

**Hashemite University**  
**Prince Al-Hussein Bin Abdullah II Faculty for**  
**Information Technology**  
**Department of Computer Information Systems**

## **Medicina**

*A project submitted  
In partial fulfillment of the requirements for the degree of  
Bachelor in Computer Information Systems*

### **By**

Abdalrahman Adnan AL-Mahrouq	2239464
Saja Mahdi AL-shraa'h	2234629
Tasneem Ibrahim Abukaff	2134965
Yaqoub Yousef Mahmoud	2238883
Mohammed Mamoun ALhaj	2232797

### **Supervised by**

Dr. Subhieh (Moh'd Faraj) S. El-Salhi

### **Committee Member Names**

Dr. Mo'taz Alhami

**May/2025**

## **ACKNOWLEDGEMENT**

We are very thankful to everyone who supported us to complete this paper effectively. We would like to thank our advisor, Dr. Subhieh (Moh'd Faraj) S. El-Salhi for overseeing the project; making it happen, we would also express our gratitude to our parents and friends for the huge support they gave to us while making this project .

## **UNDERTAKING**

This is to declare that the project entitled “ Medicina ” is an original work done by undersigned, in partial fulfillment of the requirements for the degree “Bachelor in Computer Information Systems” at the department of Computer Information Systems Department, College of Prince Al-Hussein bin Abdullah II of Information Technology , the Department of Computer Information Systems (CIS) at the Hashemite University.

All the analysis, design and system development have been accomplished by the undersigned. Moreover, this project has not been submitted to any other college or university.

- Saja Mahdi ALshraa'h \_\_\_\_\_
- Tasneem Ibrahim Abukaff \_\_\_\_\_
- Yaqoub Yousef Mahmoud \_\_\_\_\_
- Abdalrahman Adnan AL-Mahrouq \_\_\_\_\_
- Mohammed Mamoun Alhaj \_\_\_\_\_

## CERTIFICATE

It is hereby certified that the project titled “*Medicina*”, submitted by **Saja Mahdi** (2234629), **Tasneem Ibrahim** (2134965), **Abdalrahman Adnan** (2239464), **Mohammed Mamoun** (2232797), and **Yaqoub Yousef** (2238883) in partial fulfillment of the award of the Degree of Bachelor in Computer Information Systems embodies original work done by them under my supervision.

Project Supervisor : \_\_\_\_\_

Department of Computer Information Systems

## **ABSTRACT**

*In today's rapidly digitizing world, access to efficient and user-friendly healthcare services remains a significant challenge, particularly across the Arab region. Patients often struggle to find suitable doctors based on specialty, location, or insurance, while clinics face difficulties in managing appointments, schedules, and communication with laboratories.*

*Our project, Medicina, addresses these challenges through a smart, web-based healthcare platform designed to connect patients, clinics, doctors, and labs in one seamless system. The platform enables clinics to manage schedules, receive bookings, and communicate efficiently, while patients can easily register, book appointments, upload documents, and track their medical history.*

*The system was developed using React.js for the frontend and is planned to use Laravel and PostgreSQL for backend development. Key features include real-time notifications, medical profile management, and a scalable architecture supporting multiple clinics under one environment.*

*Initial results demonstrate that Medicina significantly reduces administrative load, enhances accessibility, and improves overall user experience. Future enhancements such as AI-based doctor recommendations, telemedicine modules, pharmacy integration, and wearable device synchronization aim to transform Medicina into a comprehensive digital health ecosystem.*

# Table of Contents

<b>LIST OF TABLES.....</b>	<b>vii</b>
<b>LIST OF FIGURES.....</b>	<b>viii</b>
<b>CHAPTER 1: Introduction (<i>Medicina</i>).....</b>	<b>1</b>
1.1 Overview.....	1
1.2 Project Motivation.....	1
1.3 Problem Statement.....	1
1.4 Project Objectives.....	2
1.5 System Analysis.....	2
1.6 Project Results.....	2
<b>CHAPTER 2: Related Existing System (<i>Optional</i>).....</b>	<b>3</b>
2.1 Introduction.....	3
2.2 Existing Systems.....	3
2.3 Overall Problems of Existing Systems.....	4
2.4 Overall Solution Approach.....	5
<b>CHAPTER 3: ANALYSIS AND DESIGN.....</b>	<b>6</b>
3.1 Stakeholders .....	6
3.2 Functional Requirements .....	7
3.3 Use Case Diagram .....	9
3.4 Non-Functional User Requirements .....	11
3.5 UML Diagrams .....	14
<b>CHAPTER 4: IMPLEMENTATION .....</b>	<b>16</b>
4.1 Introduction.....	16
4.2 Implementation Tasks.....	16
4.3 System Interface.....	17
<b>CHAPTER 5: SUMMARY AND FUTURE WORK.....</b>	<b>21</b>
5.1 Future Enhancements.....	21
<b>REFERENCES.....</b>	<b>22</b>

## **LIST OF TABLES**

<b>Table 1</b> :Primary and Secondary Stakeholders of the Medicina.....	6
---	---

## LIST OF FIGURES

<b>Figure 1</b> :Shows The Normal Use Case Of Medicina And The Activities That Each Side.....	10
<b>Figure 2</b> :Sequence Diagram For User Registration And Login Process.....	12
<b>Figure 3</b> :Sequence Diagram For Patient Booking .....	13
<b>Figure 4</b> :Database Table.....	14
<b>Figure 5</b> :Uml Diagram.....	15
<b>Figure 6</b> :Home Page.....	17
<b>Figure 7</b> :Login Page.....	18
<b>Figure 8</b> :Clinics Page.....	18
<b>Figure 9</b> :Doctors Page.....	19
<b>Figure 10</b> :Appointment Booking page.....	19
<b>Figure 11&amp;12</b> :Responsive Design On Mobile.....	20

# CHAPTER 1

## Introduction (*Medicina*)

### 1.1 Overview

"Medicina - A Medical Web Platform" This project aims to build a comprehensive medical website called "Medicina" that displays various clinics within a specific region and facilitates communication between patients and clinics.

