

Hospital Management System (HMS): Full Project Documentation

Software Requirements Specification (SRS)

University Bangladesh

Shuvo Mozumder
Student ID: XXX

Alamin Sarker
Student ID: XXX

Jibon Ray
Student ID: XXX

MD. Shakil
Student ID: XXX

Naimur Rahman
Student ID: XXX

Tarin Mustari
Student ID: XXX

MD. Mahedi Ahsan
Student ID: XXX

Abstract—This document presents the Software Requirements Specification (SRS) for a Hospital Management System (HMS) designed to streamline hospital operations in Bangladesh. The system aims to automate administrative processes, manage patient records, facilitate appointments, and provide real-time reporting. Inspired by leading Bangladeshi hospitals such as Ibn Sina, Labaid, and Bardem, the system incorporates features for comprehensive healthcare management including pharmacy, laboratory, and billing modules.

Index Terms—Hospital Management System, SRS, Bangladesh Healthcare, Patient Management, Medical Records

I. INTRODUCTION

A. Purpose

The purpose of this Software Requirements Specification (SRS) document is to outline the requirements for developing a comprehensive Hospital Management System (HMS) suitable for Bangladeshi healthcare facilities. The system will digitize hospital operations, improve patient care, and enhance administrative efficiency.

B. Scope

The HMS covers core hospital services including patient registration, doctor scheduling, appointment booking, billing, medical records, pharmacy management, laboratory services, and reporting. The system will be web-based with role-based access control for various user types.

C. Definitions, Acronyms, and Abbreviations

- HMS: Hospital Management System
- SRS: Software Requirements Specification
- ER: Entity-Relationship
- DFD: Data Flow Diagram
- UML: Unified Modeling Language

D. References

- 1) IEEE Standard for Software Requirements Specifications (IEEE 830-1998)
- 2) Bangladesh Medical Association Guidelines
- 3) Hospital Information System standards

E. Overview

The document is organized as follows: Section II covers system overview and scope, Section III describes user roles, Section IV details functional requirements, Section V covers non-functional requirements, Section VI presents system design, and Section VII outlines implementation plan.

II. PROJECT OVERVIEW

A. Project Title

Hospital Management System (HMS)

B. Objective

To design and develop a computerized system to manage hospital operations efficiently, inspired by Bangladesh's leading hospitals like Ibn Sina, Labaid, and Bardem.

C. Bangladesh Healthcare Context

Bangladesh's healthcare sector has seen significant growth with modern facilities like:

- Ibn Sina Hospital: Multi-specialty hospital offering cardiology, oncology, and emergency services
- Labaid Specialized Hospital: Known for cardiac care and cancer treatment
- Bangladesh Institute of Research and Rehabilitation in Diabetes, Endocrine and Metabolic Disorders (BIRDEM): Specialized in diabetes management

These institutions provide models for comprehensive healthcare management, which this HMS aims to emulate through digital solutions.

D. Goals

- Automate hospital administrative processes
- Maintain accurate patient and doctor records
- Improve patient care through digital data access
- Provide real-time reports for management decisions
- Ensure data security and privacy compliance

III. SYSTEM SCOPE

The system covers the following core hospital services:

- Patient management services
- Doctor and staff management services
- Appointment and scheduling system services
- Medical records & prescriptions services
- Billing & payment system services
- Pharmacy and inventory management services
- Laboratory management services
- Ward/room management services
- Reporting & analytics services

IV. SYSTEM USERS AND ROLES

TABLE I
SYSTEM USERS AND ROLES

User Role	Description	Example Activities
Admin	Manages the entire system	Manage users, set permissions, Add/Update logs
Doctor	Provides healthcare services	View appointments, update prescriptions, access patient history
Nurse	Assists doctor and handles patient care	Update patient vitals, monitor room status
Receptionist	Handles front-desk operations	Register patients, schedule appointments
Pharmacist	Manages medicines and stock	Issue drugs, update inventory
Accountant	Handles billing and payments	Generate invoices, Back-end
Patient	Receives services	Book appointments, view prescriptions, pay bills

V. SYSTEM MODULES

A. A. User Management Module

- User registration and login
- Role & permission control
- Password reset & profile management

B. B. Patient Management Module

- Register new patients
- Maintain patient personal details and medical history
- Search/view patient records
- Assign doctors

C. C. Appointment Management Module

- Schedule, reschedule, or cancel appointments
- Doctor-wise daily schedule view
- Notifications/reminders

D. D. Pharmacy Management Module

- Add/update medicines
- Track medicine stock and expiry dates
- Issue medicines for prescriptions
- Generate inventory reports

E. E. Billing & Payment Module

- Generate invoices
- Manage payment methods
- View billing history

F. F. Laboratory Management Module

- Record lab tests and results
- Assign lab tests to patients
- Generate lab test reports

G. G. Ward/Room Management Module

- Track room and bed availability
- Assign beds to admitted patients
- Manage discharge and room transfer

