

**SOFTWARE**  
**ENGINEERING CSE**  
**1005**  
**LAB – L20 +L21**  
**Hospital Management System**



**BATCH-8**

**Submitted by**

V Monish	22BCE8606
Ragav varma	22BCE8598
Brahmajna Maddukuri	22BCE8584
D.R.V.Amarnath	22BCE8654
Sk.M.Subani	22BCE9845

**Faculty**

**Prof. Shaik Kareemulla**

# PROBLEM STATEMENT:

In this busy world we don't have the time to wait in infamously long hospital queues. The problem is, queuing at hospital is often managed manually by administrative staff, then take a token there and then wait for our turn then ask for the doctor and the most frustrating thing - we went there by traveling a long distance and then we come to know the doctor is on leave or the doctor can't take appointments.

HMS will help us overcome all these problems because now patients can book their appointments at home, they can check whether the doctor they want to meet is available or not. Doctors can also confirm or decline appointments, this help both patient and the doctor because if the doctor declines' appointment, then patient will know this in advance and patient will visit hospital only when the doctor confirms' the

appointment this will save time and money of the patient.

Patients can also pay the doctor's consultant fee online to save their time.

## KEY FEATURES:

### PATIENT:

Patients can choose the best preferred appointments from the options provided and can also change the appointment schedule or cancel it. After appt. is confirmed by the respective doctor they can pay their consultant fee online. Patients have access to only their records.

Key functions:

- Make appointment.
- Cancel appointment.
- Update Details.
- Payment.
- View Payment History.

### DOCTOR:

Doctors can view the patient appointment list and provide the confirmation or make changes in the appointment list if required. Doctors have access to only records of those patients whom they are treating.

Key functions:

- Confirmation of appointment.
- Cancellation of appointment.
- Modification of appointment list.
- Add Prescription.

# SRS DOCUMENT

## Table of Contents

1. Introduction
  - 1.1. Purpose
  - 1.2. Scope
  - 1.3. Definitions, Acronyms, and Abbreviations
  - 1.4. References
  - 1.5. Overview
2. Overall Description
  - 2.1. Product Perspective
  - 2.2. Product Functions
  - 2.3. Design and Implementation constraints
  - 2.4. Operating Environment
3. Specific Requirements
  - 3.1. Performance Requirements
  - 3.2. Functional Requirements
  - 3.3. Non-Functional Requirements
  - 3.4. Other Requirements
4. Appendices
  - 4.1 References

# 1. INTRODUCTION

## 1.1. Purpose

This software will help the company to be more efficient in registration of their patients and manage appointments, records of patients. It enables doctors and admin to view and modify appointments schedules if required. The purpose of this project is to computerize all details regarding patient details and hospital details.

## 1.2. Scope

The system will be used as the application that serves hospitals, clinic, dispensaries or other health institutions. The intention of the system is to increase the number of patients that can be treated and managed properly.

If the hospital management system is file based, management of the hospital has to put much effort on securing the files. They can be easily damaged by fire, insects and natural disasters. Also could be misplaced by losing data and information.

## 1.3. Definitions, Acronyms, and Abbreviations

- SRS: Software Requirement Specification.
- DFD: Data Flow Diagram.
- ENT- Ear, Nose and Throat Specialist.
- BG - Blood group

## 1.4. References

P. Jalote, an Integrated Approach to Software Engineering, Narosa publication house, Edition -3 (2011).

## 1.5. Overview

Our application contains two modules – the admin module and the user module. Our application will not only help the admin to preview the monthly and/or yearly data but it will also allow them to edit, add or update records. The software will also help the admin to monitor the transactions made by the patients and generate confirmations for the same. The admin will be able to manage and update information

The user module can be accessed by both the doctors and the patients. The doctor can confirm and/or cancel appointments. The doctors can even add prescriptions for their patients using our application. The patients will be able to apply for the appointment and make transaction for the same, and can even cancel appointments with the doctors.

## 2. OVERALL DESCRIPTION

### 2.1. Product Perspective

This Hospital Patient Info Management System is a self-contained system that manages activities of the hospital.

Due to improperly managed details medical center faces quite a lot of difficulties in accessing past data as well as managing present data. The fully functional automated hospital management system which will be developed through this project will eliminate the disadvantages caused by the manual system by improving the reliability, efficiency and performance. The usage of a database to store patient, employee, stock details etc. will accommodate easy access, retrieval, and search and manipulation of data. The access limitations provided through access privilege levels will enhance the security of the system. The system will facilitate concurrent access and convenient management of activities of the medical center.

### 2.2. Product Functions

- Provide access to registered users only.
- Registration of new patients.
- Enable patient to view their record.
- Enable patient to update their record.
- Modification in schedule by patient.

### 2.3. Design and Implementation constraints

Design and implementation constraints refer to the limitations, restrictions, or requirements that must be considered during the system's design and development phases. These constraints often stem from technical, business, regulatory, or environmental factors.

#### **Regulatory Constraints**

- Compliance with Healthcare Standards: The system must comply with healthcare regulations and standards

#### **Technological Constraints**

- Platform Dependence: The HMS must run on specific operating systems and databases used by the hospital.
- Technology Stack: The system must use pre-approved programming languages and frameworks.

### 2.4. Operating Environment

The Hospital Management System (HMS) will be accessible via modern web browsers on both desktop and mobile devices, ensuring flexibility for medical staff and administrators. It will operate on a cloud-based server infrastructure to provide scalability, reliability, and secure access to patient data. The system must be compatible with major operating systems (e.g., Windows, macOS, Linux) and browsers (e.g., Chrome, Firefox, Safari, Edge) to support a diverse range of users, including healthcare professionals and administrative staff. Additionally, the system will include support for mobile devices (e.g., smartphones, tablets) to allow for real-time access to critical patient information and hospital resources from any location within the hospital network.

## **2.5. Assumptions and Dependencies**

- **Internet Connectivity:** Reliable internet access is essential for real-time updates and system functionality.
- **Initial Data:** Accurate initial data, including patient records, staff information, and inventory, is required for system setup.
- **Regulatory Compliance:** The system assumes adherence to healthcare regulations (e.g., HIPAA, GDPR).
- **Hardware Compatibility:** The system requires that hospital hardware meets necessary technical specifications.
- **Training and Support:** All users must receive adequate training, with ongoing technical support available.
- **User Compliance:** Users must follow data input standards and system protocols for optimal performance

## 3. SPECIFIC REQUIREMENTS

### 3.1. Performance Requirements

- Response time- The system will give responses within 1 second after checking the patient information and other information.
- Capacity- The system must support 1000 people at a time
- User interface- User interface screen will response within 5 seconds

### 3.2. Functional Requirements

- FR1: The system shall allow the registration of new patients and updating of existing patient information in real-time.
- FR2: The system shall facilitate the scheduling of patient appointments, including booking, modifying, and canceling appointments.
- FR3: The system shall store and manage detailed medical records for each patient, including diagnosis, treatment history, and prescribed medications.
- FR4: The system shall enable the management and assignment of medical staff shifts and duties.
- FR5: The system shall provide a patient portal for viewing personal medical records, upcoming appointments, and payment history.
- FR6: The system shall manage bed occupancy, including tracking available beds and patient assignments.
- FR7 : The system shall facilitate the management of laboratory test requests, results, and integration with external lab systems

### 3.3. Non-Functional Requirements

- NRF1: The system shall support up to 500 concurrent users.
- NRF2: The system shall ensure that page's load within 2 seconds for critical tasks (e.g., patient registration, emergency alerts).
- NRF3: The system shall ensure that database queries return results in under 5 seconds for patient data retrieval.
- NRF4: The system shall use strong, salted hashing for storing all passwords and sensitive information.
- NRF5: The system shall transmit all data via HTTPS and secure protocols to protect patient confidentiality.
- NRF6: The system shall implement role-based access control (RBAC) to ensure that only authorized personnel have access to specific functionalities (e.g., doctors, nurses, admin staff).
- NRF7: The system shall ensure 99.95% availability, particularly during peak hours and emergencies.
- NRF8: The system shall allow patient admission and discharge processes to be completed within 3 minutes.
- NRF9: The system shall provide inline help, tooltips, and context-sensitive assistance for ease of use by medical and administrative staff.
- NRF10: The system shall offer an API for future integrations with laboratory systems, pharmacy management systems, and third-party healthcare providers.
- NRF11: The system shall manage up to 10,000 patient records, including historical data and medical records.
- NRF12: The system shall support automatic backups and disaster recovery procedures to ensure data integrity.



