

DECLARATION

This is to declare that the project entitled “**Connecting Patients and Doctors For Flexible Healthcare Needs**” Submitted to Defense University College of Engineering Department of Computer Science and Engineering *in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology*. Computer Science and System Engineering is an original work carried out by ourselves.

Student Team Approval**Student Name****Signature**

1. Amen Befekadu

2. Elshaday Mekonnen

3. Hayder Aman

CERTIFICATE

This is certify that the project entitled **Connecting Patients and Doctors For Flexible Healthcare Needs** System is the work carried out by Computer Science and System Engineering students of Bachelor Technology, Defense Engineering College, Bishoftu, during the year 2024 *in partial fulfillment of the requirement for the award of the degree of Bachelor of Computer Science and System Engineering* is an original work carried out by Amen Befekadu, Elshaday Mekonnen, and Hayder Aman under the guidance. The work in this project is reliable and genuine work done by the students and has not been submitted to Defense University, College of Engineering.

Project Advisor Head of the Department

Maj. Getaneh Temesgen

Maj. Chernet Kejela (Ph.D)

Signature_____

Signature _____

Examiners Name

Signature

1. _____

2. _____

3. _____

Acknowledgments

First, we are very thankful to the almighty God for the successful completion of our project and our family for their support in helping us to reach us to learn. We would like to thank the Defence University Engineering College for establishing a wonderful college that made us achieve our dream. Next, we are also grateful for all the computer engineering department staff members who taught and helped us to reach our final goal. Specially, we would like to thank our advisor Maj. Getaneh Temesgen for his inspiration, kind guidance. He always shared his ideas to bring more completeness in our project and constantly reviewed our work. Lastly, we would like to thank all our friends and classmates for supporting us and helping us by providing moral support.

Abstract

The Connecting Patients and Doctors project is a groundbreaking initiative designed to address the pressing challenges faced by individuals seeking quality medical services. In the current healthcare landscape, traditional clinic hours and geographical limitations hinder access to timely care. Our mission is to create an innovative online platform that bridges these gaps, revolutionizing healthcare delivery.

The project's objectives include developing a user-friendly platform, verified medical professionals, creating a flexible scheduling system, and transcending geographical barriers. The significance lies in its transformative impact on healthcare 24/7 accessibility, saving time, providing convenience, accelerating information processing, and offering financial benefits to medical expertise. The project aims to build trust and bridge critical gaps in the healthcare delivery system.

Table of contents

Contents

DECLARATION.....	I
CERTIFICATE.....	II
Acknowledgments.....	III
Abstract.....	IV
Table of contents.....	V
List of Tables.....	IX
List of Figures.....	X
List of Abbreviations.....	XII
CHAPTER ONE.....	1
1.Introduction.....	1
1.1 Background of the project	2
1.2 Statement Of The Problem	3
1.3 The Proposed System.....	4
1.4 Objective of the project.....	6
1.4.1 General Objective	6
1.4.2 Specific Objectives.....	6
1.5 Beneficiaries of the new system.....	6
1.6 Significance of the project.....	7
1.7 Scope and Limitation of the project	7
1.7.1 Scope of the project	7
1.7.2 Limitation	8
1.8 Methodology of the Project.....	8
1.8.1 Data Collection Methodology	8
1.8.2 Fact Finding Techniques	9
1.8.3. System Requirement.....	9
1.8.4 System Testing Methodology.....	11

1.9 Requirement Analysis.....	11
1.9.1 Software Requirement	11
1.9.2 Hardware Requirement.....	12
1.10 Feasibility Study of the new System.....	12
1.10.1 Economic Feasibility	12
1.10.2 Tangible Benefits.....	12
1.10.3 Intangible Benefits.....	13
CHAPTER TWO.....	14
2. Literature Review.....	14
2.1 Introduction	14
2.2 Reviewed System	14
2.3 Summary of the Review.....	17
2.3.1 Research 1 Summary	17
2.3.2 Research 2 Summary	17
2.3.3 Research 3 Summary	18
CHAPTER THREE.....	19
3. Object-Oriented Analysis and Design.....	19
3.1 Introduction	19
3.1. 1 Object-Oriented analysis	19
3.1.2 Object-oriented Design.....	19
3.2 System Requirement Specification.....	20
3.2.1 Functional Requirement	20
3.2.2 Non Functional Requirement	20
3.3 System Use Case Model	21
3.3.1 What is the use case model	22
3.3.2 Use case description	22
3.3.3 What is the use case model	22
3.3.4 Analysis model.....	33

3.4 Sequence diagram and Activity Diagram	33
3.4.1 Sequence diagram.....	33
3.4.2 Activity Diagram	38
3.5 Deployment	42
3.5.1 Purpose of Deployment Diagram	42
CHAPTER FOUR.....	43
4. System Description.....	43
4.1 Introduction	43
4.1.1. Navigation	43
4.2 Register page	44
4.3 Login page.....	45
4.4. List of Users	46
4.5 List of Doctors	47
4.6 Reset Password	47
4.7 User Home Page	48
4.8 Booking Page.....	49
4.9 Apply Doctor Page	49
4.10 Doctor Notification	50
4.11 Admin Notification.....	51
4.12 Database Design	52
4.13 ER Diagram	53
4.13 Class Diagram	54
CHAPTER FIVE.....	55
5. Conclusion.....	55
5.1 Recommendation.....	55
References.....	56
Summary of Using the MERN Stack.....	57

List of Tables

Table 3.1 use case description for Login	26
Table 3.2 use case description for Registration.....	27
Table 3.3 use case description for Apply Doctor	28
Table 3.4 use case description for Book Appointment	29
Table 3.5 use case description for Doctor's Appointment	30
Table 3.6 use case description for Revoke Access For The Doctor.....	31
Table 3.7 use case description for Give Access To a Doctor.....	32

VIII

List of Figures

Figure 3.1 Admin over all control system use case diagram.....	23
Figure 3.2 Doctor over all control system use case diagram.....	24
Figure 3.3 User over all control system use case diagram.....	25
Figure 3.4 Sequence diagram for login activity	34
Figure 3.5 Sequence diagram for get all doctors activity	35
Figure 3.6. Sequence diagram for search activity.....	36
Figure 3.7 Sequence diagram for book appointment activity	37
Figure 3.8 Sequence diagram for apply doctor account activity.....	38
Figure 3.9 Activity diagram for the doctor.....	39
Figure 3.10 Activity diagram for the admin.....	40
Figure 3.11 Activity diagram for the user.....	41
Figure 3.12 Deployment diagram of the system.....	42
Figure 4.1 A screenshot of the register page section.....	44
Figure 4.2 A screenshot of the login page section.....	44
Figure 4.3 A screenshot of the list of user in Admin page Sections	46

IX

Figure 4.4 A screenshot of the list of doctor	47
Figure 4.5 A screenshot of the Forgot page sections.....	48
Figure 4.6 A screenshot of the user home page	48
Figure 4.7 A screenshot of the list user Booking page	49
Figure 4.8 A screenshot of the apply Doctor page	49
Figure 4.9 A screenshot of the Doctor notification panel.....	50
Figure 4.10 A screenshot of the Admin notification panel.....	51
Figure 4.11 Database Design	52
Figure 4.11 ER Diagram.....	53
Figure 4.3 Class Diagram	54

List of Abbreviations

DUCoE	Defense University College of Engineering.
VS Code	Visual Studio code.
OOAD	Object Oriented Analysis and Design.
OOA	Object Oriented Analysis.
OOD	Object Oriented Design.
UML	Unified Modeling Language.
EDraw	Engineering Draw
EHR	Electronic Health Record
IT	Information Technology
Telemedicine/Telehealth	Implementing Healthcare Systems Using Technology

CHAPTER ONE

1. Introduction

In today's dynamic world, the landscape of healthcare is evolving rapidly. With technological advancements and changing lifestyles, the traditional model of healthcare delivery is being challenged to adapt to the needs of modern society. One of the key aspects that demands attention is the flexibility in accessing healthcare services. Patients seek convenience, personalized care, and timely medical attention, while doctors aim for efficient practice management and meaningful patient engagement.

In an era defined by rapid technological advancement and shifting societal norms, the landscape of healthcare is undergoing a profound transformation. Gone are the days when patients were bound by the constraints of physical proximity and rigid appointment schedules. Today, individuals expect healthcare to be as flexible and accessible as the other facets of their lives.

Recognizing this paradigm shift, the concept of connecting patients and doctors through digital platforms has emerged as a beacon of innovation in the healthcare sector. This paradigm not only transcends geographical boundaries but also redefines the very essence of patient-doctor interactions. It is a fusion of technological prowess and human-centric care, aimed at fostering a healthcare ecosystem that is responsive to the diverse and evolving needs of both patients and healthcare providers.

The concept of connecting patients and doctors through digital platforms has emerged as a solution to address these evolving needs. By leveraging technology, we can bridge the gap between patients and healthcare providers, offering a flexible approach to healthcare delivery that transcends geographical boundaries and time constraints.

This project aims to explore and implement innovative strategies to connect patients with doctors for their flexible healthcare needs. Through the development of a user-friendly platform, we seek to revolutionize the way healthcare is accessed and delivered. By integrating features such as

telemedicine, appointment scheduling, remote communication we aspire to create a seamless and efficient healthcare ecosystem that prioritizes patient convenience and doctor-patient collaboration.

The need for flexible healthcare solutions is multifaceted and compelling. Modern lifestyles are characterized by hectic schedules, long commutes, and competing priorities, leaving little room for traditional healthcare appointments. Furthermore, the rise of chronic diseases and the aging population have underscored the importance of proactive and continuous healthcare management. In such a landscape, the rigid structures of conventional healthcare delivery often fall short, leaving patients feeling disconnected and underserved.

For doctors, digital healthcare platforms streamline practice management and enhance patient engagement. Through intuitive scheduling systems, electronic health records, and secure messaging functionalities, doctors can optimize their workflows, reduce administrative burdens, and focus more on delivering high-quality care to their patients. Additionally, remote monitoring technologies enable doctors to monitor patients' health metrics in real-time, enabling timely interventions and personalized treatment plans.

1.1 BACKGROUND INFORMATION

Our project aims to revolutionize healthcare delivery by addressing the challenges individuals face in accessing quality healthcare amidst their busy lives and traditional clinic constraints. We are developing a transformative system that connects patients with verified doctors, offering healthcare services beyond conventional clinic hours. Our platform facilitates easy connections between patients and qualified doctors, transcending traditional healthcare limitations. It prioritizes patient experience and accessibility, enabling healthcare services from home and minimizing wait times. By overcoming geographical barriers, our streamlined system promotes inclusivity in healthcare, making it more accessible, convenient, and patient-oriented.

The project on "Connecting Patients and Doctors for Flexible Healthcare Needs" aims to leverage technology to overcome traditional barriers in accessing healthcare services. It focuses

on establishing a digital platform that facilitates remote consultations and healthcare delivery, enabling patients to connect with doctors conveniently and efficiently.

The background of the project stems from the growing demand for flexible healthcare solutions, driven by factors such as increasing healthcare costs, demographic shifts, and advancements in telemedicine technology. Traditional healthcare delivery models often face challenges related to accessibility, affordability, and efficiency, particularly for individuals residing in remote areas or those with limited mobility.

By connecting patients and doctors through a digital platform, the project seeks to address these challenges and unlock various benefits. Patients gain greater flexibility in scheduling appointments, reduced travel time and expenses, and improved access to healthcare services, including specialist consultations. Healthcare providers benefit from enhanced efficiency, expanded reach, and opportunities for delivering more patient-centered care.

The project aligns with broader trends in healthcare innovation, emphasizing patient-centric approaches, telehealth adoption, and the integration of technology into care delivery processes. It underscores the importance of leveraging digital solutions to transform healthcare delivery, improve health outcomes, and enhance the overall patient experience.

1.2 STATEMENT OF THE PROBLEM

In the current healthcare landscape, traditional clinic hours and geographical constraints often limit individuals' access to quality medical services. This presents significant challenges for those seeking flexible healthcare options that align with their busy schedules and overcome location-related barriers. Furthermore, ensuring the credibility of healthcare providers remains a persistent issue. The absence of a centralized platform that prioritizes both convenience and

verified medical expertise contributes to these problems, highlighting the need for an innovative solution to bridge these gaps and revolutionize the healthcare delivery system.

Individuals with mobility issues or lack of reliable transportation struggle to attend in-person appointments, leading to delays in care and worsening health conditions. Traditional appointment scheduling systems are often rigid, requiring patients to adhere to predetermined clinic hours and availability slots, which can be particularly difficult for those with busy schedules or unpredictable work hours. Long wait times for appointments further contribute to patient frustration and delays in diagnosis and treatment initiation.

In addition to accessibility and scheduling issues, patients requiring specialized care encounter difficulties in accessing relevant healthcare providers, especially if they reside in remote areas. Complex referral processes and long wait times to see specialists result in delayed diagnosis, prolonged suffering, and decreased quality of life for patients with complex medical conditions. Limited communication channels and opportunities for patient engagement contribute to feelings of disconnection and dissatisfaction among patients. Additionally, socioeconomic factors such as income level, education level, and access to technology significantly impact an individual's ability to utilize digital healthcare services, exacerbating existing disparities in healthcare access and outcomes.

1.3 PROPOSED SYSTEM

The goals of the proposed system are to revolutionize healthcare delivery by creating a digital platform that connects patients with verified doctors for flexible healthcare services. This system aims to replace the traditional, location-bound, and time-restricted healthcare model with a more efficient and accessible web-based solution. By leveraging technology, the system will minimize the time and cost associated with accessing medical care, thus promoting operational efficiency and enhancing the overall patient experience.

The planned system's other goals include improving healthcare access control and data management. By digitalizing the current manual-based healthcare processes, the system will

upgrade the data management and access control system, ensuring that patient information is securely stored and easily accessible to authorized healthcare providers.

The following are the functions which can be delivered by the new system:

Improved Access to Healthcare Services: The system will provide telemedicine services and flexible appointment scheduling, ensuring patients can access healthcare services conveniently from their homes, offices, or anywhere, regardless of geographical location or schedule constraints.

