

**Course Title :**

***System Analysis and Design Sessional***

**Lab Report on :**

***SRS of Hospital Management System***

**Course Code :**

***SE 118***

**Date :**

***25 August, 2025***

**Submitted By :**

***S M Tahsin Shahav Id:2303008***

***Sree Madhurjo Chandra Sarker Id:2303006***

***Airin Akter Id : 230335***

***1<sup>st</sup> Batch***

***Department of Software Engineering***

**Submitted To :**

***Shifat Ara Rafiq***

***LECTURER***

***Department of Software Engineering***

## **Step 1: Introduction :**

Healthcare institutions are increasingly dependent on technology to streamline operations, reduce paperwork, and improve service quality. A Hospital Management System (HMS) is a comprehensive digital solution designed to manage hospital activities efficiently, including patient registration, appointment scheduling, billing, laboratory management, and medical record storage. Traditional hospital processes, which rely heavily on manual paperwork, often result in delays, human errors, and inefficiencies in managing large volumes of patient data. In contrast, a computerized HMS provides a centralized platform where patients, doctors, and administrative staff can interact seamlessly. Patients benefit from faster appointment booking and secure access to medical history, doctors gain tools for accurate diagnosis and prescription management, and hospital administrators can monitor resources, staff, and finances in real time. The adoption of such a system not only ensures accuracy and speed but also enhances patient satisfaction and reduces the workload of hospital staff. With the growing demand for quality healthcare services, the implementation of an effective Hospital Management System becomes essential for modern hospitals to remain competitive and deliver better care.

## **Project Scope :**

### **Product Description:**

The Hospital Management System is a web based management system designed to automate tasks, improve efficiency, provide real-time monitoring, manage hospital workload, ensuring data security, reducing patient harassment and improve efficiency. It is a replacement system which resolves the limitations of existing manual systems.

### **Relation to Business Goals:**

This software supports corporate objectives such as improving patients service, enhancing operational performance, ensuring data security, providing real-time monitoring, providing telemedicine platform etc. So it will create a great market value.

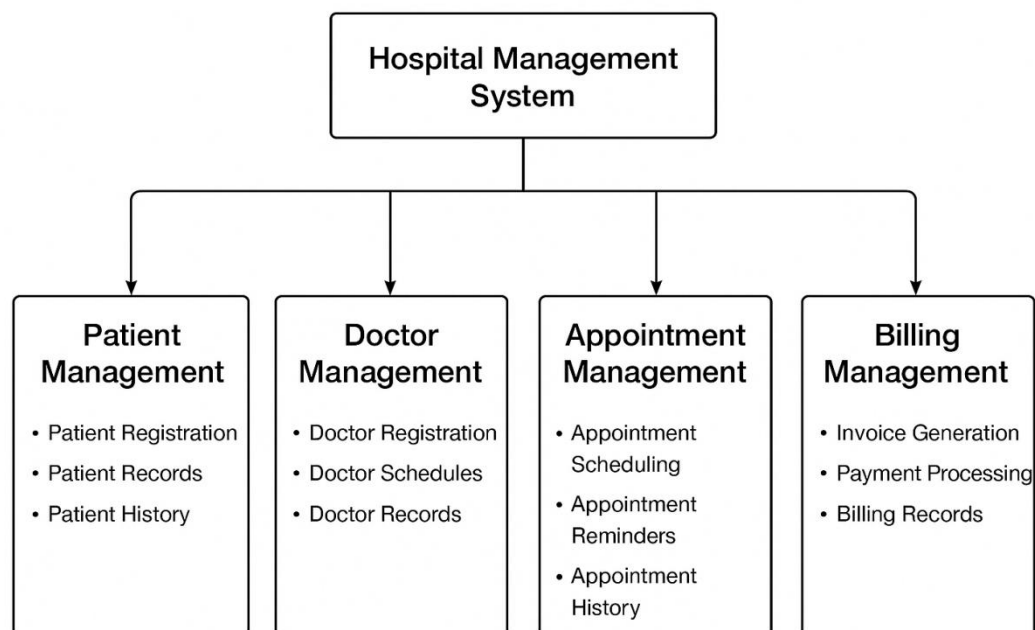
## Step 2: Overall Description :

