**📘 Project Plan Prompt – Portable POS System (Python Edition) – v2.0**

**📌 Project Title:**

**Portable POS System**

**🎯 Objective / Purpose**

To develop a **modular, fully offline Point-of-Sale (POS) system** using **Python + PySide6** designed for **Windows 11 touchscreen devices (1920×1080)**.

The application will operate **without internet/cloud dependencies** and will support:

* Point-of-Sale Transactions
* Inventory, Invoicing, and CRM
* Promotions (Promo Codes, Rewards, Vouchers)
* Order management with support for **Returns**
* Expense tracking and Supplier/Biller records
* Print preview and auto-generated receipts
* Dual Time Display: Local Time + Philippine Time

**Ideal Use Case:** SMEs, mobile vendors, water refilling stations, field merchants, or any business needing a non-cloud, reliable desktop POS system.

**🧱 Framework & Platform**

| **Category** | **Specification** |
| --- | --- |
| OS | Windows 11 (x64) |
| Language | Python 3.10+ |
| GUI Framework | PySide6 (Qt Designer .ui-based) |
| DB | SQLite3 |
| Styling | QSS Glassmorphic (Neon Blue, Frosted Panels) |
| Receipt Engine | HTML rendered via QWebEngineView |
| Output Format | PDF (QPrinter) |
| Packaging | PyInstaller (.exe format) |
| Resolution | Fixed 1920×1080 |
| Touchscreen | Fully optimized with large widgets |
| Clock Display | Local Time + GMT+8 (Philippine Time) |

**📁 Project Folder Structure**

D:\Portable\_POS\_Python\

├── brain\ # Python logic per module

│ ├── pos.py

│ ├── orders.py

│ ├── returns.py

│ ├── inventory.py

│ └── ...

├── memory\ # Local SQLite DB + JSON config

│ ├── main.db

│ └── config.json

├── ui\ # Qt Designer .ui files

│ ├── main\_window.ui

│ └── modules\

│ ├── pos\_new\_order.ui

│ ├── orders.ui

│ ├── returns.ui

│ └── ...

├── static\ # QSS, images, fonts, logos

├── templates\ # HTML templates for receipts, invoices

├── splash.py # Startup screen

├── main.py # App launcher

├── app\_context.py # Singleton global state manager

├── validators.py # Shared validators (e.g., mobile format)

**🧩 Development Phases**

| **Phase** | **Description** |
| --- | --- |
| 1 | Project structure + splash + main window |
| 2 | Sidebar nav via QStackedWidget |
| 3 | UI blueprints for POS, CRM, Inventory |
| 4 | SQLite backend integration + validator logic |
| 5 | Add Print Preview, Rewards logic, Export features |
| 6 | Full Testing, .exe Packaging via PyInstaller |

**🎨 UI/UX Guidelines**

| **Element** | **Spec** |
| --- | --- |
| Resolution | 1920×1080 fixed layout |
| Sidebar Width | 300px (glassmorphic) |
| Content Width | 1620px |
| Forms Scroll | ❌ No scrolling (forms must fit) |
| Table Scroll | ✅ Scrollable |
| Button Size | Touchscreen-friendly |
| Fonts/Icons | Custom neon-glow style |
| Receipt Style | 80mm thermal (HTML) |
| Layout Engine | Qt Designer + QStackedWidget |
| Time Display | Top-right corner: Local Time + GMT+8 |

**📂 Sidebar Navigation**

**Updated order with Returns module inserted:**

| **Main Menu** | **Submenus (if any)** |
| --- | --- |
| 📊 Dashboard | — |
| 🛒 Point of Sale | ➕ New Order ♻️ Update Order |
| 🎁 Promotions | 💸 Promos 🎉 Rewards 🎟️ Vouchers |
| 📦 Orders | — |
| 🔁 Returns | 📥 Return Orders 📦 Return Items |
| 💰 Expenses | — |
| 🧾 Invoicing | 🛠️ Supplies 🧹 Maintenance 💳 Billing |
| 📚 Inventory | — |
| 👥 Customers | — |
| 📜 Rewards Ledger | — |
| 🏦 Suppliers/Billers | 🧑‍💼 Suppliers 🧾 Billers 🏛️ Banks |
| 📅 Calendar | — |
| 📈 Reports | — |
| 🛠️ Settings | — |

**🔢 Auto ID Format System**

| **Type** | **Format Example** |
| --- | --- |
| Customer | C-YYYYMMDD-#### |
| Order | O-YYYYMMDD-#### |
| Return | 0000000001 (10-digit padded) |
| Invoice | I-YYYYMMDD-#### |
| Supplier | S-YYYYMMDD-#### |
| Biller | B-YYYYMMDD-#### |
| Rewards | 0000000001 (10-digit padded) |
| Voucher | 0000000001 (10-digit padded) |

* All IDs reset **daily** at 00:00 GMT+8.
* Only generated via backend logic — not editable from UI.

**🔒 Offline-First Design Rules**

| **Rule** | **Enforced** |
| --- | --- |
| No internet requirement | ✅ |
| No pip/npm runtime dependencies | ✅ |
| All data saved locally | ✅ |
| No syncing across PCs | ✅ |
| Optional PDF printing only | ✅ |
| All features must work offline | ✅ |

**💡 Core Design Decisions (Confirmed)**

| **Area** | **Implementation** |
| --- | --- |
| Navigation | QStackedWidget |
| App State | AppContext singleton |
| Validators | validators.py (mobile, ID, email, etc.) |
| Print Engine | QWebEngineView (HTML to PDF or printer) |
| Exports | .csv and .xlsx via pandas |
| Storage | SQLite local file |
| POS receipt print | Default to PDF, thermal printer optional |
| Discount / VAT | VAT toggleable, rate editable via Settings |
| Touch UI | Fully supported |

**✅ System Scope Summary**

| **Concern** | **Status** |
| --- | --- |
| <10,000 records supported | ✅ |
| 3+ years of data planned | ✅ |
| Single-user only | ✅ |
| Hardware printer optional | ✅ |
| Export & reporting built-in | ✅ |
| Reward point system | ✅ |
| Voucher & Promo codes | ✅ |
| Invoice-linked expenses | ✅ |
| Order Returns w/ tracking | ✅ |
| Fully offline mode | ✅ |

Let me know when you're ready to proceed with:

* 🔁 **Return Orders** & **Return Items** blueprint
* 📅 **Calendar** module
* 📈 **Reports** module
* or the finalized 💳 **POS – Update Order** module.

**🛠️ Settings Module – Developer Blueprint (v1.3)**

**📄 Page Title: Settings**

**🎯 Purpose**

Centralized configuration for managing system-wide settings:

* Business identity
* Rewards logic and expiry
* POS behavior (VAT, discounts, printing)
* Receipt & invoice formatting
* Print settings
* Future: User roles & permissions

Stored locally with no internet dependency. Modular per-section save logic.

**🧱 Layout Overview**

| **Component** | **Description** |
| --- | --- |
| Sidebar | 300px — tab buttons (glassmorphic style) |
| Main Panel | 1320px — loads section via QStackedWidget |
| Scroll Behavior | Scrollable only on content area |
| Save Mechanism | Each section has its own Save button |
| Style | Neon-blue glassmorphic theme on fixed 1920×1080 layout |

**🔹 Sidebar Tabs (Settings Sections)**

**1. General Settings**

Core business identity and display settings. Also includes VAT settings.

| **Field** | **Type** | **Notes** |
| --- | --- | --- |
| Business Name | Text | Required — shown on receipts/invoices |
| Business Motto | Text | Optional — shown below name |
| Branch Name | Text | Optional |
| Contact Number | Text | Optional |
| Business Address | TextArea (3 lines) | Required |
| Timezone | Dropdown | Default: GMT+8 (Asia/Manila) |
| Currency Symbol | Text | Default: ₱ |
| **VAT Rate (%)** | Numeric (0–100) | Default: 12% (editable). Affects receipts and invoice calculations |
| ✅ Save Button | — | Commits all above fields to database |

**2. Rewards Conversion Settings**

| **Field** | **Type** | **Notes** |
| --- | --- | --- |
| Points to Amount | Numeric | e.g., 1 Point = ₱1 |
| Effective Date | Date Picker | Default: today |
| Display Label | Read-only | “1 Point = ₱1 since MM/DD/YYYY” |
| Set New Rate | Modal | Confirmation + save to history |
| History | SQLite | For Rewards Ledger reference |
| ✅ Save Button | — | Validates and saves only when changed |

**3. Rewards Global Validity**

| **Field** | **Type** | **Notes** |
| --- | --- | --- |
| Validity in Days | Numeric | e.g., 180 days |
| Effective Date | Date Picker | Default: today |
| Display Label | Read-only | “Points expire after X days from date earned” |
| ✅ Save Button | — | Used by Rewards Ledger expiry logic |

**4. Printer Configuration**

| **Field** | **Type** | **Notes** |
| --- | --- | --- |
| POS Printer Type | Dropdown | 80mm / Half A4 / Full A4 |
| Invoice Print Size | Dropdown | Half A4 or Full A4 |
| Auto-Print on Save | Toggle | Yes / No |
| Enable Print Preview | Toggle | Yes / No |
| ✅ Save Button | — | Controls print behavior system-wide |

**5. ID Format Preview (Read-only)**

| **Entity** | **Format** |
| --- | --- |
| Customer | C-YYYYMMDD-#### |
| Order | O-YYYYMMDD-#### |
| Invoice | I-YYYYMMDD-#### |
| Supplier | S-YYYYMMDD-#### |
| Biller | B-YYYYMMDD-#### |
| 🔒 Format fixed — auto-generated in backend only |  |

**6. System Info (Read-only)**

| **Field** | **Details** |
| --- | --- |
| App Version | e.g., v1.0.0 |
| Last Updated | File or code metadata |
| DB File Path | Path to local main.db |
| Storage Used | File size estimate |
| System Mode | Always "Offline" |

**7. User Roles (Future)**

UI available but locked in v1.0.

| **Field** | **Description** |
| --- | --- |
| User Roles | Admin, Manager, Cashier (future) |
| Permissions | Per-module |
| Users List | Username, Role, Status |
| Password Policy | Optional security config |

**🖨️ Print Preview & Formatting**

**🧾 POS Receipt Preview (80mm)**

| **Option** | **Type** |
| --- | --- |
| Show Logo | Toggle |
| Business Header | Editable |
| Date Format | Dropdown |
| Include Customer Info | Toggle |
| Include VAT Breakdown | Toggle |
| Footer Message | TextArea |
| Show OR Number | Toggle |
| ✅ Save Button | Saves receipt style |
| 🔘 Test Print | Simulated preview via QWebEngineView |

**📄 Invoice Preview (Half A4 / Full A4)**

| **Option** | **Type** |
| --- | --- |
| Print Size | Dropdown |
| Show Logo | Toggle |
| Show Branch Info | Toggle |
| Header Title | Text (default: INVOICE) |
| Include OR Number | Toggle |
| Include Signature | Toggle |
| Show Notes Section | Toggle |
| Border Lines | Toggle |
| ✅ Save Button | Saves format config |
| 🔘 Test Print | Live invoice rendering |

**🧠 Backend API – /brain/settings.py**

| **Method** | **Endpoint** | **Purpose** |
| --- | --- | --- |
| GET | /api/settings | Get all settings |
| POST | /api/settings/update | Update section |
| GET | /api/settings/conversion | View historical conversion rates |
| POST | /api/settings/conversion/update | Set new conversion rate |
| POST | /api/settings/validity/update | Set reward expiry days |
| GET | /api/settings/printer | Load printer config |
| POST | /api/settings/printer/update | Save printer config |
| GET | /api/settings/print | Get formatting options |
| POST | /api/print/preview/receipt | Render receipt HTML |
| POST | /api/print/preview/invoice | Render invoice HTML |

**📂 Data Storage – SQLite**

**Table: settings**

| **Column** | **Type** | **Example** |
| --- | --- | --- |
| key | TEXT | general.vat\_rate |
| value | TEXT/JSON | "12" |
| updated | DATETIME | 2025-08-04 21:14:03 |

**Table: conversion\_history**

| **id** | **rate** | **effective\_date** |
| --- | --- | --- |
| 1 | 1.0 | 2025-07-01 |

**Table: validity\_settings**

| **id** | **days** | **effective\_date** |
| --- | --- | --- |
| 1 | 180 | 2025-07-01 |

**✅ Summary**

| **Feature** | **Status** |
| --- | --- |
| Business details + VAT + motto | ✅ |
| Rewards conversion + validity | ✅ |
| Full print format controls | ✅ |
| Receipt/invoice print preview | ✅ |
| Fully offline, modular saving | ✅ |
| Export-ready system config | ✅ |
| Role-based access (future-ready) | ✅ |

**👥 Customers Module – Developer Blueprint v3.1**

**Page Title:** Customers  
**Screen Target:** 1920×1080 (Max Content Width: 1620px)  
**UI Style:** Glassmorphic neon-blue, responsive, fixed-width layout

**🧭 Purpose**

Manages all customer records, including profile details, order statistics, reward points, CRM status, import/export, and a full audit log of changes. Also supports **Walk-In customers** added from the POS module with minimal data.

