Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

School of Computer Science and Engineering

CLINIC MANAGEMENT SYSTEM

Team Members:

No.	Registration Number	Name
1.	20BCE0978	SRESTH KHAITAN
2.	21BCE3137	SHANTANU ANAND
3.	22BCE2379	SARAH PATRO
4.	22BCI0243	NIKHIL MATHUR

Faculty – Dr. Swathi J.N

Course – Software Engineering – BCSE301P

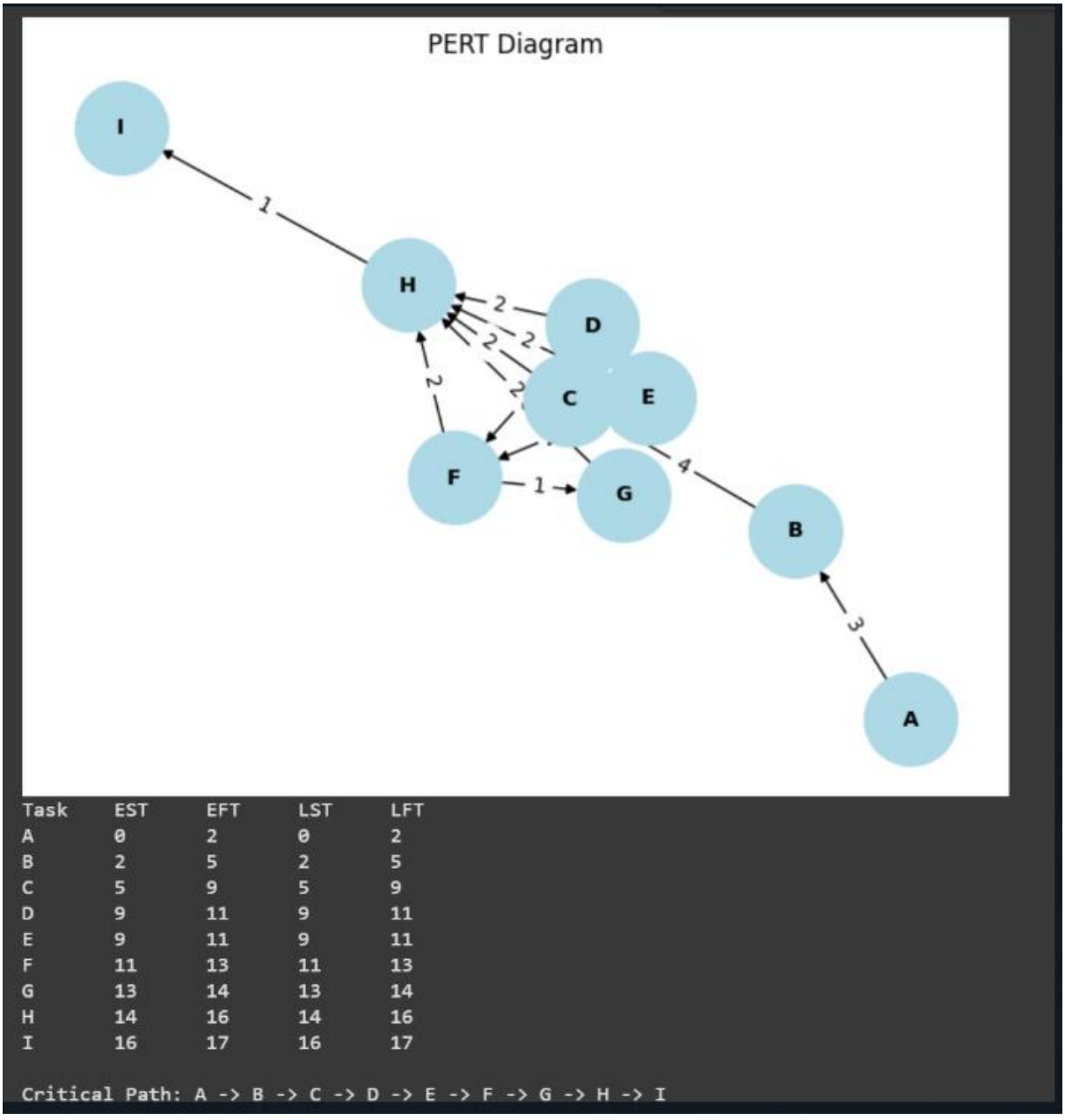
Lab Slot – L37+L38

Winter Semester 24-25

Activity Network/ PERT Chart

Activity Number	Label	Description	Duration (Weeks)
A	Project Initiation & Planning	Define scope, objectives, stakeholders, timeline, and requirements	2
B	System Design	Define architecture, roles, wireframes, and finalize design	3
C	Core Functionalities Implementation	Develop patient registration, appointment scheduling, billing, and user authentication	4
D	Lab Test Management	Implement lab test request, tracking, and results upload	2
E	Prescription Management	Implement prescription generation and saving	2
F	Reporting & Audit Logs	Generate reports and implement system logs	2
G	System Backup	Implement manual backup and data recovery	1
H	Testing	Perform unit, integration, system, and security testing	2
I	Deployment	Deploy system incrementally and train users	1
J	Maintenance	Monitor system performance and apply fixes	

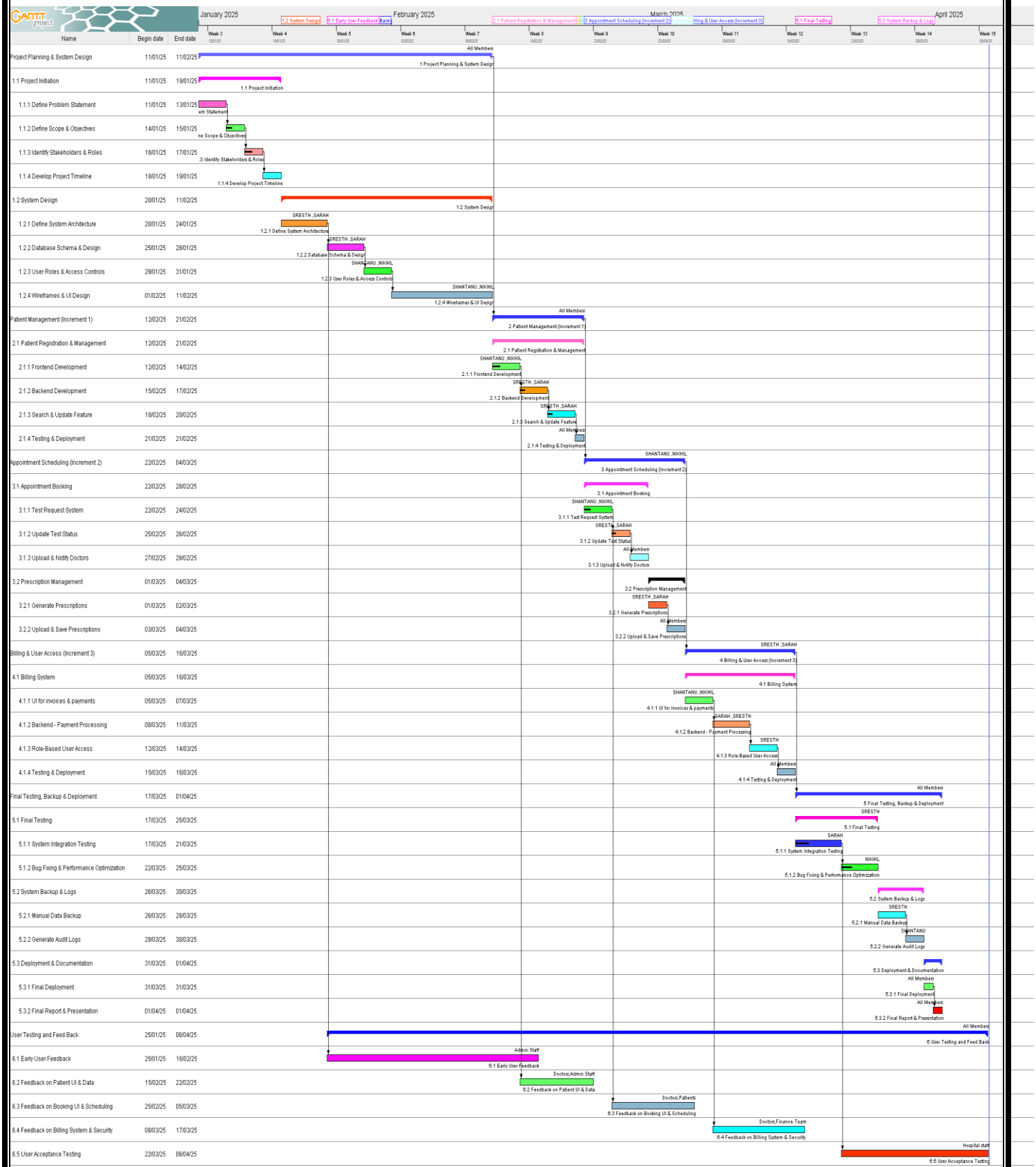
Task Label	Predecessor(s)	Duration (Weeks)
A	-	2
B	A	3
C	B	4
D	C	2
E	C	2
F	D, E	2
G	F	1
H	C, D, E, F, G	2
I	H	1
J	I	Ongoing



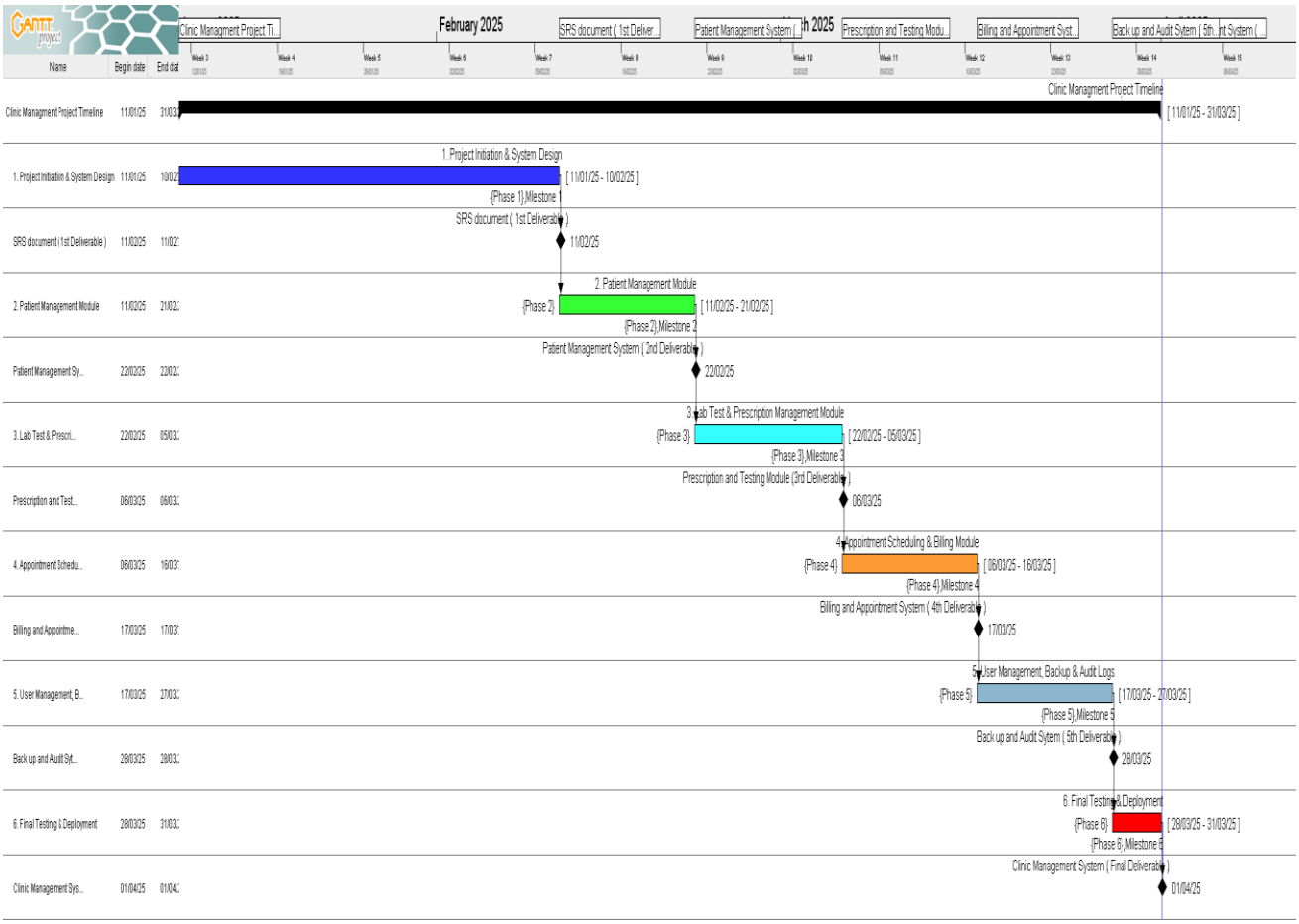
Task	EST	EFT	LST	LFT
A	0	2	0	2
B	2	5	2	5
C	5	9	5	9
D	9	11	9	11
E	9	11	9	11
F	11	13	11	13
G	13	14	13	14
H	14	16	14	16
I	16	17	16	17

