖼️ | 🔧 Repair Log|

+-------------------------------------------------------------------------------------+

[ 📍 View Route Map ] [ ⎘ Export Report ]

**Mobile UI – Salesperson Visit Logging**

**+--------------------------------------------------+**

**| 📍 My Visit – Outlet: ABC Retailers |**

**+--------------------------------------------------+**

**| 🕒 Check-in: 10:15 AM |**

**| 📍 GPS: Accurate 📶 |**

**| 📷 Photo: [📸 Take Store Photo] |**

**| 🧾 Notes: [\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_] |**

**| 🛒 Order Taken: ₹5,200 |**

**| 🎁 Promo Applied: Festive Offer |**

**| [✅ Check-out] |**

**+--------------------------------------------------+**

**Route Plan – Daily View (Field Staff)**

**+--------------------------------------------------+**

**| 🧭 Today's Visits – 18-Sept-2025 |**

**+--------------------------------------------------+**

**| 1. ABC Retailers – 📍 2.1 km – 🕒 10:00 AM |**

**| 2. XYZ Store – 📍 3.5 km – 🕒 12:00 PM |**

**| 3. New Outlet Visit – 🆕 Not in system |**

**+--------------------------------------------------+**

**| [Start Navigation] [Mark as Skipped] |**

**Manager Route Map (Web)**

**Live GPS tracking heatmap:**

* **Show breadcrumb trail of all users’ daily movements**
* **Color-coded markers: ✅ Visited, ⏳ Ongoing, ❌ Skipped**

## Visit Outcomes

| **Captured Data** | **Used For** |
| --- | --- |
| 📍 GPS coordinates | Proof of visit |
| 🕒 Check-in/out time | Productivity and time-in-field |
| 🧾 Notes / Orders | Conversion ratio tracking |
| 📷 Photos | Visual audit (store shelf, branding) |
| 👥 New Outlet | Expansion pipeline |

**Role-Based Access**

| **Role** | **View Visits** | **Create** | **Edit/Update** | **GPS Logs** |
| --- | --- | --- | --- | --- |
| **Admin / CEO** | **✅ All** | **❌** | **❌** | **✅ All** |
| **Head Sales Manager** | **✅ All** | **❌** | **❌** | **✅ All** |
| **Sales Coordinator** | **✅ Team** | **❌** | **✅ Notes** | **✅ Team** |
| **Area Supervisor** | **✅ Team** | **✅** | **✅** | **✅ Team** |
| **Salesperson** | **✅ Own** | **✅** | **✅** | **Own Only** |
| **Merchandiser** | **✅ Own** | **✅** | **✅ Branding** | **Own Only** |
| **Technician** | **✅ Own** | **✅** | **✅ Repairs** | **Own Only** |

# 20. Workflow & Multi-step Approvals

**Purpose:**

Ensure business-critical actions (like discounts, promotions, contract approvals, new customer onboarding, and exception orders) go through **structured multi-level approvals**.

**Key Use Cases:**

| **Use Case** | **Example** |
| --- | --- |
| 🧾 Discount Approval | 20% discount request above allowed limit |
| 🎯 Promotion Launch Approval | New promo needing Head Manager sign-off |
| 🧍 Customer Onboarding | New HORECA client requiring doc verification |
| ⚙️ Manual Order Review | Unusual order from restricted zone |
| 💼 Contract / Pricing Deal | Long-term pricing contract approval |

