Vendor Management Application: Purpose and Scope

Purpose

The **Vendor Management Application** is designed to streamline and automate the tracking of vendor-related data for a company, ensuring efficient management of employee-vendor associations, purchase orders (POs), client contracts, payment schedules, and exit processes.

The primary goals are:

1. **Centralize Vendor & Employee Data** – Maintain records of employees, their associated vendors, and contract details.
2. **Track Client POs & Billing** – Monitor client contracts, PO rates, and validity periods.
3. **Manage Payment Schedules** – Record payable dates and payment status updates.
4. **Facilitate Exit Processes** – Update employee/vendor exit details and archive records.
5. **Ensure Data Accuracy & Security** – Provide role-based access to prevent unauthorized modifications.

Scope

The application will cover the following stages of vendor management:

Stage 1: Employee & Vendor Onboarding

- **Fields:**
  - Employee Name
  - Date of Joining (DOJ)
  - Vendor Name
  - PO Rate (Agreed rate with the vendor)
  - Skill (Employee’s role/competency)
- **Functionality:**
  - Add, edit, and archive employee-vendor associations.
  - Validate PO rates against vendor contracts.

Stage 2: Client PO Management

- **Fields:**
  - Client PO Number
  - Client PO Start & End Date
  - Client Name
  - Client PO Rate (Billing rate to the client)
- **Functionality:**
  - Track active and expired POs with alerts for renewals.
  - Compare vendor PO rates vs. client PO rates for margin analysis.

Stage 3: Payment Tracking

- **Fields:**
  - Payment Status (Pending/Processed)
  - Payable Dates
  - Invoice References (if applicable)
- **Functionality:**
  - Generate payment reminders for accounts teams.
  - Log payment confirmations.

Stage 4: Exit Management & Data Archival

- **Fields:**
  - Exit Date
  - Reason for Exit (Contract end, termination, etc.)
  - Final Payment Status
- **Functionality:**
  - Archive records while retaining audit trails.
  - Generate reports on vendor/employee turnover.

Out of Scope

- Payroll processing (only payment tracking, not execution).
- Vendor performance reviews (can be added later).
- Tax compliance (handled by external finance systems).

Target Users

- **HR/Admin Teams** – Manage employee-vendor mappings.
- **Finance Teams** – Track POs and payments.
- **Operations Managers** – Monitor contract timelines.
- **Auditors** – Access historical records for compliance.

Key Benefits

✅ **Reduced Manual Errors** – Automated data entry & validation.
✅ **Improved Compliance** – Audit-ready records of all transactions.
✅ **Better Financial Control** – Visibility into vendor vs. client billing rates.
✅ **Scalability** – Supports growing vendor/employee counts.

