***SOFTWARE REQUIREMENT SPECIFICATION FOR***



***HOSPITAL***

***MANAGEMENT SYSTEM***

**21K-3079 Rayyan Ahmed**

**21K-3051 Abdul Rafay**

**21K-3072 Minal Alam**

**21K-3090 Zain Tariq**

**Table of Contents**

# Introduction

1.1 Purpose

1.2 Document Conventions

1.3 Intended Audience and Reading Suggestions

1.4 Product Scope

1.5 References

# Overall Description

2.1 Product Perspective

2.2 Product Functions

2.3 User Classes and Characteristics

2.4 Operating Environment

2.5 Design and Implementation Constraints

2.6 User Documentation

2.7 Assumptions and Dependencies

# External Interface Requirements

3.1 User Interfaces

3.2 Hardware Interfaces

3.3 Software Interfaces

3.4 Communications Interfaces

# System Features

4.1 Patient Module

4.1.1 Login Functionality for Patients

4.1.2 Update medical information

4.1.3 Appointment Book

4.2 Doctor Module

4.2.1 Login Functionality for Doctors

4.2.2 Display patient

4.2.3 Assign prescriptions

4.3 Admin Module

4.3.1 Login Functionality for Hospital Administrators

4.3.2 Display Records

4.3.3 Add records

4.3.4 Update Records

4.3.5 Delete Records

# Non-Functional Requirements

5.1 Performance Requirements

5.2 Security Requirements

5.3 Reliability Requirements

5.4 Availability Requirements

5.5 Maintainability Requirements

5.6 Portability Requirements

# Test and Validation Strategy

6.1 Test Plan

6.2 Test Cases

6.3 Validation Strategy

# Illustrations

7.1 Use case diagram

7.2 Activity diagram

7.3 Gantt chart

**Conclusion**

**Glossary**

# Introduction

## 1.1 Purpose

The purpose of this Software Requirements Specification (SRS) document is to outline the requirements and specifications for the development of a Hospital Management System. The system aims to streamline the hospital's operations and provide an efficient and organized platform for managing patient data and appointments.

**1.2 Document Conventions**

This document follows the IEEE Standard for SRS documentation.

## 1.3 Intended Audience and Reading Suggestions

This document is intended for software developers, project managers, and other stakeholders involved in the development of the Hospital Management System. It is recommended to read the entire document to gain a complete understanding of the system requirements and specifications.

## 1.4 Product Scope

The Hospital Management System will be a web-based application that will include three main modules: Patient, Admin, and Doctor. Each module will have specific functionalities to manage patient data and appointments. The system will be scalable and will support multiple users simultaneously.

**1.5 References**

No external reference was used in this system

# Overall Description

## 2.1 Product Perspective

The Hospital Management System is an independent software system designed to manage and maintain the hospital's patient data and appointments. The system will be used by an admin that will be responsible for managing doctor’s and patient’s info. And patients can book appointments and can update their disease. While doctor can view his/her patients and assign prescription to them and this all will be done via simple easy to use UI

## 2.2 Product Functions

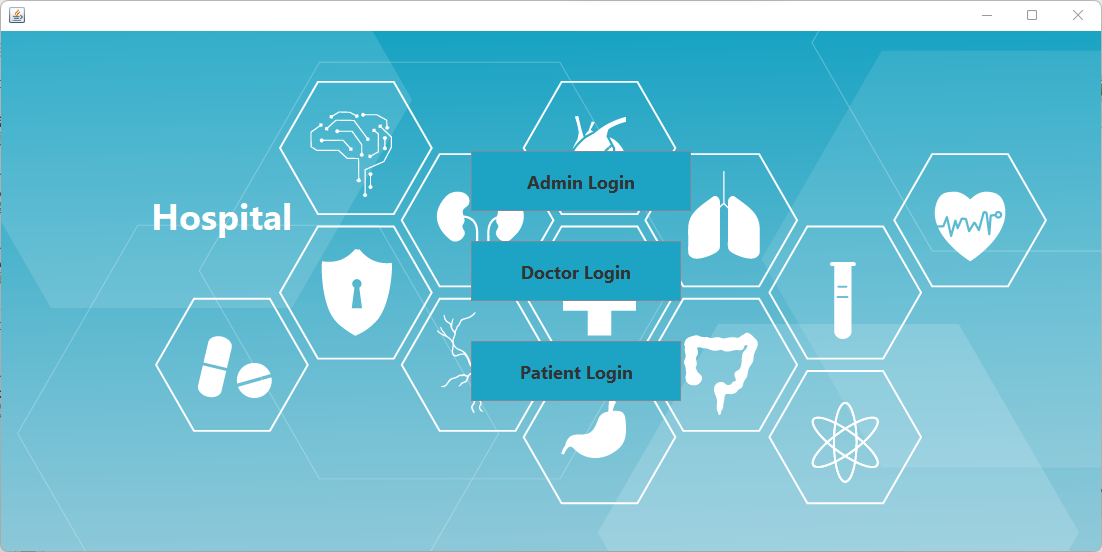
**2.2.1 External Interface Requirements**

## 3.1 User Interfaces

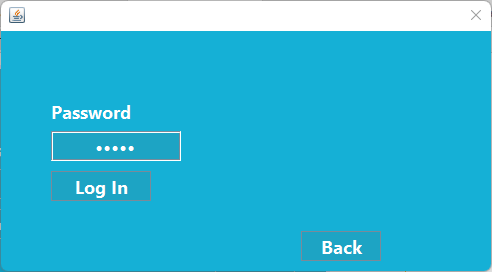
The Hospital Management System will have a user-friendly interface designed to provide easy access to all the features and functionalities. The user interface will be designed using modern design principles to ensure a pleasant user experience. The interface will be intuitive and straightforward to use, with clear and concise labels and buttons.