Critical Path: A -> B -> C -> D -> E -> F -> G -> H -> I

Gantt Chart


January 2025
February 2025
March 2025
April 2025
Name
Begin date
End date
Project Planning & System Design 11/01/25 11/02/25
1.1 Project Initiation 11/01/25 19/01/25
1.1.1 Define Problem Statement 11/01/25 13/01/25
1.1.2 Define Scope & Objectives 14/01/25 15/01/25
1.1.3 Identify Stakeholders & Roles 16/01/25 17/01/25
1.1.4 Develop Project Timeline 18/01/25 19/01/25
1.2 System Design 20/01/25 11/02/25
1.2.1 Define System Architecture 20/01/25 24/01/25
1.2.2 Database Schema & Design 25/01/25 28/01/25
1.2.3 User Roles & Access Controls 29/01/25 31/01/25
1.2.4 Wireframes & UI Design 01/02/25 11/02/25
Patient Management (Increment 1) 12/02/25 21/02/25
2.1 Patient Registration & Management 12/02/25 21/02/25
2.1.1 Frontend Development 12/02/25 14/02/25
2.1.2 Backend Development 15/02/25 17/02/25
2.1.3 Search & Update Feature 18/02/25 20/02/25
2.1.4 Testing & Deployment 21/02/25 21/02/25
Appointment Scheduling (Increment 2) 22/02/25 04/03/25
3.1 Appointment Booking 22/02/25 28/02/25
3.1.1 Test Request System 22/02/25 24/02/25
3.1.2 Update Test Status 25/02/25 26/02/25
3.1.3 Upload & Notify Doctors 27/02/25 28/02/25
3.2 Prescription Management 01/03/25 04/03/25
3.2.1 Generate Prescriptions 01/03/25 02/03/25
3.2.2 Upload & Save Prescriptions 03/03/25 04/03/25
Billing & User Access (Increment 3) 05/03/25 16/03/25
4.1 Billing System 05/03/25 16/03/25
4.1.1 UI for invoices & payments 05/03/25 07/03/25
4.1.2 Backend - Payment Processing 08/03/25 11/03/25
4.1.3 Role-Based User Access 12/03/25 14/03/25
4.1.4 Testing & Deployment 15/03/25 16/03/25
Final Testing, Backup & Deployment 17/03/25 01/04/25
5.1 Final Testing 17/03/25 25/03/25
5.1.1 System Integration Testing 17/03/25 21/03/25
5.1.2 Bug Fixing & Performance Optimization 22/03/25 25/03/25
5.2 System Backup & Logs 26/03/25 30/03/25
5.2.1 Manual Data Backup 26/03/25 28/03/25
5.2.2 Generate Audit Logs 29/03/25 30/03/25
5.3 Deployment & Documentation 31/03/25 01/04/25
5.3.1 Final Deployment 31/03/25 31/03/25
5.3.2 Final Report & Presentation 01/04/25 01/04/25
User Testing and Feed Back 25/01/25 06/04/25
6.1 Early User Feedback 25/01/25 16/02/25
6.2 Feedback on Patient UI & Data 15/02/25 22/02/25
6.3 Feedback on Booking UI & Scheduling 25/02/25 05/03/25
6.4 Feedback on Billing System & Security 08/03/25 17/03/25
6.5 User Acceptance Testing 22/03/25 06/04/25

Timeline Chart



SOFTWARE REQUIREMENTS SPECIFICATION REPORT

1. Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) document is to define the functional, non-functional, and system requirements for the Clinic Management System (CMS). This system is designed to streamline clinic operations by managing patient records, appointments, medical history, billing, and laboratory tests. It will enhance operational efficiency, reduce manual effort, and improve patient care through a centralized, secure, and user-friendly interface.

This document also outlines the constraints, assumptions, and dependencies of the project, ensuring clarity and consistency in the development process. Furthermore, it defines the system's expected attributes, such as scalability, security, reliability, and maintainability, to ensure long-term usability and adaptability.

1.2 Scope

The Clinic Management System is being developed for a small clinic, enabling seamless management of patient data, appointment scheduling, and financial transactions. The system will serve multiple stakeholders, including receptionists, doctors, lab staff, and administrators, each with specific roles and permissions.

The key functionalities include:

- **Patient Management:** Register new patients, update existing records, and store medical history.
- **Appointment Scheduling:** Book, reschedule, or cancel appointments based on doctor availability.
- **Doctor Consultation Module:** View patient history, record diagnoses, and issue prescriptions.
- **Billing and Payments:** Generate invoices, process payments, and manage financial transactions.
- **Lab Test Management:** Handle test requests, upload results, and notify relevant personnel.
- **User Role Management:** Assign access permissions to different stakeholders.
- **Security & Compliance:** Implement role-based access control (RBAC) and data encryption to ensure compliance with healthcare regulations.

This system will be developed within a **three-month timeframe** with essential features, allowing for incremental enhancements in future versions.

1.3 Definitions, Acronyms, and Abbreviations

Term	Description
SRS	Software Requirements Specification
CMS	Clinic Management System
RBAC	Role-Based Access Control
HIPAA	Health Insurance Portability and Accountability Act (US Healthcare Regulation)
GDPR	General Data Protection Regulation (EU Data Privacy Law)
UI/UX	User Interface/User Experience
DBMS	Database Management System

1.4 References

Reference	Description
IEEE Std 830-1998	IEEE Recommended Practice for Software Requirements Specifications
Clinic Policy Documents	Internal policies defining data handling and security regulations
Software Development Best Practices	Industry guidelines for secure and maintainable software development

1.5 Overview

This SRS document is structured to provide a comprehensive understanding of the Clinic Management System requirements. The upcoming sections will cover:

- **Overall Description:** Product Perspective, product Functions, user characteristics, general constraints, assumptions and dependencies, entity relationships and data flow diagrams.
- **Specific Requirements:** External interface requirements, Functional and non-functional requirements, user roles, design constraints and system features.
- **Constraints Requirements:** Limitations, assumptions, and dependencies.
- **Hardware and software requirements**.

This document serves as a foundation for developers, testers, and stakeholders, ensuring alignment with the clinic’s operational needs and project goals.

2. Overall Description

2.1 Product Perspective

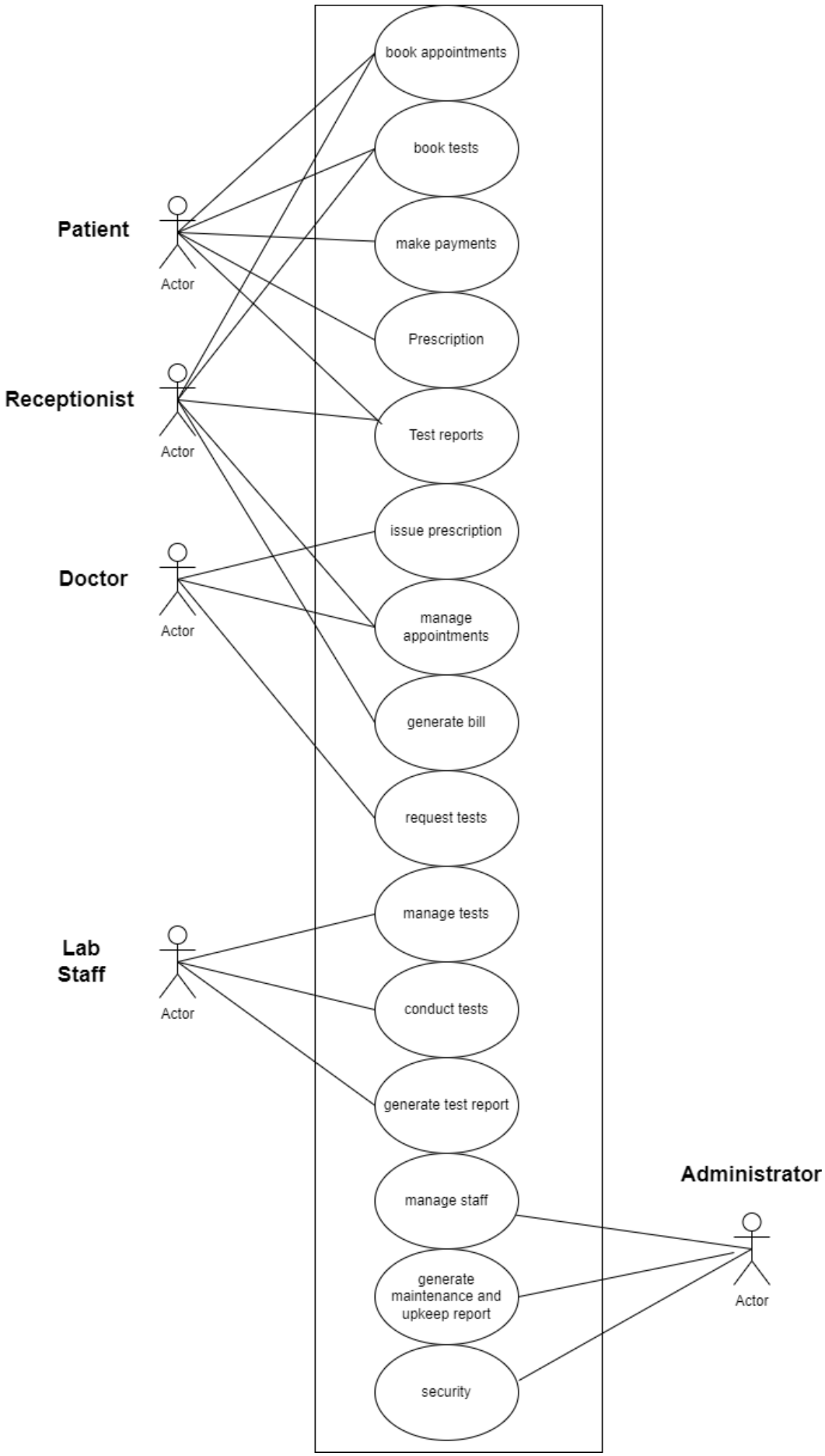
The Clinic Management System (CMS) is a standalone software application designed to assist small clinics in managing patient records, appointments, billing, and administrative tasks. The system will serve as an initial implementation, with future enhancements possible through incremental updates.

The CMS will integrate with third-party services for SMS and email notifications, as well as payment gateways for online transactions. The system will be accessible via web and desktop interfaces, ensuring flexibility for clinic staff.

2.2 Product Functions

The system shall provide the following core functionalities:

- **Receptionist Functions:**
  - Patient registration and profile management.
  - Appointment scheduling, rescheduling, and cancellation.
  - Billing and payment processing.
- **Doctor Functions:**
  - Access to patient medical records.
  - Prescription and treatment plan management.
  - Appointment tracking and management.
- **Admin Functions:**
  - Staff and role management.
  - Financial and billing oversight.
  - Compliance and security management.
- **Lab Staff Functions:**
  - Test request management and execution tracking.
  - Test result reporting and history management.
  - Inventory tracking for lab supplies.


book appointments
book tests
make payments
Prescription
Test reports
issue prescription
manage appointments
generate bill
request tests
manage tests
conduct tests
generate test report
manage staff
generate maintenance and upkeep report
security
Patient
Actor
Receptionist
Actor
Doctor
Actor
Lab Staff
Actor
Administrator
Actor

2.3 User Characteristics

The system will cater to the following user groups:

- **Receptionists:** Basic computer proficiency required for handling patient registration and scheduling.
- **Doctors:** Familiar with medical records management and prescription handling.
- **Admins:** Responsible for overseeing clinic operations, requiring familiarity with administrative and financial tasks.
- **Lab Staff:** Need access to test request and result management features, with minimal technical training.

