

---

# **Software Requirements Specification for <Hospital Management System>**

**BANGLADESH UNIVERSITY OF PROFESSIONALS**

**FACULTY OF SCIENCE & TECHNOLOGY**

**DEPT. OF COMPUTER SCIENCE AND ENGINEERING (CSE)**

**<Team Members:**

**Ayesa Begum (2252421043)**

**Shanjida Afrin (2252421087)**

**Ishfar Binte haque (2252421107)**

**Labiba Alam (2252421111)**

**Nusrat Jahan (2252421121)>**

**Date of Submission: 07/05/2025**

# Table of Contents

<b>Table of Contents</b>	<b>ii</b>
<b>1. Introduction</b>	<b>1</b>
1.1 Purpose	1
1.2 Document Conventions	1
1.3 Intended Audience and Reading Suggestions	1
1.4 Product Scope	2
1.5 References	2
<b>2. Overall Description</b>	<b>3</b>
2.1 Product Perspective	3
2.2 Product Functions	3
2.3 User Classes and Characteristics	3
2.4 Operating Environment	4
2.5 Design and Implementation Constraints	4
2.6 User Documentation	4
2.7 Assumptions and Dependencies	4
<b>3. External Interface Requirements</b>	<b>5</b>
3.1 User Interfaces	5
3.2 Hardware Interfaces	5
3.3 Software Interfaces	6
3.4 Communications Interfaces	6
<b>4. System Features</b>	<b>6</b>
4.1 Register	6
4.2 Login	7
4.3 Doctor Scheduling and Management	7
4.4 Appointment Booking	7
4.5 Medical Records Management	8
4.6 Room and Bed Allocation Management	8
4.7 Emergency Case Handling	8
4.8 Feedback and Complaint Management	9
4.9 Filtered Searching Process	9
4.10 Health History and Reports Access (for patients)	9
<b>5. Other Nonfunctional Requirements</b>	<b>9</b>
5.1 Performance Requirements	9
5.2 Safety Requirements	10
5.3 Security Requirements	10
5.4 Software Quality Attributes	10
5.5 Business Rules	11

# 1. Introduction

## 1.1 Purpose

*This document describes the software requirements for the Hospital Management System (HMS), a stand-alone program designed to efficiently manage hospital operations. This includes medical record processing, patient registration, doctor and appointment scheduling, and emergency care services. The latest version of the SRS covers all aspects of the system for both web and mobile platforms. [1][2]*

## 1.2 Document Conventions

- *Bold font is used to highlight important information, such as user roles and necessary modules.*
- *The features of the system are enumerated in bullet points (•).*
- *Its numbering is based on the **IEEE SRS standard** (e.g., 2.1, 2.2).[1]*
- *Technical terms are used consistently to avoid ambiguity.*

## 1.3 Intended Audience and Reading Suggestions

*This document is intended for the following audiences:*

**Developers:** *To understand and carefully plan the system's essential elements.*

**Project managers:** *are in charge of monitoring the project's progress and ensuring that all specifications are fulfilled.*

**Testers :** *are in charge of developing and executing appropriate test cases that comply with the specifications of the system.*

**End users:** *(for example, physicians and patients) Find out more about the system's expected features and capabilities.*

**Documentation Authors:** *To provide user guides and training materials in accordance with the specifications of the system.*

*It is advised that you read Parts 1 (Introduction) and 2 (Overall Description) before diving into the functional and non-functional demands in the following sections.[1]*

## 1.4 Product Scope

*The Hospital Management System (HMS) was created to automate and streamline vital hospital operations, including scheduling appointments and doctors, storing medical records, responding to emergencies, and tracking hospital data. Its main objectives are to:*

- *Get rid of paper-based manual processes.*
- *Improve departmental communication*
- *Boost hospital operations' overall effectiveness*
- *Minimize the possibility of human mistake*

*Because it is made to work on both mobile and internet platforms, this technology is crucial to the digital revolution in contemporary healthcare.[2]*

## 1.5 References

[1] IEEE, IEEE Recommended Practice for Software Requirements Specifications, IEEE Std 830-1998, Oct. 1998. [Online]. Available: <https://ieeexplore.ieee.org/document/720574>

[2] OpenMRS, "OpenMRS - Open Source Medical Record System," [Online]. Available: <https://openmrs.org>. [Accessed: May 5, 2025].

