Electronic Health Record (EHR) Information System

Author: Dipendra Thapa
Course: Information Systems in Health Care (WS25/26)
Submission Date: 18-01-2026

Abstract

Electronic Health Records (EHRs) have become essential in modern healthcare systems as they allow secure and comprehensive storage and retrieval of patient data. This project develops an EHR system tailored for doctors, providing functionality to manage patient records efficiently. The system enables doctors to register, log in, and perform CRUD operations on patient information, including demographic data, medical history, laboratory results, medications, immunization status, diagnostic reports, and billing information. The backend is implemented using Python Flask and MySQL, while the frontend uses HTML5, CSS3, and Bootstrap 5 to provide a responsive and intuitive interface. Image upload functionality and date selection are integrated to improve usability. A brief usability evaluation using a standard questionnaire demonstrates the system's effectiveness and ease of use. This report describes the system's design, technical architecture, functionalities, and evaluation, along with future prospects for improvement.

1. Introduction

Electronic Health Records are digital versions of patients' medical histories and are increasingly replacing traditional paper records. EHRs centralize critical information, making it accessible to authorized healthcare professionals and improving patient care. They enhance communication between providers, reduce the likelihood of medical errors, and streamline administrative processes. Moreover, EHRs facilitate data-driven decision-making, allowing healthcare providers to identify patterns, monitor patient outcomes, and plan preventive care.

The EHR system presented in this project was developed from scratch to serve the perspective of a single doctor managing patients independently. Unlike large-scale hospital systems, this system focuses on the essential functionalities required to manage patient data efficiently, ensuring that each doctor can view and control only their patients' records. The system is secure, easy to use, and scalable. It integrates common features such as demographic data entry, health condition checkboxes, radio buttons for gender selection, date pickers, and image uploads for medical reports. The EHR system also includes the ability to maintain detailed medical history, diagnoses, treatment plans, and billing information, providing a comprehensive solution for small-scale healthcare environments.

Historically, EHRs have evolved from simple patient information repositories to complex systems integrating laboratory results, imaging studies, prescriptions, and decision support systems. The development of this project builds on this evolution, implementing modern web technologies while maintaining simplicity and usability. The chosen architecture combines a Python Flask backend

with MySQL database storage and a responsive HTML5 and Bootstrap 5 frontend, allowing the system to run reliably in different environments. This report further details the system design, technical architecture, functional capabilities, and evaluation results.

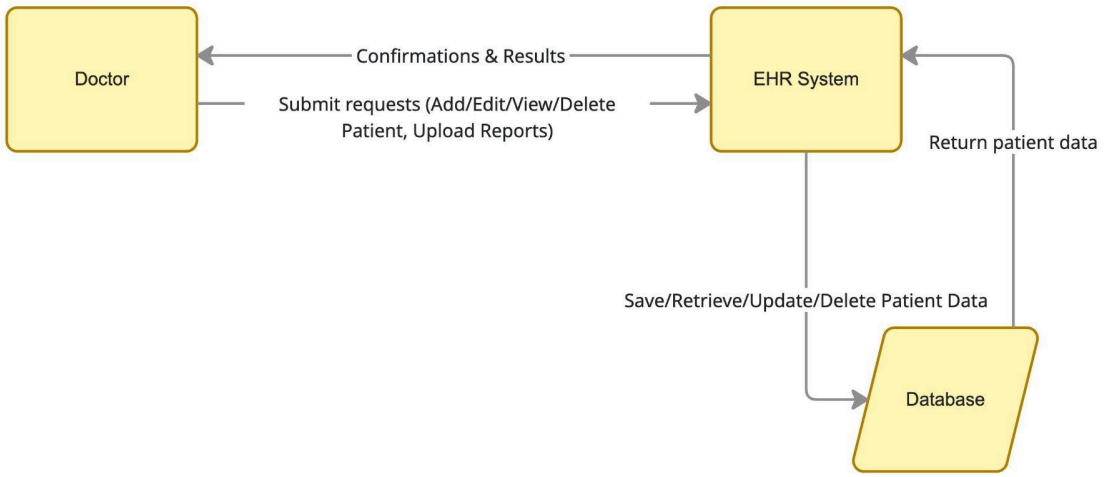


Figure 1 – Conceptual diagram showing doctor-patient data flow

2. Website Design and Technical Architecture

The EHR system is structured around three main components: the frontend, the backend, and the database. The frontend provides the interface through which doctors interact with the system. It is built using HTML5, CSS3, and Bootstrap 5, offering a responsive and consistent layout across devices. The interface includes a navigation menu for static informational pages such as About and Contact, as well as dynamic pages for registration, login, and patient management.

