



Ministry of Higher Education and Scientific Research

University of Abdelhamid Ibn Badis Mostaganem

Faculty of exact sciences and computer science

Department of Mathematics and Computer Science

Project report of license computer science

Option: **Informatic Systems**

Theme:

Development of AI-assisted medical practice management tools

Student: AZZI Ahmed

Supervisors:

- Dr. BENKERDAGH Saliha
- Dr. HENNI Khadidja

University year 2024-2025

List of figures

Figure number	Figure name	Page number
Figure 1	UML Class diagram	21
Figure 2	UML Use Case diagram	22
Figure 3	Activity diagram	24
Figure 4	Deployment Diagram	26
Figure 5	Client Authentication Sequence Diagram	27
Figure 6	Client Signup Sequence Diagram	28
Figure 7	Book Appointment Sequence Diagram	29
Figure 8	Doctor Signup Sequence Diagram	30
Figure 9	Restore Password Sequence Diagram	32
Figure 10	Client Review Sequence Diagram	33
Figure 11	Write Prescription Sequence Diagram	34
Figure 12	Doctor Management Schedule Sequence Diagram	35
Figure 13	View Status Sequence Diagram	36
Figure 14	Write Report and Manage Sequence Diagram	37
Figure 15	Manage Appointment Sequence Diagram	38
Figure 16	Landing Page Interface	42
Figure 17	How to Book an Appointment Interface	43
Figure 18	Services Interface	43
Figure 19	FAQ interface	44
Figure 20	Contact Us Interface	44
Figure 21	Authentication Interface	45
Figure 22	Sign-up Interface	46
Figure 23	Forget Password Interface	47
Figure 24	Confirm Code Interface	47
Figure 25	Reset Password Interface	48
Figure 26	Dashboard Interface	49
Figure 27	Search for Doctor's Interface	50
Figure 28	View Doctor's Profile Interface	51

Figure 29	Appointment Booking Interface	52
Figure 30	Appointment History Interface (When Selecting An Appointment)	53
Figure 31	Appointment History Interface (When Not Selecting an Appointment)	54
Figure 32	View Appointment Details Interface	55
Figure 33	Manage Personal Account Interface	56
Figure 34	The Virtual Waiting Room Interface	57
Figure 35	Doctor Registration Interface	58
Figure 36	Doctor's Dashboard Interface	59
Figure 37	View Patients List Interface	60
Figure 38	View Patient Details Interface	61
Figure 39	View Medical Records Interface	62
Figure 40	View Appointments Interface	63
Figure 41	Add Appointment Interface	64
Figure 42	Generate Report Interface	65
Figure 43	Appointment Details Interface	66
Figure 44	Manage Schedule Interface	67
Figure 45	Admin Dashboard Interface	68
Figure 46	View Doctor's Interface	69
Figure 47	Manage Doctor's Profile	70
Figure 48	View Patients' Interface	71
Figure 49	Manage Applicants	72
Figure 50	View Applicant Details	73
Figure 51	"Contact Us" List Review Interface	74
Figure 52	Contact Us Message Details Interface	75
Figure 53	Error Interface	76

List of abbreviations

EMR	Electronic Medical Records
HIPAA	Health Insurance Portability and Accountability Act
UML	Unified Modelling Language
OCL	Object Constraint Language
RUP	Rational Unified Process
SRS	Software Requirements Specification
HTML	Hypertext Markup Language
CSS	Cascading Style Sheet
IBM	International Business Machine
ORM	Object Rational Mapping
UI	User Interface
KPI	Key Performance Indicator
AI	Artificial Intelligence
BMC	Business Model Canvas

FAQ	Frequently Asked Questions
------------	----------------------------

List of Tables

Table number	Table name	Table page
Table 1	Comparative Study of The Existing Platforms Table	15
Table 2	Advantages and Drawbacks of Existing Platforms Table	17
Table 3	Launch Costs Table	85
Table 4	Charges Table	85
Table 5	Employees Charges Table	86
Table 6	Amortization Table	86

Table of content

List of figures	4
List of abbreviations	6
List of Tables	7
Table of content	8
Chapter 1 General Context and Software Requirements Specification	15
1.1. Introduction	15
1.2. Existing services in the market	16
1.3. Comparative study	17
1.4. Needs in Healthcare Considered in Doctorino	21
1.5. Addressing Limitations: Doctorino's Approach	21
1.6. Problem Statement and Objectives of Our Work	22
1.7. Software Requirements Specification (SRS)	23
1.7.1. Introduction	23
1.7.2. Objective	23
1.7.3. Scope	23
1.7.4. Definitions, Acronyms, and Abbreviations	23
1.7.5. Overview	23
1.7.6. General Description	24
1.7.7. Specific Requirements	25
1.8. Conclusion	26
Chapter 2 Conceptual Design and Modeling	27
2.1 Introduction	27
2.2 Entity-Relationship Modelling for MongoDB	27
2.2.1 Data Model Diagram	27
2.2.2 Data Flow Diagram	28

2.3 Definition of UML	28
2.4 Our Modelling Approach	28
2.5. Conclusion	49
Chapter 3 AI Integration in Healthcare	50
3.1 Introduction to AI in Doctorino	50
3.1.1 Overview and Objectives	50
3.1.2 Benefits for Patients and Doctors	51
3.2 AI Models Implementation	51
3.2.1 ECG Analysis for Cardiologists	51
3.2.2 Mammography Analysis	52
3.2.3 Specialty-Based Model Delivery	54
3.3 Doctor Bot Feature	55
3.3.1 Conversational Interface Design	55
3.3.2 Appointment Booking Assistance	56
3.3.3 Health Advice Generation	56
3.4 Technical Architecture	57
3.4.1 Model Integration Framework	57
3.4.2 Data Processing Pipeline	58
3.5 Security and Privacy	58
3.5.1 HIPAA Compliance Measures	58
3.5.2 Data Protection Strategies	60
3.6 Evaluation and Future Development	61
3.6.1 Performance Metrics	61
3.6.2 Planned AI Enhancements	61
3.7 Conclusion	63
Chapter 4 Project Implementation	64
4.1 Introduction	64
4.2 Implementation Tools	64

4.2.1 Database Management System	64
4.2.2 Containerization Platform	64
4.2.4 Real-Time Communication	65
4.2.5 Object-Document Mapping (ODM)	65
4.2.6 User Interface Design	65
4.3. Presentation of the platform	66
4.3.1. Landing page	66
4.3.2. The patients interfaces (mobile)	72
4.3.2.1 Authentication	72
4.3.2.2. Sign-up	73
4.3.2.3. Forget Password	74
4.3.2.4 Reset password interface	75
4.3.2.5 Search for doctors	75
4.3.2.6 View Doctor profile	77
4.3.2.7 Appointment booking	78
4.3.2.8 View appointment details	79
4.3.2.9 Manage personal account	80
4.3.2.10 The Bot Interface	81
4.3.3. The doctor's interfaces (Desktop app)	82
4.3.3.1 Registration	82
4.3.3.2 Doctor dashboard	82
4.3.3.3 View patients list	83
4.3.3.4 View medical reports	85
4.3.3.5 View all appointments	86
4.3.3.6 Appointment details	88
4.3.3.7 Manage schedule	88
4.3.5. The admin interfaces (Website)	89
4.3.5.1 Admin dashboard	89

4.3.5.2 View doctors	90
4.3.5.3 Manage doctor profile	91
4.3.5.4 View users (patients)	91
4.3.5.5 Manage applicants	92
4.4. Perspective	92
General Conclusion	94
Annex Economic Study and Prototype Cost for Startups and BMC	95
Introduction	95
Business Model Canvas	97
Business Plan	98
Launch Costs	98
Charges	98
Employees charges	99
Amortization	99
Conclusion	99
Bibliography	100

Abstract

The emergence of information and communication technologies has revolutionized how organizations manage information, including healthcare institutions. This report investigates the design and implementation of Doctorino, a digital health platform that leverages Electronic Medical Records (EMRs) to enhance patient care. Doctorino facilitates informed clinical decision-making for doctors by providing real-time access to patient medical histories within a user-friendly interface. The platform further improves healthcare delivery through efficient appointment scheduling and management functionalities. By streamlining doctor-patient interactions and enabling data-driven care, Doctorino has the potential to improve healthcare outcomes and patient satisfaction. This report analyses the technical aspects of Doctorino's EMR integration, appointment management system, and user interface design, evaluating their contribution to the platform's effectiveness in a clinical setting. Additionally, the report explores the potential impact of Doctorino on patient privacy and data security, outlining best practices for ensuring HIPAA compliance within the platform.

General introduction

The world of technology has revolutionized various aspects of our lives, and the field of healthcare is no exception. In this era of digital advancements, we are introducing a cutting-edge medical platform that aims to enhance the patient-doctor interaction and streamline the appointment management process.

Our project is centred around developing a user-friendly online platform where patients can easily search for doctors and schedule appointments, while doctors can efficiently manage their schedules, appointments, and access patient records. By leveraging the power of technology, we aim to simplify the healthcare experience for both patients and doctors.

With our platform, patients will have the convenience of searching for doctors based on various criteria such as specialization, location, and availability. They can view detailed profiles of doctors, including their qualifications, experience, and patient reviews, helping them make informed decisions when choosing a healthcare provider. Once a suitable doctor is found, patients can easily book appointments online, eliminating the need for lengthy phone calls or in-person visits.

For doctors, our platform will provide a comprehensive set of tools to manage their appointments effectively. They can view and update their schedules in real-time, ensuring optimal time management and minimizing scheduling conflicts. Additionally, doctors will have access to patient records and medical history, allowing them to provide personalized care and make informed decisions during consultations.

Furthermore, our platform will prioritize data security and privacy, ensuring that sensitive medical information is protected at all times. We will adhere to the industry's best practices and comply with relevant regulations such as the Health

Insurance Portability and Accountability Act (HIPAA) to maintain the confidentiality and integrity of patient data [2]

In summary, our project aims to revolutionize the healthcare industry by introducing a state-of-the-art medical platform. By providing an intuitive interface for patients to find doctors and book appointments, and empowering doctors with efficient appointment and record management capabilities, we strive to improve the overall healthcare experience and contribute to better patient outcomes.

Our report is structured into three main chapters that systematically detail the development of our project. The first chapter provides an in-depth presentation of the context surrounding our initiative, outlining the challenges and opportunities in medical office management. This chapter also specifies the system requirements of our solution, delineating the functionalities and technical specifications of the Doctorino platform based on comprehensive analysis and stakeholder input.

The second chapter focuses on the modelling aspects using Unified Modelling Language (UML) and Object Constraint Language (OCL). It describes how we utilized the Unified Process (RUP) methodology to model the system, illustrating the architecture, components, interactions, and behaviours of Doctorino. This chapter showcases our approach to designing a robust and scalable solution that meets the complex needs of medical offices and enhances patient care.

The third chapter delves into the implementation phase of our solution. It discusses the practical steps taken to develop and deploy the Doctorino platform, including software development methodologies employed, technological choices made, and challenges encountered during implementation. This chapter concludes with insights gained from the implementation process and reflections on how Doctorino aims to revolutionize medical office management and improve healthcare delivery.

Chapter 1

General Context and Software Requirements

Specification

1.1. Introduction

In this chapter, we delve into the foundational aspects that shape the development and functionality of the Doctorino platform. We begin by providing an overview of the broader context within which Doctorino operates, exploring the challenges and opportunities in medical office management that have motivated the creation of this innovative solution. Subsequently, we detail the specific software requirements essential for the successful implementation of Doctorino. These requirements are meticulously defined to ensure that the platform not only meets the diverse needs of medical professionals and patients but also adheres to industry standards and best practices. By outlining these requirements, we aim to establish a clear framework that guides the design,

development, and eventual deployment of Doctorino, ultimately enhancing efficiency, communication, and patient care within healthcare environments

1.2. Existing services in the market

There are plenty of services in the market that encompass a diverse array of platforms aimed at enhancing healthcare accessibility and efficiency. Some of the most dependable examples include:

- **Practo (2008, India):** Established in 2008, Practo¹ has become a popular platform in India. It functions as a bridge between patients and doctors, allowing appointment booking, communication, and potentially even online prescription management (depending on regulations).
- **MyChart (1997, United States):** Developed in 1997, MyChart² is specifically used within the United States by patients of healthcare institutions that utilize Epic EHR (Electronic Health Record) systems. It's not a general telehealth platform, but rather a patient portal linked to their specific healthcare provider. MyChart allows patients to view medical records, schedule appointments (availability dependent), and message their providers.
- **ZocDoc (2007, United States):** Founded in 2007, ZocDoc³ is a well-established platform in the United States. It simplifies the process of finding doctors, booking appointments (both in-person and potentially video consultations with some providers), and managing healthcare interactions.

¹ www.practo.com (Accessed on May 25th, 2024)

² www.mychart.org (Accessed on May 25th, 2024)

³ www.zocdoc.com (Accessed on May 25th, 2024)

- **Doxy.me (2013, Global):** Launched in 2013, Doxy.me⁴ prioritizes secure and user-friendly video conferencing specifically designed for telemedicine consultations. HIPAA compliance ensures patient privacy. While Doxy.me has global reach, it doesn't offer features like appointment booking or patient record management.
- **SimplePractice (2014, United States):** Established in 2014, SimplePractice⁵ caters particularly to therapists and counsellors in the United States. It offers a comprehensive suite of tools for managing appointments, communicating with clients securely, and streamlining practice management tasks.
- **VSee (2008, Global):** Founded in 2008, VSee⁶ is a global telemedicine platform with a strong presence in the United States. It allows healthcare providers to conduct video consultations, manage appointments, and securely share patient records. VSee caters to a wider range of healthcare professionals than SimplePractice.
- **MDlive (2009, United States):** Launched in 2009, MDlive⁷ is a US-based platform that connects patients with various healthcare providers for online consultations. It addresses a broader range of medical needs compared to platforms focused on specific specialties like ZocDoc. This can include urgent care, mental health consultations, or chronic disease management.

1.3. Comparative study

The following table (Table 1) provides a comparative study of existing platforms dedicated to medical scheduling. This analysis aims to highlight key features, functionalities, and performance metrics across various platforms currently available in the market. By examining these factors side by side, we can gain valuable insights into their strengths, limitations, and potential applications within healthcare settings. This comparative overview serves as a resource for stakeholders looking to evaluate and select the most suitable platform to enhance scheduling efficiency and patient care delivery.

⁴ <https://doxy.me> (Accessed on May 25th, 2024)

⁵ www.simplepractice.com (Accessed on May 25th, 2024)

⁶ <https://vsee.com> (Accessed on May 25th, 2024)

⁷ www.mdlive.com (Accessed on May 25th, 2024)

Table 1: Comparative study of the existing platforms

ZocDoc	X	X		X	X	
Practo	X			X		
Doxy.me			X	X		X
SimplePractice	X			X		
VSee			X	X		
MDLive				X		
Epic		X		X	X	
MyChart	X		X			

The table below (Table 2) summarizes the advantages and drawbacks of existing platforms in the context of medical scheduling. This comparative analysis provides a comprehensive view of the strengths and limitations of each platform, offering insights into their usability, efficiency, and impact on patient care management. Evaluating these factors side by side facilitates a clearer understanding of how each platform addresses operational challenges and supports healthcare professionals in optimizing scheduling processes. This overview serves as a valuable reference for stakeholders seeking to identify the most suitable platform that aligns with their organizational needs and enhances overall healthcare service delivery

Table 2: Advantages and drawbacks of existing platforms

Platform	Advantages	Drawbacks
ZocDoc	• Online appointment scheduling • Patient reviews and ratings • Mobile app for easy access	• Limited to appointment scheduling and reviews
Practo	• Appointment booking and management • Teleconsultation services • Health records management	• Relatively high cost
Doxy.me	• Simple and easy-to-use interface • Compliant telemedicine • Free version available	• Limited features compared to paid alternatives
SimplePractice	• Comprehensive practice management • Appointment scheduling and billing • Telehealth integration	• Learning curve for setup and customization
VSee	• Secure video conferencing • Telemedicine solutions for clinics and hospitals	• Cost may be prohibitive for smaller practices

MDLive	• 24/7 access to telehealth services • Wide range of specialties covered • Integration with health plans	• Limited to telehealth services
Epic	• Clinical decision support and analytics	• High cost of implementation and maintenance • Complexity • Interoperability challenges
MyChart	• Appointment scheduling and reminders Secure messaging	• Depends on healthcare provider's adoption of Epic system

1.4. Needs in Healthcare Considered in Doctorino

Before discussing the specific features of Doctorino, it's crucial to outline the fundamental needs within the healthcare sector that our platform addresses. In today's healthcare landscape, there is a pressing demand for improved patient-provider communication, streamlined appointment management, and enhanced data security measures. Patients require easy access to comprehensive information about healthcare providers, including qualifications and patient feedback, to make informed decisions. Healthcare professionals seek efficient tools to manage their schedules and patient interactions effectively, while ensuring the confidentiality and integrity of patient data.

1.5. Addressing Limitations: Doctorino's Approach

Our healthcare platform stands as a beacon of trust and transparency, where every patient journey begins with meticulously crafted doctor profiles and genuine patient reviews, fostering a bond built on reliability and honesty. From appointment scheduling to seamless calendar synchronization, our expanded capabilities cater to the modern pace of life, ensuring no patient is left waiting with automated reminders and efficient waitlist management. With a fortress of

robust data security measures, we safeguard the sanctity of sensitive patient information, prioritizing confidentiality above all else. Embracing innovation, and redefining the healthcare experience with personalized solutions tailored to individual needs. By integrating with a multitude of Electronic Health Record (EHR) systems, we bridge accessibility gaps, empowering patients and doctors alike with universal compatibility. Our platform streamlined appointment bookings, we promote efficiency, empowering patients to take charge of their well-being while optimizing the healthcare journey for all involved.

1.6. Problem Statement and Objectives of Our Work

Medical office management presents a myriad of complex challenges, necessitating innovative solutions. Our project aims to address several key issues within this domain:

1. **Data Security:** Safeguarding patient information is paramount. Our system must implement robust security measures to protect sensitive data, ensuring confidentiality and integrity through effective authentication protocols [3].
2. **Interdepartmental Communication:** Ineffective communication among different doctors or medical departments can result in treatment delays, errors, and disjointed care. Our solution involves facilitating seamless communication channels within our health system, fostering collaboration and coordination among doctors to enhance patient care.
3. **Appointment Management:** Appointment Optimization: Ensuring optimal utilization of resources and minimizing patient wait times are crucial for enhancing the efficiency of healthcare delivery. Leveraging our reorientation algorithm and doctor scheduling system, our system will intelligently allocate appointment slots based on various factors such as doctor availability and patient needs. This dynamic approach to

appointment management aims to optimize patient flow, reduce wait times, and enhance overall patient satisfaction [4].

1.7. Software Requirements Specification (SRS)

1.7.1. Introduction

The Software Requirements Specification (SRS) for Doctorino serves as a comprehensive document outlining the functional and non-functional requirements essential for the development and deployment of our healthcare platform. This document meticulously defines the features, capabilities, and performance expectations of Doctorino. By detailing these specifications, we establish a clear roadmap for the design, implementation, and testing phases.

1.7.2. Objective

This document serves as a comprehensive guide for our project, outlining the essential requirements to be fulfilled. The intended recipient is the project member consisting of the developer: Azzi Ahmed, as part of my Master's graduation and startup project.

1.7.3. Scope

Doctorino will include a website for patient appointment booking and reminders, as well as functionalities for searching doctors, finding open doctor's office or pharmacies using a map. Additionally, a desktop application will be developed for

doctors to manage appointments, medical records, and semi-automatic prescription writing.

1.7.4. Definitions, Acronyms, and Abbreviations

Doctorino: The platform's name, Doctorino, is a creative blend of "Doctor" and the diminutive suffix "-ino," suggesting a friendly, approachable, and supportive digital health assistant. The name reflects the platform's mission to promote well-being and empower users to manage their health effectively with the guidance of a virtual doctor-like presence.

1.7.5. Overview

This document outlines the essential components and structure for the development of the software system. It begins with an Introduction detailing the purpose, scope, definitions, references, and an overview. The General Description section covers environmental factors, functions, user characteristics, constraints, assumptions, and dependencies. Specific Requirements delve into detailed functional requirements, performance measures, database structure, design constraints, and system attributes.

1.7.6. General Description

1.7.6.1. Environment

The system will consist of a web-based application accessible to patients for appointment booking and reminders. The desktop application will be designed for doctors to manage appointments, medical records, and prescriptions.

Interface:

- Website: Developed using HTML, CSS, JavaScript and other frameworks, compatible with modern web browsers.
- Desktop Application: Developed using Typescript, Electron, React.js and Tailwind CSS, compatible with Windows and macOS operating systems.

1.7.6.2 Functions:

The key functions of the system include:

- Patient Appointment Booking: Patients should be able to schedule appointments with doctors through the website.
 - Appointment Reminders: The system should send reminders to patients regarding their upcoming appointments.
 - Doctor Search: Patients should be able to search for doctors based on specialties, locations, or other relevant criteria.
 - Map Integration: The system should integrate with a map service to allow users to find open doctor's offices or pharmacies near their location.
- Doctor Desktop Application: Doctors should have access to a desktop application for managing appointments, medical records, and prescription writing.

1.7.6.3 User Characteristics:

The system will cater to two main user roles:

- Patients: Individuals seeking medical appointments and information. They should have basic computer and internet browsing skills.
- Doctors: Medical practitioners who will use the desktop application for managing appointments, medical records, and prescriptions. They should have a working knowledge of computers and software applications.

1.7.6.4 Constraints:

- Security Constraints: The system should adhere to strict data privacy and security regulations to protect patient information.

- **Compatibility Constraints:** The website should be compatible with modern web browsers, and the desktop application should be compatible with Windows, Linux and macOS operating systems.

1.7.7. Specific Requirements

1.7.7.1 External Interface Requirements:

The system should integrate with the following external services:

- **Map Service:** Integration with a map service to display open cabinets or pharmacies based on user location.

1.7.7.2 Functional Requirements:

- **Patient Appointment Booking:** Patients should be able to select a preferred doctor, date, and time for their appointment and receive confirmation.
- **Appointment Reminders:** Patients should receive automated reminders via email or SMS regarding their upcoming appointments.
- **Doctor Search:** Patients should be able to search for doctors based on specialties, locations, or other relevant criteria.
- **Map Integration:** Users should be able to view a map displaying open medical offices or pharmacies near their location.
- **Doctor Desktop Application:** Doctors should be able to manage appointments, view and update medical records, and generate semi-automatic prescriptions using the desktop application.

1.7.7.3 Performance Requirements:

- **Response Time:** The system should provide a responsive user interface, with fast loading times for web pages and desktop application functionalities.
- **Scalability:** The system should be able to handle multiple concurrent users without significant performance degradation.
- **Data Security:** The system should incorporate robust security measures to protect patient data from unauthorized access or breaches.

1.8. Conclusion

This chapter provided an overview of the system's objectives and scope, including the functionalities to be implemented. The subsequent chapters will delve into the detailed specifications and design of the website and desktop application as described in this requirements document.

Chapter 2

Conceptual Design and Modeling

2.1 Introduction

In this chapter, we will model our system using both Entity-Relationship Diagrams (ERD) and the Unified Modelling Language (UML), following the Rational Unified Process (RUP) methodology. The ERD helps illustrate the data structure and flow within the MongoDB-based system, while UML diagrams provide a comprehensive visualization of the architecture, functionalities, and interactions within the platform. Each diagram offers a structured representation of the system's components, their relationships, and the flow of operations. Through these visualizations, stakeholders gain insights into the design and functionality of the system, facilitating effective communication and understanding of its underlying structure.

2.2 Entity-Relationship Modelling for MongoDB

Although MongoDB is a NoSQL, document-oriented database, ERD can still be used to represent its collections and their relationships. In our project, we incorporate two primary types of ERD representations:

2.2.1 Data Model Diagram

The Data Model Diagram describes the logical structure of the MongoDB collections, including documents, fields, and references. This model provides clarity on how data is organized and how different collections relate to each other, such as through embedded documents or referenced IDs.

2.2.2 Data Flow Diagram

The Data Flow Diagram (DFD) illustrates the flow of data within the system, including external entities, processes, data stores (collections), and data movement. It is useful for visualizing how data is input, processed, stored, and output throughout the application.

2.3 Definition of UML

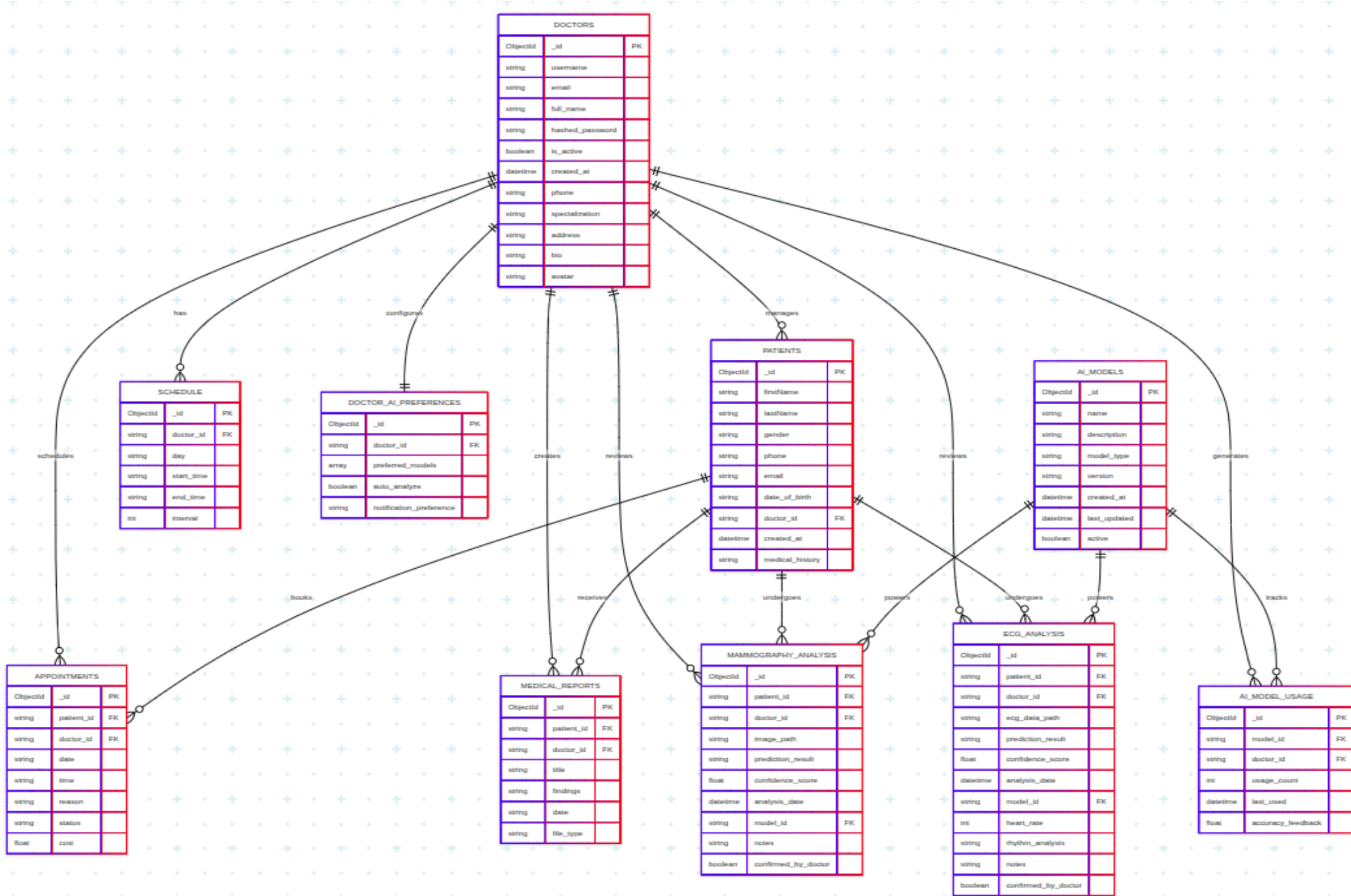
UML (Unified Modelling Language) is a graphical modelling language based on pictograms. It emerged in the field of software engineering, specifically in the context of object-oriented design. UML is commonly used in software projects to specify, visualize, modify, and construct the necessary documents for the development of object-oriented software. UML provides a standard modelling language for representing software architecture [5].

