Software Requirements Specification for

IPD/OPD MANAGEMENT SYSTEM

Version 1.0 approved. Prepared By  
 GAUTAM SINGH, JUILEE SHET, YASH SINGH   
 DY PATIL (RAIT)  
 16/7/2025

OPD/IPD Record System

# Table of Contents

1. Introduction

1.1 Purpose

1.2 Objectives

1.3 Document Conventions

1.4 Intended Audience

1.5 Product Scope

1.6 References

2. Overall Description

2.1 System Context

2.2 Stakeholders

2.3 Product Functions

2.4 User Characteristics

2.5 Operating Environment

2.6 Constraints

2.7 Documentation

2.8 Assumptions

3. External Interface Requirements

3.1 User Interfaces

3.2 Hardware Interfaces

3.3 Software Interfaces

3.4 Communication Interfaces

4. System Features

4.1 Core Patient & User Management

4.2 OPD Management

4.3 IPD Management

4.4 Billing Management

4.5 Administration & Reporting

5. Non-Functional Requirements

5.1 Performance

5.2 Safety

5.3 Security

5.4 Scalability

5.5 Reliability

5.6 Usability

5.7 Maintainability

5.8 Business Rules

5.9 Others

6. Appendices

6.1 Glossary

6.2 System Architecture

6.3 Data Model

# Introduction

## Purpose

This document provides a detailed Software Requirements Specification (SRS) for the Integrated Inpatient/Outpatient Department (IPD/OPD) Record System, version 1.0. Its purpose is to define the complete scope of the product, outlining all functional and

non-functional requirements necessary for its development, deployment, and maintenance. This SRS will serve as the guiding document for project managers, developers, testers, and all involved stakeholders.

## Project Goals and Objectives

The primary goal of the IPD/OPD Record System is to digitize and integrate patient health records and associated administrative workflows across outpatient and inpatient seflings to enhance patient care, improve operational efficiency, and ensure data integrity.

### Measurable Objectives:

* **Objective 1:** Reduce patient registration and admission time by 40% within 12 months of launch.
* **Objective 2:** Decrease medication administration errors by 25% within the first year by implementing the eMAR and e-prescription modules.
* **Objective 3:** Improve billing accuracy by 95%, reducing revenue leakage from missed charges within 18 months.
* **Objective 4:** Ensure system uptime of 99.9% to provide a consistently reliable service for clinical operations.

## Document Conventions

This SRS conforms to the IEEE 830-1998 standard.

* **Priorities:** Requirements are prioritized as High, Medium, or Low. High priority indicates a core feature essential for the Minimum Viable Product (MVP).
* **Requirement Numbering:** Functional Requirements (FR) and Non-Functional Requirements (NFR) are uniquely identified for traceability.
* **Definitions:** "Shall" indicates a mandatory requirement. "Should" indicates a desirable

feature. "Will" refers to a future event or system action.

## Intended Audience and Reading Suggestions

This document is intended for a diverse audience, including:

* **Developers:** To understand the system's functionalities and constraints for implementation.
* **Project Managers:** To oversee project progress, scope, and resource allocation.
* **Testers:** To develop test cases and verify system compliance with requirements.
* **Hospital Administrators & Staff (Doctors, Nurses):** To understand the system's capabilities and how it addresses their needs.
* **System Administrators:** To understand operational requirements and maintenance aspects.

It is suggested that readers begin with the "Introduction" and "Overall Description" for a high-level understanding. Developers and testers should then proceed to "System Features" and "Other Non-Functional Requirements".

## Product Scope

The IPD/OPD Record System is a comprehensive, centralized digital platform designed to manage the patient journey from outpatient consultation to inpatient admission, treatment, and discharge. The system will replace fragmented paper-based records and disparate software with a unified ecosystem. Its purpose is to provide a single source of truth for all patient clinical and administrative data, accessible in real-time to authorized personnel. This system directly supports the corporate goal of leveraging technology to deliver superior healthcare services and improve patient outcomes.

## References

This section lists other documents or Web addresses to which this SRS refers.

* IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications.
* The Digital Personal Data Protection Act (DPDPA), 2023, India.
* Health Insurance Portability and Accountability Act (HIPAA) Security Rule.

# Overall Description

## Product Perspective and System Context

The IPD/OPD Record System is a new, self-contained, cloud-native platform designed to serve as the central information system for a hospital's clinical and administrative operations. It will integrate with existing, but separate, laboratory and radiology information systems (LIS/RIS) via HL7 messaging or APIs. The system's major components include user interfaces for different clinical and administrative roles, a modular backend built on microservices, and secure external API integrations.

## Stakeholders

* **Patients:** Individuals receiving care.
* **Doctors (Consultants, Surgeons):** Primary clinical users responsible for diagnosis and treatment.
* **Nurses:** Users responsible for patient care, medication administration, and monitoring.
* **Receptionists/Front Desk Staff:** Users responsible for patient registration and appointment scheduling.
* **Hospital Administrators:** Management personnel overseeing operations and requiring reports.
* **Lab Technicians/Pharmacists:** Users who will interact with the system for orders and results.
* **System Administrators:** Technical team maintaining the platform.

## Product Functions Summary

The IPD/OPD Record System will provide the following major functions:

* **Patient Information Management:** Centralized registration and lifelong electronic health record for every patient.
* **OPD Management:** Appointment scheduling, consultation notes, e-prescriptions, and outpatient billing.
* **IPD Management:** Admission, Discharge, Transfer (ADT) workflows, bed management, clinical charting, and medication administration.
* **Billing and Financials:** Integrated billing for all services, charge capture, and financial

reporting.

* **System Administration:** User management, role-based access control, and system auditing.

## User Classes and Characteristics

The system identifies distinct user classes, each with specific roles and access levels.

### Doctors (Clinical Users):

* + **Characteristics:** Tech-savvy professionals who require quick access to comprehensive patient data. They need efficient interfaces for writing notes, placing orders, and reviewing patient history.
  + **Key Interactions:** Viewing patient dashboard, writing consultation notes, creating e-prescriptions, placing lab/radiology orders, viewing patient history, managing admission requests.

### Nurses (Clinical Users):

* + **Characteristics:** Frontline caregivers who are constantly mobile. They require

easy-to-use interfaces, often on tablets or mobile devices, for real-time charting and medication administration.

* + **Key Interactions:** Managing ADT requests, updating vital signs, administering medication via eMAR, maintaining nursing notes, managing patient diet plans.

### Receptionists (Administrative Users):

* + **Characteristics:** The first point of contact for patients. They need a simple, fast interface for searching, registering patients, and managing appointments.
  + **Key Interactions:** Registering new patients, searching for existing patients, booking/rescheduling/canceling appointments, managing patient queues, collecting initial payments.

### Hospital Administrators (Management Users):

* + **Characteristics:** Management personnel focused on operational efficiency, resource utilization, and financial performance. They need access to high-level dashboards and detailed reports.
  + **Key Interactions:** Viewing bed occupancy rates, monitoring financial reports (revenue, outstanding payments), generating operational reports (patient volumes, wait times).

### System Administrators (Technical Users):

* + **Characteristics:** Technical personnel responsible for the overall operation, maintenance, and security of the platform.
  + **Key Interactions:** User account management, role and permission configuration, system monitoring, audit log review, managing master data (e.g., service charges).

## Operating Environment

* **Hardware Platform:** Cloud infrastructure (e.g., AWS, Azure, GCP).
* **Operating System:** Linux-based container orchestration environments (e.g., Kubernetes).
* **Other Software Components:** Docker containers, message brokers (e.g., RabbitMQ), relational (PostgreSQL) and NoSQL (MongoDB) databases, third-party APIs.
* **Client Devices:** Modern web browsers (Chrome, Firefox, Safari, Edge) on desktops and tablets.

## Design and Implementation Constraints

* **Regulatory Constraints:** As a "Data Fiduciary" under the DPDPA, the system shall obtain explicit, informed consent for all personal data collection and processing. All data handling must comply with relevant healthcare data protection laws.
* **Technical Constraints:** The backend architecture shall be microservices-based to allow independent scaling of services like Patient-Service, Appointment-Service, and

Billing-Service.