Medicina is a web-based platform that connects patients with different clinics. The system includes:

- User registration (patients)
- Clinic registration
- A dashboard for each user type
- Detailed clinic information display
- Appointment booking functionality

### 1.2 Project Motivation

- Improved Access to Healthcare: Simplifying appointment bookings and reducing wait times.
- Flexible Medical Management: Enabling clinics to efficiently manage doctors' schedules.
- Enhanced Trust: A doctor rating system by patients to increase transparency.
- Technological Integration: Integrating lab results with patient records to enhance medical decision-making.
- Patient Convenience: A simple user interface that allows patients to access their medical history and book appointments with ease.

### 1.3 Problem Statement

Many patients struggle to find suitable clinics near their locations or face difficulties accessing updated information about doctors and available times. Our platform addresses this issue by providing a unified, easy-to-use interface with reliable data.

## **1.4 Project objectives**

- Facilitate the process of finding suitable clinics
- Reduce the time and effort needed to book appointments
- Improve the quality of digital medical services
- Provide a system that can be expanded to support more advanced features such as initial diagnosis or remote consultation .

## **1.5 System Analysis**

User Types:

1. Patient: Create an account, browse clinics, book appointments
2. Clinic: Register, manage appointments, display doctor info

Database Entities:

- Patients table
- Clinics table
- Doctors table
- Appointments table
- Specialties table

## **1.6 Project Results**

- A fully functional website with an integrated clinic database
- Login system for patients and clinics
- Well-structured display of clinics and doctors
- Effective appointment booking system

## CHAPTER 2

### Related Existing System

#### 2.1 Introduction

This chapter discusses existing medical platforms and systems that offer similar services to our proposed system "Medicina." The goal is to understand how these systems function, their key features, and their limitations. This analysis will guide the development of our own platform by identifying areas of improvement and innovation.

#### 2.2 Existing Systems

##### 1. Vezeta: [1]

Vezeta was founded in February 2012 by Egyptian entrepreneurs Amir Barsoum and Ahmed Badr. The company began as "DrBridge" as an electronic medical record (EMR) solution provider for doctors, but later refocused its strategy to become a digital platform connecting patients with doctors, significantly transforming the healthcare sector in the Middle East and North Africa (MENA) region.

Features:

- Doctor search by specialty, city, and rating.
- Appointment booking and rescheduling.
- Some integration with insurance.

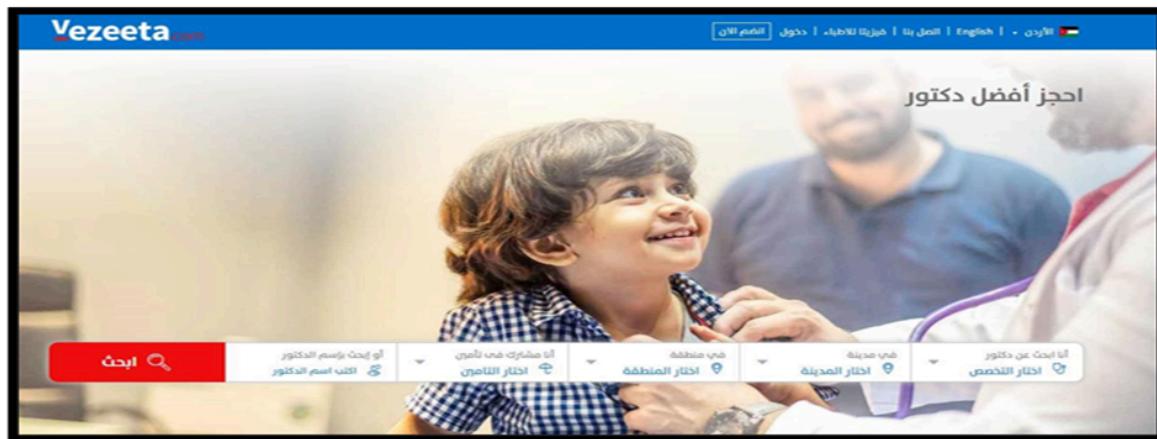


Figure 2.1:Home page for Vezeta website

Figure 2.1 The homepage of Vezeta's website serves as a comprehensive gateway to its digital healthcare services.

## 2. Altibbi: [2]

Altibbi is an Arabic-focused medical platform that offers telehealth services and online consultations. It connects patients to licensed doctors and provides health content and articles.

Features:

- 24/7 online consultations.
- Medical information and content.
- Limited appointment booking features for physical visits.



Figure 2.2 :Home page for Altibbi

Figure 2.2 Altibbi's homepage serves as a comprehensive portal for Arabic-speaking users seeking reliable medical information and services.

## 2.3 Overall Problems of Existing Systems

- Limited regional support: Some platforms do not fully support certain regions or cities.
- Weak insurance filtering: Insurance filtering is either limited or outdated, making it difficult for patients to know which doctors accept their coverage.
- Lack of small clinic inclusion: Many platforms focus on large hospitals or well-known clinics, leaving out smaller medical centers.
- Limited control for clinics: Clinics often cannot manage schedules or doctors dynamically within the platform.

- No comprehensive stakeholder system: Few platforms offer dedicated roles and dashboards for all parties (clinics, doctors, labs, admins).
- Insufficient patient filtering tools: Patients cannot always search using combined filters like insurance + location + specialty.
- No detailed visit data: Platforms rarely collect visit duration or track physician-specific schedules in a flexible way.
- Weak notification systems: Limited or no notifications/reminders via email/SMS for patients or clinics.

## 2.4 Overall Solution Approach

Our proposed platform, Medicina, addresses these issues with a comprehensive and flexible system that serves all stakeholders: patients, doctors, clinics, labs, and administrators.

Key Solution Highlights:

- Advanced Filtering: Search by specialty, city, insurance company, and available time slots.
- Flexible Scheduling: Clinics can manage doctor schedules, pause availability (e.g., vacations), and assign doctors under one clinic account.
- Patient-Friendly Interface: Simple booking, cancellation (with rules), medical history access, and profile creation with health background.
- Real-Time Availability: Appointment slots are based on each doctor's average visit time and dynamic availability.
- Notifications & Reminders: Automated notifications via email or SMS for new appointments and reminders.
- Detailed Profiles & Statistics: Clinics and doctors manage their profiles, update insurance partnerships, and access performance statistics.
- Admin Oversight: An admin panel to manage all users, approve/reject registrations, and monitor platform health.
- Lab Integration: Labs can upload test results to patient records for doctor access.
- Doctor Ratings: Patients can review doctors post appointment, improving trust and transparency.