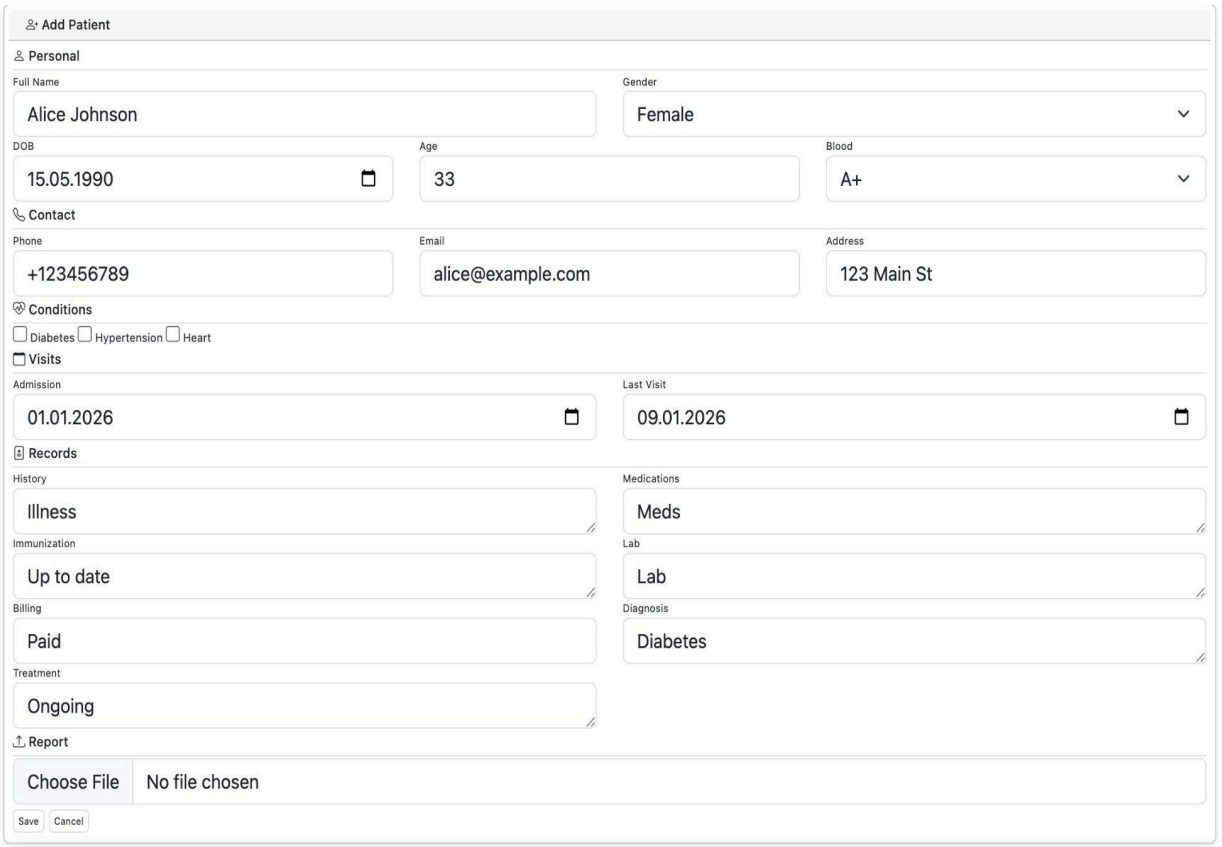
The backend is implemented with Python Flask (version 3.0.0) and handles all server-side logic, including routing, session management, form processing, and data validation. Flask-Mail (version 0.9.1) is incorporated to manage email communications, including account confirmation and password reset. The backend ensures that only authorized doctors can access their patient records and provides secure handling of sensitive data, such as medical histories and lab results. All input data are validated to prevent erroneous entries and ensure data integrity.

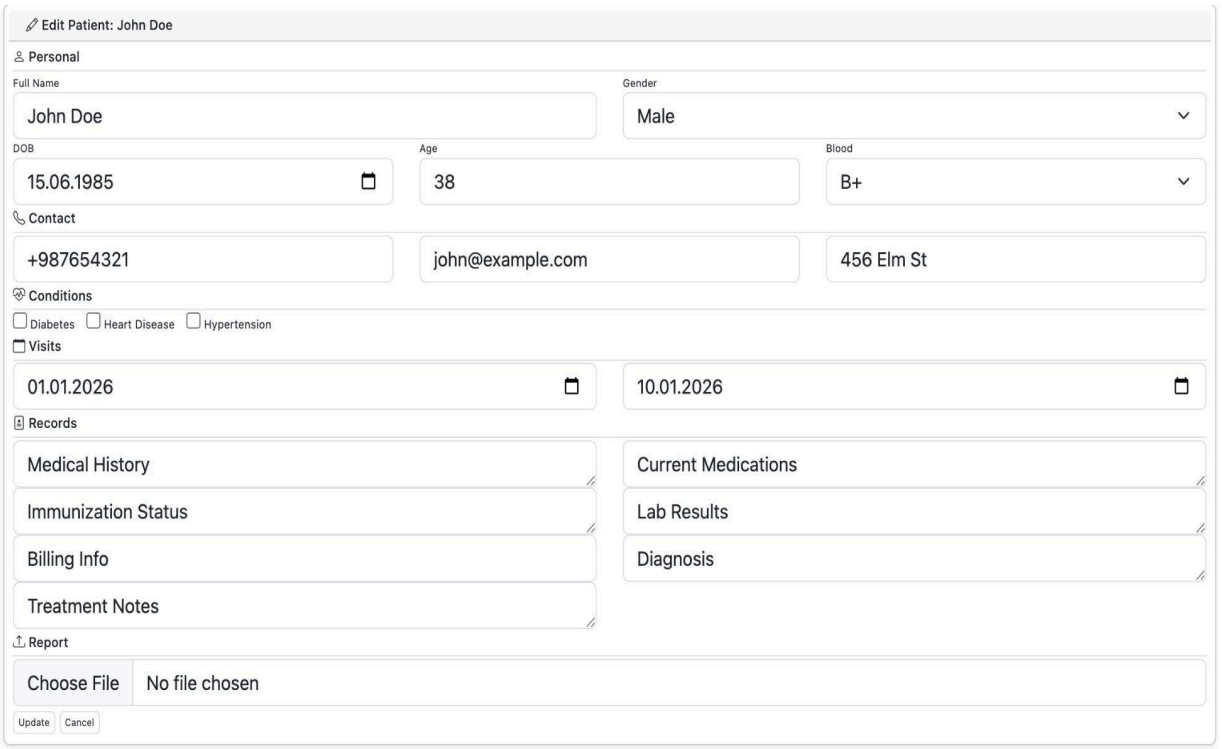
Data storage is managed using MySQL (version 8.0.44). The database schema includes tables for doctors, patients, SUS responses, and contact inquiries. Each patient record is linked to a specific doctor, ensuring access control. The patients table stores demographic data, medical history, laboratory results, medications, immunization status, diagnostic reports, and billing information. Uploaded images are stored in the server file system, with the file path saved in the database. The database also contains indexes on frequently queried columns, improving performance for large datasets.