Streamlined Care Coordination and Continuity: The system will facilitate seamless communication and collaboration between patients and healthcare providers, as well as among multidisciplinary care teams, by integrating with electronic health record systems and enabling secure data sharing.

Reduction of Healthcare Disparities: The system aims to address healthcare disparities by providing equitable access to healthcare services for all patients, regardless of their socioeconomic status, geographical location, or technological literacy.

Optimization of Healthcare Delivery: By implementing telemedicine capabilities, flexible appointment scheduling, and digital health tools, the system will optimize healthcare delivery by reducing wait times, improving efficiency, and maximizing the use of healthcare resources.

Promotion of Patient-Centered Care: The system will prioritize the needs, preferences, and values of patients in healthcare decision-making, promoting shared decision-making, patient engagement, and collaborative care through a user-centric digital platform

Overall, the proposed system aims to transform healthcare access, delivery, and experience by harnessing the power of digital technology and patient-centered approaches. By improving access to care, enhancing patient engagement, and advancing the quality and efficiency of

healthcare delivery, the system holds immense potential for revolutionizing the healthcare landscape.

1.4 Objective of the project

1.4.1 General objectives

To develop and implement a comprehensive digital healthcare platform that connects patients and doctors for flexible healthcare delivery, addressing the evolving needs of modern healthcare consumers and providers.

1.4.2 Specific objectives

- To Design and develop a user-friendly digital platform with an intuitive interface.
- Offer patients the ability to schedule appointments online .
- Enable secure messaging and communication channels between patients and healthcare providers .
- Implement care coordination workflows and alerts to notify healthcare providers appointments they have
- Implement data security measures, including encrypting passwords, access controls, to protect patient information and maintain confidentiality and privacy.

1.5 Beneficiaries of the new system

- 1) **Patients:**Patients benefit from increased access to healthcare services, including telemedicine consultations, flexible appointment scheduling, and on-demand access to health resources.
- 2) **Healthcare Providers:**Healthcare providers benefit from tools and technologies that streamline practice operations, optimize appointment scheduling, and reduce administrative burdens

- 3) **The General Healthcare Systems** :By focusing on patient-centered care, preventive services, and outcomes-based models, the project aligns with value-based care principles, promoting higher quality care and better health outcomes for patients.

1.6 Significance of the Project

The project aims to transform healthcare delivery by leveraging technology to connect patients with verified doctors for flexible healthcare services.

The significance of the system includes:

- Patients can access a variety of doctors within the same field or different specialties, ensuring they receive the most suitable care from verified professionals.
- Patients have the option to choose doctors based on their qualifications, allowing for more personalized and informed healthcare decisions.
- Doctors have an additional means of earning income through the platform, expanding their opportunities beyond traditional clinical settings.

1.7 Scope and limitation of the project

1.7.1 Scope of the project

The scope of a project aimed at connecting patients and doctors for flexible healthcare needs encompasses the development of a robust digital platform. This platform would serve as a seamless interface facilitating various aspects of healthcare delivery.

Core features would include:

- appointment scheduling
- Integration of messaging capabilities would enable remote consultations
- Doctor's can manage their

1.7.2 Limitations of the Project:

- **Technological Barriers:**
 - The project's effectiveness may be limited by technological barriers, such as limited internet access, low digital literacy, or outdated technology infrastructure, particularly in underserved or rural areas.

1.8 Methodology of the Project

1.8.1. Data Collection Methodology

To gather information and requirements that are used for our project we will use the following techniques.

- **Interview:** - To gather about the operations of the organization, and how they organize data which helps us to analyze whether we may include or not in our proposed project and also to gather other important information we will include in our system.
- **Observation:** -To gather information related to our project our group members will observe the old system which helps us to know what things we must include in the project.
- **Document analysis:** - We will also try to analyze the relevant document system of the land management systems, to have important information for our project work.

1.8.2 Fact-Finding Techniques

- **Literature Review:** To review existing academic literature on a particular topic. Analyzing and summarizing relevant academic articles, books, and research papers.
- **Surveys and Questionnaires:** To collect data and opinions from a sample of individuals. Designing and distributing surveys or questionnaires to gather quantitative or qualitative information.
- **Experiments:** Purpose: To test hypotheses and gather empirical data. Conducting controlled experiments in a laboratory or real-world setting to observe and measure outcomes.
- **Online Research:** To gather information from online sources. Utilizing online databases, academic journals, and reputable websites to collect relevant data.

1.8.3 System Requirements

1.8.3.1 Programming Tools

For the backend, we have used Node.js and Express.js. Node.js is a powerful, open-source, cross-platform JavaScript runtime environment, while Express.js is a flexible Node.js web application framework. Together, they provide a robust and scalable foundation for building web applications.

Some advantages of using Node.js and Express.js are:

- **Increased Efficiency & Usability:** Node.js offers high performance and speed, which is essential for handling concurrent connections efficiently.
- **Compatibility:** Node.js and Express.js are compatible across all major operating systems, including Windows, UNIX, and macOS.
- **Asynchronous Processing:** Node.js's non-blocking I/O operations make it ideal for handling multiple simultaneous requests and processing large volumes of data.

- **Easy to Understand:** Node.js and Express.js have a straightforward syntax, making them easy to learn and use for developers familiar with JavaScript.
- **Integration:** These tools offer seamless integration with various databases and other web technologies.
- **Cost Advantages:** Node.js and Express.js are open-source and widely supported, reducing development and deployment costs.

The frontend of the project is made with state-of-the-art tools to make it responsive and easy to use for non-technical users. Some of these tools are:

- HTML, CSS, JavaScript for implementing the UI.
- React and React-Router for handling routing in the frontend.
- Redux for state management.
- Firefox and Chrome for testing.

1.8.3.2 Database Tools

For the database, we have used MongoDB, a popular open-source NoSQL database. Unlike traditional relational databases, MongoDB is a document-oriented database that stores data in flexible, JSON-like documents. This allows for the efficient handling of large volumes of unstructured data and provides high performance, scalability, and ease of development. MongoDB is widely adopted by numerous large organizations for its ability to support modern, data-intensive applications. We have used MongoDB's latest stable version with its robust querying and indexing capabilities for optimal performance and reliability.

1.8.3.3 Documentation tools

For writing this documentation we have used Microsoft word 2019.

1.8.3.4 Presentation tools

For making and showcasing we have used Power point 2019.

1.8.3.5 Diagramming tools

For Diagramming we have used E-draw Max, an online tool for making diagrams, relational diagrams and flowcharts.

1.8.4 System Testing Methodology

Software testing is a method to check whether the actual software product matches expected requirements and to ensure that the software product is defect free. It involves execution of software/system components using manual or automated tools to evaluate one or more properties of interest. The purpose of software testing is to identify errors, gaps or missing requirements in contrast to actual requirements.

Those testing methods are:

- **Unit testing:** To test the independent module using this mechanism of testing.
- **Integration testing:** Using this type of testing method to test the modules which are independent and dependent on each other.
- **System Testing:** using these methods will test the functionality of all modules considered as a single system.

1.9 Requirement Analysis

In the progress to completion this project we are going to use the following software and hardware.

1.9.1 Software Requirement

To do our project we will plan to use software such as:

- **Powerpoint and MS-word:** for Documentation and presentation
- **E-draw Max:** for scheduling, use case, class diagram, sequence diagram, and any other designing related task.
- **Window -11 operating system version 2022:** has good Graphical User Interface and also, we are familiar with it. It also supports Google chrome to run on it.

1.9.2 Hardware Requirement

The following hardware tools are the tools that we are going to use for our project.

- **USB flash (16GB):** we use it to back up our project if unexpected.
- **Computer (Laptop):** for editing and writing the document.

1.10 Feasibility Study of the New System

Feasibility study is essential to evaluate the cost and benefits of the new system. Based on the feasibility study, a decision is made on whether to proceed or cancel the project.

Need of the Feasibility Study:

- It determines the potential of the existing system.
- It identifies the problems of the existing system.
- It defines all goals of the new system.
- It finds all possible solutions to the problems of the existing system.

1.10.1 Economic Feasibility

The system to be developed is economically feasible and the benefits outweigh the costs. By computerizing the healthcare delivery process, the project reduces costs associated with traditional healthcare operations. The new system brings a number of tangible and intangible benefits, making it a valuable investment.

1.10.2 Tangible Benefits

- **Cost Reduction:** Reduce costs by minimizing the need for physical infrastructure, transportation, and administrative tasks.
- **Error Reduction:** Reduce errors by automating appointment scheduling and patient records management.
- **Increase Speed of Activity:** Technology enables faster communication, appointment scheduling, and access to medical records.

1.10.3 Intangible Benefits

Using a digital healthcare delivery platform offers numerous intangible benefits. Some of them are:

- Reduce Resource Consumption
- Increase Security
- Increase Management Flexibility
- Faster Decision Making
- Boost Healthcare Provider Morale
- Increase Information Processing Efficiency

CHAPTER TWO

2. Literature Review

2.1 Introduction

The purpose of this literature review is to examine existing studies related to the development and implementation of digital platforms aimed at improving healthcare delivery by connecting patients with doctors in a flexible and efficient manner. By exploring relevant literature, this review will provide a foundation for the project titled "Connecting Patients and Doctors for Flexible Healthcare Needs," which aims to leverage technology to overcome traditional barriers in accessing healthcare services.

2.2 Reviewed Systems

Literature Review 1: Development of a Telemedicine Platform for Remote Consultations

Introduction The study "Development of a Telemedicine Platform for Remote Consultations" by Smith et al. (2021) explores the creation of a telemedicine system designed to provide remote healthcare services. The platform enables patients to schedule and conduct virtual consultations with healthcare providers, improving accessibility for individuals in remote or underserved areas.

Advantages

- Enhanced accessibility to healthcare services for patients in remote locations.
- Reduced need for physical visits, minimizing travel time and costs.
- Real-time consultations with healthcare providers, ensuring timely medical advice.
- Integration with electronic health records (EHR) for comprehensive patient care.

Tasks

1. Design and development of a secure telemedicine platform with video conferencing capabilities.
2. Implementation of scheduling and appointment management features.
3. Integration with EHR systems for seamless data sharing between patients and providers.
4. Development of mobile applications for patient and provider access.

Limitations

The study identifies challenges related to ensuring data security and privacy, as well as the need for reliable internet connectivity for optimal platform performance. Additionally, the system requires ongoing maintenance and updates to keep up with evolving telemedicine standards and regulations.

Citation ([Smith et al., 2021](#))

Literature Review 2: Doctor-patient sharing network medical service system:

This study proposes a network-based system that utilizes the Internet to enable seamless communication and sharing of medical information between different levels of hospitals and patients. The system integrates provincial/municipal hospital systems with community and central hospitals, as well as patient terminals, facilitating a comprehensive network for medical service delivery (Zhang, 2017).

Advantages:

- Enhances communication and medical information sharing among various hospital systems.
- Improves accessibility to medical services for patients across different regions.
- Reduces redundancy by integrating multiple levels of healthcare services.

Tasks:

1. Design and implement the network communication infrastructure to connect various hospital systems.
2. Develop a patient terminal interface to facilitate patient access to medical services.
3. Integrate community medical systems with central and tertiary hospital systems for comprehensive service delivery.

Limitations:

- The system relies heavily on the availability and reliability of internet connectivity.
- Potential privacy and security concerns regarding patient data sharing across multiple platforms.

- Requires significant initial investment and ongoing maintenance for network infrastructure.

Literature Review 3: Doctor-Patient Social Networking to Improve Specialist Care Coordination:

This study introduces a digital health technology paradigm that uses social networking to enhance communication and coordination among medical specialists. By integrating data across diagnostic modalities, the system facilitates remote consultations and improves care coordination, particularly in underserved areas. This model shows promise in reducing hospital admissions and improving patient outcomes ([Kagan, 2020](#)).

Advantages:

- Enhances communication and coordination among medical specialists through a social networking platform.
- Integrates data across diagnostic modalities, simplifying access to patient information.
- Facilitates remote consultations, improving care delivery in underserved areas.

Tasks:

1. Design and develop a digital health technology platform based on social networking principles.
2. Integrate various diagnostic modalities to streamline access to patient data.
3. Implement remote consultation features to connect specialists with patients in need.

Limitations:

- Relies on widespread adoption and engagement from medical professionals to be effective.
- Potential data privacy concerns due to the use of social networking technologies.
- Requires continuous updates and maintenance to ensure the platform remains relevant and secure.

2.3 Summary of the Review of All Three Papers

The review of the three papers highlights the significant advancements in leveraging technology to improve healthcare delivery through enhanced communication, real-time monitoring, and specialist coordination.

1, Development of a Telemedicine Platform for Remote Consultations (Smith et al., 2021)

The study by Smith et al. (2021) details the creation of a telemedicine platform designed to facilitate remote consultations between patients and healthcare providers. This innovative system aims to enhance healthcare accessibility for individuals in remote or underserved areas by allowing them to schedule and conduct virtual consultations. Key advantages of the platform include increased accessibility to healthcare services, reduced need for physical visits and associated travel costs, real-time consultations for timely medical advice, and integration with electronic health records (EHR) for comprehensive patient care.

The development tasks outlined in the study include designing a secure platform with video conferencing capabilities, implementing scheduling and appointment management features, integrating with EHR systems, and creating mobile applications for both patients and providers. Despite its potential, the study acknowledges challenges such as ensuring data security and privacy, the need for reliable internet connectivity, and the requirement for ongoing maintenance and updates to adhere to evolving telemedicine standards and regulations.

Citation: Smith et al., 2021

2, Doctor-Patient Sharing Network Medical Service System (Zhang, 2017):

This study introduces a comprehensive network-based system aimed at improving communication and medical information sharing across different hospital levels and between hospitals and patients. By using the Internet as the primary mode of communication, the system integrates various hospital systems, including community and central hospitals, to provide seamless and accessible medical services. Despite its potential to enhance healthcare delivery, the system faces challenges such as dependency on reliable internet connectivity and potential privacy concerns regarding patient data sharing.

