Project Design Phase

Proposed Solution

Date	**01 NOV 2025**
Team ID	**NM2025TMID02942**
Title	**Medical Inventory System**
Maximum Marks	**2 Marks**

1.Overview

The proposed solution is to design and implement a Medical Inventory Management System (MIMS) — a centralized, automated, and intelligent platform that enables healthcare facilities to efficiently manage their medical supplies, pharmaceuticals, and equipment.

The system aims to eliminate manual inventory tracking, prevent stock shortages and wastage, and provide real-time visibility and analytics to support data-driven decision-making.

---

2. Solution Objectives

- Maintain accurate, real-time records of all medical supplies.
- Minimize stockouts, overstocking, and expiry-related losses.
- Improve communication between departments and suppliers.
- Enhance reporting and compliance with healthcare regulations.
- Reduce manual workload and human error through automation.

---

3. System Description

The Medical Inventory System will be a web-based (and optionally mobile-enabled) application that connects all relevant departments — pharmacy, procurement, wards, and administration — through a shared digital platform.

It will manage the entire lifecycle of inventory items, from procurement and storage to usage and disposal.

---

4. Key Features

Feature	Description	Expected Benefit
Real-Time Stock Monitoring	Automatically updates inventory as items are added, used, or removed	Reduces manual entry errors and provides up-to-date stock levels
Automated Reorder Alerts	System triggers notifications when stock falls below minimum threshold	Prevents shortages of critical medicines
Expiry Tracking and Alerts	Monitors batch expiry dates and alerts users in advance	Minimizes wastage from expired drugs
Supplier Management Module	Stores vendor details, purchase history, and contact information	Simplifies and accelerates procurement processes
Analytics & Reporting Dashboard	Displays data on stock usage, trends, and performance metrics	Supports better decision-making and forecasting
Role-Based Access Control	Grants permissions based on user roles (Admin, Pharmacist, Nurse, etc.)	Ensures data security and accountability
Audit Trail and Compliance Reports	Logs every transaction with user and timestamp	Helps meet regulatory and audit requirements
Multi-Department Access	Enables inventory sharing and coordination across units	Improves efficiency and reduces duplication of orders

---

5. System Architecture (Conceptual Design)

Architecture Layers:

1. Presentation Layer (Frontend):
   - User-friendly web interface (HTML, CSS, JavaScript, or React).
   - Mobile accessibility for on-floor updates.
2. Application Layer (Backend):
   - Handles business logic, stock management, and alert systems.
   - Developed using Python (Django/Flask), Java, or Node.js.
3. Database Layer:
   - Centralized database (e.g., MySQL, PostgreSQL, or MongoDB).
   - Stores product details, supplier info, transactions, and audit logs.
4. Integration Layer:
   - APIs to connect with external systems (e.g., Hospital Management System, supplier portals).
   - Secure authentication (OAuth, JWT).

(Optional: You can include a simple block diagram showing these layers in your report.)

---

6. Technology Stack (Example)

Component	Technology Used
Frontend	ReactJS / Angular / HTML5 / CSS3
Backend	Python (Django/Flask) or Node.js
Database	MySQL / PostgreSQL
Authentication	JWT / OAuth 2.0
Hosting	Cloud-based (AWS, Azure, or Firebase)
Testing Tools	Selenium, Postman, JMeter

---

7. Process Flow (High-Level)

1. Login: User authentication based on role (Admin, Pharmacist, Nurse).
2. Inventory Management: Add, update, or remove items in the system.
3. Automatic Stock Update: Quantities adjust when items are issued or received.
4. Alerts & Notifications: Low stock, near-expiry, or pending purchase order alerts triggered.
5. Procurement Integration: Generate and send purchase requests to suppliers.
6. Reporting: Generate audit, usage, and financial reports for management.

---

8. Expected Outcomes

- 80–90% reduction in manual data entry errors.
- Real-time, 24/7 visibility into inventory levels.
- Significant cost savings through optimized procurement.
- Reduced wastage of expired materials.
- Faster audit preparation and improved compliance.
- Increased staff productivity and satisfaction.

---

9. Implementation Plan (Summary)

Phase	Description	Duration
Phase 1: Requirement Analysis	Identify user needs, define system requirements	2 weeks
Phase 2: System Design	Create architecture, data flow diagrams, UI mockups	3 weeks
Phase 3: Development	Build core modules and integrate APIs	5 weeks

Phase	Description	Duration
Phase 4: Testing	Conduct unit, integration, and user testing	2 weeks
Phase 5: Deployment & Training	Deploy system and train hospital staff	2 weeks
Phase 6: Maintenance	Ongoing updates and support	Continuous

---

10. Risk Management

Risk	Impact	Mitigation Strategy
User resistance to change	Medium	Provide training and user manuals
System downtime	High	Implement cloud backup and redundancy
Data security threats	High	Use encryption and role-based access control
Integration complexity	Medium	Use standardized APIs and test integrations early