* **Main Page:**

In main page, the user has a choice to login as an Admin, Doctor or as a Patient.

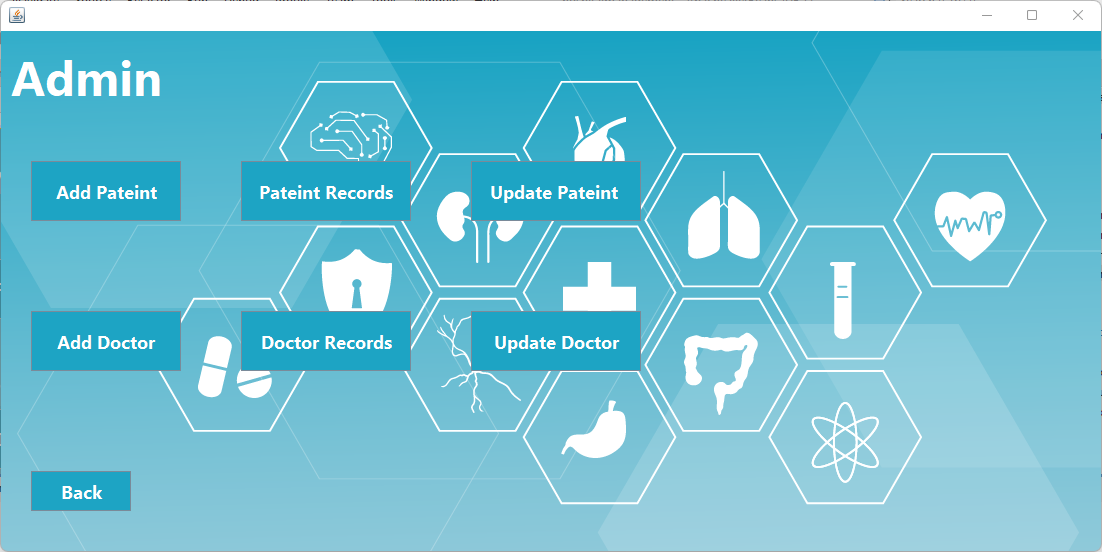


* **Admin Login**



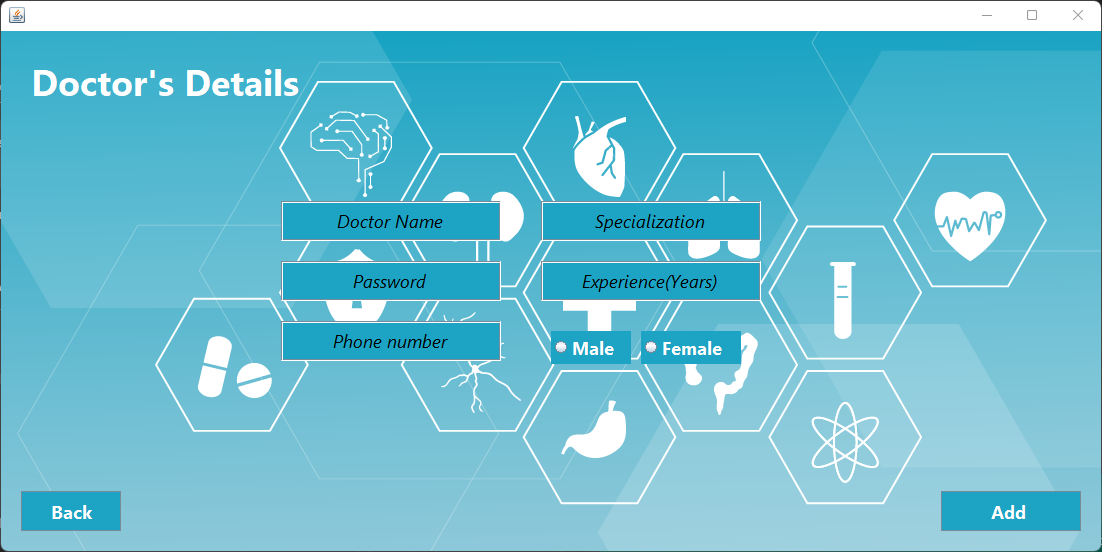
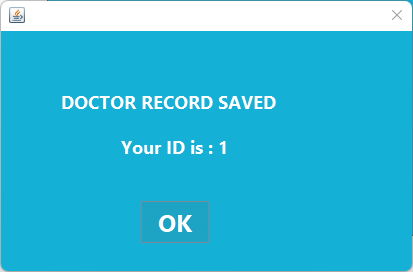
**Admin Module:**

* **Admin’s main page**

Admin has the responsibility of adding, updating and maintaining the Doctor and Patient’s records.