2.4 Our Modelling Approach

We will present different model diagrams as follows:

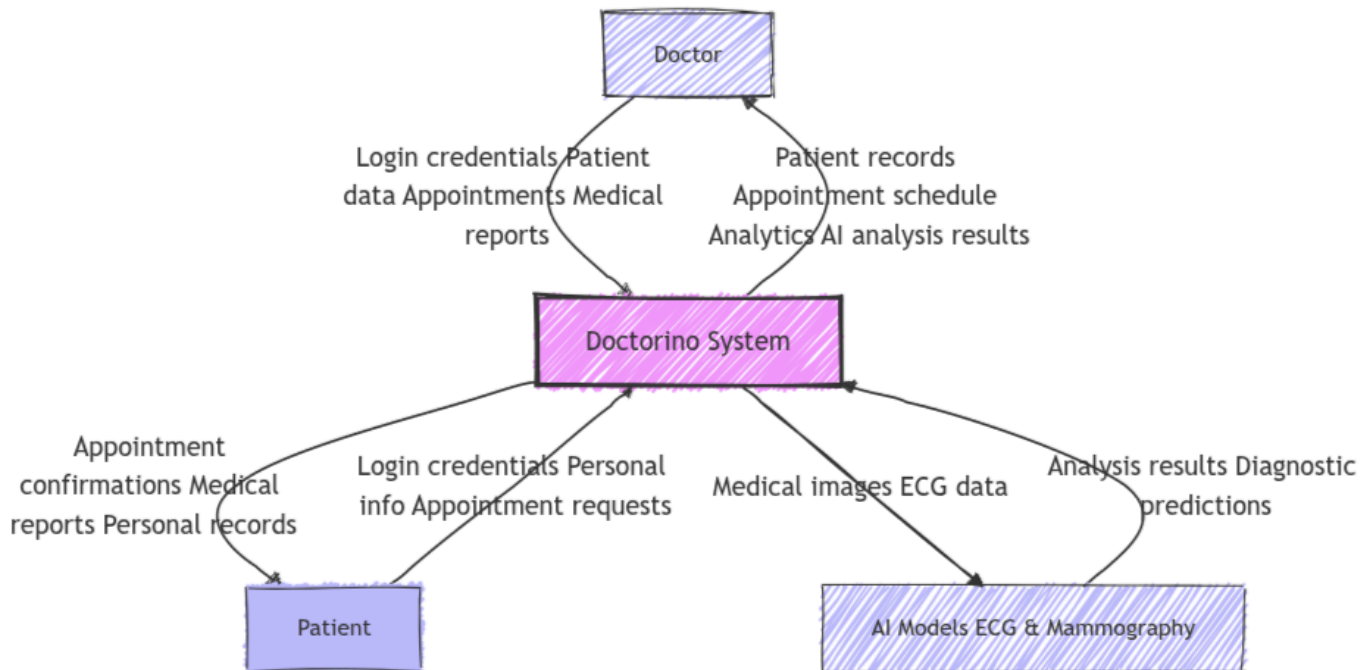
- **Entity-Relationship Diagrams (ERD)**
 - Data Model Diagram
 - Data Flow Diagram
- **UML Diagrams**
 - Use Case Model Diagram
 - Class Model Diagram
 - Activity Model Diagram
 - Sequence Model Diagram
 - Deployment Model Diagram

Description of The Data Model Diagram:



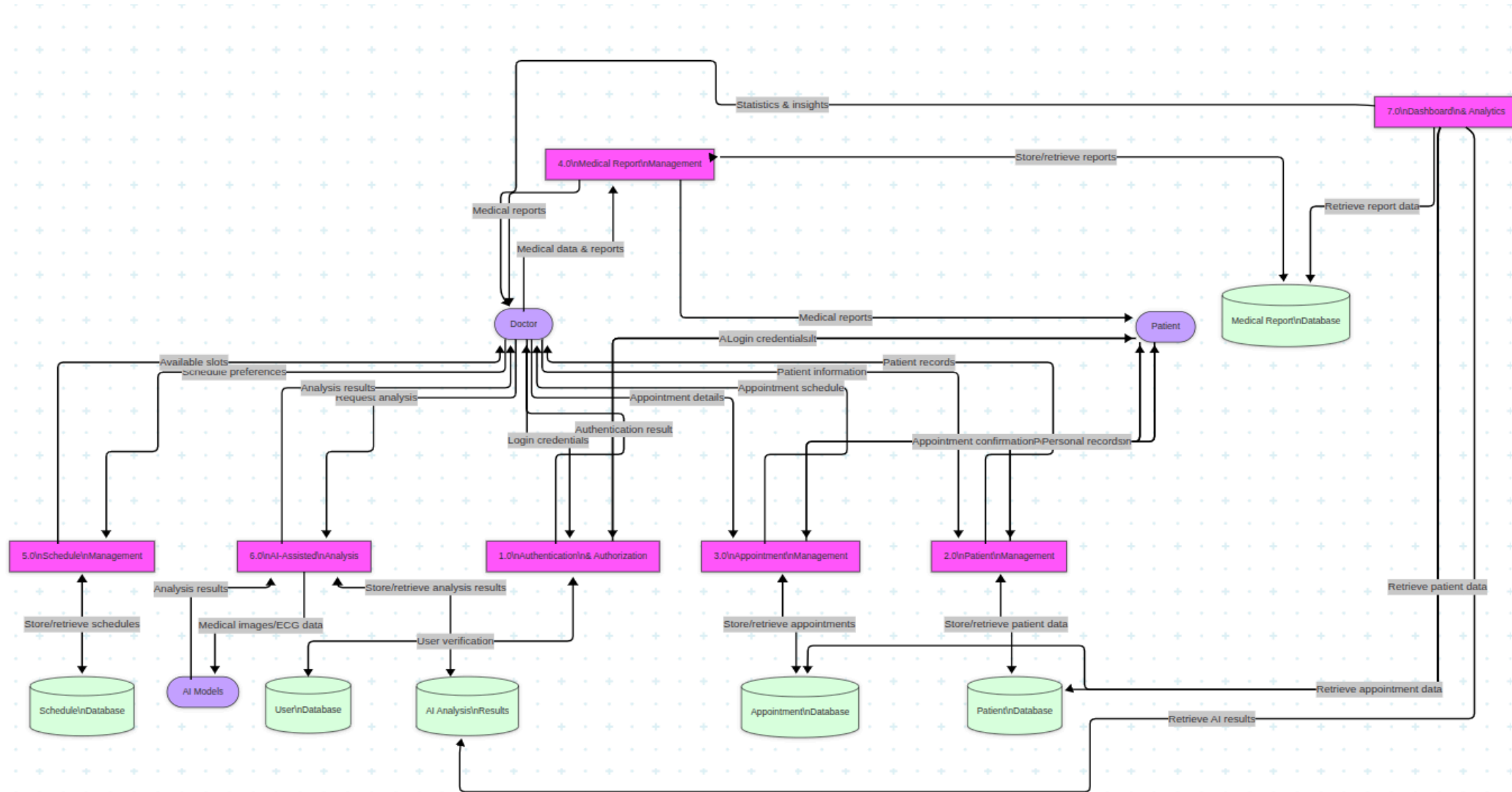
Description of The Data Flow Diagram: Level 0 DFD (Context Diagram):

A high-level Data Flow Diagram that shows a system as a single process with its interactions (data flows) between external entities (like users or other systems). It gives an overview of the entire system without detailing internal processes.



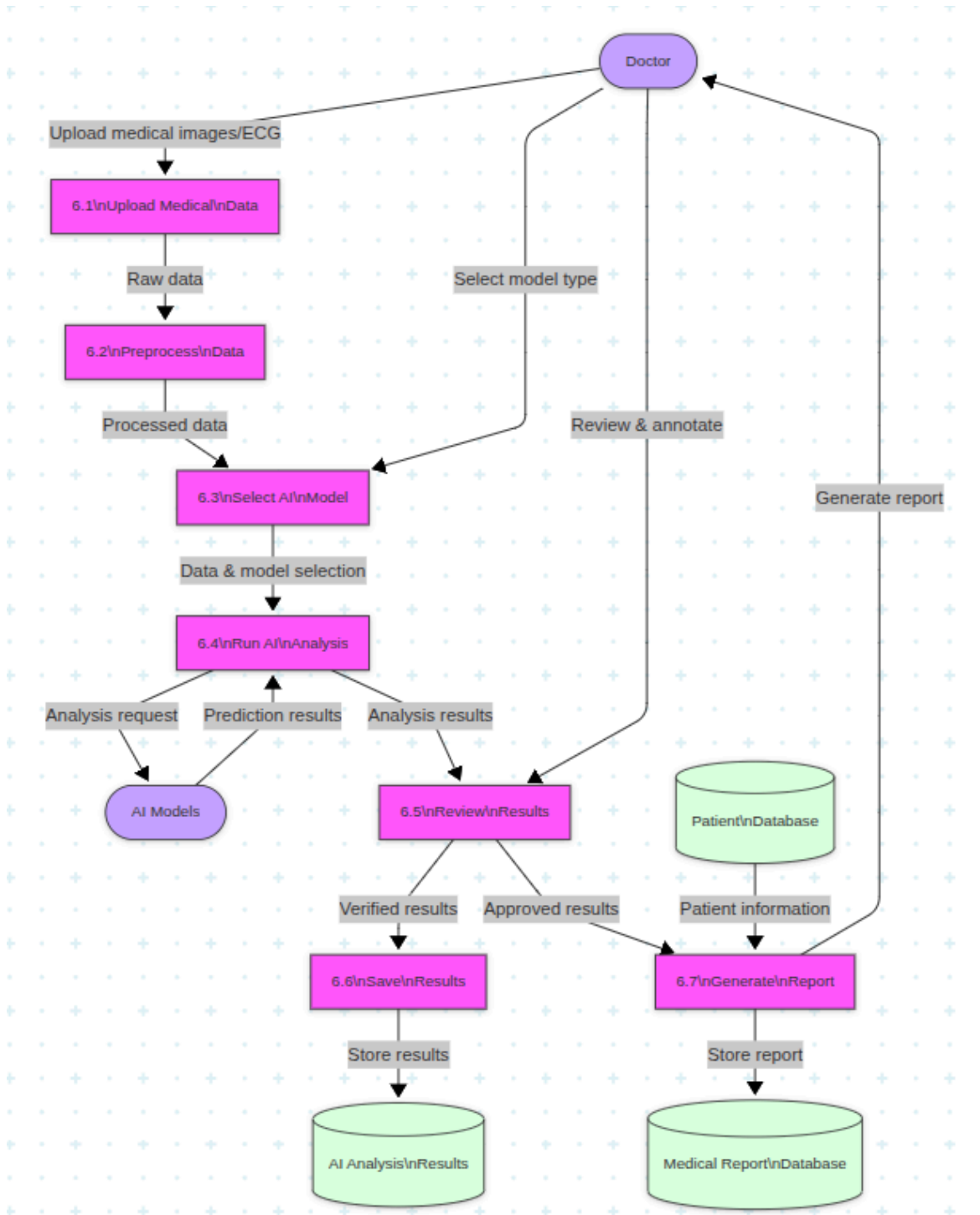
Level 1 DFD (Main Processes):

A Data Flow Diagram that breaks down the single process from the Level 0 DFD into its main sub-processes. It shows how data moves between these sub-processes, external entities, and data stores, providing more detail about the system's internal workings.



Level 2 DFD (AI-Assisted Analysis Process):

A more detailed Data Flow Diagram that breaks down a specific Level 1 process—such as AI-assisted analysis—into smaller sub-processes. It shows the flow of data between these sub-processes, data stores, and external entities to explain how the AI analysis works step by step.



Description of The Class Diagram:

The class diagram in (Figure 1) depicts the structure and relationships of a healthcare system. The main classes include Client, Doctor, Pharmacy, Secretary, Admin, Appointment.

Clients and Doctors are associated with a Location and can book Appointments with each other. Doctors can have Secretaries working with them and can write Reports for Appointments. Clients can review Doctors, and Admins manage the system. The diagram also includes enumerations for Account Status, Gender, and Appointment Status. Additionally, there is a class for List of applicants, which contains information about potential doctors applying to join the system. The classes have various properties and methods to represent their attributes and behaviours within the healthcare system.

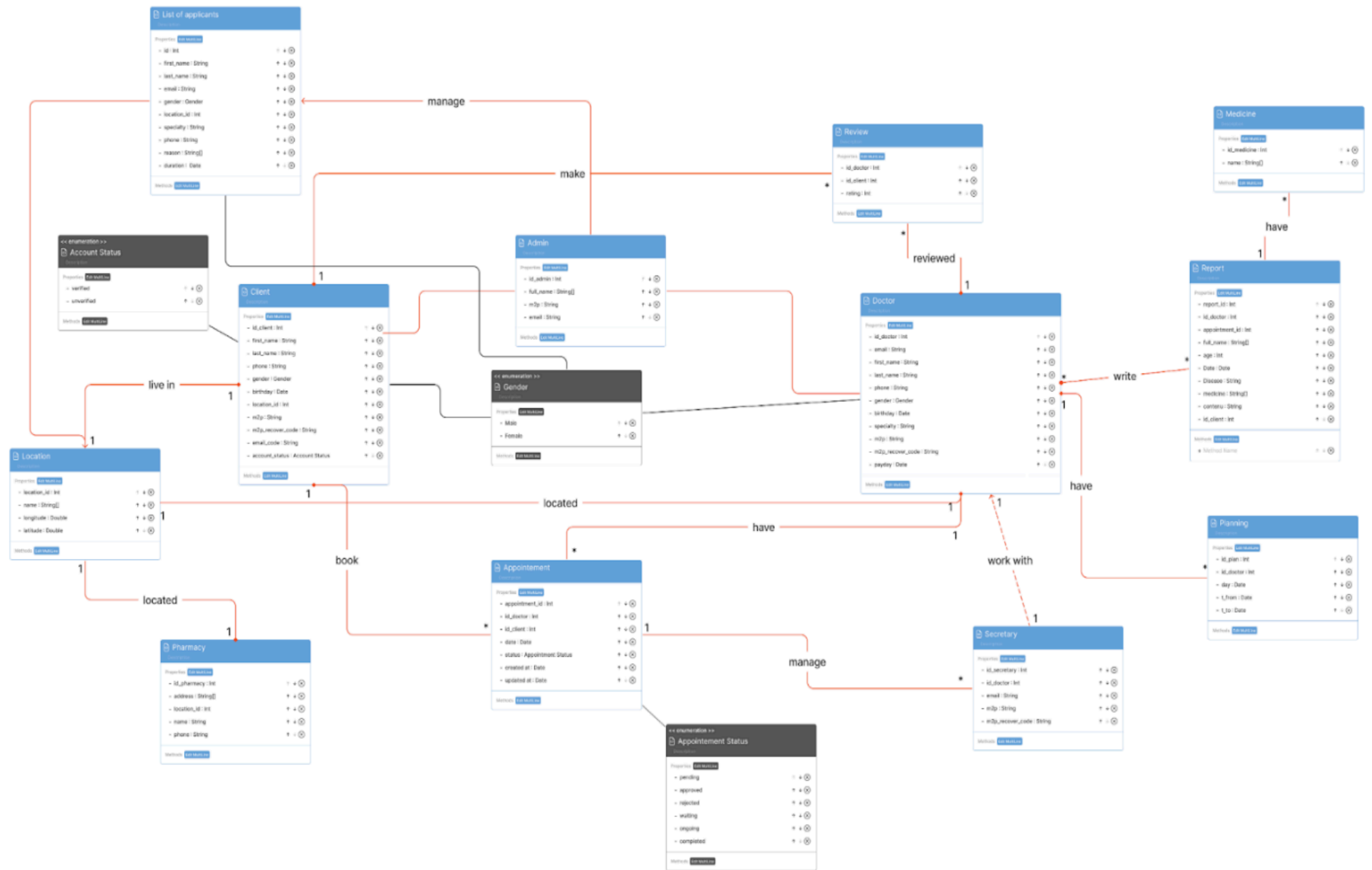


Figure 1 : UML Class Diagram

Description of the use case diagram:

The use case below in (Figure 2) specifies the events in a Doctorino system and how those events flow, it shows what goals wanted to reach by the 4 actors (Patient-Doctor-Admin-Secretary). Those actors share some events that are required to complete another event we're talking here about inscription & login and restore password actions surely every actor has his particular events because with different aims come different approaches, This Client role allows users to view analytics and access their account information

Doctors can manage reports, manage data (presumably medical data), their schedule, view analytics, and access their account.

Secretaries have the ability to manage appointments, prescriptions, and access their account. Admins have the most extensive access, including managing doctors, clients, the platform itself, and pharmacy accounts. Additionally, admins can handle signup processes, user authentication, manage reviews, check appointments, and restore or manage accounts.

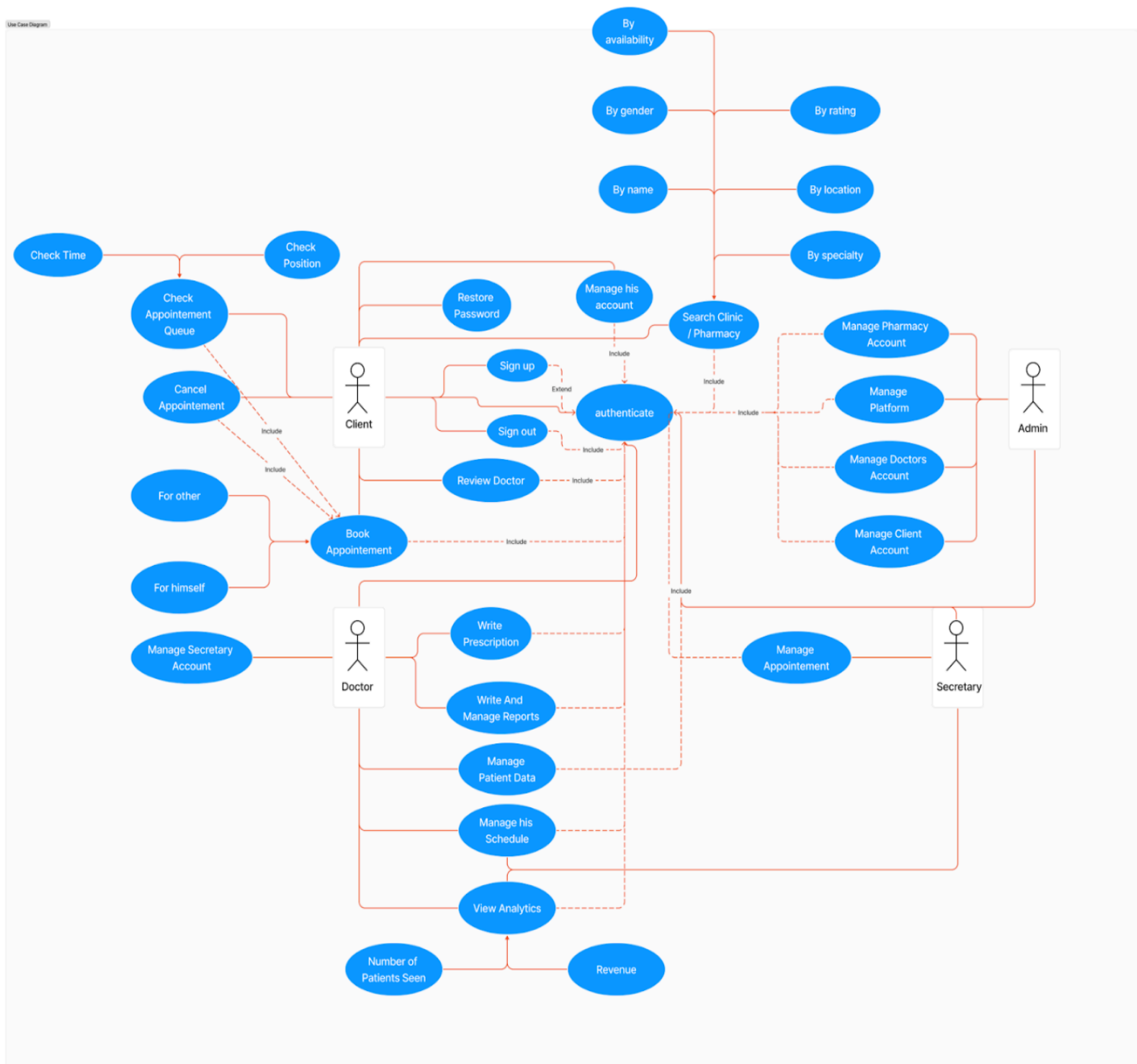


Figure 2: UML Use Case diagram

Description of the activity diagram:

The diagram in (Figure 3) illustrates the sequential steps required to access the primary feature for clients, which is "**Book Appointment.**" It begins with the

initial decision of visiting the website, where users are directed based on whether they are new or returning visitors. Upon successful login, users gain access to various features, including appointment booking. The booking process unfolds as follows:

1. Use the search function to locate clinics nearby or based on specific criteria.
2. From the search results, select the clinic that best meets your requirements.
3. Schedule your appointment by selecting a suitable date and time slot.
4. Upon completing the booking, you will receive a notification confirming the appointment.

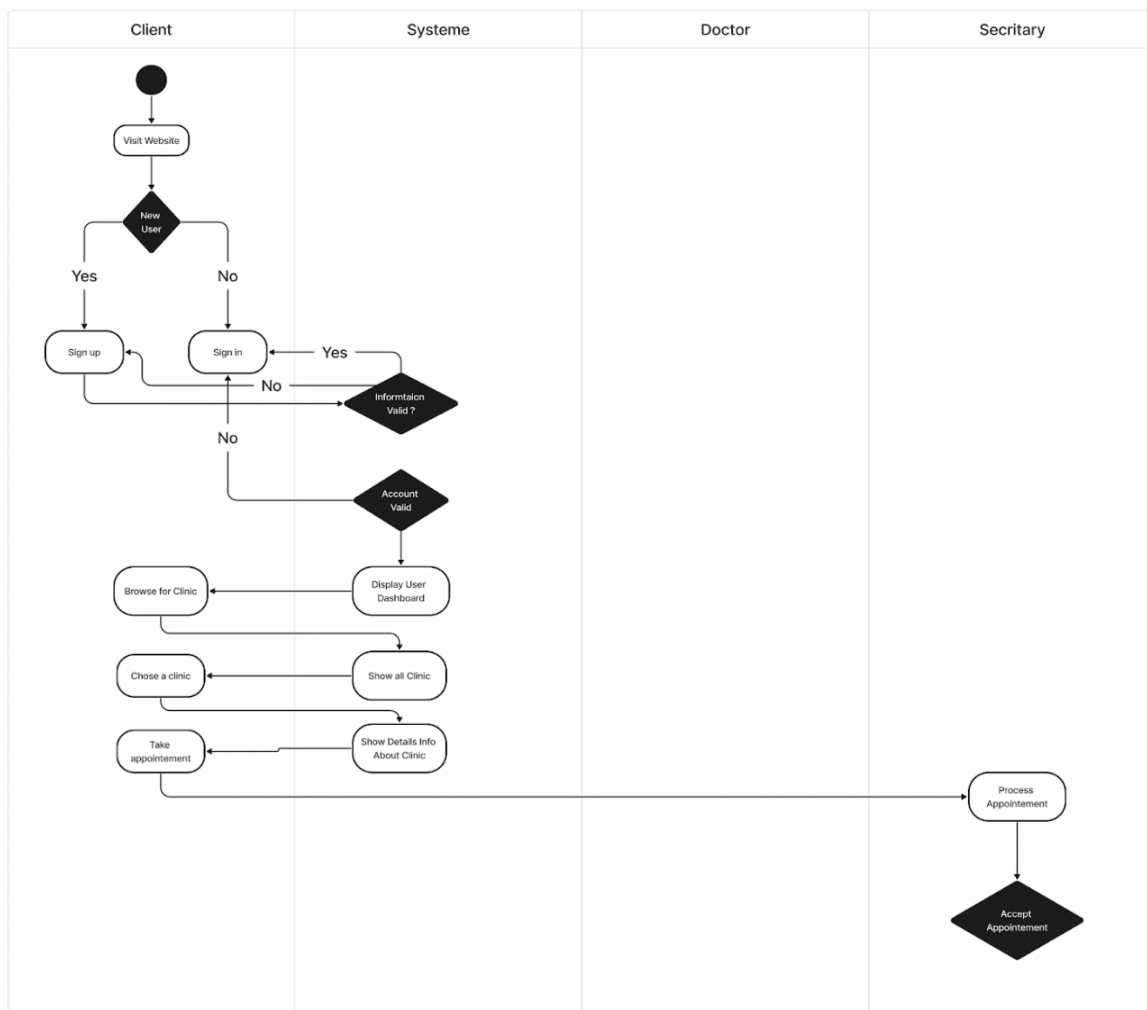


Figure 3: Activity diagram

Description of the deployment diagram

This deployment diagram in (Figure 4) illustrates the real-world infrastructure behind the Doctorino system. It maps out the software components, such as the web browser interface and the backend server, and how they interact with the physical hardware elements.

the doctor and secretary, use desktop applications and web browsers to access a server.