## CHAPTER 3

### ANALYSIS AND DESIGN

#### **3.1 Stakeholders**

In the context of the Medicina, stakeholders include all individuals and entities who interact with the system or are influenced by its operation. Understanding their roles and requirements is essential to building a successful, user-centric platform.

**Table 1: Primary and Secondary Stakeholders of the Medicina.**

Stakeholder Type	Role
<b>Primary Stakeholders</b>	
Patient	A user who registers on the platform to search for clinics or doctors, book appointments, view their medical history, and manage their profile.
Doctor	A medical professional who logs into the system to review upcoming appointments, access patient records, and submit diagnostic reports.
Clinic	An entity responsible for managing affiliated doctors, clinic profile information, and appointment scheduling.
Lab	Uploads lab test results for patients based on doctor or clinic input.
<b>Secondary Stakeholders</b>	
System Administrator (Admin)	Oversee the platform's technical and operational aspects, verify registration requests from doctors, clinics, and labs, and monitor the system.
Medical Institutions / Partners	Organizations that sponsor or adopt the platform for internal use.

## **3.2 Functional Requirements**

System will be able to:

### 1. User Management

- Registration & Login

- Users (patients, doctors, clinics, labs) can register using an email address or via Google authentication.
- Each user type must complete their profile based on role:
  - Doctors: skills, specialization
  - Clinics: specialties, services
  - Labs: available tests and certifications
- Admin must approve clinic, doctor, and lab accounts before activation.

- Account Management

- Users can update their personal or professional details.
- The system sends notifications for:
  - Account approval or rejection
  - Profile updates
  - Login alerts

### 2. Appointment and Profile Management

- For Patients

- Search for doctors or clinics by:
  - Name
  - Specialty
  - Location
  - Insurance provider

- Book, reschedule, or cancel appointments.
  - View appointment history and medical records.
- For Doctors
    - View scheduled appointments.
    - Review patient history.
    - Add diagnosis, prescriptions, and follow-up notes.

- For Clinics
  - Manage associated doctors (add/remove, assign schedules).
  - Update clinic details and services offered.

### 3. Lab Integration

- Labs Can:
  - View test orders submitted by doctors.
  - Upload lab test results to the patient's profile.
  - Notify both doctor and patient when results are available.

### 4. Booking and Request Handling

- Patients Can:
  - Search for available appointment time slots.
  - Book appointments directly.
  - View appointment status (confirmed, canceled).

## 5. Notifications and Communication

- Notifications

- System sends email or in-app alerts for:
  - Appointment confirmations, reschedules, or cancellations
  - New messages from doctors
  - Uploaded test results

- Communication

- Patients and doctors can exchange messages within the platform.
- Clinics and labs can send updates or instructions to patients.

### 3.3 Use Case Diagram

Use case diagrams are a fundamental part of the high-level requirement analysis for Medicina. These diagrams illustrate how users (actors) interact with the system to perform specific tasks, providing a visual representation of the system's behavior from the user's perspective.

- In the context of this platform, the following components were identified to construct the use case diagrams:
  - Key Functionalities to be represented as use cases, including user registration, appointment booking, doctor profile browsing, lab result submission, and admin approval processes.
  - Actors interacting with the system, such as patients, doctors, clinic administrators, lab technicians, and system administrators.
  - Relationships between actors and use cases, defining which users can access certain functionalities and how they interact with system components.
  - The main purpose of the use case diagram is to model user interaction with the system in a clear and structured way. This helps stakeholders and developers understand user needs and ensure that the design phase aligns with system goals.