* **Security Constraints:** All cryptographic keys used for data encryption shall be managed using a secure vault service (e.g., AWS KMS). All data in transit and at rest shall be encrypted.
* **Interoperability:** The system shall support HL7 messaging standards for integration with external LIS and RIS systems.

## User Documentation

The following user documentation components will be delivered:

* **Online Help:** Context-sensitive help accessible within the application.
* **User Manuals:** Comprehensive role-based guides (for Doctors, Nurses, Receptionists) in PDF format.
* **FAQs Section:** A publicly accessible section addressing common queries.
* **Tutorials/Walkthroughs:** Short, guided tours for key functionalities upon first login.

## Assumptions and Dependencies

* **Internet Connectivity:** Users are assumed to have stable internet access to interact with the system.
* **Third-Party API Availability:** The system depends on the continuous availability of external APIs (e.g., SMS Gateway for notifications).
* **User Data Accuracy:** The effectiveness of the system depends on the accuracy and completeness of data entered by hospital staff.
* **Staff Training:** It is assumed that the hospital will provide adequate training to all user classes to ensure proper adoption and usage of the system.

# External Interface Requirements

## User Interfaces (UI/UX Philosophy)

* **Role-Based Dashboards:** The system will feature distinct dashboards tailored to the needs of each user class (Doctor, Nurse, Receptionist, Admin).
* **Mobile-First & Responsive Design:** The interface will be designed for tablets and desktops, ensuring a seamless experience across devices used in a hospital sefling.
* **Clarity and Simplicity:** The design will minimize clicks and use universally understood medical icons and clear typography. The goal is a "zero-training-required" experience for common tasks.
* **Consistency:** Adherence to a consistent design system across all modules for a cohesive user experience.
* **Feedback Mechanisms:** Clear visual and textual feedback for user actions, system status, and error conditions.

## Hardware Interfaces

* **Barcode/QR Code Scanners:** The system should be able to accept input from USB or Bluetooth barcode/QR code scanners for patient identification (from wristbands) and medication verification.
* **Label Printers:** The system shall interface with thermal printers to print patient

wristbands and sample labels.

## Software Interfaces

### SMS Gateway API (e.g., Twilio):

* + **Purpose:** For sending appointment reminders, OTPs for patient portal access, and critical alerts.
  + **Error Handling:** The system shall implement a retry mechanism for failed SMS delivery.
* **LIS/RIS Integration:** The system will use RESTful APIs or an HL7 message bus to:
  + Send lab and radiology orders to respective systems.
  + Receive structured results back from LIS/RIS.
* **Internal API (RESTful):** All microservices will expose RESTful APIs for inter-service communication, adhering to OpenAPI specifications.

## Communications Interfaces

* **HTTPS:** All API endpoints and web traffic shall enforce TLS 1.3 for secure communication.
* **WebSockets:** Will be used for real-time updates on clinician dashboards, such as new lab results, bed status changes, and nurse call alerts.
* **Message Bus (e.g., RabbitMQ):** Used for asynchronous, event-driven communication between microservices to ensure loose coupling and fault tolerance, particularly for order processing.

# System Features

## Module 1: Core Patient & User Management (MVP)

This module focuses on the foundational functionalities for managing patient and user identities.

## Feature 1.1: Patient Registration and Record Management

* + - * **Description and Priority:** Enables the creation and maintenance of a unique electronic health record for every patient. This is of **High** priority as it is the foundation of the entire system.

### Stimulus/Response Sequences:

* + - * + **Stimulus:** Receptionist selects "Register New Patient."
        + **Response:** System presents a registration form. Receptionist fills in demographic details. System validates data and checks for duplicates based on mobile number and government ID. System creates a new patient record with a unique Patient ID (PID) and generates a printable PID card.

