



Object Oriented Programming

SEMESTER PROJECT REPORT

Table of Contents

1. [Background](#)
2. [Introduction](#)
3. [Literature Review](#)
4. [Methodology](#)
5. [Conclusion](#)
6. [GUI screenshots](#)
7. [CSV files](#)
8. [UML diagrams](#)
9. [Flowchart](#)

The document contains all the required sections and diagrams. Additionally, the files for these parts and more are further included in the zip folder.

Group Members : Uzair , Ali Ahmad , Gulwarina Muska Saleem

Background

Hospital Management Systems (HMS) have become indispensable in modern healthcare, streamlining operations, reducing errors, and enhancing patient care. Traditional paper-based record-keeping is prone to misplacement, transcription errors, and delayed information sharing. In response, software solutions have evolved to manage patient records, appointments, billing, clinical workflows, and communications in an integrated, digital environment. Our HMS, developed for the CS-212 Object-Oriented Programming course under instructor Nazia Perwaiz, addresses these needs by providing a comprehensive, modular platform that covers the full lifecycle of patient care—from registration and vitals monitoring to prescription, emergency alerts, reporting, and billing.

Introduction

This report presents an in-depth overview of the Hospital Management System project. The primary objectives are to:

1. **Automate and secure** user registration, authentication, and role-based access.
2. **Enable remote vitals monitoring** with CSV import, manual entry, real-time threshold checks, and data visualization.
3. **Support clinical workflows** including doctor feedback, prescription management, and emergency alerting.
4. **Facilitate communication** via chat and video consultations.
5. **Manage appointments, notifications, and billing** in a unified interface.
6. **Generate analytical reports** and graphs for both patients and providers.

Literature Review

Several commercial and open-source HMS solutions exist today:

- **MediTech Expanse:** A comprehensive proprietary suite, MediTech excels in interoperability and analytics but comes at high cost and complexity.
- **GNU Health:** An open-source alternative focusing on public health and hospital workflow, with strong reporting capabilities but limited real-time communication modules.
- **OpenMRS:** Widely adopted in low-resource settings, OpenMRS offers modular patient record management, but requires significant configuration and lacks built-in vitals CSV import and emergency alert features.

Compared to these, our system emphasizes real-time vitals monitoring (CSV and manual), automatic emergency alerts, built-in chat/video integrations, and lightweight deployment suitable for educational and small-to-mid-sized clinical settings. It balances functionality with ease of customization, leveraging object-oriented design to allow straightforward extension of modules such as NotificationService or AppointmentManager.

Methodology / System Overview

The HMS is organized into nine functional domains, each realized by one or more collaborating modules. User interactions are captured via web-based dashboards and forms, while server-side services enforce business rules, data validation, storage, and notifications.

1. User Management & Authentication

- **Patient, Doctor, Administrator Registration**
 - **Action:** User selects role, completes registration form with personal details (name, email, password).
 - **Response:** System hashes password, creates account in User base table, and—for non-admin roles—sends activation link via email.
 - **Rules:** Email must be unique; password must meet complexity (≥ 8 characters, one uppercase, one digit). Role determines accessible dashboard views.

- **Login/Logout & Role-Based Dashboards**
 - **Action:** User submits credentials on login page.
 - **Response:** System verifies hash, issues session token, and redirects to role-specific dashboard (AdminDashboard, DoctorDashboard, PatientDashboard).
 - **Rules:** Inactive accounts cannot log in; sessions expire after 30 minutes of inactivity.

2. Health Data Upload & Monitoring

- **CSV Import of Vitals**
 - **Action:** Patient navigates to “Upload Vitals,” selects a CSV file containing timestamped readings (heart rate, blood pressure systolic/diastolic, temperature, oxygen saturation).
 - **Response:** Backend parses CSV, validates each row ($0 < \text{heartRate} < 300$, $30 \leq \text{systolic} \leq 200$, etc.), stores valid entries in VitalsDatabase; aggregates errors reported back to user.
 - **Rules:** File size ≤ 2 MB; columns must match prescribed schema; invalid rows are skipped with detailed feedback.
- **Manual Vitals Entry**
 - **Action:** Patient opens “Add Vital” form, enters single readings.
 - **Response:** System performs same range checks, stores entry, and immediately checks for critical thresholds.
 - **Rules:** All required fields must be completed; timestamp defaults to current server time if omitted.
- **Critical-Threshold Detection**
 - **Mechanism:** After each new reading, VitalSign.isCritical() evaluates thresholds (e.g. temperature > 39 °C, heart rate > 120 bpm).
 - **Response:** If critical, EmergencyAlert.checkVitalsAndTrigger() invokes NotificationService.notifyDoctor() and NotificationService.sendSMS() to

emergency contacts; an in-app notification pops up on the patient and doctor dashboards.

3. Doctor Feedback & Prescription Workflow

- **Viewing Patient Vitals & History**
 - **Action:** Doctor selects a patient from their “Assigned Patients” list.
 - **Response:** System queries VitalsDatabase.getVitalsForPatient(), MedicalHistory.getHistoryByPatient(), and displays trends graphically (via Chart.js on client side).
- **Adding Prescriptions / Recommendations**
 - **Action:** Doctor clicks “Add Prescription,” fills medication name, dosage, frequency, duration, and optional notes.
 - **Response:** Backend saves to Prescription table, links to patient’s MedicalHistory, and triggers ReminderService.scheduleMedicationReminder().
- **Feedback Status Updates**
 - **Action:** Doctor marks feedback as “Reviewed” or “Action Required.”
 - **Response:** UI updates status badge; if “Action Required,” patient receives an email via EmailNotification.

4. Emergency Alert System

- **Automatic Alerts**
 - **Process:** Vitals import or manual entry crossing a threshold calls EmergencyAlert.triggerAlert().
 - **Response:** Creates an EmergencyAlert record, notifies doctor and emergency contacts immediately through SMS and email, and flags patient dashboard with urgent notification.
- **Panic Button**

- **Action:** Patient presses “Panic” in dashboard.
- **Response:** Calls PanicButton.manualPanic(), which invokes the same notification pipeline as automatic alerts, including geolocation data if available.
- **Notification Delivery**
 - **Channels:** In-app (toast notifications), Email (EmailNotification.notify()), SMS (SMSNotification.notify()).
 - **Rules:** Rate-limit SMS to one per five minutes per patient to avoid spamming.

5. Chat & Video Consultation

- **Real-Time Chat**
 - **Action:** User opens chat window with selected contact.
 - **Response:** Establishes WebSocket via ChatServer, messages routed through ChatClient, persisted in ChatMessage table.
 - **Rules:** Messages ≤ 1 000 characters; profanity filter applied server-side.
- **Video Link Generation**
 - **Action:** Doctor or patient clicks “Start Video Consult.”
 - **Response:** VideoCall.generateMeetingLink() calls Google Meet API (or Zoom SDK) to create a unique link, which is stored and shared in chat.

6. Appointment Scheduling & Management

- **Requesting, Approving, Cancelling**
 - **Action:** Patient selects doctor, date, and time; submits request via AppointmentManager.requestAppointment().
 - **Response:** Creates Appointment with status “Pending.” Doctor sees request on dashboard and can click “Approve” or “Reject,” invoking AppointmentManager.approveAppointment() or .cancelAppointment().

- **Rules:** Time slots are locked per doctor schedule; overlapping appointments are rejected. Only doctors can change “Pending” to “Approved” or “Rejected.”
- **Dashboard Display**
 - **Action:** Both roles see calendar or list view of upcoming “Confirmed” appointments.
 - **Response:** Query Appointment.getDailySchedule() filtered by user role and ID.

7. Notifications & Reminders

- **Appointment Reminders**
 - **Mechanism:** ReminderService.scheduleAppointmentReminder() enqueues jobs via a scheduler (e.g. Quartz) to send reminders 24 hours and 1 hour before the appointment.
 - **Channels:** Email and in-app notifications.
- **Medication Reminders**
 - **Mechanism:** Based on prescription schedule, ReminderService.sendMedicationReminder() fires at prescribed times.
 - **Rules:** Reminders stop after prescription end date.
- **Abnormal-Vitals Alerts**
 - **Covered** in the Emergency Alert System—critical readings trigger immediate notifications.

8. Data Visualization & Reporting

- **Graphing Vital Trends**
 - **Action:** User clicks “View Trends.”
 - **Response:** Frontend fetches aggregated trend data via VitalsDatabase.getTrendData(), renders line/bar charts showing vitals over time.

- **Exporting Health Reports**
 - **Action:** User clicks “Download Report.”
 - **Response:** Backend compiles a PDF (using iText or ReportLab) containing sections for Vitals History tables, embedded trend graphs, Prescription History, and Doctor Feedback notes; streams file to user.
 - **Formats:** PDF and optional CSV export for raw data.

9. Billing & Invoicing

- **Generating Invoices**
 - **Action:** Admin or Doctor marks services rendered; invokes Bills.generateInvoice().
 - **Response:** Creates line items (consultation fee, tests, medication), computes total, saves to Bill table, and links invoice file (PDF) for download.
- **Tracking Payments**
 - **Action:** Billing clerk marks invoice as “Paid” or “Pending.”
 - **Response:** Updates Bill.status; if overdue by 30 days, triggers reminder emails via ReminderService.

Conclusion

This Hospital Management System brings together a wide array of functionalities—secure user management, vitals monitoring with emergency alerts, rich clinical workflows (prescriptions, feedback, appointments), real-time communication, billing, and comprehensive reporting—within a cohesive, object-oriented architecture. Its modular design allows easy extension: new notification channels, integration with additional telehealth APIs, or advanced analytics modules can be plugged in with minimal disruption. This report has outlined the system’s background, positioned it against existing market offerings, and detailed the methodology and system overview, demonstrating its capability to enhance healthcare delivery in academic and small-to-mid-sized clinical settings.

GUI Screenshots

Login Screen

Secure authentication for patients, doctors, and administrators.