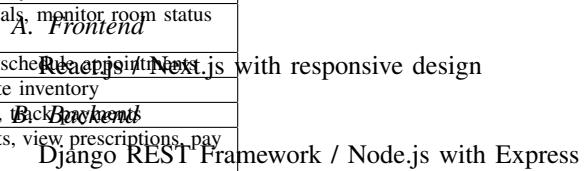
H. H. Reporting Module

- Daily/Monthly reports
- Doctor-wise patient count
- Revenue and expense analysis

I. I. Authentication & Authorization Module

- Secure login via JWT
- Role-based access control

VI. SYSTEM ARCHITECTURE



C. Database

PostgreSQL / MySQL with data encryption

D. API

RESTful API with GraphQL support

E. Hosting

- Frontend: Vercel / Netlify
- Backend: DigitalOcean / AWS
- Database: Managed cloud database

VII. SYSTEM ANALYSIS

A. Functional Requirements

- 1) *FR1: Patient Registration:* The system shall allow patients to register online with personal and medical information.
- 2) *FR2: Appointment Booking:* The system shall enable patients to book appointments with doctors based on availability.
- 3) *FR3: Medical Record Management:* The system shall store and retrieve patient medical records securely.
- 4) *FR4: Billing System:* The system shall generate bills and receipts for services rendered.
- 5) *FR5: User Access Control:* The system shall control user access by roles and permissions.
- 6) *FR6: Reporting:* The system shall generate various reports for management analysis.

B. Non-Functional Requirements

1) Performance:

- Page load time: ≤ 2 seconds
- Concurrent users: 100+

2) *Security*:

- Password encryption
- JWT authentication
- Data privacy compliance (HIPAA-like standards)

3) *Scalability*: The system shall handle increasing user load and data volume.

4) *Reliability*: 99.9% uptime with backup and recovery mechanisms.

5) *Usability*: Intuitive interface for all user roles.

6) *Compatibility*: Cross-browser support and mobile responsiveness.

VIII. DATA MODELING

A. Entity-Relationship Diagram

The ER diagram includes the following main entities:

- User (user_id, name, email, role, password)
- Patient (patient_id, name, age, gender, contact, address, medical_history)
- Doctor (doctor_id, name, specialization, contact, schedule)
- Appointment (appointment_id, patient_id, doctor_id, date, time, status)
- Prescription (prescription_id, patient_id, doctor_id, date, medicine_list, note)
- Medicine (medicine_id, name, stock, expiry_date, price)
- Bill (bill_id, patient_id, date, total_amount, payment_status)
- LabTest (test_id, patient_id, test_name, result, date)
- Room (room_id, type, status, assigned_patient)

B. Relationships

- User 1:N Patient (one doctor/nurse can manage multiple patients)
- Patient 1:N Appointment
- Doctor 1:N Appointment
- Patient 1:N Prescription
- Doctor 1:N Prescription
- Prescription N:N Medicine
- Patient 1:N Bill
- Patient 1:N LabTest
- Room 1:1 Patient (admitted)

IX. SYSTEM DESIGN

A. UML Diagrams

1) *Use Case Diagram*: The use case diagram shows interactions between actors (Admin, Doctor, Patient, etc.) and the system.

2) *Class Diagram*: Defines entities, attributes, and relationships.

3) *Activity Diagram*: Represents workflows such as patient registration and appointment booking.

4) *Sequence Diagram*: Shows step-by-step interactions for processes like patient admission.

5) *DFD (Data Flow Diagram)*: Level 0, 1, and 2 diagrams showing data flow through the system.

6) *ER Diagram*: Detailed database structure with cardinalities.

B. Database Schema

Tables with primary keys, foreign keys, and constraints as per ER model.

X. IMPLEMENTATION PLAN

A. Phase 1: Requirement Analysis

- SRS Document completion
- Stakeholder interviews
- Feature prioritization

B. Phase 2: System Design

- UML diagrams creation
- Database schema design
- UI/UX wireframes

C. Phase 3: Development

- Frontend development
- Backend API development
- Database implementation
- Module integration

D. Phase 4: Testing

- Unit testing
- Integration testing
- System testing
- User acceptance testing

E. Phase 5: Deployment

- Production environment setup
- Data migration
- Go-live preparation

F. Phase 6: Maintenance

- Bug fixes
- Feature enhancements
- Performance monitoring

XI. TOOLS AND TECHNOLOGIES

A. Programming

- Python (Django REST Framework)
- JavaScript (React.js/Next.js)

B. Database

PostgreSQL with backup and replication.

C. Modeling

- Draw.io for diagrams
- PlantUML for UML diagrams

D. Testing

- Postman for API testing
- Jest for frontend testing
- Pytest for backend testing

E. Version Control

Git with GitHub for collaboration.

F. Deployment

- Vercel for frontend
- DigitalOcean for backend
- Docker for containerization

XII. DELIVERABLES

- SRS Document (this document)
- System Design Diagrams (Use Case, ER, DFD)
- Database Schema
- Prototype Screens (Figma mockups)
- Source Code Repository
- Test Cases and Results
- User Manual
- Deployment Guide

XIII. CONCLUSION

This SRS provides a comprehensive foundation for developing a Hospital Management System tailored for Bangladesh's healthcare needs. The system incorporates best practices from leading local hospitals while ensuring compliance with international standards for healthcare software. The modular design allows for phased implementation and future scalability.