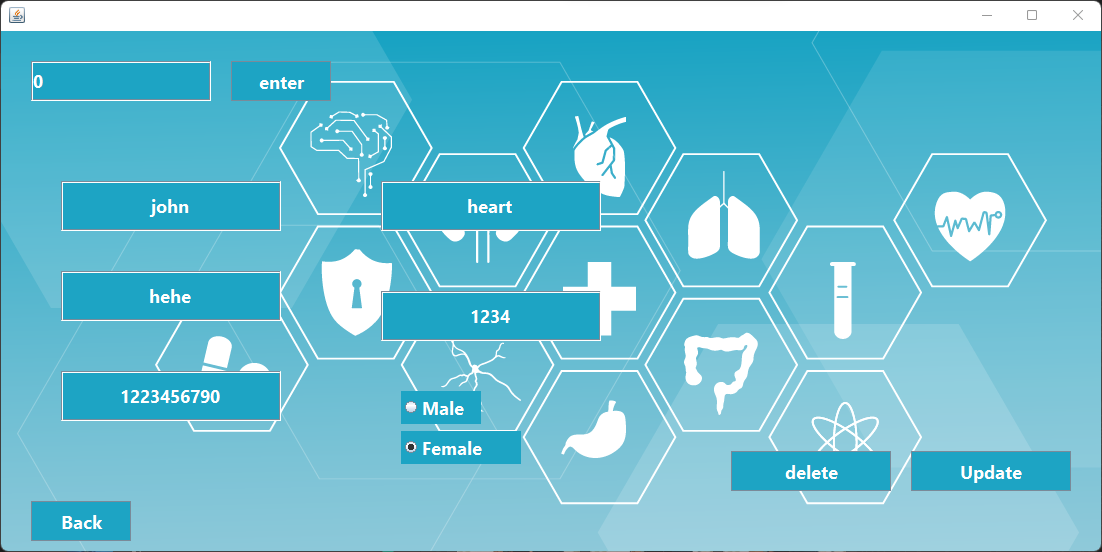
* **Doctor’s Registration**

Adding all the necessary details required for registering a doctor and assigning them a unique ID to avoid any collisions.



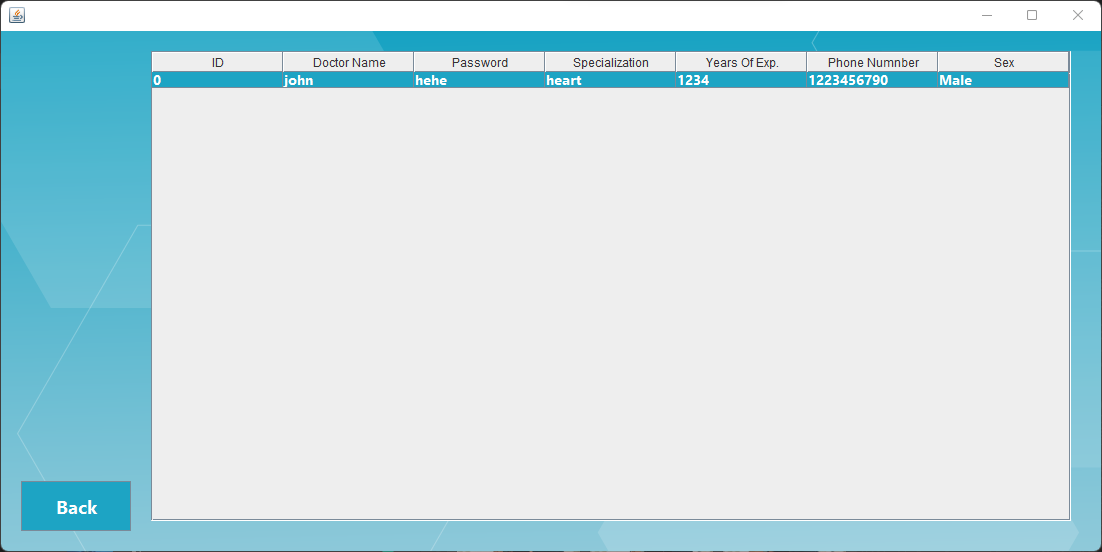
* **Updating or Deleting Doctor’s Details**

Updating any information or deleting the registration of doctor.