### 3.4 Other Requirements

#### **SAFETY REQUIREMENTS:**

If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage and reconstructs a more current state by reapplying or redoing the operations of committed transactions from the backed-up log, up to the time of failure.

#### **SECURITY REQUIREMENTS:**

1. Want take the responsibility of failures due to hardware malfunctioning.
2. Warranty period of maintaining the software would be one year.
3. Additional payments will be analyzed and charged for further maintenance.
4. If any error occurs due to a user's improper use. Warranty will not be allocated to it.
5. No money back returns for the software.

## 4.1 REFERENCES

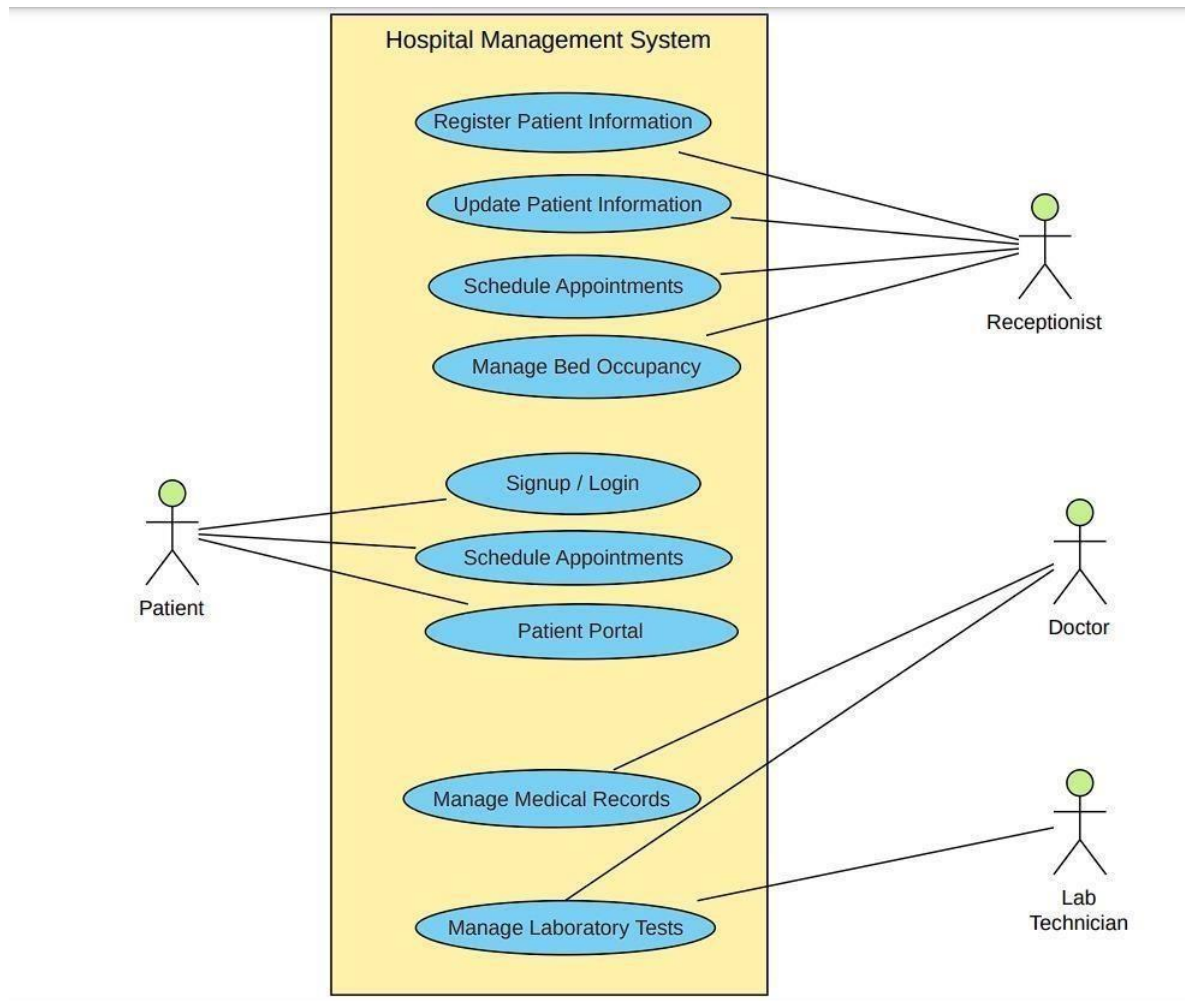
- IEEE Std 830-1998: IEEE Recommended Practice for Software Requirements Specifications.
- HIPAA: Health Insurance Portability and Accountability Act.
- GDPR: General Data Protection Regulation.
- HL7: Health Level Seven International standards for electronic health information exchange.