3, Doctor-Patient Social Networking to Improve Specialist Care Coordination (Kagan, 2020): This paper presents a novel digital health technology that uses social networking principles to enhance communication and coordination among medical specialists. The system integrates diagnostic data to facilitate remote consultations and improve patient care, especially in underserved areas. By streamlining access to patient information and enabling efficient care coordination, the model shows promise in reducing hospital admissions and improving patient outcomes. Nevertheless, the system requires widespread adoption by medical professionals and faces data privacy challenges.

CHAPTER THREE

3. System Analysis

3.1 Introduction

This chapter deals with analyzing the proposed system by using different Unified Modeling Language (UML) analysis modeling techniques such as use case diagrams, the use case descriptions (scenarios), sequence diagrams, activity diagrams, analysis class diagrams, and user interface prototype. A UML diagram is a graphical representation of part of a model, typically showing a number of elements connected by relationships. Once the actors and use cases have been identified, the development of the use cases ensues, accompanied by the formulation of textual descriptions, also known as scenarios. The Sequence diagram is depicted based on the use cases which are developed for the proposed system. Activities will be represented by the activity diagrams. The object-oriented paradigm took its shape from the initial concept of a new programming approach, while the interest in design and analysis methods came much later.

3.1. 1 Object-Oriented analysis

Object Oriented Analysis (OOA) seeks to understand (analysis) a problem domain

(the challenge we are trying to address) and identifies all objects and their interaction.

3.1.2 Object-oriented Design

Object-oriented Design (OOD) developed (designs) the solution.

3.2 Requirement specification

3.2.1 Functional Requirement

The functional requirements for a system describe what the system should do. These requirements specify the inputs, processes, and outputs that the system needs to handle. In the context of the our project(Connecting Patients and Doctors For Flexible Healthcare Needs) the functional requirements include:

- The system should allow users (Admin, Doctor and patient) to register by providing necessary details
- The system should differentiate between user roles (Admin, Doctor and patient) and grant appropriate permissions based on their roles.
- Users should be able to create, update, and alter profiles, including personal details, contact information and qualifications.
- The system should enforce strict security measures to protect user data, including secure authentication, password encryption, and role-based access control.
- Users should be able to search Doctor based on different criteria, such as name, qualifications, or experience.
- The system should support creating and managing appointments and displaying appointments for the Doctors .

3.2.2 Non-functional requirements

Non-functional requirements, as the name suggests, are requirements that are not directly concerned with the specific services delivered by the system to its users[6]. Non-functional requirements are defined as the requirements that specify criteria that can be used to judge the operations of the system in certain conditions rather than specific behaviors [4]. These requirements provide criteria for evaluating the system's performance and behavior within its working environment. The functional requirements include:

- **Usability:** The system should have an intuitive and user-friendly interface, allowing users to easily navigate and perform tasks.
- **Performance:** The system should exhibit fast response times, ensuring quick execution of tasks and minimal delays.
- **Reliability:** The system should be highly reliable, with minimal downtime or disruptions in service.
- **Security:** The system should implement robust security measures to protect sensitive data and ensure the privacy of user information.
- **Scalability:** The system should be able to handle increasing amounts of data and user load without significant performance degradation.
- **Compatibility:** The system should be compatible with different browsers, operating systems, and devices commonly used by the users.
- **Maintainability:** The system should be modular and well-structured, allowing for easy maintenance, updates, and enhancements.
- **Availability:** The system should be available to users according to defined service level agreements, ensuring access during specified timeframes.

3.3 System Use case model

3.3.1 What is the Use case?

A Use Case is a UML modeling element that describes how a user of the proposed system interacts with the system to perform a discrete unit of work. Use case diagrams are usually referred to as behavior diagrams which are used to describe a set of actions (use cases) that some system or systems (subject) should or can perform in collaboration with one or more external users of the system (actors).

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well.

Use case diagrams have 4 major elements

- Actors
- System
- Use Cases
- Relationships between actors and use cases

After a deep study of the system we found different system users and categorized them in the following actor

- 1) Admin
- 2) Doctor
- 3) Patient

3.3.2 Use case description

The following are some of the use cases we identified in our system:

UC-01 Login

UC-02 Register

UC-03 Apply for Doctor

UC-04 Book Appointment

UC-05 Doctor's Appointments

UC-06 Revoke Access from a doctor

UC-07 Give Access to a doctor

Admin

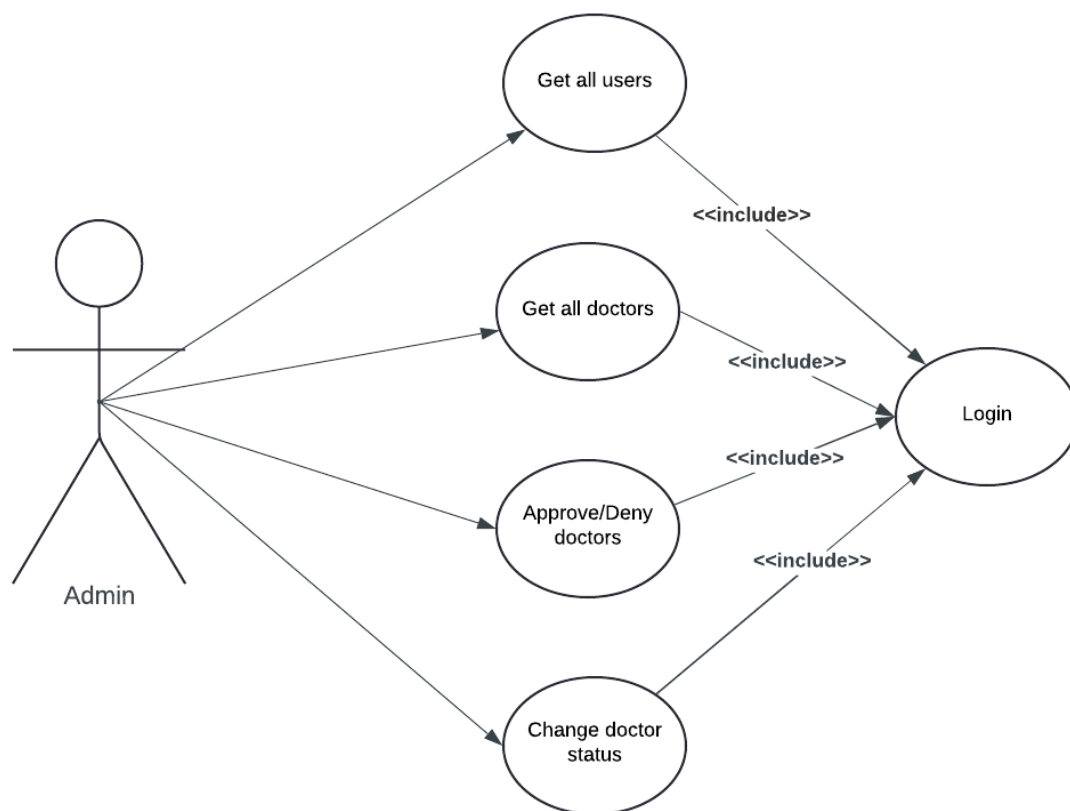


Figure 3.1 Admin overall control system use case diagram.

Doctor

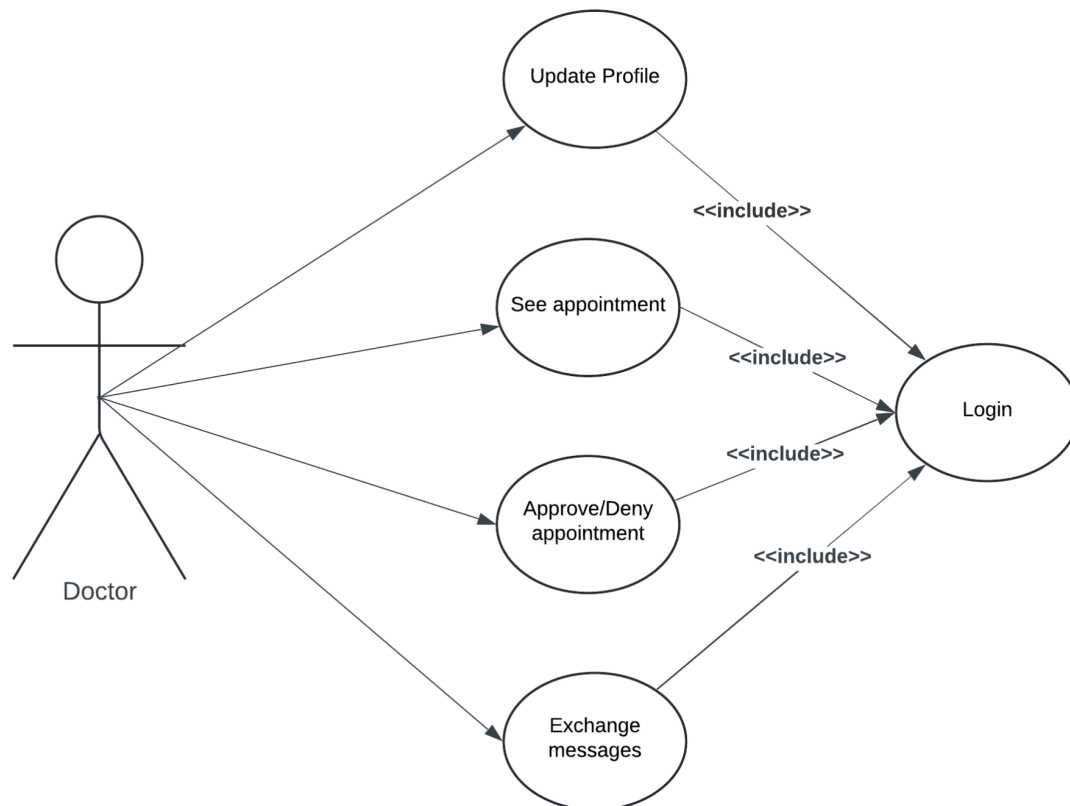


Figure 3.2 Doctor overall control system use case diagram.

User

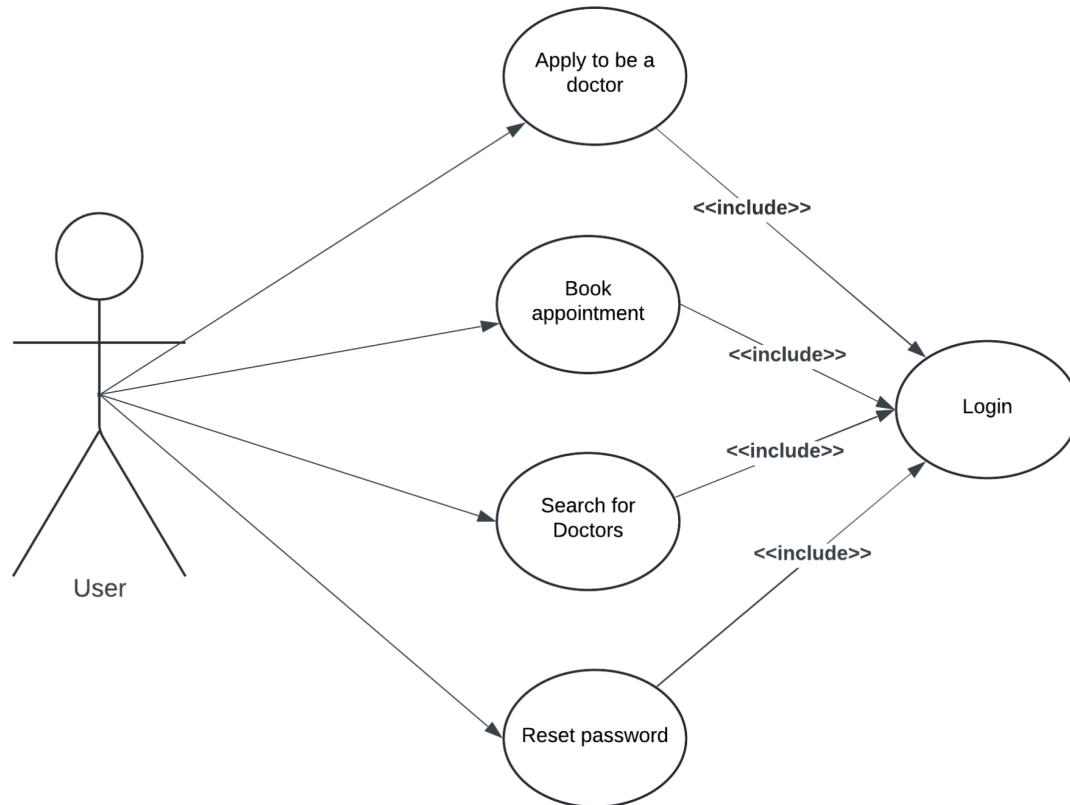


Figure 3.3 User overall control system use case diagram.

Use Case Name	Login
Use Case Number	UC_01
Description	This use case is used by the users to login to the system.
Actors	Admin, Doctor , User
Preconditions	The user must have an email and password.
Flow Event	<ol style="list-style-type: none">1. Users must access the system by a browser.2. The system displays a homepage.3. The homepage displays a login modal.4. The user enters their email and password.5. The user clicks the login button.6. The system authenticates the given information.7. The system redirects to the dashboard page.
Post conditions	The system displays the dashboard or the restricted page.
Alternate flow of events	The user enters the wrong set of email and password, and they get shown an incorrect email/password error message.

Table 3.1 Use case description for Log in

Use Case Name	Register
---------------	----------

Use Case Number	UC_02
Description	This use case is used by the users to register to the system.
Actors	Admin, Doctor , User
Preconditions	The user must have an email address that does not exist in the system.
Flow Event	<ol style="list-style-type: none">1. Users must access the system by a browser.2. The system displays a homepage.3. The User clicks on the register link.4. The user enters their First Name , Last Name, Email and Password , Confirm Password.5. The user clicks the register button.6. The system checks for errors in the way the data is given.7. The system checks for duplicate entries.8. The system redirects to the dashboard page.
Post conditions	The system displays the dashboard page and saves the given personal information.
Alternate flow of events	The user enters the wrong format of the required information and gets shown an error they made and what the correct format is.