**🧱 Layout Structure**

| **Section** | **Height Ratio** | **Scroll** |
| --- | --- | --- |
| Customer Form Tile | 1/3 | ❌ No |
| Customer Table Tile | 2/3 | ✅ Yes |

**👤 Customer Form Tile**

| **Field** | **Type** | **Validation / Notes** |
| --- | --- | --- |
| First Name | Text | Required |
| Last Name | Text | Optional |
| Mobile Number | Text | Required, format: 09xxxxxxxxx or +639xxxxxxxxx |
| Email | Text | Optional, valid email format |
| Address | TextArea | ✅ Required by default — ❌ Optional if Source = Walk-In |
| Source | Multi-select | Default: SMS; options: SMS, Walk-In, FB Chat |
| Type | Dropdown | Default: Silver; options: VIP, Gold, Silver, Wholesaler |
| Status | Dropdown | Default: Active; options: Active, Inactive, Blacklisted |

* ⚠️ If Source includes **Walk-In**, the Address field becomes optional (visual cue: "(optional for Walk-Ins)")
* Save and Cancel buttons appear only after edits

**📋 Table Header Controls**

* 🔍 **Search Box**
* 🔽 **Search Field Dropdown**: Any, ID, Name, Mobile, Email, Address
* ✅ **Search Button**
* 📌 **Status Filter** (checkbox + dropdown)
* 🏷️ **Type Filter** (checkbox + dropdown)
* 🔄 **Interchange Button** – swaps order of Status and Type filters
* 📥 **Import Button** – opens modal
* 📤 **Export Button**
* 🔢 **Show List Selector** – 10 / 25 / 50 (default: 10)

**📊 Table Columns**

| **Column** | **Description** |
| --- | --- |
| Customer ID | Format: C-YYYYMMDD-#### (#### resets daily) |
| Name | First + Last Name |
| Mobile | Validated Philippine number |
| Email | Optional |
| Address | Full Address (blank if Walk-In and not filled) |
| Source | CSV style e.g., SMS, Walk-In |
| Lifetime Orders | Total # of completed orders |
| Lifetime Sales | ₱ value of all completed orders |
| Active Rewards | Active Points (e.g., “85 PTS = ₱85”) |
| Type | Silver / VIP / Gold / Wholesaler |
| Status | Active / Inactive / Blacklisted |
| Date Joined | Format: August 02, 2025 (stored as MM/DD/YYYY) |
| Action | “View Profile” button (opens slide-up panel) |

✅ Table supports pagination, sorting, and multi-filter search

**📥 Import Modal**

* Fields required in file:  
  First Name, Last Name, Email, Mobile, Address, Source
* Valid Source values: SMS, Walk-In, FB Chat
* Address field: ✅ Required unless Source includes Walk-In
* Result after import:  
  “✔️ 46 imported, 2 updated, 1 skipped (invalid phone)”

**📤 Export Behavior**

* Format: CSV & XLSX
* Filename: customers\_export\_YYYYMMDD-HHMMSS.csv
* Includes all visible or filtered rows

**🔍 View Profile Modal (Slide-Up – 95% Height)**

**📄 Section 1: Customer Profile**

| **Field** | **Notes** |
| --- | --- |
| Customer ID | Read-only |
| Date Joined | Full formatted date |
| First/Last | Editable |
| Mobile/Email | Editable |
| Address | Editable (optional if Walk-In) |
| Source | Editable multi-select |
| Type / Status | Editable |
| Active Points | Auto-pulled from Rewards Ledger |

**📈 Section 2: Sales Summary**

| **Metric** | **Notes** |
| --- | --- |
| Lifetime Sales | ₱ Total of all completed orders |
| Orders This Week/Month | Based on system time |
| On Hold Orders | Only shown if >0 |
| 🔗 Button | “Go to Orders” → Filters Orders module by ID |

**🎁 Section 3: Rewards Summary**

* Shows:
  + Global Conversion Rate: e.g., “1 Point = ₱1”
  + Validity Duration: e.g., “Valid for 180 days”
  + Lifetime / Active / Used / Expired Points (with ₱ value)
* 🔗 Button: “Go to Rewards Ledger”

**🧾 Section 4: Change Logs**

| **Column** | **Description** |
| --- | --- |
| Date | Timestamp (MM/DD/YYYY hh:mm:ss) |
| From | Old value |
| To | New value |

✅ Fields tracked: Name, Email, Mobile, Address, Source, Type, Status

**🧠 Backend Endpoints**

| **Endpoint** | **Purpose** |
| --- | --- |
| GET /api/customers | Fetch paginated, filtered list |
| POST /api/customers | Create new customer |
| PUT /api/customers/:id | Update profile |
| GET /api/customers/export | Export customers |
| POST /api/customers/import | Validate and import file |
| GET /api/customers/:id | Load full profile for viewing |
| GET /api/customers/:id/logs | View change history |
| POST /api/customers/logs | Save change log entry |

**✅ Feature Checklist**

| **Feature** | **Status** |
| --- | --- |
| Walk-In creation via POS | ✅ |
| Address optional for Walk-In source | ✅ |
| Profile with Rewards + Orders Summary | ✅ |
| Slide-up View/Edit Profile with change logs | ✅ |
| Import/export with validations | ✅ |
| Type & Status filtering + swap | ✅ |
| Full audit logs per customer | ✅ |

**📚 Inventory Module – Updated Blueprint (v2.0)**

📄 **Page Title:** Inventory Management  
🎯 **Purpose:** To manage item records and track stock quantities, reorder levels, and movement history—including supplies from invoices.

**🧱 Layout Structure**

| **Panel** | **Width** | **Description** |
| --- | --- | --- |
| Left | 300px | Item Groups tile |
| Right | ~1320px | Item Table + Top Controls |

**🔍 Top Controls (Above Table)**

| **Control** | **Behavior** |
| --- | --- |
| 🔍 Search Bar | Search by Item Name, SKU, Barcode |
| 🔽 Filter by Group | Filters items by group |
| 📤 Export Button | Exports all visible inventory data |
| 📥 Import Button | Imports inventory entries (basic info only) |
| 🔄 Show List Dropdown | 10 / 25 / 50 (default: 10) |

**📋 Inventory Table Columns**

| **Column** | **Description** |
| --- | --- |
| SKU (Item ID) | 6-digit, starts with Group ID (e.g., 02 → 020001) |
| Item Name | Full name of item |
| Description | Optional |
| Unit Price (₱) | Selling price |
| Current Stock | Updated automatically |
| Reorder Level | Threshold level |
| Item Group | Group Name (ID) |
| Barcode | Optional |
| Status | Active / Archived |
| Action | Edit only (no delete) |

**➕ Add / Edit Item Modal**

| **Field** | **Behavior** |
| --- | --- |
| Item Group | Required dropdown |
| Item ID (SKU) | Auto-generated, readonly |
| Item Name | Required |
| Description | Optional |
| Unit Price | Required |
| Current Stock | Required |
| Reorder Level | Required |
| Barcode | Optional |
| Status | Default: Active |
| Save Button | Validates & updates item |

🔒 Changing **Item Group** is **disabled** (affects SKU logic)

**📁 Item Group Panel (Left)**

* Shows:
  + Group Name (Group ID)
* "All Items" default option
* Edit / Add / Archive Group Modal
  + Fields: Group ID (2-digit), Group Name (text)

**🧮 Stock Movement Tracking – New Section**

📂 **Stock History Button** (per item)

* Opens a **slide-up panel** showing stock movement history:

**🔁 Stock Movement Log Table**

| **Column** | **Description** |
| --- | --- |
| Date | Timestamp of transaction |
| Type | Source of change (e.g., Supply Invoice, POS Sale, Manual Adj.) |
| Reference | ID or label (e.g., I-20250801-0001 or O-20250802-0002) |
| Qty Changed | + for addition, − for deduction |
| Final Stock | Resulting stock after change |
| Note | Optional remark (e.g., “Received from supplier”) |

🔎 Filters:

* Date Range
* Type (Supply, Sale, Manual, Return)
* Item Reference

📤 Export button for filtered movement history

**⚠️ Stock Update Logic**

Integrated with:

| **Module** | **Behavior** |
| --- | --- |
| 🧾 Supplies | Adds quantity to Current Stock; logs entry in movement log |
| 🛒 POS | Deducts from stock if Confirmed; logs sale |
| 🔁 Return (Orders) | Adds back stock for returned items |
| 🛠 Update Order | Restocks canceled Hold orders |
| ✏️ Manual Edit | Logs manual changes via UI (future feature) |

📦 Movement entries written to stock\_movements table on every relevant transaction.

**🧠 Backend Tables (Additional)**

**stock\_movements**

| **Column** | **Type** |
| --- | --- |
| id | INT (PK) |
| item\_id | TEXT |
| date | DATETIME |
| type | TEXT (Sale, Supply, Return, Manual) |
| reference\_id | TEXT |
| qty\_changed | INTEGER |
| final\_stock | INTEGER |
| note | TEXT |

✅ All logs pulled using item\_id and filters  
✅ Exportable to CSV/XLSX

**📤 Import Behavior (No change)**

* Fields:
  + Item Group ID (required)
  + Item Name (required)
  + Unit Price, Stock, Reorder Level
  + Description / Barcode (optional)
* Duplicate check: Name + Group + Barcode

**✅ Feature Recap**

| **Feature** | **Status** |
| --- | --- |
| Inventory table with filters & search | ✅ |
| Add/Edit item with SKU lock | ✅ |
| Import/Export | ✅ |
| Stock auto-update from POS & Supplies | ✅ |
| Stock movement log w/ export | ✅ |
| Group-based item filtering | ✅ |
| Glassmorphic, touch-optimized UI | ✅ |

**🏛️ Banks Submodule – Developer Blueprint (v1.0)**

📄 **Page Title:** Philippine Banks List

**🎯 Purpose**

Manage a master list of official Philippine banks, used by supplier/biller/invoicing modules. Fully editable, searchable, and importable.

**🖥️ Layout Overview**

* **Full-width layout** (1320px)
* **Search bar** – top left (filters by Bank Name or Short Name)
* **Control Buttons** – top right:  
  ➕ Add ✏️ Edit 🗑️ Delete 📥 Import 📤 Export

**📋 Table Columns**

| **Column** | **Description** |
| --- | --- |
| **Bank Name** | Full official name (required) |
| **Short Name** | Common abbreviation (required) |
| **Action** | Edit / Delete buttons |

* Row click **does not** open edit — explicit action buttons used
* Columns auto-resize based on content length

**➕ Add / ✏️ Edit Bank Modal**

| **Field** | **Type** | **Validation** |
| --- | --- | --- |
| Bank Name | Text | Required, unique |
| Short Name | Text | Required, unique |

* 🟦 Save button validates both fields
* On Edit: fields are pre-filled

**🗑️ Delete Bank**

* 🔒 Requires confirmation modal:

“Are you sure you want to delete [Bank Name]? This cannot be undone.”

**📥 Import (CSV/XLSX)**

| **Column** | **Notes** |
| --- | --- |
| Bank Name | Required |
| Short Name | Required |

* **Duplicate Check:** Based on Bank Name *or* Short Name
* **Behavior:** Skip and log duplicates
* **Summary Example:**

12 added, 3 duplicates skipped, 1 error

📁 **File Upload Modal** with:

* Import format guide
* File basket (drag/drop)
* Import / Cancel buttons

**📤 Export (CSV/XLSX)**

| **Behavior** | **Value** |
| --- | --- |
| Scope | Full list (filtered or unfiltered) |
| Fields | Bank Name, Short Name |
| Filename | banks\_export\_YYYYMMDD-HHMMSS.csv |

**🧠 Backend API Endpoints**

| **Method** | **Endpoint** | **Description** |
| --- | --- | --- |
| GET | /api/banks | Get all banks |
| POST | /api/banks | Add a bank |
| PUT | /api/banks/:id | Edit existing bank |
| DELETE | /api/banks/:id | Delete a bank |
| POST | /api/banks/import | Import banks (CSV) |
| GET | /api/banks/export | Export banks |

**🛠 Validations & Notes**

* Both fields are **required and unique**
* Allow only **official Philippine banks** (manual validation or controlled list in future)
* Max length:
  + Bank Name: 100 chars
  + Short Name: 20 chars
* Delete only possible if not referenced by any supplier or invoice (optional in v1.0)

**🧑‍💼 Suppliers Submodule – Developer Blueprint (v1.0)**

📄 **Page Title:** Supplier Management

**🎯 Purpose**

Manage supplier records used across inventory and billing systems. Supports import/export, status tagging (instead of deletion), and local-only storage.