The system architecture ensures a clean separation between the frontend, backend, and database, supporting maintainability and scalability. Doctors interact with the frontend through a web browser, and the backend processes requests, communicates with the database, and returns responses. This architecture allows the system to be extended with additional features, such as

reporting and data analytics, in the future.

Figure 2 – System architecture diagram showing frontend, backend, and database interactions


Add Patient
Personal
Full Name
Alice Johnson
Gender
Female
DOB
15.05.1990
Age
33
Blood
A+
Contact
Phone
+123456789
Email
alice@example.com
Address
123 Main St
Conditions
Diabetes
Hypertension
Heart
Visits
Admission
01.01.2026
Last Visit
09.01.2026
Records
History
Illness
Medications
Meds
Immunization
Up to date
Lab
Lab
Billing
Paid
Diagnosis
Diabetes
Treatment
Ongoing
Report
Choose File
No file chosen
Save
Cancel


Edit Patient: John Doe
Personal
Full Name
John Doe
Gender
Male
DOB
15.06.1985
Age
38
Blood
B+
Contact
+987654321
john@example.com
456 Elm St
Conditions
Diabetes
Heart Disease
Hypertension
Visits
01.01.2026
10.01.2026
Records
Medical History
Current Medications
Immunization Status
Lab Results
Billing Info
Diagnosis
Treatment Notes
Report
Choose File
No file chosen
Update
Cancel

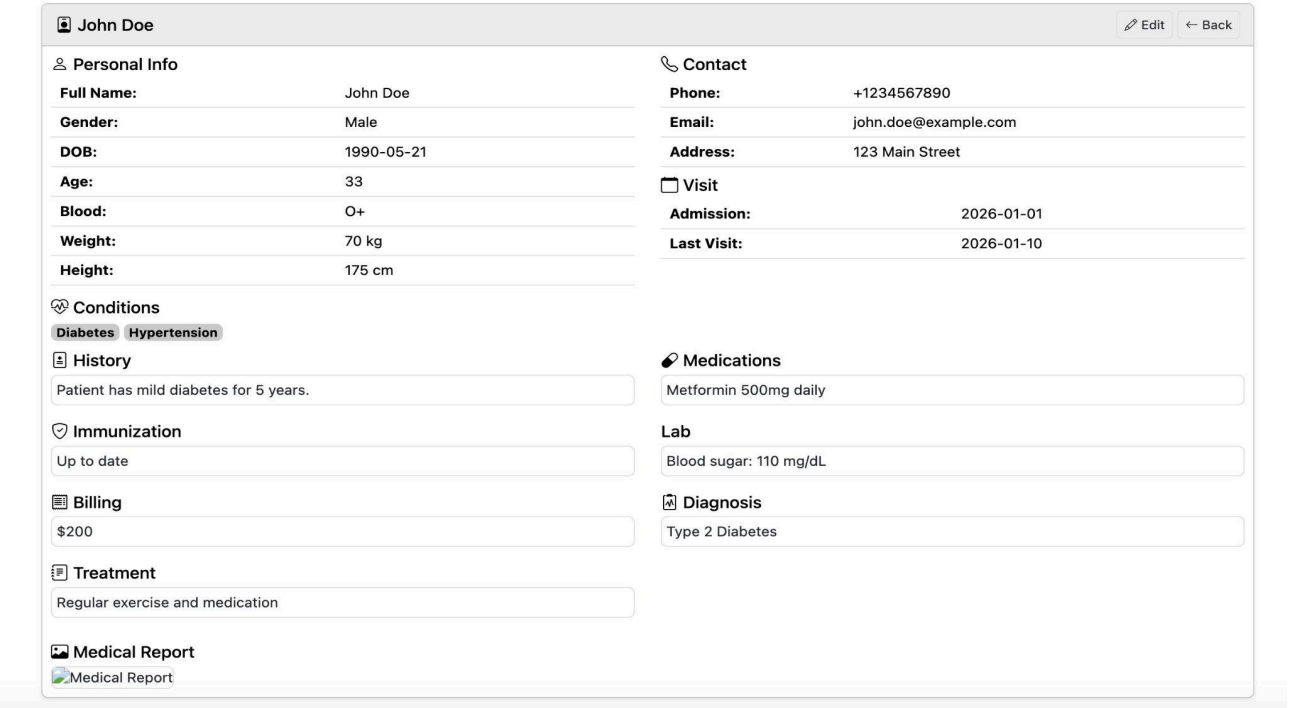
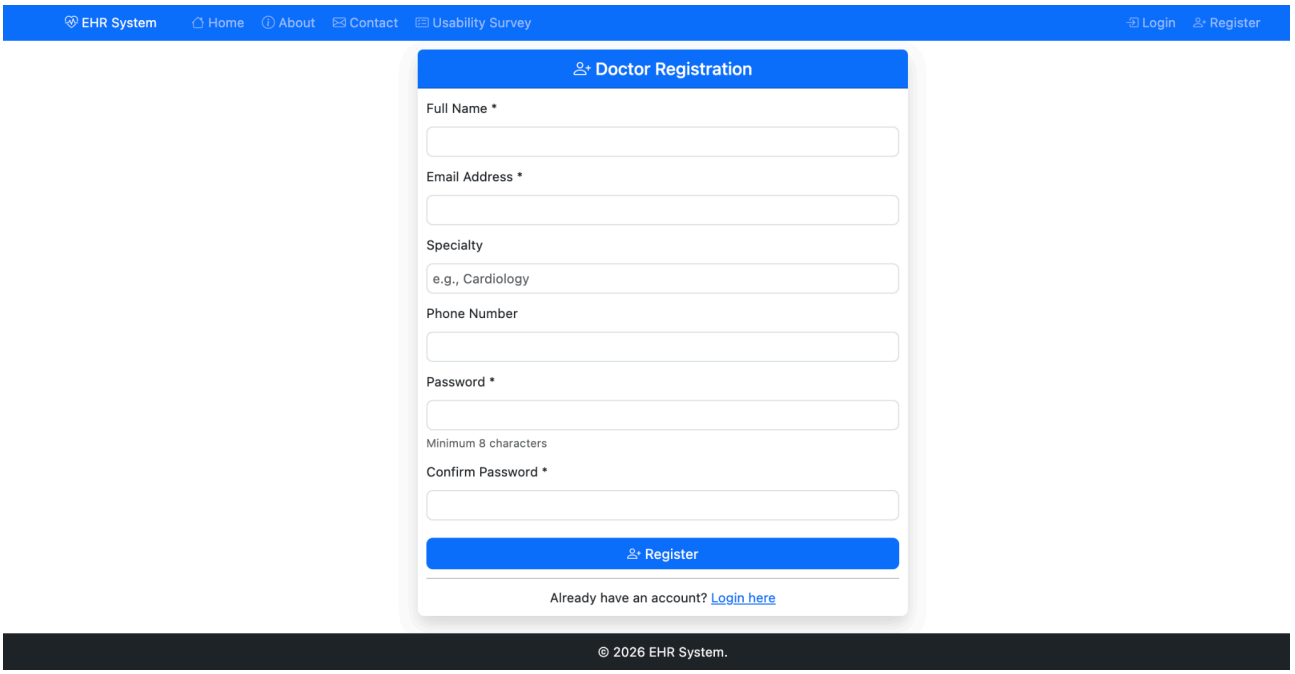