### Granular Functional Requirements:

* + - * + **FR-01.1:** The system shall allow registration via a mobile number verified by a One-Time Password (OTP) sent to the patient.
        + **FR-01.2:** The system shall capture essential demographic data: First Name, Last Name, Date of Birth, Gender, Mobile Number, Address, and an optional Government ID number.
        + **FR-01.3:** The system shall automatically check for potential duplicate patient records in real-time during registration and alert the user.
        + **FR-01.4:** Upon successful registration, the system shall generate a unique, non-editable Patient ID (PID).

## Feature 1.2: User Account Management

* + - * **Description and Priority:** Allows System Administrators to create and manage user accounts for hospital staff. This is of **High** priority for system security and access control.

### Stimulus/Response Sequences:

* + - * + **Stimulus:** Administrator navigates to the "User Management" section and clicks "Create User."
        + **Response:** System displays a form to enter user details (Name, Department, Role). Admin assigns a role (e.g., Doctor, Nurse). System creates the account and sends an activation link to the user's email.

### Granular Functional Requirements:

* + - * + **FR-02.1:** The System Administrator shall be able to create, update, deactivate, and delete user accounts.
        + **FR-02.2:** The system shall provide predefined roles (Admin, Doctor, Nurse, Receptionist) with a default set of permissions.
        + **FR-02.3:** The System Administrator shall be able to assign one or more roles to a user.

## Module 2: Outpatient Department (OPD) Management

This module handles all workflows related to outpatient visits.

## Feature 2.1: Appointment Scheduling

* + - * **Description and Priority:** Allows receptionists and patients (via a future portal) to book, reschedule, and cancel appointments with specific doctors. This is of **High** priority.

### Stimulus/Response Sequences:

* + - * + **Stimulus:** Receptionist searches for a doctor's availability.
        + **Response:** System displays the doctor's calendar with available slots. Receptionist selects a slot, links a patient record, and confirms the booking. System sends an SMS confirmation to the patient.

### Granular Functional Requirements:

* + - * + **FR-03.1:** The system shall allow users to view doctor schedules by day, week, and month.
        + **FR-03.2:** The system shall allow booking of appointments for registered patients.
        + **FR-03.3:** The system shall send automated SMS and/or email reminders to patients 24 hours before their scheduled appointment.
        + **FR-03.4:** The system shall manage appointment statuses (Scheduled, Checked-In, Completed, Canceled, No-Show).

## Feature 2.2: Outpatient Consultation Workflow

* + - * **Description and Priority:** Provides doctors with the tools to manage an outpatient consultation. This is of **High** priority.

### Stimulus/Response Sequences:

* + - * + **Stimulus:** Doctor logs in and views their appointment queue for the day. Doctor selects a "Checked-In" patient.
        + **Response:** System displays the patient's clinical dashboard, including past visit history, allergies, and recent lab results. Doctor records presenting complaints, diagnosis, and treatment plan in a structured template.

### Granular Functional Requirements:

* + - * + **FR-04.1:** The system shall provide a doctor's dashboard showing a real-time queue of waiting patients.
        + **FR-04.2:** The system shall provide structured templates for entering consultation

notes (e.g., SOAP notes).

* + - * + **FR-04.3:** Doctors shall be able to place electronic orders for laboratory tests and radiology exams directly from the consultation screen.
        + **FR-04.4:** The system shall display prominent allergy alerts if the patient has a recorded allergy.

## Feature 2.3: E-Prescription Management

* + - * **Description and Priority:** Allows doctors to generate legible, error-free electronic prescriptions. This is of **High** priority for patient safety.

### Stimulus/Response Sequences:

* + - * + **Stimulus:** During a consultation, a doctor clicks "Create e-Prescription."
        + **Response:** System displays an interface to search for and add medications from a pre-loaded drug database. Doctor specifies dosage, frequency, and duration. System checks for potential drug-to-drug interactions. Doctor digitally signs and finalizes the prescription, which can be printed or sent to the pharmacy.

### Granular Functional Requirements:

* + - * + **FR-05.1:** The system shall include a searchable, pre-populated database of medications.
        + **FR-05.2:** The system shall allow doctors to create and save frequently used prescription templates.
        + **FR-05.3:** The system should provide basic clinical decision support, such as alerts for drug allergies and potential drug-to-drug interactions.
        + **FR-05.4:** The system shall generate a printable, legally compliant prescription.