**🖥️ Layout Overview**

* **Full-width table layout** (1320px wide)
* **Search bar** top left – filters by:
  + Supplier Name
  + Supplier ID
  + Contact Person
  + Email
* **Control buttons** top right:
  + ➕ Add Supplier
  + 📥 Import
  + 📤 Export

**📋 Table Columns**

| **Column** | **Description** |
| --- | --- |
| **Supplier ID** | Auto-generated: S-YYYYMMDD-#### |
| **Supplier Name** | Required name of supplier |
| **Contact Person** | Optional contact person name |
| **Email** | Optional, must be valid format |
| **Phone Number** | Required, 09xxx or +639xxx format |
| **Address** | Optional multiline |
| **Notes** | Optional notes, max 200 chars |
| **Status** | One of: Active, Inactive, Delisted |
| **Action** | ✏️ Edit button |

* Columns auto-adjust to realistic length
* Default sorting: Newest Supplier ID first

**➕ Add / ✏️ Edit Supplier Modal**

| **Field** | **Type** | **Required** | **Validation** |
| --- | --- | --- | --- |
| Supplier Name | Text | ✅ | Max 100 chars |
| Contact Person | Text | ❌ | Optional |
| Email | Text | ❌ | Must match email format |
| Phone Number | Text | ✅ | Must be 09xxx... or +639xxx... |
| Address | Textarea | ❌ | Multiline supported |
| Notes | Textarea | ❌ | Max 200 chars |
| Status | Dropdown | ✅ | Active (default), Inactive, Delisted |
| Save Button | Action | ✅ | All validations apply |

**📥 Import Details**

**Accepted CSV/XLSX Fields:**  
Supplier Name (required)  
Contact Person (optional)  
Email (optional)  
Phone Number (required)  
Address (optional)  
Notes (optional)  
Status (optional) – defaults to Active if missing

**Behavior:**

* Checks for duplicate Supplier Name + Email combo
* Skips or updates existing ones based on system rules
* **Result Modal Example:**

“12 added, 3 skipped (duplicates), 1 invalid email”

**Import Modal Includes:**

* Format instructions
* File basket (drag and drop)
* Import and Cancel buttons

**📤 Export Details**

| **Attribute** | **Description** |
| --- | --- |
| Scope | All visible suppliers (filtered or not) |
| Format | CSV and XLSX |
| Filename Format | suppliers\_export\_YYYYMMDD-HHMMSS.csv |

**🔍 Table Filter Features**

| **Feature** | **Description** |
| --- | --- |
| Search Input | With dropdown: ID, Name, Person, Email |
| Status Filter | Active / Inactive / Delisted |
| Show List Selector | 10 / 25 / 50 per page |
| Export + Import | Visible at top right |
| Pagination | Prev / Next + page numbers |

**🧠 Backend API Endpoints**

| **Method** | **Endpoint** | **Description** |
| --- | --- | --- |
| GET | /api/suppliers | Get list of suppliers |
| POST | /api/suppliers | Add a new supplier |
| PUT | /api/suppliers/:id | Update supplier info/status |
| POST | /api/suppliers/import | Import CSV/XLSX |
| GET | /api/suppliers/export | Export list to CSV/XLSX |

**🛠 Validation Rules & Notes**

* **Phone Number:** must match PH format (09xxx or +639xxx)
* **Email:** if present, must match standard email regex
* **No delete function:** instead, use **Status: Active / Inactive / Delisted**
* **Supplier ID Format:** S-YYYYMMDD-#### (daily reset counter)
* **Auto-generate IDs on creation** (readonly on edit)

**🧾 Billers Submodule – Developer Blueprint (v1.0)**

📄 **Page Title:** Maintenance Companies (Billers) Management

**🎯 Purpose**

Manage and maintain a detailed list of billers—companies that issue maintenance, operational, or expense invoices. Allows advanced profile tracking, contact person management, and secure status-based archiving (no deletion).

**🖥️ Layout Overview**

* Full-page table layout (1620px wide max)
* Top bar includes:
  + 🔍 Search input + dropdown (Company Name, Biller ID, Contact Person, Email)
  + ⬇️ Export button
  + ⬆️ Import button
  + ➕ Add Biller button

**📋 Biller Table Columns**

| **Column** | **Description** |
| --- | --- |
| **Biller ID** | Auto-generated B-YYYYMMDD-#### format |
| **Company Name** | Required |
| **Contact Number** | Required, PH format (09xxx or +639xxx) |
| **Email** | Optional |
| **Primary Contact Person** | Pulled from Contacts table (with isPrimary) |
| **Primary Contact Number** | Auto-filled from selected primary contact |
| **Address** | Required |
| **Type** | Maintenance, Operational, Expense, Others |
| **Status** | Active (default), Inactive, Delisted |
| **Action** | 🔍 View Profile (slide-up modal) |

* Column widths auto-adjust to field content
* Pagination + Show List selector (10 / 25 / 50)
* Table is filtered by Status and Type (checkbox toggled with 🔄 interchange)

**➕ Add / ✏️ Edit Biller Modal**

| **Field** | **Required** | **Validation / Notes** |
| --- | --- | --- |
| Company Name | ✅ | Max 100 chars |
| Contact Number | ✅ | Must match PH mobile format |
| Email | ❌ | Valid format if provided |
| Primary Contact Person | ❌ | Can be left blank |
| Primary Contact Number | ❌ | If person provided, number is optional |
| Address | ✅ | Multiline, max 3 lines |
| Type | ✅ | Dropdown: Maintenance, Operational, Expense, Others |
| Status | ✅ | Default: Active |

**🔍 View Biller Profile (Slide-up Modal – 95% height)**

Split into 3 parts:

**1. 📄 Biller Basic Info**

* Fields:
  + Biller ID (readonly)
  + Company Name
  + Contact Number
  + Email
  + Type
  + Status
* Editable inline with Save / Cancel buttons on change

**2. 👥 Contact Persons**

| **#** | **Name** | **Contact Number** | **Channel** | **Status** | **Is Primary** | **Action** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | Juan dela Cruz | 0917xxxxxxx | SMS/Call | Active | 🔘 | Edit |
| 2 | Jane S. | 0918xxxxxxx | Email | Delisted | ⚪ | Edit |

* #: internal unique per-biller ID
* **Channel:** Dropdown: SMS/Call, FB Chat, Email
* **Status:** Dropdown: Active, Delisted
* **Is Primary:** Radio button (only one active primary)
  + ⚠️ Delisted cannot be primary
* **Action:** Inline "Edit" per row
* ✅ Add Contact button
  + Adds a new blank row (auto-assigns #)
  + Save Contact button appears on add/edit

**3. 📜 Change Logs**

* Scrollable internal panel
* Displays:
  + DateTime
  + Changed Field
  + Old Value → New Value
  + Logged as Biller ID | Timestamp | Change
* Example:

**[2025-08-02 13:30]** Status: Active → Delisted  
**[2025-08-02 13:42]** Contact Number: 0917xxx → 0918xxx

**📥 Import Details**

**Accepted Fields in CSV/XLSX:**

| **Column** | **Requirement** |
| --- | --- |
| Company Name | Required |
| Contact Number | Required |
| Email | Optional |
| Primary Contact Person | Optional |
| Primary Contact Number | Optional |
| Address | Required |
| Type | Optional (defaults to Maintenance) |
| Status | Optional (defaults to Active) |

**Behavior:**

* Duplicates checked via: Company Name + Email
* Summary: 12 added, 2 skipped, 1 failed (bad email)
* Import modal shows guide and file basket

**📤 Export Details**

| **Attribute** | **Description** |
| --- | --- |
| Scope | All (filtered or unfiltered) |
| Format | CSV/XLSX |
| Filename Format | billers\_export\_YYYYMMDD-HHMMSS.csv |

**🧠 Backend API Endpoints**

| **Method** | **Endpoint** | **Description** |
| --- | --- | --- |
| GET | /api/billers | Fetch all billers (w/ filters) |
| POST | /api/billers | Create new biller |
| PUT | /api/billers/:id | Edit biller details |
| POST | /api/billers/import | Import billers |
| GET | /api/billers/export | Export full biller list |
| GET | /api/billers/:id/contacts | Get biller's contact persons |
| POST | /api/billers/:id/contacts | Add contact person |
| PUT | /api/billers/:id/contacts/:contactId | Edit contact person |
| GET | /api/billers/:id/logs | Get biller change logs |

**🛠 Validations & Notes**

* **Phone Number:** Must follow PH format (09xxx or +639xxx)
* **Email:** Must be valid if filled
* **No delete function:** Only Status update to Inactive/Delisted
* Only **one Primary Contact** per biller allowed
* **Contact Person Table** scrollable within modal
* Default Type: **Maintenance**
* Default Status: **Active**

**📘 Rewards Ledger — Updated Blueprint**

📄 **Page Title:** Rewards Ledger

**🎯 Purpose**

Tracks all customer reward point transactions—**earned, used, expired, or reversed**—including manual adjustments and system-generated entries from POS activity. Ensures integrity of records linked to Orders and Customers.

**🧱 Layout Overview**

**🔷 Global Summary (Top)**

* **Conversion Rate:** e.g., 1 Point = ₱1.00
* **Expiry Duration:** e.g., Points expire in 180 days
* 🔒 *Both read-only; editable only via Rewards Settings module*

**🧰 Toolbar Controls (Below Summary)**

* 🔍 **Search Box** with Dropdown Selector:  
  ▸ Fields: Order ID, Customer ID, Rewards ID, Name, Mobile
* 🎯 **Filter by Type**:  
  ▸ Options: Earned, Used, Expired, Reversed
* 🔽 **Show List Selector**: 10 / 25 / 50 per page
* 📤 **Export Button** (CSV/XLSX)
* ➕ **Add Points Button** (opens modal)
* ➖ **Deduct Points Button** (opens modal)

**📋 Rewards Ledger Table**

| **Column** | **Description** |
| --- | --- |
| **Rewards ID** | 10-digit ID, starts at 0000000001 |
| **Entry Date** | Date of ledger entry (manual/system) |
| **Order ID** | Related paid order |
| **Order Date and Time** | Timestamp when linked order was marked “PAID” |
| **Customer ID** | Format: C-YYYYMMDD-#### |
| **Name** | Full name of customer |
| **Mobile** | 11-digit Philippine number |
| **Points** | Points earned or deducted |
| **Equivalent** | Peso value based on conversion rate at the time |
| **Type** | One of: Earned, Used, Expired, Reversed |
| **Expiry Date** | Calculated based on global setting at time of entry |
| **Conversion Rate** | e.g., 1 Point = ₱1.00 |
| **Expiry Duration** | Days from entry to expiry |

✅ **Pagination** included with Prev / Next buttons

**🔄 Type Definitions**

| **Type** | **When Used** | **Source** |
| --- | --- | --- |
| **Earned** | Points earned via successful transaction or manual grant | POS or Manual |
| **Used** | Points deducted via redemption or manual deduction | POS or Manual |
| **Expired** | Points expired after validity duration | System |
| **Reversed** | Rollback of previously earned points due to order return/void | POS > Order Return Flow |

**➕ Add Points Modal**

| **Field** | **Description** |
| --- | --- |
| **Customer ID** | Required, auto-fills Name + Mobile |
| **Order ID** | Required, auto-fills Order Date and Time |
| **Points** | Required, numeric input |
| **Conversion Rate** | Auto-suggest from global setting |
| **Expiry Duration** | Auto-suggest from global setting |

🟢 System-generated on Save:

* Rewards ID
* Entry Date
* **Type:** "Earned"

🔘 **Buttons:** Save, Cancel

**➖ Deduct Points Modal**

| **Field** | **Description** |
| --- | --- |
| **Customer ID** | Required, auto-fills Name + Mobile |
| **Order ID** | Required, auto-fills Order Date and Time |
| **Conversion Rate** | Auto-filled based on selected “Earned” record |
| **Choose How to Deduct** | Dropdown: Points or Amount |
| **Value to Deduct** | Numeric input (interpreted as points or equivalent peso) |

🟢 System-generated on Save:

* Rewards ID
* Entry Date
* **Type:** "Used"

🔘 **Buttons:** Save, Cancel

**🔁 Reversed Entries (NEW Behavior)**

* Created **automatically** by system when:
  + Full or partial order return is processed
  + Order is voided after being Paid
* Pulls original “Earned” entry and creates matching “Reversed” record:
  + Points: Same as original earned amount (or proportional in partial)
  + Conversion Rate and Expiry Duration: Retained from original
  + Order ID and Customer ID are reused
  + **Note:** No manual “Reversed” entries allowed

📓 **Ledger Note** (shown on hover or as tooltip):

“Points reversed due to order return (O-YYYYMMDD-####)”

**📤 Export Behavior**

* Exports all visible records
* Includes all fields from ledger table
* Filename: rewards\_ledger\_export\_YYYYMMDD-HHMMSS.csv

**🧠 Backend API Endpoints**

| **Method** | **Endpoint** | **Description** |
| --- | --- | --- |
| GET | /api/rewards/ledger | Retrieve ledger entries (with filters) |
| POST | /api/rewards/ledger/add | Add “Earned” record |
| POST | /api/rewards/ledger/deduct | Add “Used” record |
| GET | /api/rewards/ledger/export | Export rewards ledger |
| POST | /api/rewards/ledger/reverse | System-only endpoint for order reversals |

**🛠 Validations**

* Valid order and customer must exist
* Entry date = system date (no override)
* No manual entries for **“Reversed”**
* Deducted points cannot exceed current available
* Rewards ID must be unique

**💸 Promos Module – Final Blueprint**

**📄 Page Title:**

**Promos**

**🧭 Purpose:**

Manage promotional campaigns by creating flexible promo codes. Supports discounts, freebies, Buy X Get Y logic, quantity conditions, min purchase rules, date validity, and redemption tracking.