[3] MySQL, "MySQL 8.0 Reference Manual," Oracle, 2024. [Online]. Available: <https://dev.mysql.com/doc/refman/8.0/en/>

[4] Google Wallet, "Google Wallet API Overview," Google Developers, 2024. [Online]. Available: <https://developers.google.com/wallet>

[5] Dropbox, "Dropbox for Developers," Dropbox Inc., 2024. [Online]. Available: <https://www.dropbox.com/developers>

[6] S. B. Choi and H. J. Kim, "Enhancing system availability with cloud computing for healthcare applications," *IEEE Transactions on Cloud Computing*, vol. 9, no. 1, pp. 154-161, Jan.-Mar. 2021. doi: 10.1109/TCC.2020.2974630

## 2. Overall Description

### 2.1 Product Perspective

*The Hospital Management System (HMS) is a self-contained software application developed to streamline and automate key hospital operations such as patient data management, appointment scheduling, billing, and inventory control. It is designed to replace traditional manual processes, improve efficiency, reduce human error, and ensure better coordination among hospital departments.*

### 2.2 Product Functions

*The major functions of the HMS include:*

- Register
- Login
- Doctor Scheduling and Management
- *Appointment booking*
- *Medical records management*
- *Room and bed allocation management*
- *Emergency care handling (form submission)*
- *Feedback and complaint management*
- *Filtered searching process*
- *Health history and reports access (for patients)*

### 2.3 User Classes and Characteristics

**Doctors:** *Can view and update patient records, add prescriptions, view schedules, create available slots, cancel appointments. Moderate technical skills.*

**Patients:** *Can access appointment booking and medical history, cancel already booked appointments* General users, minimal technical knowledge.

## 2.4 Operating Environment

**Hardware:** Desktop PCs (minimum Core i5, 8GB RAM), tablets, smartphones.

**Operating Systems:** Windows 10/11, Android/iOS for mobile apps.

**Browser Support:** Chrome, Firefox, Edge (latest versions).[3]

**Database:** MySQL .[4]

## 2.5 Design and Implementation Constraints

There will be two major constraints in developing this product:

1. Making real-time updates to the database and allowing access at the same time.
2. Number of users accessing the online database at the same time might affect database updation speed.

## 2.6 User Documentation

The following user documentation components will be delivered along with the Hospital Management System (HMS)

- **User Manual:** A comprehensive guide detailing how to use each module, including registration, appointment booking, medical record access, and more.
- **Quick Start Guide:** A brief, user-friendly guide for new users to get started quickly with the main functions.
- **Online Help System:** Context-sensitive help available within the application interface to guide users through features as they use them.

## 2.7 Assumptions and Dependencies

- *It is assumed that the hospital infrastructure includes reliable internet connectivity to support real-time updates and cloud database access.*
- *The system assumes that external components like **SQL databases** work without critical bugs/[3][4][5]*

- *Any change in regulatory policies (e.g., patient data protection laws) might affect system requirements and compliance modules.*

### 3. External Interface Requirements

#### 3.1 User Interfaces

*The user-friendly graphical interface of the Hospital Management System will make it easy for all users (patients, doctors, admin staff) and everyone: standard iconography, appropriately named navigation buttons (Home, Doctor, Patient, Appointments etc. ) as well as tooltips that make it easier for the user for guidance.*

*All screens have other common interface elements (the logo header, search bars and a easy to use navigation sidebar); standard buttons such as Submit, Cancel, Back, Help and everything else that's provided on the screen are found consistently on related screens. Error messages are displayed in pop-up alerts and the directions provided are clearly described. For over time, the font styles and colors are chosen to be visually appealing and easy to read.*

*Some sample user interface components include:*

- *Login screen with username and password fields*
- *Dashboard displaying daily reports and system stats*
- *Patient Record Form with dropdowns, checkboxes, and calendar input*
- *Appointment Booking Page with calendar view and doctor selection*
- *Billing Page showing charges and payment summary in tabular form*