# SYSTEM MODELS:

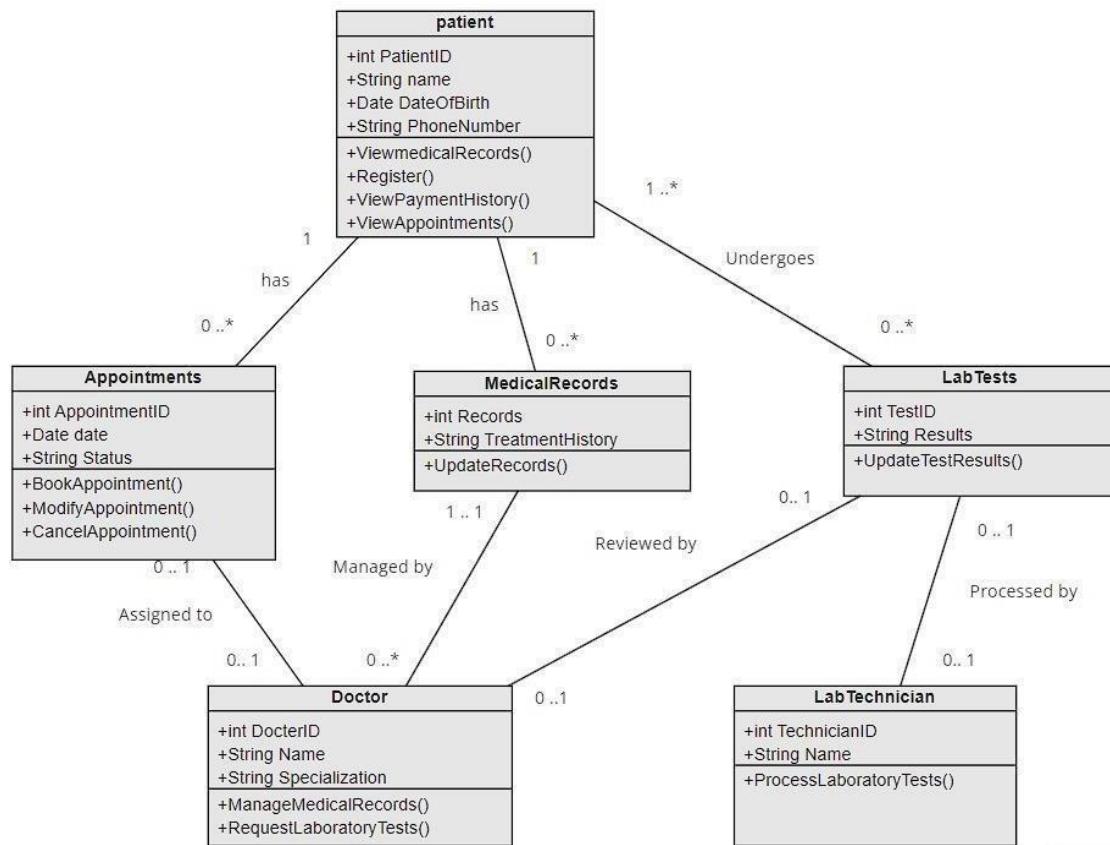
## 1.1. Use Case Diagram

## 1.2. Class Diagram

### USE CASE DIAGRAM:

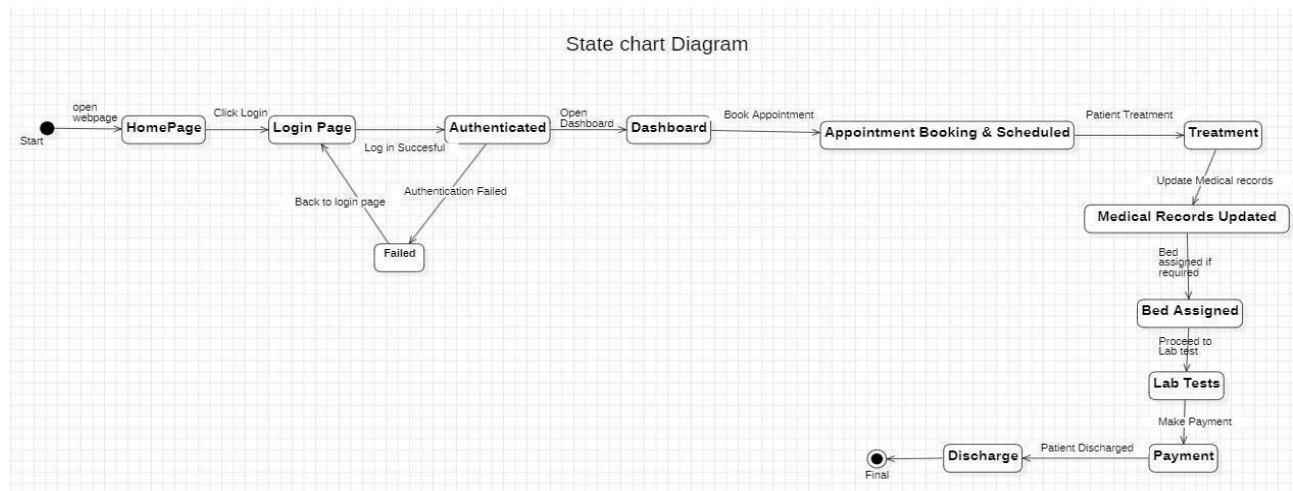


## CLASS DIAGRAM:

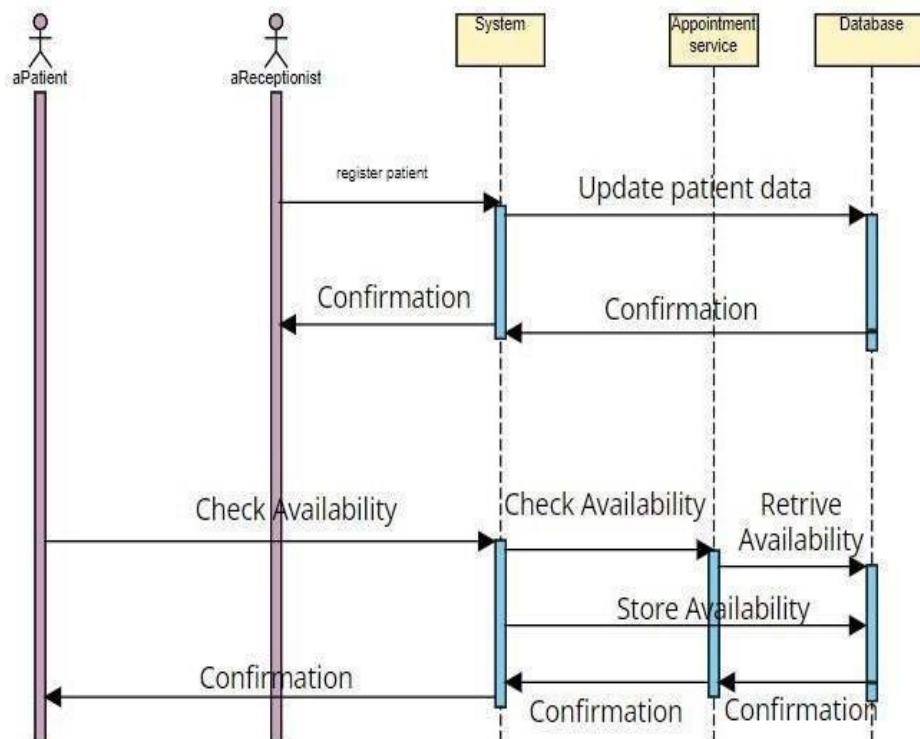


# UML DIAGRAMS:

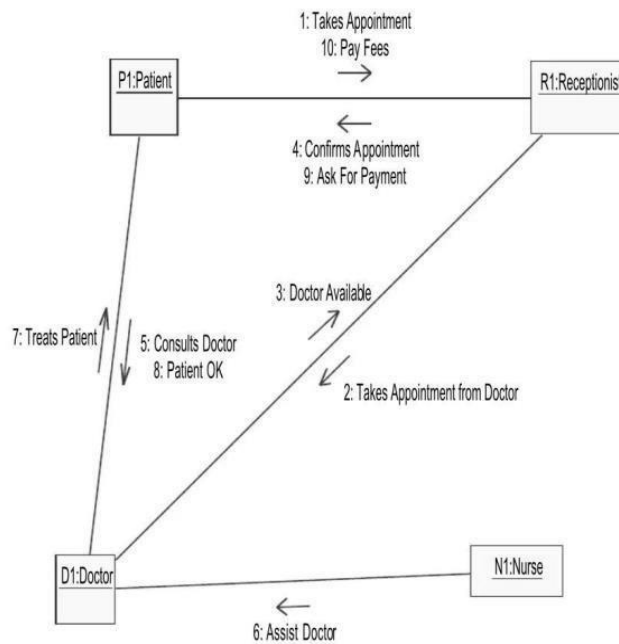
## STATE CHART DIAGRAM:



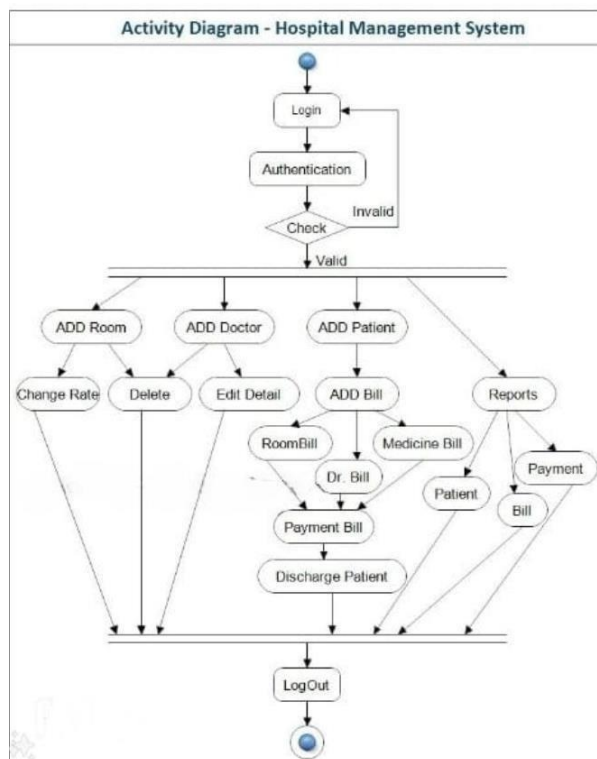
## SEQUENCE DIAGRAM



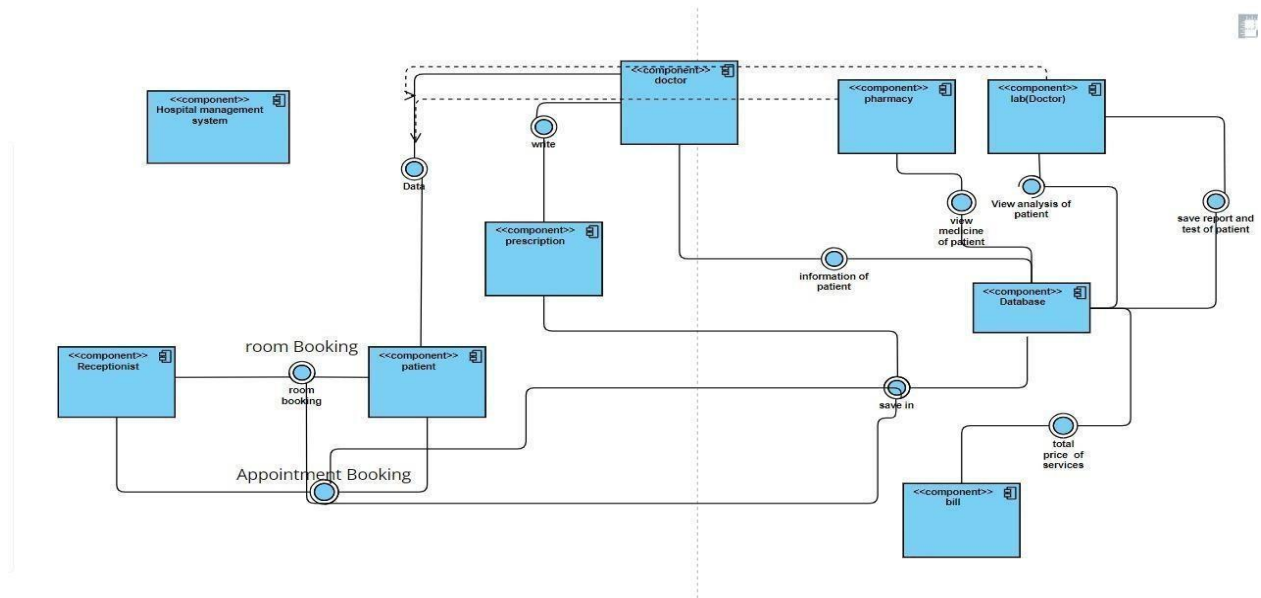
### Collaboration Diagram:



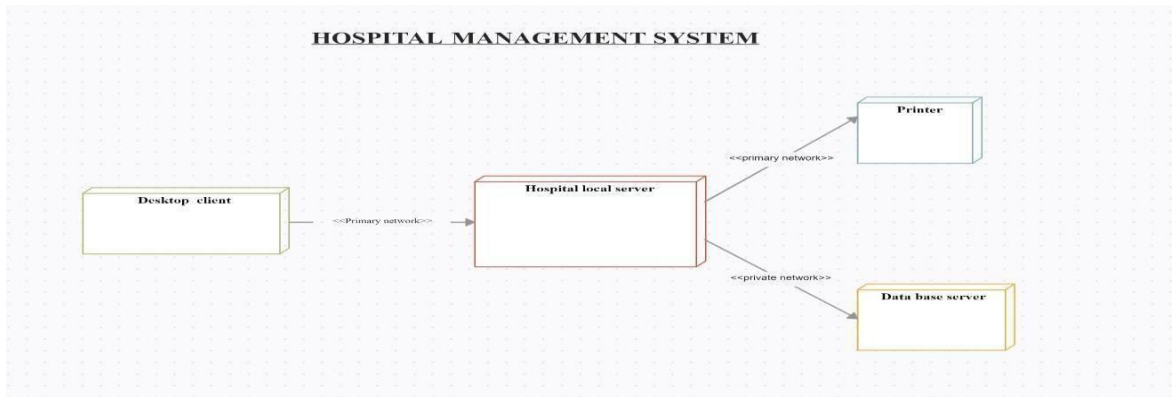
### ACTIVITY DIAGRAM:



## COMPONENT DIAGRAM:



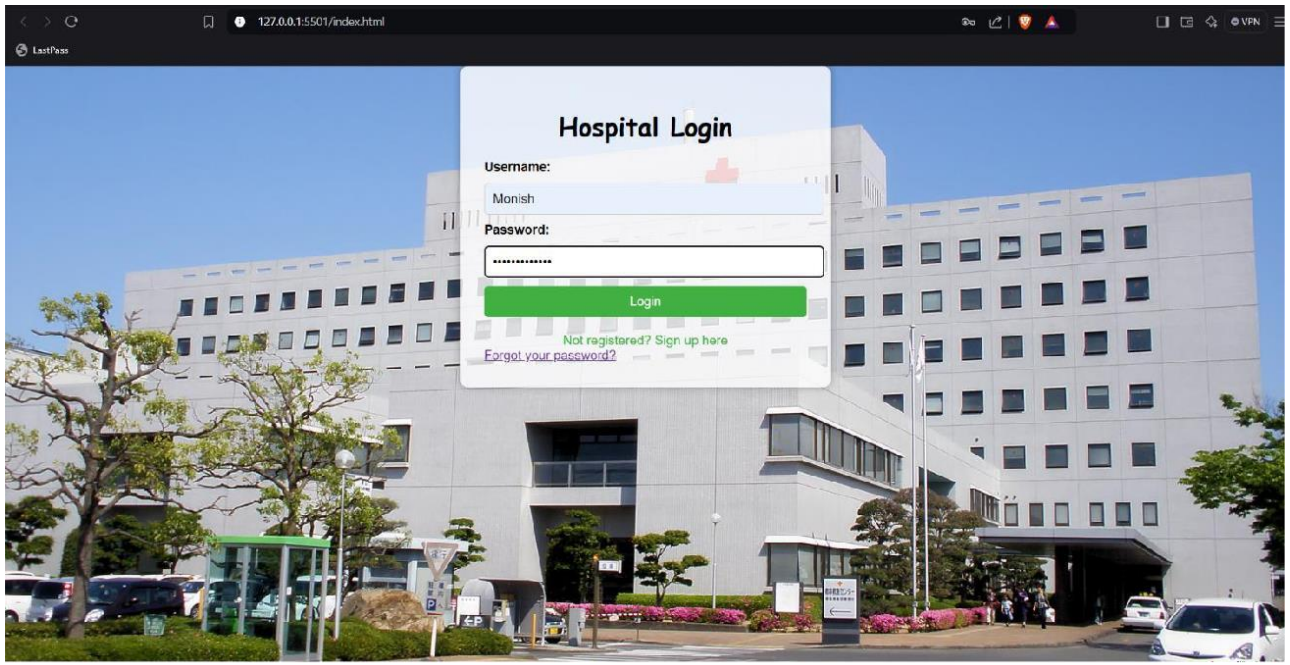
## DEPLOYMENT DIAGRAM:



# Coding of Project:

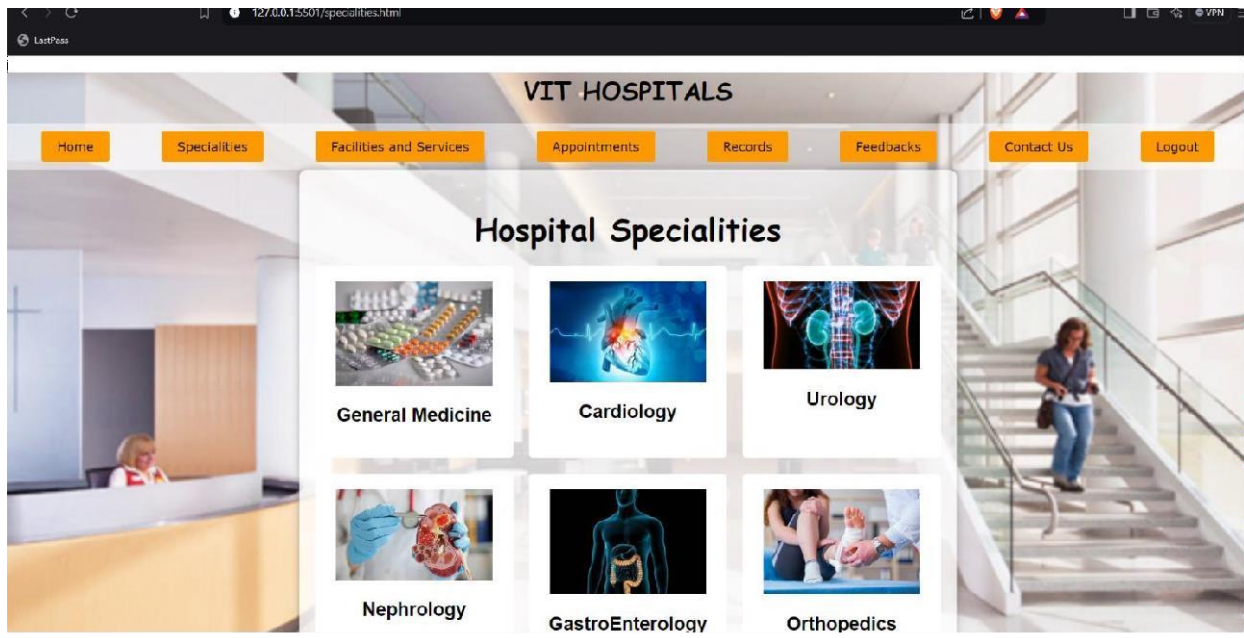
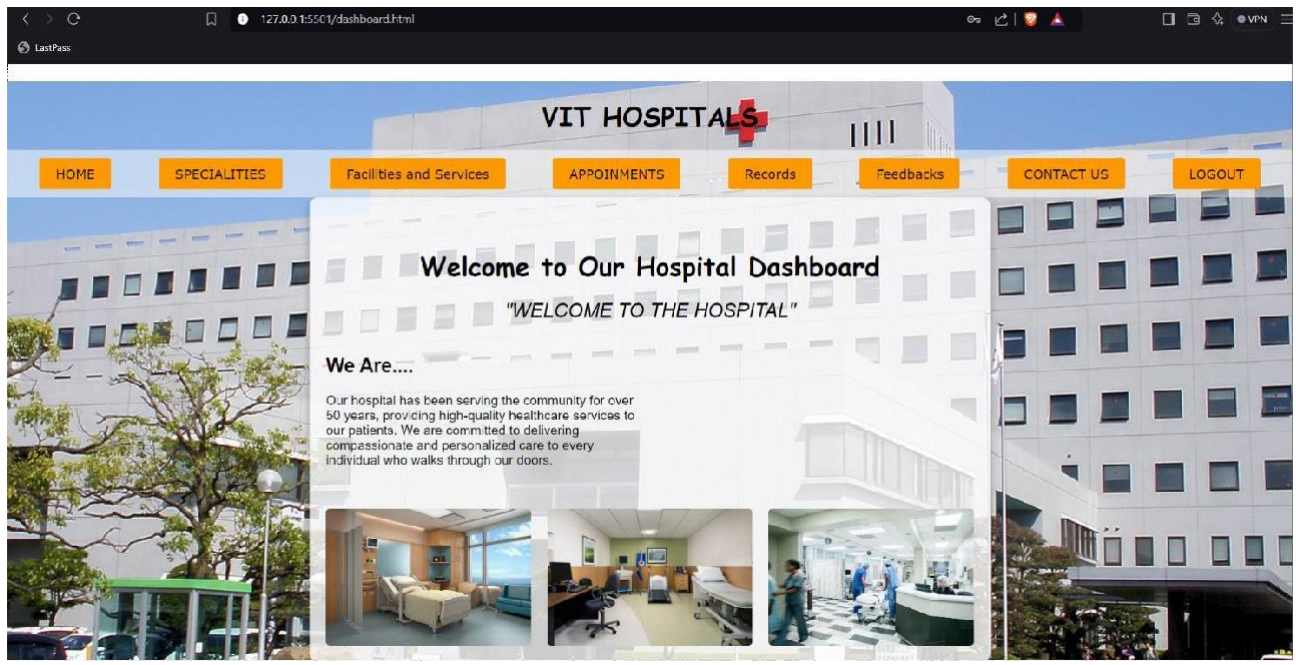
GitHub Link: [https://github.com/Monish36/Team\\_8\\_HMS](https://github.com/Monish36/Team_8_HMS)