The server runs a MySQL database, Connector DB, Environment Node JS and a website/API

JavaScript and web browser is used on both the client and admin side

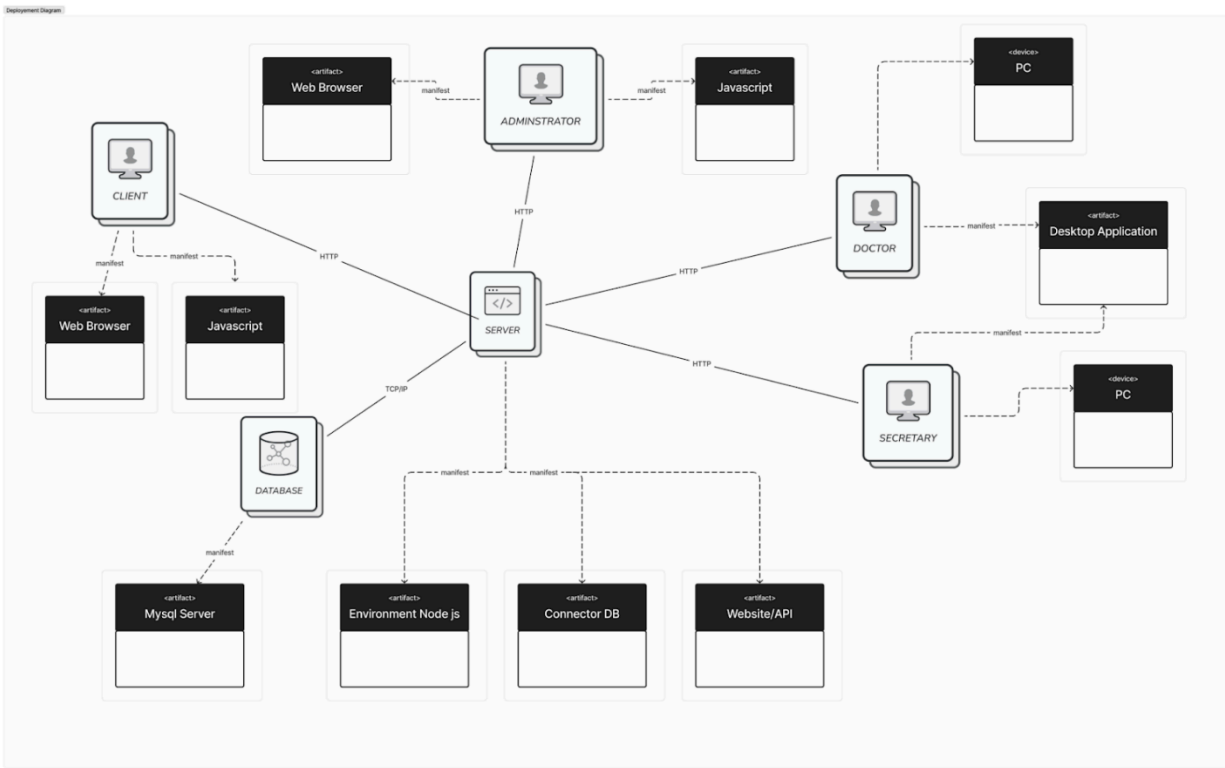


Figure 4: Deployment Diagram

Description of the Client Authentication Sequence Diagrams

Sequence Diagrams capture the interaction between objects in the context of collaboration without neglecting time. In our system all the sequence diagrams share the same elements Client System and database, in some of them we may find admin or secretary elements.

The diagram below in (Figure 5) focus on the action of “**Authenticate**”, to reach that point, we pass through the following stages:

1. The Client sends a login request to the System.
2. Information checking stage: The System then verifies the information by querying the database. Whether the information actually exists or not
3. Depending on the previous stage, you will know the status of your application either be rejected or redirected to the dashboard

Both cases contain messages explaining the outcome of the request (Successful message or Invalid data message)

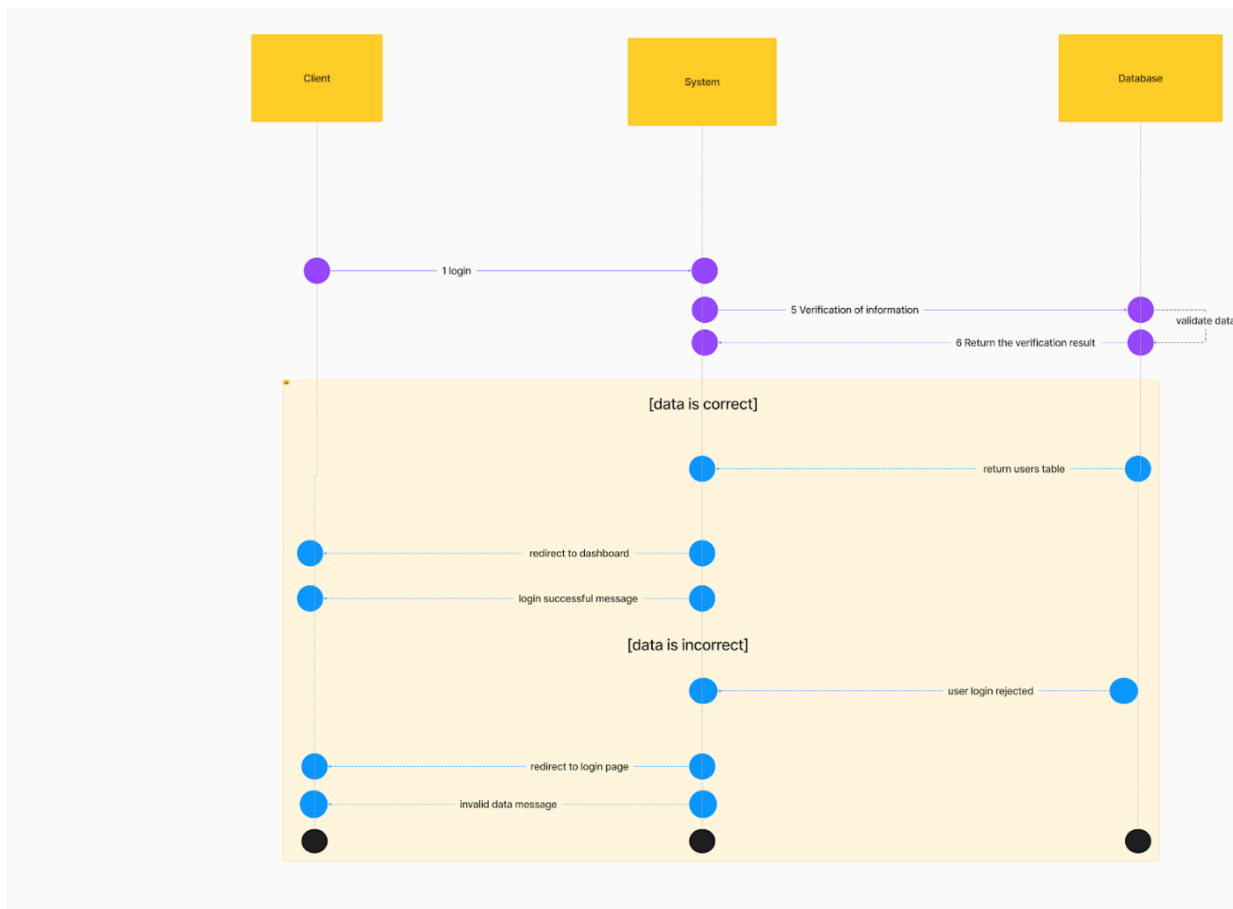


Figure 5: Client Authentication Sequence Diagram

Description

In the sequence diagram below in (Figure 6) the Client sends a **signup** request to the System, which verifies the information with the Database. In this stage; the process goes a step further than the previous diagram's second stage. Here, the system verifies if the information already exists in the database. If it does, registration is completed, and the user receives an instructive message and gets access to his account

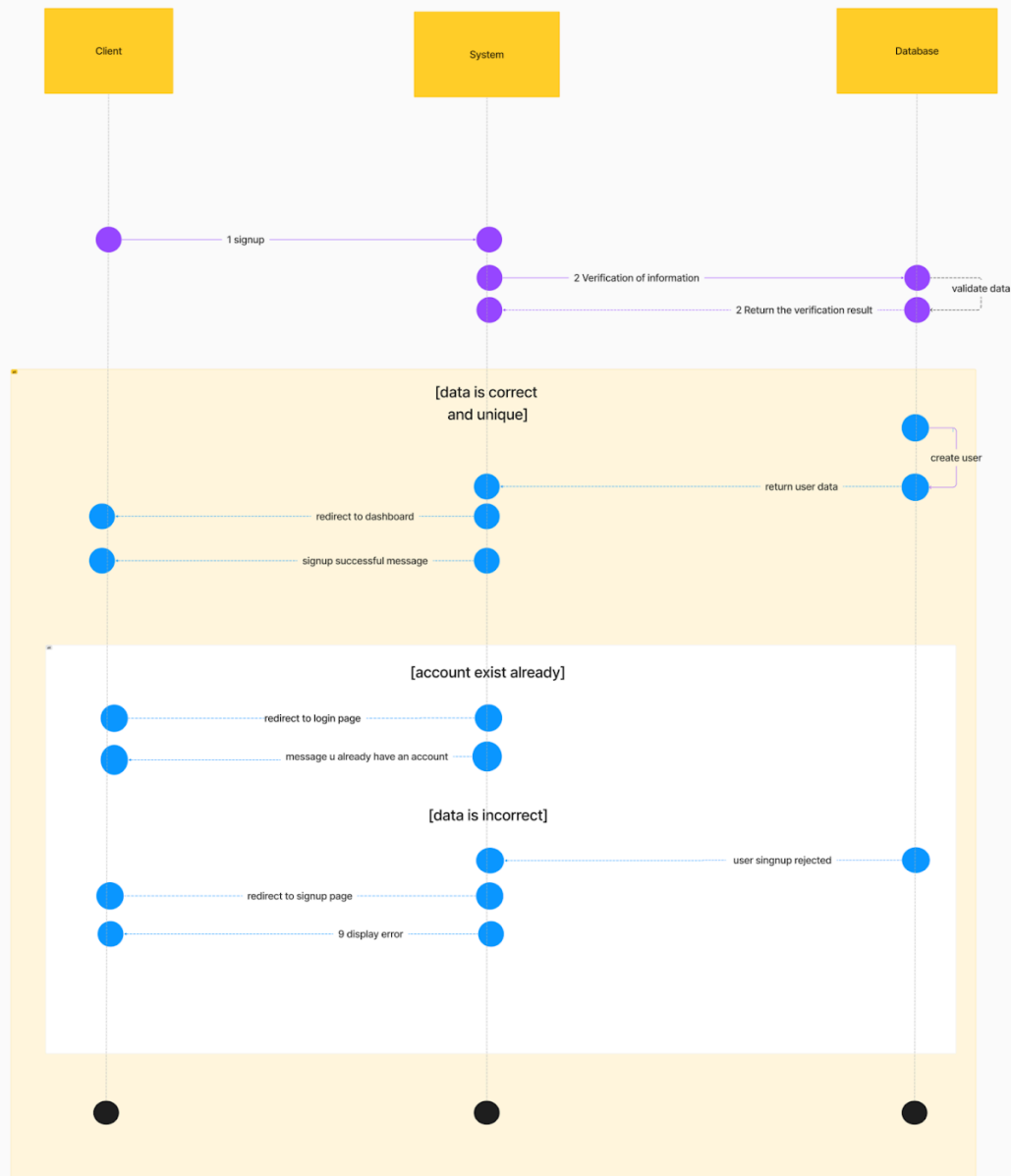


Figure 6: Client Signup Sequence Diagram

Description:

In the sequence diagram below in (Figure 7) it begins with the user logging into his account, the steps of which are explained in the diagram. The first goes to the search box and searches for the doctor in question. The system returns from the database with available doctors so that the user can **request an appointment**. the appointment gets stored in the database with a pending status while waiting to the second party (doctor & secretary) response

His request (Client) may be rejected or accepted, and he will know the moment the doctor makes his decision

- Accepted: The System updates the appointment status, and the Client is notified with an accepted message.
- Rejected: The System updates the appointment status, and the Client is notified with a rejected message.

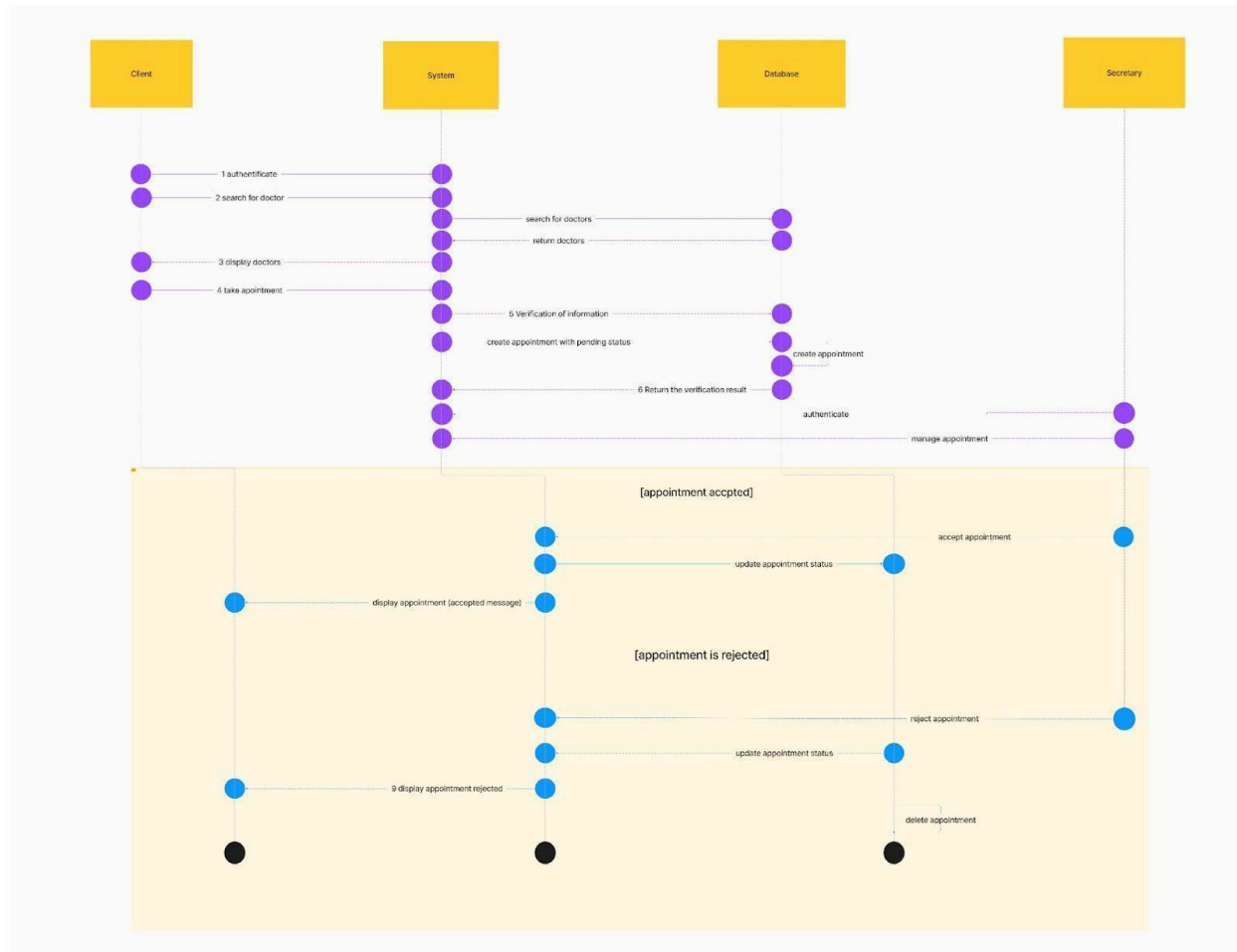


Figure 7: Book Appointment Sequence Diagram

Description

In the sequence diagram below in (Figure 8) the Doctor initiates the process and fills the form. His information passes through two types of verification:

The first, which is mentioned previously, does not require the presence of a human, and an automatic one that handles syntax errors.

The second type is carried out by the administrator, whose goal is to verify the seriousness of the applicant by addressing the reason box, verifying the identity, and investigating identity theft.

In case of acceptance, the payment link will be sent via email and the applicant will receive his account

In the event of rejection, a mail explaining the reason will be sent

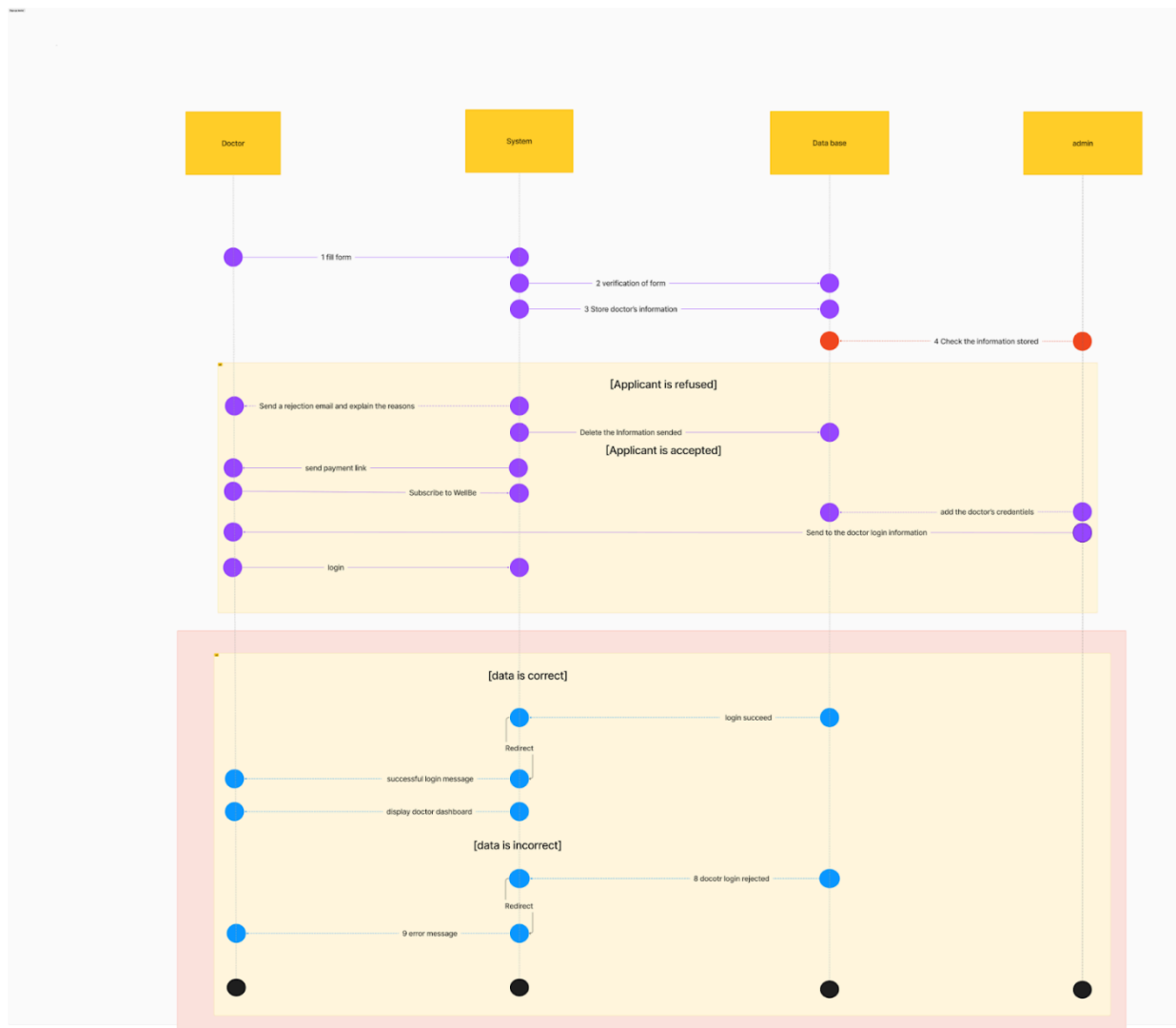


Figure 8: Doctor Signup Sequence Diagram

Description:

In the sequence diagram below in (Figure 9) the user requests a **reset**, enters their email, and the system verifies it. If valid, a one-time code is emailed to the user. They enter the code, and if correct, they create a new password and log in. If wrong, they can try again with a new code until successful.

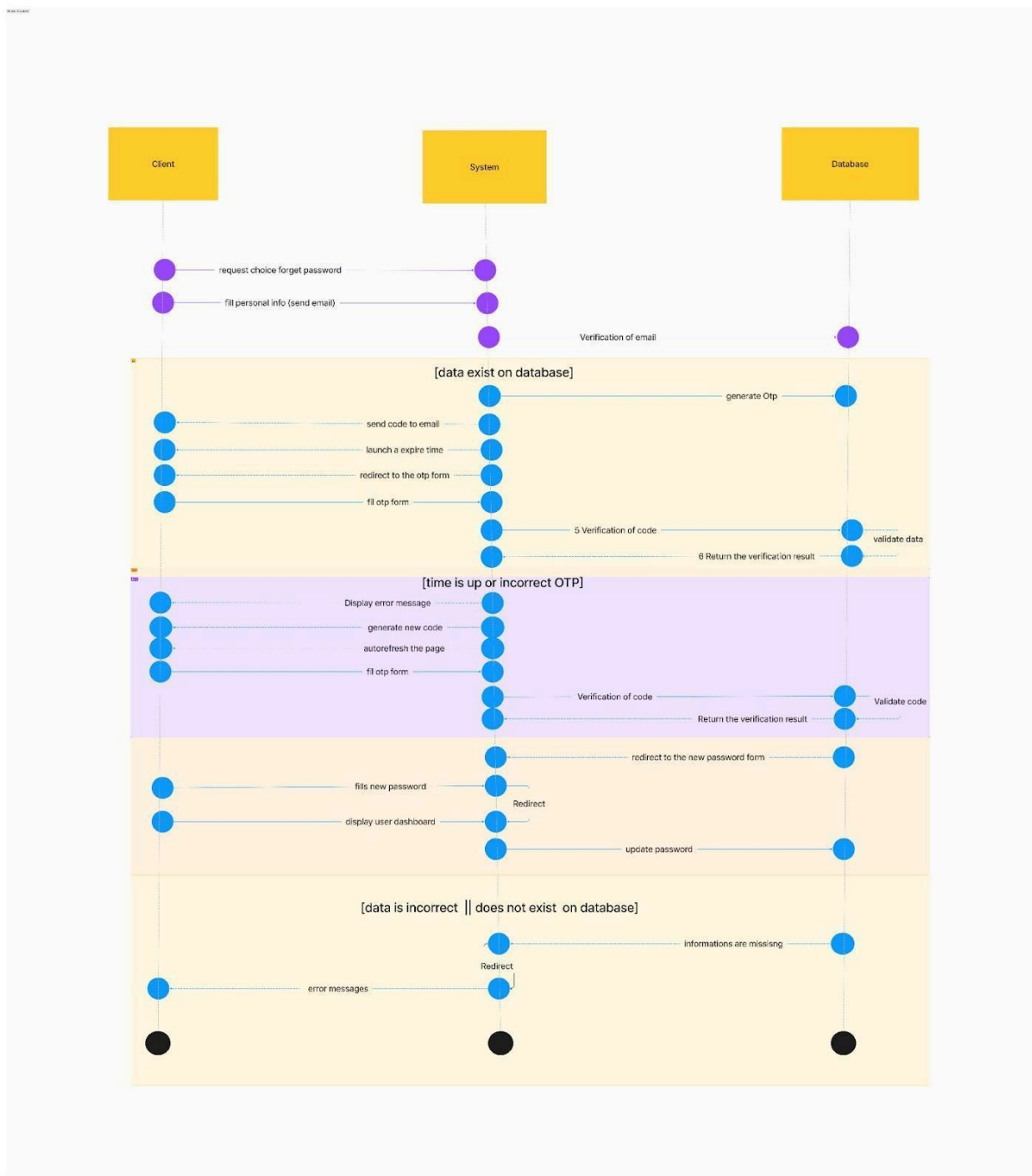


Figure 9: Restore Password Sequence Diagram

Description:

In the sequence diagram below in (Figure 10) the system guides the user through two main options: searching for a cabinet or clinic and rating one. Upon selecting search, the user enters criteria and receives relevant results. Choosing the **rate feature**, the system to verify the user's appointment history with the chosen

facility. If no prior appointment exists, the system rejects the rating and displays an error message. Conversely, a confirmed appointment allows the user to submit a rating and review, which the system then stores for future reference.

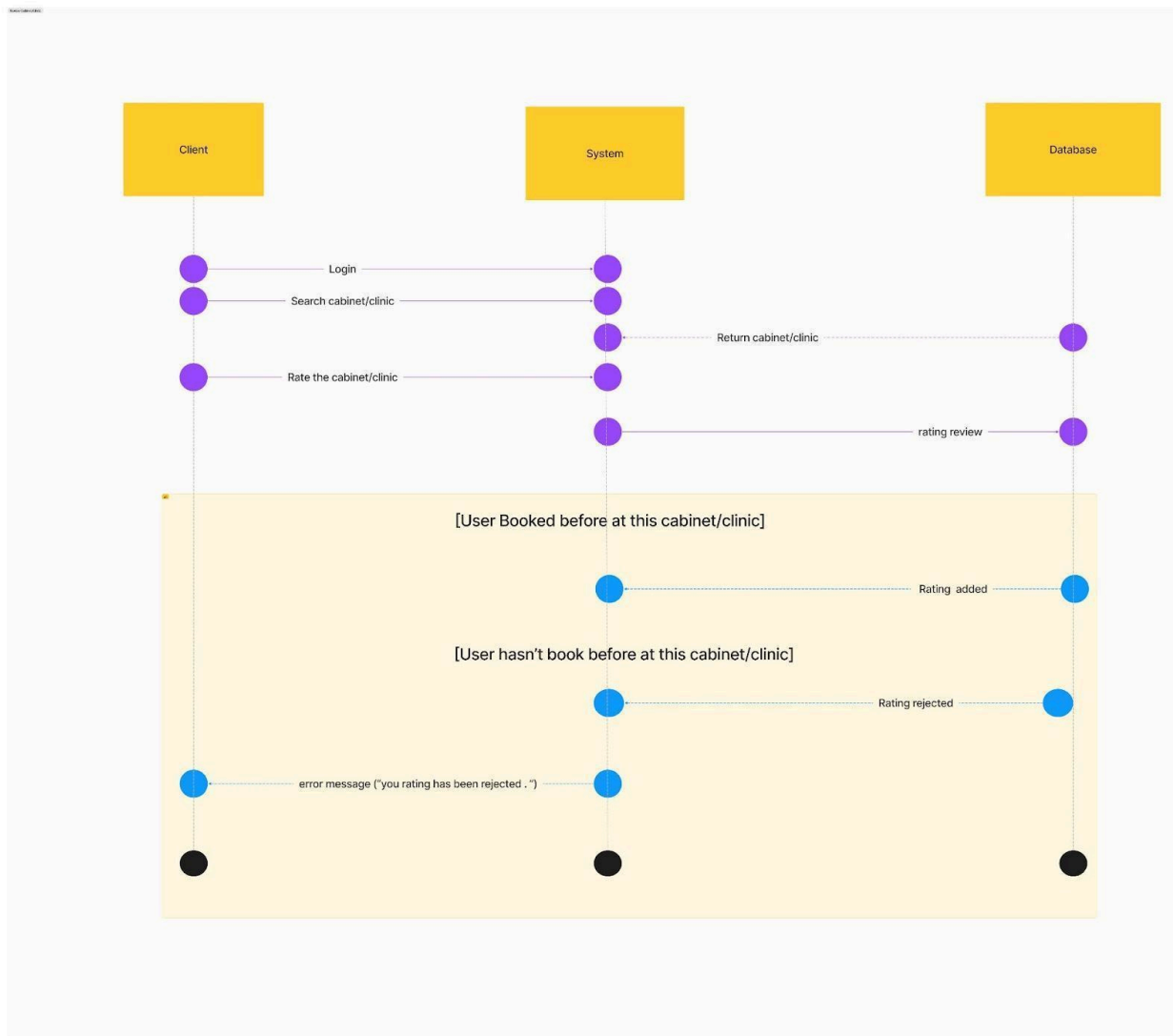


Figure 10: Client Review Sequence Diagram

Description:

In the sequence diagram below in (Figure 11) the doctor initiates the report system, prompting authentication. Upon successful login, choosing "new report" directs them to a writing page for saving. From this point he can select medicines from the medicines existent in database after saving he has the option to print the prescription