1. Sign-Up / Registration

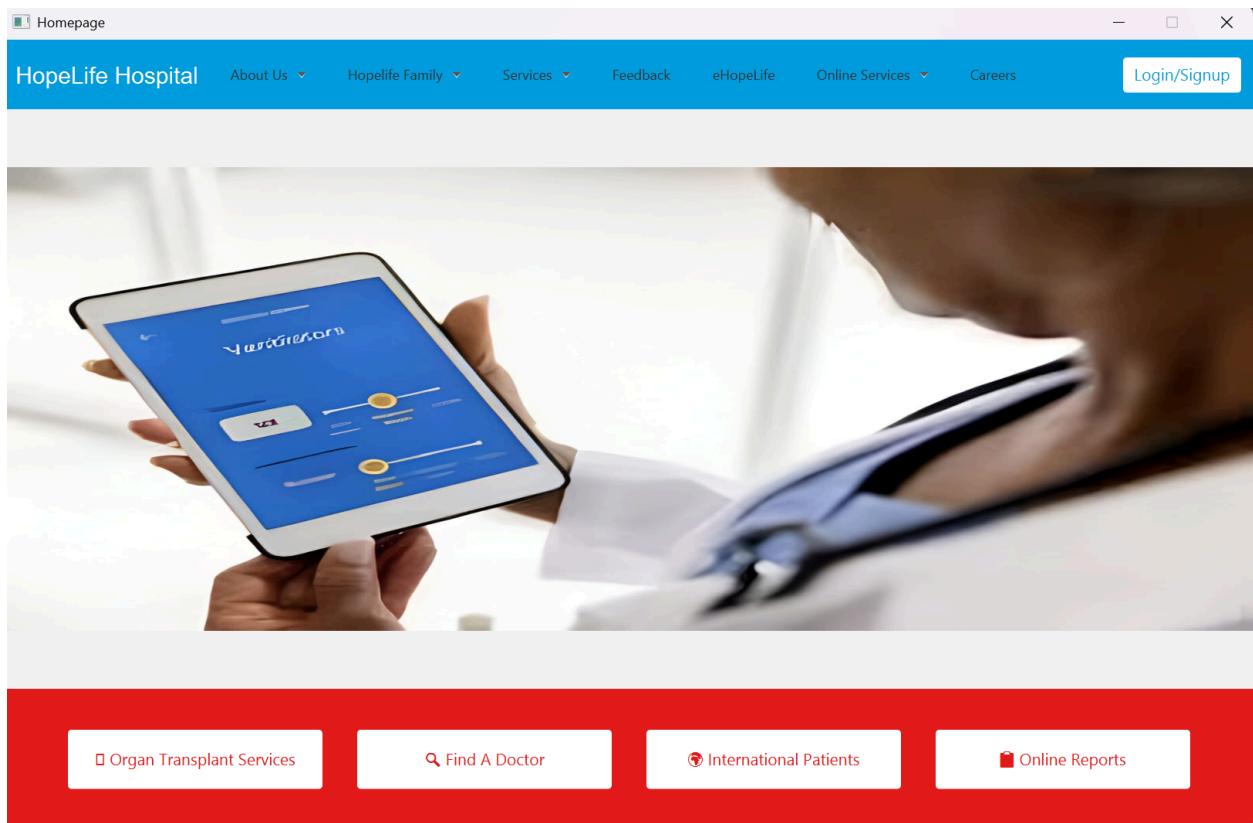


Figure 1: Homepage

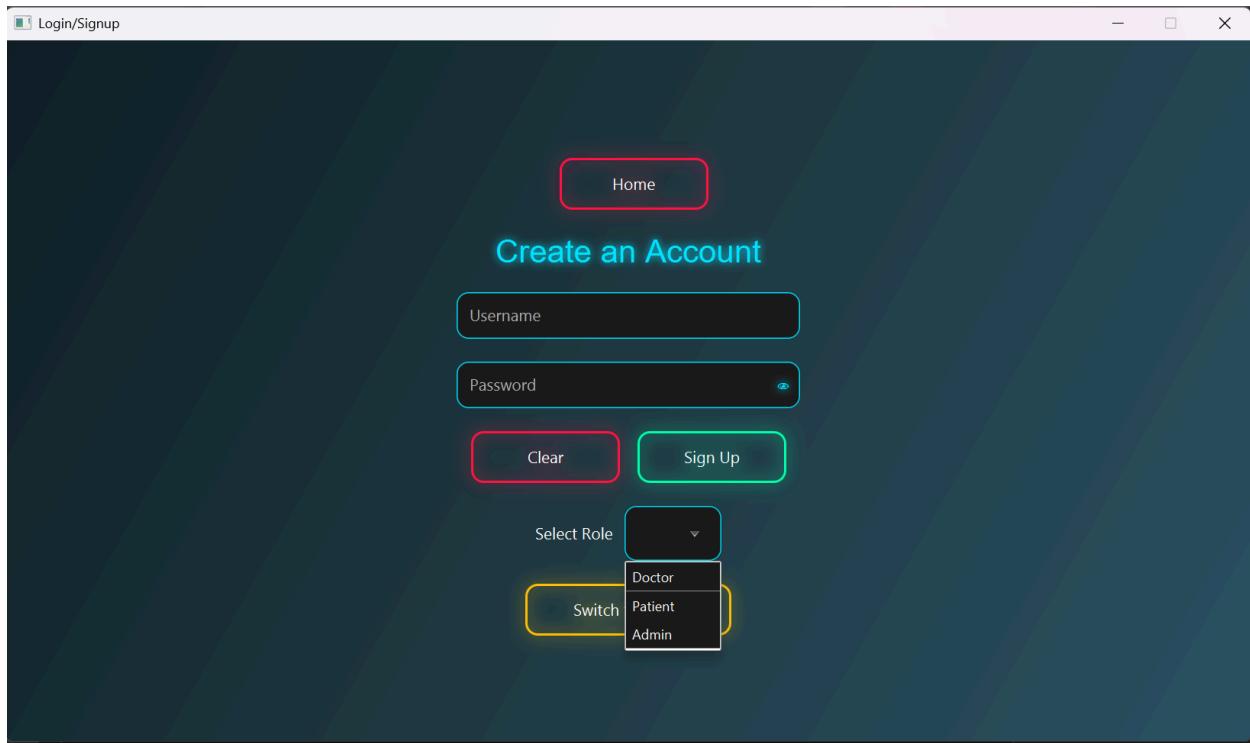


Figure 2:: Sign up page

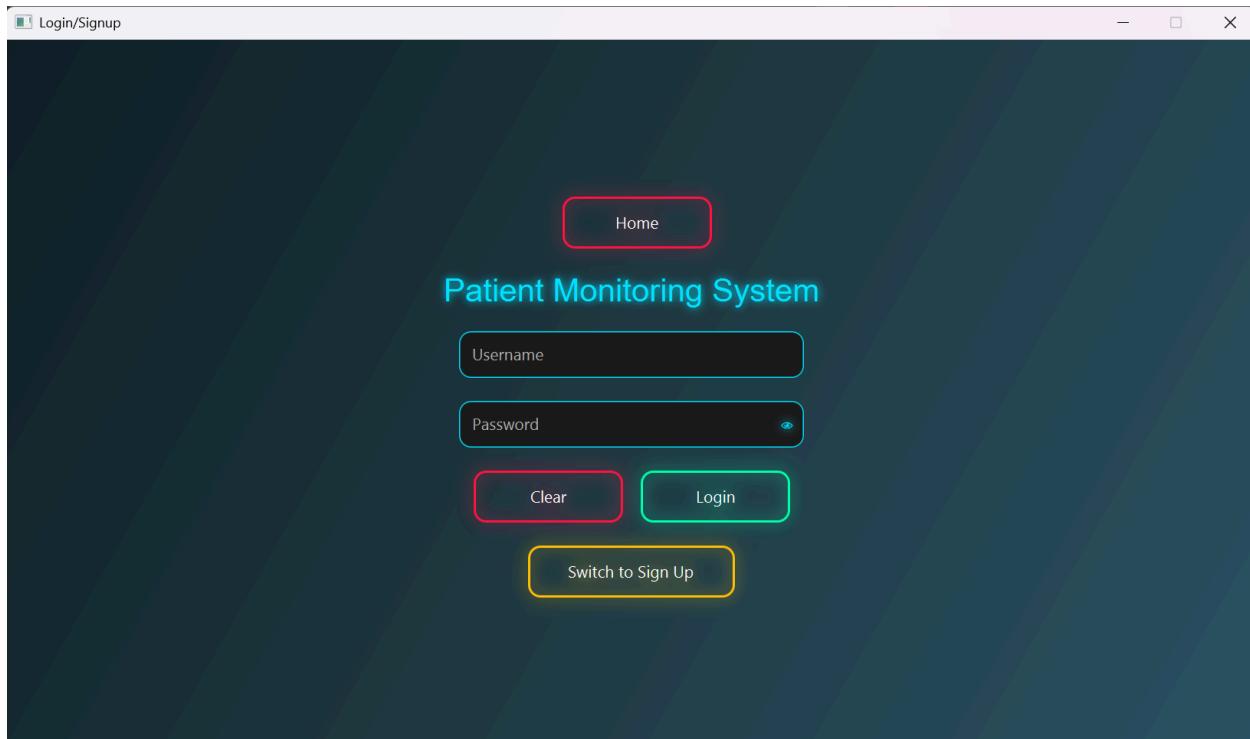


Figure 3: login page

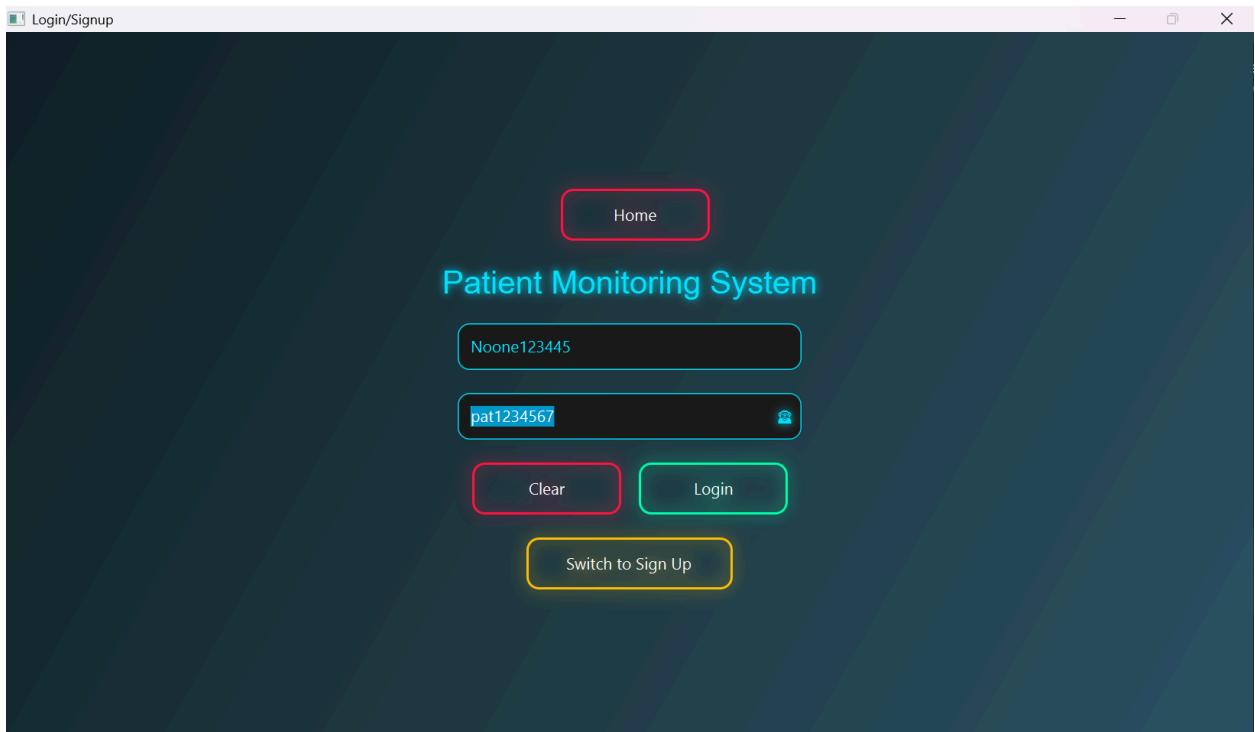


Figure 4: View of logging into the account

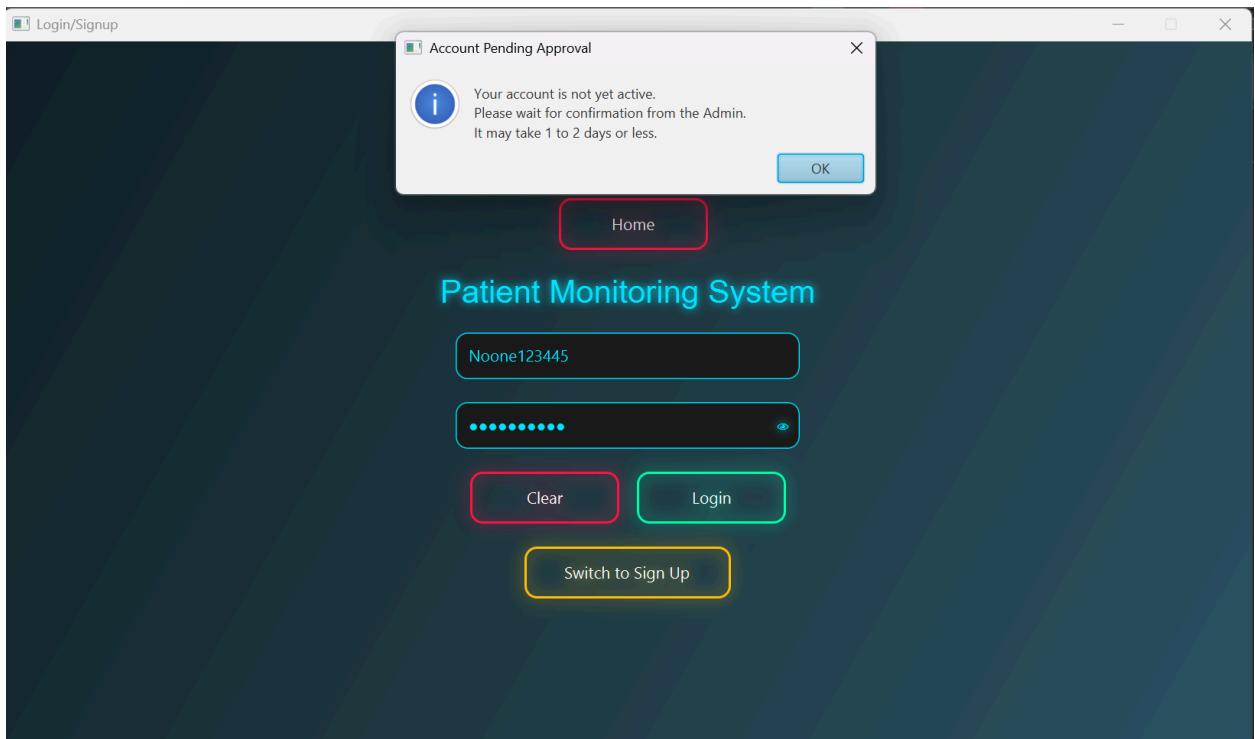


Figure 5: Inactive accounts that are yet waiting for admin appporval cant login

Admin Signup

Username:

Name: Email:

Phone Number: Password:

Address:

Age:

Gender:

Department:

Role:

Sign Up

This screenshot shows the 'Admin Signup' interface. It includes fields for basic user information like Username, Name, Email, Phone Number, and Password. It also includes more specific fields for an administrator such as Address, Age (a dropdown menu with '25' selected), Gender (a dropdown menu), Department (a dropdown menu), and Role (a dropdown menu). A prominent green 'Sign Up' button is located at the bottom left.

Figure 6: Administrator registration form including admin-level selection.

Doctor Signup

Username:

Name: Email:

Phone Number: Password:

Address:

Age:

Gender:

Specialization:

License No:

Experience (Years):

This screenshot shows the 'Doctor Signup' interface. It includes fields for basic user information like Username, Name, Email, Phone Number, and Password. It also includes more specific fields for a doctor such as Address, Age (a dropdown menu with '30' selected), Gender (a dropdown menu), Specialization (a dropdown menu), License No. (a dropdown menu), and Experience (Years) (a dropdown menu). A vertical scroll bar is visible on the right side of the window.

Figure 7: Standard doctor registration interface.

Patient Signup

Username: Email:

Name:

Phone Number: Password:

Address:

Age:

Gender:

Blood Group:

Allergies:

DUST
POLLEN

Figure 8: Patient sign-up page with form fields for name, email, password.

2. Account Activation

Figure 9: Process of verifying and activating user accounts.

Patient Dashboard

Upload vitals, view medical reports, and access doctor feedback.

The screenshot shows a patient dashboard with the following components:

- Header:** Login/Signup, Logout, and a notification icon.
- Left Sidebar (HopeLife Hosp...):**
 - Dashboard
 - My Doctors
 - Appointments
 - Details
 - Messages
 - Feedbacks
 - Prescriptions
 - My Vitals
 - Medical History
 - Add Vitals
 - Send Mail
- Profile Section:** Patient Mr No oneex, Male - 21, freefirebase5657@gmail.com, 0324543533.
- Health Information:**
 - Present Diseases: [LIVER_DISEASE]
 - Allergies: [MEDICATIONS]
 - Blood Group: A+
- User Details:**

User ID :	Noone12345	Blood Group :	A+
Address :	Nothing here now no	Next Appointment :	N/A
Account Status :	Inactive	Primary Doctor :	Hamza Raheem