Figure 3 – Wireframe of dashboard and patient CRUD pages

3. EHR Information System – Functionality and User Manual

3.1 Registration and Login

Doctors begin by registering for an account, providing their full name, email, and password. Optional information such as specialty and phone number can also be entered. A confirmation email is sent to the doctor, and the account becomes active after verification. Strong password policies are enforced to enhance security. The login page allows registered doctors to enter their credentials to access the system, while the forgot password feature enables secure password recovery.



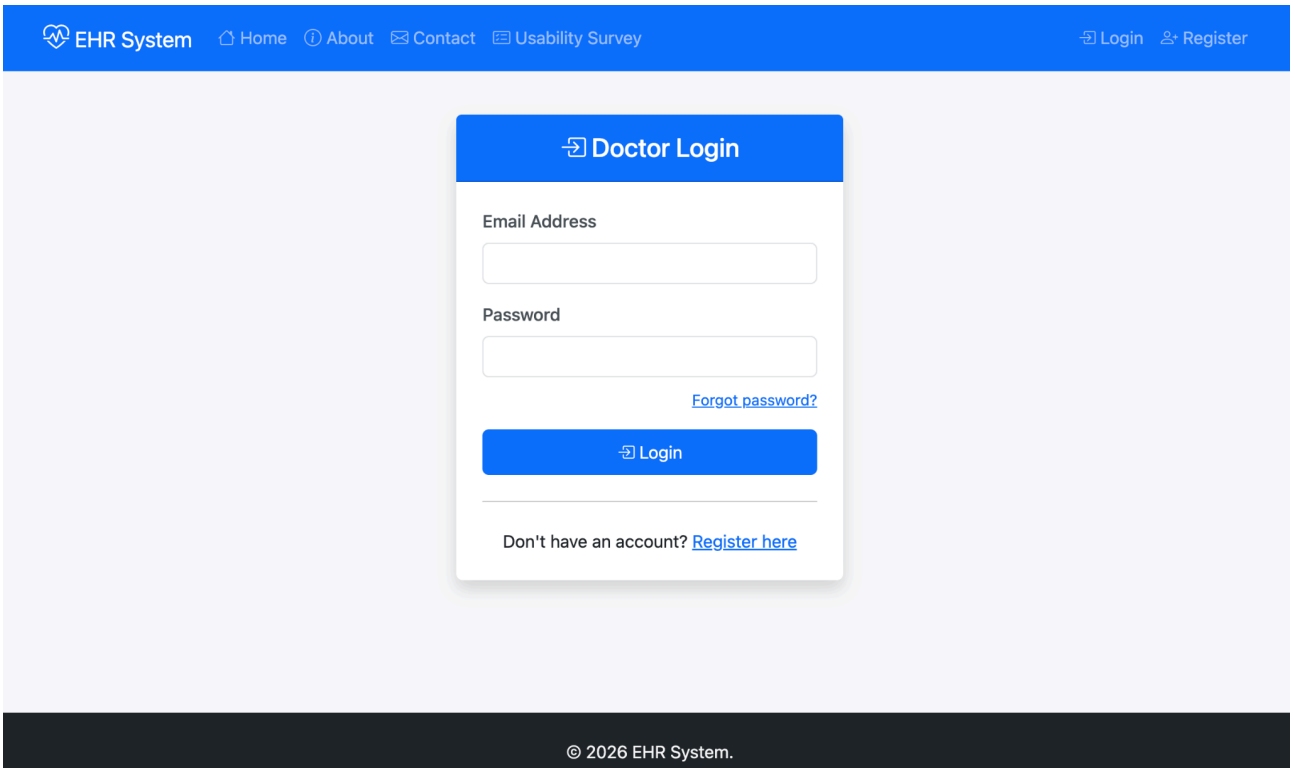


Figure 4: Screenshot of registration and login pages

3.2 Dashboard

Once logged in, doctors access a dashboard that provides a summary of their patient records, quick access to CRUD operations, and links to static informational pages. The dashboard is designed to be simple, responsive, and intuitive, allowing doctors to navigate between patient management and system features easily.

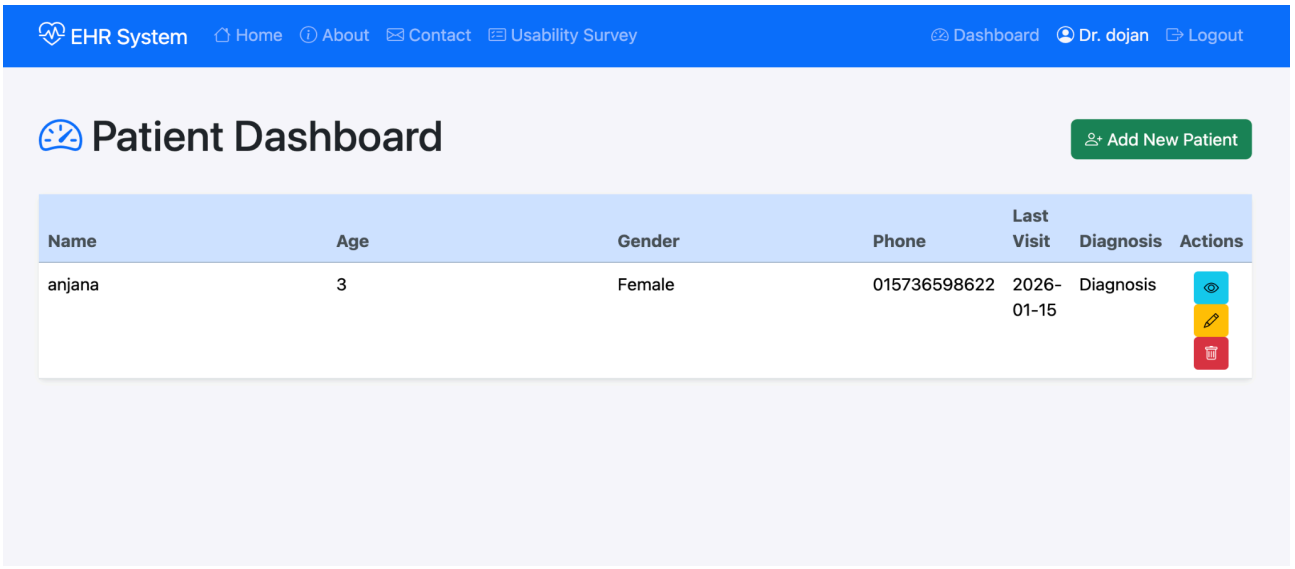
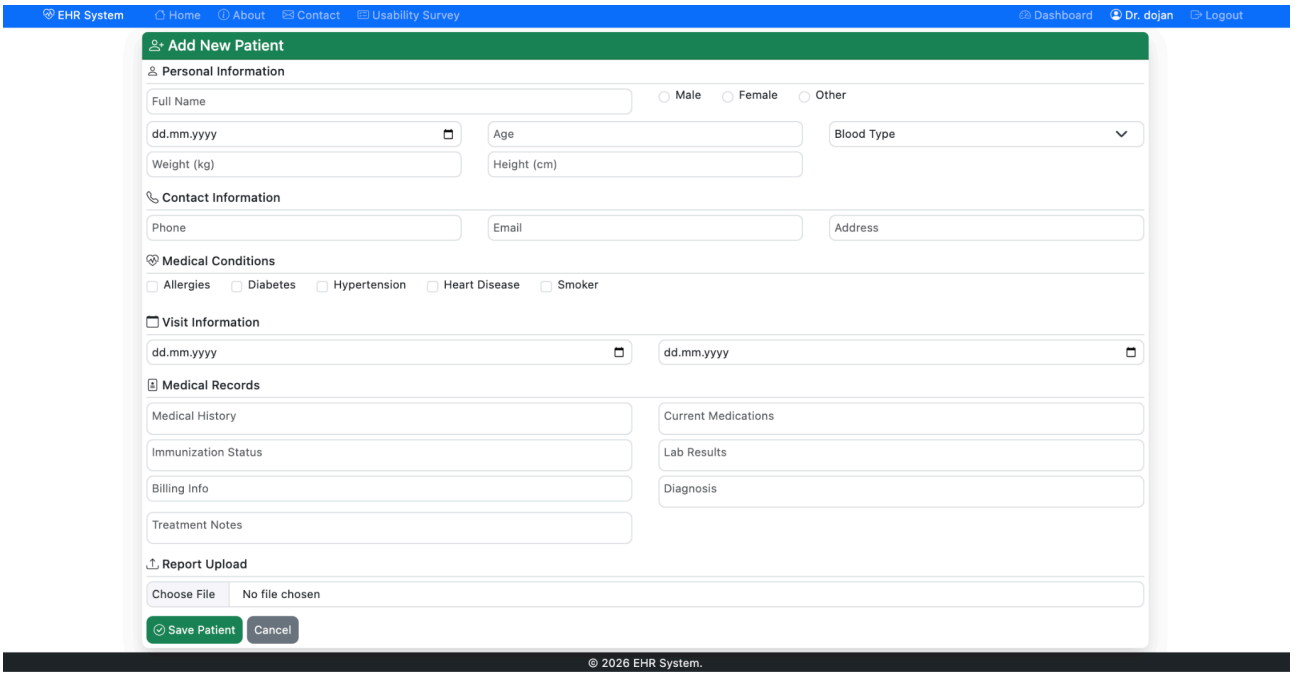