Table 3.2 Use Case Description for Registration

Use Case Name	Apply for Doctor
Use Case Number	UC_03
Description	This use case is used by the users to apply to become a verified doctor to the system.
Actors	Admin, Doctor , User
Preconditions	The user must have already registered as a user in the system.
Flow Event	<ol style="list-style-type: none"> 1. Users must login to the system by a browser. 2. The system displays a homepage. 3. The User clicks on the Apply for doctor tab. 4. The user enters their First Name, Last Name ,Phone Number , Address , Specialization , Experience , and Fee Per Consultation and their respective convenient timings and other informations . 5. The user clicks the submit button. 6. The system checks if the doctor has already applied before . 7. The system sends the data verification 8.The system sends the given data to the db to store .
Post conditions	The system redirects the user to the user dashboard .

Table 3.3 use case description for Apply for Doctor

Use Case Name	Book Appointment
Use Case Number	UC_04
Description	This use case is used by the users to book appointments in the system.
Actors	Doctor , User
Preconditions	The user must have already registered and logged in as a user in the system.
Flow Event	<ol style="list-style-type: none"> 1. Users must login to the system by a browser. 2. The system displays lists of verified doctors 3. The User clicks one of the doctors. 4. The user enters date and time to check availability 5. The user clicks the submit button. 6. The system checks if the doctor has already an appointment on the same day and time . 7. If not the system sends the appointment data for approval to doctor . 8.The doctor approves or deny the appointment
Post conditions	The system redirects the user to the user dashboard .
Alternate flow of events	The user a date where the doctor is pre occupied and the system declines the request .

Table 3.4 use case description for Book Appointment

Use Case Name	Doctor's Appointments List
Use Case Number	UC_05
Description	This use case is used by the Doctor to see the list of appointments he has .
Actors	Doctor, User
Preconditions	The Doctor must have already registered and logged on as a user in the system.
Flow Event	<ol style="list-style-type: none">1. Doctors must login to the system by a browser.2. The system displays a dashboard .3. The Doctor clicks on the Appointments button.4. The Doctor checks the list of appointments it has .5. The Doctor clicks any of the appointments he has and can check details .6. The Doctor Accepts or Denys appointment request
Post conditions	The system redirects the user to the user dashboard .

Table 3.5 use case description for Doctor's Appointments

Use Case Name	Revoke Access from a doctor
Use Case Number	UC_06
Description	This use case is used by the Admin to Revoke Access from a doctor
Actors	Admin , Doctor
Preconditions	1: Admin must login to the system 2: The Doctor must have already registered and logged on as a Doctor in the system.
Flow Event	1. Admin must login to the system by a browser. 2. The system displays a dashboard . 3.The Admin checks the list of Doctors it has . 4. The Admin selects the desired Doctor to be blocked. 5. The system removes the privilege from the doctor.
Post conditions	The system returns a list of Doctors it has.

Table 3.6 use case description for Revoke Access from a Doctor

Use Case Name	Give Access to a doctor
Use Case Number	UC_07
Description	This use case is used by the Admin to give Access to a Doctor
Actors	Admin , Doctor
Preconditions	1: Admin must login to the system 2: The Doctor must have already registered, logged on as a user in the system and applied to a doctor.
Flow Event	1. Admin must login to the system by a browser. 2. The system displays a dashboard. 3. The Admin checks the notification list of users they applied for a doctor. 4. The Admin selects the desired User to be a doctor. 5. The system gives the privilege to a user.
Post conditions	The system returns a list of Doctors it has.

Table 3.7 use case description for Give Access to a doctor

3.4 Analysis Model

The requirements model is the first technical representation of a system. Modeling process uses a combination of text and diagrams to represent software requirements (data, function, and behavior) in an understandable way. Software engineers build requirements models using requirements elicited from customers.

Building analysis models helps to make it easier to uncover requirement inconsistencies and omissions. The System Analysis Model is made up of class diagrams, sequence or collaboration diagrams and Activity diagrams. Between them they constitute a logical, implementation-free view of the computer system that includes a detailed definition of every aspect of functionality. This model:

- Defines what the system does, not how it does it.
- Defines logical requirements in more detail than the use case model, rather than Physical solution to the requirements.
- Leaves out all technology details, including system topology.

3.4 Sequence diagrams and Activity Diagrams

3.4.1 Sequence diagram

A sequence diagram in a Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. A sequence diagram shows object interactions arranged in time sequence. It shows the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario

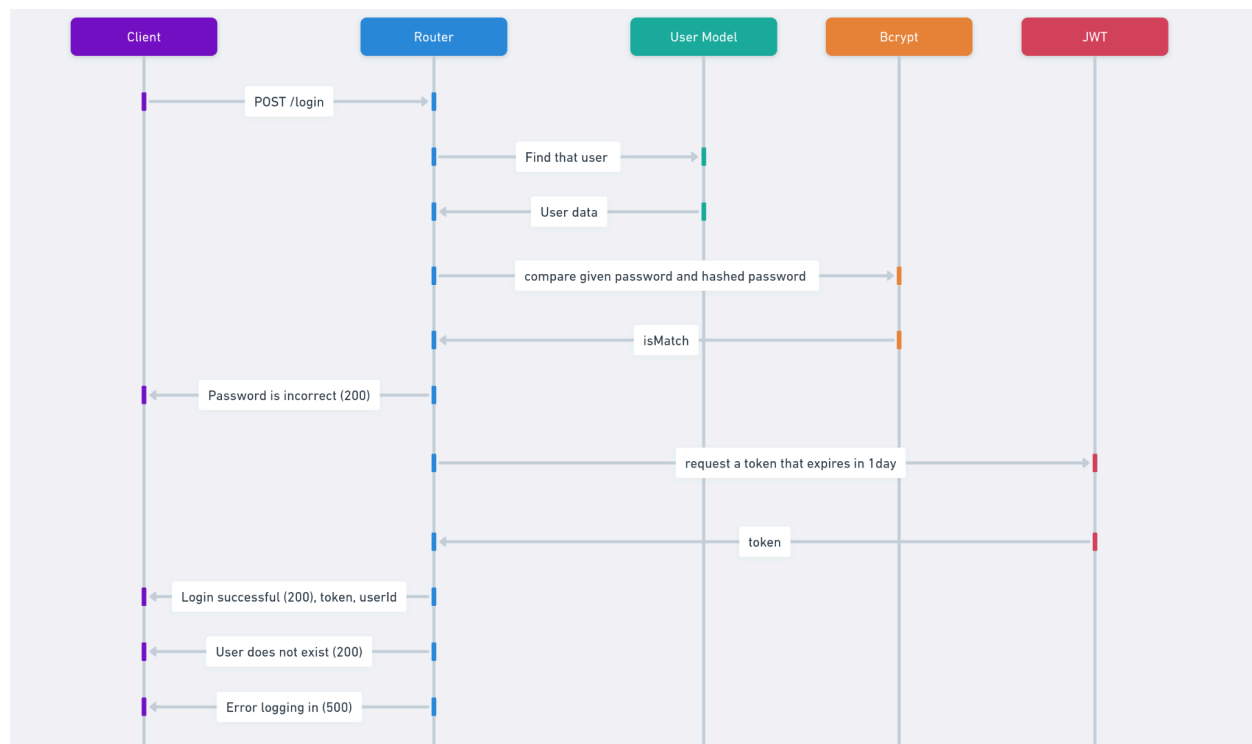


Figure 3. 4 Sequence diagram for Login activity system

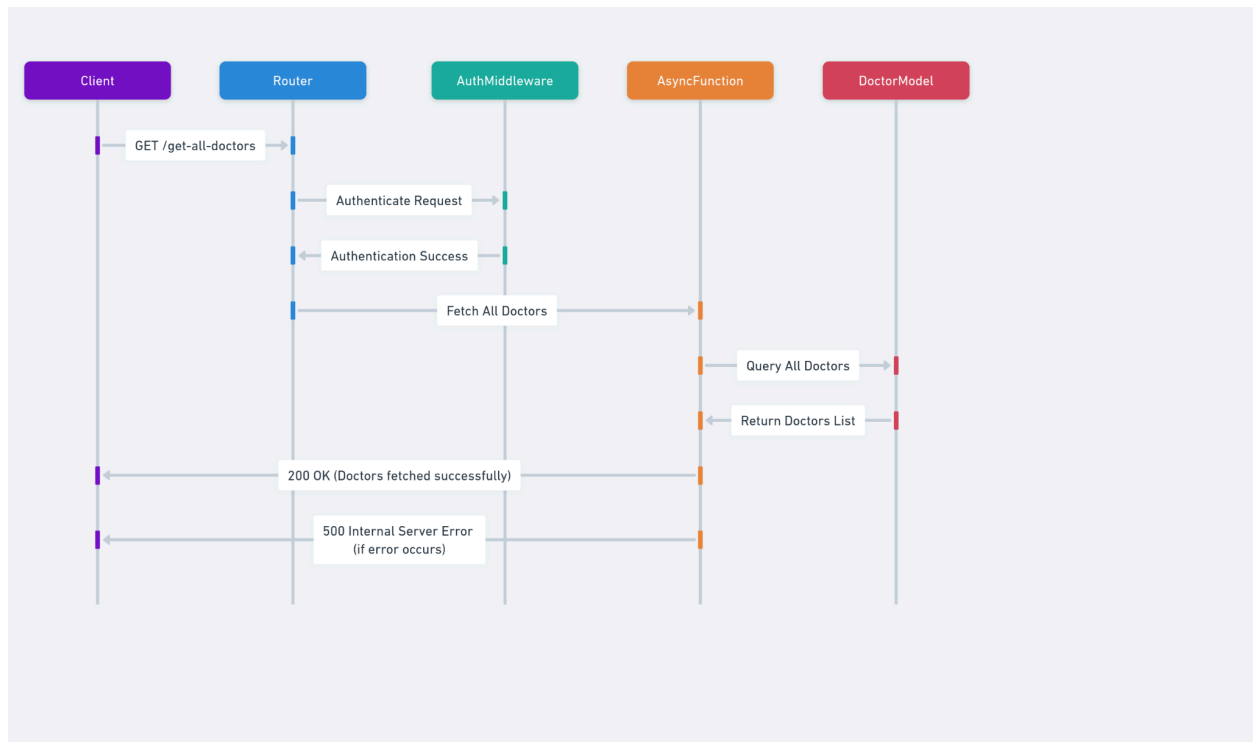


Figure 3. 5 Sequence diagram for get-all-doctors activity system



Figure 3. 6 Sequence diagram for Search activity

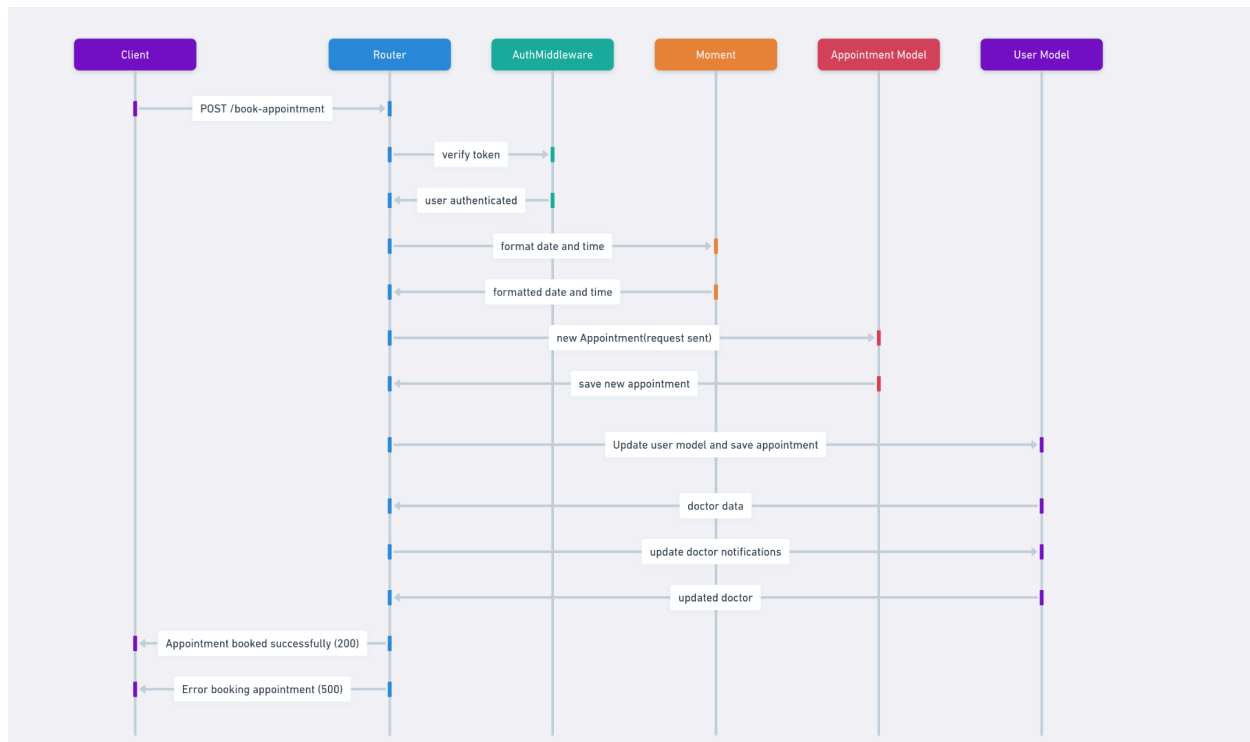


Figure 3. 7 Sequence diagram for Book-appointment activity system

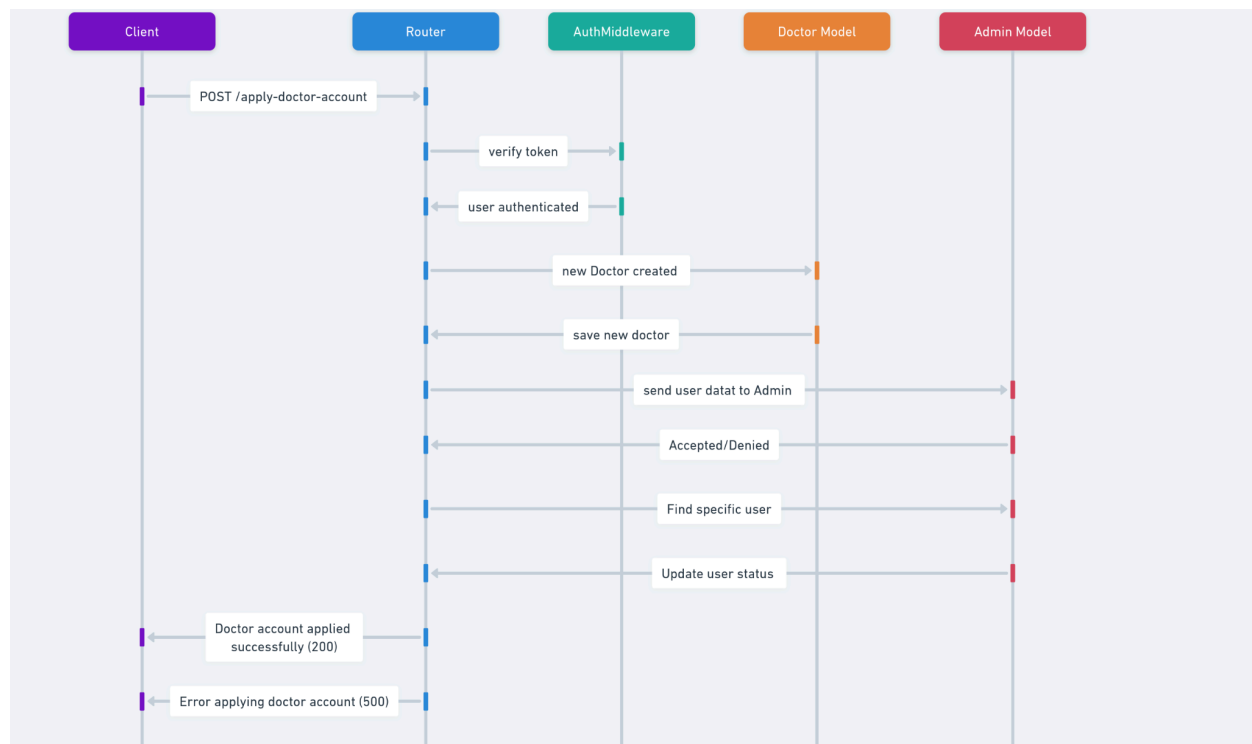


Figure 3. 8 Sequence diagram for apply-doctor-account activity system

3.4.2 Activity Diagram for the system

Activity diagram is another important diagram in UML to describe dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. So the control flow is drawn from one operation to another. This flow can be sequential, branched or concurrent. Activity diagrams deal with all types of flow control by using different elements like fork, join etc.

The purposes of activity diagram can be described as:

- Draw the activity flow of a system.
- Describe the sequence from one activity to another.
- Describe the parallel, branched and concurrent flow of the system

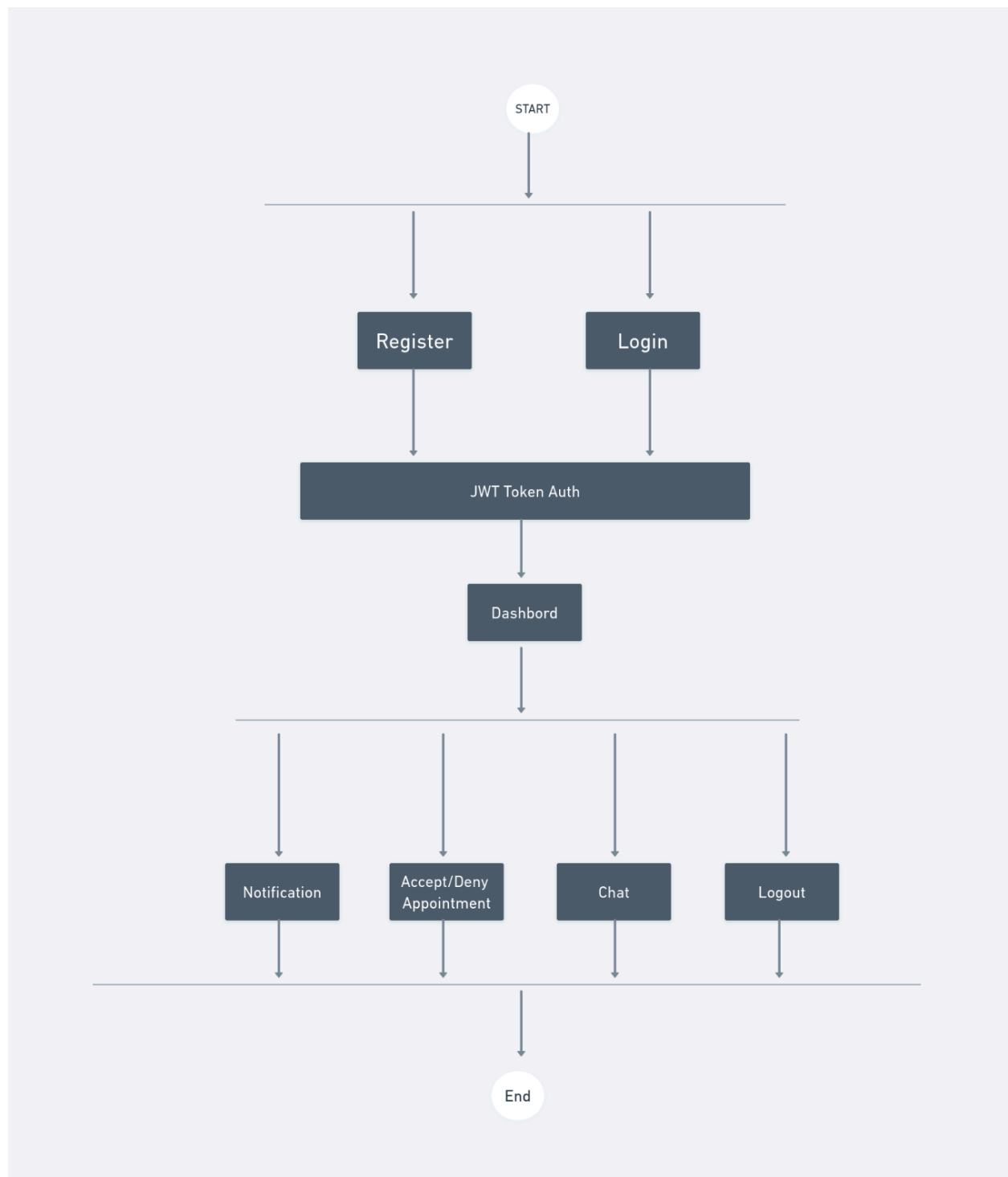


Figure 3.9 Activity diagram for the Doctor

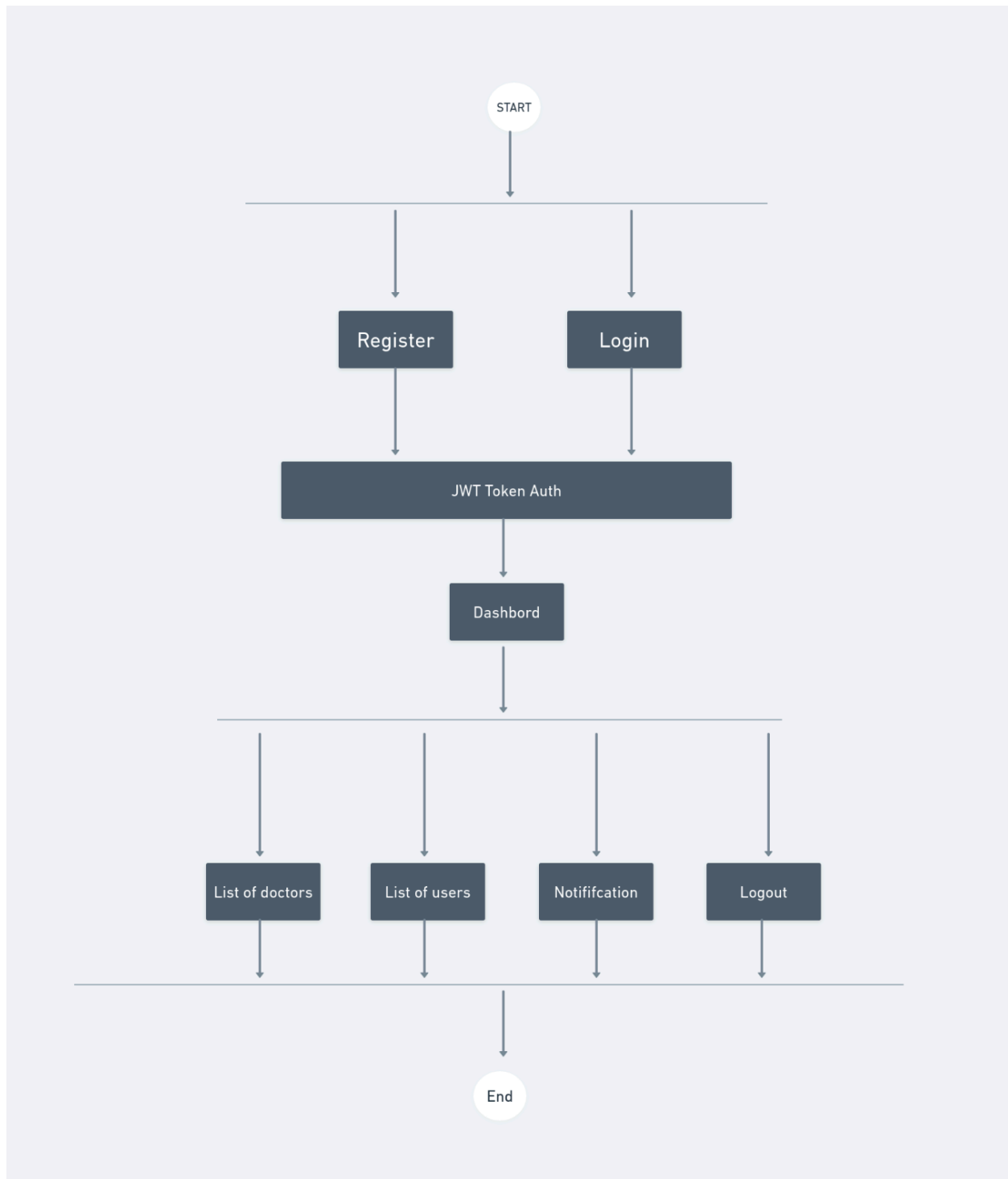


Figure 3.10 Activity diagram for the Admin

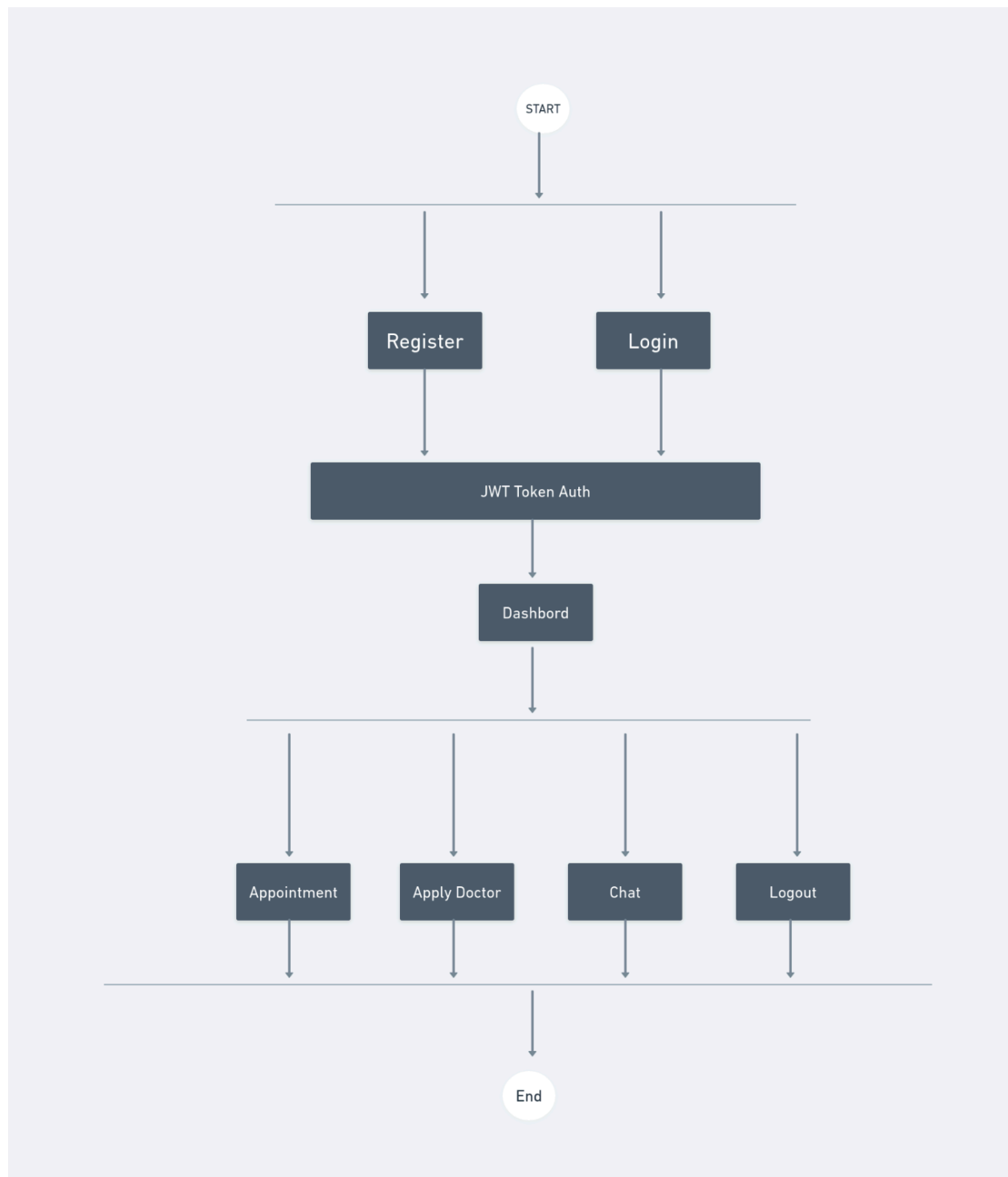


Figure 3.11 Activity diagram for the user

3.5 Deployment

Deployment Diagram is a type of diagram that specifies the physical hardware on which the software system will execute. It also determines how the software is deployed on the underlying hardware. It maps software pieces of a system to the device that are going to execute it.