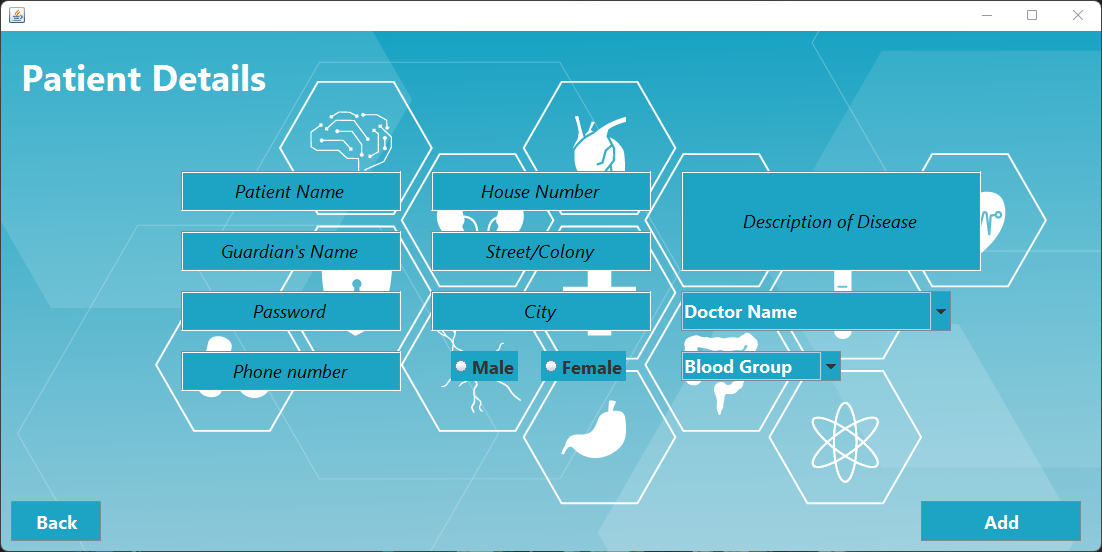


* **Doctor Records**

Shows all the records of doctors.

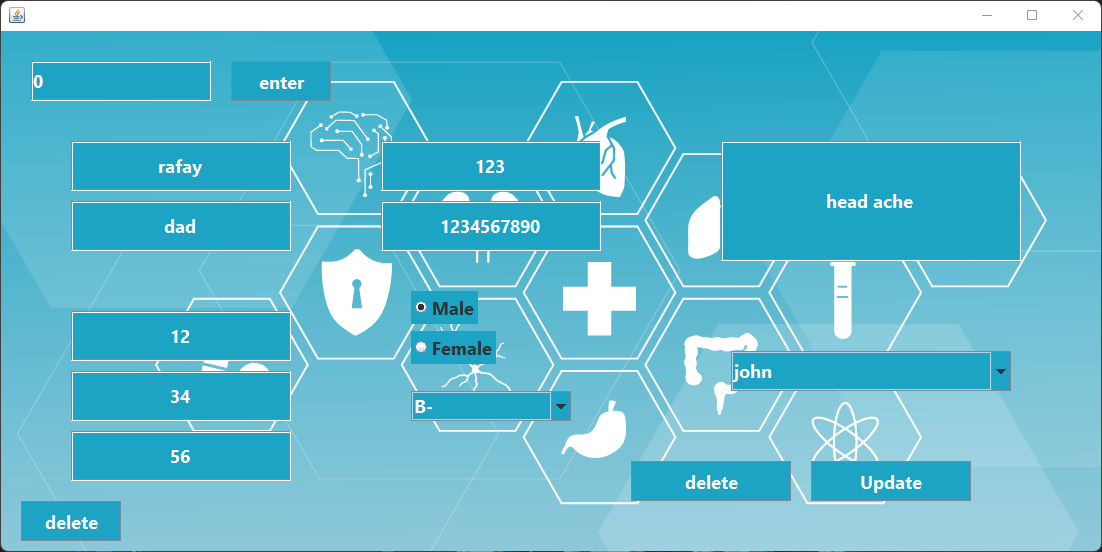


* **Patient Registration**

Adding all the necessary details required for registering a patient and assigning them a unique ID to avoid any collisions.

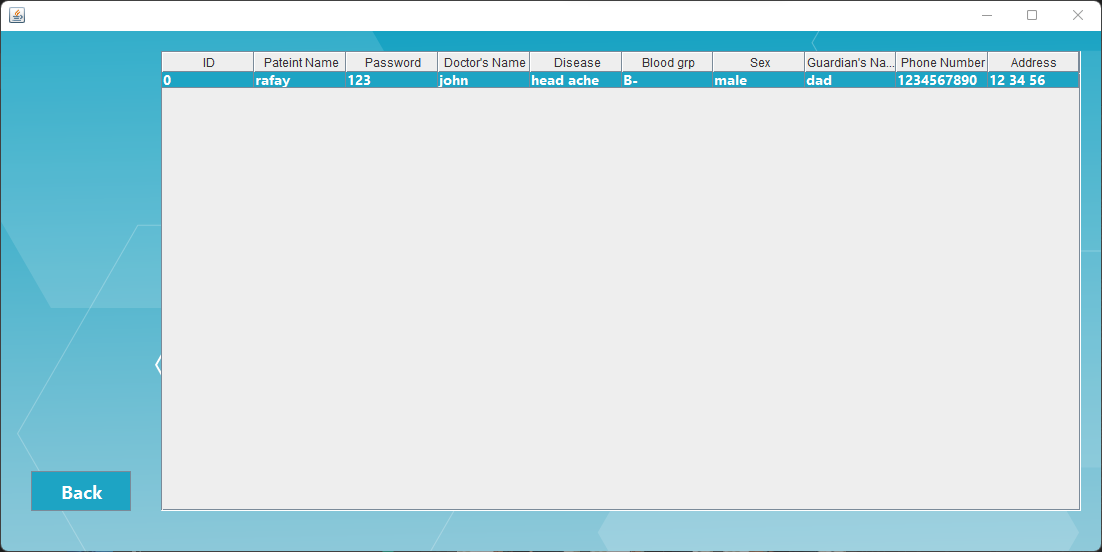
* **Updating or Deleting Patient’s Details**

Updating any information or deleting the registration of patient.



