

Inventory Management & Reorder Automation System (IMRAS)

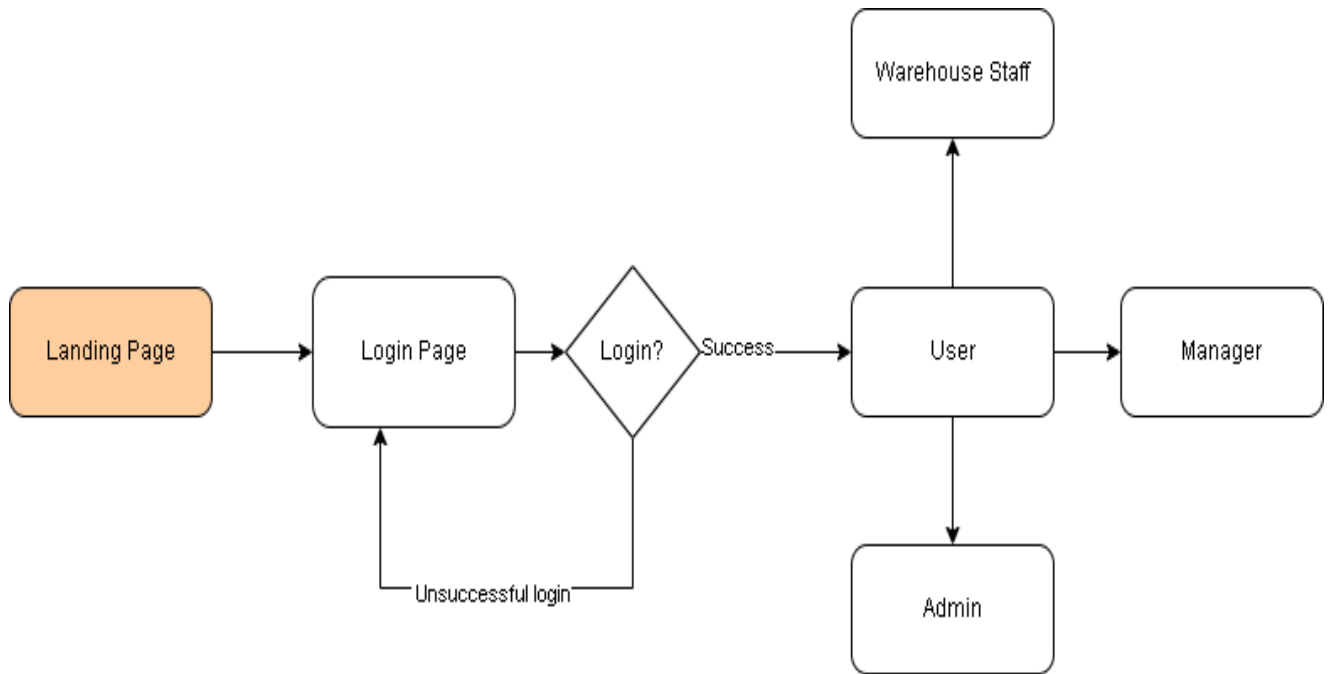
Problem Statement: Many small-to-medium operations rely on spreadsheets or disconnected systems for stock tracking, causing stock outs, overstocking, expired items, and inaccurate demand forecasts. Purchase and warehouse teams lack automated reorder signals, batch/lot tracking, and easy auditability.

Objectives: To create a full-stack Inventory Management System that tracks stock in real-time, supports batch/lot/expiry where applicable, automates reorder suggestions, manages suppliers/POs, and provides role-based access for Warehouse Staff, Inventory Manager, and Admin.

Scope of project:

- Roles: Warehouse staff, Inventory Manager, Admin
- User management
- Items management
- Auto reorder suggestions and reorder rule management
- PR, PO, GRN creation/approval workflow
- Real time stock movement
- Batch/serial number, expiry tracking

Simple system flow:

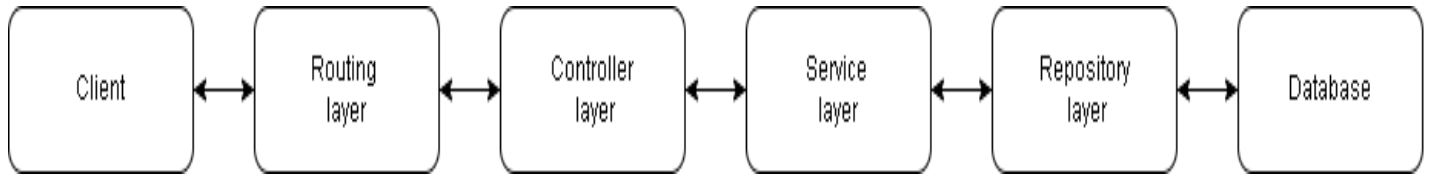


Technologies used:

Usage	Technology
UI	React.js
Routing	React Router
API Calls	Axios
Backend	Node.js + Express.js
Database	PostgreSQL

Backend Implementation idea:

- **Architecture:** Layered monolith
- **Flow diagram:**



- **Justification:**

1. Routing Layer

- Maps HTTP methods and URLs to controller actions
- Applies middleware such as authentication and validation
- Acts as the single entry point for incoming request

2. Controller Layer

- Parses incoming requests and extracts required data
- Delegates processing to the service layer
- Formats and returns HTTP responses

3. Service Layer

- Implements core business logic and rules
- Handles validation and authorization decisions
- Coordinates data access through repositories