### 1. Product Perspective :

- The Hospital Management System is a new, self-contained system designed to manage hospital's whole activities.
- It is a replacement system where it resolves the limitations of existing manual for inefficiency, lack of scalability and poor data accuracy.
- This product interacts with external systems such as payment gateways, cloud servers, IoT devices, sensors, etc.
- The product is a web-based and accessible via standard browsers and it's functionality is dependent on a stable internet connection.

### 2. Product Features :

The Hospital Management System is designed to manage hospital workload , ensuring data security, reducing patient harassment and improve efficiency. it is a replacement system which resolves the limitations of existing manual systems.



This is a diagram for some main features for the Hospital Management System but there are also some other features.

### ● Example Features :

- **Appointment Scheduling:** Allows patients to book appointments online or over the phone.
- **Tracking :** Tracks user movement within the hospital.
- **Prescribing :** Allows doctors to send prescriptions directly to the hospital's database.
- **Billing and Invoicing :** Generates accurate bills for services, procedures, and medications.
- **Doctor's Portal :** A personalized dashboard for physicians to manage their patient load, view schedules, access records, and write notes.
- **Ward and Bed Management :** This module optimizes hospital resources and improves the patient admission process.
- **Staff Scheduling :** Helps managers create and manage work schedules for doctors, nurses, and other staff.
- **Accessibility :** Allows authorized staff to access documents securely from anywhere within the hospital network.
- **Reports & Analytics :** Analyzing patient reports, billing reports, and overall hospital performance.
- **User Role-Based Access :** Restricts access to sensitive data based on the user's role like doctor, staff, admin.
- **Feedback & Surveys :** Allows the hospital to gather patient feedback on their experience, which can be used to improve service quality.
- **User Authentication & Login :** Patients, doctors, and administrators can all log in securely.

### 3. User Classes and Characteristics :

#### (a) Patients/ End Users :

- **Frequency of use:** Weekly to monthly
- **Functions used:** Login, view available doctor, make appointment, payment, get hospitalized.
- **Expertise:** Low to moderate technical knowledge (basic web using skills).
- **Privileges:** Limited to their uses.
- **Importance:** Most important user class.

### **(b) Doctors :**

- **Frequency of use:** Periodic (when advising patients or during login).
- **Functions used:** Login, view appointment, consult patients, regular checkup.
- **Expertise:** Moderate to high technical knowledge (familiarity with IoT devices)
- **Privileges:** Medium access rights .
- **Importance:** Secondary, supports main users.

### **(c) System Administrators :**

- **Frequency of use:** Occasional
- **Functions used:** System management, software updates, security monitoring
- **Expertise:** High technical knowledge (IT professionals)
- **Privileges:** Full administrative access
- **Importance:** Secondary but vital for long-term system reliability.

## **4. Operating Environment :**

### **Hardware Platform:**

- IT devices (Windows) for end-user app access.
- Cloud server for data storage and analytics.

### **Operating System:**

- Device level: Embedded OS
- Web Access : Windows(10+)

### **Software Dependencies:**

- HTTP protocol for IoT communication
- Cloud database (Firebase )

## **5. Design and Implementation Constraints :**

### **Design Constraints →**

These are rules, conditions, or restrictions that affect how the system should be designed. They limit the choices of architecture, technology, or structure. Example:

1. The system must run on Windows only.

2. The system must run with stable internet connection.
3. The system will block user if login details are wrong three times and if unauthorized access is found etc.

### **Implementation Constraints →**

These are restrictions that developers must follow while writing code and implementing the system. They limit how the system can be built.