## Module 3: Inpatient Department (IPD) Management

This module manages the complete lifecycle of an admifled patient.

## Feature 3.1: Admission, Discharge, and Transfer (ADT)

* + - * **Description and Priority:** Manages the entire inpatient admission, inter-ward transfer, and discharge process. This is of **High** priority.

### Stimulus/Response Sequences:

* + - * + **Stimulus (Admission):** Doctor recommends admission from an OPD consultation. An

ADT request is sent to the IPD front desk.

* + - * + **Response:** IPD desk staff view the request, assign an available bed, complete financial formalities, and formally admit the patient. The patient's status in the system changes to "Admifled."

### Granular Functional Requirements:

* + - * + **FR-06.1:** The system shall allow authorized users to create an admission request for a patient.
        + **FR-06.2:** The system shall manage the patient's status throughout the ADT lifecycle (Requested, Admifled, Transferred, Discharged).
        + **FR-06.3:** The system shall allow for the transfer of a patient from one bed/ward to another, maintaining a log of all transfers.
        + **FR-06.4:** The system shall support a planned discharge process, including generating a discharge summary.

## Feature 3.2: Bed Management

* + - * **Description and Priority:** Provides a real-time visual overview of all beds in the hospital, their status, and patient assignments. This is of **Medium** priority.

### Stimulus/Response Sequences:

* + - * + **Stimulus:** A nurse or ward manager accesses the "Bed Board."
        + **Response:** The system displays a graphical representation of all wards and rooms. Each bed is color-coded to indicate its status (e.g., Occupied, Available, Needs Cleaning, Blocked). Hovering over an occupied bed shows key patient details.

### Granular Functional Requirements:

* + - * + **FR-07.1:** The system shall provide a real-time, graphical bed board view.
        + **FR-07.2:** Users shall be able to filter the bed board view by ward and room type.
        + **FR-07.3:** The system shall allow authorized users to change the status of a bed (e.g., from "Discharge Pending" to "Needs Cleaning" to "Available").

## Feature 3.3: Electronic Medication Administration Record (eMAR)

* + - * **Description and Priority:** Provides nurses with an electronic flowsheet to document medication administration, enhancing safety and accuracy. This is of **High** priority.

### Stimulus/Response Sequences:

* + - * + **Stimulus:** A nurse accesses the eMAR for a specific patient.
        + **Response:** The system displays a timeline of all scheduled medications for the patient. The nurse selects a medication due for administration, scans the patient's wristband barcode and the medication's barcode to verify. The system validates the "Five Rights" (Right Patient, Right Drug, Right Dose, Right Route, Right Time). Upon successful validation, the nurse administers the drug and signs off electronically.

### Granular Functional Requirements:

* + - * + **FR-08.1:** The eMAR shall automatically populate with medication orders from doctors.
        + **FR-08.2:** The system shall support barcode scanning for patient and medication verification before administration.
        + **FR-08.3:** The system shall log the exact time, date, and user for every medication administration action.
        + **FR-08.4:** The system shall flag overdue medication administrations with visual alerts.

## Module 4: Billing and Financial Management

This module automates all financial transactions related to patient care.

## Feature 4.1: Service and Charge Master

* + - * **Description and Priority:** A centralized repository for all billable services, items, and their prices. This is of **High** priority for accurate billing.

### Stimulus/Response Sequences:

* + - * + **Stimulus:** A finance administrator needs to update the price of a lab test.
        + **Response:** The administrator accesses the Charge Master interface, searches for the specific lab test code, and updates the associated price. The new price is effective immediately for all future billing.

### Granular Functional Requirements:

* + - * + **FR-09.1:** The system shall provide a centralized interface for an administrator to define and manage all billable services, procedures, and items.
        + **FR-09.2:** Each item in the charge master shall have a unique code, description, and price.
        + **FR-09.3:** The system should support different price lists for different patient categories (e.g., General, Insurance).

## Feature 4.2: Inpatient and Outpatient Billing

* + - * **Description and Priority:** Automates the process of capturing charges and generating consolidated bills. This is of **High** priority.