**Web UI – Workflow Dashboard (Manager View)**

**+--------------------------------------------------------------------------------+**

**| 🧾 Workflow & Approvals Dashboard |**

**+--------------------------------------------------------------------------------+**

**| 🔍 Search Request | Filter: [ Type ▼ ] [ Status ▼ ] | ⎘ Export |**

**+--------------------------------------------------------------------------------+**

**| Request ID | Type | Requested By | Date | Status | 🔍 View | ✅ |**

**|------------|-------------|--------------|------------|----------|---------|----|**

**| #REQ1021 | Discount | Akash M. | 18-Sep | 🟡 Pending| 🔍 | |**

**| #REQ1022 | Promo Launch | Meena P. | 17-Sep | ✅ Approved| 🔍 | |**

**| #REQ1023 | Onboarding | Sneha B. | 16-Sep | ❌ Rejected| 🔍 | |**

**+--------------------------------------------------------------------------------+**

**Create New Approval Workflow (Admin Only)**

**[ Create Workflow Template ]**

**Workflow Name: [ Special Discount Approval ]**

**Request Type: [ Discount Request ▼ ]**

**Step 1:**

**Approver Role: [ Area Supervisor ▼ ]**

**Condition: [ Discount > 5% ]**

**Step 2:**

**Approver Role: [ Sales Coordinator ▼ ]**

**Condition: [ Discount > 10% ]**

**Step 3:**

**Approver Role: [ Head Sales Manager ▼ ]**

**Condition: [ Discount > 20% ]**

**[ Save Workflow Template ]**

**Mobile UI – Approval Inbox (for Managers)**

**+--------------------------------------------------+**

**| 🧾 My Approvals (2 Pending) |**

**+--------------------------------------------------+**

**| 🔔 Discount Request – #REQ1021 |**

**| From: Akash M. | ₹3,200 @ 25% Discount |**

**| Reason: “Client bulk order” |**

**| [✅ Approve] [❌ Reject] [📎 View Details] |**

**+--------------------------------------------------+**

**Workflow Logic Engine (Back-End)**

**🔧 Rule Builder Options:**

* **Trigger by: User Role, Product, Customer Type, Region, Discount %, Value**
* **Approval Escalation: auto-escalate after X hours**
* **Notifications: In-app + Email + (optional SMS)**
* **Attachments: upload contracts, photos, approvals**

**Role-Based Workflow Participation**

| **Role** | **Can Request** | **Can Approve** | **Can Escalate** | **Admin Templates** |
| --- | --- | --- | --- | --- |
| **Salesperson** | **✅** | **❌** | **❌** | **❌** |
| **Area Supervisor** | **✅** | **✅ (Step 1)** | **✅** | **❌** |
| **Sales Coordinator** | **✅** | **✅ (Step 2)** | **✅** | **❌** |
| **Head Sales Manager** | **✅** | **✅ (Final)** | **✅** | **❌** |
| **Admin / CEO** | **✅** | **✅** | **✅** | **✅** |

**Smart Workflow Features**

| **Feature** | **Description** |
| --- | --- |
| **🔄 Auto-routing** | **Skip steps if approver = requester** |
| **📎 Attach supporting docs** | **e.g., customer letter, volume contract** |
| **⏳ SLA Timers** | **Track turnaround times** |
| **📬 Notification system** | **Approver reminders, escalations** |
| **📈 Approval Analytics** | **Avg approval time, # pending, rejection rate** |

# 21. Delivery Scheduling & Proof of Delivery (POD)

**Objective:**

Enable efficient planning of delivery routes, order dispatching, and real-time proof of delivery collection from customers, all linked with your **orders and field staff movement**.

**Key Features:**

| **Category** | **Functions Included** |
| --- | --- |
| 📅 Delivery Planning | Route planning, order bundling, vehicle assignment |
| 🧑‍🤝‍🧑 Field Execution | Driver/Salesperson view, navigation, delivery status |
| ✍️ POD Collection | Customer signature, photo, geolocation, time-stamped |
| 📦 Returns/Issues | Report damage, partial delivery, or customer refusal |

Web UI – Delivery Dashboard (Logistics Team)

+----------------------------------------------------------------------------------+

| 🚚 Delivery Scheduling Dashboard |

+----------------------------------------------------------------------------------+

| 📅 Date: [ 18-Sep-2025 ▼ ] | Filter: [ Vehicle ▼ ] [ Status ▼ ] | ⎘ Export CSV |