Example:

1. The program must be written in Java language only.
2. The system must use hospital-approved authentication system.
3. Integration middleware may be restricted to licensed or approved tools etc.

## **Step 3: System Features :**

### **1. Feature Name : User Authentication & Login**

#### **● Description and Priority**

- Description: Provides secure access to the system by verifying the user's identity before allowing transactions.
- Priority: High
- Priority Components (1–9 scale):
  - Benefit: 9 (essential for security)
  - Penalty: 9 (severe if absent, leads to fraud)
  - Cost: 4 (moderate implementation cost)
  - Risk: 8 (critical if security fails)

#### **● Stimulus/Response Sequences**

- User visits web and attempts to login
- System prompts for credentials (PIN, password, username).
- User enters input.
- System validates input against backend database.
- If valid → grant access. If invalid → show error message and retry option.

- **Functional Requirements**

- FR1. The system shall allow users to log in using unique credentials (PIN/password/biometric).
- FR2. The system shall lock the account after three consecutive failed attempts.
- FR3. The system shall display error messages for invalid credentials.
- FR4. The system shall log all authentication attempts.

## **2. Feature Name : Appointment Scheduling**

- **Description and Priority**

- Description: Provides access to make appointment and schedule
- Priority: Medium
- Priority Components (1–9 scale):
  - Benefit: 9 (essential for patients)
  - Penalty: 9 (severe if absent, leads to useless)
  - Cost: 4 (moderate implementation cost)
  - Risk: 7 (critical if security fails)

- **Stimulus/Response Sequences**

- User visit web and attempt to make appointment
- System prompts available doctor
- User enters input info.
- System validates input against backend database.
- If valid → appointment done. If invalid → show error message and retry option.

- **Functional Requirements**

- FR1. The users should give proper details for making appointment
- FR2. Doctors should available for appointment
- FR3. The system shall display error messages for invalid appointment.
- FR4. The system can cancel appointment based on doctor and user's case.

### **3. Feature Name : Prescribing**

- **Description and Priority**

- Description: Provides Prescribing details to the users and database.

- Priority: Medium

- Priority Components (1–9 scale):

- Benefit: 9 (essential for patients and database)

- Penalty: 9 (severe if absent, leads to useless)

- Cost: 8 (moderate implementation cost)

- Risk: 9 (critical if security fails)

- **Stimulus/Response Sequences**

- Doctor prescribe patients

- Attempt to send prescribe details to database

- System analyze it and send it

- Database saves the details

- **Functional Requirements**

- FR1. The users should give properly prescribe

- FR2. Doctors should immediately save and send prescription details

### **4. Feature Name : Billing and Invoicing**

- **Description and Priority**

- Description: Provides total cost details and bill it

- Priority: Medium high

- Priority Components (1–9 scale):

- Benefit: 9 (essential for patients and system)

- Penalty: 9 (severe if absent, leads to useless)

- Cost: 4 (moderate implementation cost)

- Risk: 7 (critical if security fails)

- **Stimulus/Response Sequences**

- User visit web and attempt to make payment

- System prompts total cost as bill

- User make payment

- System confirm payment  
○ If valid → gives invoice. If invalid → show error message and retry option.



- **Functional Requirements**

- FR1. The users should ensure internate connection
- FR2. System should accurately calculate total cost
- FR3. System should save a copy of invoice in database

## **5. Feature Name : Reports & Analytics**

- **Description and Priority**

- Description: Analyzing patient reports, billing reports, and overall hospital performance.

- Priority: Medium

- Priority Components (1–9 scale):

- Benefit: 8 (essential for usability)
- Penalty: 7 (if absent, leads to loss usability)
- Cost: 4 (moderate implementation cost)
- Risk: 5 (medium if security fails)

- **Stimulus/Response Sequences**

- User visit web and attempt to report and rate
- System prompts report and rating option
- User enters their report and rating
- System analyse the report and rating.