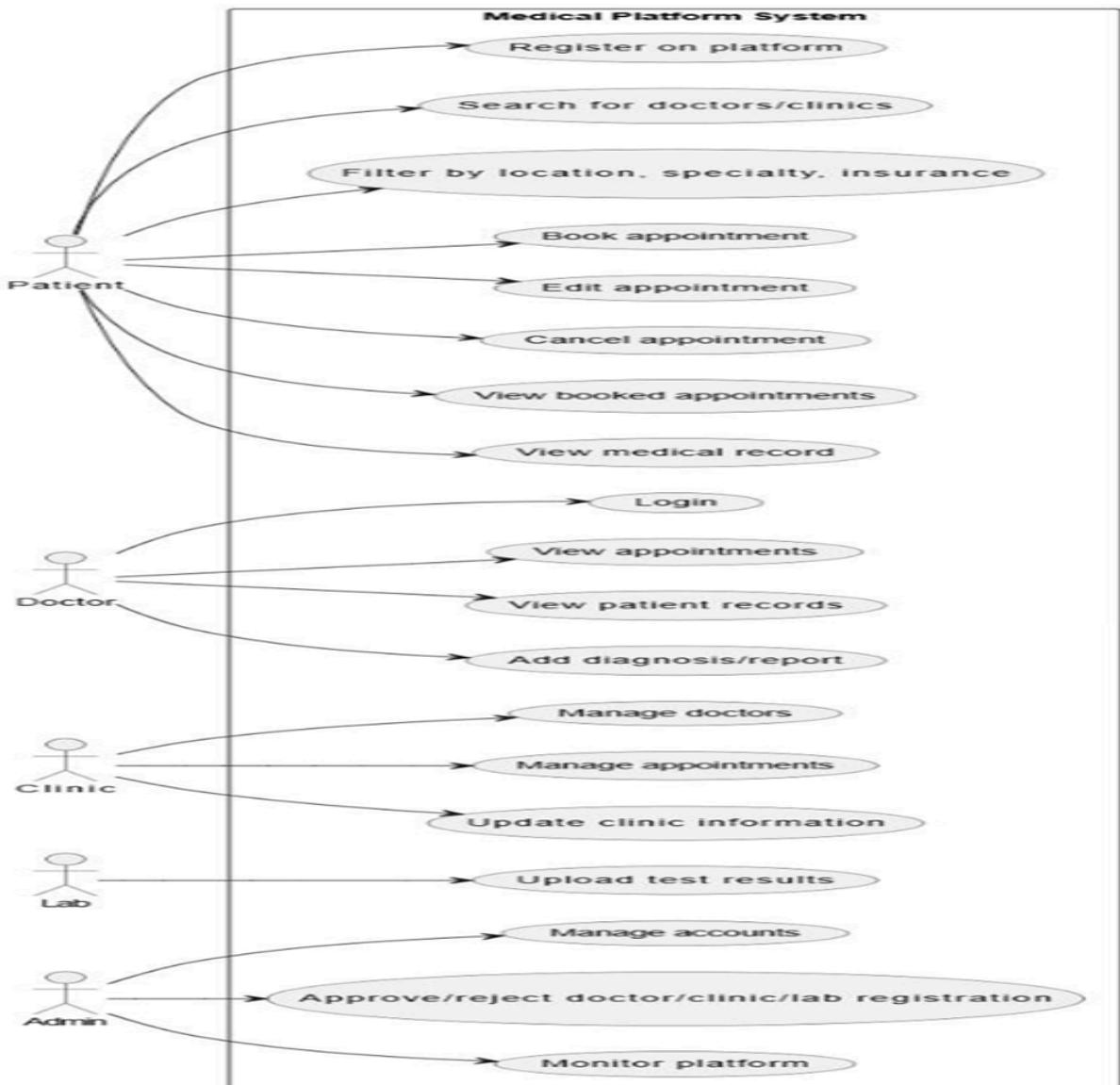


Figure 3.1 use case diagram

Figure 3.1: shows the normal use case of Medicina and the activities of each side.

This use case diagram outlines the key interactions between different users and the Medical Platform System. Patients can register on the platform, search for doctors or clinics, filter results by location, specialty, or insurance, and manage appointments by booking, editing, or canceling them. They can also view their medical records and previous bookings. Doctors can log in to view upcoming appointments, access patient records, and add medical diagnoses or reports. Clinics have the ability to manage associated doctors, handle appointment scheduling, and update clinic information. Labs can upload test results to patient profiles. Lastly, administrators oversee the entire system by managing accounts, approving or rejecting registrations for doctors, clinics, and labs, and monitoring platform activity to ensure smooth operation. This diagram helps illustrate the core functionality and role-based access within the platform.

### **3.4 Non-Functional User Requirements**

As part of the next development phase, we plan to implement the following non-functional requirements to ensure the system is efficient, secure, and easy to use:

- Performance
  - Backend API responses (once developed) will be optimized for fast response times.
  - We aim to make the React interface load within 2–3 seconds on a standard internet connection.
- Scalability
  - The system will be designed with simple scalability in mind, allowing it to handle more users or features in the future.
- Availability
  - We plan to host the frontend online with minimal downtime.
  - The system should be reliably available to users during testing and presentation.
- Security
  - We will use HTTPS to secure communication between the frontend and backend.
  - Basic security practices such as input validation and CSRF protection will be applied in the Laravel backend.
- Maintainability
  - Our code will be clean, well-structured, and commented to support future maintenance or improvements.
  - We will organize the project files to make development and debugging easier.

- Usability
  - The user interface will be simple, clear, and responsive (work on both desktop and mobile).
  - We will ensure the system is easy to navigate even for non-technical users.
- Portability
  - The system will support all major browsers (Chrome, Firefox, Edge, Safari).
  - We will provide a stable deployment, using the appserv environment

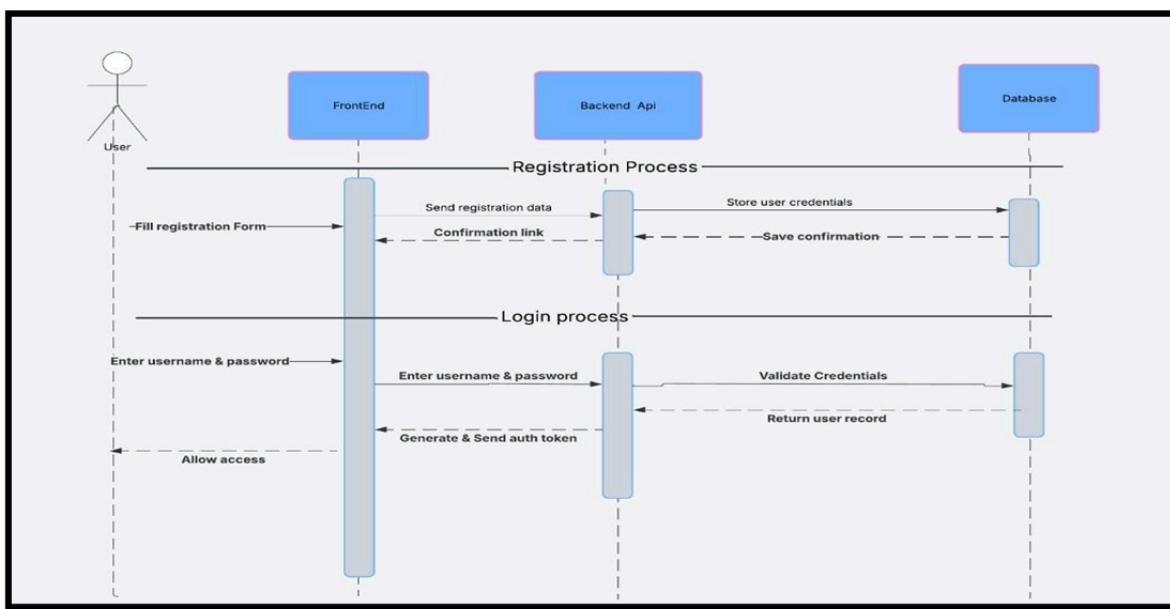


Figure 3.2 : Sequence diagram for patient booking

The diagram shows how Medicina handles user registration and login. During registration, the user submits their info via the frontend, which sends it to the backend to store in the database and generate a confirmation link. For login, the user enters credentials, which are sent to the backend for validation. If correct, the system returns user data and an authentication token, allowing secure access. The flow ensures smooth interaction between frontend, backend, and database.