+----------------------------------------------------------------------------------+

| Delivery ID | Route / Vehicle | Orders | Driver | Status | 📍 Map | 🔧 |

|-------------|------------------|--------|-------------|---------|--------|--------|

| DLV-1025 | Route-7 / VAN-1 | 12 | Mohan K. | 🟡 In Progress | 🗺️ | 🔄 |

| DLV-1026 | Route-9 / VAN-2 | 8 | Suresh P. | ✅ Completed | 🗺️ | 🔍 |

| DLV-1027 | Unassigned | 5 | — | ❌ Pending | — | ➕ Assign |

+----------------------------------------------------------------------------------+

Mobile UI – Field Agent Delivery View

+-------------------------------------------------------------+

| 🚚 Today’s Deliveries (Route-7) |

+-------------------------------------------------------------+

| 1️⃣ ABC Retailers – ₹5,200 – 📍 2.1 km |

| 🕒 ETA: 10:30 AM | Status: ⏳ Not Delivered |

| [Start Navigation] [Mark Delivered] |

+-------------------------------------------------------------+

| 2️⃣ XYZ Mart – ₹7,800 – 📍 3.5 km |

| 🕒 ETA: 11:15 AM | Status: ✅ Delivered |

| Proof: 📷 Photo, ✍️ Signature |

+-------------------------------------------------------------+

**Proof of Delivery (POD) Collection**

Upon delivery completion, collect:

* ✅ Digital Signature (on-screen)
* 📸 Photo of delivered goods
* 📍 GPS location tag
* 🧾 Optional comments (e.g., “Delivered to security guard”)

All POD data is stored with timestamp & linked to order ID.

**Exceptions Handling**

| **Issue Type** | **Field App Action** |
| --- | --- |
| 🚫 Customer Unavailable | Mark as “Customer Not Available” – auto-reschedule |
| 🔁 Partial Return | Log returned items with reason |
| 📦 Damaged Item | Take photo + damage tag |
| ❌ Refused Delivery | Mark as “Rejected by Customer” + notes |

**Web View – POD Reports**

**[ POD Report – 18-Sep-2025 ]**

**| Order ID | Customer | Delivered By | Time | POD Type | Status |**

**|----------|------------------|--------------|-----------|--------------|---------|**

**| ORD-7890 | ABC Retailers | Mohan K. | 10:32 AM | Sign + Photo | ✅ |**

**| ORD-7891 | XYZ Mart | Mohan K. | 11:10 AM | Signature | ✅ |**

**| ORD-7892 | City Superstore | Suresh P. | — | — | ❌ Failed |**

**Smart Delivery Features**

| **Feature** | **Benefit** |
| --- | --- |
| **📦 Auto Route Optimization** | **Minimize travel time for daily deliveries** |
| **🛰️ Real-time GPS Tracking** | **Live location of field staff** |
| **📬 Delivery Notifications** | **SMS/WhatsApp alerts to customers** |
| **📎 Linked to Order/Invoice** | **Pulls from order module, confirms payment** |
| **🧾 Auto-POD Generation** | **PDF report with signature, photo, timestamp** |

**Role-Based Access**

| **Role** | **View Schedule** | **Assign Routes** | **Complete POD** | **View Reports** |
| --- | --- | --- | --- | --- |
| **Admin / CEO** | **✅ All** | **✅** | **❌** | **✅ All** |
| **Logistics Manager** | **✅ All** | **✅** | **❌** | **✅** |
| **Area Supervisor** | **✅ Region Only** | **❌** | **❌** | **✅ Team** |
| **Salesperson / Driver** | **✅ Own Only** | **❌** | **✅** | **Own Only** |
| **Technician** | **✅ Service only** | **❌** | **✅ (service POD)** | **Own Only** |

**Integrations**

| **Module** | **Integration Point** |
| --- | --- |
| **🧾 Orders** | **Orders assigned to delivery automatically** |
| **👤 Customers** | **Address, contact, delivery instructions pulled in** |
| **📊 Reporting** | **On-time %, failed attempts, returns** |

# 22. Database Schema Reference (Full Details)

This section documents every table, column, datatype, keys, references, defaults, indexes, and developer notes. Use this as the canonical schema reference.