- Upcoming Appointments:** (Empty)
- Attended Appointments:** (Empty)

Figure 10: Patient dashboard overview showing key metrics and navigation.

- **Assigned Doctor View**

The screenshot shows a web application interface for a patient. The top navigation bar includes 'Login/Signup', a bell icon, and a 'Logout' button. On the left, a sidebar menu for 'HopeLife Hosp...' lists various options: Dashboard, My Doctors (selected), Appointments, Details, Messages, Feedbacks, Prescriptions, My Vitals, Medical History, Add Vitals, and Send Mail. The main content area is titled 'My Doctors' and displays a table with one row of data:

Doctor	ID	Email	Phone
Hamza Raheem	hamzarahmeem23	muzair.bsd24seecs@seecs.edu.pk	0234568243

Figure 11: Patient's assigned doctor details and contact info.

- **Appointment Viewing**

The screenshot shows the same web application interface. The top navigation bar and sidebar menu are identical to Figure 11. The main content area is titled 'Today's Appointments' and displays a table with three rows of data:

Date/Time	Doctor	Patient Name	Appointment ID
2025-05-06 21:30	Noone123445		app122
2025-05-25 21:30	Noone123445		app32423
2025-05-02 14:30	Noone123445		appt001

Below the table, there is a section labeled 'Details:' with a text input field.

Figure 12: List of upcoming and past appointments.

● Vitals & Trends

Add Vital Signs

Record your current health metrics

Heart Rate (bpm):	e.g. 72
Blood Pressure (mmHg):	e.g. 120/80
Temperature (°C):	e.g. 36.8
Oxygen Level (%):	e.g. 98
Respiratory Rate (breaths/min):	e.g. 16
Glucose Level (mg/dL):	e.g. 90
Cholesterol Level (mg/dL):	e.g. 180
BMI:	e.g. 22.5
Hydration Level (%):	e.g. 60
Stress Level (/10):	e.g. 5

Upload Vitals **Save Vitals**

Figure 13: Manual vitals entry modal for heart rate, blood pressure, temperature.

Select Vitals CSV

File name: CSV Files

Name	Date modified
vital	4/29/2025 10:49 PM
Wallpaper	2/25/2025 10:53 PM
Visual Studio	12/6/2024 12:27 PM
Unity_Essentials_Download	1/31/2025 1:04 AM
SQL COURSE	11/4/2024 12:32 AM
software	3/26/2025 3:04 PM
SFML-2.6.1	11/4/2023 3:56 AM
Racing Game	12/27/2024 11:43 AM

Upload Vitals **Save Vitals**

Figure 14: Patients uploading with csv files.

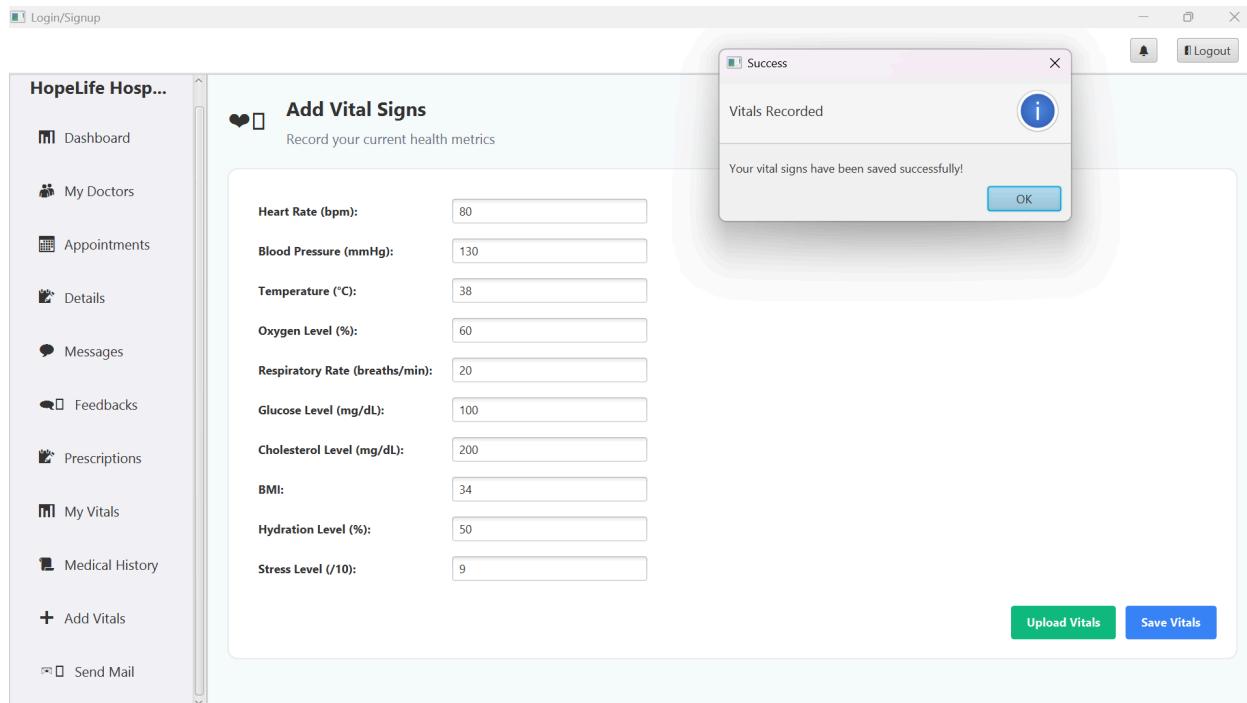


Figure 15: Vitals Recorded successful message

HopeLife Hosp...		MY VITAL SIGNS	
		Your recorded health metrics	
Heart Rate:	85 bpm	Heart Rate:	78 bpm
Blood Pressure:	130 mmHg	Blood Pressure:	130 mmHg
Oxygen Level:	95%	Oxygen Level:	99%
Temperature:	37.2°C	Temperature:	38.0°C
Respiratory Rate:	18 breaths/min	Respiratory Rate:	19 breaths/min
Glucose Level:	105.0 mg/dL	Glucose Level:	100.0 mg/dL
Cholesterol Level:	200.0 mg/dL		
BMI:	26.2		
Hydration Level:	72.0%		
Stress Level:	4/10		

Figure 16: Table view of historical vital sign entries.