### Stimulus/Response Sequences:

* + - * + **Stimulus:** A patient is being discharged. The billing officer initiates final bill generation.
        + **Response:** The system automatically collates all unbilled charges from the patient's entire stay (room rent, doctor fees, medications from eMAR, lab tests). The system generates a consolidated, itemized bill. The officer can add discounts, apply insurance, and accept payment.

### Granular Functional Requirements:

* + - * + **FR-10.1:** The system shall automatically capture charges for services rendered (e.g., consultations, lab tests, medications) and post them to the patient's account in

real-time.

* + - * + **FR-10.2:** The system shall be able to generate interim bills for inpatients upon request.
        + **FR-10.3:** The system shall generate a final, detailed, and itemized bill upon patient discharge.
        + **FR-10.4:** The system shall support multiple payment modes, including cash, credit/debit cards, and UPI.

## Module 5: System Administration & Reporting

This module provides tools for system oversight, management, and data analysis.

## Feature 5.1: Administrative Dashboard

* + - * **Description and Priority:** Provides a high-level, real-time overview of key hospital metrics for administrators. This is of **Medium** priority.

### Stimulus/Response Sequences:

* + - * + **Stimulus:** A hospital administrator logs into the system.
        + **Response:** The system displays a dashboard with key performance indicators (KPIs) in graphical formats, such as current bed occupancy rate, total outpatient visits

today, total revenue collected, and average patient wait time.

### Granular Functional Requirements:

* + - * + **FR-11.1:** The admin dashboard shall display key operational metrics in real-time.
        + **FR-11.2:** The dashboard widgets shall be configurable by the administrator.
        + **FR-11.3:** Users should be able to drill down from dashboard widgets to view detailed reports.

## Feature 5.2: Audit and Reporting

* + - * **Description and Priority:** Provides tools for auditing system activity and generating various reports. This is of **High** priority for compliance and analysis.

### Stimulus/Response Sequences:

* + - * + **Stimulus:** An administrator needs to check who accessed a specific patient's record.
        + **Response:** The administrator uses the audit trail feature, filters by the Patient ID, and views a log of all users who accessed the record, along with timestamps.

### Granular Functional Requirements:

* + - * + **FR-12.1:** The system shall log every access and modification of patient data, accessible only to authorized administrators for audit purposes.
        + **FR-12.2:** The system shall provide a comprehensive reporting module with

pre-defined reports (e.g., Patient Demographics, Billing summaries, Bed Occupancy reports).

* + - * + **FR-12.3:** The system shall allow reports to be exported to standard formats like PDF and CSV.

# Other Non-Functional Requirements

## Performance Requirements

* **NFR-01.1 (Latency):** 95th percentile API response time for critical read operations (e.g., search patient, fetch patient record) shall be < 500ms.
* **NFR-01.2 (Load Time):** Web page Largest Contentful Paint (LCP) shall be < 2.5 seconds for all primary user dashboards.
* **NFR-01.3 (Throughput):** The system shall support 200 concurrent active users during

peak hours (10 AM - 2 PM) without degradation in response times.

## Safety Requirements

* **NFR-S1.1:** The system shall prevent the display of patient Personally Identifiable Information (PII) to unauthorized users.
* **NFR-S1.2:** The system shall implement robust data validation for critical medical information, including medication dosages, allergies, and lab value units, to prevent potential data entry errors.
* **NFR-S1.3:** In case of system failure, critical data (active patient records, medication orders, recent transactions) shall be recoverable within the defined RTO/RPO to minimize impact on patient care.
* **NFR-S1.4:** The system shall have mandatory verification steps (e.g., barcode scanning) in the eMAR module to minimize the risk of medication errors.