Figure 5: Screenshot of dashboard

3.3 Patient Management

Doctors can perform full CRUD operations on patient records. Adding a new patient involves entering information such as name, age, weight, height, blood type, gender, health conditions via checkboxes, dates of birth, admission, and last visit, text areas for medical history, medications, diagnosis, treatment notes, and uploading medical reports as images. Editing a patient record pre-fills the existing information, allowing updates to any field while preserving previously entered data. Viewing a patient record presents all information in a read-only format, including uploaded images, for quick reference. Deletion of a patient record requires confirmation to prevent accidental data loss.



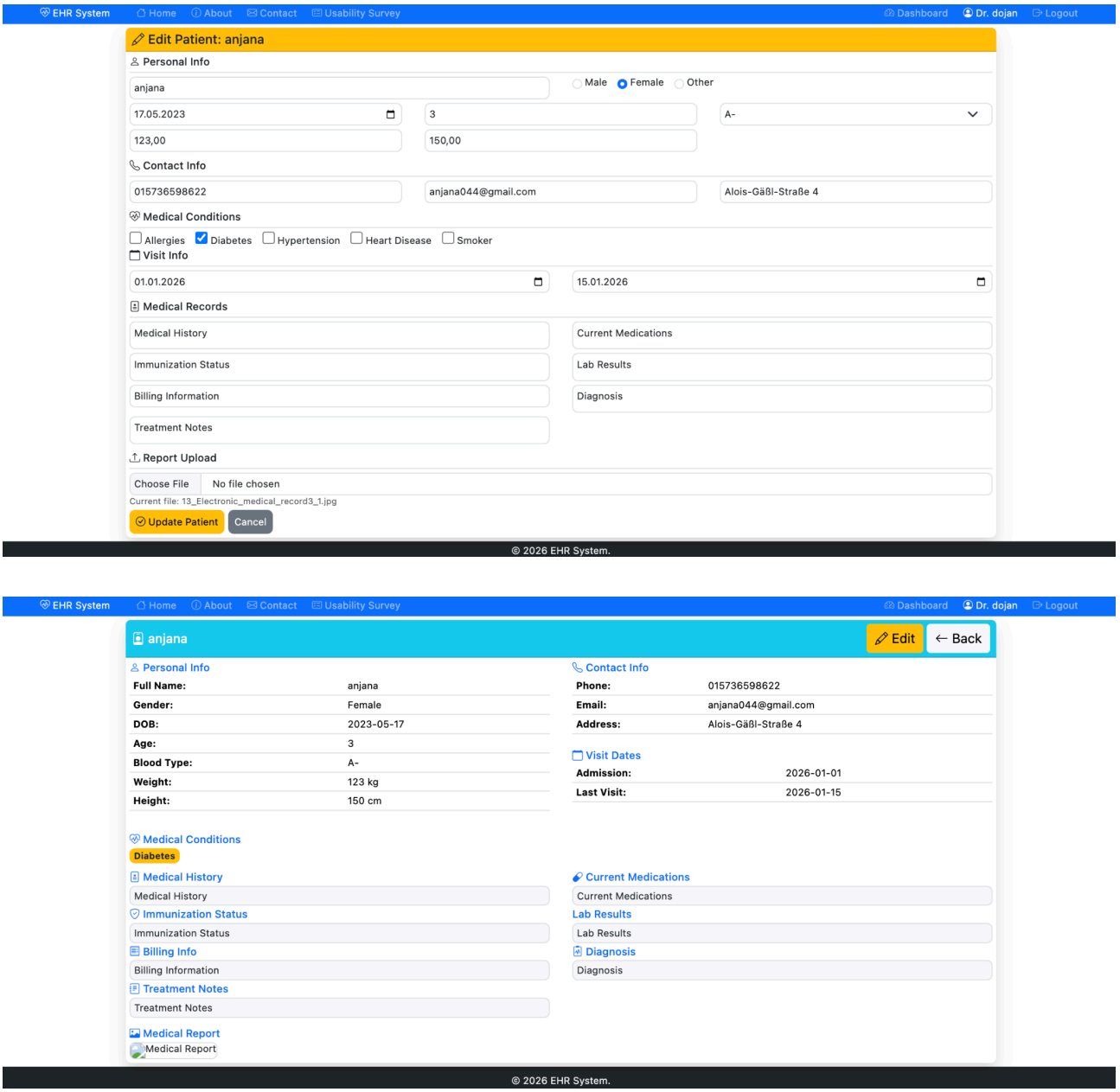


Figure 6: Screenshots of Add, Edit, View patient pages

3.4 Image Upload and Date Selection

The system supports medical report uploads in common formats such as PNG, JPEG, GIF, and PDF, stored in a secure folder with file paths saved in the database. Date pickers enable selection of dates for birth, admission, and last visit, improving usability and ensuring consistency in date formats.

---

4. Evaluation

The usability of the EHR system was assessed using a standard System Usability Scale questionnaire. Ten standard questions were included to capture feedback on system complexity,

ease of use, and user confidence. Responses are collected on a five-point scale and used to calculate an overall usability score. The evaluation confirmed that the system provides a clear, intuitive interface and supports efficient management of patient records. While more extensive testing with multiple users could provide additional insights, the current evaluation demonstrates that the system meets usability expectations for its intended use.