- **Functional Requirements**

- FR1. The system shall allow users to report and rate about service

- FR2. The admin shall change the system acording to the report and rating if needed

## **6. Feature Name : User Role-Based Access**

### **• Description and Priority**

○ Description: Restricts access to sensitive data based on the user's role like doctor, staff, admin.

○ Priority: High

○ Priority Components (1–9 scale):

- Benefit: 9 (essential for security)
- Penalty: 9 (severe if absent, leads to loss data)
- Cost: 7 (moderate implementation cost)
- Risk: 9 (critical if security fails)

### **• Stimulus/Response Sequences**

○ User visit web and login

○ System give access according to their need

○ System restricts for unauthorized access

### **• Functional Requirements**

○ FR1. The system shall track all users activity in the system

○ FR2. The system shall display warning messages for unauthorized access

○ FR3. System shall block user if unauthorized access is found

## **Step 4: External Interface Requirements :**

### **1. User Interfaces :**

• **Logical Design:** Login screen → Dashboard → Module Access (Patient, Doctor, Billing, Pharmacy, Reports). Consistent navigation bar across all modules.

• **GUI Standards:** Responsive web design (desktop + mobile). Corporate theme: hospital branding (logo, colors: blue/white). Standard placement of navigation (top menu), footer with quick links.

• **Error Message Conventions:** “Invalid login. Please try again.”, “Appointment slot not available.”, “Payment failed. Try again later.”

- **UI Components Required:** Login/Sign-up screens, Patient dashboard (appointments, reports, billing), Doctor dashboard (schedule, patient records), Admin dashboard (hospital reports, staff management), Pharmacy module (stock, sales)

- **Accessibility Considerations:** Support for Bangla + English, Large-text mode for elderly patients, Color-blind–friendly theme

- **Example for HMS:**

- Login Page:** Username, password, forgot password option

- Dashboard:** Tiles for Patients, Doctors, Appointments, Billing, Pharmacy .

- Reports:** PDF/Excel export button on every reporting page

## 2. Hardware Interfaces :

- **Devices the system interacts with:**

- PC/Desktop & Mobile Devices:** Used by hospital staff, doctors, patients

- Printers:** For printing bills, prescriptions, lab reports

- Barcode Scanners (optional):** For patient ID cards, medicine stock management

- Servers:** Hospital central database server

- **Data/Control Interactions:**

- Patient records stored/retrieved from server database

- Billing and lab reports automatically generated for printing

- **Protocols / Physical Connections:**

- Devices connected over LAN/Wi-Fi inside hospital

- Cloud backup server accessed via secure internet connection

## 3. Software Interfaces :

- **Operating System:** Runs on Windows; Android/iOS

- **APIs:** Payment Gateway (bKash, Nagad, Card), SMS/Email Gateway for notifications

- **Data Exchange:** JSON format for API communication . Example:  
{ "patient\_id": 101, "appointment\_status": "confirmed", "bill\_amount": 2000 }

#### 4. Communications Interfaces :

- **Communication Methods:** HTTPS protocol for all client-server communication, SMS/Email for appointment confirmations & reminders
- **Security Considerations:** Encrypted patient data (AES-256), Role-based authentication (patient, doctor, admin)
- **Performance Expectations:** Appointment booking confirmation within 2 seconds, Billing report generation within 3 seconds, Real-time notification delivery ( $\leq 5$  seconds delay)

#### 5. Nonfunctional Requirements :

##### **(1) Performance Requirements**

- System must handle 500+ concurrent users.
- Appointment booking response time  $\leq 2$  seconds.
- Billing process must support 1000+ transactions per day.
- Database queries optimized for high performance.

##### **(2) Safety Requirements**

- **Data Backup:** Automatic daily backup to cloud storage.
- **Disaster Recovery:** Recovery point objective (RPO)  $\leq 24$  hours.
- **System Downtime:** Maximum unplanned downtime  $\leq 2$  hours/year.
- **Error Handling:** Prevent system crashes with safe failover.

