



Premier University

Chattogram

Software Development Project Proposal

Hospital Management System

Submitted by

Name	ID
Shadman Al Muksit	0222220005101003
MD Nishadul Islam Chy Shezan	0222220005101014
Sariful Islam	0222220005101023
Rimjhim Dey	0222220005101039

Section: A Batch: 42 Session: Spring 2025

Supervisor

Dhrubajyoti Das
Assistant Professor,
Dept. of Computer Science and Engineering

July 2025

Contents

1	Project Overview	2
1.1	Objective	2
1.2	Problem Statement	2
1.3	Scope	2
1.4	Technologies	2
2	Related Projects	3
3	Tentative Features	3
4	System Design	3
4.1	Requirement Gathering	3
4.2	Feasibility Analysis	4
4.3	System Architecture	4
4.4	Data Flow	4
5	Project Timeline (Gantt Chart)	5
6	Deliverables	5

1 Project Overview

1.1 Objective

Develop a robust, secure, and user-friendly Hospital Management System (HMS) to streamline patient care, resource allocation, staff management, and billing. The system will operate strictly within the hospital premises, with all access limited to authorized staff.

1.2 Problem Statement

Many hospitals struggle with manual patient tracking, inefficient bed and ambulance allocation, and inconsistent record-keeping. There is an urgent need for a secure, digital platform to centralize operations, reduce manual errors, and enforce clear staff responsibilities via strict role-based access.

1.3 Scope

The HMS will support:

1. Patient registration, appointment booking, and history lookup (inpatient & outpatient)
2. Bed and ward management with cleaning/maintenance workflow
3. Ambulance dispatch tracking and management
4. Staff management with strict role-based permissions
5. Invoicing at booking, discharge, and during treatment
6. Essential inventory tracking
7. API-based notifications for key workflow events
8. Batch operations (bulk admission/discharge)
9. Incident reporting
10. Audit trail for all actions

Patient self-access is not supported. The system will be deployed locally using PHP (Laravel) and MySQL, ensuring full data control.

1.4 Technologies

1. **Backend:** Laravel (PHP)
2. **Database:** MySQL
3. **Frontend:** Blade Templates (Laravel), optional Bootstrap
4. **Deployment:** Local server environment (e.g., XAMPP/LAMP)
5. **Tools:** Git for version control

2 Related Projects

1. **OpenMRS:** An open-source hospital system focusing on modular design and role-based security. Our project will offer a more streamlined, staff-centric solution with local deployment and simple inventory/incident modules.
2. **MedKey:** Commercial hospital management suite. Our solution will provide similar core functionality for smaller hospitals, with a focus on simplicity and easy local deployment.

3 Tentative Features

1. User Authentication: Secure login, role assignment (Reception, Doctor, Nurse, Ward Manager, Ambulance Staff, User Admin)
2. Patient Management: Registration, history, outpatient/inpatient flows
3. Appointment & Bed Management: Bookings, requests, assignments, post-discharge cleaning tracking
4. Ambulance Management: Status, dispatch workflow, logs
5. Billing & Invoicing: Generation at all stages, multiple payment methods
6. Inventory Tracking: For wards and ambulances
7. Notifications: API-based alerts for workflow events
8. Batch Operations: Bulk admission/discharge (CSV)
9. Incident Reporting: Staff logs, admin review
10. Dashboards: Real-time insights for key resources
11. Audit Trail: Action logs for compliance

4 System Design

4.1 Requirement Gathering

1. Informal consultation with medical professionals (two practicing doctors) to understand real-world pain points and workflow priorities in hospital settings
2. Analysis of workflows and features in hospital management systems currently used by local hospitals (through observation and online research)
3. Comparative study of publicly available hospital management platforms and documentation found on the internet
4. Identification and synthesis of key functional requirements: strict role-based permissions, bed/ambulance workflows, batch operations, and automated notifications
5. Design of a customized system model based on observed best practices, medical feedback, and unique team insights

4.2 Feasibility Analysis

Technical Feasibility: The stack (Laravel, PHP, MySQL) is robust, widely used in healthcare, and well-supported for local deployment.

Economic Feasibility: Open-source stack ensures low cost.

Operational Feasibility: Role-based UI and dashboards ensure quick staff adoption.

Schedule Feasibility: The project plan fits within a standard academic semester.

4.3 System Architecture

1. **Client Layer:** Browser-based, responsive UI for desktops/tablets using Laravel Blade
2. **Server Layer:** Laravel handles business logic, workflow, security, notifications
3. **Data Layer:** MySQL manages all persistent data, with backups enabled

4.4 Data Flow

Staff actions such as patient admission, bed assignment, and billing are processed through a browser interface. The backend system (Laravel) enforces role-based workflows, updates the MySQL database, and triggers notifications. Dashboards and reports are generated in real-time from the database. The following diagrams clarify these concepts.