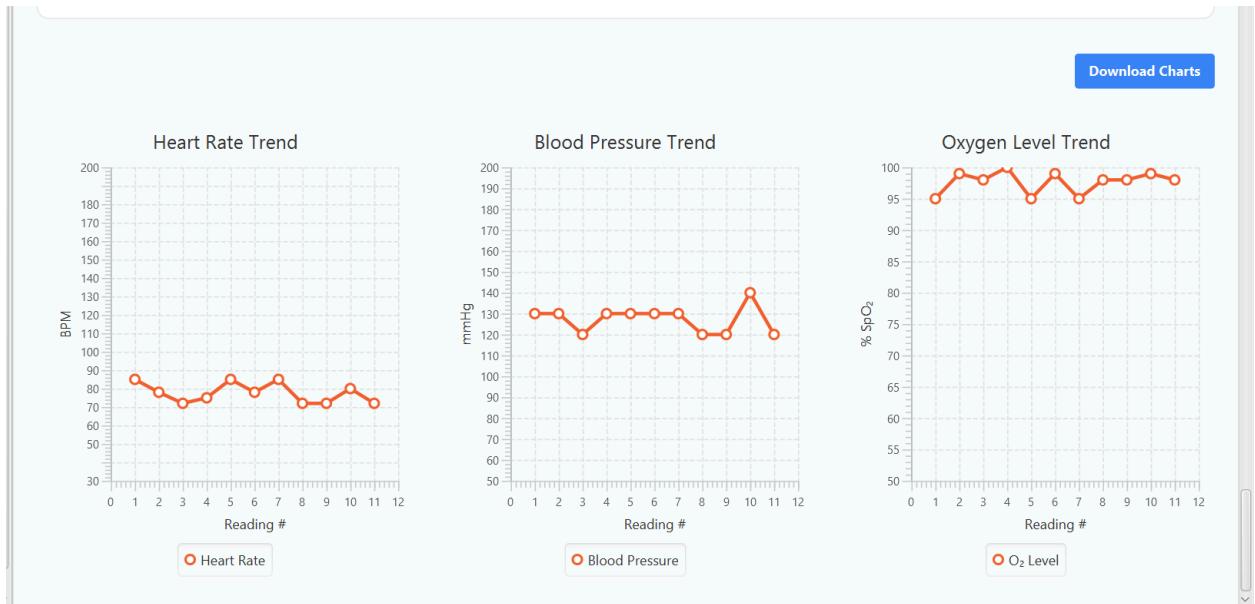


Figure 17: Line charts showing trends in vitals over time.

● Prescriptions & Records

The screenshot shows the "MY PRESCRIPTIONS" section of the app. It lists two prescriptions:

- PREScription #4ad7bf2a-5d3b-a891-904a51463aa5**
 - Medications:**
 - Instructions: Take 1 tablet orally twice a day after meals.
 - Schedule: Morning and Evening
 - Duration: 2025-05-09 to 2025-05-30
 - Total Quantity: 28 tablets
 - Prescribed by: Hamza Raheem
- PREScription #6fa3e7e6-02e9-4269-9e64-8c7ebfb80c2d**
 - Medications:**
 - Instructions: take 3 times a day
 - Schedule: Morning, Evening, Night
 - Duration: 2025-05-11 to 2025-05-29
 - Total Quantity: 20 Tablets Total
 - Prescribed by: Hamza Raheem

Figure 18: List of current and past prescriptions.

- **Communication**

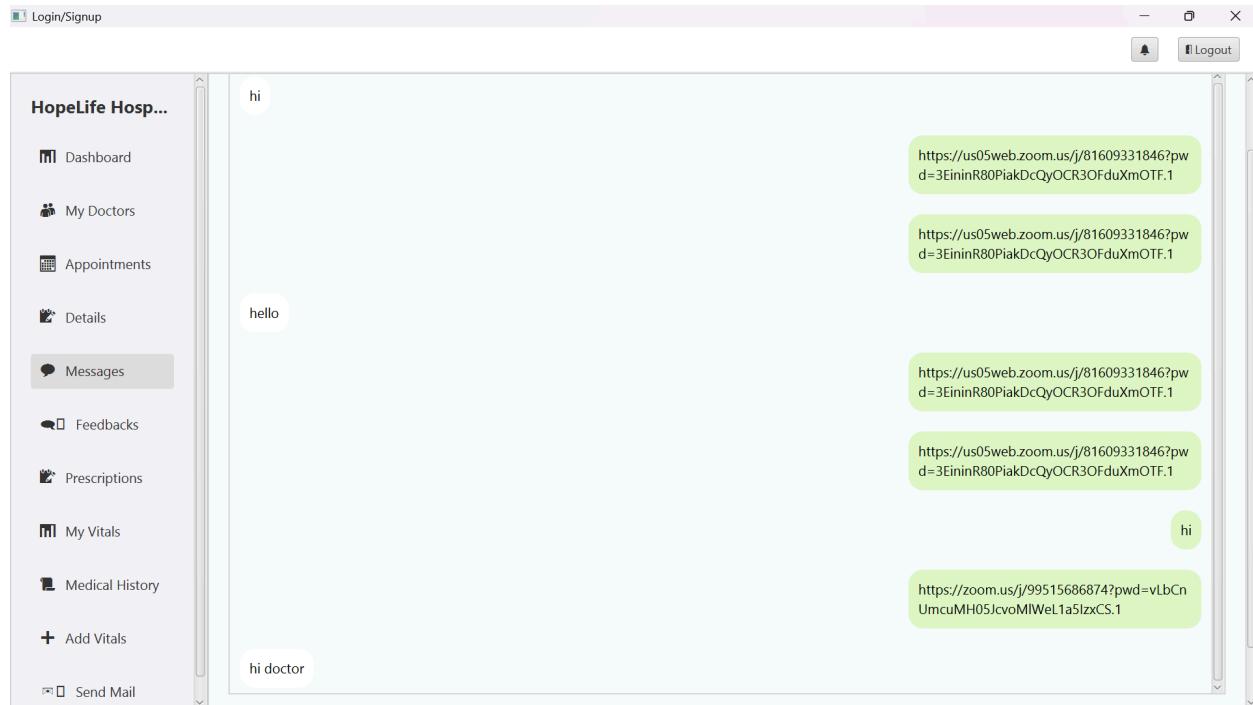


Figure 19: Patient's chat interface with doctor.

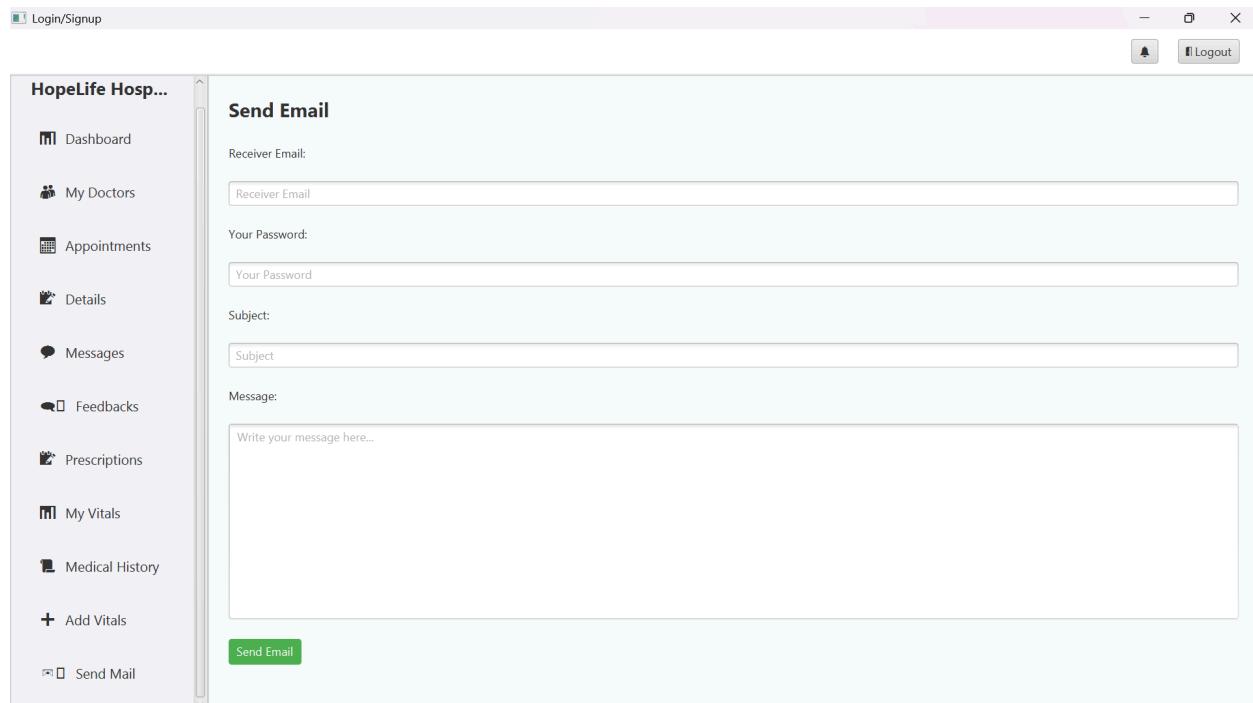


Figure 20: Email messaging screen for patient to send detailed queries.

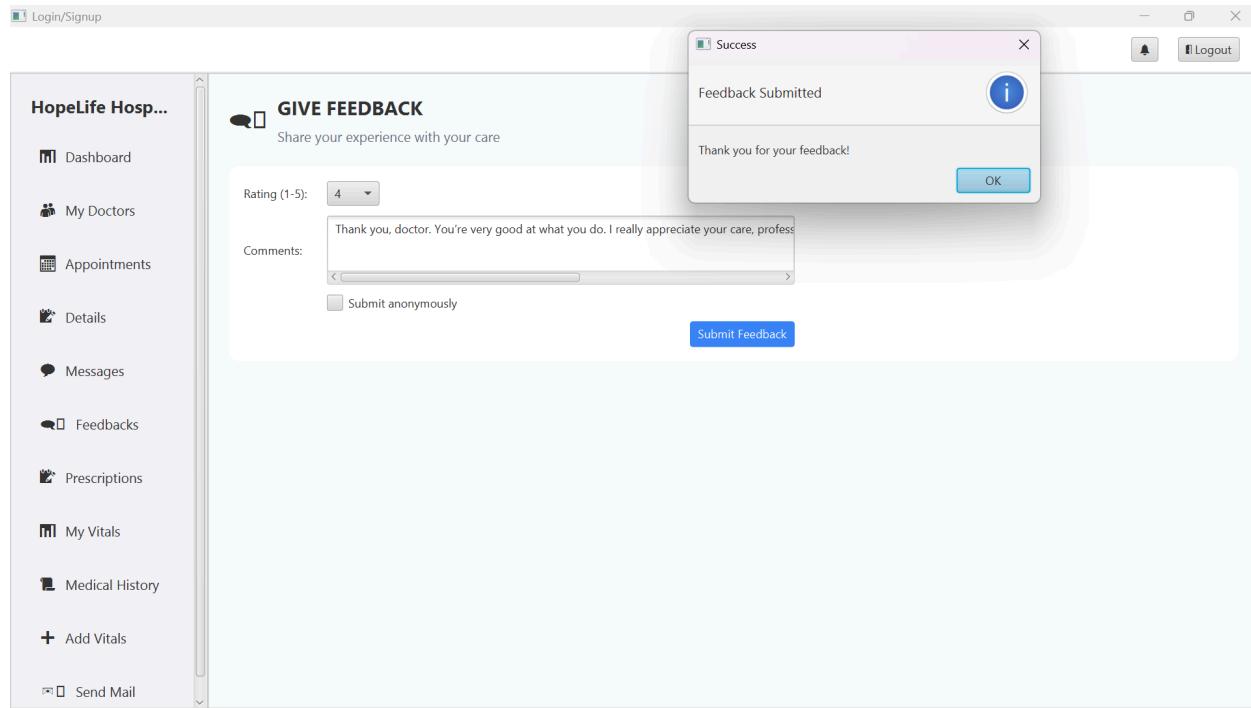


Figure 21: Patient adding feedback

Doctor Dashboard

View patient data, provide medical feedback, and schedule appointments.