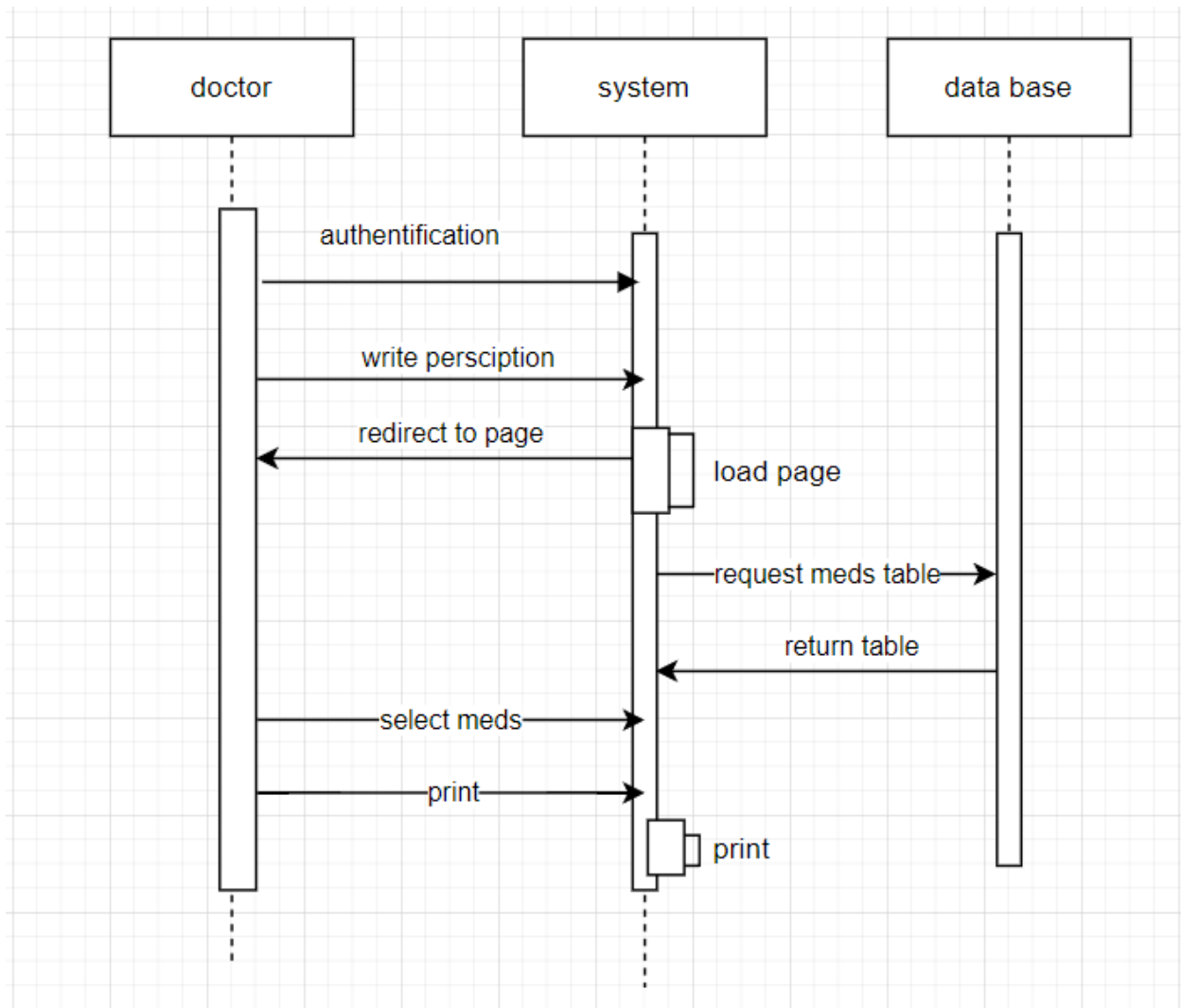


Figure 11: Write Prescription Sequence Diagram

Description:

In the sequence diagram below in (Figure 12) the doctor initiates by sending an authentication message to the system. System responds with a "manage schedule" message, indicating successful login and offering options. System redirects the doctor to a separate interface for schedule manipulation. Doctor makes changes to

the schedule within the system. Upon completion, a "save changes" message is sent to the system

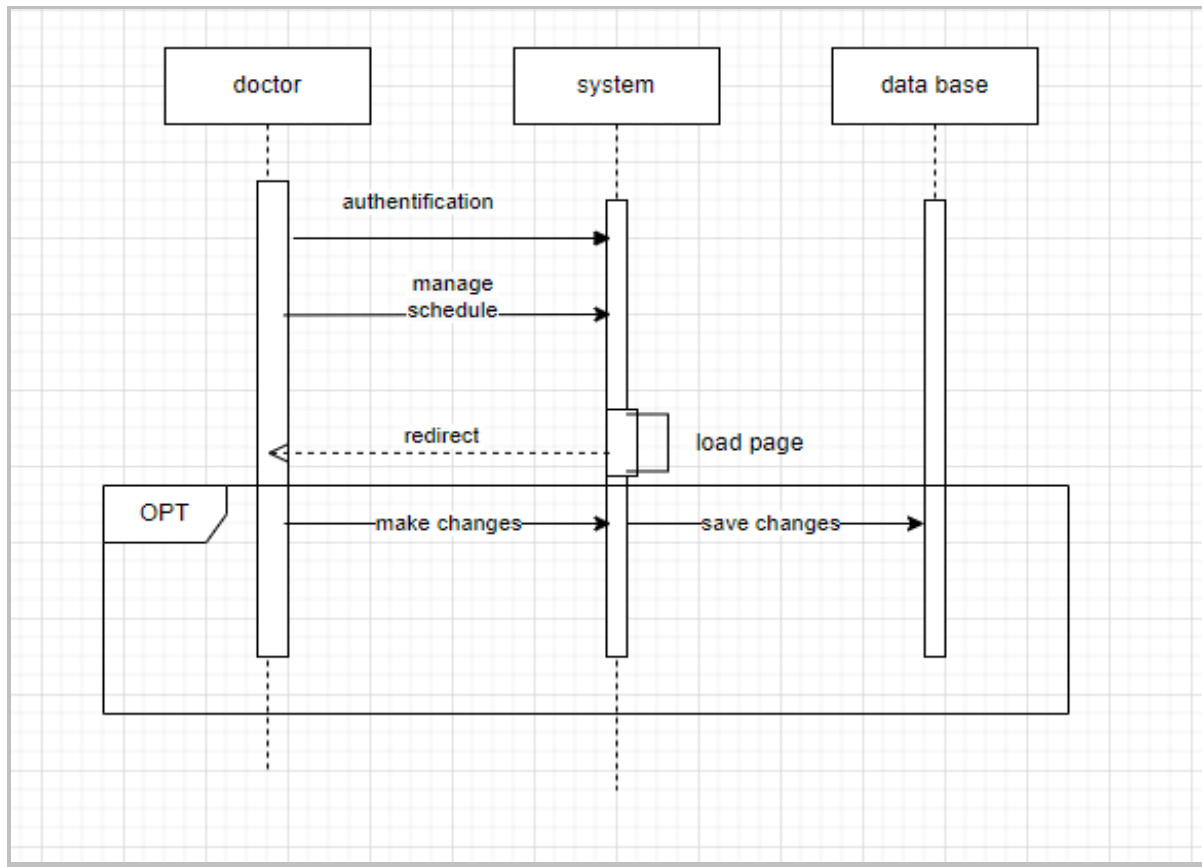


Figure 12: Doctor Management Schedule Sequence Diagram

Description:

The sequence diagram below in (Figure 13) illustrates a process for a user to **check their status** through a doctor system. The user initiates the interaction by sending a request. The doctor system authenticates the request, and then offers two options: view status or load status. Selecting "view status" redirects the user to a separate page for viewing. Choosing "load status" prompts the doctor system to retrieve the status information from the database. Once retrieved, the doctor system displays the status information to the user.

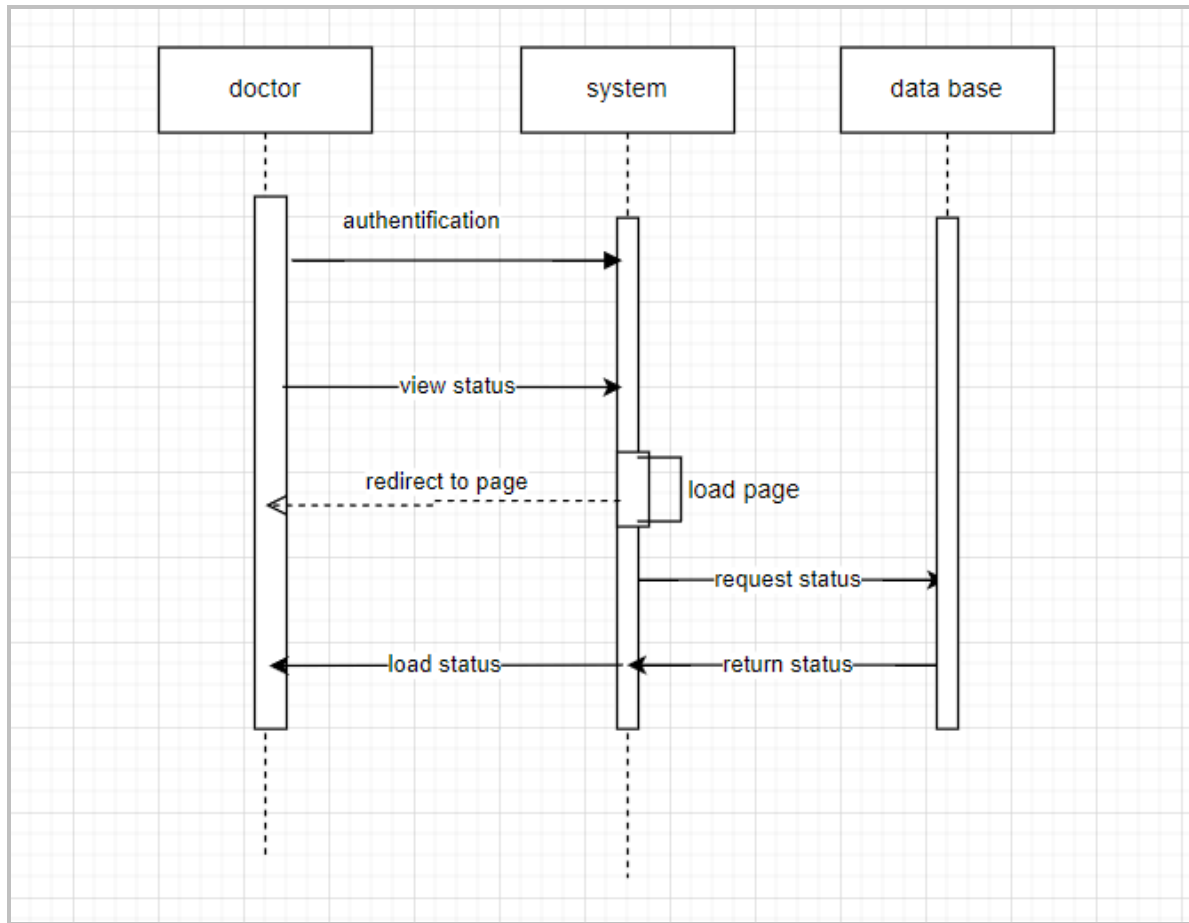


Figure 13: View Status Sequence Diagram

Description:

This sequence diagram in (Figure 14) outlines a doctor's interaction with a report management system. After authentication, the doctor can either **create a new report** or manage existing ones. Choosing "new report" leads to a writing interface. Selecting "existing report" offers options to view or modify past entries, ensuring efficient report creation, access, and updates.

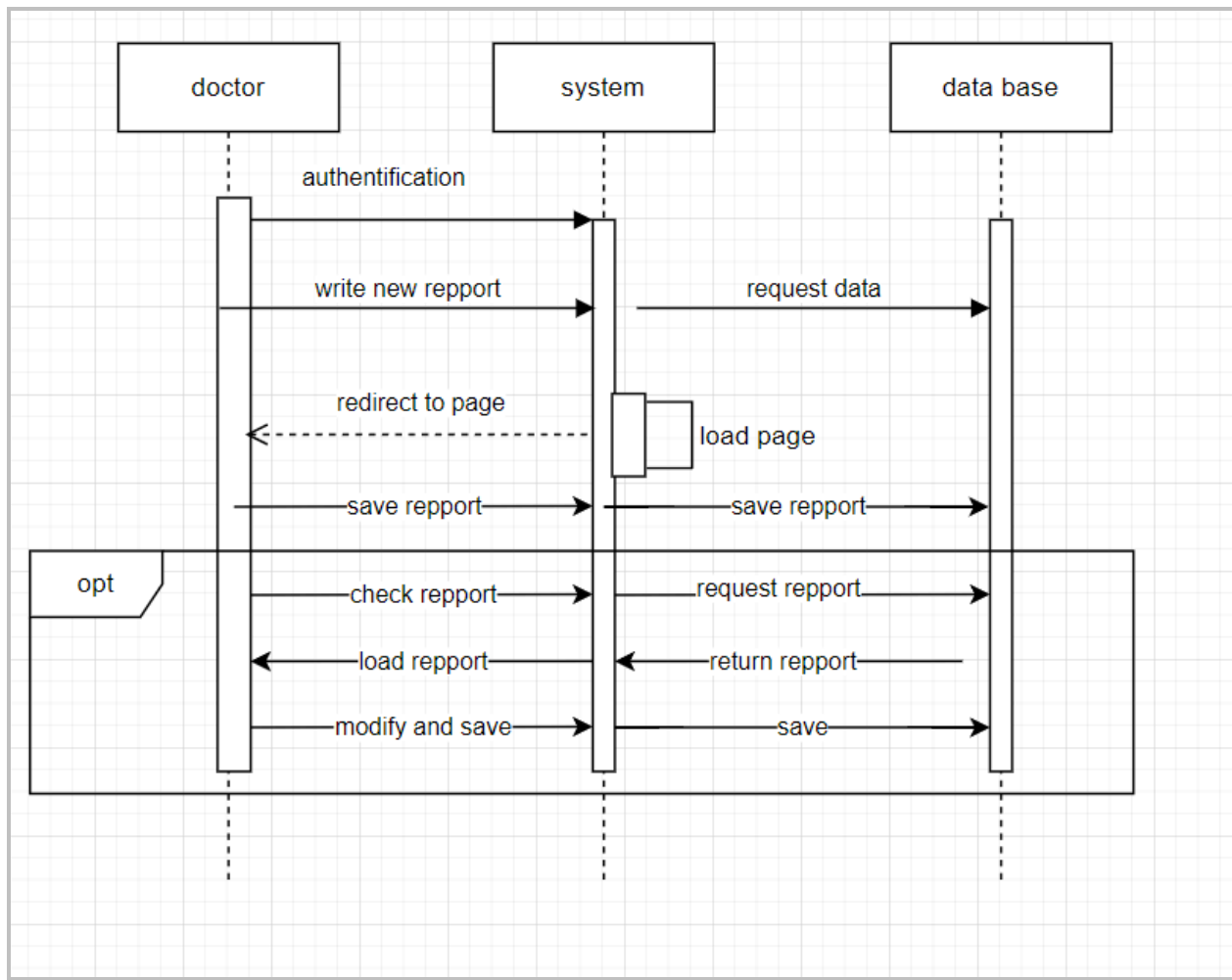


Figure 14: Write Report and Manage it Sequence Diagram

Description:

The sequence diagram in (Figure 15) depicts a simplified appointment **management system**. It starts with a secretary authenticating themselves with the system. Upon successful login, the secretary has two options after checking the new appointments

Either accepts and changes the status and saving it or decline and the appointment request get deleted

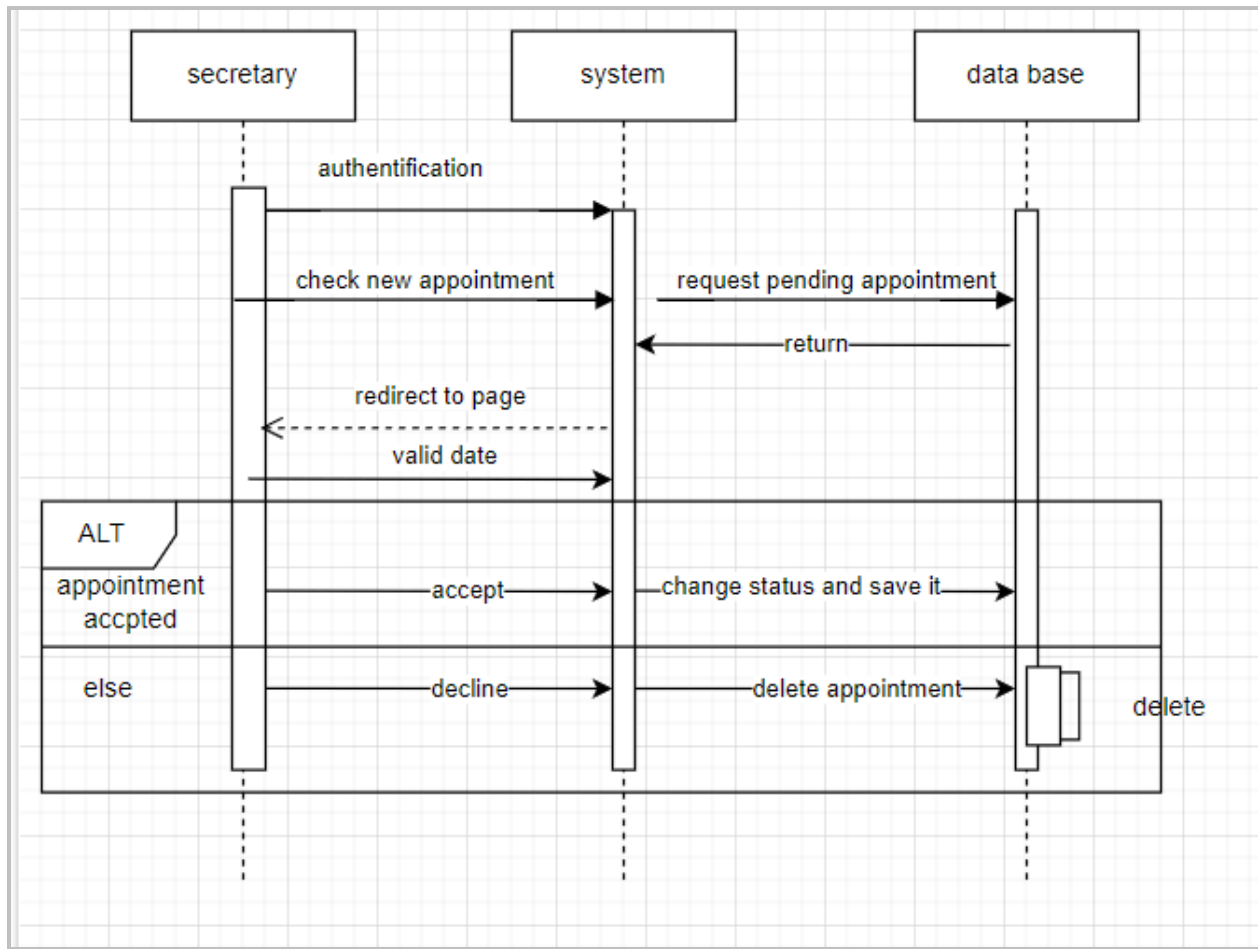


Figure 15: Manage Appointment Sequence Diagram

2.5. Conclusion

In conclusion, we have presented the modelling of both the static and dynamic aspects of our system using a subset of visual aids similar to UML diagrams and applying a unified process for design and modelling. This design is a crucial and necessary step to effectively develop our interactive platform and achieve our goals.

Chapter 3

AI Integration in Healthcare

3.1 Introduction to AI in Doctorino

3.1.1 Overview and Objectives

Artificial Intelligence (AI) has emerged as a transformative force in healthcare, offering unprecedented opportunities to enhance diagnostic accuracy, streamline clinical workflows, and improve patient outcomes. The Doctorino platform integrates AI technologies strategically to address specific challenges faced by healthcare professionals while maintaining the human-centered approach essential to quality care.

The primary objectives of AI integration in Doctorino include:

1. Enhancing diagnostic capabilities through specialized AI models tailored to different medical specialties
2. Reducing the cognitive load on healthcare professionals by automating routine analytical tasks
3. Improving patient engagement through intelligent conversational interfaces
4. Ensuring that AI augments rather than replaces clinical judgment

Doctorino's approach to AI implementation focuses on practical applications that deliver immediate value to both healthcare providers and patients, while establishing a foundation for future innovations as AI technologies continue to evolve.

3.1.2 Benefits for Patients and Doctors

For healthcare professionals, Doctorino's AI integration offers several key benefits:

- Increased efficiency through automated analysis of medical images and signals
- Enhanced diagnostic confidence with AI-powered second opinions
- Specialty-specific tools that align with existing clinical workflows
- Time savings that allow for more meaningful patient interactions
- Continuous learning opportunities through AI-generated insights

Patients benefit from Doctorino's AI features through:

- More accurate and timely diagnosis
- Reduced waiting times for test results
- 24/7 access to basic health information through the Doctor Bot
- Streamlined appointment booking and management
- Personalized health recommendations based on their medical history

By carefully balancing technological innovation with practical clinical needs, Doctorino creates an ecosystem where AI serves as a valuable assistant to healthcare professionals rather than a replacement for human expertise and compassion.

3.2 AI Models Implementation

3.2.1 ECG Analysis for Cardiologists

The ECG Analysis module represents one of Doctorino's core AI implementations, specifically designed to assist cardiologists in interpreting electrocardiogram data. This feature employs a sophisticated machine learning pipeline to analyze ECG signals and identify potential cardiac abnormalities.

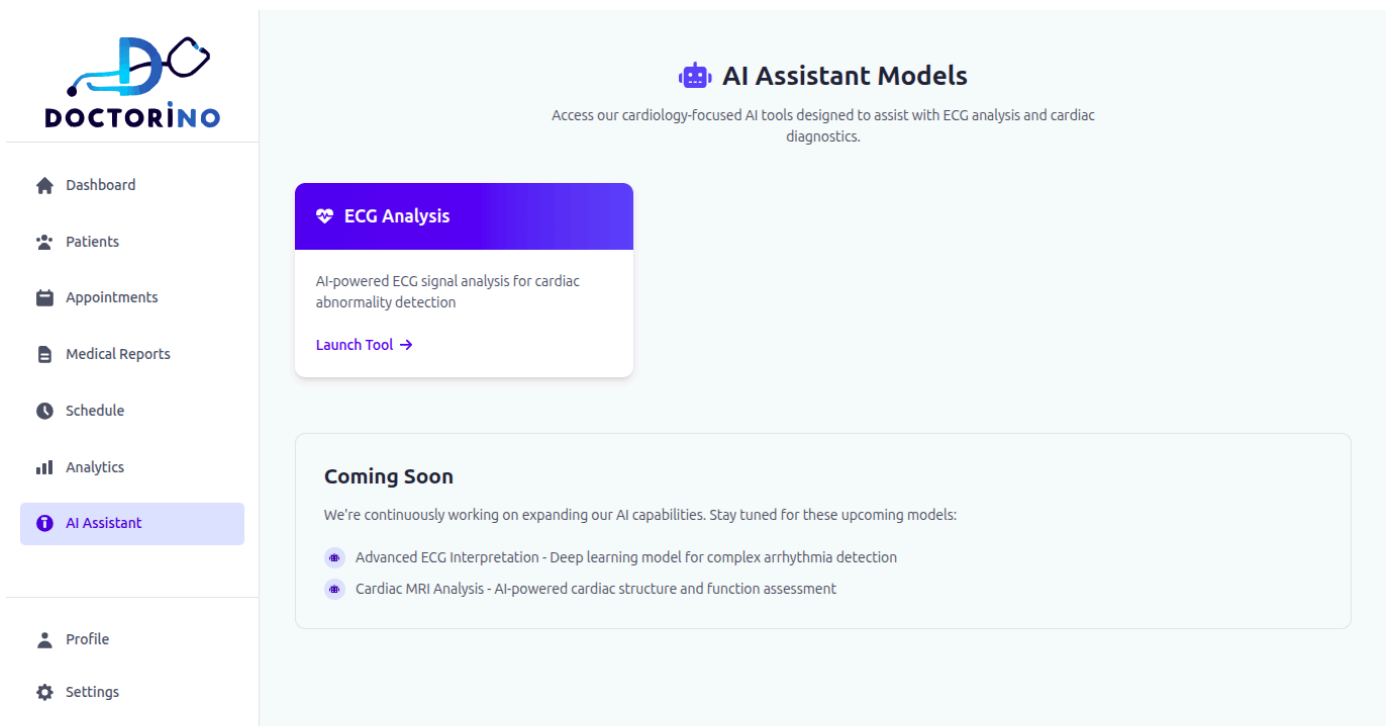
The ECG Analysis system follows a multi-stage process:

1. **Signal Preprocessing:** Raw ECG data undergoes noise reduction using a hierarchical Kalman filter, which effectively removes baseline wander, power line interference, and muscle artifacts while preserving the critical morphological features of the ECG signal.
2. **Segmentation:** The preprocessed signal is divided into segments for detailed analysis, allowing the model to evaluate different portions of the cardiac cycle independently.
3. **Feature Extraction:** Key features are extracted from each segment, including wave amplitudes, intervals, and morphological characteristics that are relevant to cardiac diagnosis.
4. **Classification:** A deep learning model analyzes these features to classify the ECG into different categories of cardiac conditions, providing both segment-level and overall predictions.
5. **Confidence Scoring:** Each prediction is accompanied by a confidence score, helping cardiologists prioritize cases that may require additional review.

The user interface for ECG Analysis is designed with cardiologists' workflow in mind, featuring:

- An intuitive file upload mechanism for ECG data
- Real-time visualization of the original and processed signals
- Clear presentation of analysis results with confidence metrics
- Segment-by-segment breakdown of detected abnormalities
- Visual indicators that highlight areas of potential concern

This implementation significantly reduces the time required for routine ECG interpretation while flagging complex cases for detailed review by specialists.



3.2.2 Mammography Analysis

The Mammography Analysis module employs advanced computer vision techniques to assist radiologists in breast density classification and abnormality detection. This AI-powered tool aims to improve the accuracy and efficiency of mammogram interpretation, particularly in screening contexts where radiologists must review large volumes of images.