A screenshot of a computer screen

AI-generated content may be incorrect.

A diagram of a diagram

AI-generated content may be incorrect.

## Table: users

Description: System users (salespeople, admins, drivers, etc.).

Columns:

• id (int) -- PK, identity(1,1)

• username (varchar) -- Login ID

• is\_active (char(1)) -- Default 'Y', not null, soft-delete flag

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id (self) or system user

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) -- Concurrency/log instance

Developer Notes:

Always insert createdby. Consider adding columns: email, mobile, role\_id, password\_hash, last\_login, timezone.

## Table: customers

Description: Customer master (retailers, wholesalers, distributors).

Columns:

• id (int) -- PK, identity(1,1)

• name (varchar) -- Customer display name

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

Extend with addresses, contact\_person, phone, email, credit\_limit, territory\_id.

## Table: products

Description: Product master catalog.

Columns:

• id (int) -- PK, identity(1,1)

• name (varchar) -- Product name

• product\_category\_id (int) -- FK -> product\_categories.id

• product\_subcategory\_id (int) -- FK -> product\_subcategories.id

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

Consider SKU, unit\_of\_measure, pack\_size, weight, barcode, hs\_code.

## Table: product\_categories

Description: Product category master.

Columns:

• id (int) -- PK, identity(1,1)

• name (varchar) -- Category name

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

## Table: product\_subcategories

Description: Product subcategory master linked to category.

Columns:

• id (int) -- PK, identity(1,1)

• category\_id (int) -- FK -> product\_categories.id

• name (varchar) -- Subcategory name

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

## Table: batch\_lots

Description: Batch/lot tracking for products.

Columns:

• id (int) -- PK, identity(1,1)

• product\_id (int) -- FK -> products.id

• batch\_number (varchar) -- Batch/lot identifier

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

Add manufacture\_date, expiry\_date, quantity, warehouse\_id if needed.

## Table: serial\_numbers

Description: Serialized items for warranty and traceability.

Columns:

• id (int) -- PK, identity(1,1)

• product\_id (int) -- FK -> products.id

• batch\_id (int) -- FK -> batch\_lots.id

• serial\_number (varchar) -- Unique serial for item

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

Ensure unique constraint on (product\_id, serial\_number) or serial\_number globally.

## Table: promotions

Description: Promotion master (campaigns, schemes).

Columns:

• id (int) -- PK, identity(1,1)

• name (varchar) -- Promotion name

• type (varchar) -- Promotion type (discount, bogo, rebate)

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

Add start\_date, end\_date, target\_customer\_group, terms\_and\_conditions.

## Table: promotion\_parameters

Description: Parameters/variables for promotions to make them data-driven.

Columns:

• id (int) -- PK, identity(1,1)

• promotion\_id (int) -- FK -> promotions.id

• param\_name (varchar) -- Parameter key

• param\_type (varchar) -- Type: number/string/date/percent

• param\_value (varchar) -- Stored as text

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

Used by promotion engine to compute discounts/entitlements.

## Table: sales\_targets

Description: Defines sales targets for categories or products.

Columns:

• id (int) -- PK, identity(1,1)

• product\_category\_id (int) -- FK -> product\_categories.id

• start\_date (datetime) --

• end\_date (datetime) --

• target\_amount (decimal) --

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

Can be extended for product-level, territory-level or team-level targets.

## Table: sales\_target\_assignments

Description: Links targets to sales people.