- **Dashboard & Patient List**

Login/Signup

Logout

Hopelife Hospital

- [Dashboard](#)
- [Assigned Patients](#)
- [Appointments](#)
- [Details](#)
- [Messages](#)
- [Feedbacks](#)
- [Prescriptions](#)
- [Vitals](#)
- [Medical History](#)
- [Send Mail](#)

Doctor Dashboard - Feature Details

Dashboard: View an overview of your profile, availability, and key metrics.

Assigned Patients: View and manage the list of patients assigned to you.

Appointments: Check and manage your scheduled appointments.

Messages: Communicate with patients via the messaging system.

Feedbacks: View feedback provided by patients.

Prescriptions: Create and manage prescriptions for patients.

Vitals: Monitor and review patients' vital signs.

Medical History: Access the medical history of your patients.

Send Mail: Send emails to patients or other staff members.

For any queries, contact us: Email: support@hopelifehospital.com Phone: +1-800-123-4567

Figure 22: In-app help documentation for doctors.

Figure 23: Doctor's dashboard listing assigned patients and status

- **Appointment Workflow**

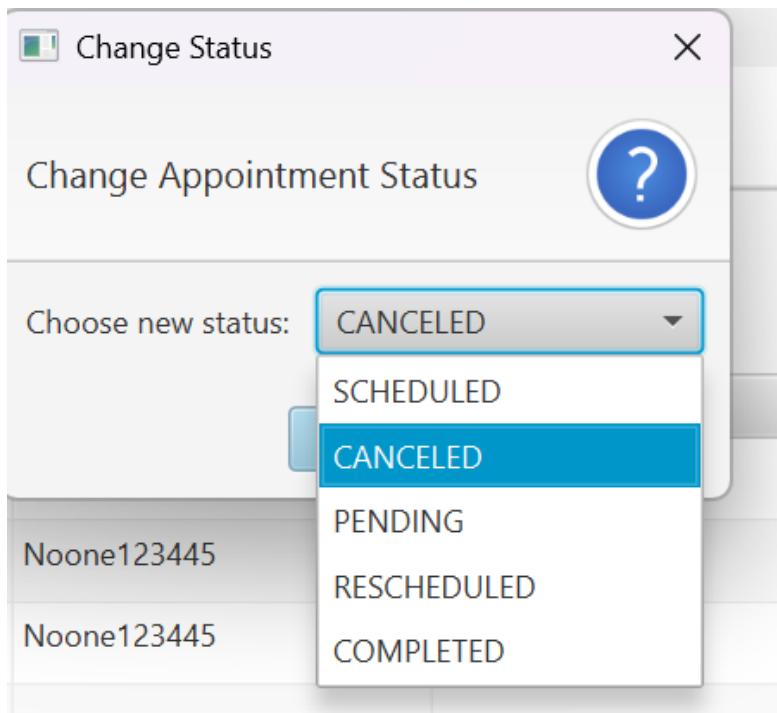


Figure 24: Interface for doctor to confirm or cancel appointment requests.

Figure 25: Doctor Appointment view

- **Clinical Actions**

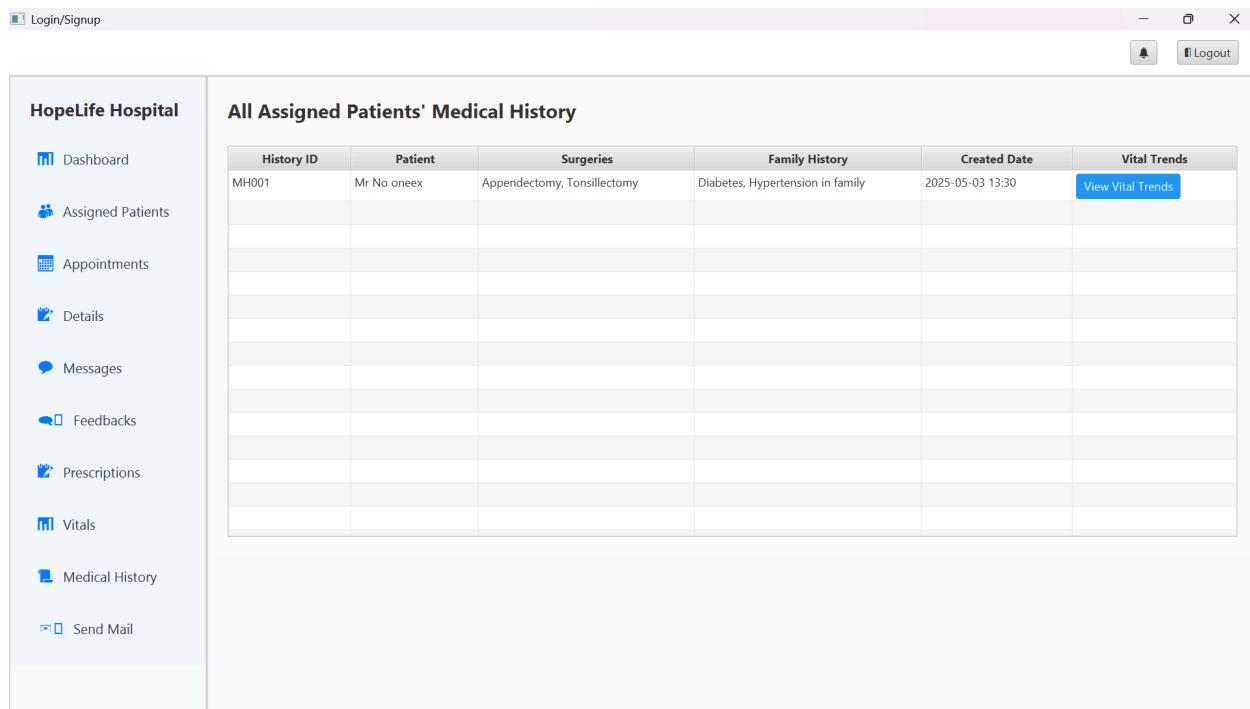


Figure 26: Medical History view for all patients the doctor has and their vital signs

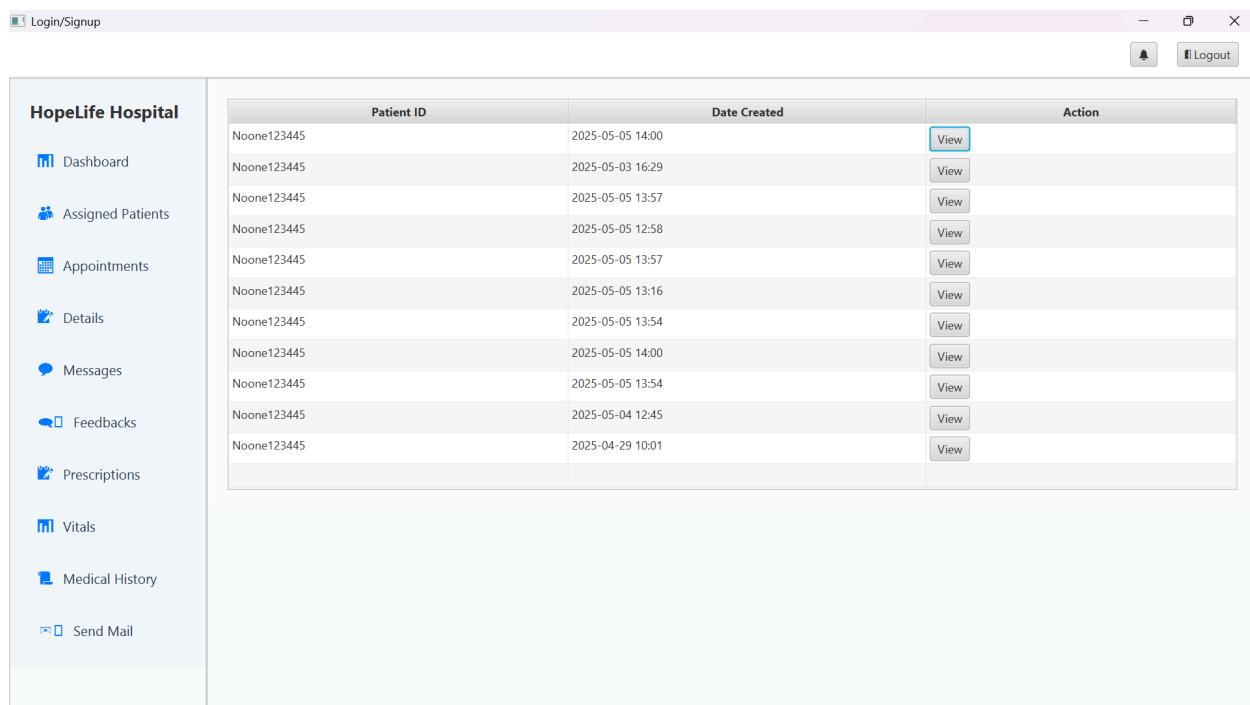


Figure 27: Doctor seeing vital signs

The screenshot shows a prescription entry interface. At the top right, there is a 'Success' message box with an 'i' icon that says 'Prescription added successfully!' with an 'OK' button. The main form has fields for Patient ID (Noone12345), Dosage Instructions (take 3 times a day), Start Date (5/11/2025), End Date (5/29/2025), Dosage Schedule (Morning, Evening, Night), and Quantity (20 Tablets Total). A 'Submit Prescription' button is at the bottom.

Figure 28: Prescription entry form with medication lookup.