System Architecture

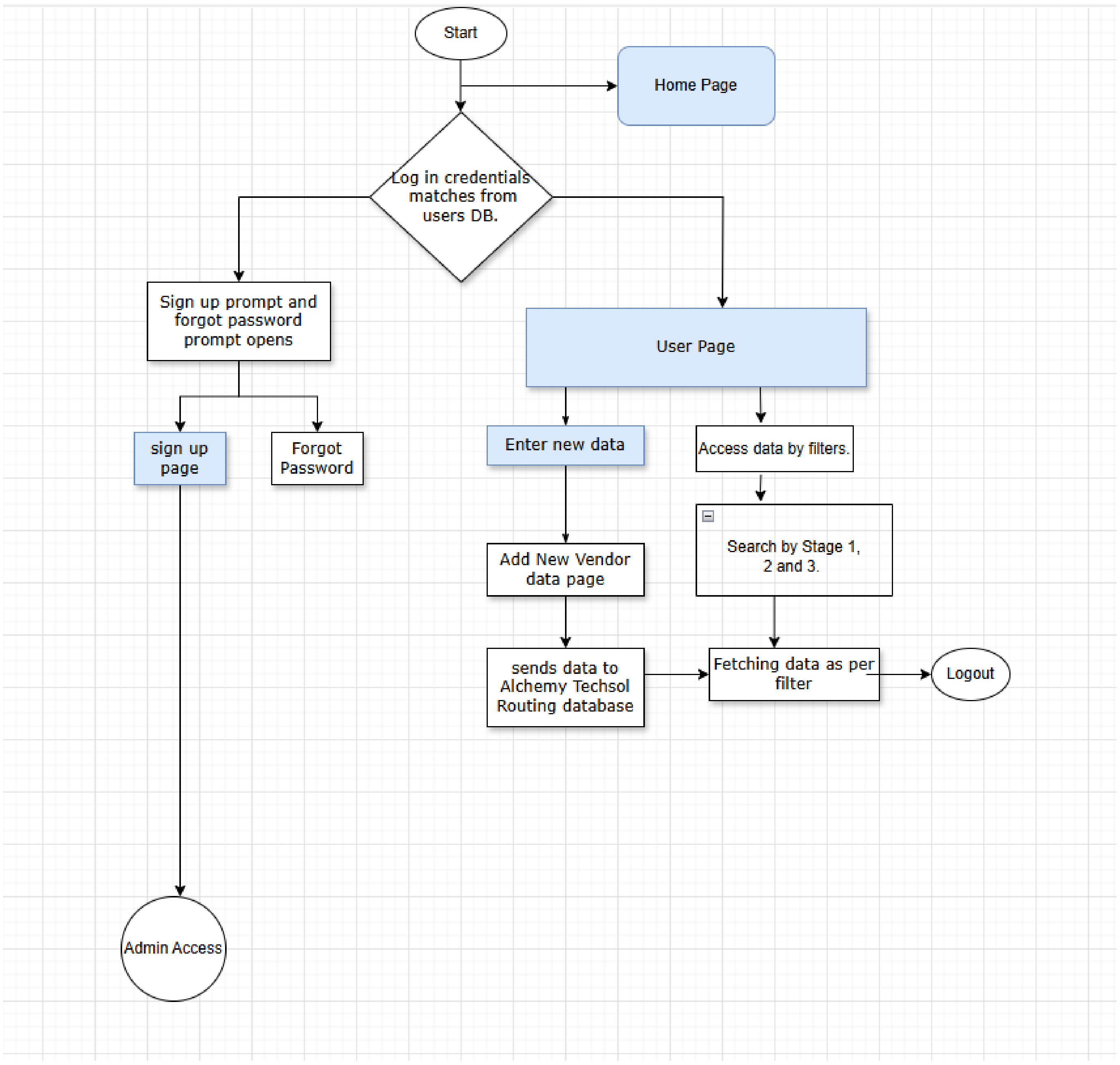
Components Breakdown:

1. **Frontend (Web/Mobile App)**
    - **Tech Stack:** React.js
    - **Features:**
        - Dashboard for HR, Finance, and Operations.
        - Forms for data entry (Employee, Vendor, PO, Payments).
        - Reports & Alerts (e.g., PO expiry reminders).
2. **Backend (Server + APIs)**
    - **Tech Stack:** Node.js (Express) / Python (Django/Flask) / Java (Spring Boot).
    - **Key Modules:**
        - **Auth Service** (Role-based access: Admin, HR, Finance).
        - **Employee-Vendor Manager** (Stage 1).
        - **PO & Client Manager** (Stage 2).
        - **Payment Tracker** (Stage 3).
        - **Exit & Data Archival** (Stage 4).
3. **Database**
    - **Tech Stack:** PostgreSQL
    - **Tables:**
        - Employees (Name, DOJ, Vendor, Skill).
        - Vendors (Vendor Name, PO Rate, Contract Terms).
        - ClientPOs (PO No, Client, Start/End Date, Rate).
        - Payments (Status, Due Date, Invoice Ref).
        - ExitRecords (Exit Date, Reason, Final Payment).

4. **Step 3: Role-Based Access Control (RBAC)**

Role	Permissions
Admin	Full access (CRUD all stages).
HR	Stage 1 (Employee-Vendor mapping).
Finance	Stage 3 (Payments) + Reports.
Operations	Stage 2 (Client POs).

Control Flow Diagram for Vendor Management App.