*It 's responsive in that it will fit on screen sizes different from your monitor and will be usable on your desktops / laptops or tablets.*

#### 3.2 Hardware Interfaces

*This system is intended to run on general-purpose computing devices such as:*

- **Operating System:** *Windows 10 or Windows 11*
- **Processor:** *Intel Core i3 or above (1.7 GHz minimum speed recommended)*
- **RAM:** *Minimum 2 GB (4 GB or more recommended for smooth performance)*
- **Storage:** *At least 500 MB of free disk space for application and local database files*
- **Input Devices:** *Keyboard and mouse/touchpad*
- **Display:** *Any standard monitor with 1024x768 resolution or higher*

*No special hardware components are required beyond common office equipment.*

### 3.3 Software Interfaces

*The system is developed using widely adopted web technologies and tools:*

**Languages Used:** HTML, CSS, JavaScript

**Database:** MySQL

**Server Environment:** XAMPP

**IDE:** Visual Studio Code

**Platform Compatibility:** Windows 10/11

*The application interacts with the MySQL database to store and retrieve information such as doctor details, patient details, login, register, appointment records and reports. SQL queries are used to perform CRUD (Create, Read, Update, Delete) operations.*

*Although it's a standalone web-based application hosted on a local server via XAMPP, it can be extended in the future to connect with cloud services or APIs (e.g., for SMS/email notifications, online payments, or third-party lab results).*

### 3.4 Communications Interfaces

*Since the system is running on a local server set-up (using XAMPP), communication is almost exclusively on the local machine or network. It is compatible with standard internet protocols such as:*

**HTTP/HTTPS:** For accessing the application via a browser

**FTP** (optional: backups / updates): Uploading files to / from remote servers (if required in future updates)

*If it is extended to online use, the use of secure communication ( HTTPS and encrypted connections) will be recommended in data protection perspective (no browser-specific dependencies — the system is tested on common browsers such as Chrome and Firefox.*

## 4. System Features

*This chapter provides an overview of the core functionalities of the Hospital Management System, detailed here. Within every healthcare industry section a unique feature is described to improve physician and patient productivity. Regarding system setup, administrator status is not required for intuitive user interaction. The features smoothly allow users to experience doing hospital-related tasks, starting from registering to accessing health reports. [1][2]*

### 4.1 Register

**Description:** New users (patients and doctors) may create accounts.



**Functional Requirements:**

- Name, email, phone, as well as password, enable user registration.
- Email verification or password verification activates accounts.
- At registration time users are able to choose just what their role will be. The role can be as a doctor or as a patient.
- There is a unique identifier for each account that exists.

## 4.2 Login

**Description:** Current users can access their accounts securely.

**Functional Requirements:**

- Users can enter using a password plus email/username.
- Dashboards for use with role-based redirection either by doctor or else by patient.
- Passwords are encrypted.

## 4.3 Doctor Scheduling and Management

**Description:** Doctors may set when they are free for scheduled meetings so management knows.

**Functional Requirements:**

- Time slots that are available can be created by the doctors.
- People can change slots or remove them.
- Patient booking depends solely on available slots.

## 4.4 Appointment Booking

**Description:** Allows patients to schedule appointments with doctors.

**Functional Requirements:**

- Patients select doctors through name or specialty.
- People see dates and times available when they request appointments.
- Patients can cancel appointments.

## 4.5 Medical Records Management

**Description:** Maintains and manages health-related data of patients.

**Functional Requirements:**

- Doctors input diagnoses and also input prescriptions and test results.
- They can view patients' records, but they cannot modify them.
- Stored securely, records can be downloaded as well.
- A record of history is kept.

## 4.6 Room and Bed Allocation Management

**Description:** Inpatients are assigned to hospital rooms in addition to beds.

**Functional Requirements:**

- Rooms are assigned to patients depending on availability.
- Records comprise duration of stay and also room type.
- Rooms are released automatically upon their discharge.