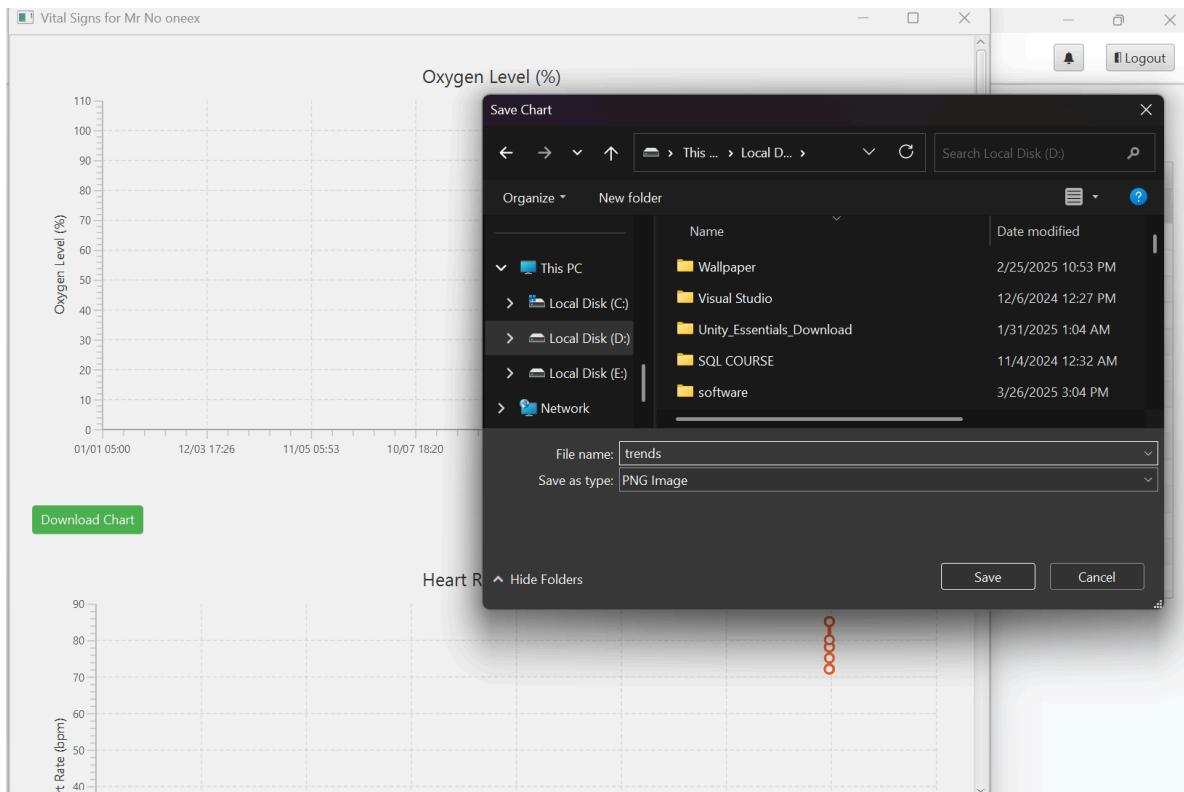


Figure 29: Doctor saving Medical History

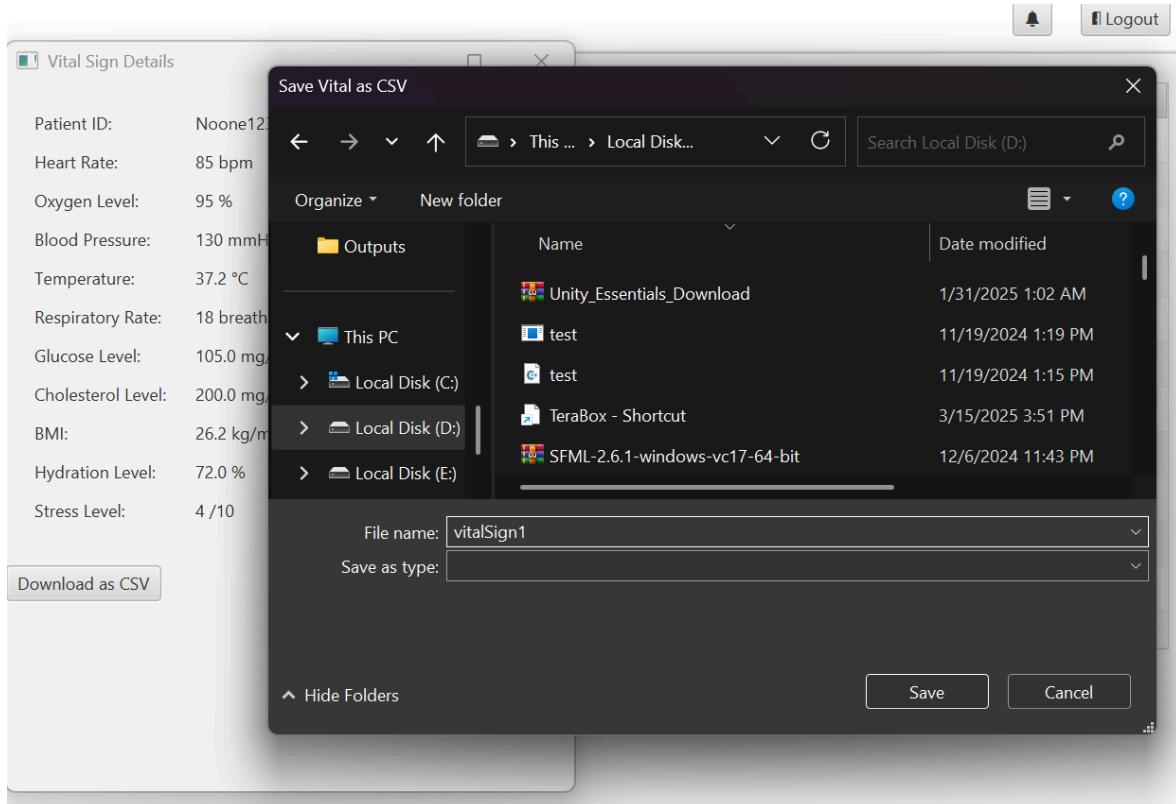


Figure 30: Doctor saving vitals

The screenshot shows the "Feedbacks" section of the "HopeLife Hospital" application. The left sidebar contains navigation links: Dashboard, Assigned Patients, Appointments, Details, Messages, Feedbacks, Prescriptions, Vitals, Medical History, and Send Mail. The main area is titled "Feedbacks" and displays a table of feedback entries. The table has columns: Date, Patient, Rating, Status, Feedback ID, and Change Status. One entry is visible:

Date	Patient	Rating	Status	Feedback ID	Change Status
2025-05-08 23:07	Noone12345	3	PENDING	8b140805-0d03-4ae2-80da-736d73689f53	<button>Set Status</button>

Below the table, there is a "Details:" label followed by a text input field.

Figure 31: Feedback view that status can be set for

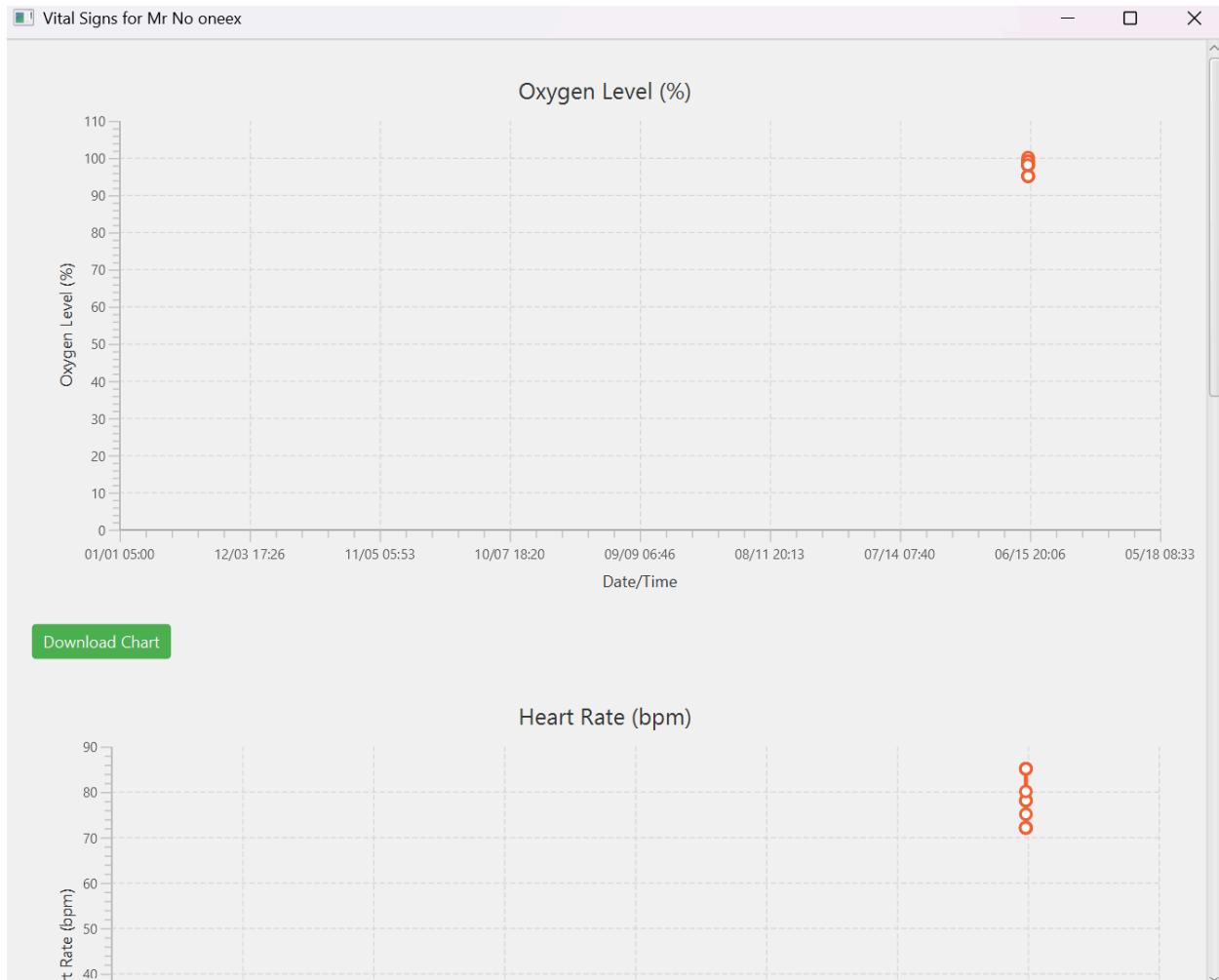


Figure 32: Charts of vital signs in medical history tab that can be downloaded. Visible to both the doctor and the patient

- **Communication**

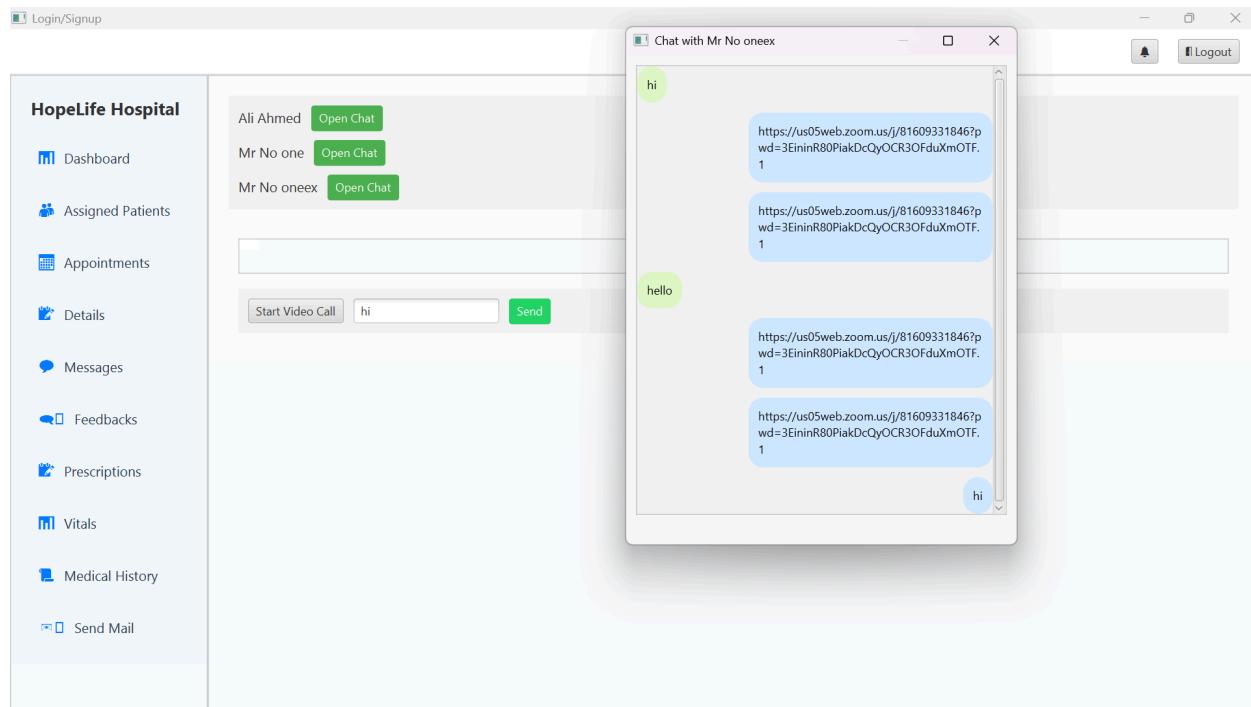


Figure 33: Real-time chat window for doctor to message patients.