Start
Home Page
Log in credentials matches from users DB.
Sign up prompt and forgot password prompt opens
User Page
sign up page
Forgot Password
Enter new data
Access data by filters.
Search by Stage 1, 2 and 3.
Add New Vendor data page
sends data to Alchemy Techsol Routing database
Fetching data as per filter
Logout
Admin Access

1. Database Schema (ER Diagram)

Here’s the **normalized database structure** for your app:

Tables Breakdown:

Table	Key Fields	Description
EMPLOYEE	employee_id (PK), name, doj, skill	Employee details.
VENDOR	vendor_id (PK), vendor_name, vendor_po_rate	Vendor contracts.
CLIENT_PO	po_id (PK), client_name, po_number, start/end_date, client_rate, vendor_id (FK)	Client POs linked to vendors.
PAYMENT	payment_id (PK), po_id (FK), status, payable_date	Payment schedules.
EXIT_RECORD	exit_id (PK), employee_id (FK), exit_date, reason	Archived employee/vendor exits.

2. API Endpoint Specifications

RESTful APIs with JWT authentication.

2.1 Employee-Vendor Management (Stage 1)

Endpoint	Method	Description	Request Body Example
/api/employees	POST	Add employee-vendor mapping.	{name: "John", doj: "2024-01-01", vendor_id: "V001", skill: "Dev"}
/api/employees/{id}	GET	Fetch employee details.	-
/api/vendors	POST	Add a new vendor.	{vendor_name: "ABC Corp", vendor_po_rate: 50}

2.2 Client PO Management (Stage 2)

| /api/client-pos | POST | Create a client PO. | {client_name: "XYZ Ltd", po_number: "PO1001", vendor_id: "V001", client_rate: 70} |
| /api/client-pos/{id} | PUT | Update PO end date/rate. | {end_date: "2024-12-31"} |

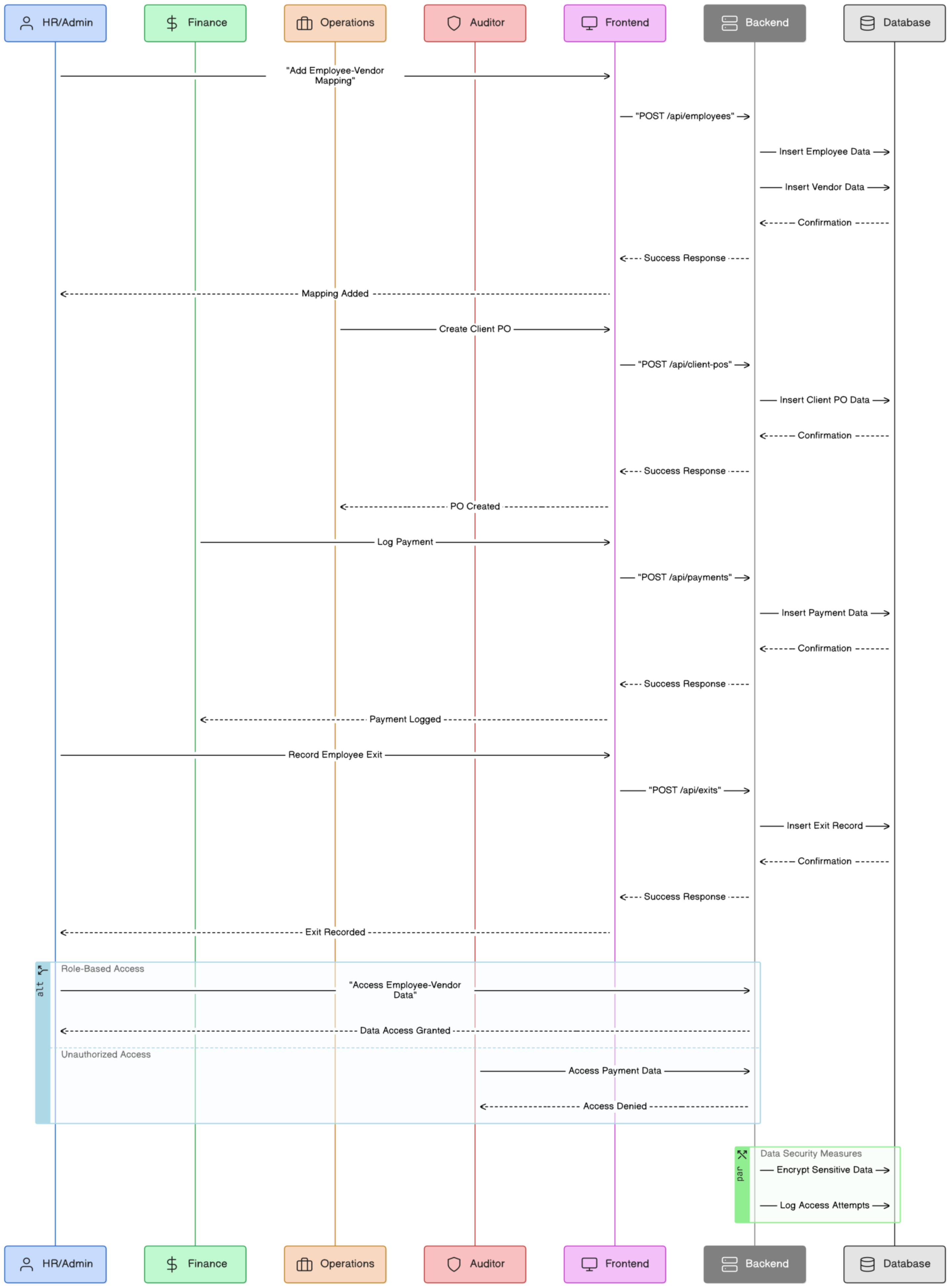
2.3 Payment Tracking (Stage 3)