## LOG-IN PAGE:

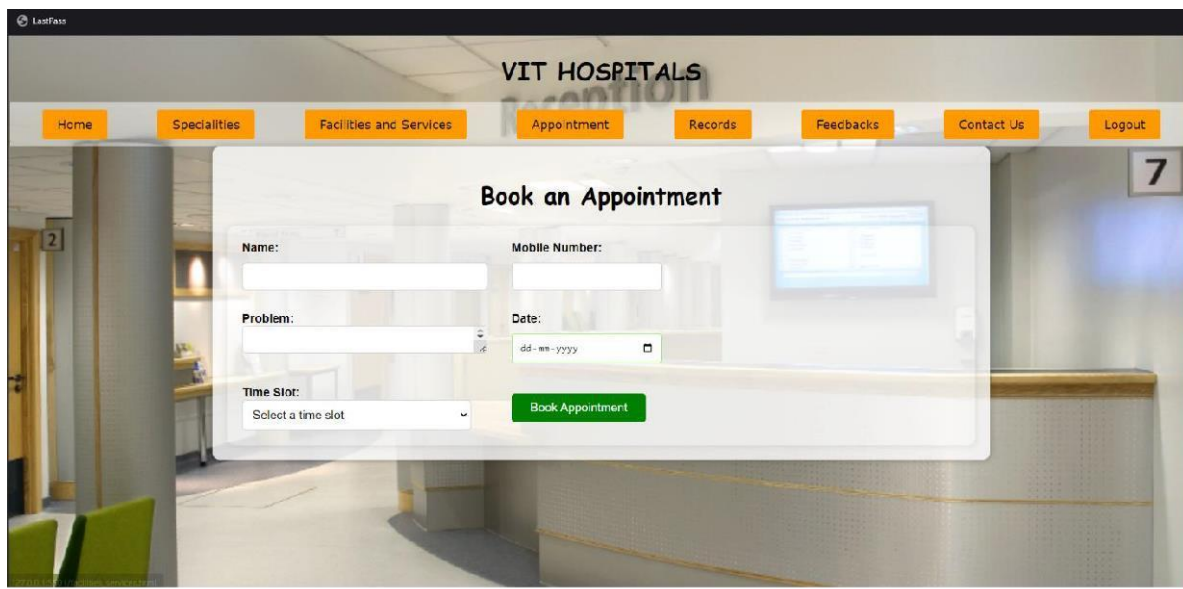
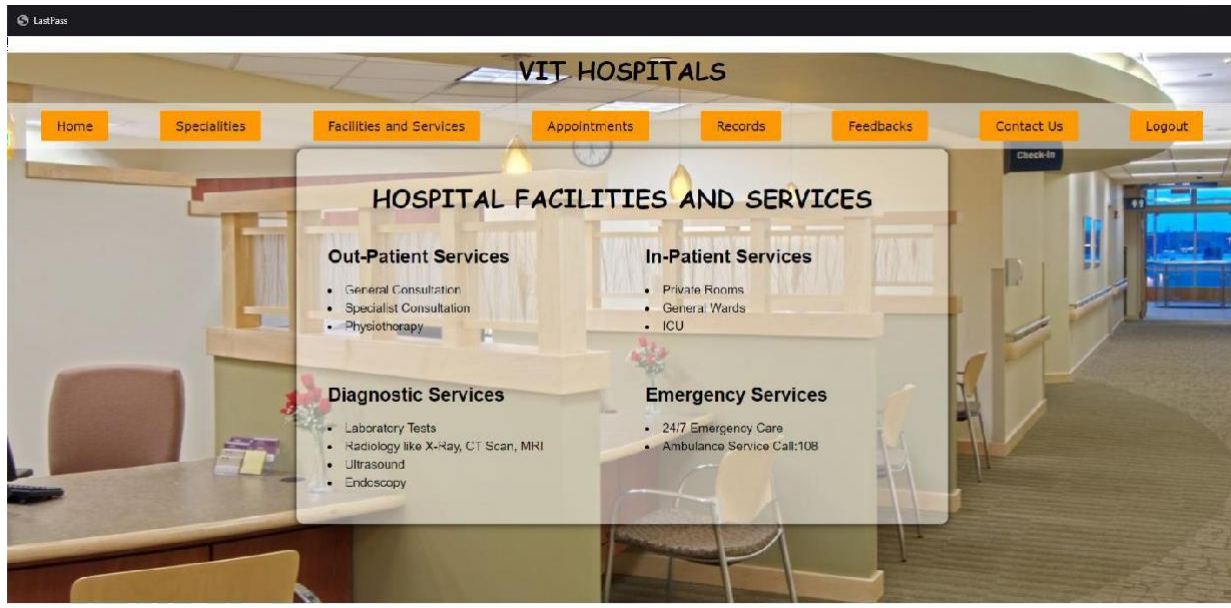




## DASHBOARD & HOME PAGE:



## FACILITIES & APPOINTMENT PAGE:



## RECORDS & FEEDBACK PAGE:

VIT HOSPITALS

Home Specialities Facilities and Services Appointment Records Feedbacks Contact Us Logout

### Medical Records

Name:

Medical Condition:

Doctor:

Medicines Prescribed:

Date:

VIT HOSPITALS

Home Specialities Facilities and Services Appointment Records Feedbacks Contact Us Logout

### Feedback

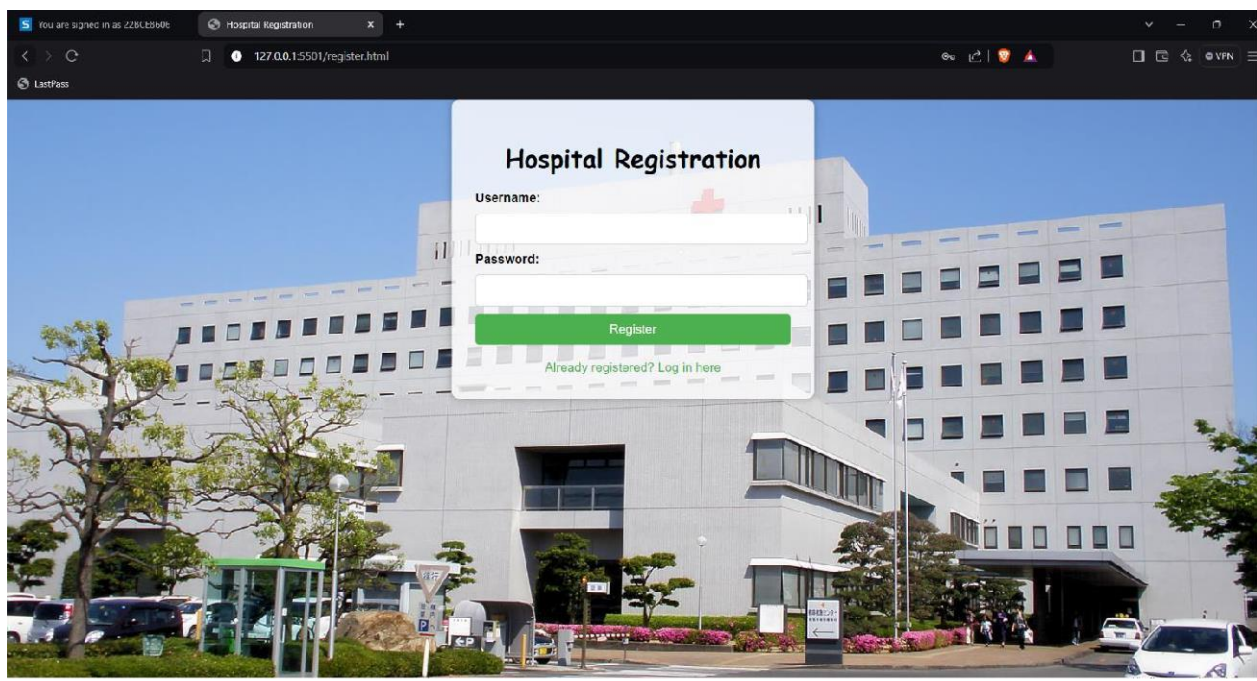
Name:

Email:

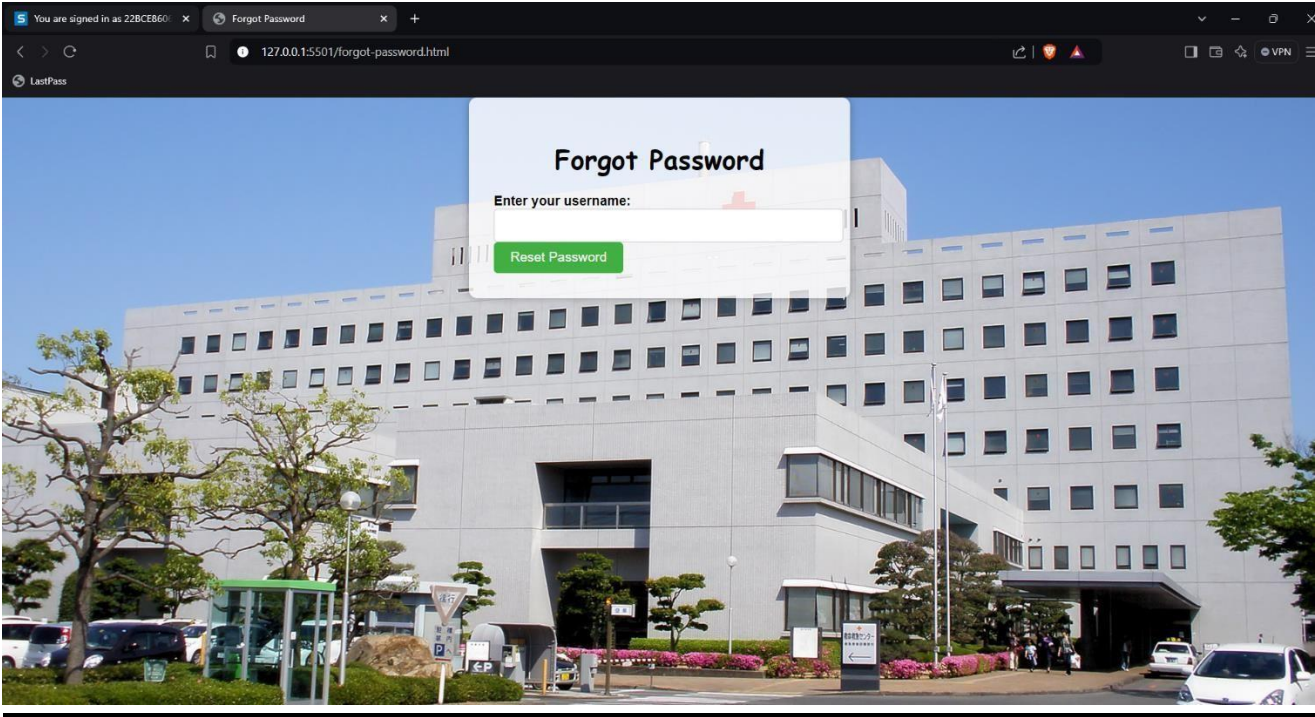
**Feedback:**  
Your feedback here...



## CONTACT US & REGISTRATION PAGE:

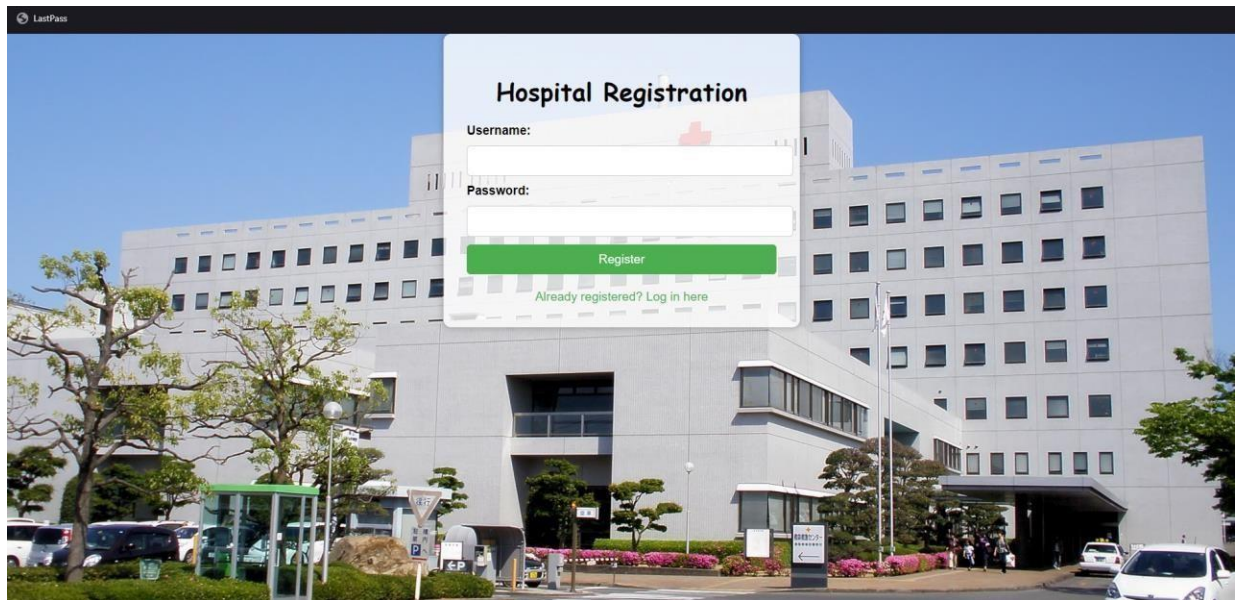


**FORGOT PASSWORD:**



# UNIT-TESTING:

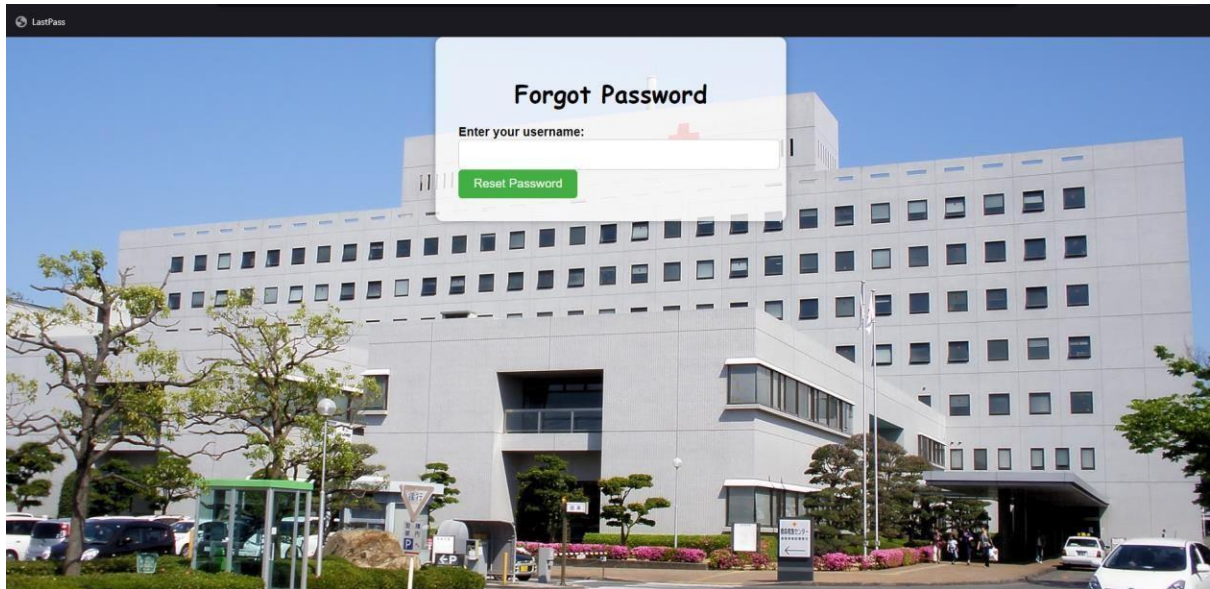
## SIGNUP FORM:



S.N O	TEST CASE	EXPECTED RESULT	TEST CASE
1.	Enter Valid Username and Password and Click on Register/Signup button	Software shows <b>REGISTRATION SUCCEFUL!</b> And redirect to Log-in page	Successful
2.	Enter Already used Username and password and Click on Signup button	Software shows USER NAME already Exists, Please enter valid username	Successful
3.	Enter valid username and click on signup button	Software shows Please Fill this, Field.	Successful
4.	Enter password with less than 6 characters and click on Signup button	Software shows please enter correct format password	Successful

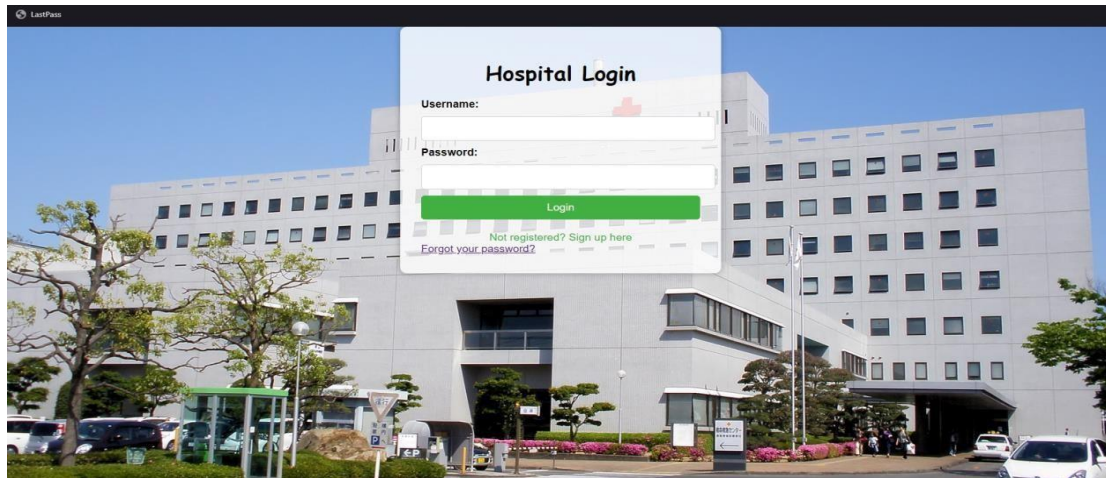


## FORGOT PASSWORD PAGE:



S.NO	TEST CASE	EXPECTED RESULT	TEST CASE
1.	Click on the forget password	It will open to the forget page	Successful
2.	Enter the valid username	It will open to create new password page	Successful
3.	Enter the invalid username	It will show that to check the valid username	Successful
4.	Enter new password	It will create new password	Successful
5	Password re-set successful	Re-direct to login page	Successful

## LOG-IN PAGE:



S.N O	TEST CASE	EXPECTED RESULT	TEST CASE
1.	Enter valid Username and Password and click on Login button	Software shows LOGIN SUCCESSFUL! and redirects to Homepage	Successful
2.	Enter invalid Username and Password and click on Login button	Software shows "Invalid Username or Password" error message	Successful
3.	Enter only Username and leave Password field empty, then click on Login button	Software shows Please Fill this, Field.	Successful
4.	Enter Username and Password with special characters and click on Login button.	Software shows "Please enter valid characters.	Successful
5.	Enter a correct Username and incorrect Password and click on Login button	Software shows "Incorrect Password. Please try again.	Successful
6.	Enter only Password and leave Username field empty, then click on Login button	Software shows "Please fill in Username field	Successful



## APPOINTMENT PAGE:

**VIT HOSPITALS**

Home Specialities Facilities and Services Appointment Records Feedbacks Contact Us Logout

### Book an Appointment

Name:

Mobile Number:

Problem:

Date:

Time Slot:

**Book Appointment**

SNO	Test cases	Expected Result	Test case
1	Verify Name field accepts valid names Name: "John Doe"	System accepts the input, and the name is displayed correctly in the appointment.	Successful
2	Verify Mobile Number field accepts valid format Mobile: "123456789"	System accepts the input, and the mobile number is displayed correctly in the appointment.	Successful
3	Verify Problem field accepts valid entries Problem: "Fever and headache"	System accepts the input, and the problem is displayed correctly in the appointment.	Successful
4	Verify Name field rejects empty values (Name: "")	System shows an error "Name is required."	Successful

## FEEDBACK PAGE:

**VIT HOSPITALS**

Home Specialities Facilities and Services Appointment Records Feedbacks Contact Us Logout

**Feedback**

Name:  Email:

**Feedback:**  
Your feedback here....

S.NO	Test cases	Expected result	Test case
<u>1</u>	Enter valid name,email id and write feedback and click on submit button	Feedback is successfully submitted and confirmation is displayed	Successful
<u>2</u>	Enter valid email id,feedback and click on submit button	it shows us to enter a valid 'NAME'	Successful
<u>3</u>	Enter a valid name,invalid email id and write feedback message and click the submit button	an error message is displayed indicating that email format is invalid	Successful
<u>4</u>	Enter a valid name,email id and write feedback message exceeding the maximum character limit	an error message is displayed that the feedback is exceeding the limit	Successful
<u>5</u>	Navigate the feedback section and check the list of submitted feedback	it shows the all correct details	Successful

## FUNCTIONAL REQUIREMENT TESTING:

Module	Test Case Description	Expected Outcome
<b>Navigation Bar</b>	Ensure that navigation bar options (Home, Specialities, Facilities and Services, Appointment, Feedbacks, Contact Us, Logout) are functional.	Each button should redirect to its respective page.
<b>Patient Management</b>	Verify patient registration captures all required details (name, contact, medical history).	Patient data should be saved accurately in the system.
<b>Appointment Booking</b>	Confirm that patients can book, modify, and cancel appointments.	Appointments should be updated in the system accordingly.
<b>Doctor Management</b>	Verify doctors can create/update profiles, including specialization and availability.	Doctor profiles should be saved and reflect accurate information.
<b>Feedback Management</b>	Verify patients can submit feedback or complaints regarding services.	Feedback should be recorded and accessible for hospital management.
<b>Contact Us</b>	Check that clicking on the map in the Contact Us section redirects to the hospital location on maps.	User is redirected to map with the hospital's location.
<b>Data Backup and Recovery</b>	Test data backup functionality and ensure data can be recovered post-system failure.	Data should be restored from backups without any loss or corruption.

GitHub Link: [https://github.com/Monish36/Team\\_8\\_HMS](https://github.com/Monish36/Team_8_HMS)

### **BATCH-8**

#### **Submitted by**

V Monish	- 22BCE8606
Ragav varma	- 22BCE8598
Brahmajna Maddukuri	- 22BCE8584
D.R.V.Amarnath	- 22BCE8654
Sk.M.Subani	- 22BCE9845

#### **Faculty**

**Prof. Shaik Kareemulla**