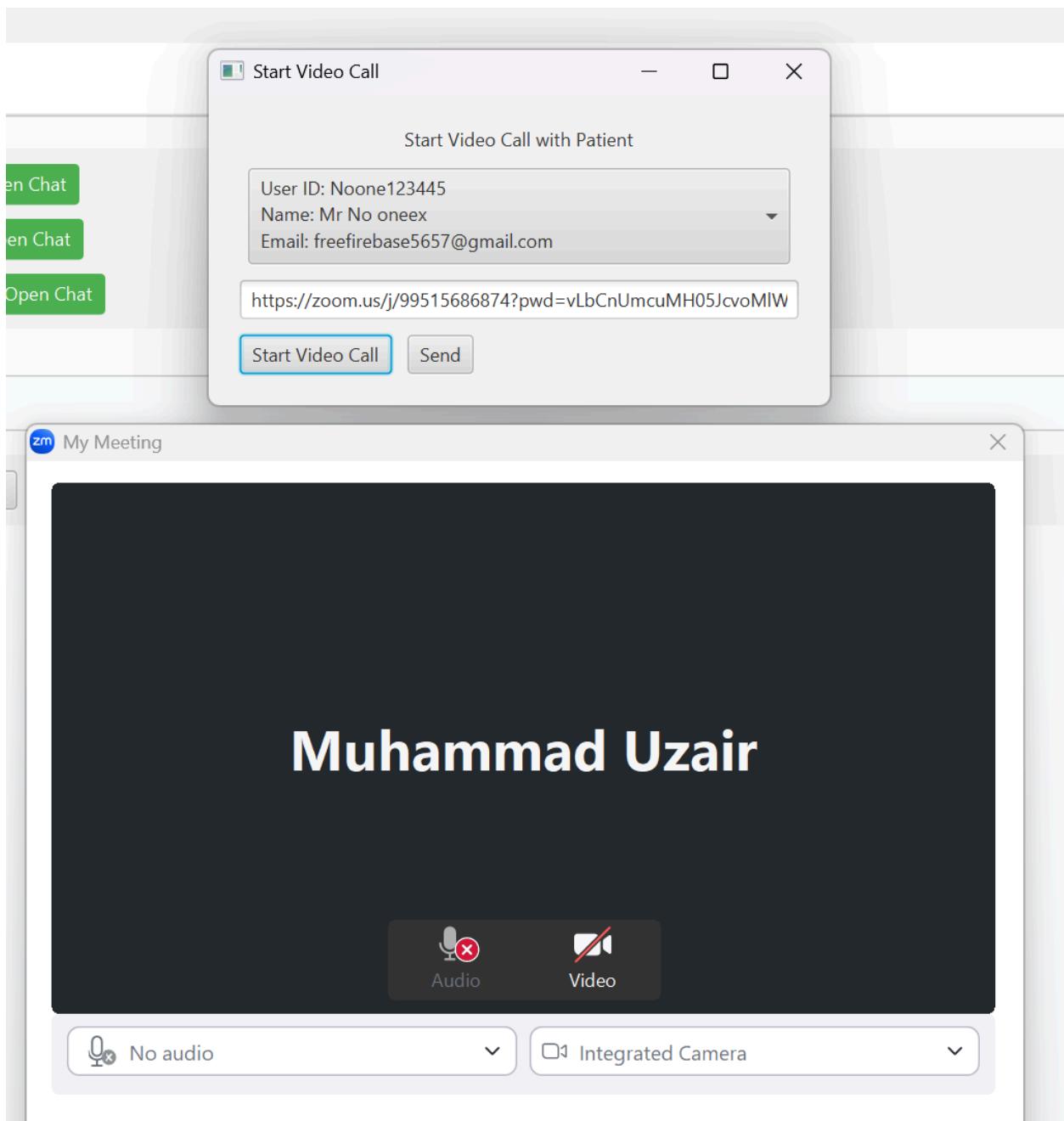


Figure 34: Doctor successful video-call

The screenshot shows the 'HopeLife Hospital' dashboard with a sidebar containing links like Dashboard, Assigned Patients, Appointments, Details, Messages, Feedbacks, Prescriptions, Vitals, Medical History, and Send Mail. The main area is titled 'Send Email' and includes fields for Receiver Email, Your Password, Subject, and Message, along with a 'Send Email' button.

Figure 35: Real time email integration on both the patient and doctor dashbaord

- **Notifications**

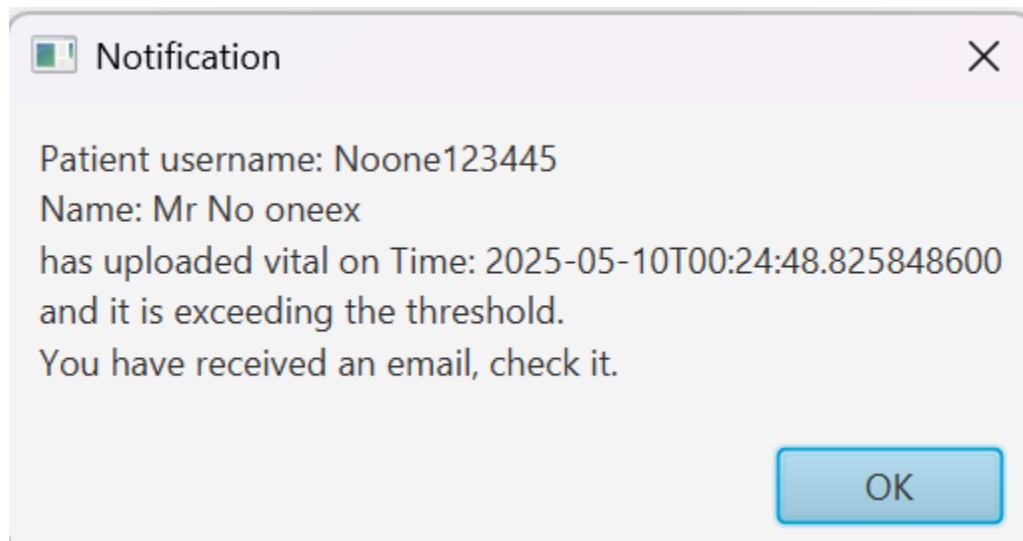


Figure 36: In-app notification and email alert sent to doctor for critical vitals.

Admin Dashboard

Manage user accounts, view system logs, and generate system reports.

- **Dashboard Overview**

The screenshot shows the Admin Dashboard for HopeLife Hospital. The left sidebar contains navigation links: Dashboard, Manage Patients, Manage Doctors, Details, Manage Appointments, and Send Reminders. The main area displays a user profile for Ahmed Ali, System Administrator, with contact information: ahmedali123@gmail.com and 0321432552. Below the profile is a summary of system metrics:

System Overview	
Total Patients:	4
Total Doctors:	2
Total Appointments:	2

Figure 37: Administrator overview with system-wide metrics.

● User Management

The screenshot shows the 'Manage Patients' section of the admin interface. On the left, there's a sidebar with a 'HopeLife Hospital' logo and links for Dashboard, Manage Patients, Manage Doctors, Details, Manage Appointments, and Send Reminders. The main area is titled 'Manage Patients' with the sub-instruction 'View and manage all patients'. It features a table with columns: Name, ID, Email, Phone, Status, and Delete. The table contains four rows of patient data, each with a red 'Inactive' status button and a red 'Delete' button.

Name	ID	Email	Phone	Status	Delete
Ahmed Khan	ahmed1234	ahmed12345@gmail.com	0324325433	Inactive	<button>Delete</button>
Ali Ahmed	aliahmed1234	aliahmed123@gmail.com	01324534654	Inactive	<button>Delete</button>
Mr No one	noone1234	noone123@gmail.com	032454353	Inactive	<button>Delete</button>
Mr No oneex	Noone123445	freefirebase5657@gmail.com	0324543533	Inactive	<button>Delete</button>

Figure 38: Admin interface to view, edit, or remove patient accounts.

The screenshot shows the 'Manage Doctors' section of the admin interface. The sidebar is identical to Figure 38. The main area is titled 'Manage Doctors' with the sub-instruction 'View and manage all doctors'. It features a table with columns: Name, ID, Email, Phone, Status, and Delete. The table contains two rows of doctor data, each with a red 'Inactive' status button and a red 'Delete' button.

Name	ID	Email	Phone	Status	Delete
Ahmed Khan	Ahmed1234	ahmed12345@gmail.com	0324325433	Inactive	<button>Delete</button>
Hamza Raheem	hamzarahmeem23	muzair.bsds24seecs@seecs.edu.pk	0234568243	Inactive	<button>Delete</button>

Figure 39: Admin interface for managing doctor profiles and permissions.

- **Appointment Management**

The screenshot shows a web-based application interface for managing appointments. On the left is a sidebar with the title "HopeLife Hospital" and a list of navigation items: Dashboard, Manage Patients, Manage Doctors, Details, Manage Appointments, and Send Reminders. The main content area is titled "Manage Appointments" and contains a table of scheduled appointments. The table has columns for Date/Time, Patient ID, Doctor ID, Appointment ID, and Actions. Two rows are visible in the table:

Date/Time	Patient ID	Doctor ID	Appointment ID	Actions
2025-05-06 21:30	Noone123445	hamzarahmeem23	app122	<button>Cancel</button>
2025-05-02 14:30	Noone123445	hamzarahmeem23	appt001	<button>Cancel</button>

Figure 40: Admin view of all appointment requests and statuses.

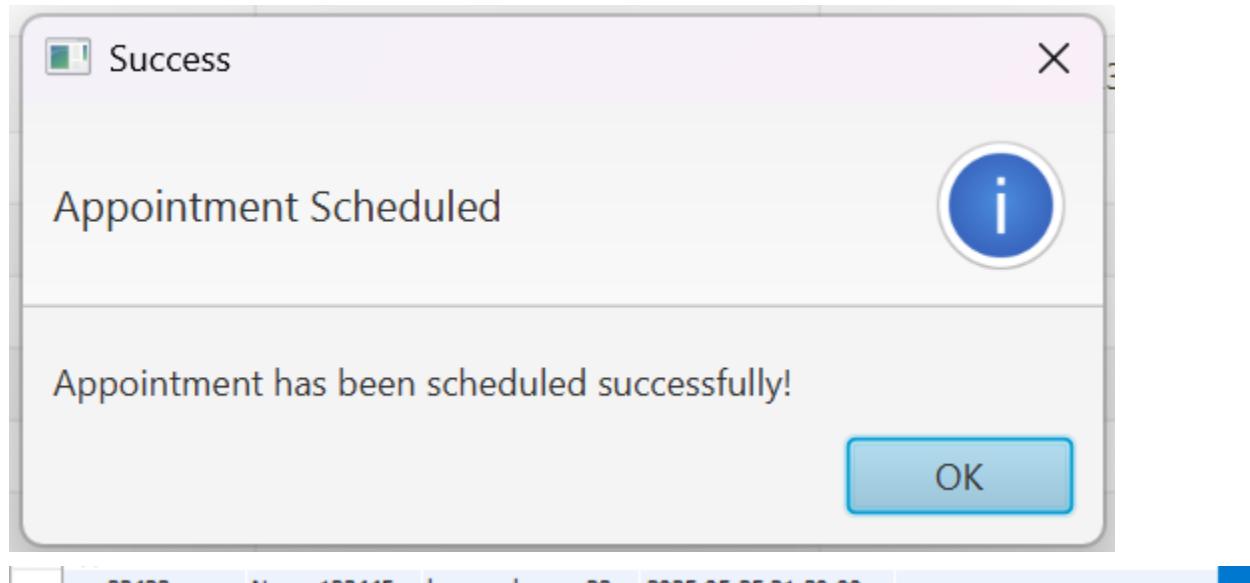


Figure 41: Confirmation message after successfully updating an appointment.