| /api/payments | POST | Log a payment. | {po_id: "PO1001", status: "Pending", payable_date: "2024-06-15"} |
| /api/payments/overdue | GET | List overdue payments. | - |

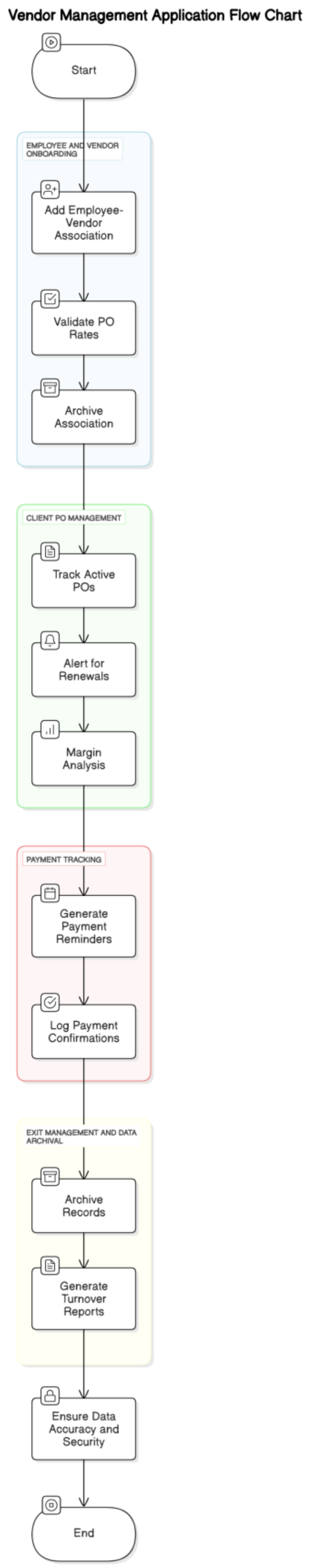
2.4 Exit Management (Stage 4)

| /api/exits | POST | Record an exit. | {employee_id: "E001", exit_date: "2024-05-20", reason: "Contract End"} |