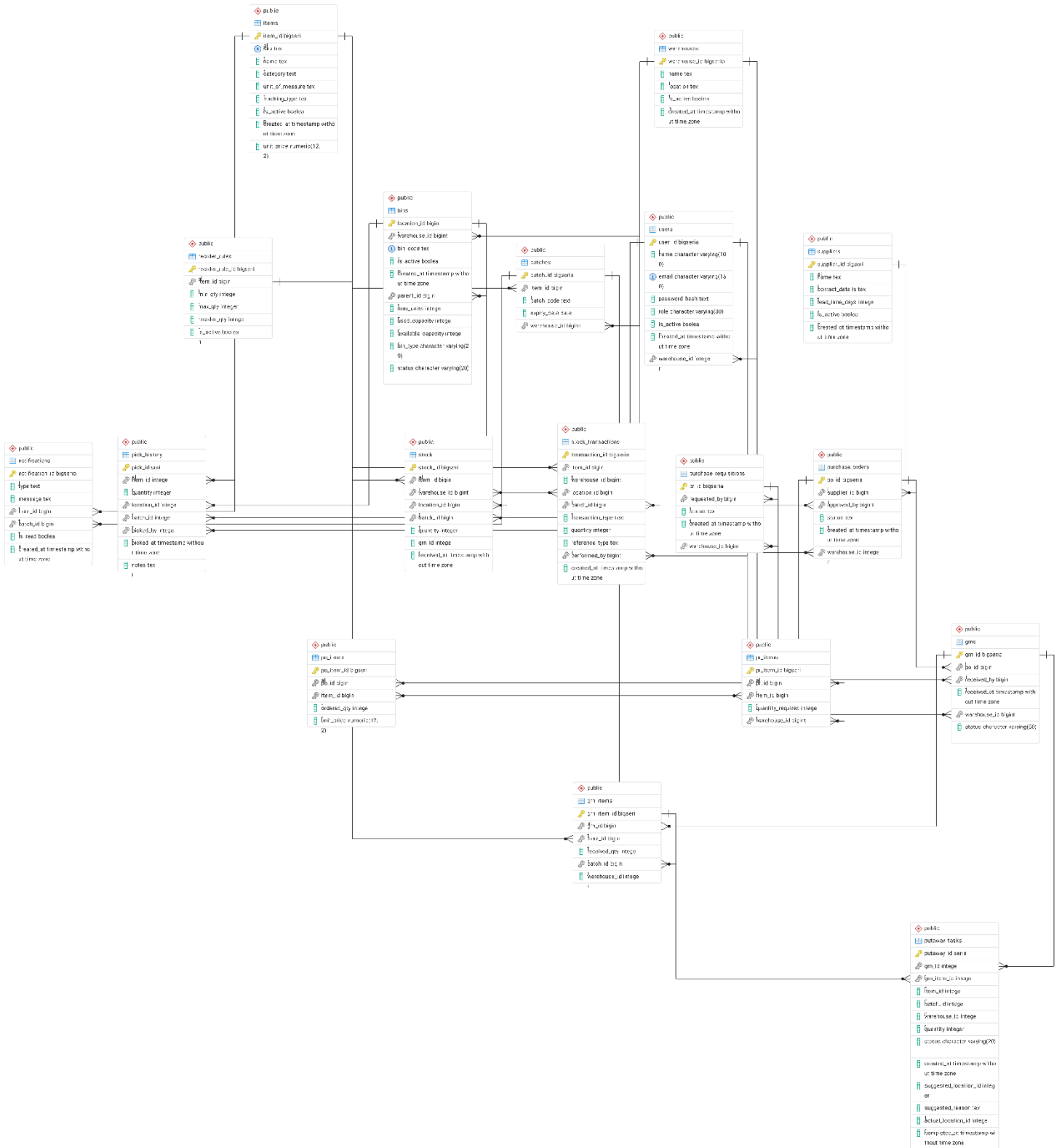
4. Repository Layer

- Encapsulates database queries.
- Provides data access methods to services
- Abstracts persistence details from business logic

5. Database Layer

- Stores and retrieves application data persistently
- Enforces data integrity using constraints and transactions
- Optimizes performance through indexing and schema design

ER Diagram:

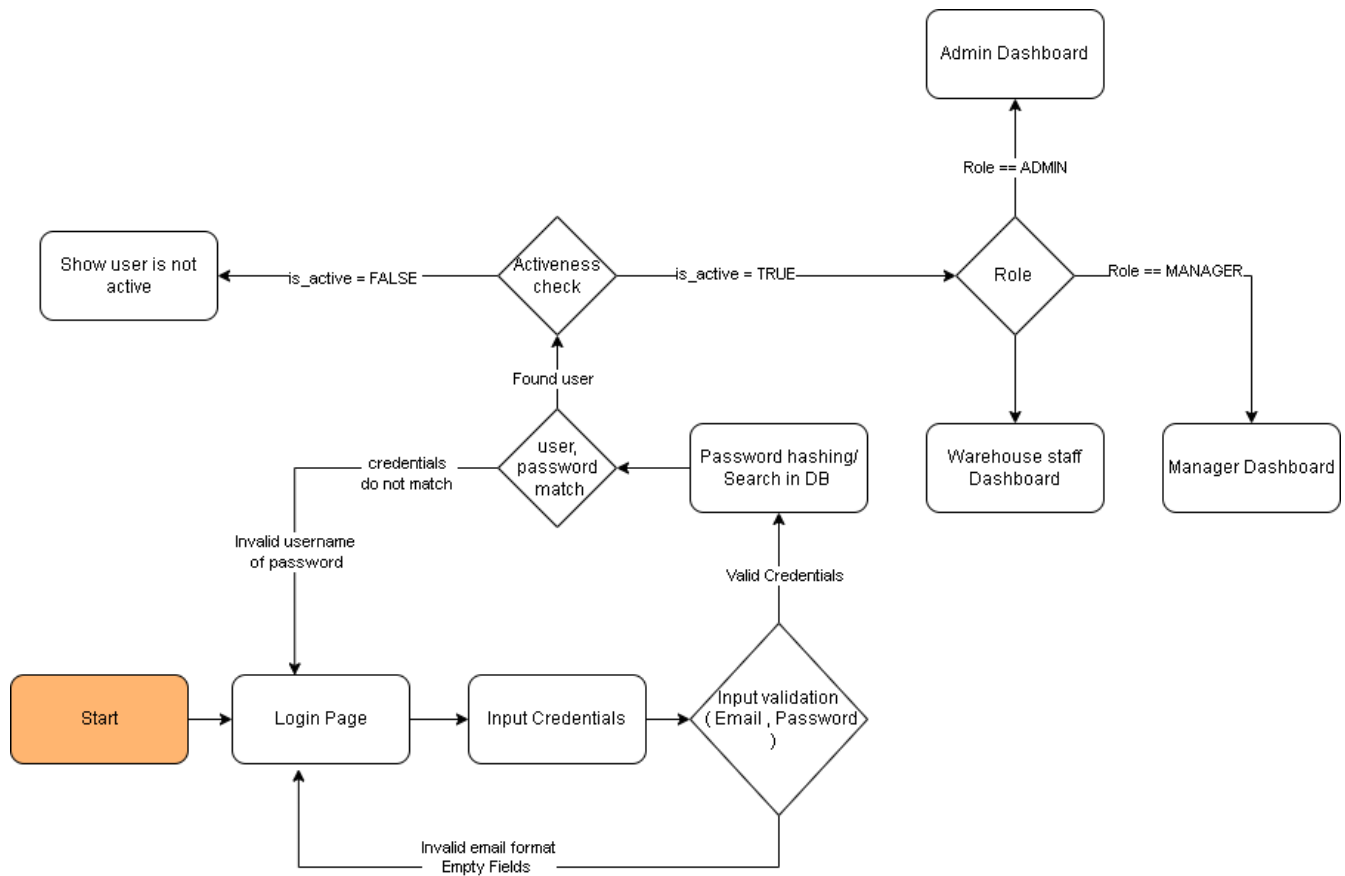


Module 1: Authentication and Authorization

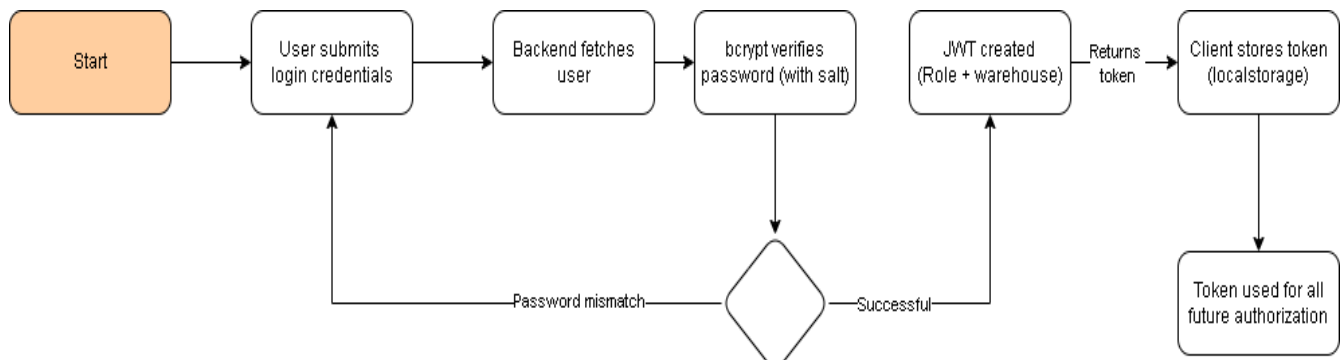
- **Goal:** To provide secure user authentication and role-based authorization, ensuring controlled access to IMRAS functionalities and protecting critical reorder automation processes.
- **Objectives:**
 - Prevent unauthorized access to inventory and reorder operations
 - Verify user identity using secure authentication mechanisms
 - Enforce role-based access control across system features
 - Enforce warehouse based access control for warehouse staff and Manager (here consideration is for one warehouse there is only one manager and multiple warehouse staff)
- **Key implementation idea:**
 - **Assumptions:**
 - Users are pre-registered by Admin.
 - Each user has at least one warehouse assigned.
 - Admin user is already registered through seeding.
 - **Data entities required:**
 - **User table –**
 - User_id – Unique identifier
 - Name – Name for each user
 - Email – Login credential
 - Password – Login credential (stored in the form of hash)
 - Role – included to enforce RBAC
 - Is_active – Used for verification of activeness
 - Warehouse_id – To enforce warehouse based control
 - **Technologies (Module specific):**

Purpose	Technology
Password Security	bcrypt
Authentication	JWT
Authorization	RBAC + WBAC
DB Access	node-postgres(pg)

○ Flow diagram for login:



○ Internal working of login:



- **JWT Structure:**

- **Payload:**

- 1. User_id
 - 2. Role
 - 3. Warehouse_id

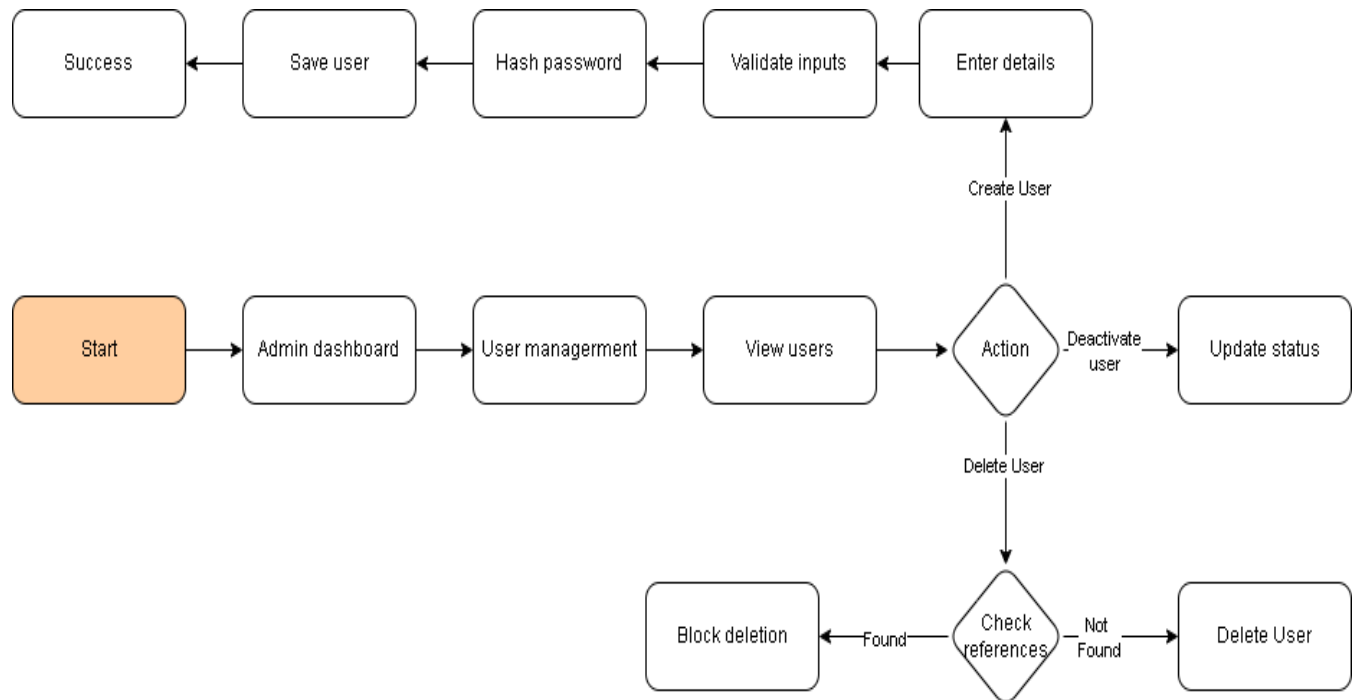
- **Response:**

- 1. Token
 - 2. User.id
 - 3. User.name
 - 4. User.role
 - 5. User.warehouse_id

Module 2: User Management

- **Goal:** The goal of the User Management module in IMRAS is to provide a centralized and secure mechanism for managing system users, including creating, updating, activating, deactivating, and deleting user accounts.
- **Objectives:**
 - To allow administrators to create and manage user accounts within the system.
 - To assign appropriate roles to users based on organizational responsibilities.
 - To associate users with specific warehouses for controlled data access.
 - To enable activation and deactivation of user accounts without data loss.
- **Actors:**
 - Admin (primary actor)
 - Manager / Staff (indirectly affected)
 - For Admin:
 - Create users
 - Assign roles
 - Assign warehouses
 - Activate / deactivate users
 - Delete users (with constraints)
- **Assumption:**
 - Only admin has right to create the user.
 - There should be only one manager for a warehouse.
 - There can be multiple staff for a single warehouse.

Flow diagram for User management:



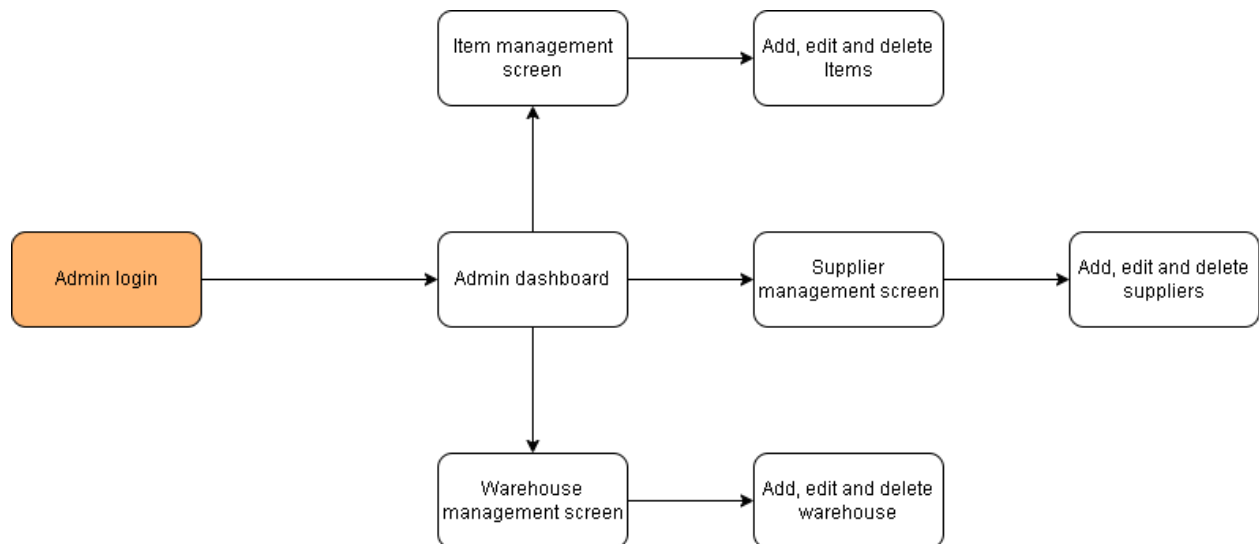
Input fields for Admin to create User:

1. Name
2. Email
3. Password
4. Role – (Staff , Manager)
5. Warehosue – (Warehouses fetched from available warehouses in db)

Module 3: Item, supplier and warehouse management:

- **Purpose:**
 - The purpose of this module is to centrally define and manage all inventory-related master data required by other operational modules of IMRAS.
 - It ensures that items, suppliers, and warehouses are standardized, validated, and consistently referenced across the system without performing any stock inflow or outflow operations.
- **Goal:**
 - Enable accurate configuration of inventory parameters (name, location), item parameters (SKU, item name, unit of measure, unit price)
 - Support downstream modules (Procurement, Sales, Stock Movement, Reporting)
- **Primary actor:** System Administrator

Flow diagram for module 3:



Module 4: Reorder rules

- **Purpose:** The purpose of the Reorder Rules module is to define and manage inventory replenishment policies that determine *when* and *how much* an item should be reordered. This module serves as a configuration and decision-support layer, supplying rules and thresholds to operational modules without executing any purchasing or stock transactions.
- **Goal:**
 - Establish consistent and automated reorder logic
 - Prevent stockouts and overstocking through predefined rules
 - Separate reorder decision rules from procurement execution
- **Primary actor:** System administrator (can add, delete and edit rules)
- **Assumption:**
 - Each item has only one reorder rule
- **Primary fields in reorder rule:**
 - Item name
 - Min qty
 - Max qty
 - Reorder qty
- **Flow diagram:**



Module 5: Procurement management

- **Purpose:** The purpose of the Procurement Management module is to manage the complete purchasing lifecycle—from identifying procurement needs to supplier engagement and purchase order approval—without performing physical stock receipt or inventory updates.

This module acts as the execution layer that converts reorder recommendations and manual requests into authorized purchase orders.

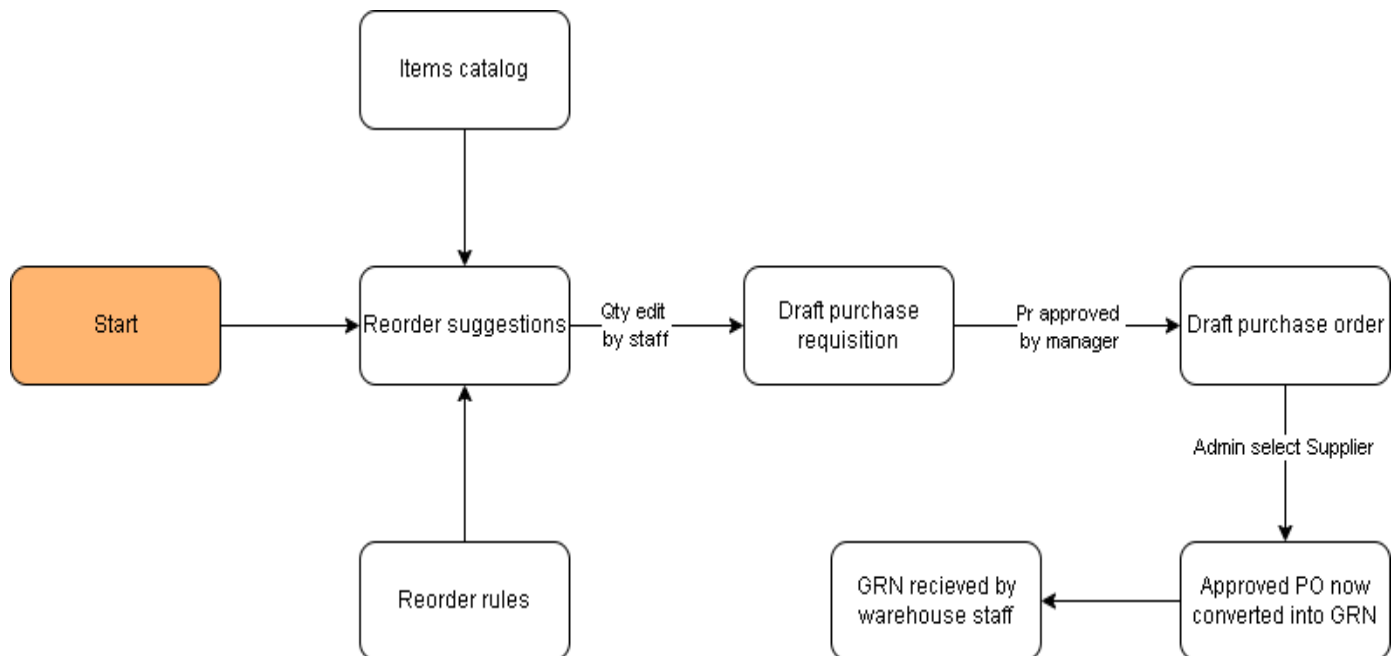
- **Goals:**
 - Enable structured and auditable procurement workflows
 - Convert reorder rules and manual requests into purchase orders
 - Maintain supplier accountability and purchasing history
 - Enforce approval controls and budget discipline
- **Primary Actors:**

Actor	Responsibility
Warehouse Staff	<ol style="list-style-type: none">1. Creates draft purchase requisition from reorder suggestion.2. Edits quantity required3. Receives good receipt notes4. Can edit received qty while accepting the GRN
Warehouse manager	<ol style="list-style-type: none">1. Approves purchase requisition drafted by staff
System administrator	<ol style="list-style-type: none">1. Creates purchase orders from approved purchase requisition2. Approve pending purchase orders.3. Can select supplier while approving purchase orders

- **Notes:**

- The reorder suggestions are derived from reorder rules.
- The authority of creating and approving purchase requisition can be different.
- In this system reorder suggestions are visible to warehouse staffs.
- The stock transaction(Updating) takes place when putaway is completed.
- Putaway is the next step warehouse staff must have to perform after receiving GRN.
- GRN only confirms that the physical stock is arrived.