2.4 General Constraints

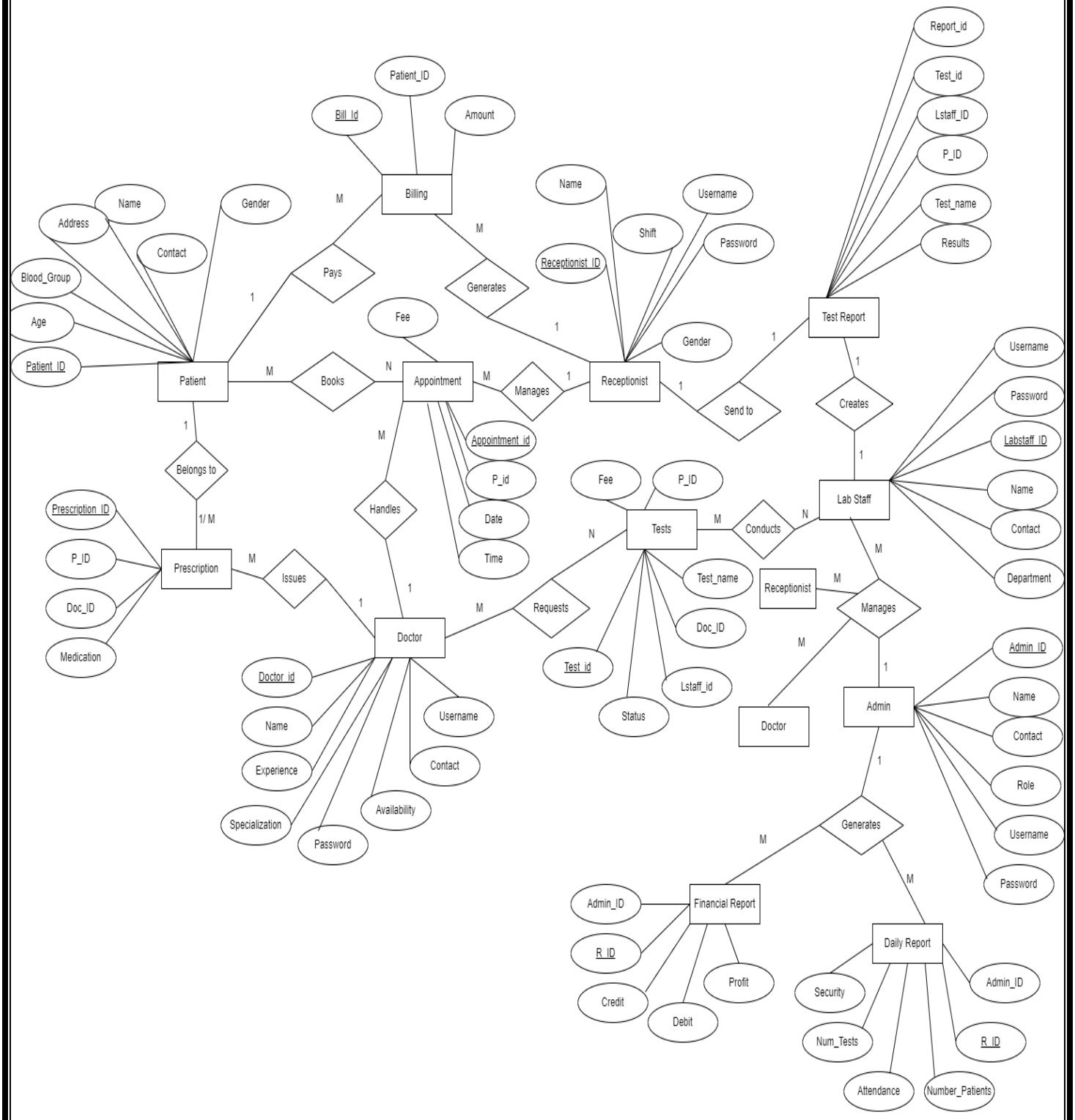
The system must operate within the following constraints:

- Must comply with healthcare data protection regulations (e.g., HIPAA, GDPR).
- Should be accessible via standard web browsers and desktop applications.
- Should support multiple user roles with appropriate access control mechanisms.
- Must ensure data security through encryption and secure authentication methods.
- Must provide a backup and disaster recovery mechanism to prevent data loss.

2.5 Assumptions and Dependencies

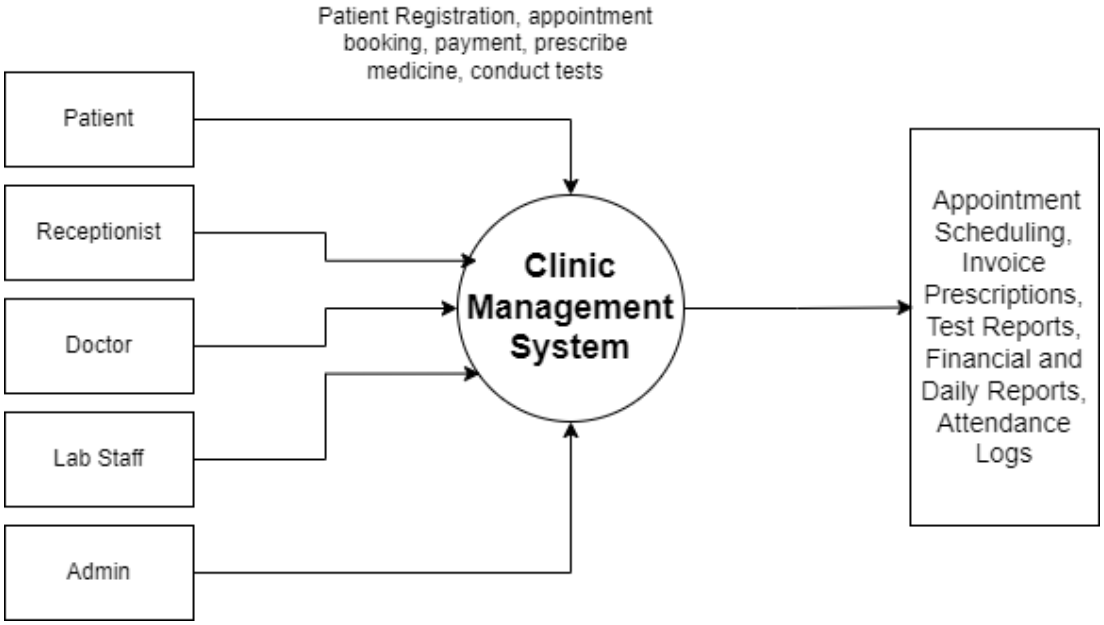
- The clinic has reliable internet connectivity to support cloud-based features such as email and SMS notifications.
- The system will be hosted on a secure and scalable server infrastructure.
- External third-party services (e.g., SMS gateways, payment processors) will be available and functional for integration.
- Users will have basic technical proficiency to operate the system with minimal training.
- Future feature enhancements will be implemented incrementally based on clinic requirements and feedback.