**🧱 Layout Structure:**

| **Section** | **Height Ratio** | **Scroll** |
| --- | --- | --- |
| Promo Form Tile | 1/3 | ❌ No scroll |
| Promo Table Tile | 2/3 | ✅ Table only |
| Max Width | 1620px | Fixed container |

**🎨 Theme & UX:**

* Futuristic glassy neon-blue theme
* No vertical scroll on form
* Promo table is scrollable
* Fully responsive in 1620×1080 layout (excl. sidebar)

**📆 Promo Summary Section:**

*“As of 08/02/2025 — 4 Active Promos, 12 Used, 5 Expired”*  
Pulled from backend /api/promos/stats

**🎛️ Promo Code Generator Tile:**

**📌 Promo Types:**

| **Type** | **Description** |
| --- | --- |
| **Fixed Discount** | ₱ off |
| **Percentage Discount** | % off |
| **Free Item** | Get 1 item free |
| **Buy X Get Y – Free** | Buy A (qty) → Get B (qty) free |
| **Buy X Get Y – ₱ Off** | Buy A (qty) → Get B (qty) at ₱ discount |
| **Buy X Get Y – % Off** | Buy A (qty) → Get B (qty) at % discount |

**🧩 Rule Fields – Dynamic:**

| **Field** | **Applies To** | **Required** |
| --- | --- | --- |
| Promo Code | All types | ✅ Unique, max 15-char, alphanumeric, case-sensitive |
| Promo Type | All types | ✅ |
| Buy Item/Group | Buy X Get Y types | ✅ |
| Buy Quantity | Buy X Get Y types | ✅ |
| Get Item/Group | Buy X Get Y types | ✅ |
| Get Quantity | Buy X Get Y types | ✅ |
| ₱ Discount Amount | Buy X Get Y – ₱ Off / Fixed Discount | ✅ if applicable |
| % Discount | Buy X Get Y – % Off / % Discount | ✅ if applicable |
| Free Item Description | Free Item | Optional |
| **Minimum Purchase Amount** | All types (optional) | e.g., ₱500 |
| Start Date | All types | ✅ |
| End Date | All types | Optional |
| Max Redemptions | All types | Optional |
| Note | All types | Optional (max 100 chars) |

**✅ Validation Rules:**

* Promo Code must be unique and ≤15 chars
* Start Date must be ≤ End Date
* Required fields per Promo Type enforced
* % Discount: 1–100 only
* ₱ Discount: ≥₱1
* Conflicting active promos (overlapping same code or logic) not allowed

**📋 Promo Table Tile:**

**Controls (Top Bar):**

* 🔍 **Search**: Promo Code or Rule Summary
* 🔽 **Filter by Status**: Active, Used, Expired, Suspended
* 📤 **Export Button**
* 📥 **Import Button**
* 👁️ **Show List**: 10, 25, 50 (default 10)

**📊 Table Columns:**

| **Column** | **Description** |
| --- | --- |
| Promo Code | 15-char, unique |
| Type | Promo Type (e.g., % Off, Buy X Get Y) |
| Rule Summary | Short auto-summary (e.g., Buy 2 from Group 01 → Get 1 @ 10% Off) |
| Min Purchase | ₱ value or None |
| Start Date | Validity start |
| End Date | Optional |
| Status | Active, Used, Expired, Suspended |
| Redemptions | # of unique customers |
| Note | Optional note (up to 100 chars) |
| Action | Edit button (modal) |

**✏️ Edit Promo Modal:**

| **Editable Fields** |
| --- |
| Start Date |
| End Date |
| Max Redemptions |
| Note |
| Status (Active, Suspended, etc.) |

**Not Editable**: Promo Code, Promo Type, Rule Logic, Redemption Counter

**🔄 Redemptions Tracker:**

* Auto-incremented on successful promo use
* Logs Order ID + Customer ID
* Unique use per customer only

**📤 Export (CSV/XLSX):**

| **Details** |
| --- |
| Scope: All promo records (filtered/unfiltered) |
| Filename: promos\_export\_YYYYMMDD-HHMMSS.csv |
| Fields: All columns + full rule config |

**📥 Import (CSV/XLSX):**

| **Required Fields** | **Notes** |
| --- | --- |
| Promo Code | 15-char, case-sensitive, unique |
| Type | Must match one of listed promo types |
| Rule Summary | Short summary used for display |
| Start Date | YYYY-MM-DD |
| End Date | Optional |
| Status | One of: Active, Used, Expired, Suspended |
| Note | Optional |

🔔 **Import Summary Modal**:  
19 added, 2 skipped (invalid date), 1 duplicate

**🔄 Pagination:**

* [ Prev ] [ currentPage / totalPages ] [ Next ]
* Consistent pagination style

**🧠 Backend API (brain/):**

| **Method** | **Endpoint** | **Description** |
| --- | --- | --- |
| GET | /api/promos | Fetch with filters |
| POST | /api/promos | Add new promo rule |
| PUT | /api/promos/:code | Update editable fields |
| GET | /api/promos/export | Export to CSV/XLSX |
| POST | /api/promos/import | Import promo data |
| POST | /api/promos/redeem | Record usage |
| GET | /api/promos/stats | Summary counts |

**🔗 Integrations:**

| **Module** | **Purpose** |
| --- | --- |
| Orders (POS) | Validates promo code & applies logic |
| Customers | Tracks usage per customer |
| Rewards (future) | Promo-driven points or bonuses |

**🎫 Vouchers Module – Final Blueprint**

**📄 Page Title:**

**Vouchers**

**🧭 Purpose:**

Manage refill vouchers issued in single or booklet formats. Vouchers include value (refill count), validity dates, and are tracked by status: Circulation, Used, Expired.

**🧱 Layout Structure:**

| **Section** | **Height Ratio** | **Scroll** |
| --- | --- | --- |
| Voucher Enrolment | 1/3 | ❌ No scroll |
| Voucher Table | 2/3 | ✅ Table only |
| Max Width | 1620px | Fixed width container |

**🎨 Theme & Behavior:**

* Neon-blue glassmorphic theme
* Fixed 1620×1080 layout (excluding sidebar)
* Scrollable table; non-scrollable form
* Controls/pagination style follows Rewards module

**📆 Voucher Summary Section:**

*“As of 08/02/2025 — 212 In Circulation, 89 Used, 37 Expired”*

Pulled from GET /api/vouchers/stats and auto-updates on data changes.

**🔷 Voucher Enrolment Tile:**

**📘 Voucher Type Dropdown:**

| **Type** | **Description** |
| --- | --- |
| **Single** | Enroll a single voucher using a 10-digit ID |
| **Multi** | Enroll a batch using a range of 10-digit IDs |

**🧾 Common Form Fields:**

| **Field** | **Type** | **Required** | **Description** |
| --- | --- | --- | --- |
| Voucher ID | Single | ✅ | 10-digit, zero-padded |
| Start ID | Multi | ✅ | Start of range (10-digit) |
| End ID | Multi | ✅ | End of range (≥ Start ID) |
| Refill Equivalent | Both | ✅ | Refill count (positive integer) |
| Start Date | Both | ✅ | Date when voucher becomes valid |
| End Date | Both | Optional | Expiry date (must be ≥ Start Date) |
| Save Button | — | — | Triggers validation and submission |

**⚠️ Validations:**

* Voucher IDs must be exactly **10 digits** and **unique**
* Multi-range: Start ID ≤ End ID
* Refill must be a **positive number**
* Valid date formatting required
* For imports: skips duplicate IDs

**📊 Voucher Table Tile:**

**🔧 Controls (Top of Table):**

* 🔍 Search by **Voucher ID**
* 🔽 **Filter by Status**: Circulation, Used, Expired
* 📤 **Export Button** (CSV/XLSX)
* 📥 **Import Button** (CSV/XLSX)
* 📄 **Show List** dropdown: 10, 25, 50 (default 10)

**📋 Table Columns:**

| **Column** | **Description** |
| --- | --- |
| Voucher ID | 10-digit, zero-padded, unique |
| Date Added | Auto-filled on enrollment |
| Refill | # of refills this voucher provides |
| Start Date | When voucher becomes usable |
| End Date | Optional expiration date |
| Status | Circulation / Used / Expired |
| Action | Edit button to open editable modal |

**✏️ Edit Voucher Modal:**

| **Field** | **Editable** |
| --- | --- |
| Refill Count | ✅ |
| Start Date | ✅ |
| End Date | ✅ |
| Status | ✅ |
| Voucher ID | ❌ |
| Date Added | ❌ |

**📥 Import Details:**

**📦 Required Fields:**

* Voucher ID (10-digit, zero-padded, unique)
* Refill
* Start Date
* Status (Circulation, Used, Expired)

**⚙️ Import Behavior:**

* Auto-fill: **Date Added = today**, **End Date = null** if missing
* Skips any duplicate Voucher ID
* Result modal: *“20 added, 3 skipped (duplicates)”*

**📤 Export Details:**

| **Export Scope** | **Format** | **Filename Format** |
| --- | --- | --- |
| Full voucher list | CSV & XLSX | vouchers\_export\_YYYYMMDD-HHMMSS.csv |

**🔄 Pagination:**

Consistent pagination:

[ Prev ] [ currentPage / totalPages ] [ Next ]

**🧠 Backend API Endpoints:**

| **Method** | **Endpoint** | **Purpose** |
| --- | --- | --- |
| GET | /api/vouchers | Fetch voucher list (with filters) |
| POST | /api/vouchers | Add new voucher(s) |
| PUT | /api/vouchers/:id | Edit refill, status, start/end dates |
| GET | /api/vouchers/export | Export vouchers to CSV/XLSX |
| POST | /api/vouchers/import | Bulk voucher import |

**🔗 Future Integrations:**

| **Module** | **Description** |
| --- | --- |
| POS/Orders | Voucher redemption validation at checkout |
| Customers | Optional link for customer-based tracking |

**💰 Module: Expenses**

📄 **Page Title:** Expenses

**🎯 Purpose**

This module tracks all business expenditures based on finalized invoices from:

* 🧾 **Supplies**
* 🧹 **Maintenance**
* 🏢 **Billing**

It mirrors the **Orders module**, showing financial outflows (vs. revenue). All entries here are system-generated — **manual expense entry is not allowed**.

**🧩 Data Sources**

Expenses are inserted automatically **only when an invoice is finalized and saved** from:

* Invoicing > Supplies
* Invoicing > Maintenance
* Invoicing > Billing

**🖥️ Layout Overview**

* Full-page responsive layout
* Table view with filters, export, print
* View Invoice modal (A5 format, read-only)

**🔍 Top Controls**

| **Control** | **Function** |
| --- | --- |
| 🔎 Search Bar | Search by: Invoice ID, Notes, Biller/Supplier Name |
| 🗂️ Source Filter | Dropdown: Supplies, Maintenance, Billing |
| 📅 Date Filter | Date Range Picker (based on Invoice Date) |
| 🏷️ Entity Filter | Supplier or Biller (auto-fetched) |
| 📤 Export Button | Export filtered/full list to CSV/XLSX |
| 🖨️ Print Summary | Generates printable summary (A5 or A4) |

**📊 Main Table Columns**

| **Column** | **Description** |
| --- | --- |
| Invoice ID | Format: I-YYYYMMDD-#### (auto-generated per invoice module) |
| Date | Date of invoice |
| Type | One of: Supplies, Maintenance, Billing |
| Notes / Remarks | Summary or notes from invoice |
| Amount | ₱ Total expense value (sum of line items) |
| Payment Method | Cash / GCash / Bank Transfer |
| Reference # | Shown if GCash or Bank Transfer |
| Bank Name | Required if Bank Transfer |
| OR Number | Optional BIR receipt number |
| Linked Entity | Supplier or Biller name (based on invoice type) |
| Action | 🔍 View Invoice (modal preview) |

**🖨️ View Invoice Modal**

* Format: **A5 (Half A4 Portrait)**
* Header with Business Info (from Settings)
* Invoice Details (ID, Date, Linked Entity)
* **Line Items** from expense\_line\_items table
* Total Amount, Payment Info
* Optional Remarks / Signature
* Read-only modal (no edits here)
* Exportable as PDF or printed directly

**📥 Line Items Table (Expanded Audit Support)**

Stored in separate table: expense\_line\_items

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id | INTEGER | Auto-increment primary key |
| invoice\_id | TEXT | Foreign key to expenses.invoice\_id |
| particular | TEXT | Item/service name |
| amount | REAL | ₱ amount per line |
| status | ENUM | Confirmed / Pending / Reversed (optional) |
| created\_at | DATETIME | Timestamp for entry creation |

✅ This table is auto-generated per invoice save.  
✅ **Updates** (via Update Invoice) will **overwrite** and sync all line items.

**📤 Export Function**

| **Feature** | **Details** |
| --- | --- |
| Format | CSV and XLSX |
| Filename | expenses\_export\_YYYYMMDD-HHMMSS.csv |
| Export Scope | Current filtered or full table |
| Includes | All main columns + total + payment info + linked entity |

**🖨️ Print Summary**

* A5 or A4 layout
* Header: Logo, Business Info, Report Title
* Table: Invoice ID, Date, Type, Amount, Supplier/Biller
* Subtotal of visible entries
* Optional footer for signature/approval

**🔒 Editing Rules**

* **Expenses module is read-only**
* Only the originating invoice module (Supplies, Maintenance, Billing) can update:
  + Invoice totals
  + Line items
  + Payment method
  + Reference # or OR#

**🔗 Module Integrations**

| **Module** | **Integration Purpose** |
| --- | --- |
| Invoicing – Supplies | Triggers expense + line items on save |
| Invoicing – Maintenance | Same as above |
| Invoicing – Billing | Same as above |
| Inventory | Used for Supplies (source of line items) |
| Suppliers / Billers | Populates Linked Entity dropdowns |
| Banks | Validates bank transfer info |
| Settings | Provides business info for invoice rendering |

**📡 Backend Routes**

| **Method** | **Endpoint** | **Description** |
| --- | --- | --- |
| GET | /api/expenses | Fetch filtered expense list |
| GET | /api/expenses/:invoice\_id/lines | Fetch all line items of a specific invoice |
| POST | /api/expenses/create\_from\_invoice | Internal use: Add new expense record |
| PUT | /api/expenses/update\_from\_invoice | Internal use: Update expense and line items |
| GET | /api/expenses/export | Export filtered/full expense list |

**🔁 Business Logic Enforcement**

| **Scenario** | **Result** |
| --- | --- |
| Save new invoice | Creates new expense record + line items |
| Update existing invoice | Updates existing expense + overwrites line items |
| View invoice in Expenses | Always reflects current saved data |
| Supplies edits price/qty | Changes reflected here after re-save |

**✅ Sample Use Cases**

| **Use Case** | **Outcome** |
| --- | --- |
| Supplier invoice saved | Expense logged + line items created |
| Edit Maintenance invoice (adjust price) | Expense total updated |
| View Billing expense for July | Accurate read-only breakdown |
| Export monthly expenses | Filters + CSV/XLSX download |
| Print summary for audit | A4 report layout with totals |

**📦 Orders Module – Finalized Developer Blueprint (v4.0)**

**📄 Page Title: Orders**

**🎯 Purpose**

Manages all sales transactions made via POS – New Order. Includes filters, updates, receipt viewing, return handling, and proper inventory + rewards synchronization.