Columns:

• id (int) -- PK, identity(1,1)

• sales\_target\_id (int) -- FK -> sales\_targets.id

• sales\_person\_id (int) -- FK -> users.id

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

## Table: bonuses

Description: Bonus slabs tied to targets.

Columns:

• id (int) -- PK, identity(1,1)

• sales\_target\_id (int) -- FK -> sales\_targets.id

• min\_amount (decimal) -- Minimum achieved to qualify

• bonus\_amount (decimal) -- Bonus payable

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

Supports multiple slabs per target.

## Table: vehicles

Description: Fleet vehicles used for deliveries or staff assignment.

Columns:

• id (int) -- PK, identity(1,1)

• vehicle\_number (varchar) -- Registration/identifier

• type (varchar) -- Vehicle type (truck, van, bike)

• assigned\_to (int) -- FK -> users.id (driver or custodian)

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

Extend with capacity, fuel\_type, registration\_expiry, insurance\_details.

## Table: visits

Description: Records field visits/interaction between salesperson and customer.

Columns:

• id (int) -- PK, identity(1,1)

• customer\_id (int) -- FK -> customers.id

• sales\_person\_id (int) -- FK -> users.id

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

Add fields: visit\_time, purpose, notes, geo\_lat, geo\_long, visit\_outcome, orders\_taken.

## Table: coolers

Description: Assets (coolers) installed at customer premises.

Columns:

• id (int) -- PK, identity(1,1)

• customer\_id (int) -- FK -> customers.id

• code (varchar) -- Asset code or tag

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

Track asset status, installation\_date, last\_service\_date.

## Table: cooler\_inspections

Description: Inspection records for coolers.

Columns:

• id (int) -- PK, identity(1,1)

• cooler\_id (int) -- FK -> coolers.id

• inspected\_by (int) -- FK -> users.id

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

Add inspection\_date, findings, action\_taken, photos\_path.

## Table: approval\_workflows

Description: Workflow header for approvals (pricing, discounts, credits).

Columns:

• id (int) -- PK, identity(1,1)

• workflow\_type (varchar) -- Type of workflow

• reference\_id (int) -- Reference PK (e.g., order\_id)

• reference\_type (varchar) -- E.g., 'order', 'promotion'

• reference\_number (varchar) -- External or human-readable ref no.

• requested\_by (int) -- FK -> users.id

• request\_date (datetime) --

• priority (varchar) --

• status (varchar) -- e.g., pending, approved, rejected

• current\_step (int) --

• total\_steps (int) --

• request\_data (text) -- JSON or text payload

• final\_approved\_by (int) -- FK -> users.id

• final\_approved\_at (datetime) --

• rejected\_by (int) -- FK -> users.id

• rejected\_at (datetime) --

• rejection\_reason (varchar) --

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

request\_data should use structured JSON. Ensure audit trail.

## Table: workflow\_steps

Description: Steps belonging to an approval workflow header.

Columns:

• id (int) -- PK, identity(1,1)

• workflow\_id (int) -- FK -> approval\_workflows.id

• step\_number (int) --

• step\_name (varchar) --

• assigned\_role (varchar) -- Role allowed to process step

• assigned\_user\_id (int) -- FK -> users.id

• status (varchar) -- pending/processed/ skipped

• comments (varchar) --

• processed\_by (int) -- FK -> users.id

• processed\_at (datetime) --

• due\_date (datetime) --

• is\_required (char(1)) -- Default 'Y', not null

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

Used by workflow engine; consider indexing workflow\_id and assigned\_user\_id.

## Table: customer\_documents

Description: Documents uploaded against customers for compliance.

Columns:

• id (int) -- PK, identity(1,1)

• customer\_id (int) -- FK -> customers.id

• document\_type (varchar) --

• document\_number (varchar) --

• issue\_date (datetime) --

• expiry\_date (datetime) --

• issuing\_authority (varchar) --

• file\_path (varchar) -- Storage path or object key

• is\_verified (char(1)) -- Default 'N', not null

• verified\_by (int) -- FK -> users.id

• verified\_at (datetime) --

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

Consider storing hash/metadata for file integrity and retention policies.