- **Flow diagram:**



- **Result:** This module only creates and receives GRN's at the end. After receiving GRN a new putaway task is created after completing that the stock will update.

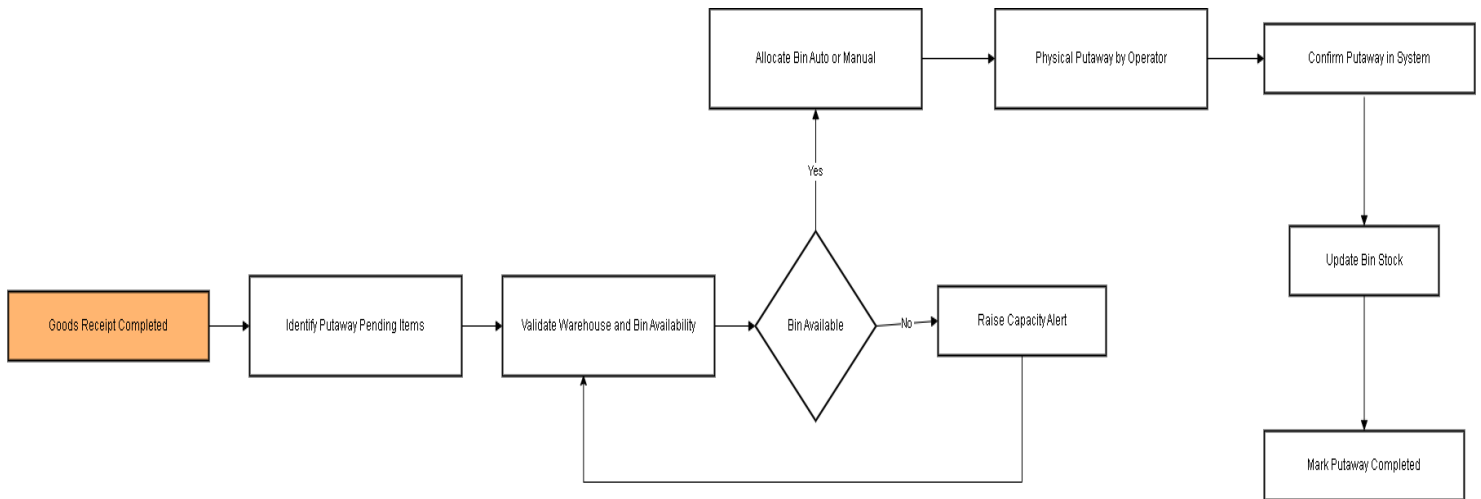
Module 6: Putaway management

- **Purpose:** The Putaway Management module ensures that inbound inventory received at a warehouse is systematically assigned to appropriate storage locations (bins) based on predefined rules such as space availability and item type.

This module minimizes manual intervention, prevents incorrect storage, and maintains real-time inventory accuracy.

- **Goals:**
 - Automate the putaway process after goods receipt
 - Ensure optimal utilization of warehouse bins
 - Maintain accurate stock-to-location mapping
 - Reduce storage errors and misplaced inventory
 - Provide traceability for all putaway transactions
- **Primary actor:**
 - **Warehouse Staff:**
 - Can see putaway tasks
 - complete tasks
 - assigns bins, and records transactions
- **Notes:**
 - This module is warehouse aware means, warehouse staff with can access and work on putaway tasks only of their corresponding warehouse.
 - This module suggests bin to staff to allocate location for new staff, it cant directly assigns the ideal location, staff must have to select that ideal location or suggested location.
 - The bins are suggested on the basis of space availability and item type(same or different)

- **Flow diagram:**



- **Result / Outcome:**

- **Inventory is stored in correct bin locations**
- **Real-time stock visibility at bin level is achieved**
- **Warehouse space is utilized efficiently**
- **Errors in storage and retrieval are significantly reduced**

Module 7: Picking Management

- **Purpose:**

The Picking Management module enables warehouse staff to perform direct item picking by entering the item SKU and quantity, while ensuring that stock is picked based on the First-In-First-Out (FIFO) principle.

This ensures proper stock rotation and prevents aging or expiry-related losses.