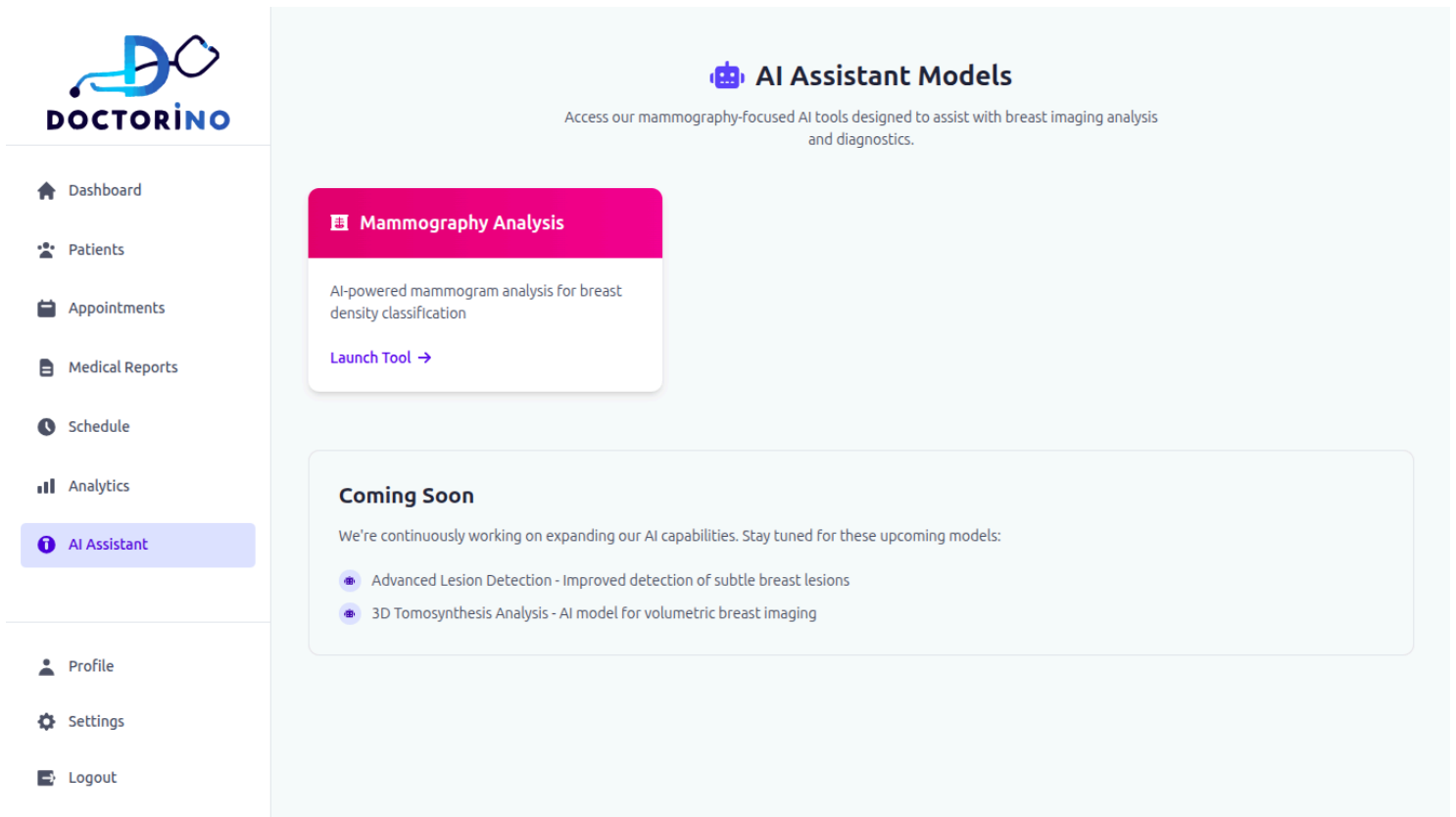
The mammography analysis pipeline consists of:

1. **Image Preprocessing:** DICOM mammography images undergo enhancement through bilateral filtering, contrast-limited adaptive histogram equalization (CLAHE), and wavelet-based denoising to improve visibility of subtle tissue structures.
2. **Breast Density Classification:** The AI model analyzes the processed images to classify breast density according to the BI-RADS density categories (A through D), which is crucial for determining cancer risk and screening protocols.
3. **Region of Interest Detection:** The system identifies and highlights regions that may warrant closer examination by the radiologist.
4. **Confidence Assessment:** Each classification is accompanied by a confidence score, allowing radiologists to focus their attention on cases where the AI is less certain.

The user interface for Mammography Analysis features:

- Support for standard DICOM format mammography images
- Side-by-side display of original and AI-enhanced images
- Clear visualization of the AI's classification results
- Detailed explanation of the reasoning behind the AI's assessment
- Recommendations based on the detected breast density category

This tool is particularly valuable for radiologists specializing in mammography, as it provides a consistent second opinion that can help reduce interpretation variability and potentially catch subtle findings that might otherwise be missed.



3.2.3 Specialty-Based Model Delivery

A key innovation in Doctorino's AI implementation is the specialty-based model delivery system, which ensures that healthcare professionals only see AI tools relevant to their practice area. This approach reduces cognitive overload and increases the likelihood of AI adoption by presenting only contextually appropriate tools.

The implementation includes:

1. **User Specialty Detection:** The system identifies the healthcare professional's specialty through their profile information.

2. **Dynamic Tool Presentation:** Based on the detected specialty, the AI Models dashboard dynamically adjusts to show only relevant tools:
 - Cardiologists see the ECG Analysis tool
 - Radiologists specializing in mammography see the Mammography Analysis tool
 - General practitioners see a broader range of basic diagnostic tools
1. **Specialty-Specific Interfaces:** Each tool's interface is tailored to match the terminology, workflow, and visualization preferences common in the respective specialty.
2. **Future Model Roadmap:** The dashboard also communicates upcoming AI features relevant to each specialty, setting expectations for future capabilities.

This targeted approach ensures that specialists can quickly access the AI tools most valuable to their practice without navigating through irrelevant features, significantly improving the user experience and clinical utility of the platform.

3.3 Doctor Bot Feature

3.3.1 Conversational Interface Design

The Doctor Bot represents Doctorino's patient-facing AI implementation, designed to provide accessible healthcare information and assistance through a natural language conversational interface. The bot employs sophisticated natural language processing (NLP) to understand patient queries and generate helpful, accurate responses.

Key aspects of the conversational interface design include:

1. **Natural Language Understanding:** The system processes patient inputs to identify intents (what the patient wants to accomplish) and entities (specific information like symptoms, dates, or medical terms).
2. **Context Maintenance:** The bot maintains conversation context across multiple exchanges, allowing for natural follow-up questions and clarifications.

3. **Tone and Empathy:** Responses are crafted to convey appropriate empathy and professionalism, acknowledging the emotional aspects of healthcare interactions.
4. **Medical Terminology Management:** The system can recognize and simplify medical terminology when responding to patients while maintaining accuracy.
5. **Escalation Protocols:** Clear indicators for when a human healthcare professional should be involved, with appropriate handoff mechanisms.

The interface is designed to be accessible across multiple platforms, including web and mobile applications, with consistent functionality and appearance to maintain a seamless user experience.

3.3.2 Appointment Booking Assistance

One of the primary functions of the Doctor Bot is to streamline the appointment booking process, reducing administrative burden on clinic staff while improving patient satisfaction. The appointment booking assistant leverages AI to understand patient needs and match them with appropriate healthcare providers and time slots.

The implementation includes:

1. **Need Assessment:** Through conversational exchanges, the bot determines the nature of the patient's medical concern, urgency level, and specialty requirements.
2. **Provider Matching:** Based on the assessment, the system suggests appropriate healthcare providers, considering factors such as specialty, availability, location, and the patient's history.
3. **Schedule Optimization:** The bot accesses real-time availability data to suggest convenient appointment slots, prioritizing urgent cases appropriately.
4. **Confirmation and Reminders:** Once an appointment is booked, the system handles confirmation, adds the appointment to the patient's calendar, and schedules reminders.

5. **Rescheduling Assistance:** The bot can also help patients reschedule or cancel appointments when needed, applying appropriate policies regarding notice periods.

This feature significantly reduces the friction in appointment management, leading to better resource utilization for healthcare providers and improved access to care for patients.

3.3.3 Health Advice Generation

The Doctor Bot includes capabilities for providing basic health information and advice, carefully designed to complement rather than replace professional medical consultation. This feature helps patients access reliable health information while clearly communicating the limitations of AI-generated advice.

The health advice system incorporates:

1. **Knowledge Base Integration:** Access to a curated database of medical information, regularly updated to reflect current best practices and guidelines.
2. **Query Classification:** Algorithms that distinguish between queries that can be safely answered by the bot and those that require professional medical attention.
3. **Personalization:** When appropriate and with proper authorization, the system can reference the patient's medical history to provide more relevant information.
4. **Disclaimer Management:** Clear communication about the limitations of AI-generated advice and appropriate recommendations for when to seek professional care.
5. **Follow-up Prompts:** Suggestions for additional questions or information that might be relevant to the patient's initial query.

All health information provided by the Doctor Bot undergoes regular review by medical professionals to ensure accuracy and appropriateness, with special attention to areas where advice might change based on emerging research or guidelines.

3.4 Technical Architecture

3.4.1 Model Integration Framework

Doctorino employs a modular model integration framework that allows for consistent deployment, monitoring, and updating of AI models across the platform. This architecture facilitates the addition of new models and the improvement of existing ones without disrupting the overall system.

The framework consists of:

1. **Model Registry:** A centralized repository that maintains information about all deployed models, including version history, performance metrics, and deployment status.
2. **API Gateway:** A unified interface for model invocation that handles authentication, request validation, and routing to the appropriate model service.
3. **Model Containers:** Each AI model is deployed in a containerized environment that includes all necessary dependencies, ensuring consistency across development and production environments.
4. **Inference Servers:** Specialized servers optimized for model inference, with support for GPU acceleration where appropriate.
5. **Monitoring System:** Real-time tracking of model performance, usage patterns, and system health to identify issues proactively.

This architecture enables Doctorino to maintain a diverse ecosystem of AI models while ensuring consistent performance, security, and reliability across the platform.

3.4.2 Data Processing Pipeline

The data processing pipeline is a critical component of Doctorino's AI infrastructure, responsible for transforming raw medical data into formats suitable for analysis by AI models. The pipeline is designed to handle various data types, including images, signals, and structured clinical data.

Key components of the data processing pipeline include:

1. **Data Ingestion:** Secure mechanisms for importing data from various sources, including file uploads, device integrations, and electronic health record systems.
2. **Validation and Sanitization:** Automated checks to ensure data quality, completeness, and conformance to expected formats.
3. **Preprocessing:** Specialty-specific preprocessing steps, such as noise reduction for ECG signals or enhancement for mammography images.
4. **Feature Extraction:** Identification and computation of relevant features that serve as inputs to the AI models.
5. **Batch and Stream Processing:** Support for both batch processing of historical data and real-time processing of incoming data.
6. **Result Storage and Retrieval:** Efficient mechanisms for storing analysis results and making them available to healthcare professionals through the user interface.

The pipeline is designed with fault tolerance and scalability in mind, allowing it to handle varying loads while maintaining consistent performance and reliability.

3.5 Security and Privacy

3.5.1 HIPAA Compliance Measures

As a healthcare platform handling sensitive patient information, Doctorino implements comprehensive measures to ensure compliance with the Health Insurance Portability and Accountability Act (HIPAA) and similar regulations worldwide. These measures are particularly important for AI features, which often require access to detailed medical data.

Key HIPAA compliance measures include:

1. **Data Encryption:** All patient data is encrypted both in transit and at rest, using industry-standard encryption protocols.
2. **Access Controls:** Granular permission systems ensure that users can only access data necessary for their role, with special provisions for AI system access.

3. **Audit Logging:** Comprehensive logging of all data access and modifications, including those performed by AI systems during analysis.
4. **Business Associate Agreements:** Proper agreements with all third-party services involved in AI processing or data handling.
5. **Training and Policies:** Regular training for all staff on HIPAA requirements and internal policies governing AI system development and deployment.
6. **Incident Response Plan:** Documented procedures for responding to potential data breaches or security incidents involving AI systems.

These measures are continuously reviewed and updated to reflect changes in regulations and best practices, ensuring ongoing compliance as the platform evolves.

3.5.2 Data Protection Strategies

Beyond regulatory compliance, Doctorino implements additional data protection strategies specifically designed for AI applications in healthcare. These strategies aim to minimize privacy risks while maximizing the utility of AI models.

The data protection approach includes:

1. **Data Minimization:** AI models are designed to operate with the minimum necessary patient data, reducing exposure of sensitive information.
2. **Anonymization and Pseudonymization:** Where appropriate, patient identifiers are removed or replaced before data is processed by AI systems.
3. **Federated Learning:** When possible, model training is performed using federated learning approaches that keep patient data local while allowing models to learn from distributed datasets.
4. **Differential Privacy:** Implementation of differential privacy techniques to add controlled noise to datasets, preventing the identification of individual patients while preserving overall patterns.
5. **Model Isolation:** AI models are deployed in isolated environments with restricted network access, reducing the risk of unauthorized data exfiltration.
6. **Regular Security Assessments:** Ongoing security testing, including penetration testing and vulnerability scanning, with special attention to AI components.

These strategies work together to create a robust data protection framework that safeguards patient privacy while enabling the beneficial use of AI in healthcare applications.

3.6 Evaluation and Future Development

3.6.1 Performance Metrics

Doctorino employs a comprehensive evaluation framework to assess the performance of its AI models, ensuring they meet high standards of accuracy, reliability, and clinical utility. This framework includes both technical metrics and measures of real-world impact.

Key performance metrics include:

1. Technical Accuracy Metrics:

- a. Sensitivity and specificity for diagnostic models
- b. Mean absolute error for regression tasks
- c. F1 score for classification tasks
- d. Area under the ROC curve (AUC) for overall model performance

2. Clinical Utility Measures:

- a. Time saved per case compared to manual analysis
- b. Concordance with expert opinions
- c. Detection of findings missed by human reviewers
- d. False positive and false negative rates in clinical context

3. User Experience Metrics:

- a. User satisfaction scores
- b. Feature adoption rates
- c. Time to proficiency for new users
- d. Frequency of manual overrides of AI recommendations

4. System Performance:

- a. Processing time per case
- b. System availability and reliability
- c. Resource utilization efficiency
- d. Scalability under varying loads

These metrics are continuously monitored and regularly reviewed to identify opportunities for improvement and to ensure that AI features continue to deliver value to healthcare professionals and patients.

3.6.2 Planned AI Enhancements

Doctorino maintains an active development roadmap for AI features, with several enhancements planned for future releases. These developments aim to expand the platform's capabilities while addressing emerging needs in healthcare.

Planned enhancements include:

1. **Expanded Specialty Coverage:**
 - a. Chest X-Ray Analysis for pulmonologists
 - b. Skin Lesion Classification for dermatologists
 - c. Cardiac MRI Analysis for advanced cardiology applications
2. **Advanced ECG Capabilities:**
 - a. Complex arrhythmia detection
 - b. Longitudinal ECG comparison
 - c. Integration with wearable ECG devices
3. **Enhanced Mammography Features:**
 - a. Improved lesion detection
 - b. 3D tomosynthesis analysis
 - c. Risk prediction based on imaging findings
4. **Doctor Bot Improvements:**
 - a. Multilingual support
 - b. Voice interface integration
 - c. Enhanced personalization based on medical history
5. **Cross-Modal Integration:**
 - a. Correlation of findings across different imaging modalities
 - b. Integration of imaging findings with laboratory results
 - c. Comprehensive patient risk assessment

These enhancements will be developed with ongoing input from healthcare professionals to ensure they address real clinical needs and integrate smoothly into existing workflows.

3.7 Conclusion

The integration of AI into the Doctorino platform represents a significant advancement in healthcare technology, offering tools that enhance diagnostic capabilities, improve efficiency, and increase access to care. By focusing on specialty-specific applications with clear clinical utility, Doctorino has created AI features that serve as valuable assistants to healthcare professionals rather than attempting to replace human expertise.

The ECG Analysis and Mammography Analysis modules demonstrate how targeted AI applications can address specific challenges in clinical practice, while the Doctor Bot shows the potential for AI to improve patient engagement and streamline administrative processes. Together, these features form a comprehensive approach to AI integration that spans the healthcare journey from appointment booking to diagnosis and treatment planning.

As AI technologies continue to evolve, Doctorino is well-positioned to incorporate new capabilities while maintaining its commitment to security, privacy, and clinical excellence. The modular technical architecture and robust evaluation framework ensure that future enhancements can be deployed efficiently and monitored effectively.

Ultimately, the success of AI in healthcare depends not just on technical sophistication but on thoughtful implementation that respects the central role of human judgment in medical decision-making. Doctorino's approach to AI integration exemplifies this philosophy, creating a platform where technology and human expertise work together to improve healthcare outcomes for all.

Chapter 4

Project Implementation

4.1 Introduction

This chapter focuses on the implementation of our platform based on the modelling presented in the previous chapter. We will discuss the different tools and languages used during the implementation process, as well as showcase our platform through its various interfaces.

4.2 Implementation Tools

For the implementation of our application, we will utilize the following tools and technologies:

4.2.1 Database Management System

We have chosen MongoDB as our database management system. MongoDB is a popular and reliable NoSQL database management system that uses a flexible, document-oriented data model for managing and manipulating data. It provides a robust solution for storing and retrieving data in our application, especially where scalability and performance with large volumes of unstructured or semi-structured data are important.

4.2.2 Containerization Platform

To ensure a consistent and portable deployment environment, we will utilize Docker. Docker allows us to package our application and its dependencies into lightweight, isolated containers. This enables easier deployment and scalability across different systems⁸.

⁸ <https://www.docker.com/> (Accessed on May 25th, 2024)

4.2.3 In-Memory Data Store

For caching and real-time functionalities, we will incorporate Redis. Redis is an in-memory data structure store that can be used as a database, cache, or message broker. It provides high-performance data storage and retrieval capabilities, making it ideal for optimizing our application's performance⁹.

4.2.4 Real-Time Communication

To enable real-time bidirectional communication between the server and clients, we will integrate “Socket.IO”. Socket.IO is a library that facilitates real-time, event-based communication using web sockets. It provides a reliable and scalable solution for building real-time features in our application¹⁰.

4.2.5 Object-Document Mapping (ODM)

For database management and seamless interaction with our MongoDB database, we will use an ODM. An ODM (Object-Document Mapping) allows us to work with MongoDB collections using object-oriented paradigms, eliminating the need to write raw queries. This simplifies database operations and enhances code maintainability.

MongoDB Compass provides a graphical interface to visualize and manage our collections and documents, making it easier to inspect data, run queries, and understand database schema. For programmatic access, we utilize ODM libraries like Mongoose (for Node.js) or MongoEngine (for Python), which map application-level objects to MongoDB documents.

⁹ <https://redis.io/> (Accessed on May 25th, 2024)

¹⁰ <https://socket.io/> (Accessed on May 25th, 2024)

4.2.6 User Interface Design

For efficient and consistent UI development across platforms, we employ modern UI frameworks tailored to each application's needs:

Doctor Web Application: We use React for building a dynamic, component-based user interface. Combined with Tailwind CSS, a utility-first CSS framework, we can rapidly develop responsive and visually consistent designs using pre-defined utility classes. This approach allows for fast prototyping and high customization without writing extensive custom CSS.

Doctorino Mobile Application: We use Flutter, a cross-platform UI toolkit developed by Google, to build native-like mobile interfaces for both Android and iOS. Flutter's widget-based architecture enables high-performance rendering and a unified codebase, which streamlines development and maintenance.

This combination of technologies ensures a modern, responsive, and user-friendly experience across web and mobile platforms.

4.3. Presentation of the platform

This section provides an in-depth exploration of the platform, offering insights into its functionalities and features. By examining the core elements of the platform, including user interfaces and system **modules**, this section aims to provide a comprehensive understanding of how the platform Doctorino functions.

4.3.1. Landing page

The interface shown in (Figure 16) illustrates the initial entry point for users who are new to the platform. It typically provides a clear overview of the platform's purpose and benefits. It includes calls to action like sign up, log in, doctor subscription, FAQ and contact us.

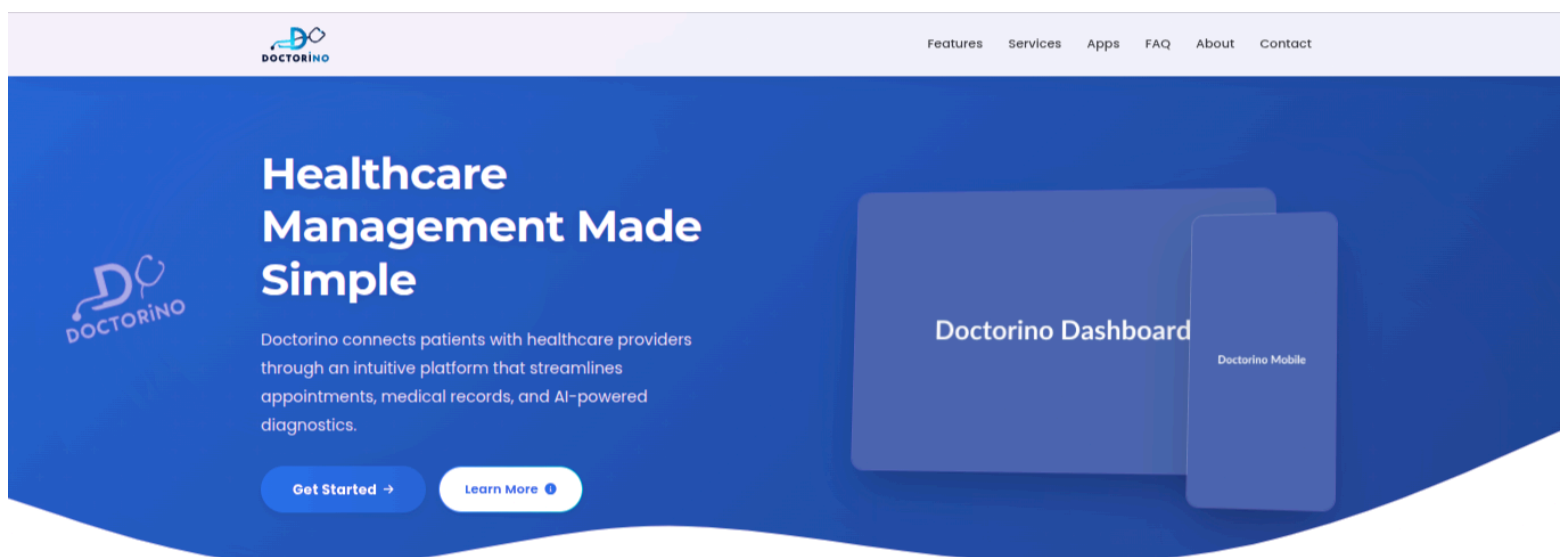


Figure 16: Landing page interface

Key Features

Discover how Doctorino can transform your healthcare experience



Appointment Management

Schedule, reschedule, or cancel appointments with ease. Get reminders and manage your healthcare calendar in one place.

[Learn more >](#)



Doctor Search

Find the right healthcare provider based on specialty, location, and availability. View profiles and patient reviews.

[Learn more >](#)



Medical Records

Access your complete medical history, test results, and prescriptions anytime, anywhere. Share information securely with your providers.

[Learn more >](#)



AI-Powered Diagnostics

Benefit from cutting-edge AI analysis for ECG, mammography, and other medical imaging to support clinical decisions.

[Learn more >](#)

Figure 17: Key Features

Our Services

Comprehensive healthcare solutions for patients and providers



Telemedicine Consultations

Connect with healthcare providers remotely through secure video consultations. Get medical advice, prescriptions, and follow-ups from the comfort of your home.



Remote Patient Monitoring

Track vital signs and health metrics with connected devices. Our platform automatically alerts healthcare providers about concerning changes in your condition.



Medication Management

Receive reminders for medication schedules, track your prescriptions, and get automatic refill notifications to ensure you never miss a dose.



Healthcare Analytics

Gain insights into health trends with comprehensive analytics. Our platform helps providers make data-driven decisions for better patient outcomes.



Patient Engagement

Stay connected with your healthcare journey through personalized education materials, progress tracking, and direct communication with your care team.



Secure Health Data

Your health information is protected with enterprise-grade security and encryption. Control who can access your data with granular permission settings.

Figure 18: services interface

Our Applications

Tailored solutions for patients and healthcare providers

Doctor Dashboard

Doctorino for Healthcare Providers

A comprehensive desktop application that helps doctors manage patient appointments, medical records, and leverage AI-powered diagnostic tools.

- Patient management dashboard
- Appointment scheduling system
- AI-powered ECG and mammography analysis
- Medical records management

[Download Desktop App](#)

Doctorino for Patients

A user-friendly mobile application that empowers patients to take control of their healthcare journey.

- Find and book appointments
- Access medical reports and history
- Receive appointment reminders
- Secure messaging with healthcare providers

[Google Play](#)[App Store](#)

Patient App

Figure 19: Our Applications

Frequently Asked Questions

Find answers to common questions about our platform

How secure is my medical data on Doctorino?

Doctorino employs enterprise-grade security measures including end-to-end encryption, secure data centers, and strict access controls. We are fully HIPAA compliant and regularly undergo security audits to ensure your medical information remains private and protected.

Can I use Doctorino with my existing healthcare provider?

How do telemedicine consultations work?

Is Doctorino available on all devices?

How does the AI-powered diagnostic system work?

What does Doctorino cost?

Figure 20 : Frequently Asked Questions

Get in touch

Pose your question , we'll be there

Name

Email

Subject

Description

Submit

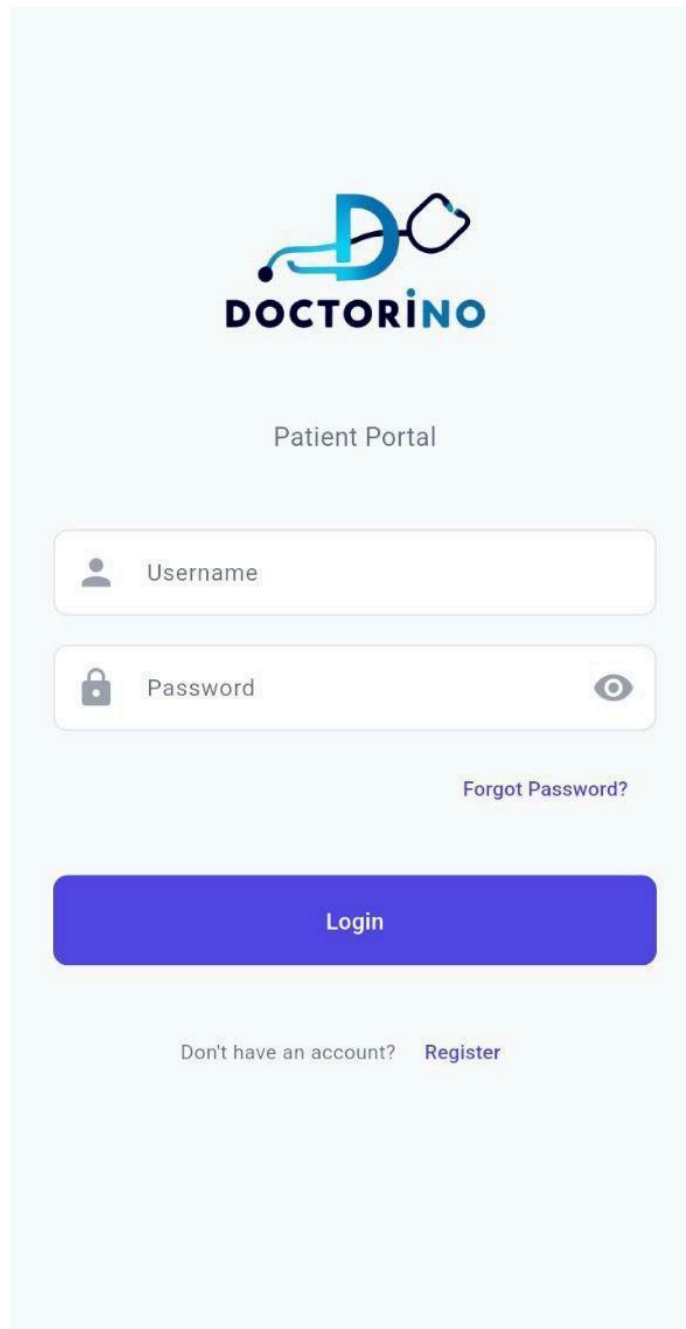


Figure 21: contact us interface

4.3.2. The patients interfaces (mobile)

4.3.2.1 Authentication

The page shown in (Figure 21) allows existing users to access the platform using their username/email and password and then he gets redirected to the dashboard page if credentials have a match. If the user faces troubles signing in, with choosing the forget password option he will be redirected to the forget password page.