**🧱 Layout Overview**

| **Section** | **Behavior** |
| --- | --- |
| 🔝 Top Panel | Fixed, no scroll |
| 📋 Orders Table | Scrollable |
| 🧾 Receipt Preview | Slide-up modal (80mm format) |
| 🔄 Return Modal | Slide-up with line-item return handling |
| 💻 Max Width | 1620px (excluding 300px sidebar) |
| 📏 Screen Height | Optimized for 1920×1080 |

**🔍 Top Panel Filter Controls**

* **Search Bar** – search by: Order #, Customer ID, Name, Mobile
* **Status Filter** – Hold, Paid, Returned, Partially Returned, Void
* **Date Picker** – Order Date From - To
* **Payment Filter** – Cash, GCash, Bank Transfer
* **Show List** – 10 / 25 / 50 per page (default: 10)
* **Export Button** – download visible/full table as .csv / .xlsx

**📋 Orders Table Columns**

| **Column** | **Description** |
| --- | --- |
| Order # | Format: O-YYYYMMDD-#### |
| Order Date | Full timestamp of order placement |
| Customer | Name (Customer ID) or “Walk-In” |
| Total Amount | Final total (includes VAT, shipping, discounts) |
| Payment Method | Cash, GCash, Bank Transfer, Split Payment |
| Reference No. | Ref number (for GCash/Bank) |
| Bank Name | Shown only for Bank Transfer |
| Status | Hold, Paid, Returned, Partially Returned, Void |
| OR # | Optional, shown only if entered |
| Action | 👁 View, ✏️ Edit, 🔄 Process Return (if Paid) |

**🧾 Receipt View Modal**

* Simulated **80mm thermal receipt** (same as POS)
* Shows:
  + Order Date & Time
  + Status Stamp:
    - ✅ VOIDED – grey
    - 🔴 RETURNED – red
    - 🟠 PARTIALLY RETURNED – orange
  + Customer Info (if present)
  + Payment Summary + OR#
  + Items Breakdown + VAT + Discounts
* **Reprint Button** – export / print view as PDF

**✏️ Edit Order Modal**

Fields shown:

* **Status Dropdown** → Options:
  + Hold, Paid, Void (see validations)
* **OR Number** → Optional field (max 20 chars)
* **Payment Method** → Select new method if changing
* **Reference Number** and **Bank Name** (shown conditionally)
* **Void Reason / Cancel Note** → Required if changing to Void

**❗ Validations**

| **Action** | **Validation** |
| --- | --- |
| Change to Paid | Requires payment info (method, ref# if needed) |
| Change to Void | Allowed **only if current status is Hold** |
| Returned/Partially Returned | Can **only be set via Process Return modal** |

**🔄 Process Return Modal**

**Available only for orders with Paid status.**

**Line Item Table:**

| **Column** | **Description** |
| --- | --- |
| # | Row number |
| Item Name | Full name from inventory |
| Qty Ordered | From original order |
| Qty to Return | Input field (1 to Qty Ordered) |
| Unit Price | From original order |
| Subtotal | Auto-calculated |
| Return Reason | Dropdown |
| Status | Returned, Partially Returned (auto-evaluated) |

**Return Reasons (Dropdown Options):**

* Contaminated Water
* Water Quality Issue
* Wrong Item Delivered
* Customer Changed Mind
* Packaging Issue
* Others (custom text field)

**Other Details:**

* **Auto-calculates refund value**
* **Inventory auto-restocks** returned items
* **Rewards points reversed proportionally**
* **Order status updated** (Returned or Partially Returned)
* **Generates Return ID** (10-digit, zero-padded)
* **Submit button** → Save Return
* **Cancel button** → Close modal

**📄 Return Receipt (Sales Return Note)**

* **Generated per Return ID**
* Format: 80mm (POS) or A5 (standard)
* Includes:
  + Return ID
  + Linked Order #
  + Customer Info
  + Date & Staff
  + Items Returned, Qty, Reason
  + Total Refund
  + Reversed Points (if any)

**📘 Rewards Ledger Integration**

| **Action** | **Ledger Entry** |
| --- | --- |
| Paid Order | Earned points |
| Returned Items | Reversed points (partial or full) |
| Voided Order | ❌ No ledger entry (no rewards earned) |

**📦 Inventory Integration**

| **Status** | **Inventory Behavior** |
| --- | --- |
| Paid | Deduct item stock |
| Returned | Restock returned items only |
| Partially Returned | Partial restock |
| Void | No stock adjustment |

**📤 Export Behavior**

* Includes all table columns
* Extra field: **Item Summary** (e.g., 2x Siomai, 1x Water)
* File Format: .csv, .xlsx
* Filename pattern: orders\_export\_YYYYMMDD-HHMMSS.csv

**🧠 Backend Tables**

**orders**

* order\_id (PK)
* date, customer\_id, total, status, payment\_method, ref\_no, or\_no, etc.

**order\_items**

* order\_id, item\_id, qty, price, subtotal

**order\_returns**

* return\_id (10-digit PK)
* order\_id, customer\_id, staff\_name, total\_refund, date, notes

**order\_return\_items**

* return\_id, item\_id, qty\_returned, unit\_price, reason

**🔗 Integrations Map**

| **Module** | **Integration** |
| --- | --- |
| POS – New Order | Origin of all orders |
| Customers | Links customer data and ID |
| Rewards Ledger | Earn and reverse points |
| Inventory | Adjust stock levels |
| Reports / Exports | Used in daily and monthly summaries |

**✅ Sample Scenarios to Test**

| **Scenario** | **Expected Behavior** |
| --- | --- |
| Change Hold to Void | ✅ Allowed |
| Change Paid to Void | ❌ Not allowed |
| Return 2 of 5 items | Partially Returned status; restock 2 |
| Return all items | Returned status; full restock |
| View receipt | Proper status stamp (VOIDED, etc.) |
| Rewards reversed | Automatically shown in Rewards Ledger |
| Export report | Accurate with item summary & status |

**🛒 POS – New Order Module**

**Page Title:** Point of Sale – New Order  
**Screen Target:** 1920×1080 (1620px usable width, excluding sidebar)  
**UI Style:** Glassmorphic, neon-blue, touchscreen-friendly, fixed layout

**🧭 Purpose**

Facilitates creation of sales orders with fast item selection, integrated payment, promo/voucher/reward support, and live receipt preview. Designed for keyboard-less touchscreen use in offline environments.

**🧱 Layout Structure**

| **Section** | **Description** |
| --- | --- |
| Left Panel (300px) | Customer Info + Item Groups (vertical scrollable list) |
| Center Panel | Items Grid filtered by group |
| Right Panel | Cart + Totals + Payment + Live Receipt Preview |

**📂 Left Panel – Item Group & Customer Info**

* **Top: Customer Info Tile**
  + Displays: Name, Mobile, Customer ID, Points Balance (if any)
  + “🔍 Find” button to search/select existing customer
  + “➕ Add Walk-In” allows quick add (Name + Mobile only, address optional)
* **Item Groups List**
  + Scrollable, each group a clickable tile
  + First group = “All Items” (default)

**🧱 Center Panel – Items Grid**

* Grid layout: Responsive (4–5 columns)
* Each tile shows:
  + 📦 Item Name
  + 🔢 SKU (Item ID)
  + 💰 Unit Price
* Tap = Add item to cart (or increment quantity)

**🧺 Right Panel – Cart + Totals + Actions + Receipt**

**🧾 Cart Section**

* Shows up to 8 items (scrolls beyond that)
* Fields per item:
  + Name, Qty, Price, Subtotal
  + ⛔ Remove icon
  + ✏️ Clickable amount field → Opens **Manual Discount Modal** (see below)

**📉 Manual Discount Modal (Per Item)**

* Trigger: Click per-item subtotal
* Fields:
  1. Type: Amount or % (dropdown)
  2. Value (numeric input)
  3. Reason (100 char max)
* Buttons: Apply / Cancel

**💰 Totals Section**

* **Subtotal**
* **Discount (Cart-Level)**
  + “Add Discount” button → opens same modal format as above
* **Shipping Fee**
  + “Add Shipping” → Modal with:
    - Label (default: “Free Shipping”)
    - Amount
* **VAT Toggle** (default ON)
  + VAT Rate is pulled from **Settings Module**
* **TOTAL** – Final amount due

**🧾 Receipt Live Preview**

* Simulated 80mm thermal format
* Always visible on lower-right (scaled preview)
* Auto-updates in real-time as cart changes

**💳 Payment Section**

* **Payment Method Dropdown**
  + 💵 Cash
  + 📱 GCash → Requires Reference No.
  + 🏦 Bank Transfer → Requires Bank Name + Reference No.
  + 🔀 **Split Payment** → Triggers Modal

**🔀 Split Payment Modal**

* Checkbox toggles for: Cash, GCash, Bank Transfer, Points
* Each section includes:
  + Input field for amount
  + GCash/Bank fields: Reference No., Bank Name
* Summary: Displays total paid and allocation
* Buttons: Apply / Cancel

**🎫 Promo / Voucher / Rewards**

* **Promo Code Field**
  + Label: PROMO: [Input] → [Apply Promo]
  + Modal confirmation before applying
* **Voucher Code Field**
  + Label: VOUCHER: [Input] → [Apply Voucher]
* **Rewards Points**
  + Display active points → [Apply Points]
  + Modal confirms how much will be used
* **Rules**:
  + Promo + Voucher + Rewards **can be used together**
  + Manual Discount + Promo **not allowed by default**
    - Override allowed with warning + required justification

**🧾 OR# Field**

* Always visible
* Required only if total > ₱20
* Max 20 characters, alphanumeric

**✅ Action Buttons**

* 💾 **Confirm Order** → Triggers validations, deducts stock, finalizes sale
* ⏸️ **Hold Order** → Saves all data (editable only in Update Order module)
* 🧾 **Print Receipt Preview** → Opens thermal receipt in modal

**🧠 Backend Considerations**

| **Endpoint** | **Purpose** |
| --- | --- |
| POST /api/orders | Create order |
| GET /api/inventory/:id | Fetch item data |
| POST /api/orders/receipt | Return receipt layout |
| POST /api/orders/deduct | Deduct inventory |
| POST /api/or/check | Check OR# duplication |
| POST /api/orders/split | Record split payment structure |

**✅ Validations**

| **Field** | **Validation** |
| --- | --- |
| Items | At least 1 item required |
| Payment Method | Required |
| GCash / Bank | Reference # required |
| Split Payment | Total must match bill |
| Promo + Manual Discount | Allowed only with override |
| OR# | Required if total > ₱20 |

**💡 Additional Notes**

* ✅ Cart edits via modals (per item and cart-wide)
* ✅ VAT Rate is from Settings (editable there only)
* ✅ Orders saved as “Hold” are editable only via **POS – Update Order**
* ✅ Returned or Refunded Orders (if tagged later) will show an ⚠️ tag
* ✅ Rewards usage logged to Rewards Ledger with type: “Used”

**♻️ Return Orders Module – Developer Blueprint**

**📄 Page Title: Return Orders**

**🎯 Purpose**

This module tracks all orders that have been fully or partially returned. It summarizes return metadata per order and provides a filtered, exportable list of processed returns for inventory, customer service, and audit reference.

**🧱 Layout Overview**

| **Section** | **Behavior** |
| --- | --- |
| Top Filter Bar | Fixed |
| Returns Table | Scrollable |
| Screen Width | Max 1620px (excluding sidebar) |
| Theme | Glassmorphic neon-blue |
| Module Position | Between **Orders** and **Expenses** in sidebar |

**🔍 Top Filter Bar**

* **Search Box** – Search by:
  + Return ID
  + Order ID
  + Customer ID
  + Name
* **Date Range Filter** – Return Date (From – To)
* **Return Type Filter** – Returned / Partially Returned
* **Show List Dropdown** – 10 / 25 / 50 entries (default 10)
* **Export Button** – CSV / XLSX full export

**📋 Return Orders Table**

| **Column** | **Description** |
| --- | --- |
| Return ID | 10-digit unique identifier (zero-padded) |
| Order ID | Linkable to original order (if feasible) |
| Type | Returned or Partially Returned |
| Amount | Total ₱ amount of refunded items (Subtotal + VAT) |
| Reason | If single-item return: shows reason. If multiple: shows “Multiple Reasons” |
| Return Date | Date of return (MM/DD/YYYY) |
| Processed By | Staff name or system user who logged the return |
| Action | 🔍 View Return Details (opens return receipt) |

**🧾 Return Receipt View Modal**

* Modal opens from the Action column.
* Shows formatted Return Note:
  + Header: Return ID, Order ID, Customer Info
  + Itemized list:
    - Item Name
    - Qty Returned
    - Unit Price
    - Reason
  + Total Refund Amount
  + Original Conversion Rate (if points were reversed)
  + Timestamp + Processed By
  + Printable layout: 80mm or A5 (uses receipt/invoice template engine)
  + “🖨 Print Return Note” button

**📤 Export Function**

* 📁 **Filename**: return\_orders\_export\_YYYYMMDD-HHMMSS.csv
* Export Scope: Filtered or full list
* Included Fields:
  + Return ID
  + Order ID
  + Type
  + Total Refund
  + Return Date
  + Processed By
  + Reason Summary

**🧠 Database Table: order\_returns**

| **Field** | **Type** | **Notes** |
| --- | --- | --- |
| return\_id | TEXT (PK) | 10-digit, zero-padded |
| order\_id | TEXT | Linked to orders.order\_id |
| type | TEXT | Returned / Partially Returned |
| total\_amount | REAL | Sum of returned item values |
| return\_date | DATE | Date of processing |
| reason\_summary | TEXT | “Multiple Reasons” or single reason |
| processed\_by | TEXT | Staff/cashier name |
| created\_at | DATETIME | Timestamp |

**🔗 Module Integrations**

| **Module** | **Integration Purpose** |
| --- | --- |
| Orders | Status updates + Return initiation |
| Inventory | Item quantities restocked |
| Rewards Ledger | Points reversed or partially deducted |
| Customers | Reflected in View Profile (under Sales Summary) |

**✅ Feature Summary**

| **Feature** | **Status** |
| --- | --- |
| Only shows Returned or Partially Returned orders | ✅ |
| Tracks processed staff | ✅ |
| Exportable table view | ✅ |
| Return Note generation (PDF/printable) | ✅ |
| Filters: Type, Date, Order ID | ✅ |
| Summary by Return ID | ✅ |
| View Return Details modal | ✅ |

**🔄 Return Items Module – Developer Blueprint**

**📄 Page Title: Return Items**

**🎯 Purpose**

The Return Items module shows a detailed breakdown of all returned products across all orders. It is used for inventory reconciliation, reporting, customer service, and reward point management.