## Security Requirements

* **NFR-04.1 (Authentication):** Passwords shall be hashed using Argon2. Multi-Factor Authentication (MFA) shall be enforced for all administrative and remote access.
* **NFR-04.2 (Authorization):** A user with a "Receptionist" role shall not be able to access clinical sections like consultation notes or eMAR, even if they know the URL. Access will be strictly controlled by roles and permissions.
* **NFR-04.3 (Data Protection):** All personally identifiable information (PII) and protected health information (PHI) in the database shall be encrypted at rest using AES-256. Data in transit shall be secured using TLS 1.3.
* **NFR-04.4 (Audit Trails):** The system shall maintain comprehensive, immutable audit trails for all C.R.U.D. (Create, Read, Update, Delete) actions on patient and financial records.
* **NFR-04.5 (Session Management):** The system shall enforce automatic session timeouts after 15 minutes of inactivity and require re-authentication.

## Scalability Requirements

* **NFR-02.1 (Statelessness):** All backend services shall be designed to be stateless to allow for horizontal scaling via load balancers, enabling the system to handle increasing

load by adding more compute instances.

* **NFR-02.2 (Elasticity):** The system's cloud infrastructure shall be configured to automatically scale resources (compute, database connections) up or down based on real-time demand fluctuations, particularly during peak hospital hours.

## Availability and Reliability Requirements

* **NFR-03.1 (Uptime):** The system's core services (patient records, ADT, eMAR) shall have a guaranteed uptime of 99.9%, measured monthly.
* **NFR-05.1 (RTO/RPO):** The Recovery Time Objective (RTO) for critical services shall be < 1 hour. The Recovery Point Objective (RPO) shall be < 5 minutes. This means the system should be back online within an hour of a disaster, with a maximum data loss of 5 minutes.
* **NFR-05.2 (Error Handling):** The system shall provide graceful error handling and user-friendly, informative messages in case of system failures or invalid inputs, guiding the user on the next steps.

## Usability and Accessibility Requirements

* **NFR-06.1 (Usability):** The system shall aim for a System Usability Scale (SUS) score of > 85, indicating excellent perceived usability among clinical and administrative staff.
* **NFR-06.2 (Accessibility):** The web interface shall be compliant with Web Content Accessibility Guidelines (WCAG) 2.1 Level AA, ensuring accessibility for users with disabilities.
* **NFR-06.3 (Learnability):** A new user (e.g., a doctor, nurse) should be able to complete their primary tasks (e.g., register a patient, complete a consultation, administer a medication) within 20 minutes of their first use without external assistance.

## Maintainability Requirements

* **NFR-07.1 (Modularity):** The system will be developed as a set of independent microservices (e.g., Patient Service, Billing Service, Appointment Service), facilitating independent development, deployment, and maintenance without impacting the entire system.
* **NFR-07.2 (Testability):** All code shall have a minimum unit test coverage of 85% to ensure code quality and reduce regressions. A comprehensive suite of integration and end-to-end tests shall also be implemented.
* **NFR-07.3 (Documentation):** All APIs shall be documented using the OpenAPI specification. All codebases and deployment procedures shall be well-documented to facilitate future maintenance and onboarding of new developers.

## Business Rules

* **BR-01:** A patient record can only be created if a unique mobile number is provided and verified.
* **BR-02:** An e-prescription cannot be finalized without a valid, registered doctor's digital signature.
* **BR-03:** A patient cannot be discharged if there is an outstanding balance on their account, unless overridden by an authorized administrator.
* **BR-04:** A bed's status must be "Available" before a new patient can be assigned to it.
* **BR-05:** Administration of a high-risk medication may require a second nurse verification within the eAR module.

## Other Requirements

* **Legal Compliance:** The system shall comply with all relevant national and local laws pertaining to healthcare data, including the Digital Personal Data Protection Act (DPDPA) 2023, and any specific state-level health regulations.
* **Localization:** The system interface should be designed to support future localization into regional languages, although the initial release will be in English.