The image shows a mobile application login screen for 'DOCTORINO'. At the top center is the logo, which features a stylized blue 'D' with a stethoscope around it, and the word 'DOCTORINO' in bold blue capital letters below it. Underneath the logo is the text 'Patient Portal'. Below this, there are two input fields: the first is labeled 'Username' with a person icon on the left, and the second is labeled 'Password' with a lock icon on the left and an eye icon on the right to toggle visibility. To the right of the password field is a link that says 'Forgot Password?'. Below these fields is a large blue button with the text 'Login' in white. At the bottom, there is a link that says 'Don't have an account? Register'.

Figure 21: Authentication interface

4.3.2.2. Sign-up

The page shown in (Figure 22) enables new users to register for the platform by creating an account with relevant information.

Register



Doctorino

Create your patient account

First Name

Last Name

Email

Phone

Male▼

Password



Confirm Password

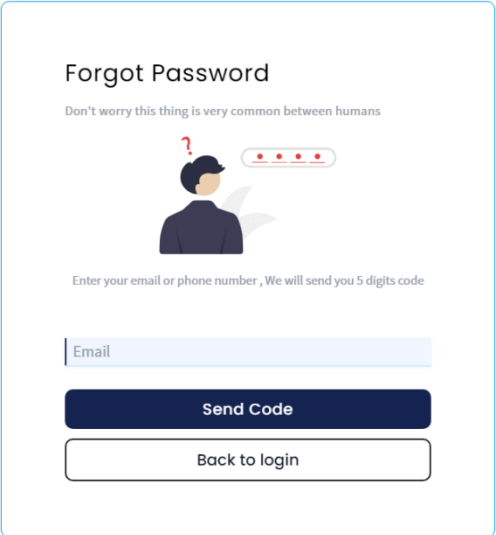


Register

Figure 22: sign-up interface

4.3.2.3. Forget Password

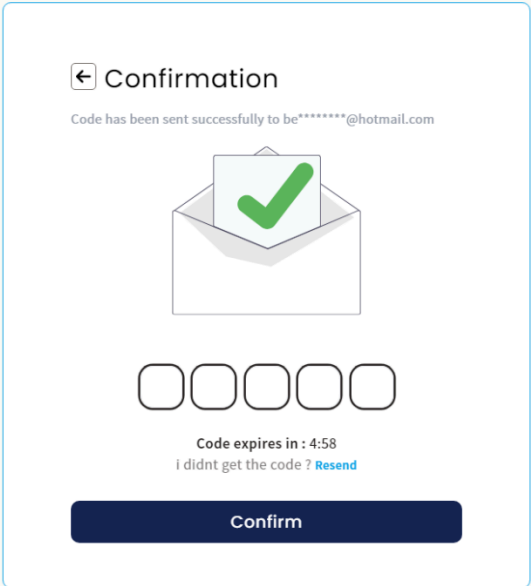
The interface shown in (Figure 23) allows patients to recover their account credentials in case they forget their password. Patients can typically enter their email address to receive a password reset link.



The interface is titled "Forgot Password" and includes a sub-header "Don't worry this thing is very common between humans". It features an illustration of a person with a question mark above their head and a speech bubble containing four red dots. Below the illustration, it says "Enter your email or phone number , We will send you 5 digits code". There is a text input field labeled "Email", a dark blue button labeled "Send Code", and a white button labeled "Back to login".

Figure 23: Forget password interface

Along with a confirm code interface (Figure 24) with a timer for code expiry and a resend button



The interface is titled "Confirmation" with a back arrow icon. It states "Code has been sent successfully to be*****@hotmail.com". Below this is an illustration of an open envelope with a green checkmark inside. There are five empty square boxes for entering the code. Below the boxes, it says "Code expires in : 4:58" and "i didnt get the code ? [Resend](#)". At the bottom is a dark blue button labeled "Confirm".

Figure 24: Confirm code interface

4.3.2.4 Reset password interface

The interface shown in (Figure 25) allows patients to set a new password for their account after receiving a password reset link.

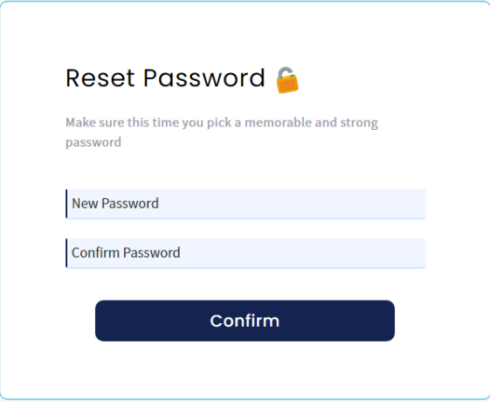
The image shows a 'Reset Password' interface within a light yellow background. The interface is a white rounded rectangle with a thin blue border. At the top, it says 'Reset Password' followed by a small orange padlock icon. Below this, a message reads: 'Make sure this time you pick a memorable and strong password'. There are two light blue input fields: the first is labeled 'New Password' and the second is labeled 'Confirm Password'. At the bottom of the form is a dark blue button with the word 'Confirm' in white text.

Figure 25: Reset password Interface

4.3.2.5 Search for doctors

The interface shown in (Figure 27) allows patients to find doctors based on various criteria such as specialization, location, patient reviews, and insurance acceptance. The search results can be displayed on a map for easy visualization.



Discover healthcare professionals



Find Doctors



Search by doctor name

Specialty

All

Cardiology

Dermatology

Endocrinology

D

Dr. ahmed azzi

Cardiologist

📍 15 wafi madani



D

Dr. soumia rokia

Mammography

📍 mazghran



Doctors



Figure 26: Search for doctors

4.3.2.6 View Doctor profile

The interface in (Figure 28) displays detailed information about the doctor, including his schedule, patient reviews, and areas of specialization as well as some calls to action like appointment booking and adding a review.

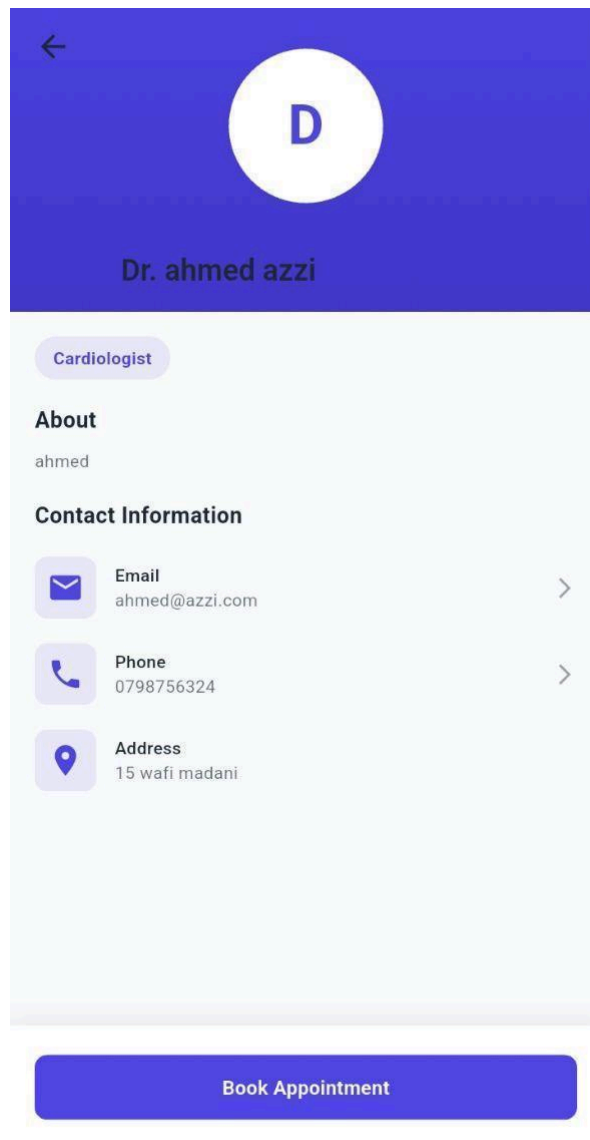


Figure 28: View doctor profile interface

4.3.2.7 Appointment booking

The interface shown in (Figure 29) guides patients through the process of scheduling an appointment with a chosen doctor. Patients need to select a date and time slot that is free and not booked by another patient.

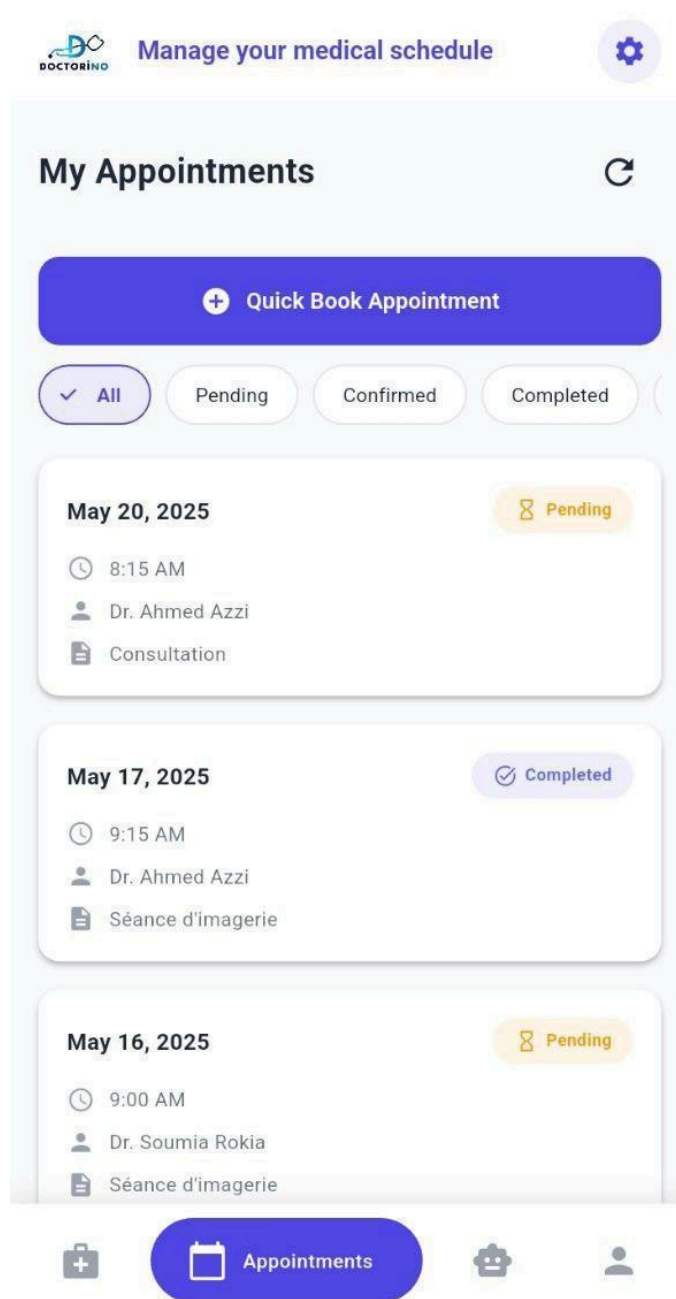


Figure 29: Appointment booking interface

4.3.2.8 View appointment details

The interface shown in (Figure 32) serves as an information hub for patients regarding a specific scheduled or past appointment with a doctor on our platform. It empowers patients to access crucial details, book an appointment to the same doctor without needing to go to the search page, and potentially explore alternative doctor suggestions.

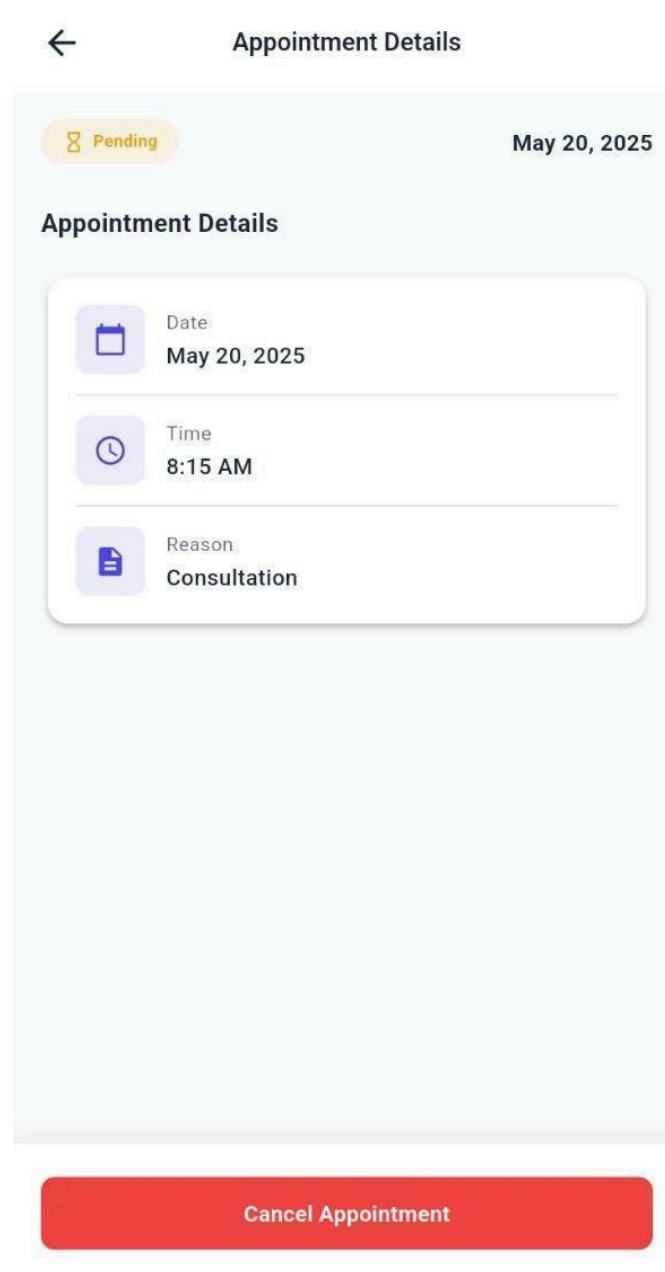


Figure 32: View appointment details interface

4.3.2.9 Manage personal account

The interface shown in (Figure 33) empowers patients to take control of their account details and maintain privacy. This page functions as a one-stop shop for patients to see their contact information and modify their password for enhanced security.

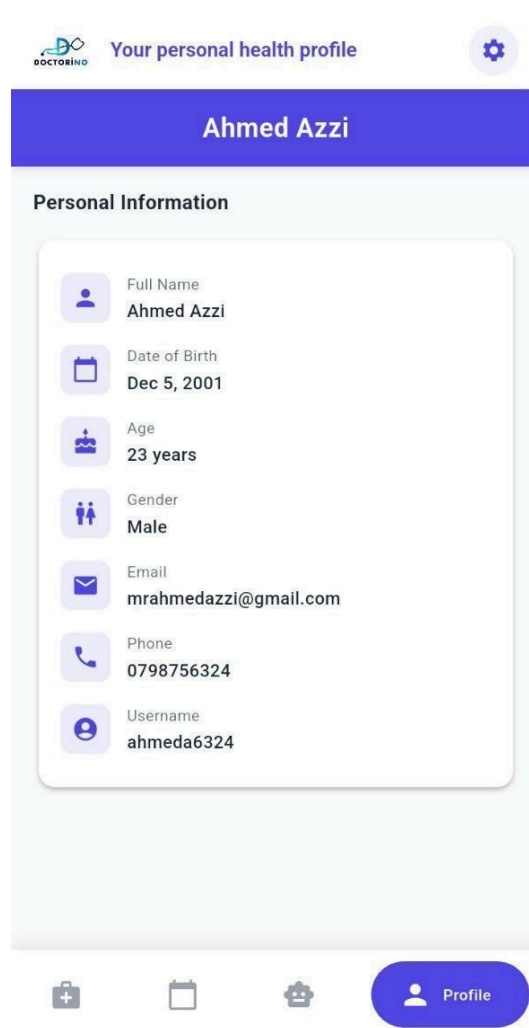


Figure 33: Manage personal account interface

4.3.2.10 The Bot Interface

The interface shown in (Figure 34) offers a user-friendly virtual health assistant that greets users with a clear welcome message and provides intuitive options for interaction. Users can easily navigate through key services such as booking an appointment, finding a doctor, or seeking health advice. The clean layout, use of conversational language, and prominent action buttons ensure that users can quickly access the support they need, enhancing their overall digital healthcare experience.

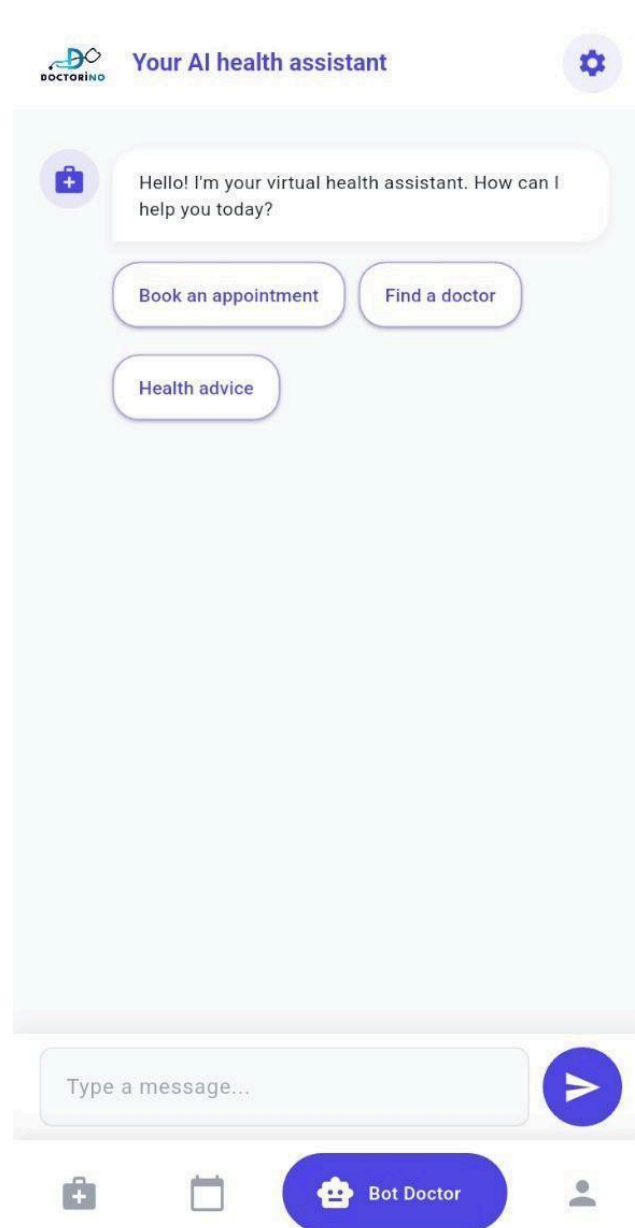
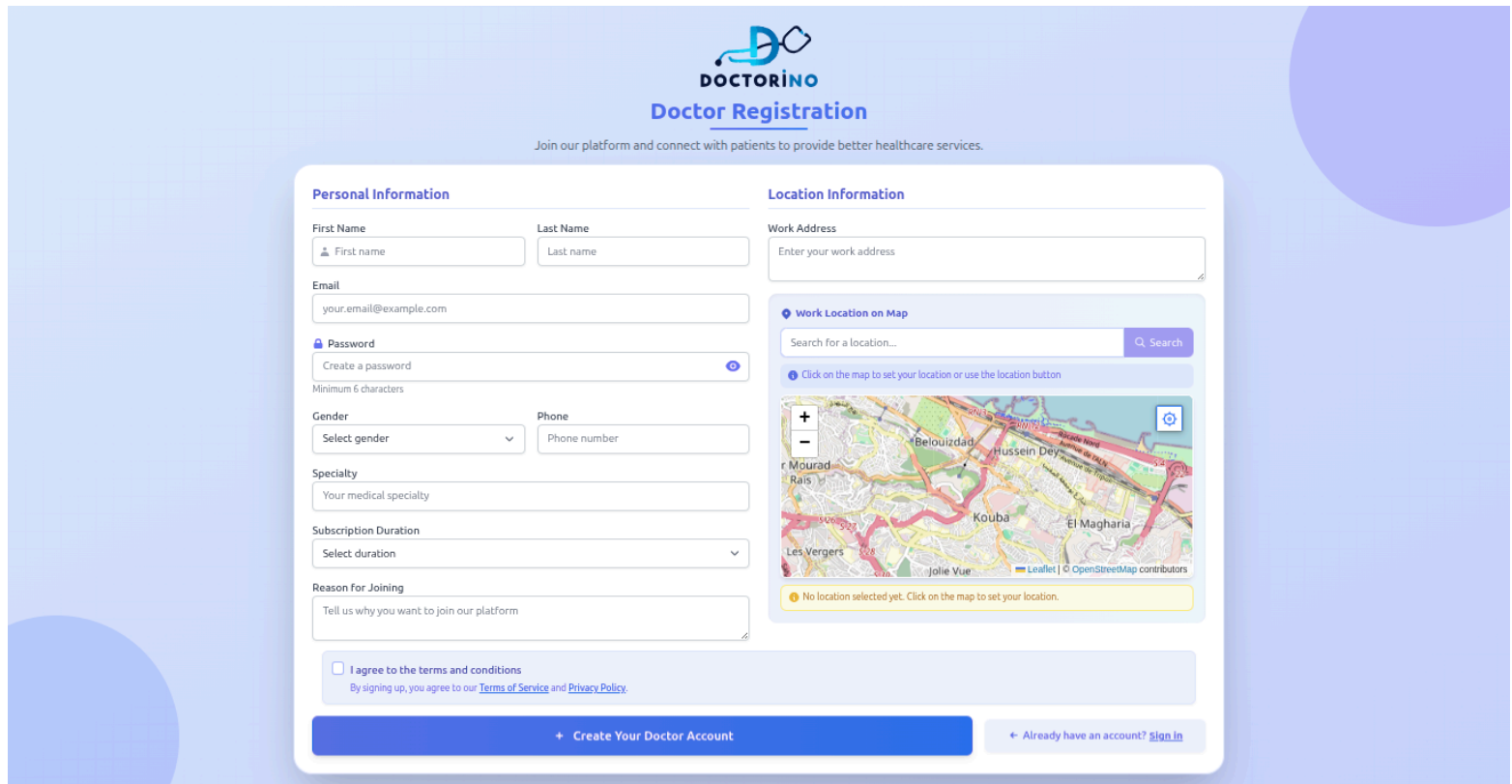


Figure 34: The virtual waiting room interface

4.3.3. The doctor's interfaces (Desktop app)

4.3.3.1 Registration

The interface shown in (Figure 35) enables new doctors to register for the platform. It typically requires information like name, specialization, location, and verification of credentials.



The screenshot displays the 'Doctorino Doctor Registration' form. At the top, the 'DOCTORINO' logo is centered, followed by the title 'Doctor Registration' and a tagline: 'Join our platform and connect with patients to provide better healthcare services.' The form is divided into two main columns: 'Personal Information' and 'Location Information'.

Personal Information:

- First Name:** A text input field with a placeholder 'First name'.
- Last Name:** A text input field with a placeholder 'Last name'.
- Email:** A text input field with a placeholder 'your.email@example.com'.
- Password:** A text input field with a placeholder 'Create a password' and a strength indicator icon. Below it, a note states 'Minimum 6 characters'.
- Gender:** A dropdown menu with 'Select gender' as the current selection.
- Phone:** A text input field with a placeholder 'Phone number'.
- Specialty:** A text input field with a placeholder 'Your medical specialty'.
- Subscription Duration:** A dropdown menu with 'Select duration' as the current selection.
- Reason for Joining:** A text area with a placeholder 'Tell us why you want to join our platform'.

Location Information:

- Work Address:** A text input field with a placeholder 'Enter your work address'.
- Work Location on Map:** A section containing a search bar 'Search for a location...' with a 'Search' button. Below the search bar, a note says 'Click on the map to set your location or use the location button'. A map of Algiers is displayed, showing various districts like Belouizdad, Hussein Dey, Kouba, El Magharia, Les Vergers, and Jolie Vue. A yellow banner at the bottom of the map area states 'No location selected yet. Click on the map to set your location.'.

At the bottom of the form, there is a checkbox for 'I agree to the terms and conditions' with a link to 'Terms of Service and Privacy Policy'. Below this, a large blue button reads '+ Create Your Doctor Account'. To the right of this button, a link says '← Already have an account? [sign in](#)'.

Figure 35: Doctor registration interface

4.3.3.2 Doctor dashboard

The interface shown in (Figure 36) serves as the central hub for doctors to manage their activity on the platform. It provides a comprehensive overview of appointments, patient interactions, and average avenue.