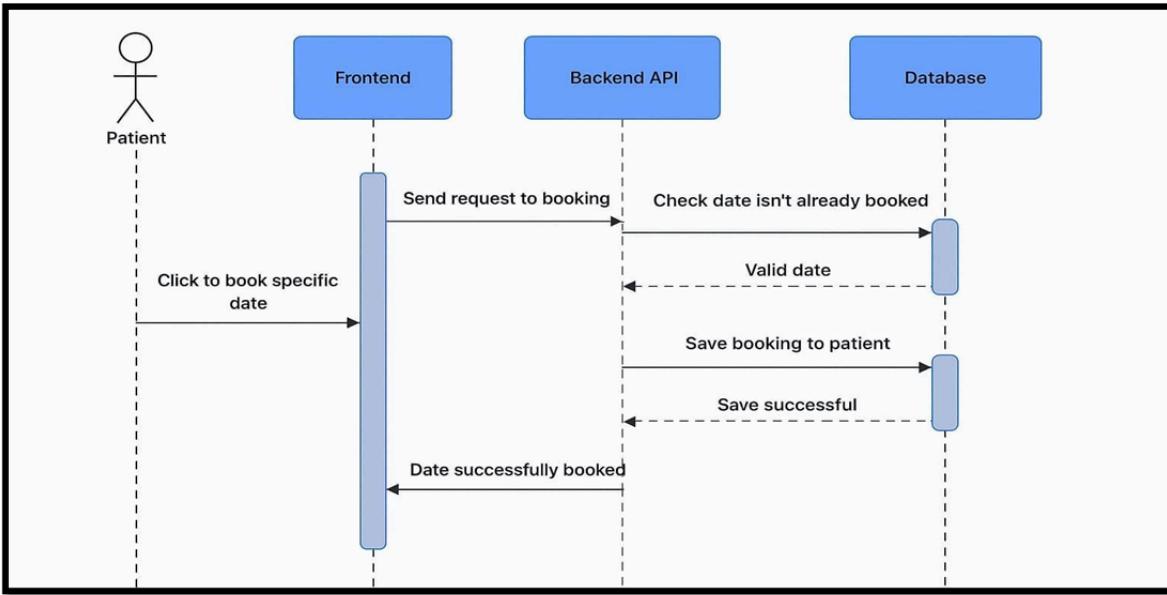


Figure 3.3 : Sequence diagram for patient booking

The diagram shows how a patient books an appointment through the Medicina system. The patient clicks on a specific date via the frontend, which sends a booking request to the backend. The backend checks with the database to ensure the date isn't already booked. If it's available, the backend saves the booking information for the patient in the database and returns a success message. The frontend then confirms to the user that the booking was completed successfully.

### 3.5 UML Diagrams:

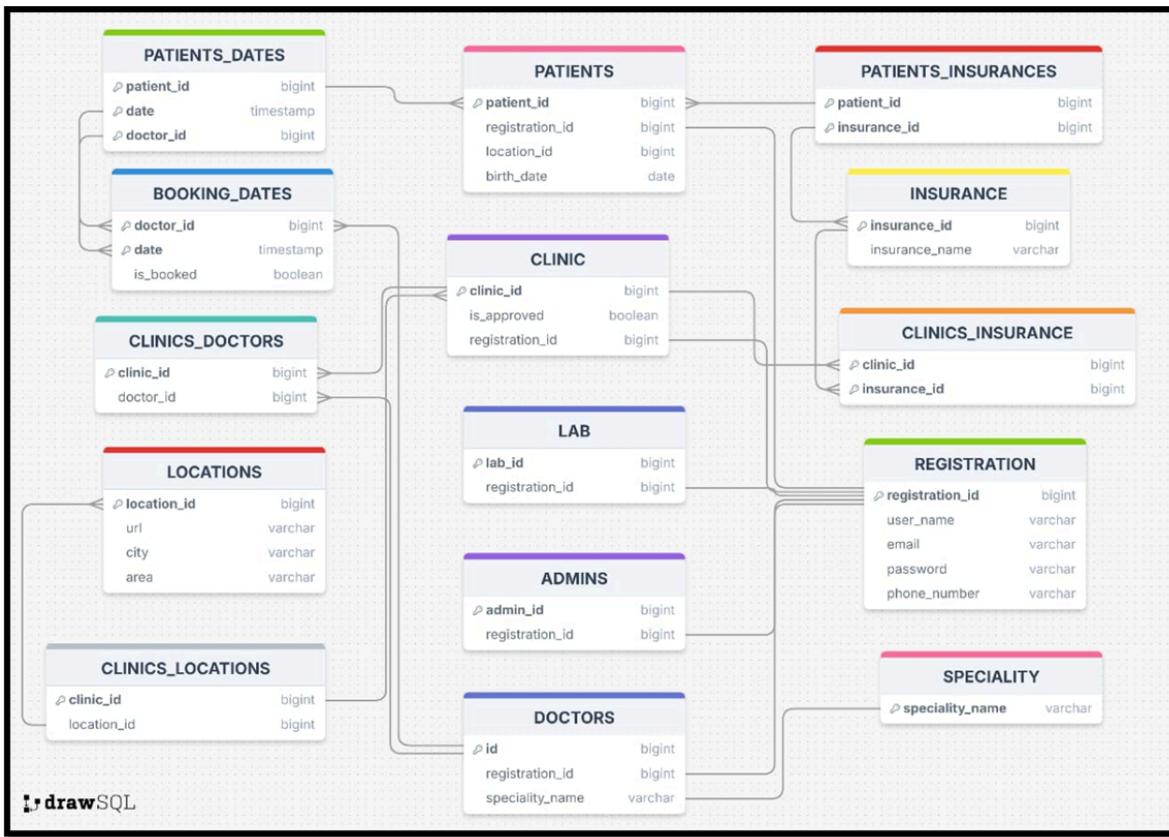


Figure 3.4 : Tables of Medicina Database

Figure 3.4 illustrates the database schema designed for the Medicina platform. It showcases how different entities in the system are related and how data is structured to support core functionalities such as user registration, clinic and doctor management, appointment booking, and insurance handling.