This view is **item-centric**, unlike Return Orders, which is order-level.

**🧱 Layout Overview**

| **Section** | **Behavior** |
| --- | --- |
| Top Filter Bar | Fixed |
| Items Table | Scrollable |
| Screen Width | Max 1620px (excluding sidebar) |
| Table Height | Full remaining screen height |
| Style | Glassmorphic neon-blue UI |

**🔍 Top Filter Controls**

* **Search Box** – Filter by:
  + Item Name
  + Order ID
  + Return ID
* **Dropdown Filters**:
  + **Return Type**: Full / Partial
  + **Return Reason**: All reasons (Contaminated Water, Packaging Issue, etc.)
* **Date Range Picker** – Return Date From - To
* **Show List Dropdown** – 10 / 25 / 50 (default: 10)
* **📤 Export Button** – CSV/XLSX of visible data

**📋 Return Items Table Columns**

| **Column** | **Description** |
| --- | --- |
| Return ID | Unique 10-digit zero-padded ID (clickable to open Return Receipt modal) |
| Order ID | Original Order number |
| Item Name | Returned product name |
| Qty Returned | Quantity returned for this item |
| Reason | Reason selected during return (e.g., Contaminated Water) |
| Return Type | Full or Partial (based on order-level context) |
| Item Amount | ₱ Total = Qty × (Selling Price + VAT) |
| Return Date | When the return was logged (MM/DD/YYYY) |
| Processed By | Name of staff/admin who processed the return |

**📄 Notes on Logic & Calculations**

* **Item Amount** = Qty × (Item Unit Price + VAT at time of order)
* **Return Type** is derived from Return Order’s overall type
* If multiple returns happen for same item across orders, each line shown separately

**📤 Export Behavior**

* 📁 **Filename**: return\_items\_export\_YYYYMMDD-HHMMSS.csv
* Included fields:
  + Return ID
  + Order ID
  + Item Name
  + Qty Returned
  + Reason
  + Return Type
  + Item Amount
  + Return Date
  + Processed By
* Scope: Current filtered results

**🧠 Database: order\_return\_items**

| **Field** | **Type** | **Notes** |
| --- | --- | --- |
| return\_id | TEXT | Linked to order\_returns.return\_id |
| order\_id | TEXT | Linked to orders.order\_id |
| item\_id | TEXT | From inventory |
| item\_name | TEXT | Stored to preserve history |
| qty\_returned | INTEGER | Number of units returned |
| reason | TEXT | Dropdown value or free text |
| return\_type | TEXT | Full / Partial |
| unit\_price | REAL | With VAT included |
| return\_date | DATE | When the return was processed |
| processed\_by | TEXT | Staff/cashier name |

**🔗 Integrations**

| **Module** | **Use Case** |
| --- | --- |
| Inventory | Qty is restocked based on returned item |
| Orders | Pulls item context & customer link |
| Rewards Ledger | Used to calculate point reversal (partial/full) |
| Return Orders | Line items linked back to parent Return ID |

**✅ Feature Summary**

| **Feature** | **Status** |
| --- | --- |
| Fully itemized return list | ✅ |
| Search + Filter by reason, item, date | ✅ |
| Return Amount auto-calculated | ✅ |
| Processed By tracking | ✅ |
| Return Type (Full/Partial) shown | ✅ |
| Export to CSV/XLSX | ✅ |
| Receipt modal link via Return ID | ✅ |

**🛠️ POS – Update Order**

📄 **Page Title:** Update Order  
📅 **Purpose:**  
This module allows **authorized users** to resume, modify, and finalize previously saved Hold orders. It supports:

* Updating cart items and quantities
* Changing or adding customer information
* Modifying or completing payments (including split payments)
* Editing VAT, discounts, OR#, and promo/voucher/reward applications
* Generating thermal-format receipt previews

**🧱 Layout Overview**

| **Section** | **Width** | **Scroll** |
| --- | --- | --- |
| 🔍 Left Panel – Order Lookup + Summary | 400px | ✅ Scrollable |
| 🛒 Right Panel – Cart + Payment + Preview | 1220px | ✅ Scroll (cart only >8 items) |

Screen size: **1920×1080 fixed layout**  
Theme: **Glassmorphic neon-blue**

**🔍 Left Panel: Order Search + Metadata**

**🔎 Order Search Controls**

* **Search bar:**
  + Search by: Order ID (O-YYYYMMDD-####), Customer Name, Customer ID
* **Filters:**
  + Date Range (From – To)
  + Sort: Newest ↔ Oldest
* **List of Hold Orders:**
  + Click to load into right panel

**📋 Display Order Info**

* **Order ID**
* **Date Created**
* **Cashier/User (auto-filled)**
* **Order Status:** Hold only
* **Customer Info (if exists):**
  + ID, Name, Mobile, Address
  + If missing: Add via Quick Add modal
    - Fields: First Name, Mobile (Address optional), auto-generate Customer ID
    - Saves to main Customers DB with Source: Walk-In

**🛒 Right Panel: Edit + Finalize Order**

**🧾 Editable Cart List**

* Scrollable after 8 items
* Columns:
  + Item Name
  + SKU
  + Qty Control (+/- or numeric input)
  + Unit Price (clickable for override)
  + Line Subtotal
  + ❌ Remove Button

**✏️ Manual Per-Item Discount**

* Click Unit Price → opens modal:
  + **Discount Type:** % or ₱
  + **Value Input**
  + **Reason Field** (max 100 chars)
  + [Apply] or [Cancel]

**➕ Add Items Panel**

* Touch-friendly grid of items (same layout as POS – New Order)
* Filter by Item Group
* Adding updates quantities in current cart

**💳 Payment Section**

**Fields**

* **Subtotal (auto)**
* **Manual Cart-Level Discount:**
  + Opens modal → % or ₱ + Reason
* **Add Shipping:**
  + Opens modal: Name (default: Free Shipping), Amount
* **VAT Toggle + Rate (auto from Settings)**
* **Promo / Voucher / Rewards Integration:**
  + **Promo Code:**
    - Input + Add Promo Button → confirmation modal
  + **Voucher Code:**
    - Input + Add Voucher Button → confirmation modal
  + **Apply Rewards Points:**
    - If customer selected
    - Shows available balance → apply as amount
    - Confirmation modal → logs "Used" in Rewards Ledger

**💸 Payment Options**

* **Payment Method:** Dropdown
  + Cash, GCash, Bank Transfer, **Split Pay**
* **Split Payment Modal:**
  + Checkbox list of payment types
  + Fields for amount per method
  + Total auto-validated vs grand total
  + Saves as "Split Payment: Cash ₱XX + GCash ₱YY..."
* **GCash/Bank Ref # + Bank Name shown conditionally**

**🧾 OR# Input**

* Optional always; **required if total > ₱20**
* Format: alphanumeric (max 20 chars)
* Always visible (highlighted if required)

**🖨️ Receipt Preview**

* Click [🧾 Receipt Preview] → modal showing full 80mm layout:
  + Store Info
  + Items, Discounts, VAT
  + Customer Info
  + Payment breakdown
  + OR#, Status, Cashier
  + Optional Footer

**✅ Action Buttons**

| **Button** | **Function** |
| --- | --- |
| ❌ Cancel Order | Requires confirmation → returns stock to inventory |
| 💾 Save Updates | Applies changes, keeps status = Hold |
| 📤 Finalize & Print | Marks status = Paid → deducts stock, rewards entry, opens receipt preview |

**🔄 Module Business Logic**

| **Event** | **Action** |
| --- | --- |
| Cancel Order | Returns all items to Inventory. Updates status to "Void". |
| Finalize Order | Deducts stock, logs Rewards (if any), applies payment |
| Apply Rewards | Deducts points from ledger, logs "Used" entry |
| Partial Discounts | Logged per item or cart, with justification |
| Apply Voucher/Promo | Validated against Rewards/Promos DB; status updated if used |
| Split Payment | Payment methods + amounts recorded in summary field |
| Reprint Receipt | Triggerable after finalize or from Orders module |

**🔒 Role & Access Logic**

* Only users with Cashier or Manager role can:
  + Load Hold orders
  + Modify and finalize orders
  + Cancel orders

**🧠 Backend Routes**

| **Method** | **Endpoint** | **Purpose** |
| --- | --- | --- |
| GET | /api/orders/hold | Fetch list of Hold orders |
| GET | /api/orders/:id | Get details of selected order |
| PUT | /api/orders/:id/update | Save changes (status remains Hold) |
| POST | /api/orders/:id/finalize | Convert to Paid; apply logic |
| DELETE | /api/orders/:id | Cancel and void the Hold order |
| POST | /api/rewards/use | Log rewards used |

**📘 Additional Database Table Impact**

* **orders** → status update, OR#, payment details, etc.
* **order\_items** → updated quantities, discounts
* **order\_returns** → not applicable here
* **rewards\_ledger** → logs "Used" if rewards applied
* **inventory** → restock (cancel) or deduct (finalize)

**✅ Final Features Checklist**

| **Feature** | **Status** |
| --- | --- |
| Load & edit Hold orders | ✅ |
| Touchscreen-friendly UI | ✅ |
| Add/remove/edit items | ✅ |
| Promo, Voucher, Rewards support | ✅ |
| Manual per-item and cart discounts | ✅ |
| GCash/Bank/Ref# validation | ✅ |
| Split Payment modal | ✅ |
| Receipt preview w/ 80mm layout | ✅ |
| OR# input + validation logic | ✅ |
| Customer info tile + quick add | ✅ |
| Role-based restrictions | ✅ |

**📅 Calendar Module – Developer Blueprint (v1.1)**

**📄 Page Title:**

**Maintenance & Holiday Calendar**

**🎯 Purpose:**

To provide a centralized, interactive calendar interface for tracking system-wide or personal reminders, alarms, and maintenance activities, as well as syncing official Philippine holidays.

**🧱 Layout & Structure**

| **Section** | **Behavior** |
| --- | --- |
| Full Calendar View | Monthly default, with Week/Day toggle |
| Top Toolbar | Add, Sync, Filter, View Mode |
| Scroll Behavior | Scrollable by month |
| Theme | Neon-blue glassmorphic style |
| Resolution | Fixed layout (1620px wide inside 1920×1080 screen) |

**🔝 Top Toolbar Controls**

* 🔘 **Add Alarm / Activity** – Opens event creation modal
* 🔘 **Sync Holidays** – Manually triggers online sync
* 🔘 **View Toggle** – Month / Week / Day
* 🔘 **Filter Checkboxes**:
  + ✅ Show Alarms
  + ✅ Show Activities
  + ✅ Show Holidays
* 🔍 **Search Bar** – Filter by keyword in title or description
* 🔽 **Type Filter** – Dropdown: All, Alarm, Activity, Holiday

**📆 Calendar Event Types & Visuals**

| **Type** | **Color** | **Icon** | **Editable** |
| --- | --- | --- | --- |
| Alarm | 🔴 Red | 🔔 | ✅ Yes |
| Activity | 🔵 Blue | 🛠 | ✅ Yes |
| Holiday | 🟢 Green | 🎉 | ❌ No |
| Recurring | Any | 🔁 icon | ✅ Per instance |

**➕ Add/Edit Alarm or Activity Modal**

| **Field** | **Type** | **Notes** |
| --- | --- | --- |
| Title | Text (required) | Activity or reminder label |
| Description | TextArea (optional) | Supports multiline |
| Date & Time | Picker (required) | Single or recurring entry |
| Type | Dropdown | Options: Alarm / Activity |
| Recurrence | Dropdown | None / Daily / Weekly / Monthly |
| Notify | Checkbox | In-app reminder shown upon app open (if enabled in Settings) |
| Link to Invoice | Optional Text Field | For future integration with Maintenance module |
| Save / Cancel | Buttons | Confirm or discard input |

* If recurring, system prompts:
  + "Edit this only" or "Edit full series"
  + Drag/drop updates only allowed per-instance unless batch-edited

**🗓️ Event Behavior**

* Clicking an event opens quick view modal with details
* Options: Edit / Delete (unless Holiday)
* Drag to reschedule individual occurrences
* Recurring indicators (🔁) shown on calendar tiles

**🎉 Holiday Sync Logic**

* Triggered by "Sync Holidays" button
* Connects to a Philippine holidays API (e.g., calendarific.com)
* Saves events in holiday\_events table
* Holidays are:
  + Shown in green with 🎉 icon
  + Not editable or deletable
  + Synced on-demand only (no background refresh)

**🔔 Notifications**

* In-app only (no email/SMS in offline version)
* Options:
  + Soft Notification (badge/reminder on app open)
  + Hard Notification (modal pop-up upon load)
* Configurable in Settings (Notifications toggle)

**🛢 Storage Structure (SQLite)**

**Table: calendar\_events**

| **Field** | **Type** | **Notes** |
| --- | --- | --- |
| id | INTEGER | Primary Key |
| title | TEXT | Required |
| description | TEXT | Optional |
| datetime | TEXT | ISO format |
| recurrence | TEXT | None / Daily / Weekly / Monthly |
| type | TEXT | Alarm / Activity |
| notify | BOOLEAN | True/False |
| linked\_invoice | TEXT | Optional |
| user\_scope | TEXT | Optional (for future user-specific filtering) |

**Table: holiday\_events**

| **Field** | **Type** | **Notes** |
| --- | --- | --- |
| id | INTEGER | Primary Key |
| name | TEXT | Holiday Name |
| date | TEXT | ISO Format |
| source | TEXT | Source of sync (API/Manual) |

**🧠 Backend API Endpoints**

| **Method** | **Endpoint** | **Purpose** |
| --- | --- | --- |
| GET | /api/calendar/events | Get all custom events |
| POST | /api/calendar/events | Add new event |
| PUT | /api/calendar/events/:id | Edit event |
| DELETE | /api/calendar/events/:id | Delete event |
| GET | /api/calendar/holidays/sync | Fetch and store holidays |
| GET | /api/calendar/holidays | List all saved holidays |

**✅ Summary of Features**

| **Feature** | **Status** |
| --- | --- |
| Add/Edit Alarms & Activities | ✅ |
| Recurring Entries | ✅ |
| Drag & Drop Events | ✅ |
| Search + Filter | ✅ |
| Manual Holiday Sync (PH only) | ✅ |
| Holiday API Integration (read-only) | ✅ |
| In-app Notification Support (local only) | ✅ |
| Linked Invoice ID (optional) | ✅ |
| Glassmorphic Layout | ✅ |

**🧾 Invoicing – Supplies (with Inventory Integration)**

**📄 Page Title:** Invoicing – Supplies  
**🎯 Purpose:** Logs supplier purchases and automatically adds quantities to inventory stock.