2.6 Entity Relationship Diagram

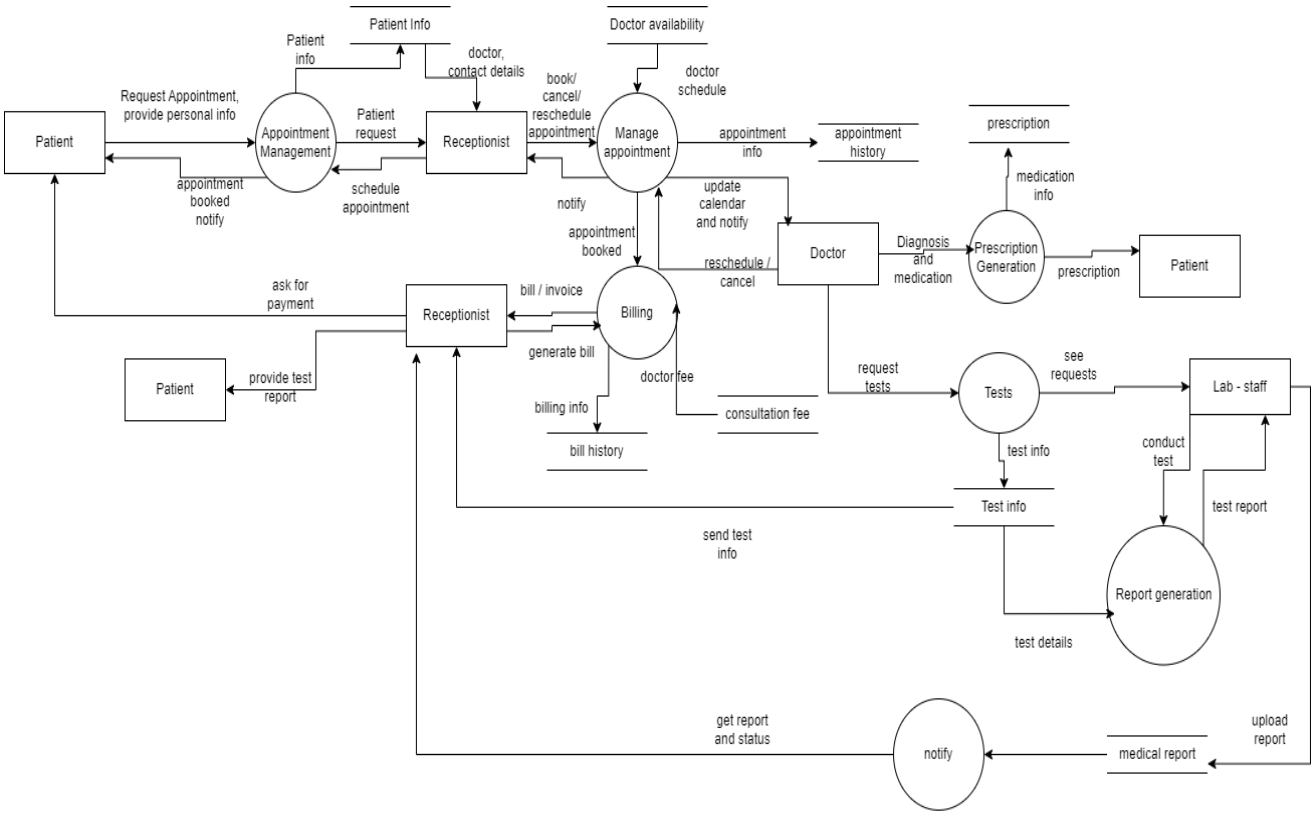


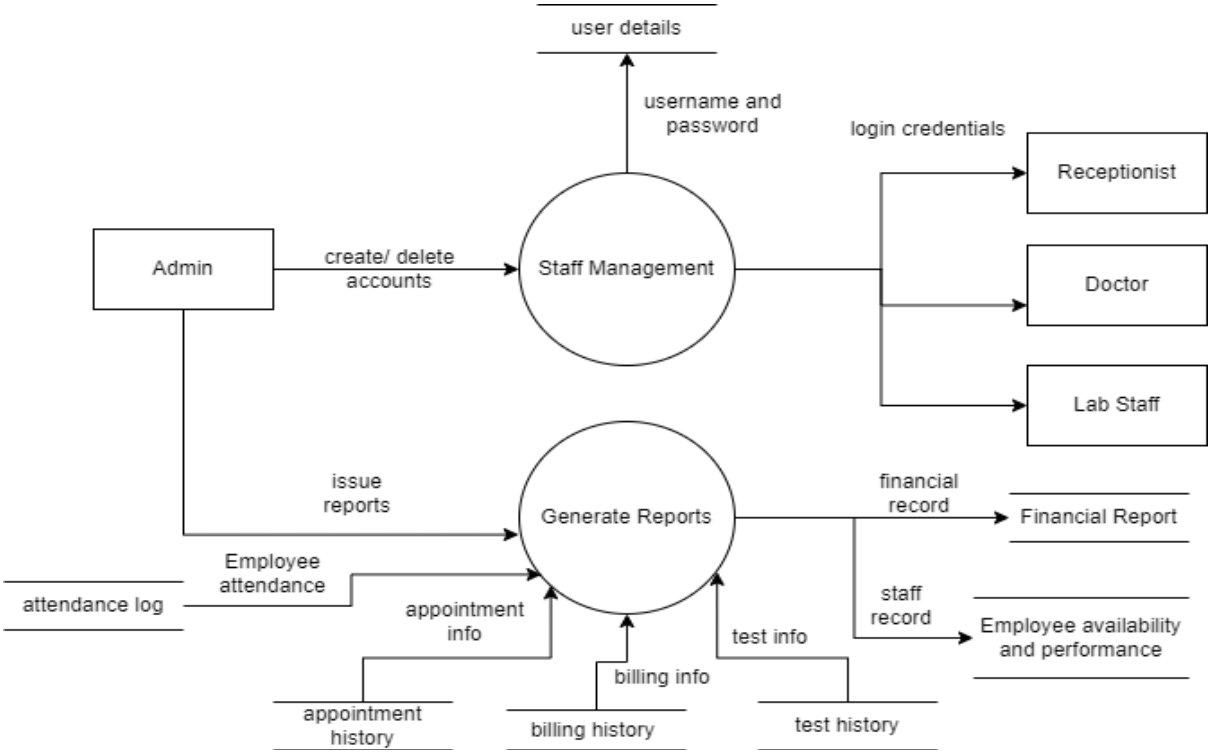
2.7 Data Flows

Context Level DFD



Level 1 DFD





3. Specific Requirements

3.1 External Interface Requirements

User Interfaces

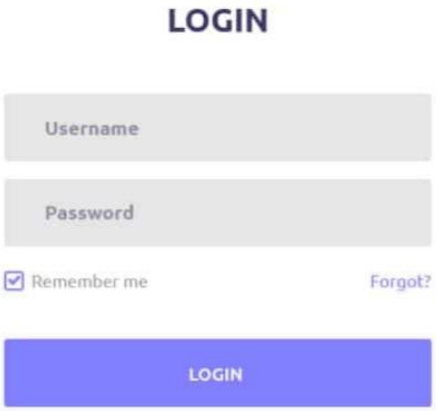
The Clinic Management System will provide an intuitive user interface (UI) for seamless interaction between the users (administrators, doctors, nurses, and patients) and the system. The UI will be designed to ensure ease of use, accessibility, and efficiency in performing various tasks such as scheduling appointments, managing patient records, generating prescriptions, and monitoring patient history.

Human Interaction with the System:

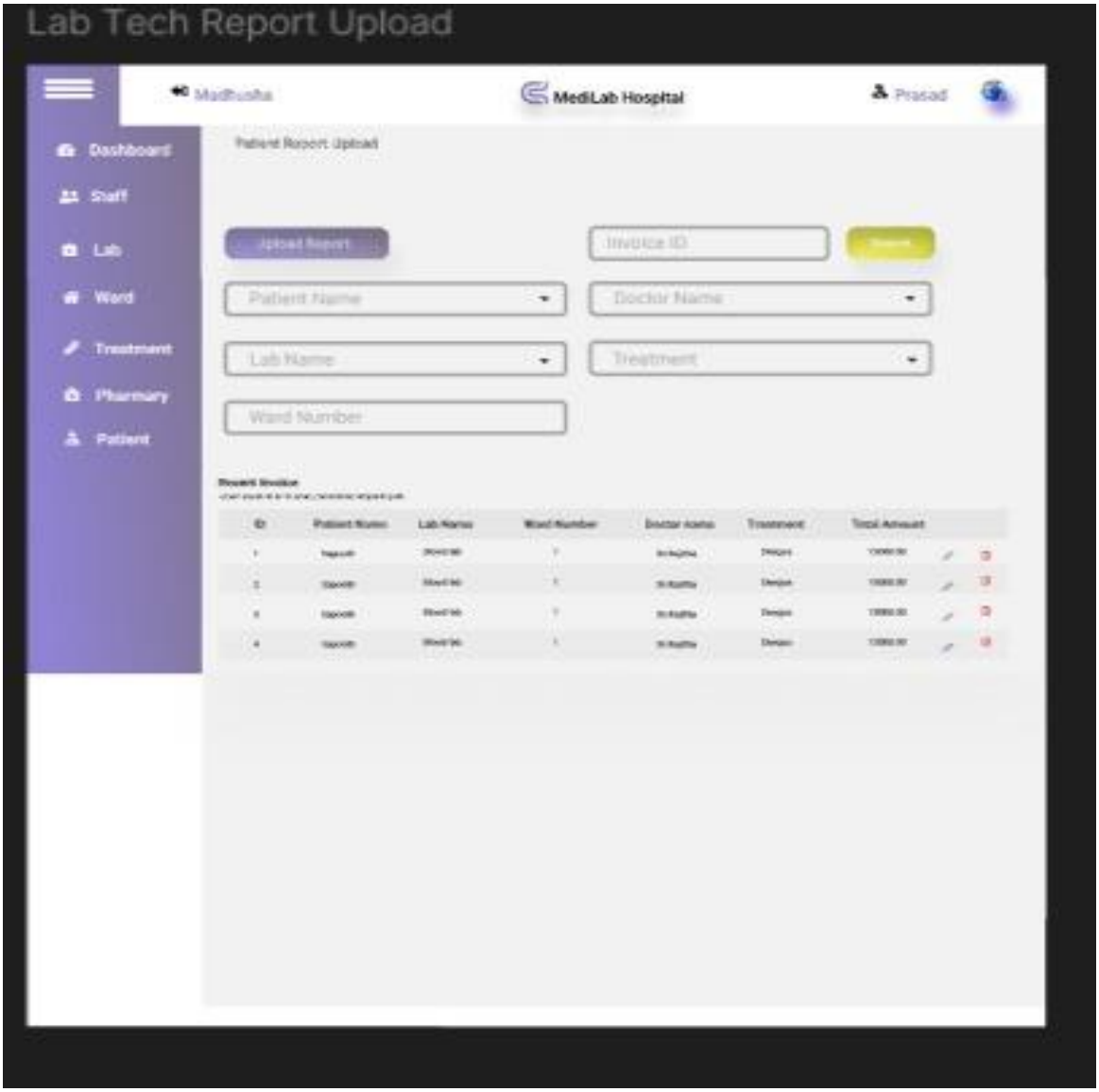
- Users will interact with the system through graphical user interfaces, which will include forms, buttons, drop-down menus, and search bars for easy navigation.
- Data input from the user (e.g., patient details, appointment information) will be through text fields, date pickers, and dropdown options.
- The system will generate visual outputs such as reports, alerts, and confirmation messages.
- Screenshots and wireframes of the user interface will be added here.