- **Goals**

- Enforce FIFO logic during picking
- Allow direct SKU-based picking without pick lists
- Automatically select oldest available stock
- Maintain bin-level and batch-level accuracy
- Reduce manual decision-making for stock selection

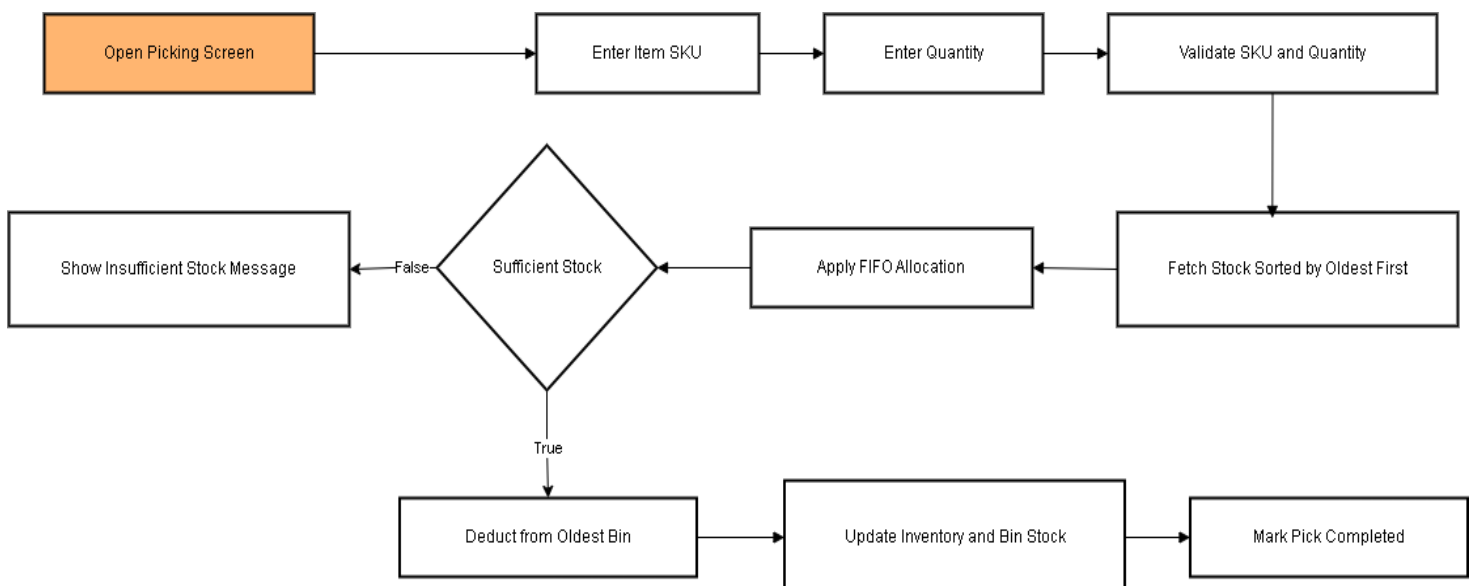
- **Primary Actors**

- **Warehouse Staff (Picker)** – Initiates and completes the picking task

- **Notes:**

- FIFO is applied based on inbound date / putaway timestamp.
- System automatically determines which bins to pick from.
- Picker does not choose bins manually, ensuring FIFO compliance.
- Quantity may be fulfilled from multiple bins if required.
- Notes field captures exceptions such as damaged or short picks.

- **Flow diagram:**



- **Result / Outcome**

- Oldest inventory is consumed first
- Inventory aging is controlled and optimized
- Stock records remain accurate and FIFO-compliant
- Picking speed is improved with minimal user input
- Full traceability of FIFO consumption is maintained

Module 8: Batch, expiry and stock age tracking

- **Purpose**

This module tracks inventory at batch level, monitors expiry dates, and calculates stock age for items stored within a specific warehouse.

Visibility is strictly limited so that a warehouse manager can view data only for their assigned warehouse.

- **Goals**

- Maintain batch-wise stock per warehouse
- Track expiry dates for warehouse-specific inventory
- Calculate stock age based on warehouse inbound dates
- Restrict data visibility to assigned warehouse only
- Support FIFO and aging-based warehouse operations