**🧭 Layout Structure**

| **Section** | **Details** |
| --- | --- |
| Toggle Selector | New Invoice ↔ Update Invoice (top toggle) |
| Left Panel (300px) | Item Groups (tiles) |
| Center Panel | Inventory Item Tiles (touch-friendly) |
| Right Panel | Invoice Cart, Totals, Save |

**📝 New Invoice Mode**

**🆔 Auto-Generated Invoice ID**

* Format: I-YYYYMMDD-####
* Counter resets daily

**📥 Required Fields**

| **Field** | **Required** | **Behavior** |
| --- | --- | --- |
| Supplier Name | ✅ | Dropdown from **Suppliers module** |
| Invoice Date | ✅ | Defaults to today |
| Payment Type | ✅ | Cash, GCash, Bank Transfer |
| Reference No. | ❌ | Required if GCash or Bank |
| Bank Name | ❌ | Required if Bank Transfer |
| OR Number | ❌ | Optional |
| Notes / Remarks | ❌ | Optional (appears in invoice footer) |

**📦 Item Selection (Inventory-Linked)**

* All items come directly from the **Inventory module**
* Selecting an item auto-fills:
  + Item Name, SKU, Unit Price
* Editable field: Quantity Received

✅ **Touch-friendly grid** based on selected Item Group  
✅ Only shows **active (non-archived)** inventory items

**🛒 Invoice Cart Panel (Right Side)**

| **Column** | **Behavior** |
| --- | --- |
| SKU | From Inventory |
| Item Name | From Inventory |
| Unit Price | From Inventory (non-editable) |
| Quantity | Editable |
| Subtotal | Computed |
| ❌ Remove | Button to remove line |

🟦 Auto-scroll if more than 8 items  
🟩 Cart total updates dynamically

**💾 Save Invoice – Behavior**

When the user presses **Save Invoice**:

* ✅ Validates all required fields
* ✅ Saves invoice into supplies\_invoices table
* ✅ Each item saved into supplies\_invoice\_items
* ✅ **Updates Inventory:**
  + inventory.current\_stock += qty\_received
* ✅ Triggers Expense log in expenses table
* ✅ Optionally opens Invoice Print Preview

**🔁 Update Invoice Mode**

* Allows editing of:
  + Date
  + Payment Info
  + Cart items (quantities or removal)
  + Notes
* Supplier Name & Invoice ID are **locked**
* Inventory quantities are **adjusted** based on the difference:
  + If original was 10 and changed to 12 → add 2
  + If changed from 10 to 7 → subtract 3

**🖨️ Invoice Print Preview (A5 Format)**

| **Section** | **Description** |
| --- | --- |
| Header | Business Info from Settings |
| Invoice Info | ID, Date, Payment Method, Ref No |
| Supplier Info | Name, Address, Contact (if available) |
| Table | Item, Qty, Unit Price, Subtotal |
| Footer | Total Amount, Remarks, OR#, Signatures |

🖨 Compatible with A5 or Half-A4  
🧾 Can be saved as PDF

**📤 Export Function**

* Export all supply invoices
* Format: .csv or .xlsx
* Includes: All fields from invoice + items

**📚 Backend Integration**

| **Module** | **Interaction** |
| --- | --- |
| 📚 Inventory | Adds stock automatically on Save |
| 💰 Expenses | Generates expense line with Amount + Supplier |
| 🏦 Banks | Validates Bank Transfer input |
| 🧑‍💼 Suppliers | Populates Supplier dropdown |
| ⚙️ Settings | Invoice format, printer config, business info |

**🧠 Backend Tables**

**supplies\_invoices**

| **Column** | **Type** |
| --- | --- |
| invoice\_id | TEXT (PK) |
| supplier\_id | TEXT |
| date | DATE |
| payment\_type | TEXT |
| ref\_number | TEXT |
| bank\_name | TEXT |
| or\_number | TEXT |
| total\_amount | REAL |
| notes | TEXT |
| created\_by | TEXT |

**supplies\_invoice\_items**

| **Column** | **Type** |
| --- | --- |
| invoice\_id | TEXT (FK) |
| item\_id | TEXT (FK) |
| qty | INTEGER |
| unit\_price | REAL |
| subtotal | REAL |

**🔁 Inventory Auto-Update Logic**

UPDATE inventory

SET current\_stock = current\_stock + :qty

WHERE item\_id = :sku;

Handled in brain/inventory.py as part of the invoice save flow.

**✅ Feature Checklist**

| **Feature** | **Status** |
| --- | --- |
| Inventory-linked item selector | ✅ |
| Auto-stock addition on save | ✅ |
| Supplier + payment validation | ✅ |
| A5 invoice print with full details | ✅ |
| Update mode with stock delta logic | ✅ |
| Expense integration | ✅ |
| Export to CSV/XLSX | ✅ |
| OR# and notes support | ✅ |
| Touchscreen friendly layout | ✅ |
| Fixed layout for 1920×1080 | ✅ |

**🧾 Invoicing > Maintenance – Full Developer Blueprint (v2.0)**

📄 **Page Title:**  
Invoicing – Maintenance

🎯 **Purpose:**  
Logs service-related transactions (cleaning, repairs, technical maintenance) linked to billers. Invoices trigger entries in the **Expenses** module and optionally schedule reminders via the **Calendar**.

**🖼️ Layout & Behavior**

| **Section** | **Description** |
| --- | --- |
| **Toggle Button** | New Invoice / Update Invoice |
| **Left Panel (300px)** | Biller and invoice details |
| **Right Panel (1320px)** | Notes entry + preview summary |
| **Scroll Behavior** | Scroll only if content overflows |
| **Touch-friendly UI** | For field selection and date/time picking |

**🧾 New Invoice Mode**

**🔢 Invoice Details**

| **Field** | **Required** | **Description** |
| --- | --- | --- |
| **Invoice ID** | Auto | Format: I-YYYYMMDD-#### (resets daily) |
| **Invoice Date** | ✅ | Auto-filled, editable |
| **Biller Name** | ✅ | Dropdown from Billers module |
| **Contact Person** | ✅ | Auto-filled with biller’s is\_primary = True contact. |
| **Contact Number** | ✅ | From selected contact |
| **Change Contact Person** | 🔘 | Opens modal selector |
| **Notes / Description** | ✅ | Multiline text, summary of maintenance activity |
| **Amount** | ✅ | Total expense amount |
| **Payment Type** | ✅ | Cash / GCash / Bank Transfer |
| **Reference #** | Conditionally Required | Required if GCash or Bank Transfer |
| **Bank Name** | Conditionally Required | Dropdown (from Banks module) |
| **OR Number** | Optional | Manual BIR OR entry |

**🛠 Change Contact Person – Modal**

Modal shows all biller-linked contacts:

| **Name** | **Contact Number** | **Channel** | **Status** | **Select Button** |
| --- | --- | --- | --- | --- |

* Only Active contacts shown
* Disallows selection of Delisted contacts
* If none exist: modal suggests visiting Billers module

**🔔 Reminder/Alarm Prompt (After Save)**

After clicking **Save Invoice**:

“Would you like to set a maintenance reminder for this activity?”

✅ Options:

* **Yes** → Opens Calendar entry modal
* **No** → Skips reminder setup

If **Yes**, the system will:

* Open **Add Event Modal** with fields prefilled:
  + **Title:** "Maintenance – [Biller Name]"
  + **Date/Time:** From invoice
  + **Type:** Activity
  + **Description:** Copied from invoice notes
  + **Notify:** Checkbox to enable in-app reminder
* Saved to Calendar module (tagged with Invoice ID)

**📤 Save Behavior**

* Invoice record stored in expenses table
* Line item (summary-level) recorded in expense\_line\_items
* Triggers print preview if enabled
* Reminder stored in calendar\_events if created

**🔁 Update Invoice Mode**

Activated via toggle. Features:

| **Field** | **Behavior** |
| --- | --- |
| **Invoice Selector** | Search by Invoice ID, Biller, Date |
| **Fields** | Editable: Notes, Contact Person, Amount, Payment Info |
| **Invoice ID & Biller** | 🔒 Not editable |
| **Save** | Updates invoice, line item, and optionally calendar |
| **Reminder** | If invoice already linked to calendar → asks if user wants to reschedule |

**🖨️ Invoice Preview (Half A4)**

| **Section** | **Details** |
| --- | --- |
| **Header** | Business Name, Address, TIN, Invoice Title |
| **Biller Info** | Name, Contact Person, Contact Number |
| **Invoice Info** | ID, Date, Notes, Amount |
| **Footer** | OR #, Signature field (optional) |
| **Print Style** | CSS clean, A5 portrait |

**📤 Export / Integration**

| **Feature** | **Behavior** |
| --- | --- |
| **Export** | CSV/XLSX: All fields + contact person details |
| **Calendar** | If reminder set, calendar entry is linked via Invoice ID |
| **Expenses** | Invoice pushes one Expense entry with matching details |
| **Banks** | Required for Bank Transfer method |
| **Billers** | Used to load name and contacts |

**🧠 Backend Tables**

**📄 expenses**

| **Field** | **Type** |
| --- | --- |
| invoice\_id | STRING |
| type | ENUM (‘Maintenance’) |
| biller\_id | FK |
| contact\_name | STRING |
| contact\_number | STRING |
| notes | TEXT |
| payment\_method | ENUM |
| ref\_number | STRING |
| bank\_name | STRING |
| or\_number | STRING |
| amount | NUMERIC |
| date | DATETIME |

**📄 expense\_line\_items**

| invoice\_id | particular | amount | status | created\_at |

**📄 calendar\_events (if reminder is created)**

| title | type | datetime | notes | linked\_invoice\_id | recurrence | notify |

**✅ Sample Workflow**

1. User creates a maintenance invoice for “AquaClean Corp”
2. Default contact “Engr. Vic” is loaded
3. User edits amount, notes, etc.
4. On save, system asks if user wants to create reminder
5. User sets it for 2 days later
6. Calendar now shows event with invoice link
7. Expenses updated with invoice & line item

**🧾 Invoicing > Billing – Developer Blueprint (v2.0)**

📄 **Page Title:**  
Invoicing – Billing

**🎯 Purpose**

Used to record regular company billing transactions (utilities, rent, services). Each entry is saved as a billing invoice, auto-logged in the **Expenses** module, and printable in A5 format.