## Table: digital\_signatures

Description: E-signature container for documents and approvals.

Columns:

• id (int) -- PK, identity(1,1)

• document\_id (int) -- Reference id of signed object

• document\_type (varchar) -- Type of document signed

• signature\_data (text) -- Base64 or structured signature payload

• signer\_name (varchar) --

• signer\_email (varchar) --

• signer\_role (varchar) --

• signed\_at (datetime) --

• ip\_address (varchar) --

• device\_info (varchar) --

• latitude (decimal(10,8)) --

• longitude (decimal(11,8)) --

• is\_valid (char(1)) -- Default 'Y', not null

• verification\_hash (varchar) -- Hash to verify integrity

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

Protect signature\_data; consider encryption at rest and access control.

## Table: price\_history

Description: Tracks price changes across customers/territories.

Columns:

• id (int) -- PK, identity(1,1)

• product\_id (int) -- FK -> products.id

• customer\_group\_id (int) -- FK -> customer\_groups.id (optional)

• territory (varchar) --

• old\_price (decimal(18,2)) --

• new\_price (decimal(18,2)) --

• effective\_from (datetime) --

• effective\_to (datetime) --

• change\_reason (varchar) --

• changed\_by (int) -- FK -> users.id

• approved\_by (int) -- FK -> users.id

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

Useful for audits and rollback of pricing changes.

## Table: customer\_groups

Description: Defines business segments and pricing groups.

Columns:

• id (int) -- PK, identity(1,1)

• name (varchar) --

• code (varchar) --

• description (varchar) --

• discount\_percentage (decimal(5,2)) --

• credit\_terms (int) --

• payment\_terms (varchar) --

• price\_group (varchar) --

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

## Table: customer\_group\_members

Description: Mapping customers into groups.

Columns:

• id (int) -- PK, identity(1,1)

• customer\_group\_id (int) -- FK -> customer\_groups.id

• customer\_id (int) -- FK -> customers.id

• joined\_at (datetime) --

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Indexes:

• (customer\_group\_id, customer\_id) UNIQUE

Developer Notes:

Enforce uniqueness to prevent duplicate membership.

## Table: delivery\_schedules

Description: Schedules for delivery orders to customers.

Columns:

• id (int) -- PK, identity(1,1)

• order\_id (int) -- Reference to sales order

• customer\_id (int) -- FK -> customers.id

• scheduled\_date (datetime) --

• scheduled\_time\_slot (varchar) --

• assigned\_vehicle\_id (int) -- FK -> vehicles.id

• assigned\_driver\_id (int) -- FK -> users.id

• status (varchar) --

• priority (varchar) --

• delivery\_instructions (varchar) --

• actual\_delivery\_time (datetime) --

• delivery\_proof (varchar) -- File path or object key

• customer\_signature (text) --

• failure\_reason (varchar) --

• rescheduled\_date (datetime) --

• is\_active (char(1)) -- Default 'Y', not null

• createdate (datetime) -- Default GETDATE()

• createdby (int) -- Not null, FK -> users.id

• updatedate (datetime) --

• updatedby (int) --

• log\_inst (int) --

Developer Notes:

Integrate with route optimization and vehicle availability.

# Additional Recommendations

1. Add audit/user metadata where missing (role\_id, email, phone).  
2. Use foreign key constraints and cascade rules carefully (prefer NO ACTION/RESTRICT on deletes).  
3. Add indexes on foreign key columns and frequently filtered columns (is\_active, product\_category\_id, customer\_id).  
4. Implement soft deletes and retention policy for documents and signatures.  
5. Consider partitioning large history tables (price\_history, serial\_numbers) by date for performance.  
6. Apply encryption-at-rest for sensitive columns (signature\_data, verification\_hash, file paths if containing secrets).