Project Design Phase

Solution Architecture

Date	03 November 2025
Team ID	NM2025TMID01909
Project Name	Medical Inventory Management
Maximum Marks	4 Marks

Architecture Overview

The **Medical Inventory Management System** follows a **three-tier architecture** consisting of:

1. **Presentation Layer** – user interfaces for web and mobile devices
2. **Application Layer** – core logic, APIs, and business rules
3. **Data Layer** – centralized, secure database for all medical inventory data

System Components & Flow

A. User Layer (Frontend / Clients)

- **Users:** Admin, Pharmacist, Nurse, Supplier
- **Access via:** Web browser or mobile app
- **Functions:**
    - View and manage stock
    - Scan items via barcode/RFID
    - Generate purchase requests
    - Receive alerts and reports

Technologies:
React.js / Angular (for web)
Flutter / React Native (for mobile)

B. Application Layer (Backend & API Services)

- Handles all business logic, validations, and data transactions.
- Ensures smooth coordination between users and the database.
- Provides RESTful APIs for web and mobile access.

Key Modules:

1. **Inventory Management Service** – Add/update/delete items, manage quantities.
2. **Alert & Notification Service** – Triggers low-stock and expiry alerts (via email/SMS).
3. **Purchase Order Service** – Generates and tracks orders.
4. **Supplier Management Service** – Stores supplier data and purchase history.
5. **Authentication Service** – Role-based access using JWT/OAuth 2.0.
6. **Audit & Reporting Service** – Tracks all activity and generates analytics reports.

Technologies:
Node.js / Django / Spring Boot
Express.js / REST API Frameworks

C. Data Layer (Database & Storage)

- Stores all records securely in a relational database.
- Maintains tables for users, items, suppliers, transactions, and logs.

Database: MySQL / PostgreSQL
Data Backup: Cloud backup via AWS RDS or Azure SQL

Integration Components

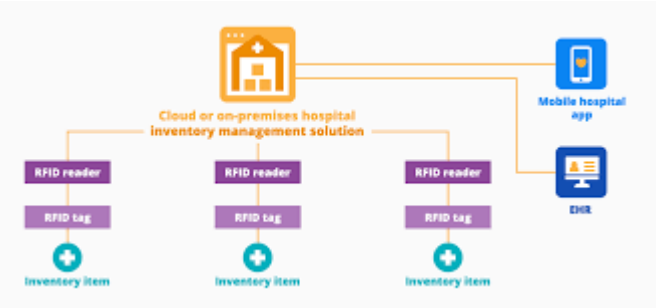
Integration	**Purpose**
Barcode/RFID Module	Fast, automated item scanning and entry
Notification API	Sends alerts via email/SMS
Analytics Engine	Generates usage, wastage, and financial insights
Cloud / On-Prem Deployment	Flexible hosting based on hospital preference

Security Architecture

- **Role-Based Access Control (RBAC):** Restricts functions based on user roles (Admin, Pharmacist, etc.)
- **Data Encryption:** Sensitive data encrypted in-transit (HTTPS) and at-rest (AES).
- **Authentication:** JWT / OAuth 2.0-based secure login.
- **Audit Logging:** Tracks all modifications for compliance.

Data Flow Summary

1. **User logs in** → Authentication Service validates credentials.
2. **User requests inventory data** → API fetches records from Database.
3. **Stock updates or issues** → Inventory Module updates database in real time.
4. **Alerts generated** → Notification Module sends warnings to relevant staff.
5. **Reports generated** → Analytics Module compiles data into dashboards.



Advantages of This Architecture

- Modular and scalable for future expansion (multi-hospital support).
- Ensures high reliability and security.
- Enables real-time tracking and automation.
- Provides flexible integration with existing hospital system