**🧱 Layout & Workflow**

| **Section** | **Description** |
| --- | --- |
| Toggle Mode | New Invoice / Update Invoice |
| Layout | 2-column layout: left form, right summary |
| Scroll | Scrollable on form if needed |
| Screen Fit | Designed for 1620px wide content area (touch-friendly) |

**🧾 New Billing Invoice**

**🔢 Invoice Header**

| **Field** | **Type** | **Behavior** |
| --- | --- | --- |
| **Invoice ID** | Auto | Format: I-YYYYMMDD-#### (reset daily) |
| **Invoice Date** | Date Picker | Defaults to today |
| **Billing Period** | Dropdown or Month Picker | Format: MM/YYYY |
| **Due Date** | Optional | Date field |
| **Biller Name** | Dropdown | Pulled from Billers module |
| **Contact Person (Name + Number)** | Auto-filled | From biller’s primary contact |
| **Change Contact Button** | Modal | Select any other Active contact from biller profile |

**🧾 Invoice Details**

| **Field** | **Required** | **Description** |
| --- | --- | --- |
| **Particular / Description** | ✅ | Short billing description (e.g., "MERALCO – July 2025") |
| **Amount Due** | ✅ | ₱ input |
| **Payment Type** | ✅ | Dropdown: Cash, GCash, Bank Transfer |
| **Reference Number** | Conditionally ✅ | Required for GCash or Bank |
| **Bank Name** | Conditionally ✅ | Dropdown (Banks module) |
| **OR Number** | Optional | Alphanumeric, 20 char max |

**✅ Save Behavior**

* Validates required fields
* Saves invoice to:
  + expenses table (type = Billing)
  + expense\_line\_items table (single line)
* Optionally opens **print preview** (if enabled in Settings)
* Automatically adds contact details if selected

**🔁 Update Billing Invoice**

| **Feature** | **Behavior** |
| --- | --- |
| **Invoice Selector** | Search by: Invoice ID, Biller, Billing Period |
| **Editable Fields** | Description, Due Date, Amount, Payment Method, Reference No., Bank, OR Number, Contact Person |
| **Not Editable** | Invoice ID, Biller Name |
| **Save Updates** | Applies changes to database + linked Expense |
| **Calendar Reminder** | Optional future enhancement (not part of current blueprint) |

**🖨️ Invoice Printing**

| **Format** | **A5 (Half A4), Portrait** |
| --- | --- |
| Header | Business Name, Address, INVOICE |
| Info | Invoice ID, Date, Billing Period |
| Biller | Name, Contact Person, Number |
| Details Table | Description + Amount |
| Payment | Method, Reference, OR |
| Footer | Signature, Notes section (optional) |

**📤 Export Function**

| **Feature** | **Behavior** |
| --- | --- |
| Export Scope | Filtered or full billing list |
| Format | CSV/XLSX |
| Filename | billing\_export\_YYYYMMDD-HHMMSS.csv |
| Columns |  |

* Invoice ID
* Invoice Date
* Billing Period
* Biller Name
* Due Date
* Description
* Amount
* Payment Type
* Reference #
* Bank Name
* OR Number
* Contact Person Name + Number

**📚 Integrations**

| **Module** | **Description** |
| --- | --- |
| **Billers** | Used to load name and contact persons |
| **Banks** | Used if payment type is Bank Transfer |
| **Expenses** | Auto-creates expense entry upon save |
| **Settings** | Controls print headers, A5 formatting |

**🧠 Backend Tables**

**📄 expenses**

| **Field** | **Type** | **Notes** |
| --- | --- | --- |
| invoice\_id | TEXT | Format: I-YYYYMMDD-#### |
| type | TEXT | Always Billing |
| biller\_id | FK | References billers table |
| contact\_name | TEXT | Selected contact |
| contact\_number | TEXT | Contact mobile |
| notes | TEXT | Billing description |
| amount | NUMERIC | Total ₱ |
| payment\_method | ENUM | Cash, GCash, Bank Transfer |
| ref\_number | TEXT | Optional |
| bank\_name | TEXT | Optional |
| or\_number | TEXT | Optional |
| invoice\_date | DATETIME | Date of billing |
| billing\_period | TEXT | Format: MM/YYYY |
| due\_date | DATETIME | Optional |

**📄 expense\_line\_items**

| invoice\_id | particular | amount | created\_at |

**✅ Example Usage**

1. User selects "MERALCO" as biller
2. Billing Period: July 2025
3. Contact auto-filled: Engr. Carlo, 0917xxxxxxx
4. Enters: Description = "Electric Bill", Amount = ₱10,200
5. Payment Type: Bank Transfer (BDO), Reference = 123ABC
6. Clicks Save → expense saved + invoice preview shown

**📈 Module: Reports**

**📄 Page Title: Reports Dashboard**

A centralized reporting module that summarizes business performance using charts, tables, and exportable reports. It includes tabs for key areas such as sales, expenses, inventory, and rewards. A placeholder is added for future custom report builder functionality.

**🧭 Purpose**

To provide summarized, filterable business data for decision-making, covering:

* Orders & Sales
* Expenses
* Inventory Movement
* Rewards Activity
* Return Logs
* Supplier Deliveries
* And in the future: **Custom Reports**

**🧱 Layout Structure**

| **Section** | **Description** |
| --- | --- |
| Sidebar (Tabs) | Vertical tab selector for each report type |
| Main Panel | Shows charts, filters, and tables depending on selected tab |
| Screen Width | Max 1620px (excluding 300px sidebar) |
| Scroll Behavior | Scroll inside main report panel only |
| Export | Each report has its own Export button |

**📁 Tabs / Report Categories**

**1. 🛒 Sales Reports**

| **Subsection** | **Description** |
| --- | --- |
| Date Range Picker | Filter by From–To (default: this month) |
| Group By | Day / Week / Month / Year |
| Chart 1 | Line chart: Daily Sales Total (₱) |
| Chart 2 | Bar chart: Top 10 Sold Items |
| Table | Orders summary: Order #, Customer, Amount, Payment Method, Status |
| Export | Orders summary (filtered) as CSV/XLSX |

**2. 💰 Expenses Reports**

| **Subsection** | **Description** |
| --- | --- |
| Filters | Type (Supplies, Maintenance, Billing), Date Range, Entity (Supplier/Biller) |
| Chart | Pie chart: Expense Breakdown by Source |
| Table | Invoice ID, Date, Linked Entity, Amount, Payment Method, OR # |
| Export | Expense table to CSV/XLSX |

**3. 📦 Inventory Movement**

| **Subsection** | **Description** |
| --- | --- |
| Date Range | From–To |
| Group By | Item, Group, Supplier |
| Table | Item ID, Name, Qty Received, Qty Sold, Remaining Stock, Last Supplied From (Invoice ID) |
| Export | Stock movement report to CSV/XLSX |

**4. 🎁 Rewards Activity**

| **Subsection** | **Description** |
| --- | --- |
| Date Range | From–To |
| Filters | Earned, Used, Expired, Reversed |
| Chart | Line graph: Points Earned vs Used per day |
| Table | Rewards ID, Customer, Order ID, Points, Equivalent, Type, Expiry |
| Export | Ledger export (filtered) to CSV/XLSX |

**5. 🔁 Returns Summary**

| **Subsection** | **Description** |
| --- | --- |
| Filter by | Date, Return Reason, Return Type (Full/Partial) |
| Chart | Bar chart: # of Returns per Reason |
| Table | Return ID, Order ID, Items Returned, Total Refunded, Processed By |
| Export | Return logs (filtered) to CSV/XLSX |

**6. 📥 Supplier Deliveries**

| **Subsection** | **Description** |
| --- | --- |
| Filter by | Supplier Name, Item Name, Date Range |
| Table | Invoice ID, Supplier, Date, Item Supplied, Qty, Unit Price, Total |
| Export | Supplies received report (per item or per supplier) |

**7. 🧠 Custom Reports *(Future)***

| **Feature** | **Description** |
| --- | --- |
| Status | Not yet interactive – placeholder only |
| UI Preview | Mock UI or disabled message like: “Custom Report Builder Coming Soon” |
| Plan | Will allow users to define report fields, filters, and generate custom CSV/XLSX exports |
| Optional | Could allow power users to run SQL-like filters or select predefined templates (Orders by Item Group, Rewards by Cashier, etc.) |

**🖨️ Export Behavior**

All reports can export the visible data as:

* .csv or .xlsx
* Filename format: reporttype\_export\_YYYYMMDD-HHMMSS.csv

**🧠 Backend API Suggestions**

| **Method** | **Endpoint** | **Description** |
| --- | --- | --- |
| GET | /api/reports/sales | Sales by date range |
| GET | /api/reports/expenses | Expenses summary |
| GET | /api/reports/inventory | Inventory movement |
| GET | /api/reports/rewards | Rewards ledger summary |
| GET | /api/reports/returns | Returns log summary |
| GET | /api/reports/supplies | Supplies received |
| GET | /api/reports/custom | Future endpoint for advanced filtering |

**✅ Feature Checklist**

| **Feature** | **Status** |
| --- | --- |
| Sales summary & filters | ✅ |
| Expenses by type & entity | ✅ |
| Inventory stock movement | ✅ |
| Rewards usage logs | ✅ |
| Returns with breakdowns | ✅ |
| Supplier delivery summary | ✅ |
| Export (all tabs) | ✅ |
| Custom Report placeholder | ✅ |
| Charts + tables per report | ✅ |

Here is the **final and complete updated blueprint** for the **📊 Dashboard Module**, with enforced enhancements and aligned integrations from all the previously finalized modules.

**📊 Module: Dashboard**

**📄 Page Title: Dashboard**

**🎯 Purpose**

The dashboard provides a dynamic overview of business health through real-time stats, visual summaries, and time-based insights across sales, expenses, rewards, and scheduled events.

**🧱 Layout Overview**

| **Section** | **Details** |
| --- | --- |
| Sidebar (Tabs) | Fixed vertical tab menu (300px width) |
| Main Panel | Right side (1320px), content changes based on active tab |
| Theme | Glassmorphic, neon-blue styling, full 1920×1080 screen fit |
| Scroll Behavior | Scroll only within each tab’s content, not the sidebar |
| Responsiveness | All charts scale appropriately within 1620px max width |

**📁 Tabs**

**1. 🛒 Sales Tab**

**🔹 Top Widgets (1/3 Height)**

| **Widget** | **Notes** |
| --- | --- |
| 🧾 Total Sales Today | Sum of all Paid orders today |
| 💳 Sales by Payment Type | % breakdown: Cash, GCash, Bank Transfer |
| 🏆 Highest-Selling Item | Based on quantity sold today |
| 🕒 Latest Order Time | Timestamp of most recent Paid order |

**🔹 Main Panel (2/3 Height)**

| **Feature** | **Description** |
| --- | --- |
| Sales Trend Line Graph | Shows Total Sales over time (daily/weekly/monthly) |
| Filters | Date Range, Payment Method, Customer (searchable) |
| Export Button | Exports filtered orders to CSV/XLSX |
| Calendar Highlight | High-sales days auto-tagged in Calendar Tab |

**2. 💸 Expenses Tab**

**🔹 Top Widgets**

| **Widget** | **Notes** |
| --- | --- |
| 💰 Total Expenses Today | From all finalized invoices |
| 🧾 Expense by Source | Pie chart (Supplies, Maintenance, Billing) |
| 🏢 Top Supplier / Biller | Based on today's expense amount |
| 🕒 Latest Expense Invoice | Shows last invoice date & ID |

**🔹 Chart Panel**

| **Feature** | **Description** |
| --- | --- |
| Expense Breakdown Pie | Slice per source or per supplier/biller |
| Filters | Date Range, Expense Type, Supplier/Biller |
| Export Button | Exports expenses to CSV/XLSX |
| Notes Section | Shows invoice remarks if available |

**3. 📊 Overall Tab**

**🔹 Summary Widgets**

| **Widget** | **Notes** |
| --- | --- |
| 📈 Sales vs Expenses (Today / Week / Month) | Bar chart or cards |
| 🧮 Net Balance | Sales - Expenses |
| ⚠️ Top 3 Expense Categories | Based on amount spent |
| 🎁 Reward Points Issued/Redeemed | With mini stacked bar or donut chart |

**🔹 Main Graph**

| **Feature** | **Description** |
| --- | --- |
| Overlay Line Chart | Sales and Expenses over time |
| Group By | Day, Week, or Month |
| Filters | Date Range Picker |

**4. 📅 Calendar Tab**

| **Feature** | **Description** |
| --- | --- |
| View | Month, Week, Day toggle |
| Events Shown | Maintenance Activities, Holidays, Alarms, Sales Peaks |
| Color Legend | 🔔 Alarms (Red), 🛠 Maintenance (Blue), 🎉 Holidays (Green), 🟢 Sales Peaks (Heatmap style) |
| Add Event Button | Adds maintenance or alarm (linking to Calendar module) |
| Sync Holidays Button | Manual trigger, gets Philippine holidays (offline safe) |
| Filters | Checkboxes: Show Alarms, Show Activities, Show Holidays, Show High Sales Days |

**🔄 Enhancements Enforced from Other Modules**

| **From Module** | **Impact on Dashboard** |
| --- | --- |
| **Orders** | Pulls order data (status, payment method, customer) |
| **Expenses** | Filters by type, links to invoices |
| **Rewards Ledger** | Points summary shown under Overall tab |
| **Returns** | Returned or partially returned orders excluded from net sales calc |
| **Supplies** | Tracked as incoming inventory + expense |
| **Maintenance** | Scheduled alarms reflected in calendar |
| **Settings** | Timezone affects timestamp display |

**🔐 Permissions / Notes**

* All tabs visible to Admins.
* View-only for Cashiers unless dashboard roles are enabled in future.
* All widgets are clickable (planned enhancement): opens full module filtered to match widget context (e.g., clicking "Top Supplier" filters Expenses).

**📤 Export Behavior**

* All tabs support exporting current view to .csv and .xlsx
* Filename: dashboard\_{tab}\_export\_YYYYMMDD-HHMMSS.csv
* Exports tables behind charts where applicable

**🧠 Backend API Suggestions**

| **Method** | **Endpoint** | **Description** |
| --- | --- | --- |
| GET | /api/dashboard/sales-summary | Sales widgets data |
| GET | /api/dashboard/sales-trend | Line chart data |
| GET | /api/dashboard/expenses-summary | Expenses widgets |
| GET | /api/dashboard/expenses-trend | Expense trend data |
| GET | /api/dashboard/overall | Combined summary of sales, expenses, rewards |
| GET | /api/dashboard/calendar-tags | Returns high-sales dates |

**✅ Feature Checklist**

| **Feature** | **Status** |
| --- | --- |
| 4-Tab structure (Sales, Expenses, Overall, Calendar) | ✅ |
| Dynamic widgets per tab | ✅ |
| Sales trends with filter/export | ✅ |
| Expense breakdowns by category/entity | ✅ |
| Rewards metrics integration | ✅ |
| Calendar visual with event types | ✅ |
| CSV/XLSX export per tab | ✅ |
| High-sales heatmap integration | ✅ |
| Timezone & date format sync | ✅ |

Let me know if you'd like a **daily auto-generated printable dashboard summary** feature or **pin-to-top custom KPIs**.