- **Feedback & Help**

The screenshot shows the HopeLife Hospital software interface. On the left is a sidebar with various menu items: Dashboard, Assigned Patients, Appointments, Details, Messages, Feedbacks, Prescriptions, Vitals, Medical History, and Send Mail. The main area is titled 'Feedbacks' and displays a table with columns: Date, Patient, Rating, Status, Feedback ID, and Change Status. A modal dialog titled 'Change Feedback Status' is open, showing a dropdown menu with options: PENDING (selected), PENDING, APPROVED, and REJECTED. The background table has one visible row with the following data:

Date	Patient	Rating	Status	Feedback ID	Change Status
2025-05-08 23:07				80da-736d73689f53	<button>Set Status</button>

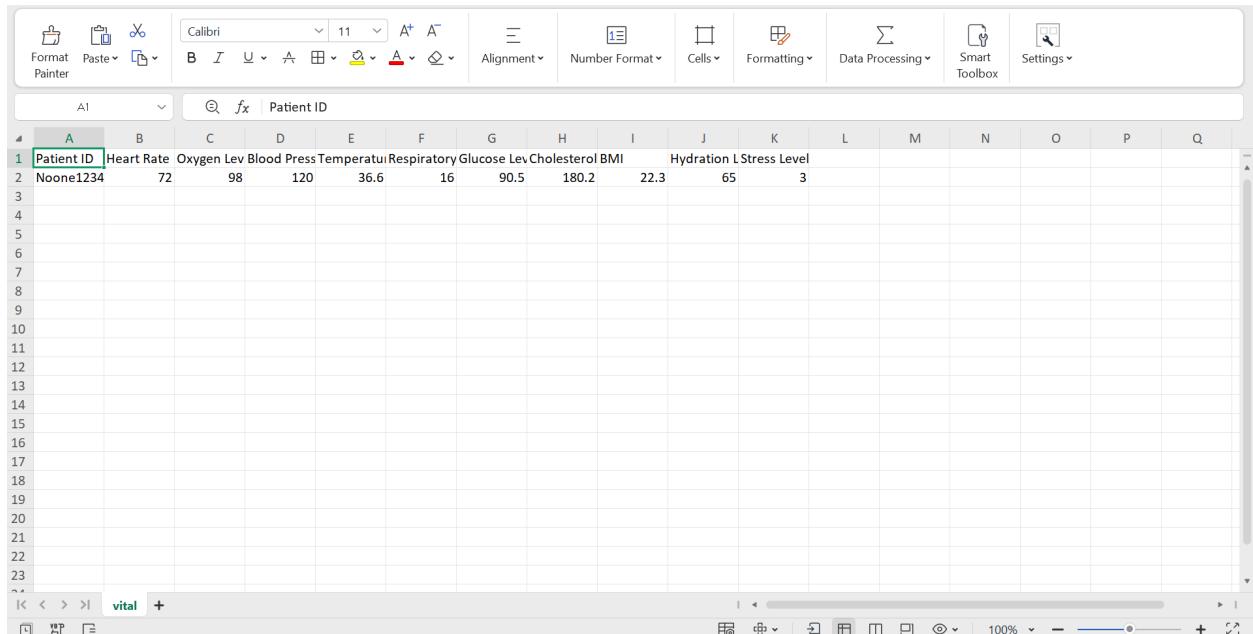
Below the table, there is a 'Details:' field with an empty text input.

Figure 42: Admin changing feedback status for clinical notes.

Database

	feedbackID	doctorID	patientID	rating	comments
▶	0ba0a4a0-b3a0-48a8-ad4e-cef7dc8455af	hamzarahmeem23	Noone123445	4	Thank you, doctor.

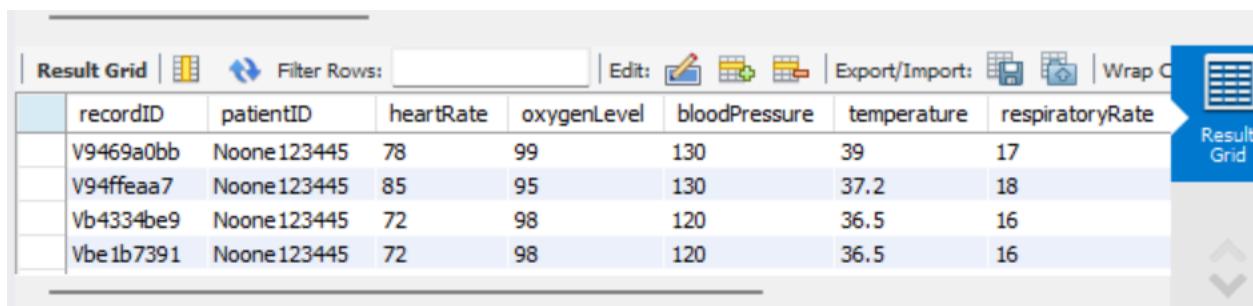
Figure 43: Feedback success on database



A screenshot of Microsoft Excel showing a CSV file named "vital". The spreadsheet has a header row with columns for Patient ID, Heart Rate, Oxygen Lev, Blood Press, Temperature, Respiratory, Glucose, LevCholesterol, and BMI. Below the header, there is one data row for "Noone1234" with values 72, 98, 120, 36.6, 16, 90.5, 180.2, 22.3, 65, and 3 respectively. The formula bar shows "Patient ID" and "fx". The ribbon menu is visible at the top.

Patient ID	Heart Rate	Oxygen Lev	Blood Press	Temperature	Respiratory	Glucose	LevCholesterol	BMI	Hydration	Stress Level
Noone1234	72	98	120	36.6	16	90.5	180.2	22.3	65	3

Figure 44: SS of CSV file



A screenshot of a database application showing a grid of vital sign data. The columns are labeled recordID, patientID, heartRate, oxygenLevel, bloodPressure, temperature, and respiratoryRate. There are four rows of data: V9469a0bb, Noone123445, 78, 99, 130, 39, 17; V94ffea7, Noone123445, 85, 95, 130, 37.2, 18; Vb4334be9, Noone123445, 72, 98, 120, 36.5, 16; and Vbe1b7391, Noone123445, 72, 98, 120, 36.5, 16. A sidebar on the right is titled "Result Grid".

recordID	patientID	heartRate	oxygenLevel	bloodPressure	temperature	respiratoryRate
V9469a0bb	Noone123445	78	99	130	39	17
V94ffea7	Noone123445	85	95	130	37.2	18
Vb4334be9	Noone123445	72	98	120	36.5	16
Vbe1b7391	Noone123445	72	98	120	36.5	16

Figure 45: Vitals success on database

6fa3e7e6-02e9-4269-9e64-8c7ebfb80c2d	Noone123445	hamzarahmeem23	take 3 times a day
--------------------------------------	-------------	----------------	--------------------

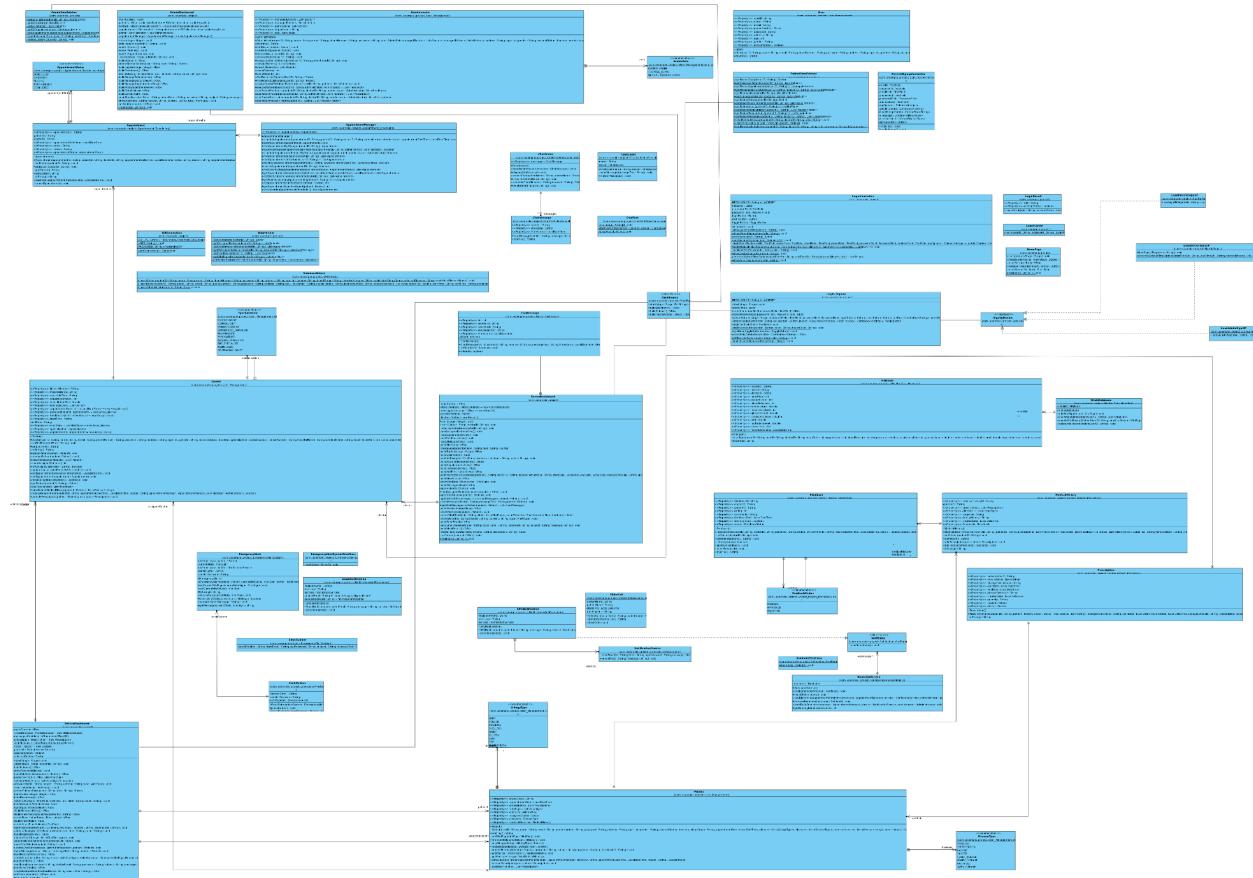
Figure 46: Prescription success on database

Future Extension

In the future, we plan to enhance the Hospital Management System by integrating AI to provide preliminary diagnostic support. During the summer, an AI module will be added that can analyze basic patient symptoms and suggest possible diseases, improving the efficiency of early diagnosis. Additionally, since the system already has a functional backend, we aim to develop a mobile application or a web-based interface to make the system more accessible and realistic for real-world use.

UML Diagrams

click on the image to view it zoomed in



CSV Samples

Sample files have been attached in the zip. Additionally the gui screenshots can be seen for their format/layout for recreation purposes.

Flowchart

click on the image to view it zoomed in