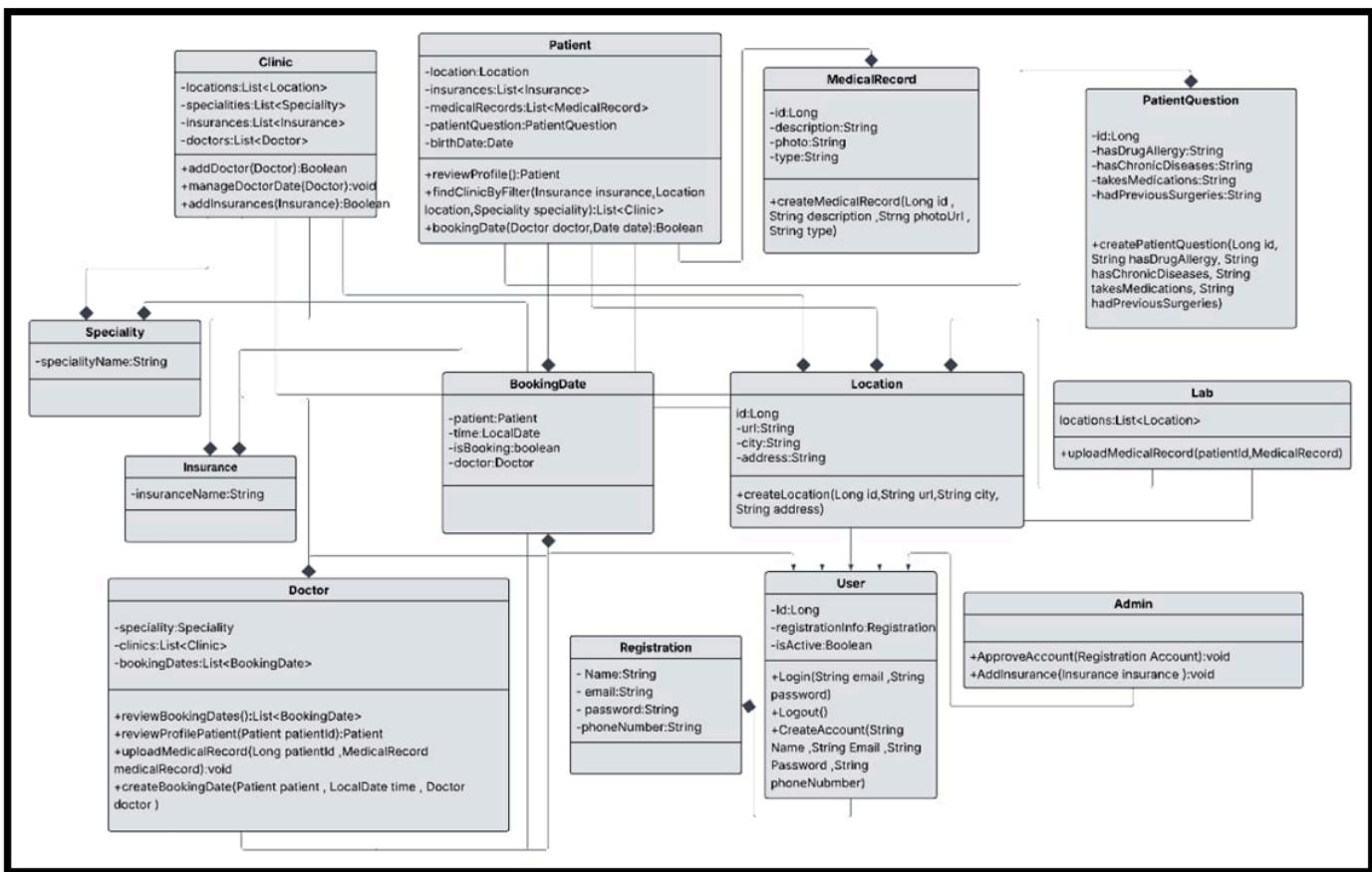


Figure 3.5 : Medicina

Figure 3.5 illustrates the **UML class diagram** of the *Medicina* system. This diagram represents the object-oriented structure of the application, showing the major classes, their attributes, methods, and the relationships between them. It is designed to support the core functionality of the system including user management, clinic administration, appointment scheduling, and patient health records.

## **CHAPTER 4**

### **Implementation**

#### **4.1 Introduction**

This chapter outlines the key implementation tasks carried out during the development of the system, along with a visual representation of the user interface, database tables, UMLs. It highlights the technologies used and provides screenshots to demonstrate how users interact with the system in its current state.

#### **4.2 Implementation Tasks:**

- Frontend Development
  - Technology Used:  
HTML, CSS, React.js, bootstrap
  - Description:  
The user interface was developed using React.js to ensure a dynamic and responsive experience. Components such as login forms, appointment booking, clinics, top banner, top navigation, Footer, etc., were created using reusable components and styled with CSS.
- Backend Development (Planned)
  - Planned Technology:  
Laravel (PHP), Postgresql
  - Description:  
The backend will be responsible for handling authentication, user data storage, and appointment management. Laravel will serve as the API backend, and Postgresql will be used as the relational database.

### 4.3 System Interface:

This section showcases the key interfaces and components of the system. Each figure represents an important part of the user interface and its functionality. These screenshots demonstrate visual design and how users interact with the system.



Figure 4.1: Home Page

Figure 4.1 demonstrates the homepage of the website, where users can view an introduction to the platform and navigate to different sections such as clinics, about, and login pages.



Figure 4.2: Login Page

Figure 4.2 explains how patients log in securely. The form includes email and password fields, with error validation and user-friendly design.