* **Patient Records**

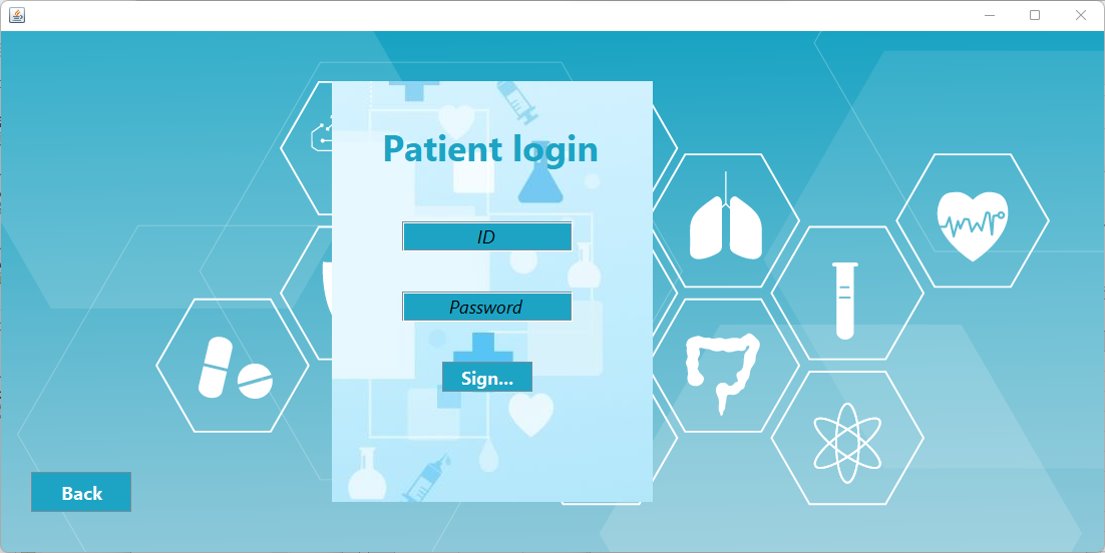
Shows all the records of patient



**Patient Module:**

* **Patient Login**

The registered patient must enter their ID and Password



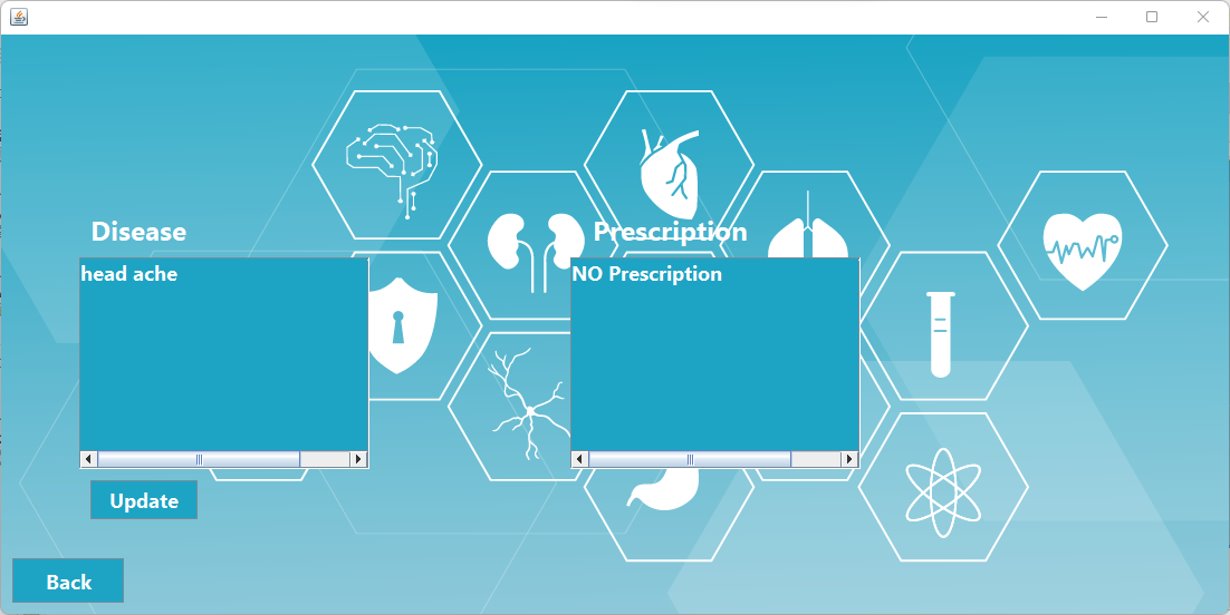
* **Book Appointment and See Prescription**

Patients can get their appointment schedule and see prescription assigned by their doctor.