3.5.1 Purpose of Deployment Diagrams

Deployment diagrams are used with the sole purpose of describing how software is deployed into the hardware system. It visualizes how software interacts with the hardware to execute the complete functionality. It is used to describe software to hardware interaction and vice versa.

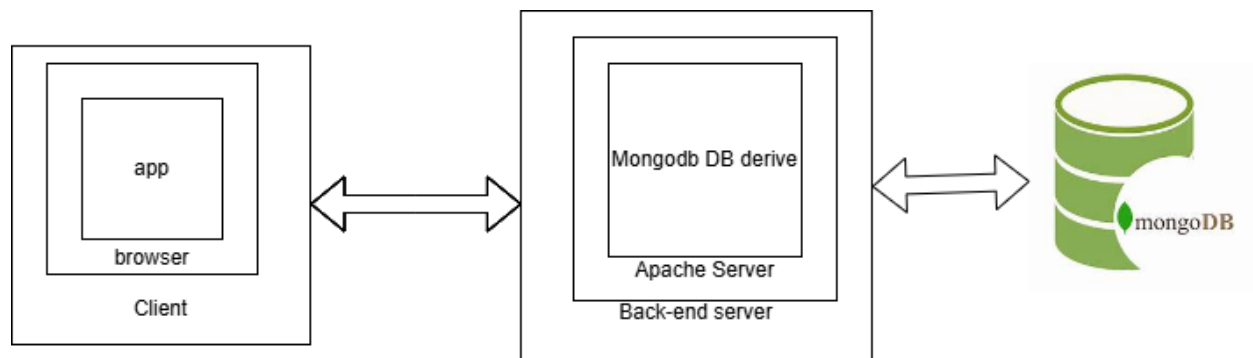


Figure 3.12 Deployment diagram of the system

CHAPTER FOUR

4. System Description

4.1 Introduction

The new proposed system named ‘Connecting Patients and Doctors For Flexible Healthcare Needs’ is a computer-based web application.

The system provides:

- A page for viewing the Lists of Doctors and their specialization .
- A page for viewing the Lists of Users.
- A page for controlling privileges .
- A page for Doctors to accept or deny appointment requests.
- A page for users to view the appointment status they requested.
- The Doctors will be able to update personal information.

This section of the document is aimed to describe the proposed system and its major function in detail. The section also provides major navigation pages with their corresponding outputs and layouts in preferred format and describes the system architecture including the database, the server and the front-end framework.

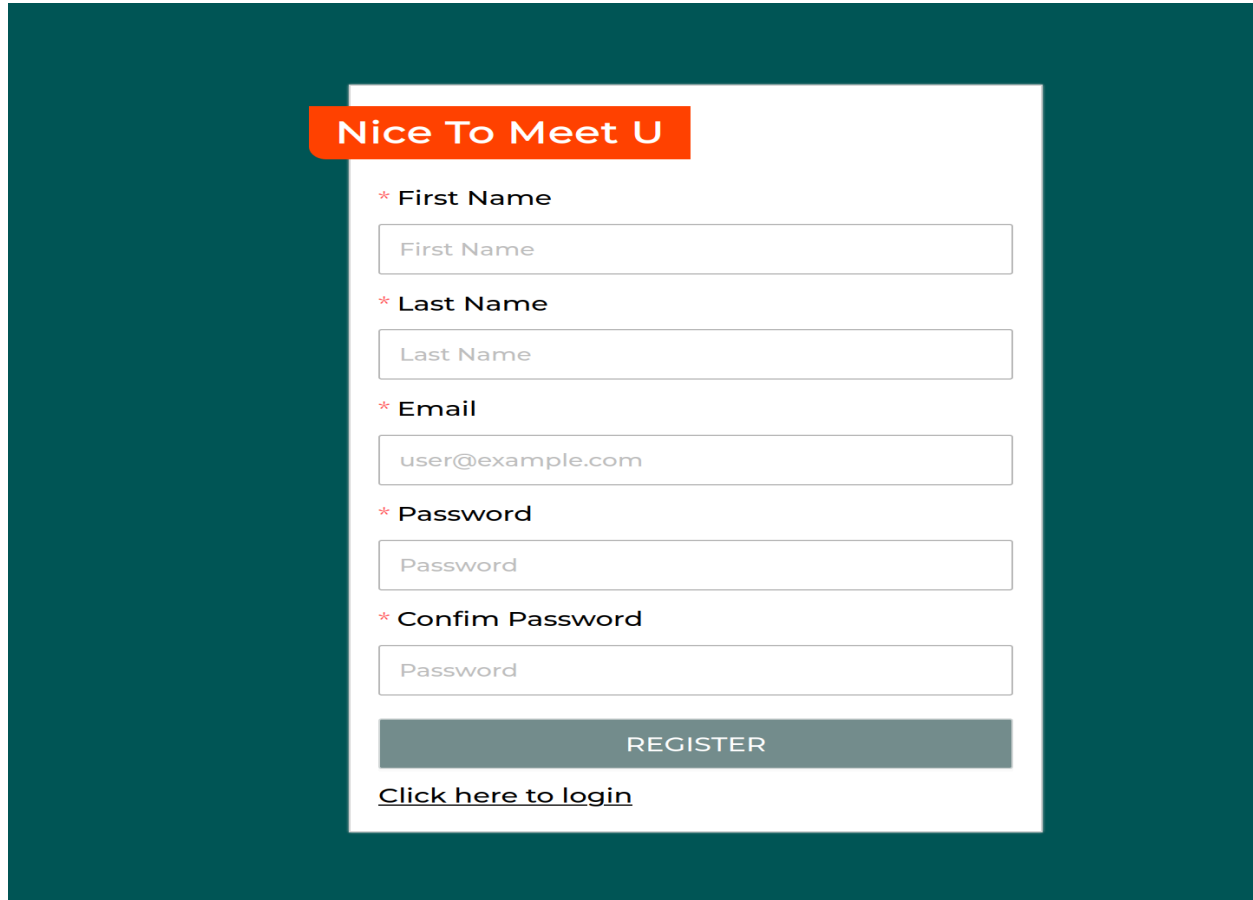
4.1.1 Navigation

Connecting Patients and Doctors For Flexible Healthcare Needs is a computer-based system and has multiple pages navigated by the user. In general, there are four main pages namely.

- **User Page** - a page where users can view all the available doctors, create appointment , can update their profile .
- **Admin Page** - a page where Admin can view all the available users ,can view all the available doctors , can change status of a doctor , approve or block doctors .
- **Doctor Page** - a page where doctros can view all the appointment they have, view all the notifications they have , accept or deny the appointment they have been given .

4.2 Register page

On this page we can create a user account by entering the required details. There are 3 parts to this page, and they are the Register Form, the Text Fields and the Register button.



The screenshot displays a registration form on a dark teal background. At the top, a red banner reads "Nice To Meet U". Below this, the form contains five required fields, each marked with a red asterisk (*):

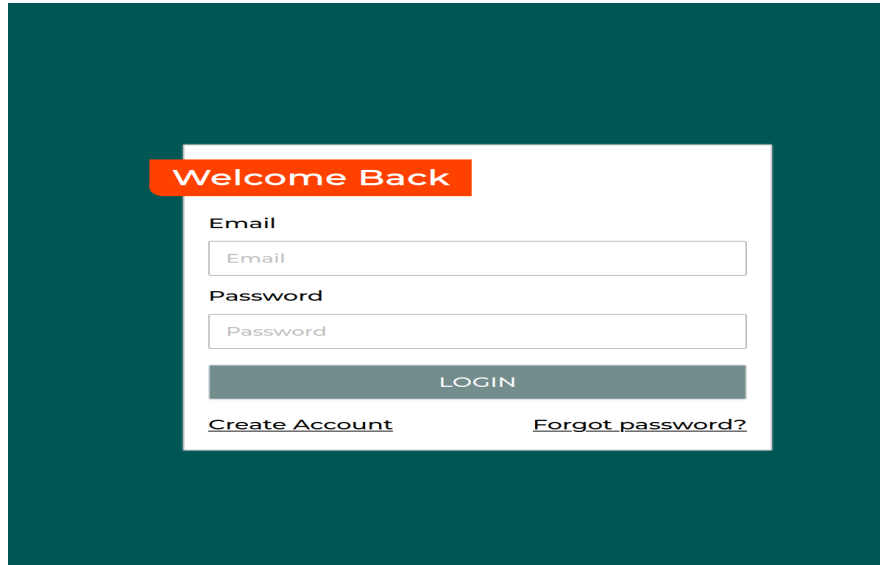
- * First Name**: A text input field with the placeholder "First Name".
- * Last Name**: A text input field with the placeholder "Last Name".
- * Email**: A text input field with the placeholder "user@example.com".
- * Password**: A text input field with the placeholder "Password".
- * Confirm Password**: A text input field with the placeholder "Password".

Below the fields is a grey button labeled "REGISTER". At the bottom of the form, there is a link that says "Click here to login".

Figure 4.1. A screenshot of the register page Section

- 1. Register Form** - This section contains everything we need to create a user to the system like the text fields for entering our information and the register button for the account-creating action.
- 2. Text Fields** - This section contains all the text fields needed to create a user account. Some of those fields are the First Name, Last Name, Email, Password and Confirm Password. The "*" symbol next to the names, email and password of text fields indicates that the text field is required.
- 3. Create Account** - This section contains the button for initiating the create account action. It first validates the inputs and starts the register action for the user.

4.3 Login Page

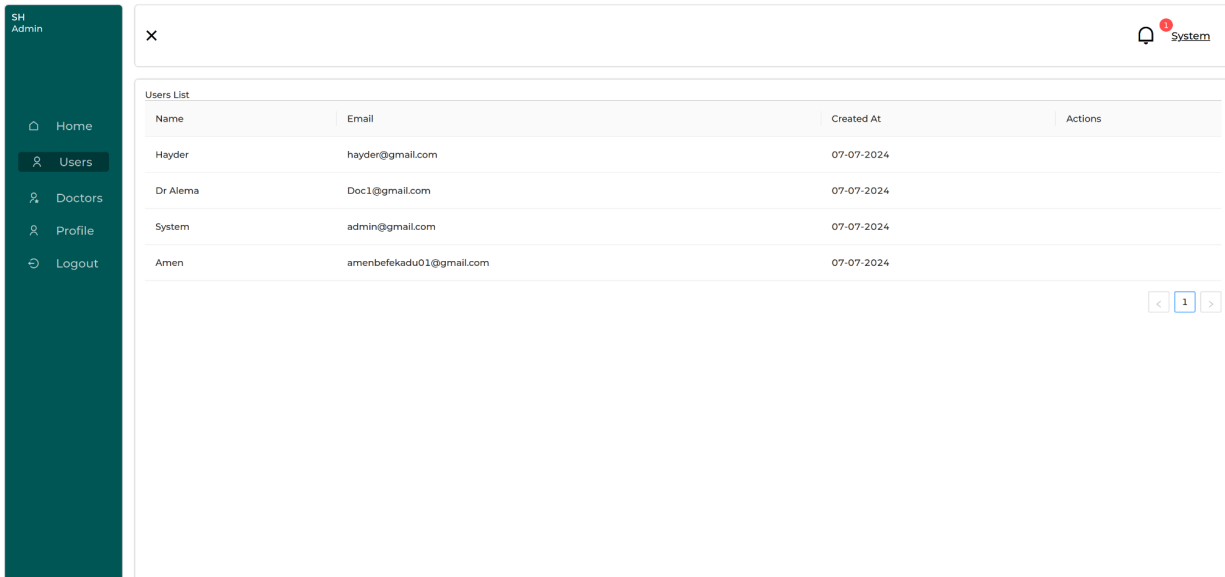


The screenshot displays a login interface on a dark teal background. At the top, a white box contains an orange banner with the text "Welcome Back". Below this, the form includes labels for "Email" and "Password", each followed by a corresponding text input field. A grey "LOGIN" button is positioned below the password field. At the bottom of the form, there are two links: "Create Account" on the left and "Forgot password?" on the right.

Figure 4.2. A screenshots of Login page Sections

- 1. Login Form** - This section contains everything we need to login to the system like the text fields for entering our information and the login button for the action.
- 2. Text fields** - This section contains the text fields for entering out email and password. The “*” symbol next to the names of text fields indicates that the text field is required.
- 3. Login Button** - This section contains the button for initiating the logging in action. It first validates the inputs and starts the logging in action for the user.

4.4 List of Users



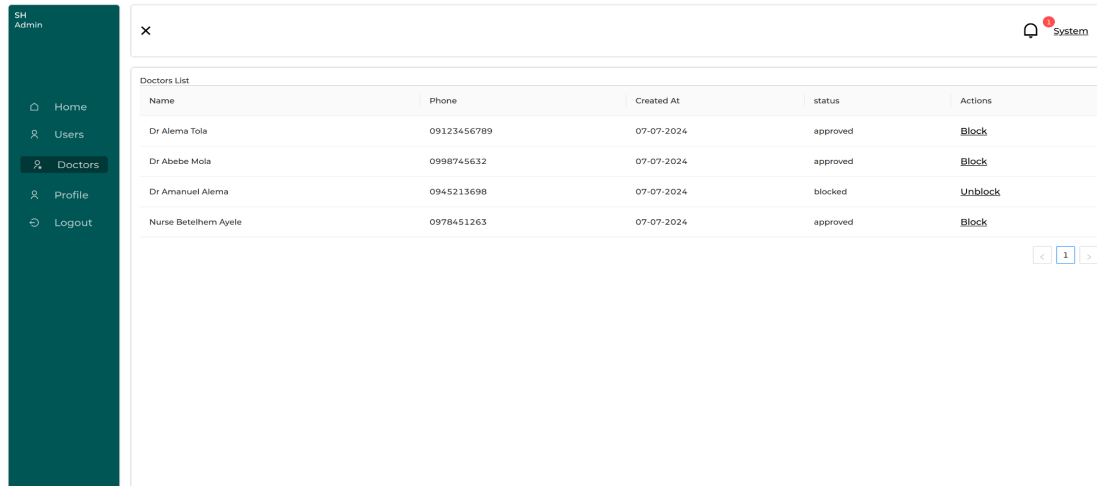
The screenshot displays the Admin interface. On the left is a dark teal sidebar with the text 'SH Admin' at the top and navigation links: 'Home', 'Users' (highlighted), 'Doctors', 'Profile', and 'Logout'. The main content area has a top header with a close button 'X' and a notification bell icon labeled 'System' with a red '1' badge. Below this is a table titled 'Users List' with columns: 'Name', 'Email', 'Created At', and 'Actions'. The table contains four rows of user data. At the bottom right of the table area is a pagination control showing '< 1 >'. The 'Actions' column is present but empty in the visible rows.