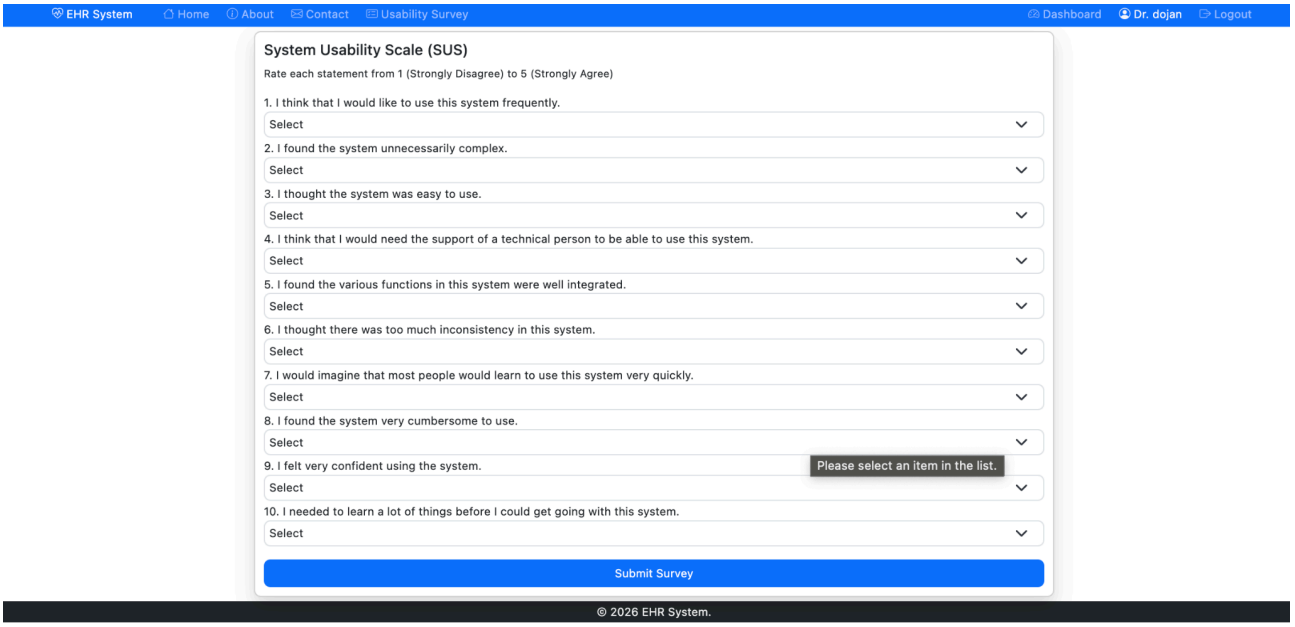


Figure 7: Screenshot of SUS questionnaire form

The usability of the EHR system was assessed using the System Usability Scale (SUS) questionnaire, completed by all registered doctors. The SUS scores were calculated for each participant, and the average score was found to be 42 out of 50. The scores ranged from 36 to 48, indicating that most users found the system intuitive, straightforward, and easy to use. This evaluation demonstrates that the EHR system provides a clear, well-structured interface, effectively supports patient record management, and meets usability expectations for its intended purpose. Overall, the high average SUS score reflects strong usability and positive user satisfaction, suggesting that the system can be confidently adopted by medical professionals for daily clinical operations.

5. Technical Implementation Details

The backend is implemented in Python Flask with the following key functionalities: routing, CRUD operations, form handling, image upload, session management, and email communication using Flask-Mail. SQL queries are used to interact with the MySQL database for inserting, updating, retrieving, and deleting patient and doctor records. Access control ensures that doctors can only manage their own patient data. The frontend uses HTML5, CSS3, and Bootstrap 5 to provide a responsive and structured interface.

The database schema includes the following primary tables: `doctors`, `patients`, `sus_responses`, and `contact_inquiries`. Each table is designed with appropriate data types and constraints, including primary and foreign keys to ensure data integrity. Indexes are created for frequently queried columns to optimize performance.