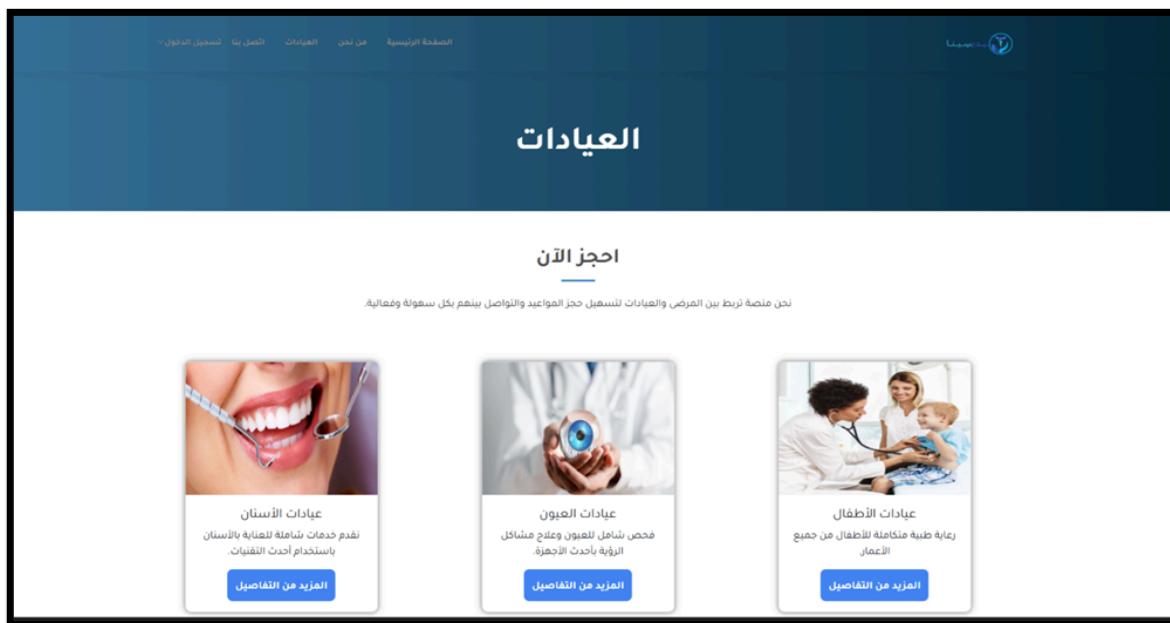


Figure 4.3: Clinics Page

Figure 4.3 explains how patients book appointments in their preferred clinics based on the description we provide for each.

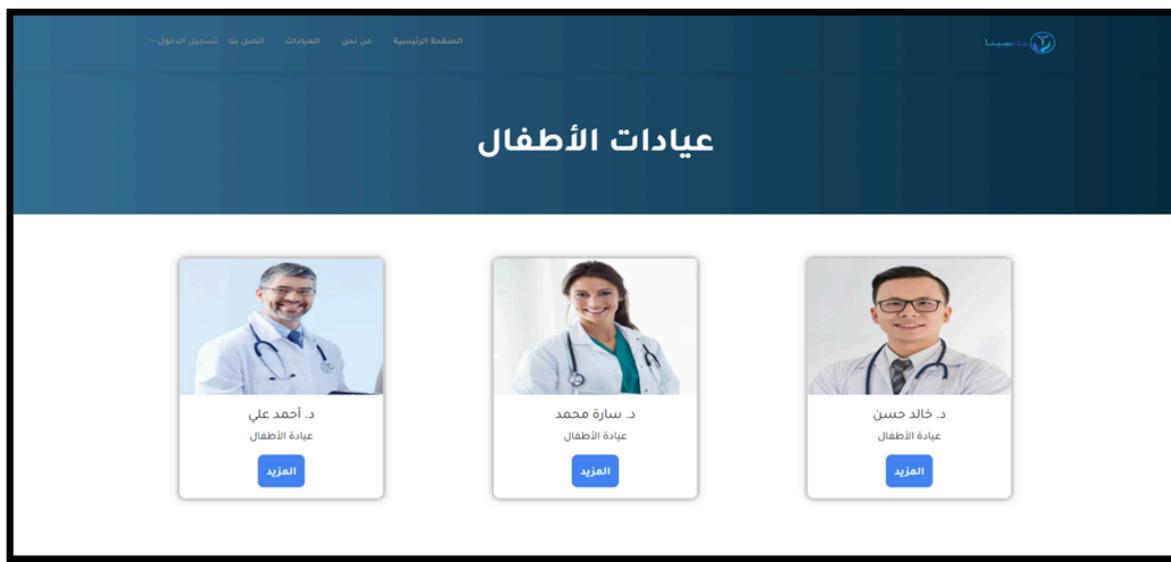


Figure 4.4: Doctors Page

Figure 4.4 displays the doctors available for a selected specialty.