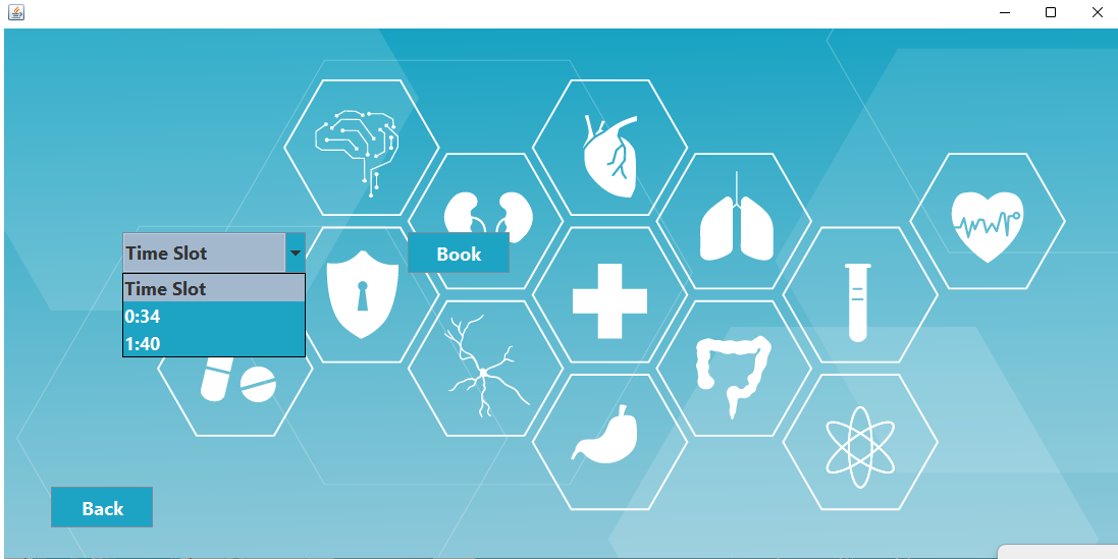
1. **See Prescription**

Here the patient can see what their doctor has prescribed them, and the patient can also update their disease.



1. **Add Appointments**

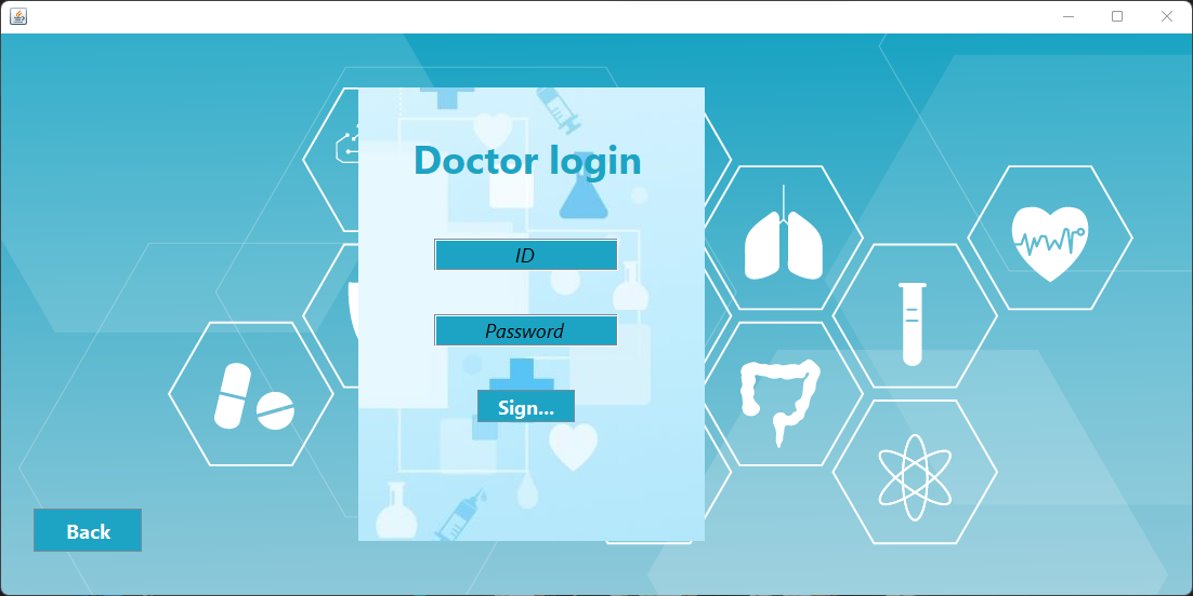
The patient can schedule their appointment in the available times set by the doctor.



**Doctor Module:**

* **Doctor Login**

The registered doctor must enter their ID and Password



****

* **Prescribing**

The doctor can give the patient a description according to the patient’s disease.