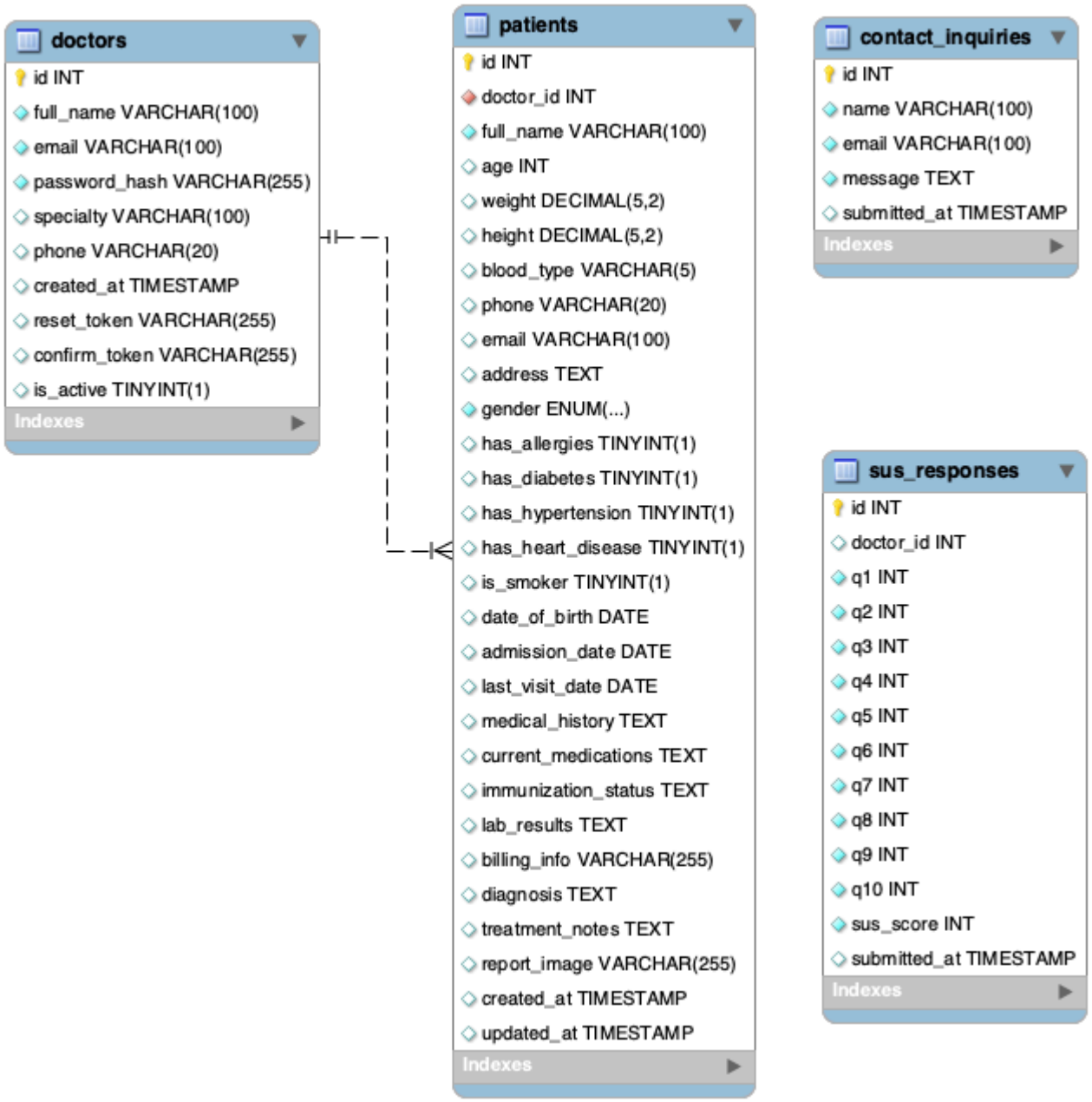


Figure 8 – Database schema diagram

6. Conclusion

The EHR system developed provides a secure, functional, and user-friendly platform for doctors to manage patient records. It incorporates all essential features for patient management, including demographics, medical history, laboratory results, medications, immunization status, diagnosis, treatment notes, and billing information. The system demonstrates ease of use, intuitive navigation, and robust functionality. Future improvements may include integration with external laboratory systems, advanced reporting, and multi-user collaboration.

As the sole developer, I designed the system architecture, implemented both frontend and backend functionalities, created the database schema, integrated email and image upload functionalities, conducted usability evaluation, and prepared this report and presentation.

---

7. References

1. Cahill, M., Cleary, B. J., & Cullinan, S. (2025). *The influence of electronic health record design on usability and medication safety: systematic review*. BMC Health Services Research. https://bmchealthservres.biomedcentral.com/articles/10.1186/s12913-024-12060-2
2. Zhang, J., & Walji, M. F. (2011). *Usability evaluation of electronic health record systems: analysis of vendor user-centered design processes*. Journal of the American Medical Informatics Association, 22(6), 1179–1186. https://academic.oup.com/jamia/article/22/6/1179/2357601
3. Holmgren, A. J., et al. (2024). *Electronic health record usability, satisfaction, and burnout for family physicians*. JAMA Network Open. https://jamanetwork.com/journals/jamanetworkopen/fullarticle/2822959