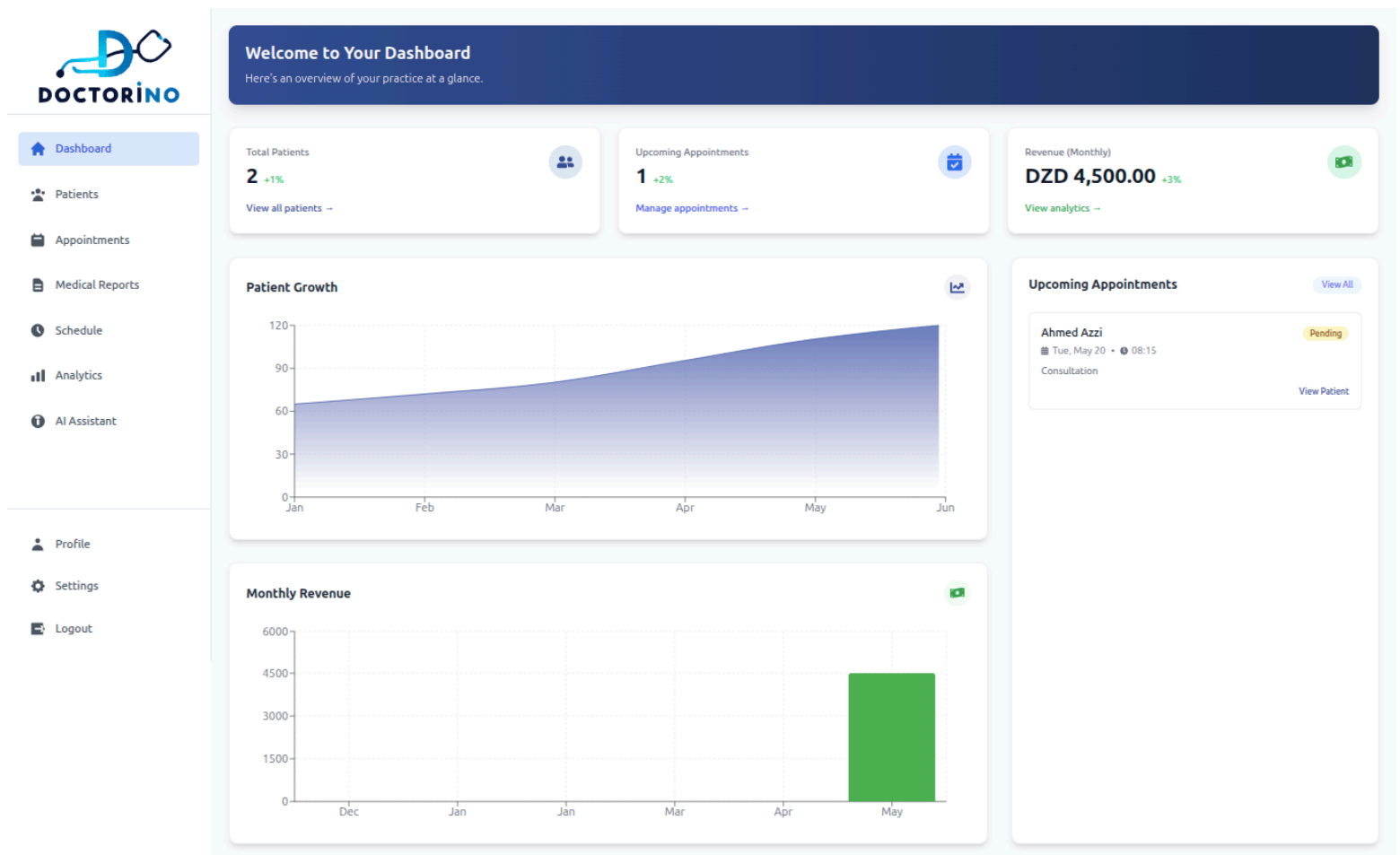
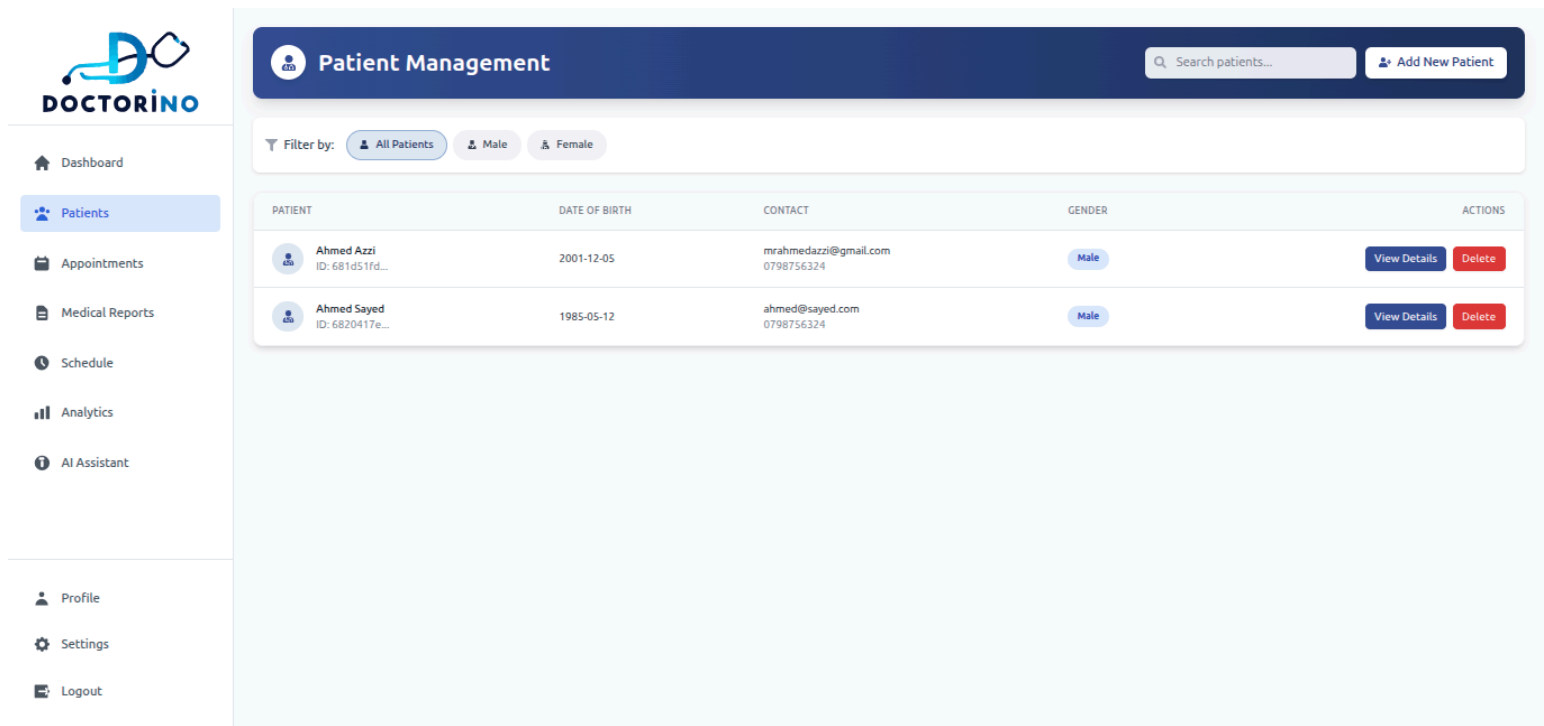


Figure 36: Doctor's dashboard interface

4.3.3.3 View patients list

The interface shown in (Figure 37) serves as a doctor's dashboard for managing their past patients' information on the Doctorino platform. It provides a list of

patients previously seen by the doctor and functionalities for filtering and viewing detailed patient records.

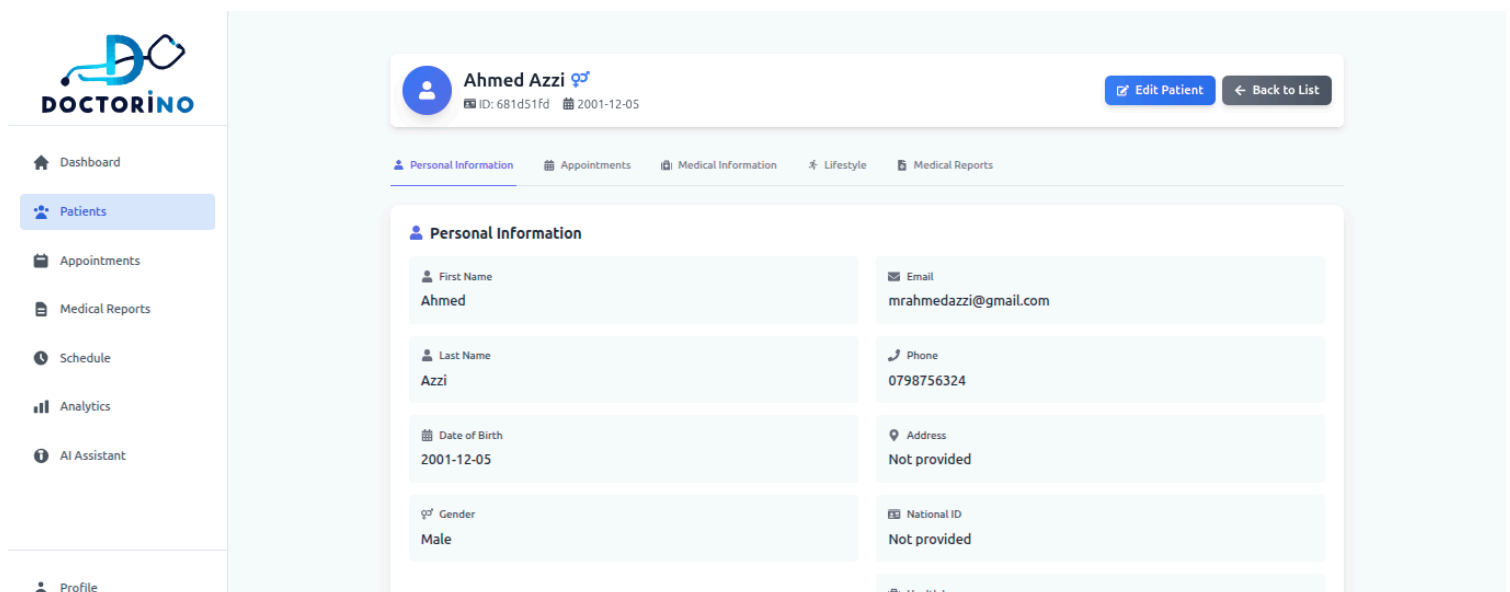


The screenshot displays the 'Patient Management' interface. On the left is a sidebar with navigation options: Dashboard, Patients (highlighted), Appointments, Medical Reports, Schedule, Analytics, AI Assistant, Profile, Settings, and Logout. The main content area has a dark blue header with the 'Patient Management' title, a search bar, and an 'Add New Patient' button. Below the header is a filter section with 'Filter by:' and buttons for 'All Patients', 'Male', and 'Female'. The main area contains a table with two patient entries.

PATIENT	DATE OF BIRTH	CONTACT	GENDER	ACTIONS
 Ahmed Azzi ID: 681d51fd...	2001-12-05	mrahmedazzi@gmail.com 0798756324	Male	View Details Delete
 Ahmed Sayed ID: 6820417e...	1985-05-12	ahmed@sayed.com 0798756324	Male	View Details Delete

Figure 37: View patients list interface

By pressing view more, the doctor may view the patient details page as in (Figure 38)



The screenshot shows the 'Patient Details' page for 'Ahmed Azzi'. The sidebar is identical to Figure 37. The main content area has a header with the patient's name, ID, date of birth, and buttons for 'Edit Patient' and 'Back to List'. Below the header is a tabbed interface with tabs for 'Personal Information', 'Appointments', 'Medical Information', 'Lifestyle', and 'Medical Reports'. The 'Personal Information' tab is active, showing a form with the following fields:

Personal Information	
 First Name Ahmed	 Email mrahmedazzi@gmail.com
 Last Name Azzi	 Phone 0798756324
 Date of Birth 2001-12-05	 Address Not provided
 Gender Male	 National ID Not provided

Figure 38: View patient details interface

4.3.3.4 View medical reports

The interface shown in (Figure 39) empowers doctors to efficiently access and review a patient's medical history on the platform. It provides a comprehensive overview of the patient's health information, facilitating informed clinical decisions.

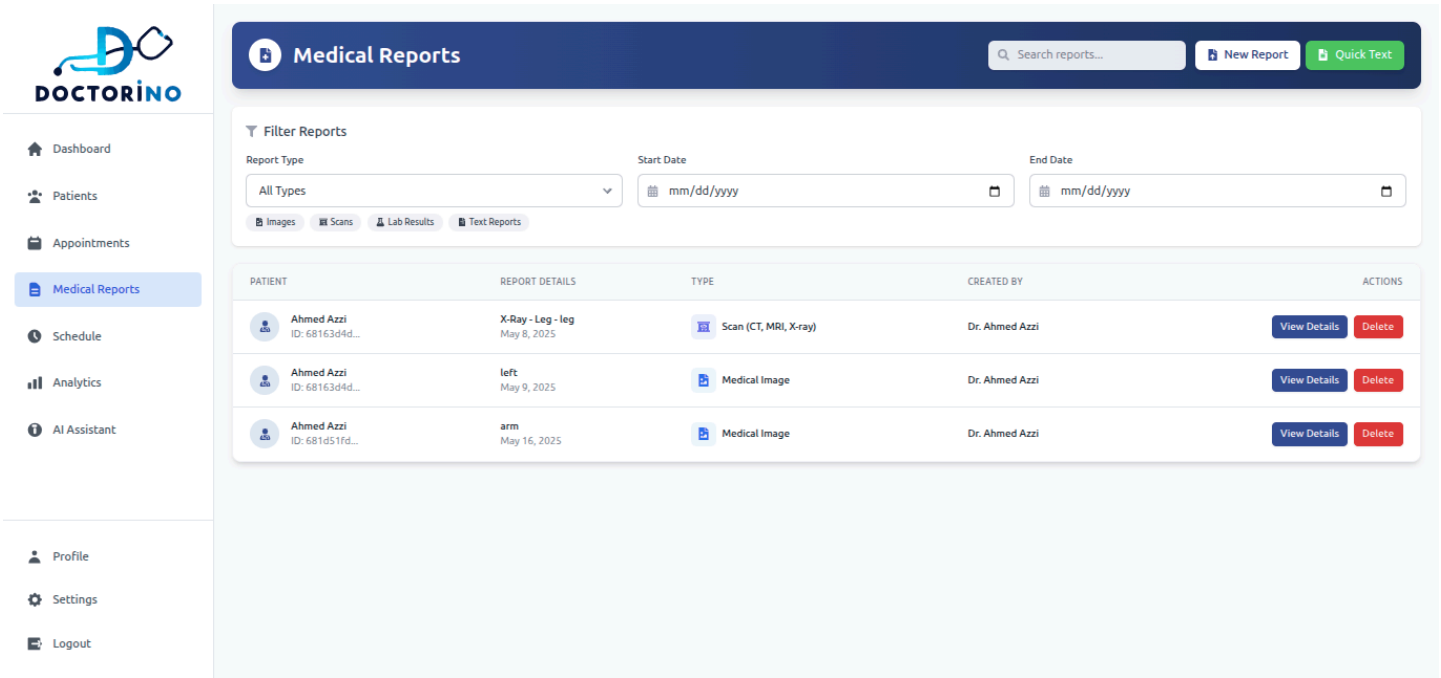


Figure 39: View medical records interface

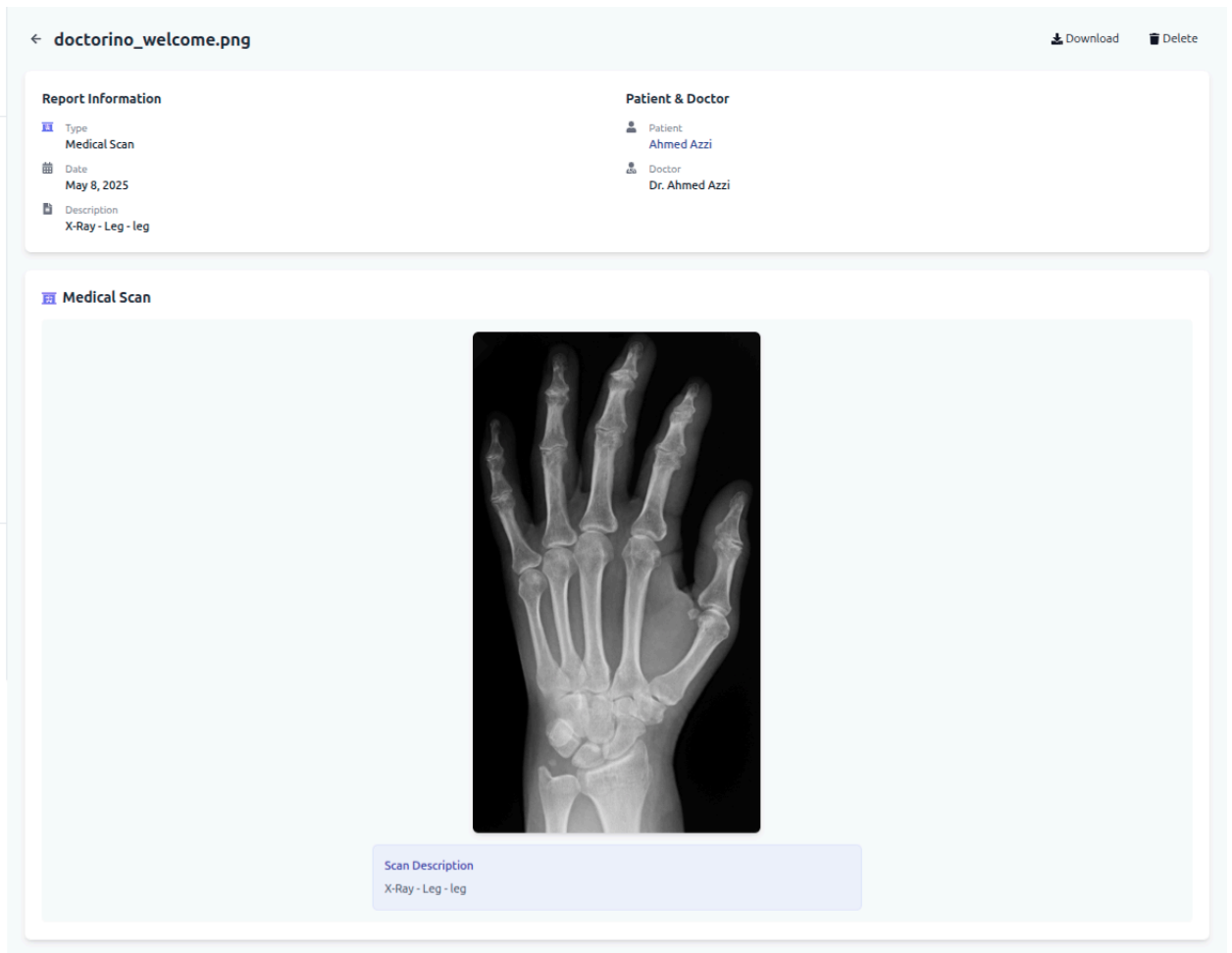
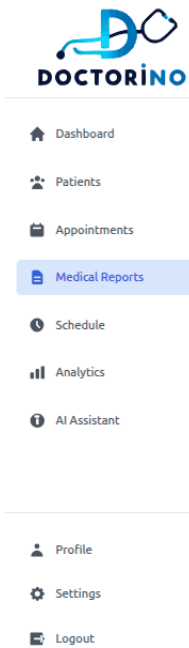


Figure 40: View medical record details interface

4.3.3.5 View all appointments

The interface shown in (Figure 40) provides doctors with a comprehensive view of their scheduled appointments, including patient details, date, time and the ability to add another appointment as well as view appointments details and write reports.

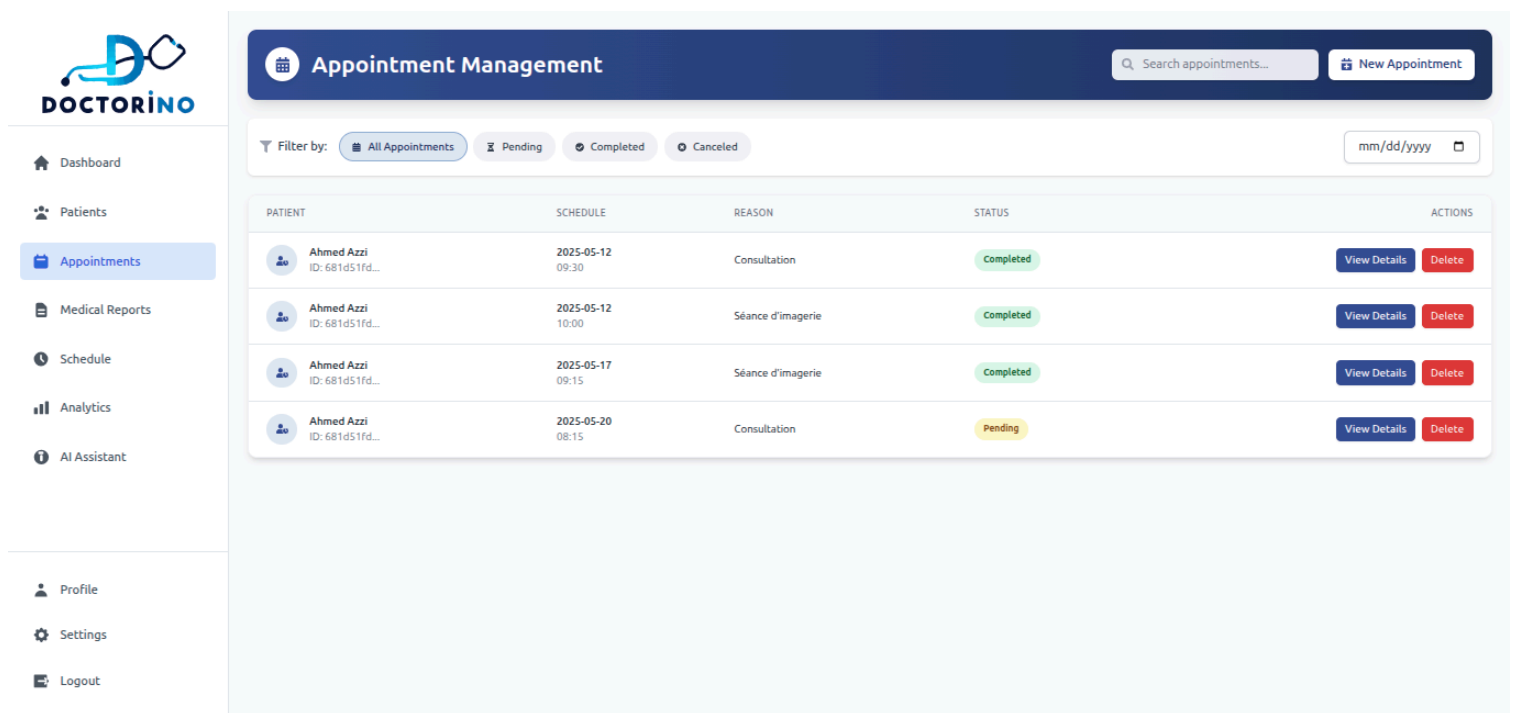


Figure 40: View appointments interface

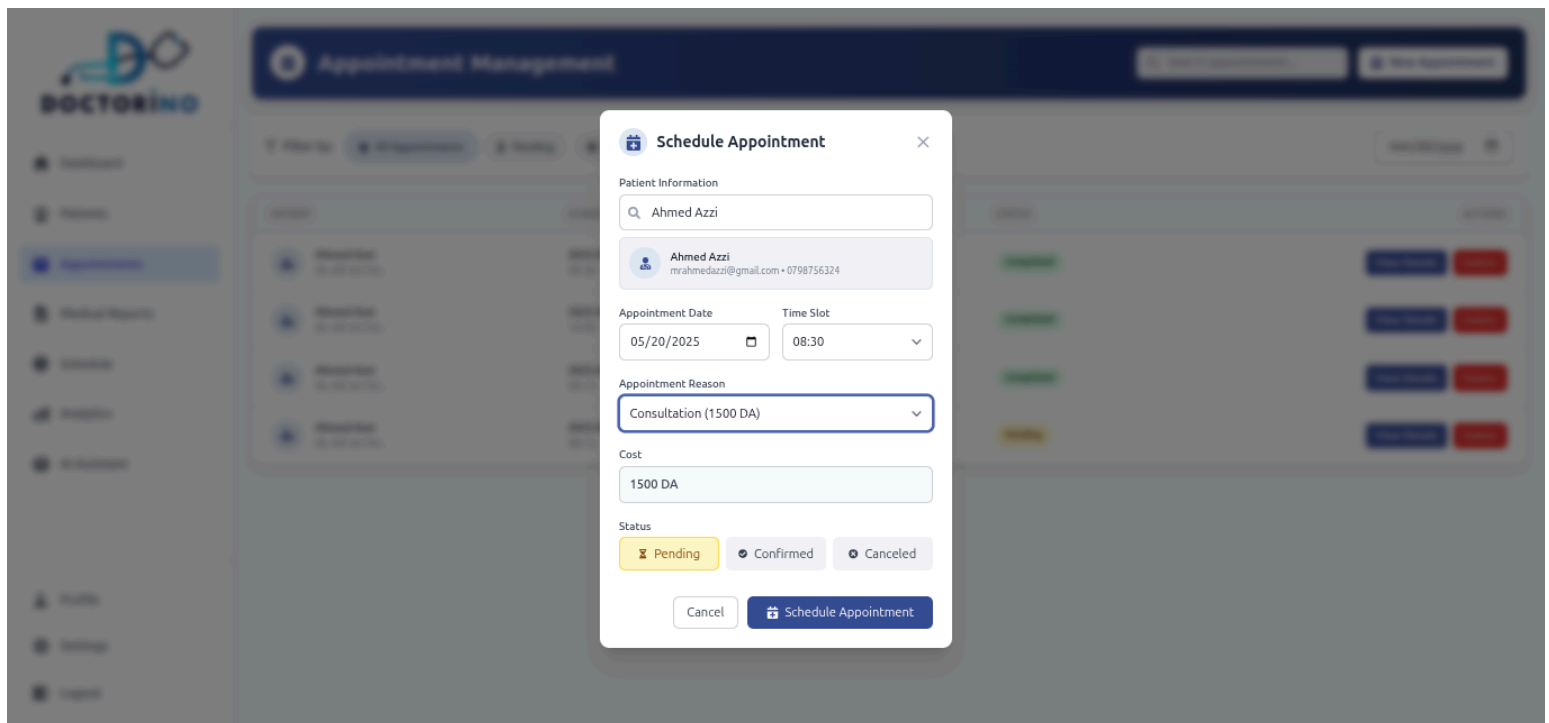


Figure 41: Add appointment interface

4.3.3.6 Appointment details

The interface shown in (Figure 43) displays detailed information about a specific appointment, including patient name and email, status of the appointment and the date and time of the appointment.

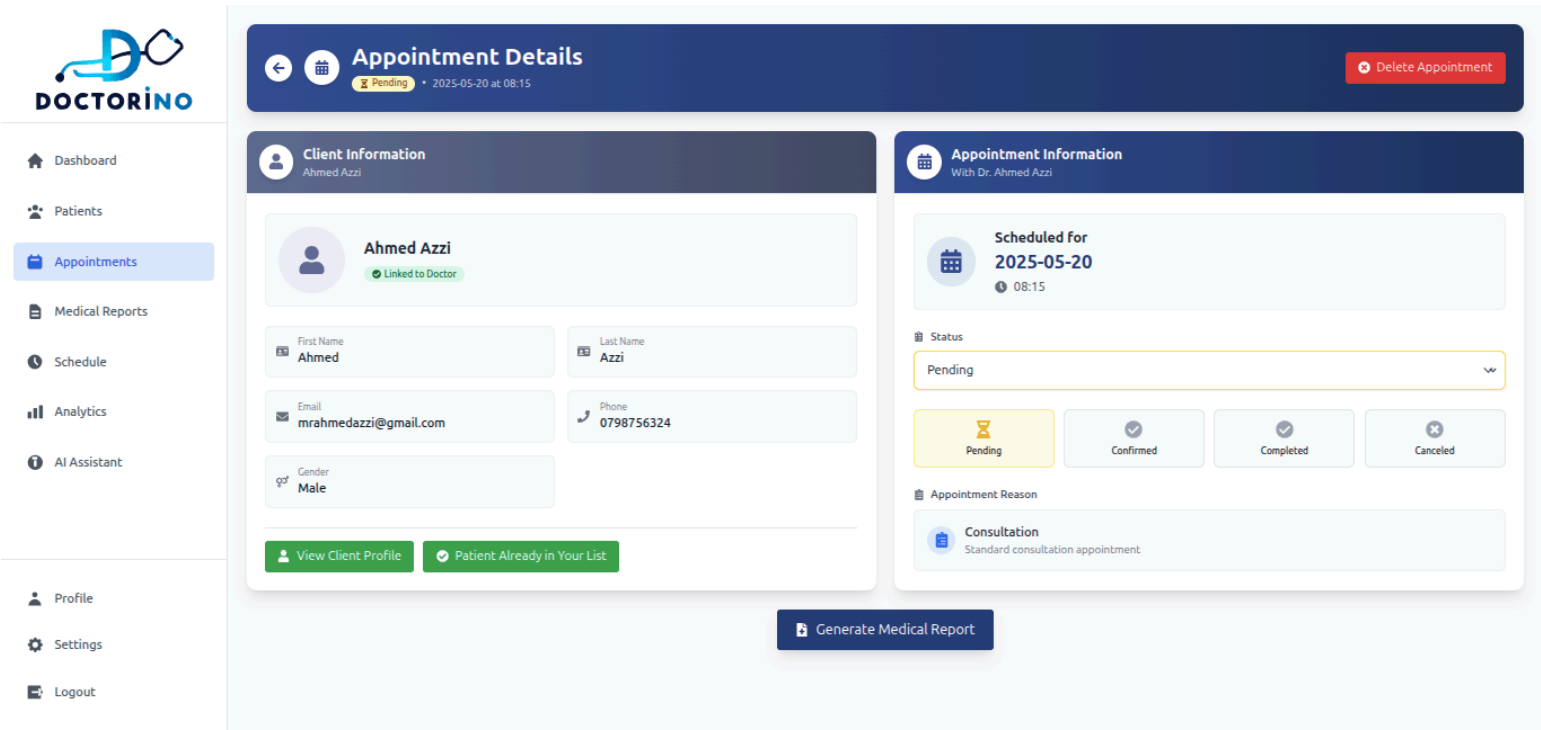


Figure 43: Appointment details interface

4.3.3.7 Manage schedule

The interface shown in (Figure 44) allows doctors to manage their appointment schedule. They are able to view old schedules, block out time for personal reasons, and potentially make changes to their schedule.