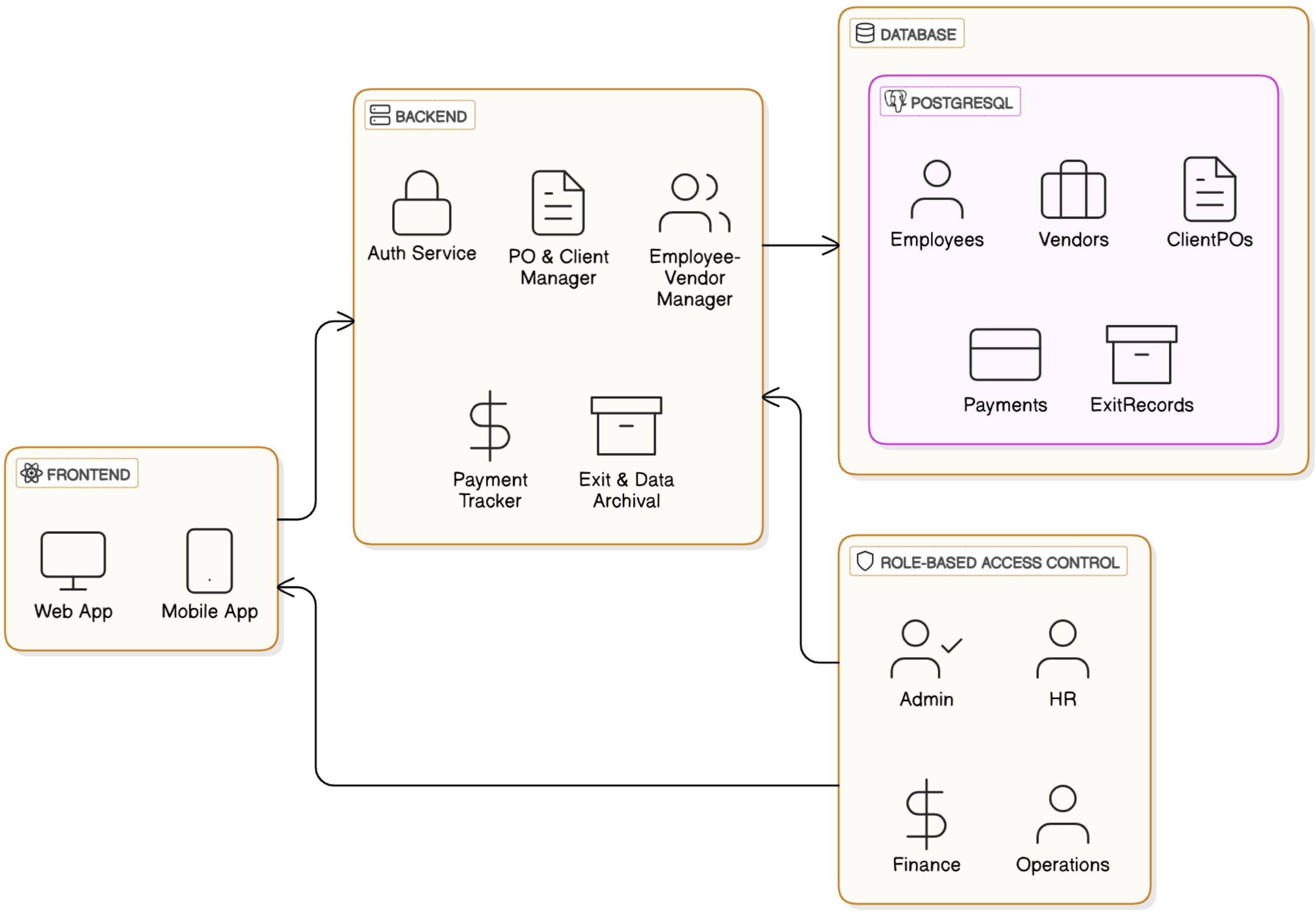
Vendor Management Application Sequence



Data Flow Diagram


Vendor Management Application Flow Chart
Start
EMPLOYEE AND VENDOR ONBOARDING
Add Employee-Vendor Association
Validate PO Rates
Archive Association
CLIENT PO MANAGEMENT
Track Active POs
Alert for Renewals
Margin Analysis
PAYMENT TRACKING
Generate Payment Reminders
Log Payment Confirmations
EXIT MANAGEMENT AND DATA ARCHIVAL
Archive Records
Generate Turnover Reports
Ensure Data Accuracy and Security
End

Vendor Management Application Architecture



DATABASE
POSTGRESQL
Employees
Vendors
ClientPOs
Payments
ExitRecords
BACKEND
Auth Service
PO & Client Manager
Employee-Vendor Manager
Payment Tracker
Exit & Data Archival
FRONTEND
Web App
Mobile App
ROLE-BASED ACCESS CONTROL
Admin
HR
Finance
Operations