## 4.7 Emergency Case Handling

**Description:** A form deals with the description for emergency requests.

**Functional Requirements:**

- Patients can submit emergency forms with information required.
- Alerts reach doctors immediately. It is possible because of the system.
- Timestamps are indicators of when forms get stored and then prioritized.

## 4.8 Feedback and Complaint Management

*Description:* Patients can submit suggestions or complaints using this option.

*Functional Requirements:*

- A form that patients can access is available. They can submit feedback or issues via it.
- Doctors can view the complaints and can address those complaints.

## 4.9 Filtered Searching Process

**Description:** System data is searchable by users. System data is also filterable by users.

**Functional Requirements:**

The availability is checked for you while you are filtering options like the doctor name and the specialty and the date.

## 4.10 Health History and Reports Access (for patients)

*Description:* Patients are given with access that is for their full health records.

*Functional Requirements:*

- Visits plus diagnoses are shown. This display is chronological.
- Reports can be viewed and also downloaded by the users.
- Access is read-only for security.

# 5. Other Nonfunctional Requirements

## 5.1 Performance Requirements

The primary performance requirement is speed of internet network so that updates to database done elsewhere are accessible in real time. In case of large number of users accessing the database at once, the speed at which updates are refreshed might go down due to traffic.

## 5.2 Safety Requirements

*This project does not involve direct physical safety risks. However, in case of system or device failures, the data flow is safely halted and rolled back to a secure previous state to prevent data corruption or compromise.*

## 5.3 Security Requirements

*The system shall ensure secure access through user authentication, requiring valid usernames and passwords for all roles (admin, doctor, patient). User data, including medical records and personal information, must be encrypted and accessible only to authorized users based on their roles. The system shall follow standard data protection policies to prevent unauthorized access, modification, or leakage of sensitive information. Regular security audits and access logs shall be maintained to ensure compliance and traceability.*

## 5.4 Software Quality Attributes

### a) Availability:

- The system shall be available 99.5% of the time to ensure continuous access for patients and hospital staff.

### b) Usability:

- The user interface shall be simple, intuitive, and designed to minimize the learning curve for users (patients, doctors, and staff).

### c) Maintainability:

- The system codebase shall be modular and well-documented to support easy bug fixes and feature enhancements.
- Changes to one module (e.g., Feedback Management) should not affect unrelated modules (e.g., Room Allocation).

### d) Flexibility:

- Application is easily modifiable by developer to maintain and update with changing environment.

### e) Robustness:

- The system shall handle invalid inputs gracefully and provide meaningful error messages.
- Emergency features (e.g., form-based emergency care requests) shall operate under degraded conditions (e.g., partial database unavailability).

### f) Correctness:

- All functionalities such as appointment scheduling, patient registration, and report access must produce accurate results based on real-time data and constraints (e.g., no double booking of doctors)

**g) Search Efficiency:**

- The system must allow users to filter and view specific attributes (e.g., "usability," "reliability") based on criteria like priority or role relevance, while hiding irrelevant details for streamlined review.

## **5.5 Business Rules**

**a) User Roles and Access Control:**

- *Only registered users (doctors, patients, and admins) can log in to the system.*
- *Doctors can access and update medical records, manage appointments, and view the doctor master table.*
- *Patients can book appointments, access their health history and reports, and submit feedback or complaints.*

**b)Registration and Management:**

- *All users must register before accessing system features.*
- *Patient registration includes room and bed allocation as part of the admission process.*

**c)Appointment and Scheduling:**

- *Appointments can only be scheduled with doctors available in the **Doctor Master Table**.*
- *Time slots must not overlap and should reflect real-time availability.*

**d)Medical Records Handling:**

- *Only authorized doctors can update patient medical records.*
- *Patients can view but not modify their health records and reports.*

**e)Emergency Case Handling:**

- *Emergency cases are prioritized and must be submitted via a special emergency form.*

**f)Complaint and Feedback Submission:**

- *Feedback and complaints must be tied to a registered patient account and are submitted via a structured form.*

***g)Data Integrity and Search:***

- *Filtered search must return accurate, role-specific data (e.g., a patient sees their own records only).*