Home Page

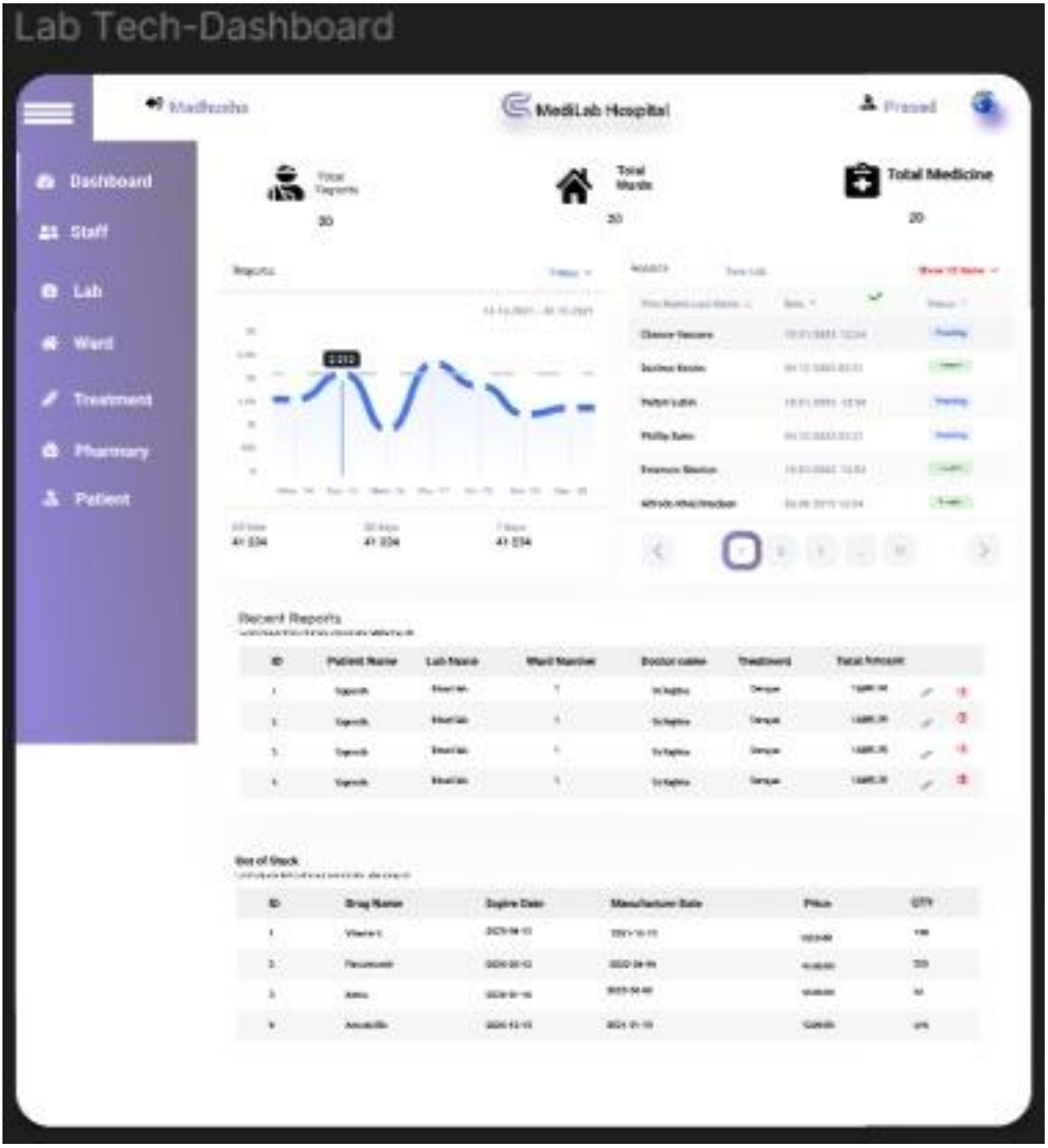
Login



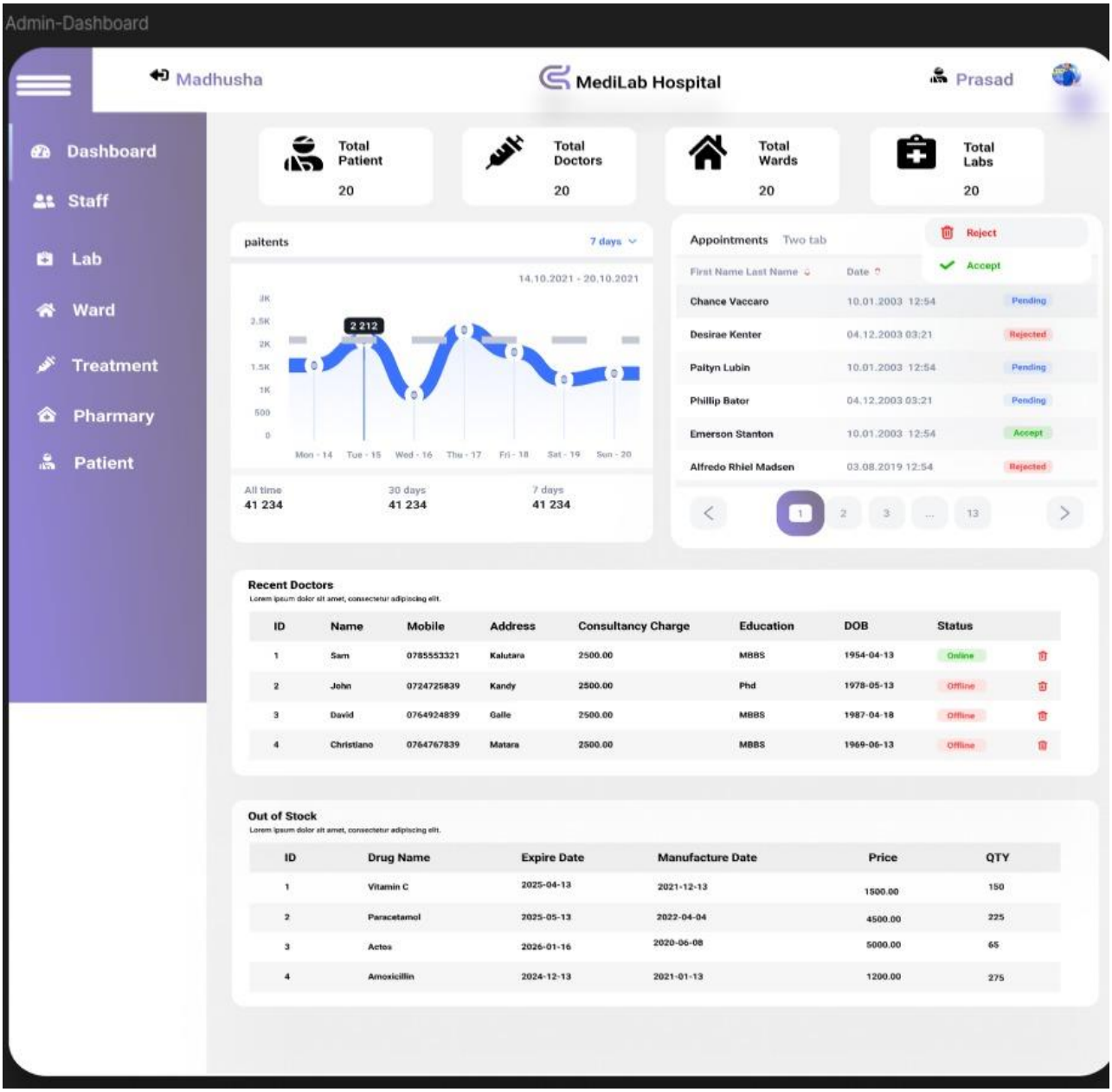
Lab Test Report Upload

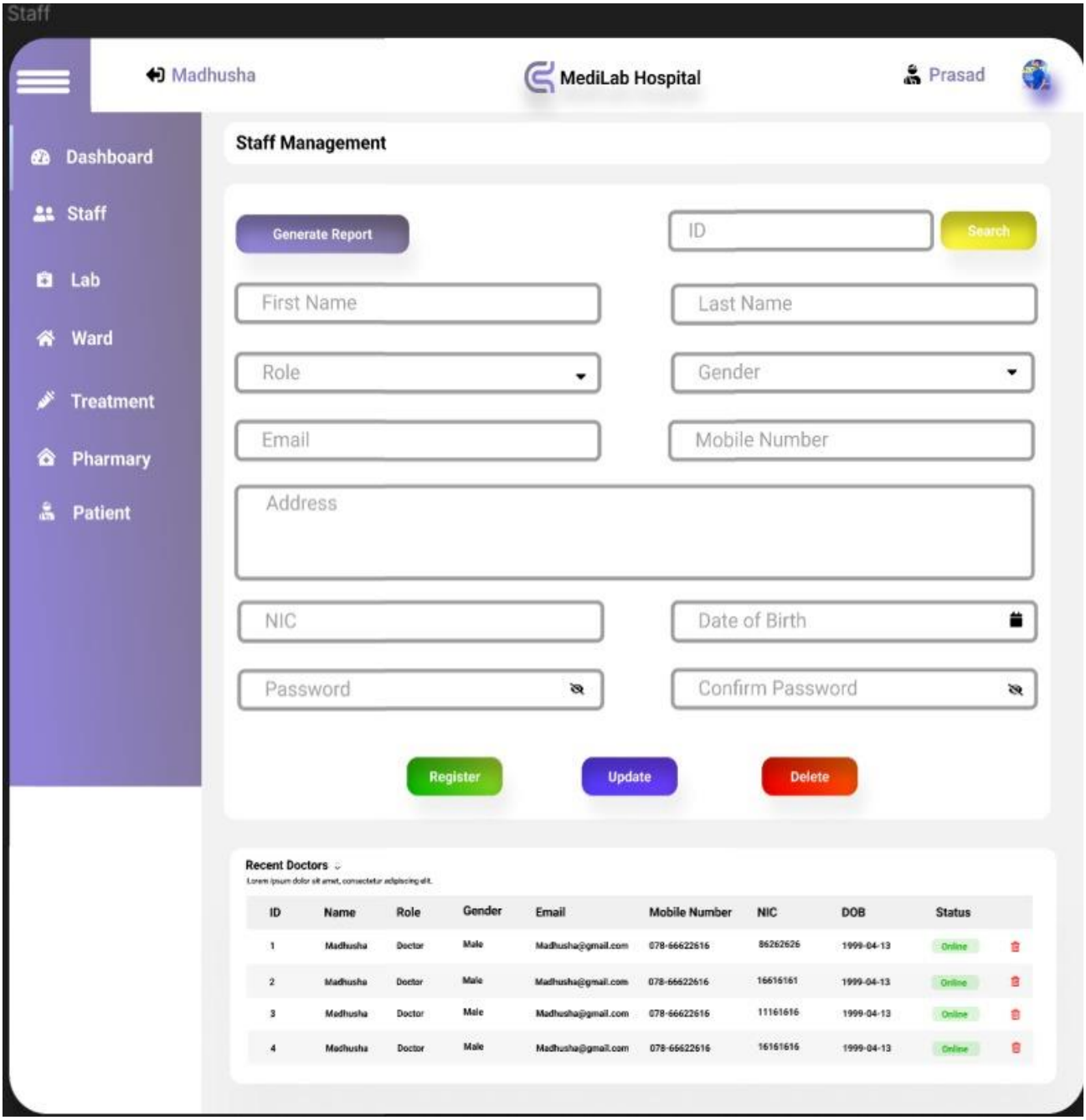


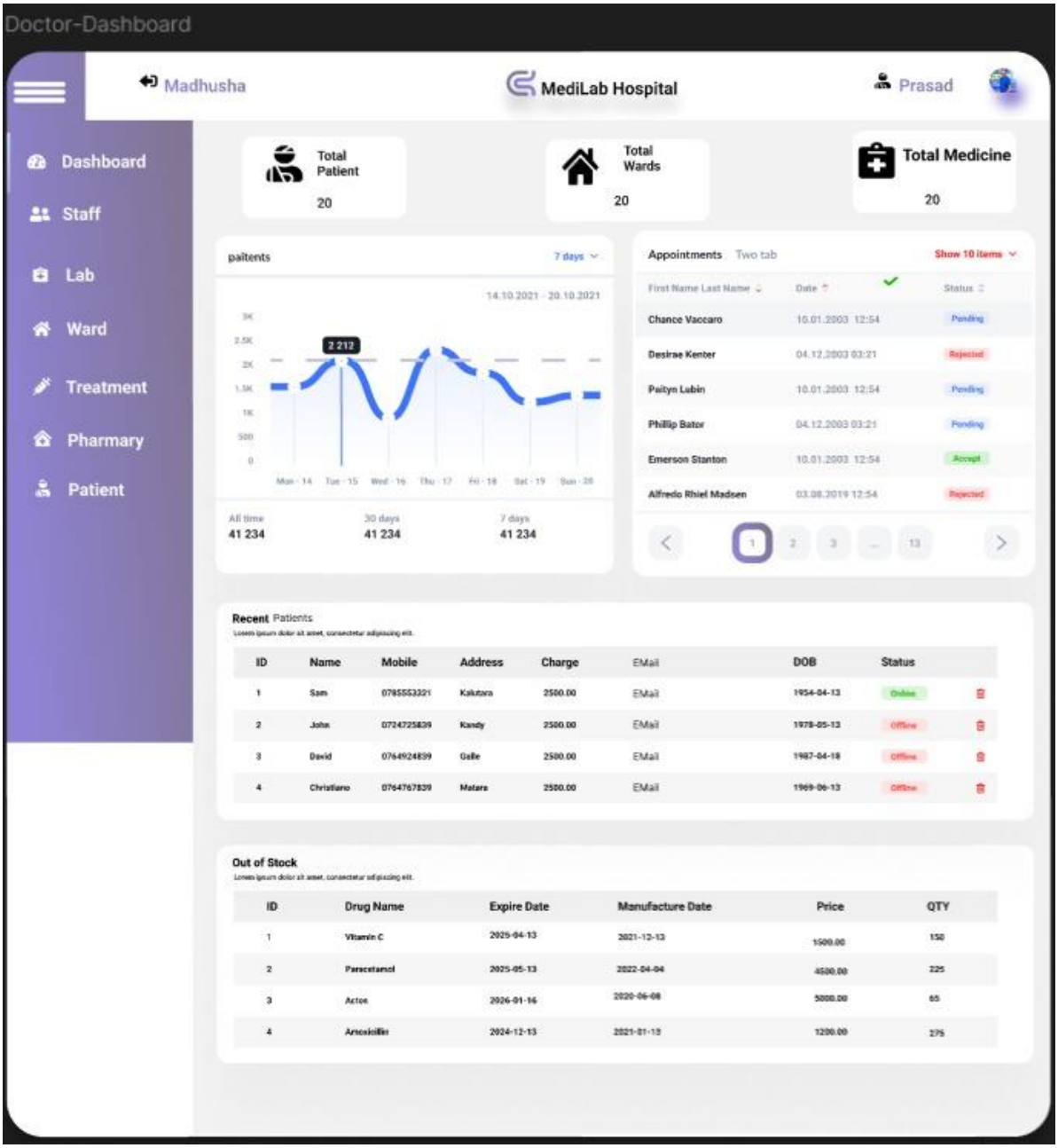
Lab Staff Dashboard

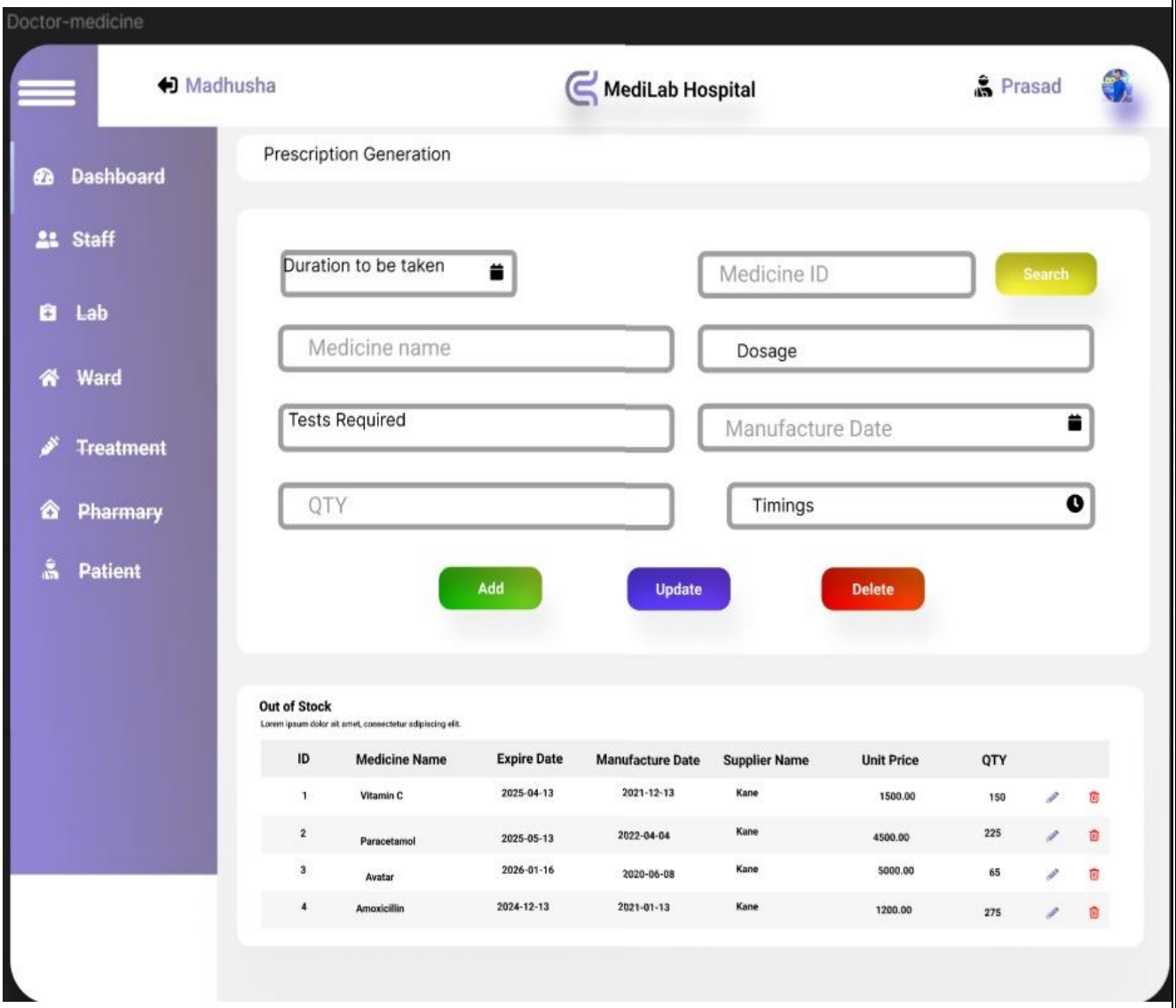


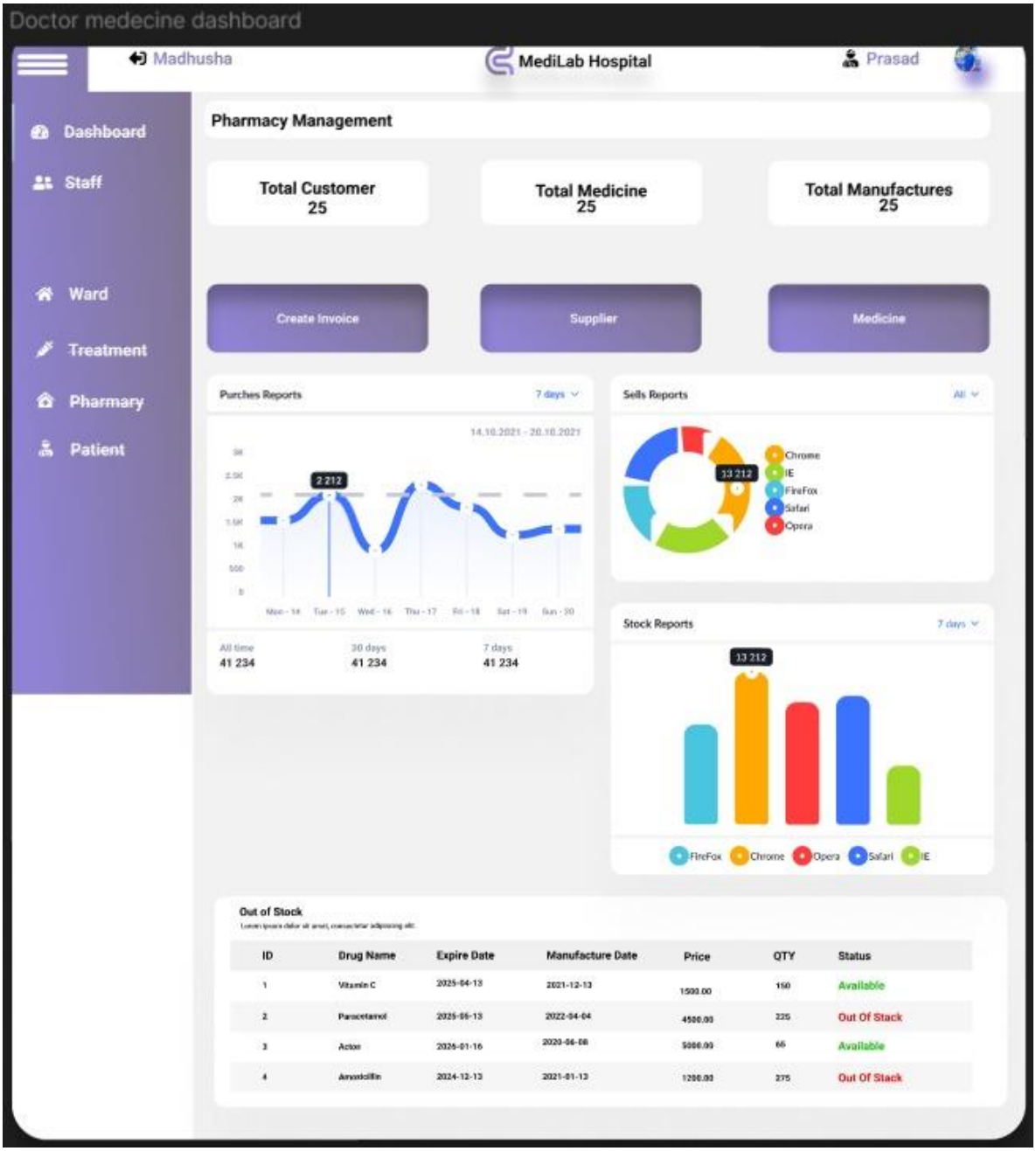
Admin Dashboard

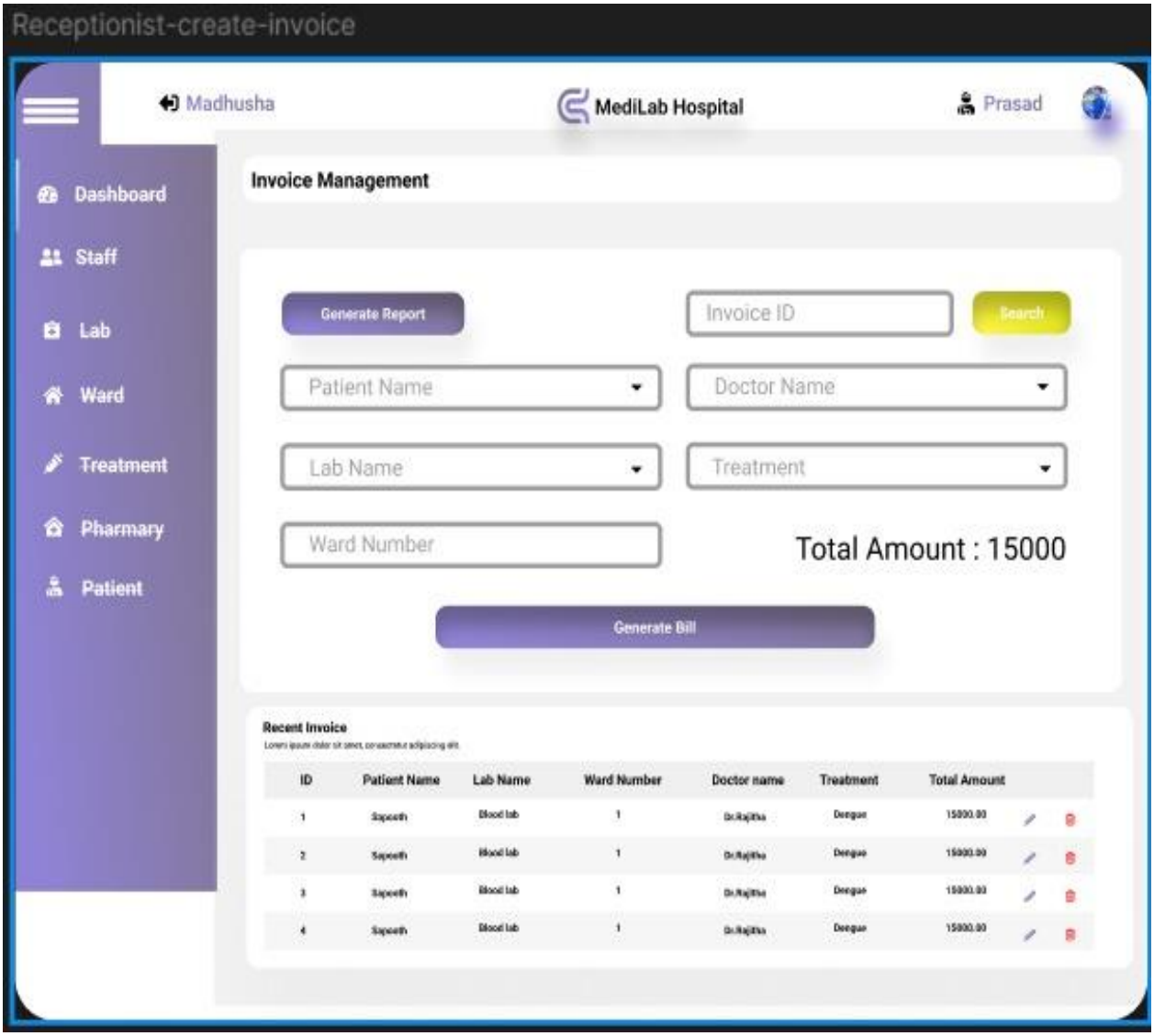


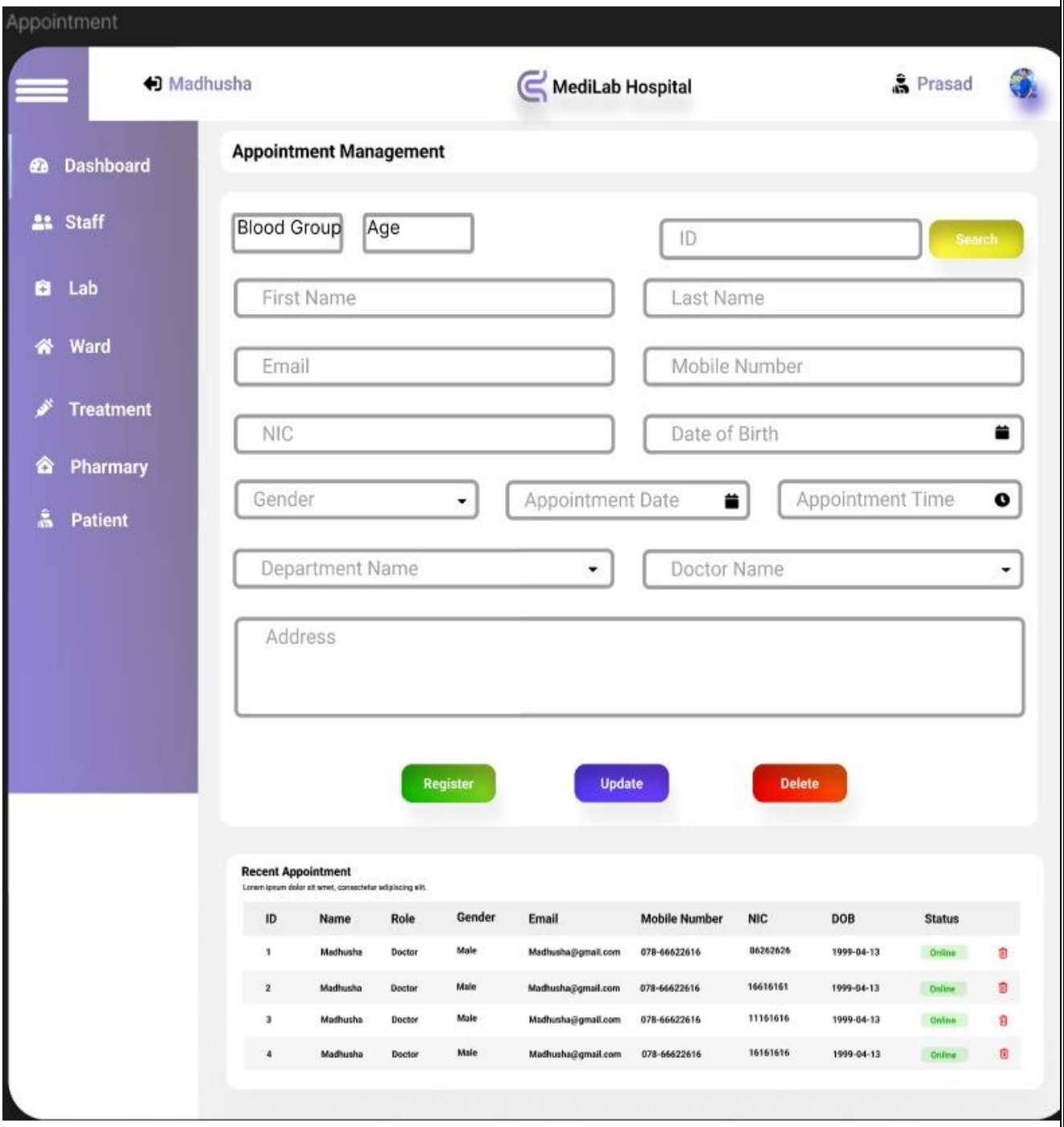

Staff
Madhusha
MediLab Hospital
Prasad
Dashboard
Staff
Lab
Ward
Treatment
Pharmary
Patient
Staff Management
Generate Report
ID
Search
First Name
Last Name
Role
Gender
Email
Mobile Number
Address
NIC
Date of Birth
Password
Confirm Password
Register
Update
Delete
Recent Doctors
ID Name Role Gender Email Mobile Number NIC DOB Status
1 Madhusha Doctor Male Madhusha@gmail.com 078-66622616 86262626 1999-04-13 Online
2 Madhusha Doctor Male Madhusha@gmail.com 078-66622616 16616161 1999-04-13 Online
3 Madhusha Doctor Male Madhusha@gmail.com 078-66622616 11161616 1999-04-13 Online