Diagram

Description automatically generated

* **Availability**

The doctor can add the time slots for when they’re available.



* **Scheduled Appointments**

The doctor can view which of their patients has scheduled their appointment for what time.



## 3.2 Hardware Interfaces

The Hospital management system does not have any specific hardware requirements. The system should have at least two cores, 2.0 Ghz clock frequency, and 1GB of ram. It is compatible with most modern devices, including desktops, laptops, tablets, and smartphones, as long as they have a reliable internet connection.

# System Features

The Hospital Management System will have the following features:

## 4.1 Patient Module

**4.1.1 Login Screen**

* A login screen that allows patients to enter their credentials to access their profile.

**4.1.2 Update Problem**

* A profile page that allows patients to update their medical information.

**4.1.3 Appointment Book**

* A page for booking appointments with doctors.

## 4.2 Doctor Module

**4.2.1 Login Screen**

* A login screen that allows doctors to enter their credentials to access their profile.

**4.2.2 Display Patient**

* This feature displays the patient

**4.2.3 Assign Prescriptions**

* A page for adding prescriptions for patients.

## 4.3 Admin Module

### 4.3.1 Login

* A login screen that allows administrators to enter their credentials to access their profile.

**4.3.2 Display records**

* A dashboard that displays patient records, and doctor records.

**4.3.3 Add record**

* A page for adding new patients and doctors to the system.

**4.3.4 Update records**

* A page for updating patients' and doctor’s information in the system.

**4.3.5 Delete records**

* A page for deleting patients' and doctor’s information in the system.

## 4.4 Security Features

* The Hospital Management System will have the following security features:
* Password-protected login screens for patients, doctors, and administrators.
* Role-based access control: The system shall enforce different levels of access based on the user's role, such as patient, doctor, and admin.

# Non-Functional Requirements

**5.1 Performance Requirements:**

* The system should be able to handle a large number of concurrent users without significant performance degradation.
* The response time for most user requests should be less than 2 seconds.
* The system should be able to process and display large amounts of data quickly and efficiently.
* The system should be able to handle data transfers over low or no bandwidth connections.

**5.2 Security Requirements:**

* The system should have strong authentication mechanisms to prevent unauthorized access.
* The system should have a role-based access control mechanism to control access to different system functionalities based on user roles.

**5.3 Reliability Requirements:**

* The system should have a high level of reliability with minimum downtime.
* The system should have data backup and recovery mechanisms to prevent data loss in case of system failure.
* The system should be able to handle errors and exceptions gracefully without affecting the overall system performance.

**5.4 Availability Requirements:**

* The system should be available 24/7 with a minimum uptime of 99.9%.
* The system should have load balancing mechanisms to distribute the load across multiple servers and ensure high availability.
* The system should have redundancy mechanisms to prevent single points of failure.

**5.5 Maintainability Requirements:**

* The system should be modular and well-structured to enable easy maintenance and enhancements.
* The system should have comprehensive documentation that includes user manuals, installation guides, and system architecture diagrams.
* The system should be designed in a way that allows easy upgrades and patches.

**5.6 Portability Requirements:**

* The system should be platform independent and should be able to run on different operating systems.
* The system should be designed in a way that allows easy migration to different hardware and software environments.
* The system should support different web browsers and mobile devices.

# Test and Validation Strategy

## 6.1 Test Plan

A comprehensive test plan will be created to ensure that the system functions according to the requirements. Each test will be documented with clear instructions and expected results.