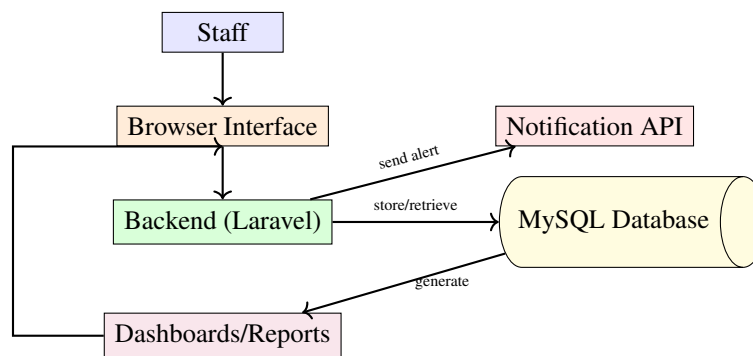


Figure 1: Component-level workflow: Staff use the browser interface, backend processes actions, stores data, sends notifications, and generates dashboards/reports for users.

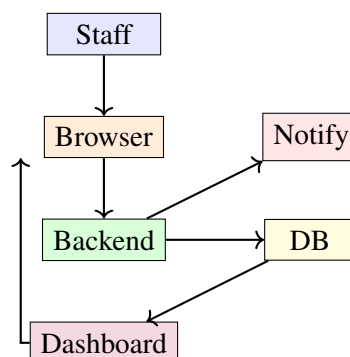


Figure 2: End-to-end data flow: Staff actions move through browser and backend, resulting in database updates, notification delivery, and real-time dashboard/report refresh for efficient hospital management.

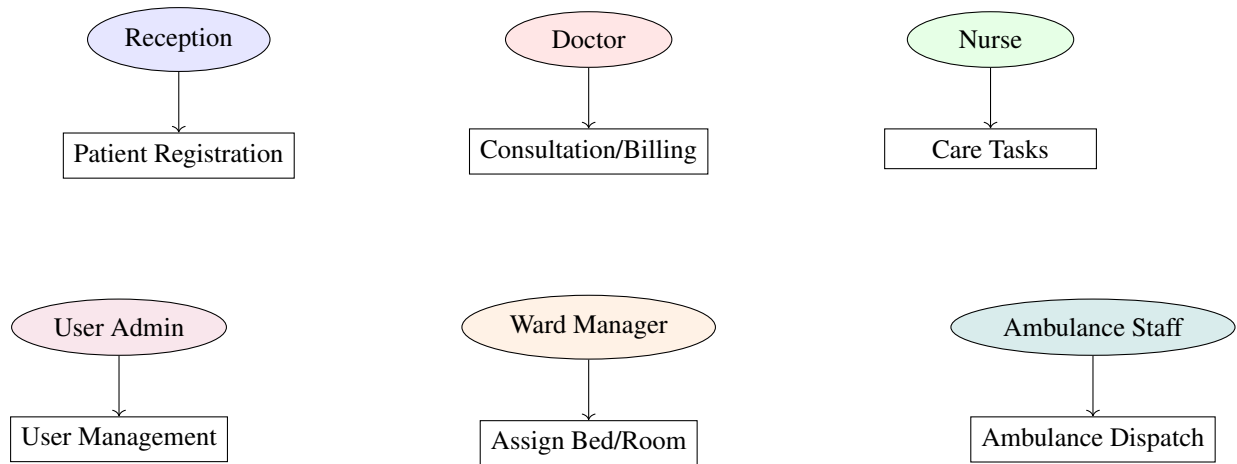


Figure 3: Key role-based use cases (arranged for readability): Reception, Doctor, Nurse, Ward Manager, User Admin, and Ambulance Staff, each linked to their core function.

5 Project Timeline (Gantt Chart)

Task	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12
Req. Analysis	X	X										
R & D	X	X										
Proposal Drafting	X	X	X									
System Design			X	X								
Frontend Dev				X	X	X						
Backend Dev				X	X	X						
Database Setup					X	X						
Integration						X	X	X				
Testing/Feedback								X	X			
Final Adjustments									X	X	X	
Proj. Submission											X	X

6 Deliverables

1. **Week 4:** Project proposal document (this document)
2. **Week 8:** MVP with core modules (auth, patient registration, bed management, invoicing)
3. **Week 12:** Final system with all features, user manual, and documentation