Name	Email	Created At	Actions
Hayder	hayder@gmail.com	07-07-2024	
Dr Alema	Doc1@gmail.com	07-07-2024	
System	admin@gmail.com	07-07-2024	
Amen	amenbefekadu01@gmail.com	07-07-2024	

Figure 4.3 A screenshot of the List of users in Admin page Sections

- 1. Admin Sidebar** - This section contains all the functionality the admin can operate .
- 2. Notification** - This section contains notifications that are requested to the admin.
- 3. Users List Interface** - This section contains all the list of users in the system .

4.5 List of Doctors

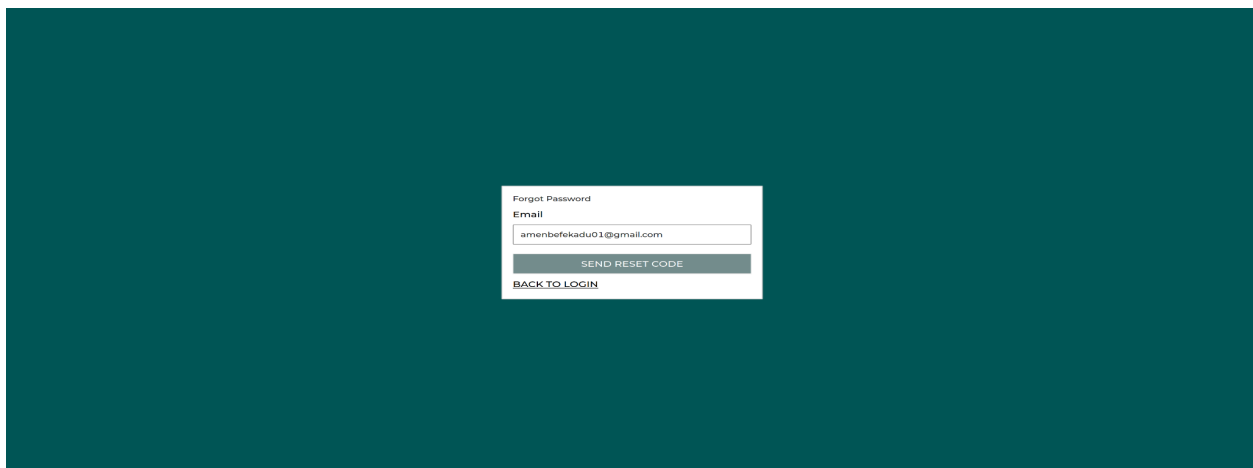


Name	Phone	Created At	status	Actions
Dr Alema Tola	09123456789	07-07-2024	approved	Block
Dr Abebe Mola	0998745632	07-07-2024	approved	Block
Dr Amanuel Alema	0945213698	07-07-2024	blocked	Unblock
Nurse Betelhem Ayale	0978451263	07-07-2024	approved	Block

Figure 4.4 A screenshot of the List of doctors in Admin page Sections

- 1. Admin Sidebar** - This section contains all the functionality the admin can operate .
- 2. Notification** - This section contains notifications that are requested to the admin.
- 3. Doctor Icon** - This section contains all the list of doctors in the system .

4.6 Reset Password



Forgot Password

Email

amenbefeakadu01@gmail.com

[SEND RESET CODE](#)

[BACK TO LOGIN](#)

Figure 4.5 A screenshot of the Forgot page Sections

- 1. Reset Form** - This section contains everything we need to reset to the system like the text fields for entering our email and the Send Reset Code button for the action.
- 2. Text fields** - This section contains the text fields for entering our email .
- 3. Send Reset Code Button** - This section contains the button for initiating the logging in action. It first validates the inputs and starts the send reset code action for the user.

4.7 User Home Page

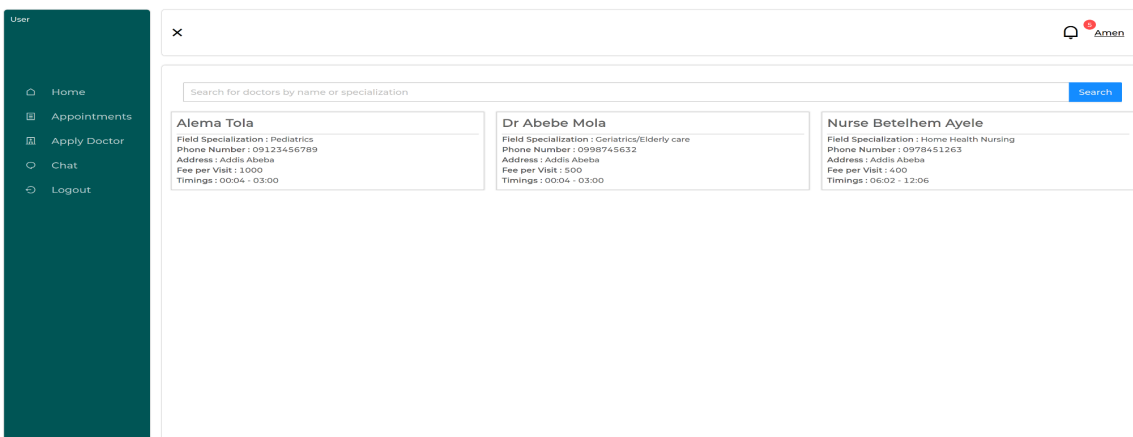


Figure 4.6 A screenshot of the user homepage

The search parts

- 1. Search Form** - This section contains a form functionality to search any verified doctor in the system .
- 2. Text Fields** - This section contains the text fields for entering Doctor first name or specialization to be searched.
- 3. Search Button** - This section contains the button for initiating the search in action. It first validates the inputs and starts the search action for the user.

The list of doctor parts

- 1. Available Doctors** - This section shows all the Doctors that are approved by the system. It contains the Name, Specialization, Phone number, Address and Fee per Visit and Timing

4.8 Booking Page

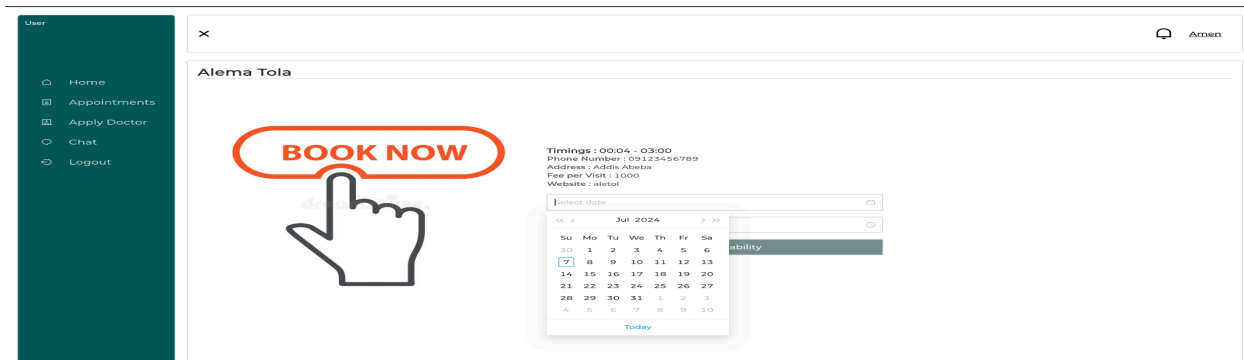


Figure 4.7 A screenshot of the user Booking page

- 1. User Sidebar** - This section contains all the functionality the user can operate .
- 2, Notification** - This section contains notifications that are requested to the admin.
- 3, Book Interface** - This section gives the user power to book an appointment where first he has to check for the availability of the doctor then the date and time are specified by the user itself.

4.9 Apply Doctor Page

Figure 4.8 A screenshot of the user Apply Doctor Page

- 1. User Sidebar** - This section contains all the functionality the user can operate .
- 2. Notification Icon** - This section contains notifications that are requested to the admin.
- 3. Apply Doctor Interface** - This section gives the use a request portal to become a verified doctor if he gives an appropriate information about his work life and all the information is sent to the admin for approval .

4.10 Doctor notification panel

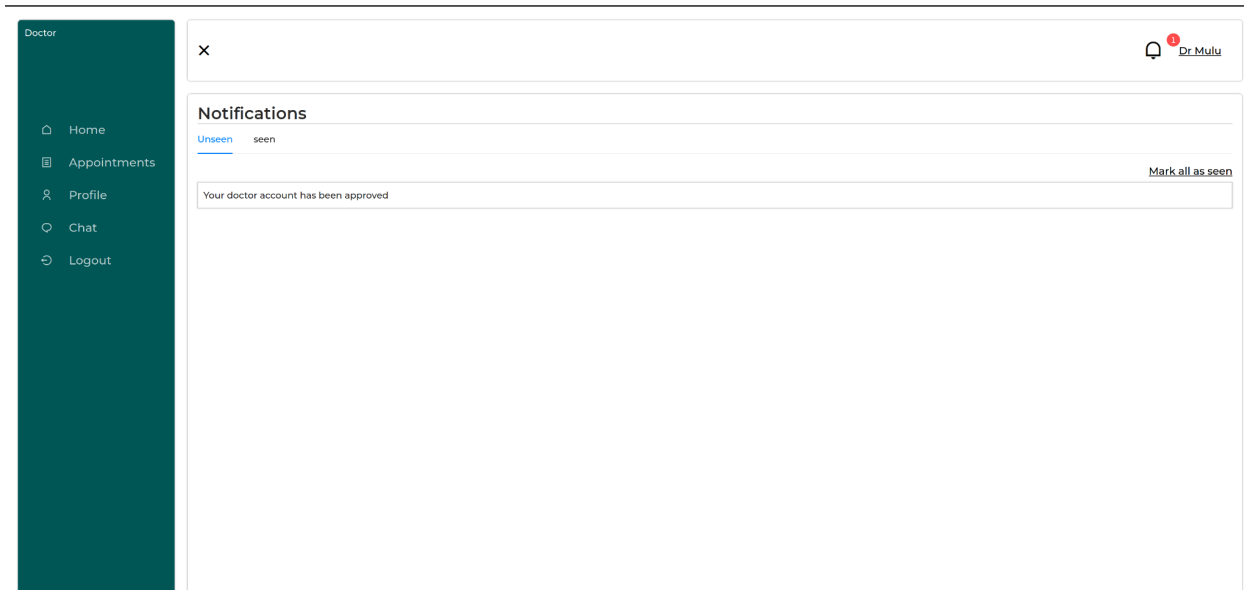
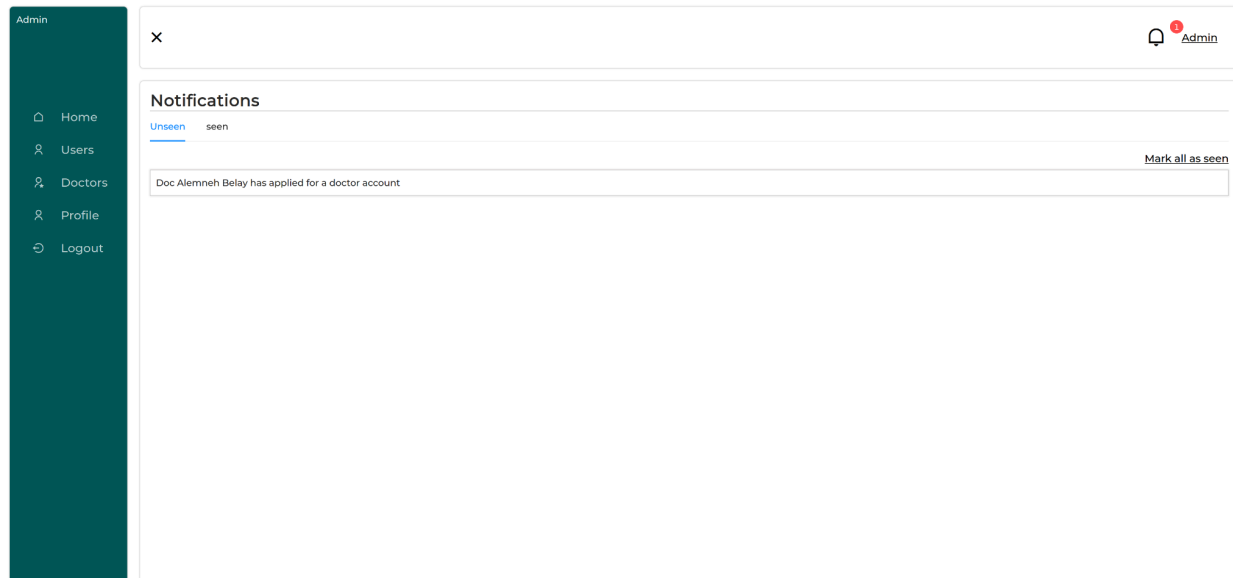


Figure 4.9 A screenshot of the Doctor notification panel

- 1. Doctor Sidebar** - This section contains all the functionality the doctor can operate .
- 2. Notification Icon**- This section contains notifications that are requested to the admin.
- 3. Notification Interface** - This section gives the Doctor all the notifications he has including if his request to become a doctor is approved or not .

4.11 Admin notification panel



- 1. Admin Sidebar** - This section contains all the Admins that have been registered .
- 2. Notification Icon**- This section contains notifications that are requested to the admin.
- 3. Notification Interface** - This section gives the Doctor all the notifications he has including if his request to become a doctor is approved or not .

4.12 Database Design

Database Design is the organization of data according to a database model. The designer determines what data must be stored and how the data elements interrelate.