**6.2 Test Cases**

**Admin wrong password:**

Graphical user interface, website

Description automatically generated

* **Wrong credentials:**

Diagram

Description automatically generated

* **Empty fields** 
  + **Phone number should have 10 digits**

**Diagram

Description automatically generated**

* **Can't add a patient without a doctor**

**Diagram

Description automatically generated**

* **Can't see patient records without adding patients**

**Diagram

Description automatically generated**

* **Drop down menu for blood groups**

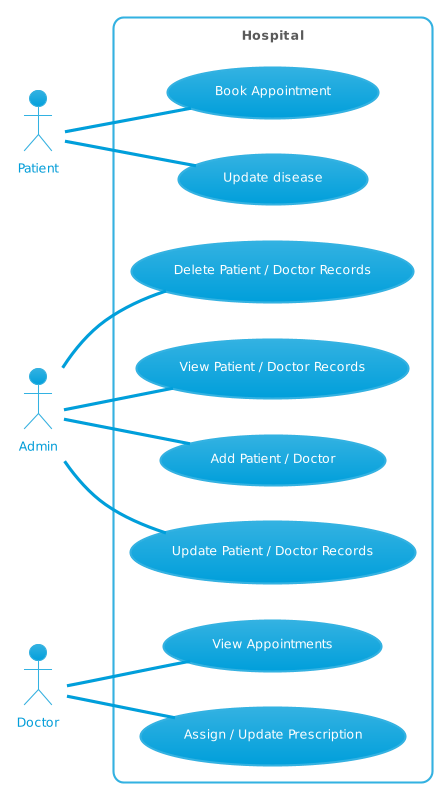
**Diagram

Description automatically generated**

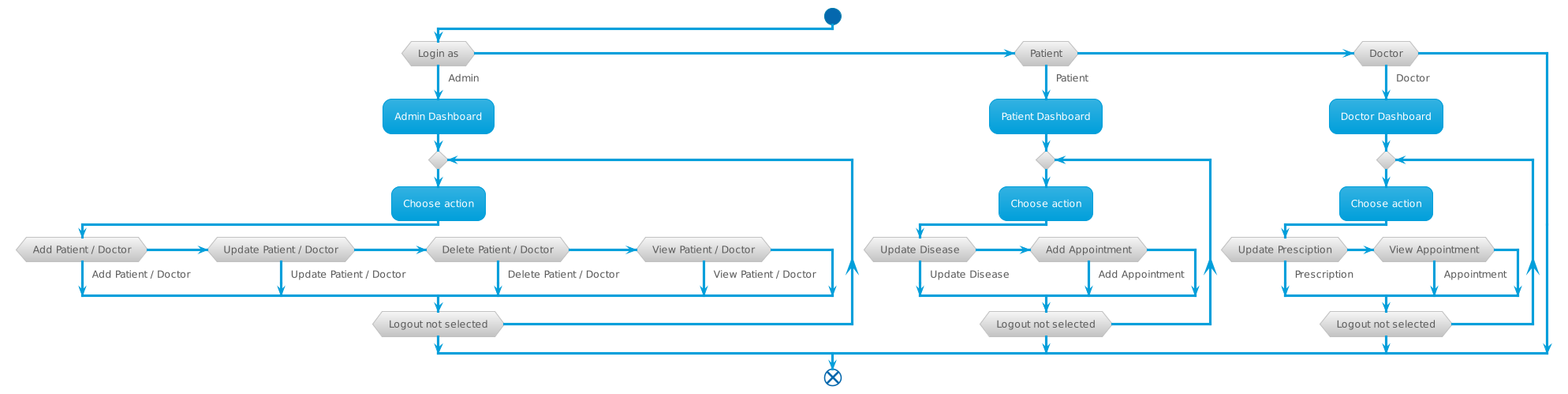
**6.3 Validation Strategy**

The validation strategy will include the verification and validation of the system to ensure that it meets the user's needs and requirements. The system will be evaluated based on its functionality, usability, and performance. The feedback from the users will be used to improve the system and ensure that it meets their needs. The validation process will continue until the system is ready for deployment.

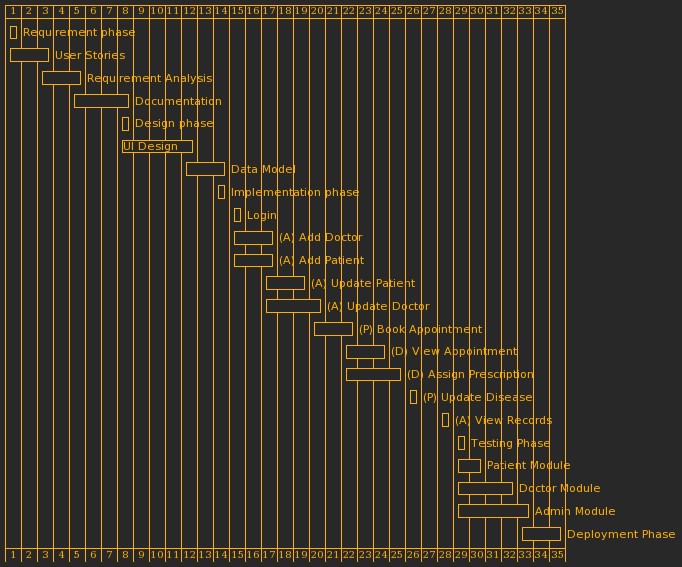
# Illustrations 7.1 Use Case Diagram



## 7.2 Activity Diagram



## 7.3 Gantt Chart



## Conclusion

The Hospital Management System is a comprehensive solution for managing hospital operations and enhancing the patient's experience. The system provides an easy-to-use interface that allows users to perform various functions such as adding new patients and doctors, managing appointments, generating reports, and much more. The system is designed to be user-friendly and accessible to all users, regardless of their technical proficiency.

## Glossary

This section provides a list of terms and definitions used throughout this document.

SRS: Software Requirements Specification. A document that outlines the requirements for a software system.

UI: User Interface. The visual elements and controls through which a user interacts with a software system.

Database: A collection of data that is organized in a way that allows it to be easily accessed, managed, and updated.

Authentication: The process of verifying the identity of a user or system.

Authorization: The process of granting or denying access to a resource based on the user's or system's identity and privileges.

GUI: Graphical User Interface. A type of user interface that allows users to interact with a software system using graphical elements such as icons, buttons, and menus.

Reliability: The ability of a software system to perform its intended functions without failure or errors over a period.

Scalability: The ability of a software system to handle increasing amounts of work or traffic without compromising its performance or stability.

Usability: The ease with which a software system can be used by its intended users.

Performance: The speed, responsiveness, and efficiency of a software system under different workloads and conditions.