##### **(3) Security Requirements**

- Confidentiality, Integrity, Availability (CIA) ensured.
- **Authentication:** Secure login with strong password policy.
- **Authorization:** Role-based access (patients cannot see other patient data).
- **Encryption:** All sensitive data encrypted (AES-256, HTTPS).
- **Compliance:** Must follow HIPAA/GDPR guidelines for medical data.

##### **(4) Software Quality Attributes**

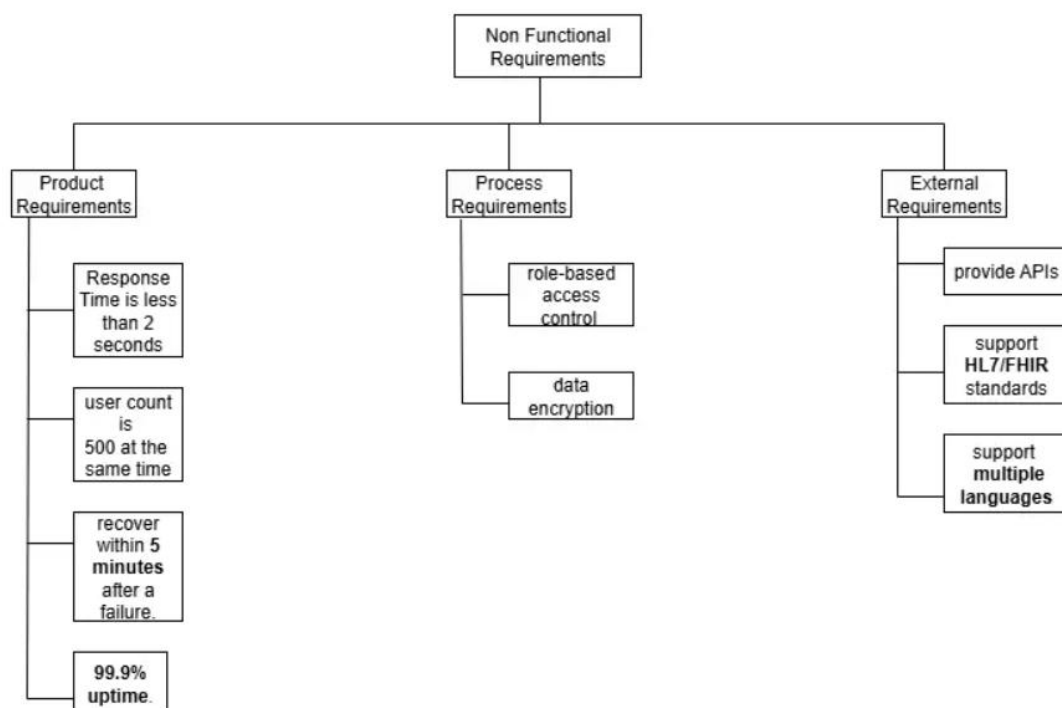
- **Usability:** Easy to learn for hospital staff; intuitive dashboard.
- **Reliability:** 99.9% uptime guaranteed.
- **Maintainability:** System updates with zero downtime.
- **Portability:** Compatible with Windows/Linux servers, mobile devices.
- **Interoperability:** Supports integration with third-party lab equipment or pharmacy software.
- **Scalability:** Able to expand for 10+ hospitals in future.

##### **Other Requirements**

- **Database Requirements:** Centralized database with replication.

- **Internationalization:** Support for Bangla & English languages.
- **Legal Requirements:** Must comply with healthcare data privacy regulations.
- **Reuse Objectives:** Modules (Patient, Billing, Pharmacy) designed for reuse in other healthcare systems.
- **Audit & Logging:** Every action logged for audit trail (who did what & when).

### Other Requirements :



**Figure 1. Non-Functional requirements of a Hospital Management System**