4 Madhusha Doctor Male Madhusha@gmail.com 078-66622616 16161616 1999-04-13 Online

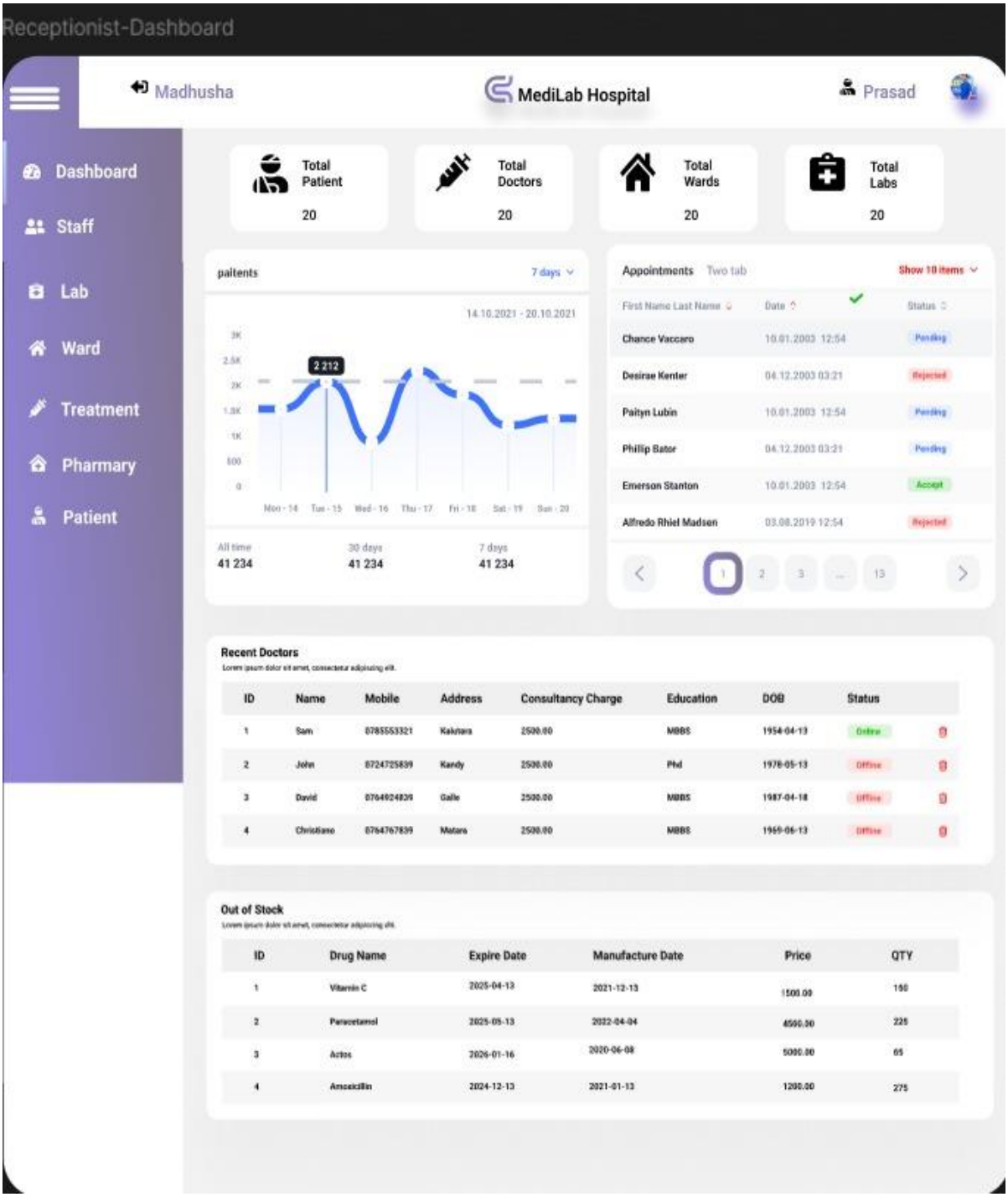

Doctor-Dashboard
Madhusha
MediLab Hospital
Prasad
Dashboard
Staff
Lab
Ward
Treatment
Pharmary
Patient
Total Patient
20
Total Wards
20
Total Medicine
20
paitents
7 days
14.10.2021 - 20.10.2021
2 212
All time
41 234
30 days
41 234
7 days
41 234
Appointments
Two tab
Show 10 items
First Name Last Name
Date
Status
Chance Vaccaro
10.01.2003 12:54
Pending
Desirae Kenter
04.12.2003 03:21
Rejected
Paityn Lubin
10.01.2003 12:54
Pending
Phillip Bator
04.12.2003 03:21
Pending
Emerson Stanton
10.01.2003 12:54
Accept
Alfredo Rhiel Madsen
03.08.2019 12:54
Rejected
Recent Patients
ID
Name
Mobile
Address
Charge
EMail
DOB
Status
1
Sam
0785553321
Kalutara
2500.00
EMail
1954-04-13
Online
2
John
0724725839
Kandy
2500.00
EMail
1978-05-13
Offline
3
David
0764924899
Galle
2500.00
EMail
1987-04-18
Offline