# Appendices

## Appendix A: Glossary

* **ADT:** Admission, Discharge, Transfer - The process of managing a patient's stay in the hospital.
* **DPDPA:** Digital Personal Data Protection Act - India's data privacy law.
* **eMAR:** Electronic Medication Administration Record - A digital log for tracking medication administration.
* **HL7:** Health Level Seven - A standard for exchanging information between medical applications.
* **IPD:** Inpatient Department - The section of a hospital where patients are admifled for overnight stays.
* **LCP:** Largest Contentful Paint - A Core Web Vital metric for web page load speed.
* **OPD:** Outpatient Department - The section of a hospital where patients receive consultations without being admifled.
* **PII:** Personally Identifiable Information - Any data that can be used to identify a specific individual.
* **RPO:** Recovery Point Objective - The maximum acceptable amount of data loss after an incident.
* **RTO:** Recovery Time Objective - The targeted time within which a business process must be restored after a disaster.
* **SRS:** Software Requirements Specification.
* **SUS:** System Usability Scale - A questionnaire for measuring perceived usability.
* **TLS:** Transport Layer Security - A protocol for encrypting internet traffic.
* **UC:** Use Case.
* **WCAG:** Web Content Accessibility Guidelines.

## Appendix B: System Architecture Overview

The system will be built on a microservices architecture deployed on a cloud platform using containers (Docker) and orchestration (Kubernetes). This architecture ensures scalability, resilience, and maintainability.

* **API Gateway:** A single entry point for all client requests, handling routing, authentication (JWT validation), and rate limiting.
* **Patient Service:** Manages all patient-related data, including demographics, registration, and search functionality.
* **User Service:** Manages user accounts, roles, permissions, and authentication logic.
* **Appointment Service:** Manages all OPD appointment scheduling, rescheduling, and status updates.
* **Encounter Service:** Manages clinical encounters, including OPD consultation notes and IPD progress notes.
* **ADT Service:** Manages the inpatient admission, discharge, and transfer workflows and

bed management.

* **Billing Service:** Manages the charge master, captures charges from other services, and generates bills.
* **Notification Service:** Manages sending all emails, SMS, and push notifications asynchronously.
* **Database:** Each service will have its own independent database (Database-per-Service paflern) to ensure loose coupling. A mix of PostgreSQL for transactional data and MongoDB for unstructured clinical notes will be used.
* **Message Bus (RabbitMQ):** Used for asynchronous, event-driven communication between services. For example, when an Encounter is completed, it publishes an event that the Billing Service subscribes to in order to capture charges.

## Appendix C: Conceptual Data Model

This is a simplified, conceptual model. The final database schema will be more detailed.

### Patients Table:

* + PatientID (PK), FirstName, LastName, DateOfBirth, Gender, MobileNumber (Unique), Email, Address, CreatedAt, UpdatedAt.

### Users Table:

* + UserID (PK), FullName, Email (Unique), HashedPassword, Role (e.g., 'Doctor', 'Nurse'), IsActive, CreatedAt.

### Appointments Table:

* + AppointmentID (PK), PatientID (FK), DoctorID (FK), AppointmentDateTime, Status ('Scheduled', 'Completed', 'Canceled'), Notes, CreatedAt.

### Admissions (ADT) Table:

* + AdmissionID (PK), PatientID (FK), AdmissionDate, DischargeDate, AdmiflingDoctorID (FK), CurrentBedID (FK), Status ('Admifled', 'Discharged').

### ConsultationNotes Table:

* + NoteID (PK), AppointmentID (FK), PatientID (FK), DoctorID (FK), SubjectiveNotes, ObjectiveNotes, Assessment, Plan, CreatedAt.

### Prescriptions Table:

* + PrescriptionID (PK), AppointmentID (FK), PatientID (FK), DoctorID (FK), CreatedAt.

### Prescription\_Items Table:

* + PrescriptionItemID (PK), PrescriptionID (FK), MedicationName, Dosage, Frequency, Duration.

### Billings Table:

* + BillID (PK), PatientID (FK), AdmissionID (FK, Nullable), TotalAmount, AmountPaid, Status ('Unpaid', 'Paid', 'Partially Paid'), BillDate.

### Bill\_Items Table:

* + BillItemID (PK), BillID (FK), ServiceCode, Description, Charge, Quantity.

### AuditTrail Table:

* + LogID (PK), UserID (FK), Timestamp, Action (e.g., 'View\_Patient\_Record', 'Update\_Billing'), EntityID (e.g., PatientID), OldValue (JSON), NewValue (JSON).