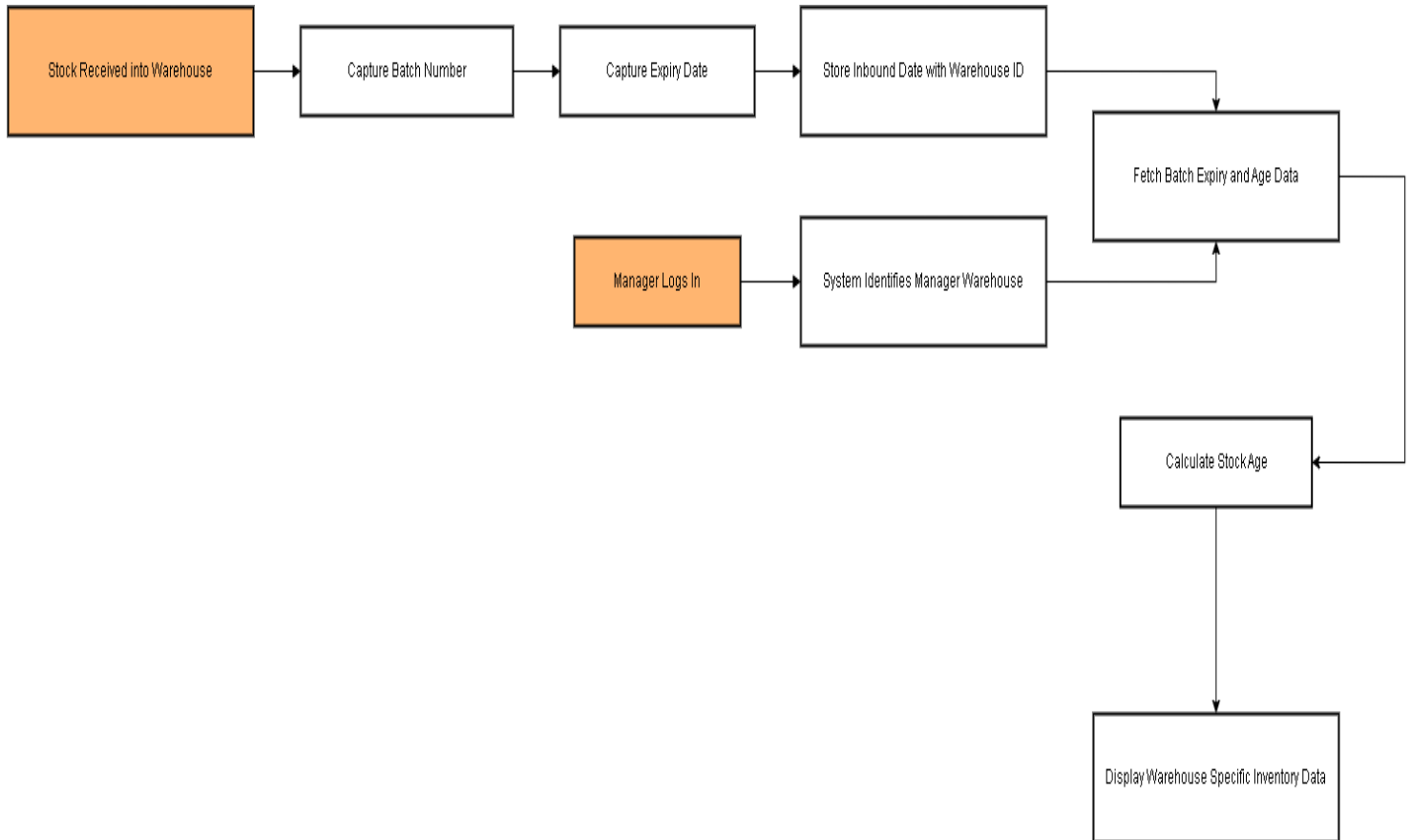
- **Primary Actors**

- **Warehouse Manager** – Views batch, expiry, and aging data for assigned warehouse

- **Process Notes**

- Each warehouse is mapped to exactly one manager.
- Manager access is filtered using warehouse_id from authentication token.
- Batch, expiry, and age calculations are performed per warehouse.
- Expired stock is excluded from picking in that warehouse only.
- Cross-warehouse visibility is not permitted

- **Flow diagram:**



- **Result:** Managers can successfully review batches expiry and stock age.

Module 9: Dashboards and reports

1. Purpose

The Dashboards & Reports module provides role-based visibility into system operations, inventory status, and performance metrics. It ensures that each user type accesses only the data relevant to their role and warehouse responsibility, while enabling management-level reporting and decision support.

2. Goals

- Provide role-specific dashboards for operational clarity
- Display real-time inventory and transaction insights
- Enable warehouse-aware reporting and analytics
- Support managerial decision-making through structured reports
- Enforce strict data access control based on user role

3. User Roles & Access Scope

User Role	Dashboard Access	Reports Access	Data Scope
Warehouse Staff	Yes	No	Assigned warehouse only
Inventory Manager	Yes	Yes	Assigned warehouse only
Admin	Yes	Yes	All warehouses combined

4. Dashboards Overview

4.1 System Administration Dashboard (Admin)

Purpose

Provides a system-wide overview of warehouses, users, and approval workflows.

Key Information Displayed

- Total active warehouses
- Total registered users
- Pending purchase order approvals
- Overview of all warehouses with status and location

4.2 Inventory Manager Dashboard

Purpose

Provides warehouse-level inventory insights for monitoring stock health and approvals.

Key Information Displayed

- Total stock value of assigned warehouse
- Items below reorder level
- Pending PR approvals
- Overstocked items
- Recent inventory transactions
- Assigned warehouse staff details

4.3 Warehouse Operations Dashboard (Warehouse Staff)

Purpose

Supports day-to-day warehouse operations and task execution.

Key Information Displayed

- Today's Goods Receipt Notes (GRNs)
- Pending putaway tasks

- Expiring batches (next 30 days)
- Low stock alerts
- List of expiring batches with days left

5. Reports Module

5.1 Purpose

The Reports module generates analytical and compliance reports related to inventory valuation, stock aging, and item movement over a selected time period.

5.2 Available Reports

- Stock Valuation Report – SKU-wise inventory value
- Stock Aging Report – Aging buckets for slow-moving stock
- Fast / Slow Moving Items Report – Item movement analysis

5.3 Report Filtering & Generation

- Reports support date range filtering
- Managers receive reports filtered by warehouse_id
- Admin reports aggregate data across all warehouses
- Reports can be generated on daily or weekly basis
- Export supported in CSV format

6. Data Access & Security Rules

- User role and warehouse ID are derived from authentication token
- All dashboard and report queries are filtered server-side
- Cross-warehouse access is strictly restricted for managers
- Admin users bypass warehouse filters for global visibility

7. Result / Outcome

- Clear operational visibility for warehouse staff
- Accurate warehouse-level control for inventory managers
- Centralized oversight for administrators
- Improved decision-making using real-time dashboards
- Secure, role-based access to sensitive inventory data

Implementation:

1. [Live site](#)
2. [Github](#)

Future Scope:

1. Addition of chatbot for admin and manager's for insights.
2. Addition of Pick request workflow(open user sign in -> requests specific item with quantity -> request shown to warehouse staff -> staff selects fifo optimized bin to pick -> request sent to manager -> manager approves request -> Pick task completes -> stock deducts)

Conclusion:

The IMRAS system provides an efficient, role-based, and warehouse-aware solution for managing inventory operations across multiple warehouses. It integrates core processes such as putaway, FIFO-based picking, batch and expiry tracking, and real-time dashboards. By enforcing strict access control, the system ensures data accuracy, security, and accountability. The reporting module further supports informed decision-making through actionable insights.