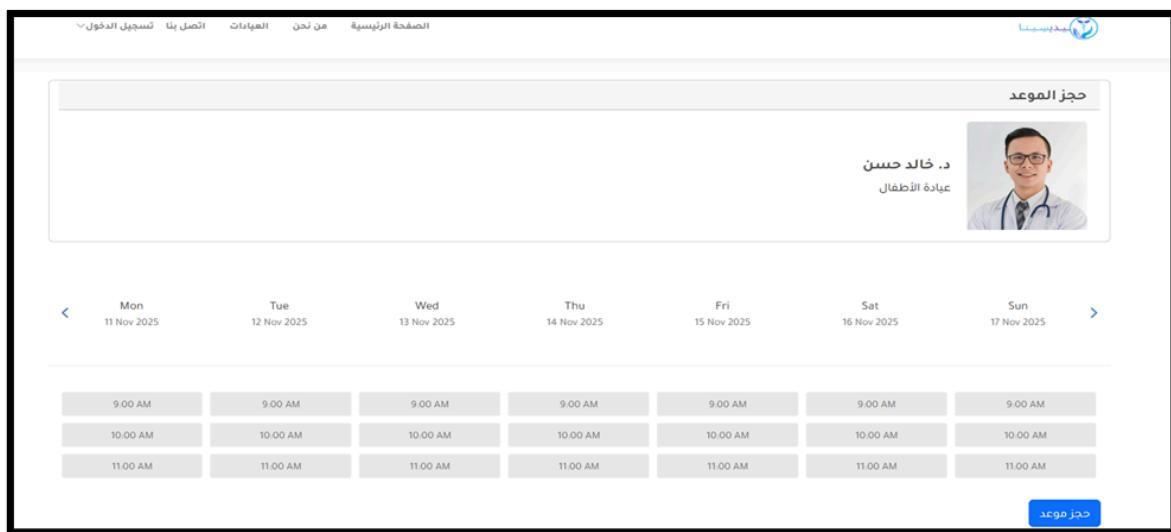
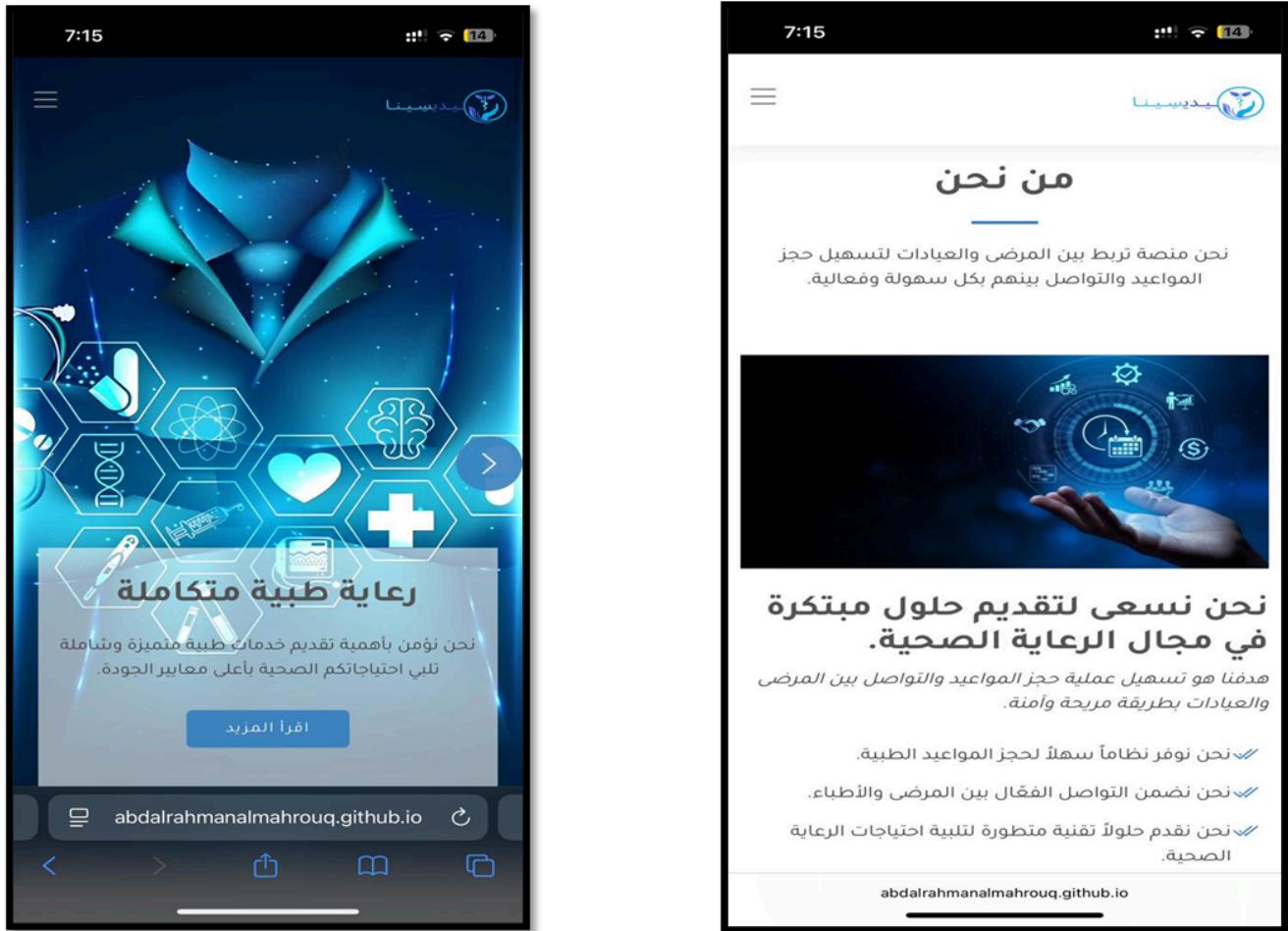


Figure 4.5: Appointment Booking Page

Figure 4.5, users can choose a clinic, select a date and time, and submit an appointment request.



Figures 4.6 and 4.7: Responsive Design On Mobile

Figures 4.6 and 4.7 show how the system adapts to smaller screen sizes such as smartphones, ensuring good user experience on all devices.

## **CHAPTER 5**

### **FUTURE WORK AND SUMMARY**

#### **5.1 Future Enhancements and Expansion Plans for Medicina**

As we look ahead, the Medicina platform is set to expand with several advanced features that will further enhance the healthcare experience:

**1-AI-Based Doctor Recommendation System:**

By analyzing user medical history, location, preferences, and feedback data, this module will provide smart suggestions to help patients find the most suitable doctor.

**2-Telemedicine Module:**

A secure video consultation system will be integrated, enabling remote diagnosis and follow-up, especially beneficial for patients in rural or underserved areas.

**3-AI Chatbot for Medical Consultation:**

An AI-driven chatbot offers 24/7 basic medical consultation, answering general questions and guiding patients through symptoms.

**4-Integration with National Health Records:**

To improve continuity of care, Medicina will aim to connect with national health record systems, enabling authorized doctors to access verified patient history securely and instantly. Syncing with popular health wearables will allow real-time tracking of patient vitals like heart rate, activity, sleep patterns, and more—giving doctors better data and enabling preventive care.

## REFERENCES

[1]

<https://jordan.vezeeta.com/en>

[2]

<https://altibbi.com/>

[3]

**React Documentation.** (n.d.). *React – A JavaScript library for building user interfaces.* Retrieved March 10, 2025, from

<https://reactjs.org/docs/getting-started.html>

[4]

W3Schools. (n.d.). *HTML Tutorial.* Retrieved March 1, 2025, from

<https://www.w3schools.com/html/>

[5]

W3Schools. (n.d.). *CSS Tutorial.* Retrieved March 4, 2025, from

<https://www.w3schools.com/css/>

[6]

W3Schools. (n.d.). *JavaScript Tutorial.* Retrieved March 7, 2025, from

<https://www.w3schools.com/js/>

[7]

Bootstrap. (n.d.). *Introduction to Bootstrap.* Retrieved January 10, 2025, from

<https://getbootstrap.com/docs/5.3/getting-started/introduction/>