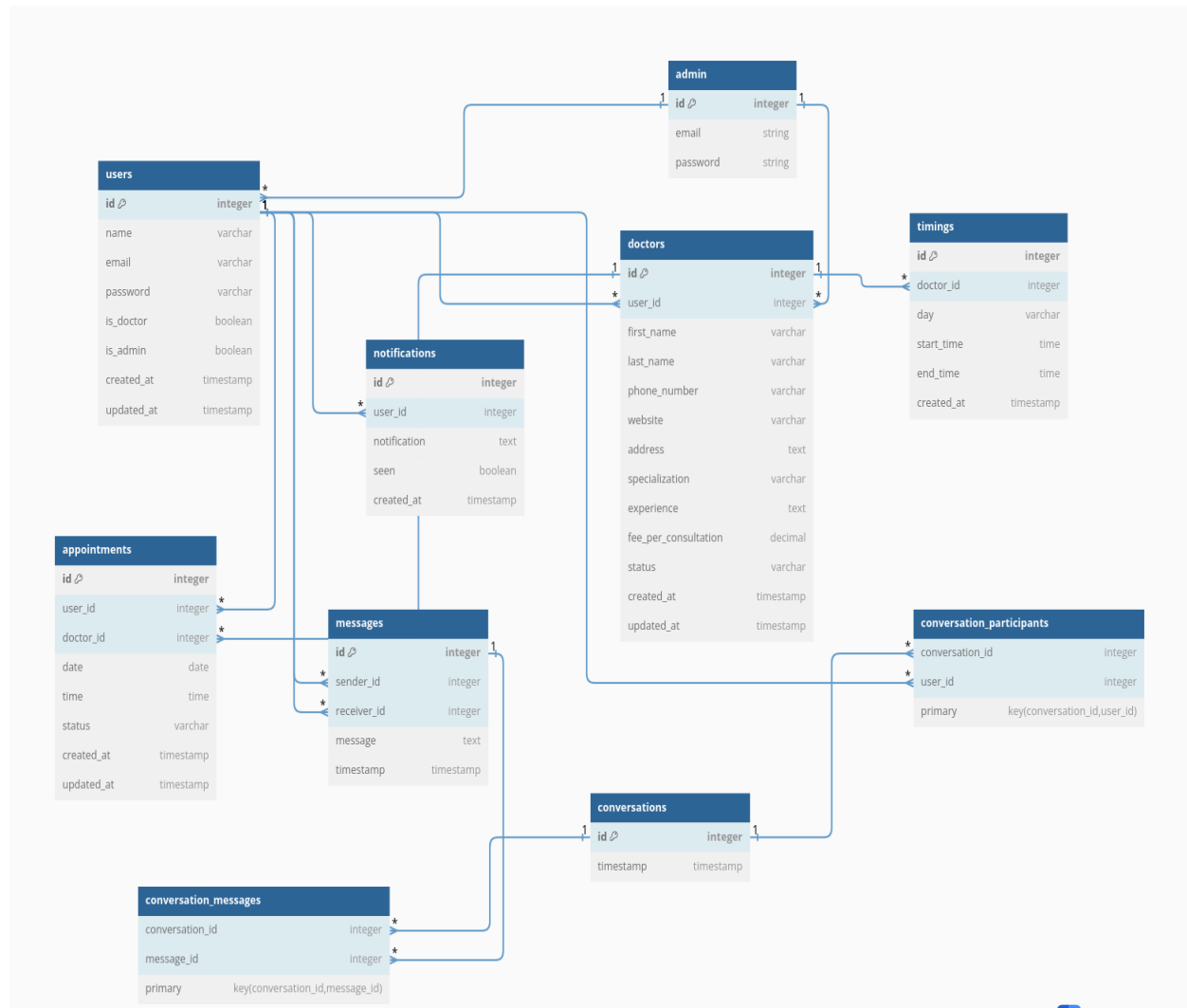


Figure 4.11 Database Design

4.13 ER- diagram.

An Entity relationship Diagram is a diagram that represents relationships among entities in a database. A graphical representation that depicts relationships among people, objects, places, concepts or events within an information technology (IT) system.

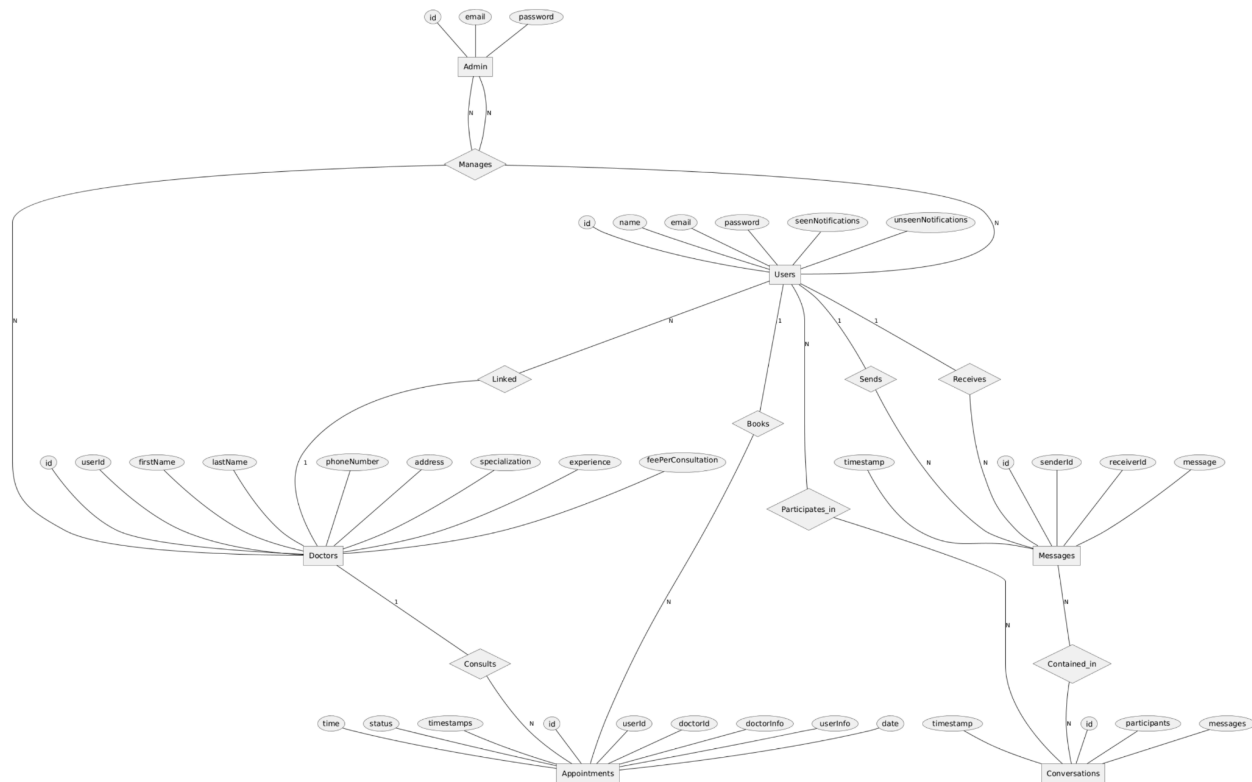


Figure 4.12 ER- Diagram

4.11. Class Diagram

The UML Class diagram is a graphical notation used to construct and visualize object oriented systems. A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system.

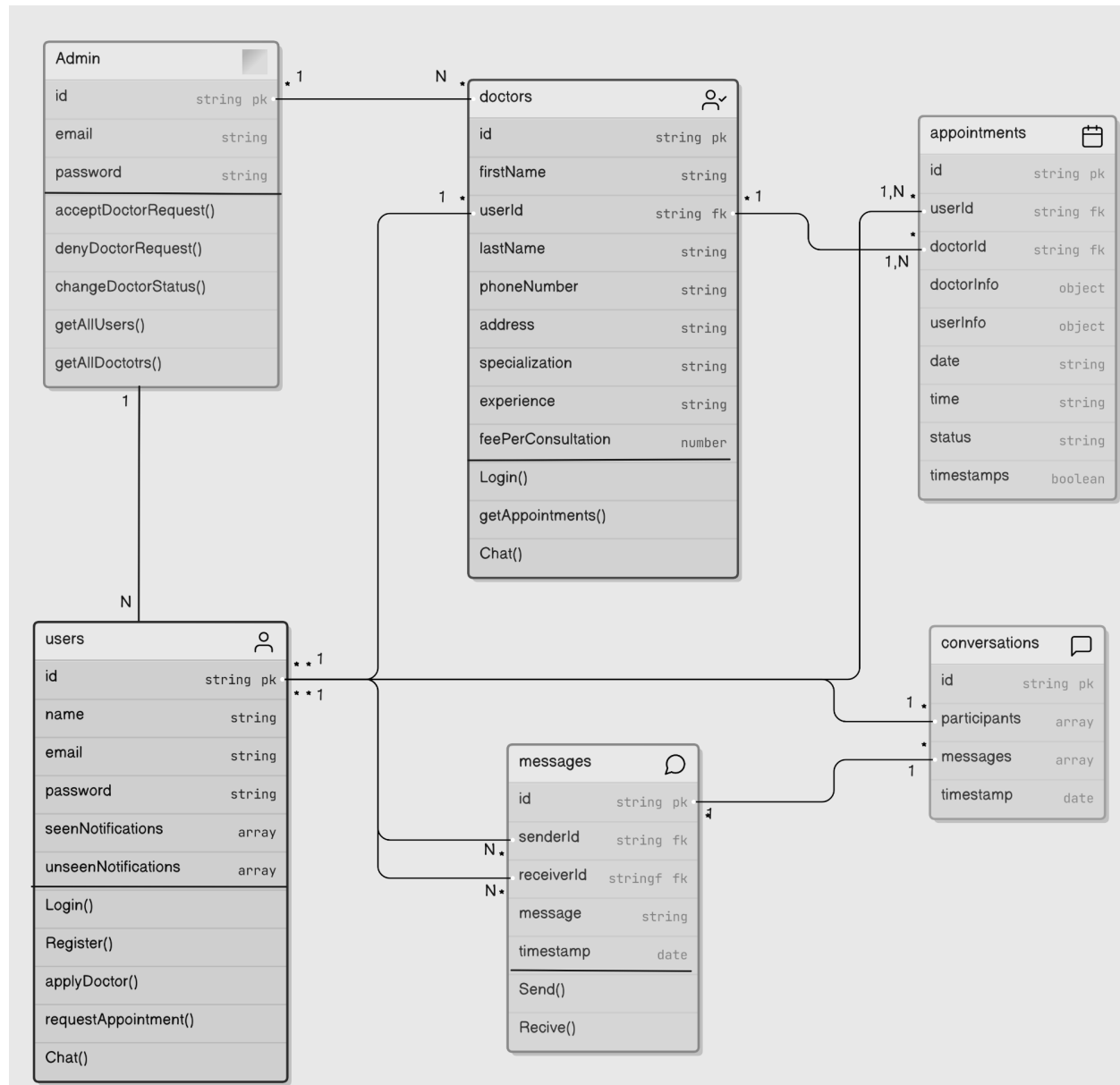


Figure 4.13 Class Diagram of the system

CHAPTER FIVE

5. Conclusion

In conclusion, the implementation of a digital healthcare platform that connects patients and doctors can significantly enhance the flexibility and accessibility of healthcare services. By leveraging advanced technologies such as telemedicine and secure messaging, the system can overcome traditional barriers associated with geographical constraints and rigid appointment schedules. This innovative approach to healthcare delivery aims to streamline practice management, improve patient engagement, and provide timely medical attention.

The proposed system will be beneficial to both patients and healthcare providers. Patients will experience greater convenience and personalized care, while doctors will benefit from optimized workflows and reduced administrative burdens. Overall, this digital platform represents a transformative step towards a more efficient, patient-centered healthcare ecosystem.

5.1 Recommendation

The digital healthcare platform presents numerous opportunities for future expansion and enhancement. As technology evolves, the system can be updated with new features to further improve its functionality and user experience. Integration with other healthcare systems, such as electronic health records and patient monitoring devices, will create a more comprehensive and seamless healthcare delivery experience.

It is recommended to continuously gather feedback from users to identify areas for improvement and to ensure the platform remains responsive to the needs of patients and healthcare providers. Additionally, promoting digital literacy and ensuring equitable access to technology will be crucial in addressing healthcare disparities and maximizing the platform's impact.

In summary, the development of a digital healthcare platform at the Defense University College of Engineering holds immense potential to revolutionize healthcare delivery. By prioritizing flexibility, accessibility, and patient-centered care, this system can greatly benefit patients and healthcare providers, leading to improved health outcomes and a more efficient healthcare system.

Reference

1. <https://www.sciencedirect.com/science/article/pii/S2667102622000791>
2. https://www.researchgate.net/publication/372051158_The_internet_of_medical_things_in_healthcare_management_a_review
3. [Abid Haleem](#),^a [Javaid](#),”<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8590973/>”
4. https://www.researchgate.net/publication/349540289_Doctor-Patient_Social_Networking_to_Improve_Specialist_Care_Coordination

Summary of Using the MERN Stack

MERN Stack Components:

- **MongoDB:** NoSQL database for storing data.
- **Express.js:** Backend web application framework for Node.js.
- **React:** Frontend library for building user interfaces.
- **Node.js:** JavaScript runtime for server-side programming.

Steps to Create a MERN Stack Project:

1. **Set Up the Environment:**
 - Install Node.js and npm.
 - Install and start MongoDB.
2. **Backend Development:**
 - Initialize a Node.js project.
 - Install Express and Mongoose.
 - Create an Express server (**server.js**).
 - Connect to MongoDB with Mongoose.
3. **Frontend Development:**
 - Create a new React app using **create-react-app**.
 - Set up a proxy in **client/package.json** to connect to the backend.
 - Develop React components and pages.
 - Fetch data from the backend using API endpoints.
4. **Database Operations:**
 - Define Mongoose models for your data.
 - Create routes for CRUD operations in Express.
5. **Deploying the Application:**
 - Build the React app for production.
 - Serve static files using Express.
 - Deploy to a cloud provider like Heroku, AWS, or Vercel.

By following these steps, you can set up, develop, and deploy a full-stack web application using the MERN stack.