4
Christiano
0764767829
Matara
2500.00
EMail
1969-06-13
Offline
Out of Stock
ID
Drug Name
Expire Date
Manufacture Date
Price
QTY
1
Vitamin C
2025-04-13
2021-12-13
1500.00
150
2
Paracetamol
2025-05-13
2022-04-04
4500.00
225
3
Actos
2026-01-16
2020-06-08
5000.00
65
4
Amoxicillin
2024-12-13
2021-01-13
1200.00
275

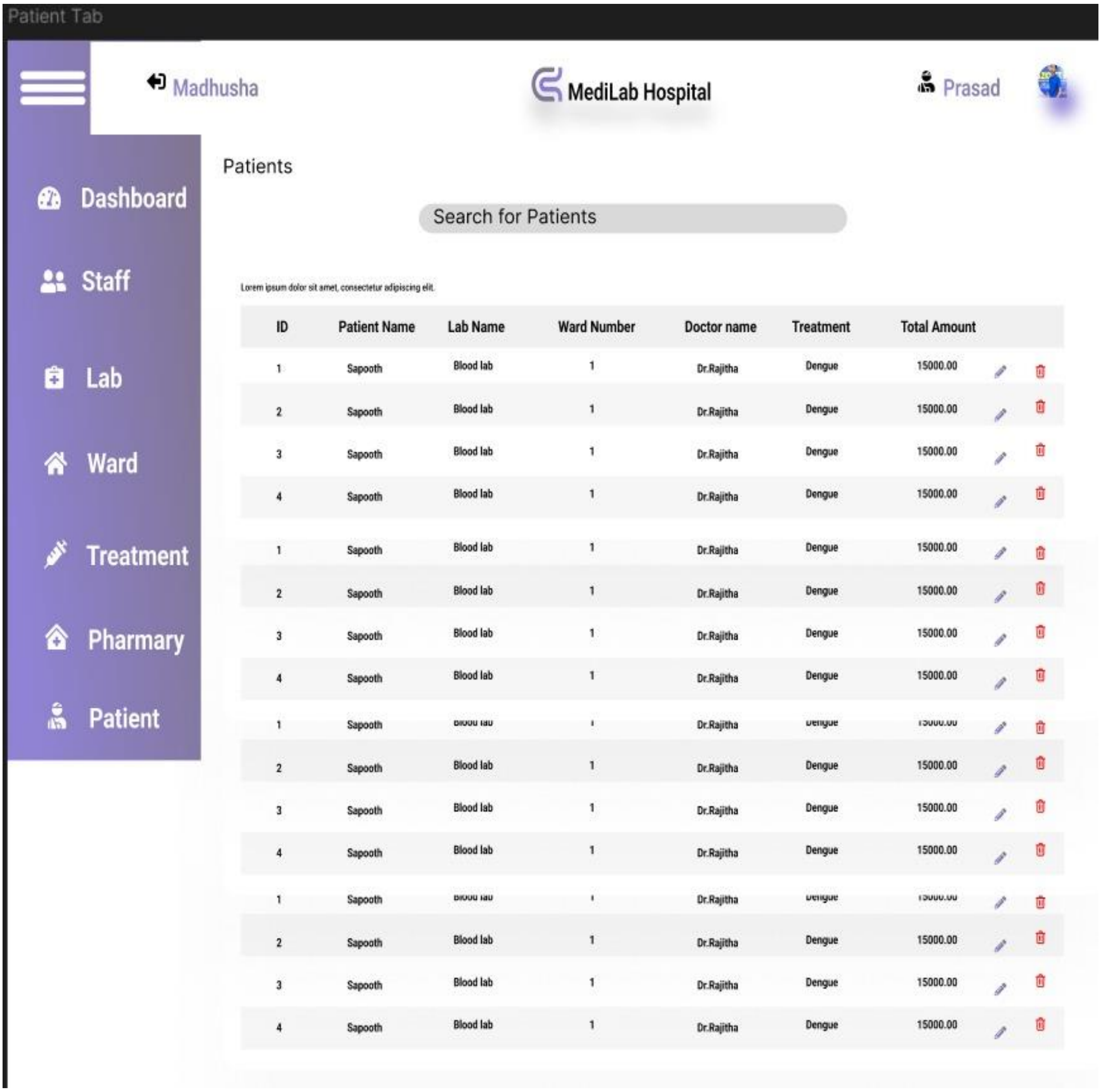

Doctor-medicine
Madhusha
MediLab Hospital
Prasad
Dashboard
Staff
Lab
Ward
Treatment
Pharmary
Patient
Prescription Generation
Duration to be taken
Medicine ID
Search
Medicine name
Dosage
Tests Required
Manufacture Date
QTY
Timings
Add
Update
Delete
Out of Stock
Lorem ipsum dolor sit amet, consectetur adipiscing elit.
ID
Medicine Name
Expire Date
Manufacture Date
Supplier Name
Unit Price
QTY
1
Vitamin C
2025-04-13
2021-12-13
Kane
1500.00
150
2
Paracetamol
2025-05-13
2022-04-04
Kane
4500.00
225
3
Avatar
2026-01-16
2020-06-08
Kane
5000.00
65
4
Amoxicillin
2024-12-13
2021-01-13
Kane
1200.00
275