The screenshot displays the 'Doctorino' application interface for 'Schedule Management'. On the left is a sidebar with navigation links: Dashboard, Patients, Appointments, Medical Reports, Schedule (highlighted), Analytics, AI Assistant, Profile, Settings, and Logout. The main content area has a dark blue header with the title 'Schedule Management' and a subtitle 'Set your working hours and manage your availability'. Below the header, there are two main sections. The first section, 'Your Weekly Schedule', shows a grid of seven days. Monday through Thursday are marked 'Working' with specific time ranges (e.g., 09:00 - 17:00 for Monday) and a '30 min between appointments' rule. Friday is marked 'Day Off' with 'Not Available'. Saturday and Sunday are marked 'Working' with time ranges (09:15 - 16:00 and 09:00 - 17:00 respectively) and a '15 min between appointments' rule. Each day has an 'Edit' button. The second section, 'Modify Schedule', allows selecting a day to modify. It shows a row of buttons for each day: Mon Set, Tue Set, Wed Set, Thu Set, Fri Off, Sat Set, and Sun Set. Below this, the 'Monday Schedule' is shown with a 'Day Off' toggle. It includes fields for 'Start Time' (09:00 AM), 'End Time' (05:00 PM), and 'Appointment Duration (minutes)' (30 minutes). At the bottom, there is an 'Appointment Slots Preview' showing a timeline from 09:00 to 16:30 in 30-minute increments. At the very bottom right are 'Cancel' and 'Save Changes' buttons.

Figure 44: Manage schedule interface

4.3.5. The admin interfaces (Website)

4.3.5.1 Admin dashboard

The interface shown in (Figure 45) provides the admin with key performance indicators (KPIs) about the platform's performance. This might include metrics

such as platform revenue, number of doctors and patients registered, and number of appointments booked.

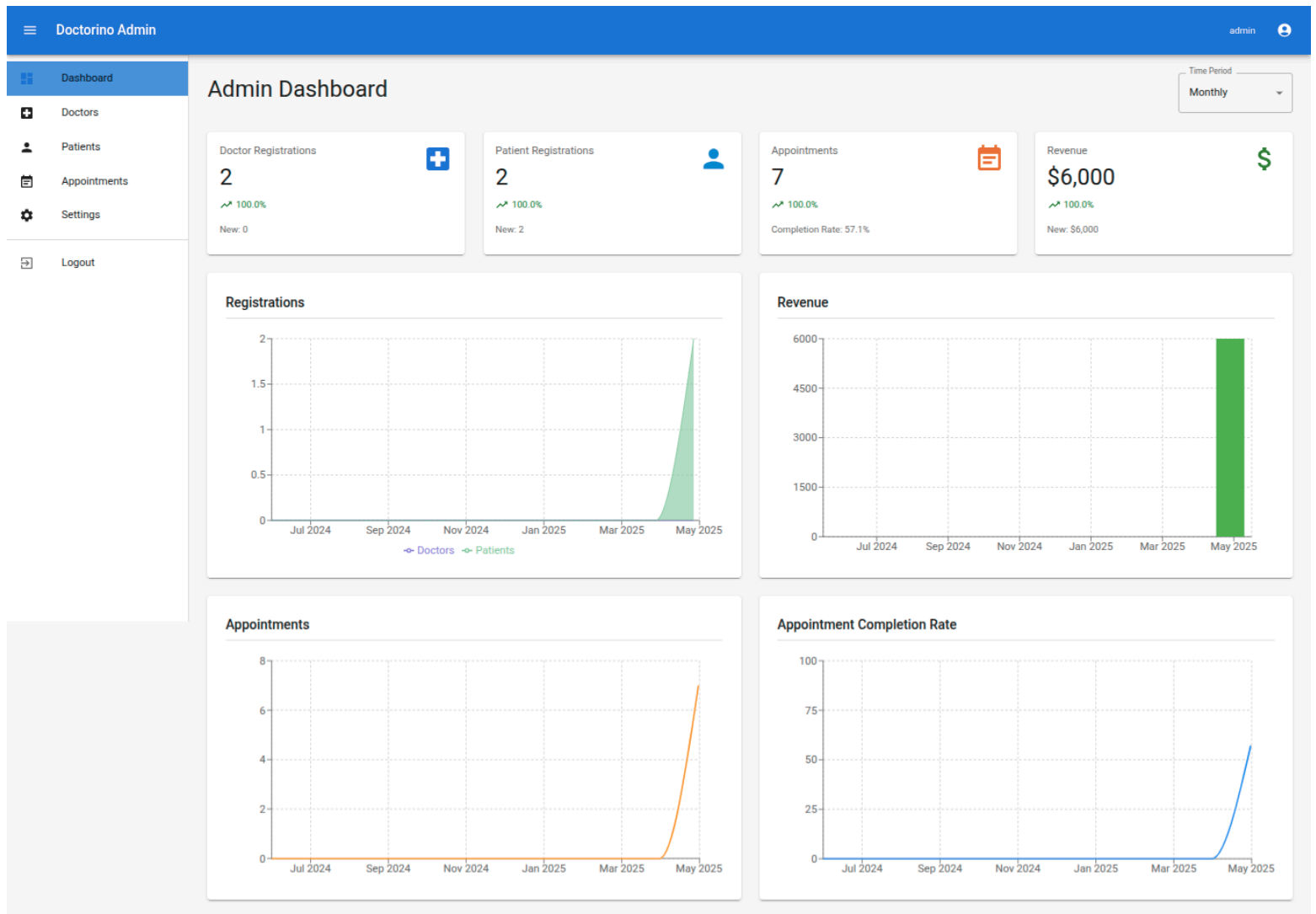


Figure 45: Admin Dashboard interface

4.3.5.2 View doctors

The interface shown in (Figure 46) allows the admin to view a list of all doctors currently registered on the platform. It may also provide options to search for specific doctors based on criteria and view their profiles.

Doctors					
Search doctors...					
Name	Username	Email	Specialization	Status	Actions
ahmed azzi	ahmed.azzi	ahmed@azzi.com	N/A	Active	View
soumia rokia	soumia.rokia	soumia@rokia.com	N/A	Active	View

Rows per page: 10 1-2 of 2

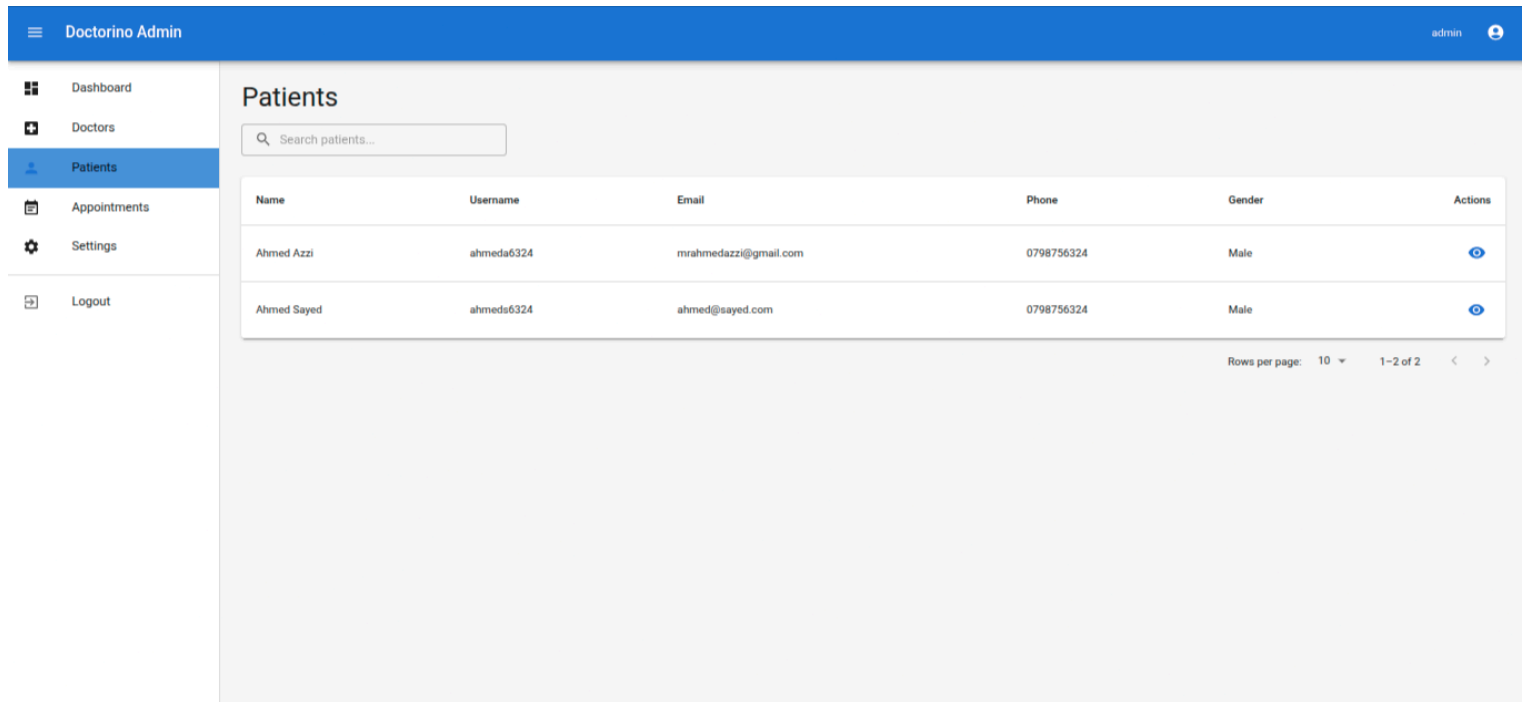
Figure 46: View Doctors interface

4.3.5.3 Manage doctor profile

The interface shown in (Figure 47) displays detailed information about a specific doctor, including their registration details and potentially patient reviews. The admin might also manage doctor status here.

4.3.5.4 View users (patients)

The interface shown in (Figure 48) allows the admin to view a list of all patients registered on the platform. It may also provide options to search for specific patients based on criteria.



The screenshot displays the 'Doctorino Admin' interface. On the left is a sidebar menu with options: Dashboard, Doctors, Patients (highlighted), Appointments, Settings, and Logout. The main content area is titled 'Patients' and features a search bar labeled 'Search patients...'. Below the search bar is a table with two columns: 'Name' and 'Actions'. The table contains two rows of patient data. The first row is for 'Ahmed Azzi' with username 'ahmeda6324', email 'mrahmedazzi@gmail.com', phone '0798756324', and gender 'Male'. The second row is for 'Ahmed Sayed' with username 'ahmeds6324', email 'ahmed@sayed.com', phone '0798756324', and gender 'Male'. Each row has a blue eye icon in the 'Actions' column. At the bottom right of the table, there is a pagination control showing 'Rows per page: 10' and '1-2 of 2'.

Name	Username	Email	Phone	Gender	Actions
Ahmed Azzi	ahmeda6324	mrahmedazzi@gmail.com	0798756324	Male	
Ahmed Sayed	ahmeds6324	ahmed@sayed.com	0798756324	Male	

Figure 48: View Patients interface

4.3.5.5 Manage applicants

The interface shown in (Figure 49) displays a list of doctors who have applied to join the platform but haven't yet been approved.

4.4. Perspective

We understand that the AI-powered virtual waiting room was a key feature you were looking forward to. While unforeseen technical challenges have prevented its implementation at this time, we are committed to providing a platform with the most advanced and reliable functionalities. We are actively exploring alternative solutions and will keep you informed of any future developments.

We appreciate your understanding and patience. We are confident that the future enhancements we've outlined will elevate your platform and deliver exceptional value for both patients and doctors.

4.5. Conclusion

In this chapter, we have outlined the key implementation tools and technologies that will form the foundation of our application development. By carefully selecting a robust set of solutions, we have laid the groundwork for a highly scalable and performant platform.

The integration of these various components will work in harmony to address the diverse requirements of our project. From data storage and caching to real-time communication and user interface development, each element has been chosen to optimize the overall performance, reliability, and maintainability of our platform.

By leveraging this powerful combination of tools and technologies, we can create a solution that is not only responsive and efficient but also adaptable to the evolving needs of our users. The careful integration and optimization of these components will be crucial in ensuring the long-term success and sustainability of our project.

With the implementation details in place, we are well-positioned to move forward and bring our application to life, delivering a remarkable user experience and meeting the high standards we have set for ourselves.

General Conclusion

In this interesting project, we have thoroughly explored the development of a scalable and performant application to address the needs of our target users. As we reflect on the outcomes and findings of this endeavour, we are confident that the solutions and strategies we have put in place will serve as a strong foundation for the continued success and growth of our application.

One of the key results of this project is the successful integration of a robust set of technologies and methodologies that have enabled us to create a system capable of adapting to the evolving requirements of our users. The careful selection and integration of these components have ensured that our application is not only scalable and performant but also maintainable and reliable.

Looking ahead, we see tremendous opportunities to further improve and enhance the subject matter of our application. By closely monitoring user feedback, analysing usage patterns, and staying abreast of industry trends, we can identify areas for improvement and implement targeted enhancements to deliver an even more engaging and valuable experience for our users.

Annex

Economic Study and Prototype Cost for Startups and BMC

Introduction

The Business Model Canvas (BMC) is widely adopted by entrepreneurial players such as incubators, venture capital firms, and chambers of commerce. It is considered one of the most used tools by startups. Developed by Alexander Osterwalder, the BMC provides a simple way to represent a company's business model. It is particularly well-suited for the creation phase, as well as the launch of

new products or services. The BMC consists of nine components (the terms may vary, but the concept remains the same).

By using the BMC, one can obtain indications about the company's financing, its value proposition, its target customer segment, and its cost structure. This tool allows for the mapping of the key elements of a startup project. It is useful not only for startups in general, but also for deep tech startups. However, it must be adapted to be effective. Indeed, a deep tech startup is not a typical startup. Although the BMC is generally comprehensive, one of its shortcomings lies in its overly general application, without distinction between a traditional startup and a deep tech startup. As each project is unique, the problems they face can differ. Therefore, it is essential to adapt the BMC to each specific case¹¹.

11

<https://eldorado.co/blog/2020/10/28/pourquoi-faut-il-adapter-le-business-model-canvas-aux-startups-deeptech> ()

Business Model Canvas

		Designed for:	Designed by:	Date:	Version:
Business Model Canvas		WellBe	WellBe team	06/03/2024	2.0
Key Partners direction de la sante	Key Activities - maintenance w updating. - Collecting and displaying users' reviews. Key Resources 1. Long term : - Software engineer. - Data base management	Value Propositions - Efficient Appointment Management. - Comprehensive Doctor Profiles. - Security and Privacy. - Remote Appointment Making - A waiting queue powered by Artificial Intelligence.	Customer Relationships - Users community to rate doctors. Channels - Web application. - Desktop application. - Social media - Marketing	Customer Segments - Patients - Doctors - clinics	
Cost Structure - Applications development and management. - Administration and monitoring. - Marketing and promotions.		Revenue Streams Subscription.			

Business Plan

Launch Costs

Table 3: Launch Costs table

Equipment	Quantity	price (DZD)
Pc	4	70,000
Desk	4	10,000
Chair	6	10,000
Server	1	150,000
Printer	2	30,000
Router	1	10,000
Engineering software		1,580,000
Office supplies		50000

Total: 2,230,000 DZD

Charges

Table 4: Charges Table

Charge	Price per month (DZD)
Water	1500
Electricity	10000
Internet	2500
Rent	20000
Marketing	70000

Total :109000 DZD

Employees charges

Table 5: Employees Charges Table

	Salary (DZD)	Quantity	total
Accountant	40000	1	40000
Software engineers	50000	4	200000

Total :240000 DZD

Amortization

Table 6: Amortization Table

	Buy price	duration	Amortization (DZD)
Pc	280000	5	4666
Desk	40000	5	666
Chair	60000	5	1000
Server	150000	7	1785
Router	10000	5	166
printer	30000	5	1000

Conclusion

The adaptation of the Business Model Canvas (BMC) allows deep tech startups to better understand their environment and demonstrate agility. The BMC can be used during fundraising and partner searches, providing a clear and flexible vision. It also fosters positive synergies, by aligning with the strategies of investors or large companies, and anticipating technology transfer mechanisms. This approach demonstrates a real strategic action. It is important to note that this adaptation proposal should not be rigid, but rather evolve according to the specific needs of the companies. The main objective is to advance deep tech startups, without locking them into constraints or excessive generalizations.

Bibliography

Chapitre 1

1. World Health Organization. *Global Strategy on Digital Health 2020–2025*. 2021.
2. Ministry of Health (Algeria). *Plan National de Santé 2019–2024*. Algerian Health Authority, 2019.
3. Shortliffe EH & Sepúlveda MJ. “Clinical Decision Support in the Era of Artificial Intelligence.” *JAMA* 320(21):2199–2200, 2018. doi:10.1001/jama.2018.17163
4. Topol E. *Deep Medicine: How Artificial Intelligence Can Make Healthcare Human Again*. Basic Books, 2019.
5. Bates DW et al. “Ten commandments for effective clinical decision support: making the practice of evidence-based medicine a reality.” *J Am Med Inform Assoc* 10(6):523–530, 2003. doi:10.1197/jamia.M1370
6. ISO 9241-210:2019. *Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems*. ISO, 2019.
7. Zhang J & Walji M. “TURF: Toward a Unified Framework of EHR Usability.” *J Biomed Inform* 45(6):1048–1055, 2012. doi:10.1016/j.jbi.2012.06.009
8. European Commission. *Regulation (EU) 2017/745 on medical devices*. Official Journal of the European Union, 2017.

9. Reddy MC & Spence PR. "Usability Challenges in Web-Based Medical Records Systems." *IEEE IT Professionals* 13(4):48–55, 2011.
doi:10.1109/MITP.2011.60
10. Office of the National Coordinator for Health Information Technology (ONC). *Usability and EHRs*. HealthIT.gov, 2014.
11. Shortliffe EH. *Knowledge Engineering: Principles and Practice*. Academic Press, 1991.
12. Greenhalgh T et al. "What is quality in health care? Reviewing the WHO definition of quality." *BMJ Qual Saf* 27(1):7–12, 2018.
doi:10.1136/bmjqs-2017-006731
13. Agarwal R, Gao G, DesRoches C & Jha AK. "Research Commentary—The Digital Transformation of Healthcare: Current Status and the Road Ahead." *Inf Syst Res* 21(4):796–809, 2010. doi:10.1287/isre.1100.0327
14. Carayon P et al. "Human factors systems approach to healthcare quality and patient safety." *Appl Ergon* 45(1):14–25, 2014.
doi:10.1016/j.apergo.2013.04.023
15. Davis FD. "Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology." *MIS Q* 13(3):319–340, 1989.
doi:10.2307/249008

Chapitre 2

16. Chen P. "The Entity-Relationship Model—Toward a Unified View of Data." *ACM Trans Database Syst* 1(1):9–36, 1976. doi:10.1145/320434.320440
17. Elmasri R & Navathe S. *Fundamentals of Database Systems*, 7^e éd. Pearson, 2016.

- 18.Date CJ. *An Introduction to Database Systems*, 8^e éd. Addison-Wesley, 2003.
- 19.Fowler M. *UML Distilled: A Brief Guide to the Standard Object Modeling Language*, 3^e éd. Addison-Wesley, 2003.
- 20.Booch G, Rumbaugh J & Jacobson I. *The Unified Modeling Language User Guide*, 2^e éd. Addison-Wesley, 2005.
- 21.Ambler SW. *The Elements of UML 2.0 Style*. Cambridge Univ. Press, 2005.
- 22.Larman C. *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design*, 3^e éd. Prentice Hall, 2004.
- 23.Fowler M. *Patterns of Enterprise Application Architecture*. Addison-Wesley, 2002.
- 24.Hoffer J, Ramesh V & Topi H. *Modern Database Management*, 12^e éd. Pearson, 2015.
- 25.ISO/IEC 19501:2005. *Information technology — Open Distributed Processing — Unified Modeling Language (UML)*. ISO, 2005.

Chapitre 3

- 26.Russell S & Norvig P. *Artificial Intelligence: A Modern Approach*, 4^e éd. Pearson, 2020.
- 27.Goodfellow I, Bengio Y & Courville A. *Deep Learning*. MIT Press, 2016.
- 28.Esteva A et al. “Dermatologist-level classification of skin cancer with deep neural networks.” *Nature* 542:115–118, 2017. doi:10.1038/nature21056

29. Rajkomar A, Dean J & Kohane I. "Machine Learning in Medicine." *N Engl J Med* 380:1347–1358, 2019. doi:10.1056/NEJMra1814259
30. Topol E. "High-Performance Medicine: The Convergence of Human and Artificial Intelligence." *Nat Med* 25:44–56, 2019. doi:10.1038/s41591-018-0300-7
31. Shorten C & Khoshgoftaar T. "A survey on Image Data Augmentation for Deep Learning." *J Big Data* 6:60, 2019. doi:10.1186/s40537-019-0197-0
32. Liu X et al. "Challenges in the Development of Machine Learning Models for Health Care." *JAMA Intern Med* 178(11):1544–1551, 2018. doi:10.1001/jamainternmed.2018.2533
33. Chicco D. "Ten quick tips for machine learning in computational biology." *BioData Min* 10:35, 2017. doi:10.1186/s13040-017-0155-3
34. Holzinger A et al. "What do we need to build explainable AI systems for the medical domain?" *Rev Comput Sci Week*, 2017.
35. Rudin C. "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead." *Nat Mach Intell* 1:206–215, 2019. doi:10.1038/s42256-019-0048-x
36. Miotto R, Wang F, Wang S, Jiang X & Dudley J. "Deep learning for healthcare: review, opportunities and challenges." *Brief Bioinform* 19(6):1236–1246, 2018. doi:10.1093/bib/bbx044
37. Rajpurkar P et al. "CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning." arXiv:1711.05225, 2017.
38. Devlin J, Chang M-W, Lee K & Toutanova K. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." *NAACL-HLT*, 2019.

39. Lee C, Yoon E & Kim S. "Clinical BERT: Modeling Clinical Notes and Predicting Hospital Readmission." arXiv:1904.05342, 2019.
40. Lituiev D et al. "Mobile health applications with AI: A systematic review." *Telemed J E Health*, 2020. doi:10.1089/tmj.2019.0156

Chapitre 4

41. Van Rossum G & Drake FL. *Python 3 Reference Manual*. Python Software Foundation, 2009.
42. Paszke A et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library." *NeurIPS*, 2019.
43. Vaswani A et al. "Attention Is All You Need." *NeurIPS*, 2017.
44. Zhang Z & Sabuncu MR. "Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels." *NeurIPS*, 2018.
45. Chollet F. *Keras: The Python Deep Learning API*. GitHub, 2015.
46. Kingma DP & Ba J. "Adam: A Method for Stochastic Optimization." *ICLR*, 2015.
47. Redmon J & Farhadi A. "YOLOv3: An Incremental Improvement." arXiv:1804.02767, 2018.
48. Chollet F. "Xception: Deep Learning with Depthwise Separable Convolutions." *CVPR*, 2017.

49. Howard AG et al. "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications." arXiv:1704.04861, 2017.
50. Microsoft Azure. *Azure Machine Learning Service Documentation*. Microsoft.
<https://learn.microsoft.com/azure/machine-learning/>
51. Google Cloud. *AI Platform Documentation*. Google Cloud.
<https://cloud.google.com/ai-platform>
52. Amazon Web Services. *Amazon SageMaker Developer Guide*. AWS.
<https://docs.aws.amazon.com/sagemaker/latest/dg/>
53. TensorFlow Team. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*.
<https://www.tensorflow.org/>
54. Apache Kafka. *Kafka: A Distributed Streaming Platform*. Apache Software Foundation.
<https://kafka.apache.org/documentation/>
55. Docker Inc. *Docker Documentation*.
<https://docs.docker.com/>
56. Kubernetes SIG. *Kubernetes Documentation*.
<https://kubernetes.io/docs/>
57. FastAPI. *FastAPI Documentation*.
<https://fastapi.tiangolo.com/>
58. ReactJS. *React – A JavaScript Library for Building User Interfaces*.
<https://reactjs.org/docs/getting-started.html>

59. Tailwind Labs. *Tailwind CSS Documentation*.

<https://tailwindcss.com/docs>

60. Schafer RW, Beal J & Cohen I. *Multirate Signal Processing for Communication Systems*. Prentice Hall, 2008.

Étude économique & Business Model Canvas

61. Osterwalder A & Pigneur Y. *Business Model Generation*. Wiley, 2010.

62. Osterwalder A, Pigneur Y, Bernarda G & Smith A. *Value Proposition Design*. Wiley, 2014.

63. Blank S & Dorf B. *The Startup Owner's Manual*. K&S Ranch, 2012.

64. Ries E. *The Lean Startup*. Crown Business, 2011.

65. Maurya A. *Running Lean*. O'Reilly, 2012.

66. Christensen CM. *The Innovator's Dilemma*. Harvard Business Review Press, 1997.

67. Porter M. *Competitive Strategy: Techniques for Analyzing Industries and Competitors*. Free Press, 1980.

68. Kaplan RS & Norton DP. *The Balanced Scorecard: Translating Strategy into Action*. Harvard Business School Press, 1996.

69. Gassmann O, Frankenberger K & Csik M. *The St. Gallen Business Model Navigator*. Wiley, 2014.

70. Johnson MW, Christensen CM & Kagermann H. "Reinventing Your Business Model." *Harvard Bus Rev*, 2008.

الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم العالي والبحث العلمي

جامعة عبد الحميد بن باديس-مستغانم-
كلية العلوم الاجتماعية و الإنسانية

تصريح شرفي خاص بالالتزام بقواعد النزاهة العلمية
لإنجاز البحث

أنا الممضي أدناه،

الطالب(ة): أحمد عزي رقم التسجيل الجامعي: 202037028706 الحامل

لبطاقة التعريف الوطنية رقم: 100010963052700007 والصادرة

بتاريخ: 2024.10.31

عن بلدية مستغانم

المسجل بكلية العلوم الدقيقة والإعلام الآلي / قسم سنة ثانية ماستر شعبة إعلام

آلي / التخصص الذكاء الاصطناعي وإنترنت الأشياء

و المكلف بإنجاز مذكرة ماستر بعنوان:

منصة طبية ذكية بمساعدة الذكاء الاصطناعي

أصرح بشرفي أنني ألتزم بمراعاة المعايير العلمية والمنهجية ومعايير الأخلاقيات

العلمية والنزاهة الأكاديمية المطلوبة في إنجاز البحث ، وأتحمل المسؤولية

الشخصية عن كل المحتوى المتضمن في البحث المذكور أعلاه .

التاريخ: 2025.06.09

إمضاء المعني



* ملحق القرار الوزاري رقم 933 المؤرخ في 28 جويلية 2016 الذي يحدد القواعد المتعلقة بالوقاية من السرقة العلمية ومكافحتها.