Doctor medecine dashboard
Madhusha
MediLab Hospital
Prasad
Dashboard
Staff
Ward
Treatment
Pharmary
Patient
Pharmacy Management
Total Customer
25
Total Medicine
25
Total Manufactures
25
Create Invoice
Supplier
Medicine
Purches Reports
7 days
2 212
All time
41 234
30 days
41 234
7 days
41 234
Sells Reports
All
33 212
Chrome
IE
FireFox
Safari
Opera
Stock Reports
7 days
33 212
FireFox
Chrome
Opera
Safari
IE
Out of Stock
ID
Drug Name
Expire Date
Manufacture Date
Price
QTY
Status
1
Vitamin C
2025-04-13
2021-12-13
1500.00
150
Available
2
Paracetamol
2025-05-13
2022-04-04
4500.00
225
Out Of Stack
3
Acton
2026-01-16
2020-06-08
5000.00
65
Available
4
Amoxicillin
2024-12-13
2021-01-13
1200.00
275
Out Of Stack


Receptionist-create-invoice
Madhusha
MediLab Hospital
Prasad
Dashboard
Staff
Lab
Ward
Treatment
Pharmary
Patient
Invoice Management
Generate Report
Invoice ID
Search
Patient Name
Doctor Name
Lab Name
Treatment
Ward Number
Total Amount : 15000
Generate Bill
Recent Invoice
ID
Patient Name
Lab Name
Ward Number
Doctor name
Treatment
Total Amount


Appointment
Madhusha
MediLab Hospital
Prasad
Dashboard
Staff
Lab
Ward
Treatment
Pharmary
Patient
Appointment Management
Blood Group
Age
ID
Search
First Name
Last Name
Email
Mobile Number
NIC
Date of Birth
Gender
Appointment Date
Appointment Time
Department Name
Doctor Name
Address
Register
Update
Delete
Recent Appointment
ID Name Role Gender Email Mobile Number NIC DOB Status
1 Madhusha Doctor Male Madhusha@gmail.com 078-66622616 86262626 1999-04-13 Online
2 Madhusha Doctor Male Madhusha@gmail.com 078-66622616 16616161 1999-04-13 Online
3 Madhusha Doctor Male Madhusha@gmail.com 078-66622616 11161616 1999-04-13 Online
4 Madhusha Doctor Male Madhusha@gmail.com 078-66622616 16161616 1999-04-13 Online


Receptionist-Dashboard
Madhusha
MediLab Hospital
Prasad
Dashboard
Staff
Lab
Ward
Treatment
Pharmary
Patient
Total Patient
20
Total Doctors
20
Total Wards
20
Total Labs
20
paitents
7 days
14.10.2021 - 20.10.2021
2 212
All time
41 234
30 days
41 234
7 days
41 234
Appointments
Two tab
Show 10 items
First Name Last Name
Date
Status
Chance Vaccaro
10.01.2003 12:54
Pending
Desirae Kenter
04.12.2003 03:21
Rejected
Paityn Lubin
10.01.2003 12:54
Pending
Phillip Bator
04.12.2003 03:21
Pending
Emerson Stanton
10.01.2003 12:54
Accept
Alfredo Rhiel Madsen
03.08.2019 12:54
Rejected
Recent Doctors
ID
Name
Mobile
Address
Consultancy Charge
Education
DOB
Status
1
Sam
0785553321
Kalutara
2500.00
MBBS
1954-04-13
2
John
0724725839
Kandy
2500.00
Phd
1978-05-13
3
David
0764924829
Galle
2500.00
MBBS
1987-04-18
4
Christiano
0764767839
Matara
2500.00
MBBS
1969-06-13
Out of Stock
ID
Drug Name
Expire Date
Manufacture Date
Price
QTY
1
Vitamin C
2025-04-13
2021-12-13
1500.00
150
2
Paracetamol
2025-05-13
2022-04-04
4500.00
225
3
Actos
2026-01-16
2020-06-08
5000.00
65
4
Amoxicillin
2024-12-13
2021-01-13
1200.00
275



Patient Tab
Madhusha
MediLab Hospital
Prasad
Dashboard
Staff
Lab
Ward
Treatment
Pharmary
Patient
Patients
Search for Patients
Lorem ipsum dolor sit amet, consectetur adipiscing elit.
ID Patient Name Lab Name Ward Number Doctor name Treatment Total Amount
1 Sapooth Blood lab 1 Dr.Rajitha Dengue 15000.00
2 Sapooth Blood lab 1 Dr.Rajitha Dengue 15000.00
3 Sapooth Blood lab 1 Dr.Rajitha Dengue 15000.00
4 Sapooth Blood lab 1 Dr.Rajitha Dengue 15000.00

3.2 Functional Requirements

3.2.1 Managing Patient Records

Purpose: The system will store, manage, and allow retrieval of patient data such as personal details, medical history, and treatment records.

- **Pre-condition:** The patient is registered in the system.
- **Input:** Patient's personal details (name, address, contact number), medical history, current medications, allergies, and recent treatments.
- **Process:** The system stores the patient's information securely in a database and allows authorized personnel to view and update the records as needed.
- **Output:** A profile of the patient with personal and medical information.
- **Post-condition:** Patient data is updated in the system for future reference and treatment.

3.2.2 Scheduling Appointments

Purpose: Allows patients to schedule appointments with doctors based on availability.

- **Pre-condition:** The patient is registered in the system, and the doctor has available time slots.
- **Input:** Patient's details, doctor's name, preferred time, and date of appointment.
- **Process:** The system verifies the doctor's availability, checks for time conflicts, and schedules the appointment. Confirmation of the appointment is sent to the patient.
- **Output:** A confirmation message of the scheduled appointment.
- **Post-condition:** Appointment is saved in the doctor's and patient's calendar, and both are notified.

3.2.3 Generating Prescriptions

Purpose: Enables doctors to generate digital prescriptions for patients.

- **Pre-condition:** The patient has had a consultation with a doctor and requires medication.
- **Input:** Patient's details, medical condition, prescribed medication, dosage, and instructions.
- **Process:** The system generates a prescription containing necessary details such as medications, dosage, and administration instructions, and stores it in the patient's record.
- **Output:** A digital prescription document.
- **Post-condition:** The prescription is stored in the system, accessible by the doctor, patient, or pharmacy.

3.2.4 Viewing Patient History

Purpose: Allows doctors and authorized staff to view a patient's medical history.

- **Pre-condition:** The patient is registered, and the system has stored their previous medical records.
- **Input:** Patient's ID or name.
- **Process:** The system retrieves and displays the patient's historical data (previous treatments, diagnoses, prescriptions).
- **Output:** A full medical history of the patient.
- **Post-condition:** The patient's history is available for review by authorized medical professionals.

3.2.5 Managing Doctor Schedules

Purpose: Allows administrators to manage doctors' schedules, including availability, shifts, and off-days.

- **Pre-condition:** The doctor is registered in the system.
- **Input:** Doctor's working hours, special availability, vacation dates, and other shift-related data.
- **Process:** The system updates and tracks the doctor's availability and manages conflicts in appointments.
- **Output:** A schedule view of each doctor's available time slots.
- **Post-condition:** The doctor's schedule is updated in the system and is available for appointment booking.

3.2.6 Patient Registration

Purpose: Registers new patients into the system.

- **Pre-condition:** The patient is not already registered in the system.
- **Input:** Patient's personal information (name, address, phone number, emergency contact, etc.), insurance details (if applicable).
- **Process:** The system creates a unique profile for the patient, stores their details, and assigns them an ID for future reference.
- **Output:** A registered patient profile.
- **Post-condition:** The patient's data is saved in the system and available for future interactions.

3.2.7 Handling Medical Billing

Purpose: Manages billing for medical services, prescriptions, and consultations.

- **Pre-condition:** A patient has received treatment or consultation, and an invoice is required.
- **Input:** Service type, consultation fees, medication charges, insurance coverage (if applicable).
- **Process:** The system calculates the total amount to be paid and generates an invoice based on the input details.
- **Output:** A detailed invoice for the patient.
- **Post-condition:** The invoice is stored in the system, and payment details can be updated once settled.

3.2.8 Reporting and Analytics

Purpose: Generates reports on patient visits, treatment history, billing, and system performance.

- **Pre-condition:** The system has accumulated sufficient data for reporting.
- **Input:** Time period, report type (e.g., patient visits, doctor performance, billing).
- **Process:** The system extracts and processes data from the database to create comprehensive reports.
- **Output:** Reports in printable formats (PDF, Excel, etc.).
- **Post-condition:** Reports are saved in the system or printed for physical distribution.

3.2.9 Notifications and Alerts

Purpose: Notifies patients, doctors, and administrators about important events such as appointment reminders or prescription refills.

- **Pre-condition:** The patient or doctor has a scheduled event or alert set up.
- **Input:** Event details (appointment time, medication schedule).
- **Process:** The system sends out notifications via email, SMS, or in-app notifications to remind users of upcoming events or deadlines.
- **Output:** Notification sent to the relevant parties.
- **Post-condition:** Alerts are marked as acknowledged in the system.

3.2.10 Managing Medical Inventory

Purpose: Tracks and manages medical supplies and inventory.

- **Pre-condition:** The clinic has a registered list of medical supplies.
- **Input:** Inventory item details (name, quantity, expiration dates).
- **Process:** The system tracks the stock levels and updates the inventory with new purchases or usage.
- **Output:** Updated inventory list.
- **Post-condition:** The inventory is updated to reflect the current stock.

3.3 Non-Functional Requirements

3.3.1 Static Non-Functional Requirements

- **Usability:** The system must be easy to navigate, with an intuitive design that caters to both medical professionals and patients.
- **Performance:** The system should handle up to 500 concurrent users without performance degradation. Each operation (e.g., searching records, scheduling) should respond within 3 seconds.
- **Security:** The system will implement user authentication, role-based access control, and encryption for sensitive data.

3.3.2 Dynamic Non-Functional Requirements

- **Scalability:** The system must be able to scale to accommodate more users, patients, and appointments over time.
- **Availability:** The system will have 99.9% uptime, with scheduled maintenance windows.
- **Backup and Recovery:** Daily backups will be performed, with the ability to restore data within 30 minutes in case of a system failure.

3.4 Design Constraints

- **Security and Compliance:** The system must comply with medical data protection regulations (e.g., HIPAA) to ensure patient data privacy and security.
- **Hardware:** The system will be compatible with commonly used hardware (desktops, tablets, smartphones) and operate on standard web browsers.
- **Software:** The system will be developed using modern web technologies, including HTML5, CSS3, JavaScript, and PHP, and will run on a secure cloud infrastructure.

Hardware Requirements:

1. **Processor**: Minimum Intel Core i3 or equivalent (2.0 GHz or higher)
2. **RAM**: 4 GB or more
3. **Hard Disk**: 500 GB or more (SSD preferred for faster performance)
4. **Display**: 15.6-inch screen with 1366x768 resolution or higher
5. **Network**: Stable internet connection for online updates, communication, and backups
6. **Peripherals**:
    - Printer for report generation
    - Barcode scanner (optional, if inventory management is included)

Software Requirements:

1. **Operating System**:
    - Windows 10/11 (64-bit)
    - Linux (Ubuntu or similar distributions for development environment)
2. **Database**: MySQL or PostgreSQL (for storing clinic data)
3. **Development Environment**:
    - IDE: Visual Studio Code, Eclipse, or IntelliJ IDEA

    - Programming Languages: Java, Python (or any relevant language used for the project)
    - Framework: Spring Boot (Java), Django (Python), or similar for backend development
4. **Web Server**: Apache Tomcat or Nginx (if web-based access is part of the system)
5. **Version Control**: Git for source code management and collaboration (GitHub or GitLab)
6. **Miscellaneous**:
    - Microsoft Office or Google Docs (for documentation)
    - Browser (Google Chrome, Mozilla Firefox) for testing and UI checks

Tools Used – draw.io, Gantt Project software, Figma