PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

UNIVERSITY OF 20 AOUT 1955 - SKIKDA

Faculty of Sciences

Department of Computer Science

Dissertation Submitted in partial fulfillment of the requirements for the master's degree in computer science.

Option: artificial intelligence

Theme:

Dawa2y

Advanced Medical tracking and assistance system

Presented by:

BACHAGHA Mousaab

BENAILI Rabah Amine

LAZREK Mohamed Heitem

Supervised by:

Dr. LAHSASNA Adel

2024 – 2025

Acknowledgements

First and foremost, we express our deepest gratitude to **Almighty God** for His guidance and countless blessings throughout this journey. Without His divine support, our success would not have been possible.

We would also like to thank our supervisor, **Dr. LAHSASNA Adel**, for his knowledge, experience, advice, and guidance, thanks to which we were able to produce high-quality work in such a short time.

We would like to thank our families for their unwavering psychological support, encouragement, and the resources they provided, all of which played a vital role in our success.

We would like to thank all the teaching staff of the computer science department for their dedication and work, which has contributed to our learning

Abstract

The rapid development of digital technologies has significantly transformed the healthcare sector by improving patient access, service efficiency, and data management. This thesis presents an enhanced version of the "Dawa2y" platform transforming it into an intelligent, AI-powered medical and appointment system designed to modernize and automate healthcare processes. The platform integrates Artificial Intelligence technologies such as Optical Character Recognition (OCR) for document digitization, Natural Language Processing (NLP) for information extraction, and a medical chatbot for patient interaction. It also includes an intelligent appointment management system that simplifies scheduling and reduces administrative burdens. The system offers secure, real-time services through both web and Android applications, improving communication between doctors, patients, pharmacies, and administrators. This work demonstrates the potential of AI-driven healthcare applications to enhance service quality, patient engagement, and operational effectiveness.

Keywords: Artificial Intelligence (AI), OCR (Optical Character Recognition), NLP (Natural Language Processing), Chatbot, Medical Application, Platform, mHealth (Mobile health) , electronic health (eHealth)

Résumé

Le secteur de la santé a connu une transformation majeure grâce au développement rapide des technologies numériques, améliorant l'accès des patients aux soins, l'efficacité des services et la gestion des données. Ce mémoire présente une version améliorée de la plateforme "Dawa2y", un système médical intelligent basé sur l'intelligence artificielle et conçu pour moderniser et automatiser les processus de santé traditionnels. La plateforme intègre des technologies telles que la Reconnaissance Optique de Caractères (OCR) pour la numérisation des documents, le Traitement Automatique du Langage Naturel (TALN) pour l'extraction d'informations, ainsi qu'un chatbot médical interactif pour l'assistance aux patients. Un système de gestion des rendez-vous intelligent permet également de simplifier la planification et de réduire les charges administratives. Le système propose des services sécurisés et en temps réel via des applications web et Android, facilitant la communication entre médecins, patients, pharmacies et administrateurs. Ce travail démontre le potentiel des applications de santé basées sur l'IA pour améliorer la qualité des services, l'engagement des patients et l'efficacité opérationnelle.

Mots-clés: Intelligence Artificielle (IA), ROC (Reconnaissance Optique de Caractères), TALN (Traitement Automatique du Langage Naturel), Chatbot, Application Médicale, Plateforme, m-Santé (santé mobile), e-Santé (santé électronique)

الملخص

شهد قطاع الرعاية الصحية تحولًا كبيرًا بفضل التطور السريع للتقنيات الرقمية، مما ساهم في تحسين وصول المرضى إلى الخدمات وزيادة كفاءة العمل وإدارة البيانات. يقدم هذا البحث تطويرًا متقدمًا لتطبيق "دوائي" كنظام ذكي يعتمد على تقنيات الذكاء الاصطناعي، يهدف إلى تحديث وأتمتة العمليات الصحية التقليدية. يتضمن النظام تقنيات حديثة مثل التعرف البصري لاستخراج المعلومات، بالإضافة (NLP) لتحويل الوثائق الورقية إلى رقمية، ومعالجة اللغة الطبيعية (OCR) على الحروف إلى روبوت دردشة طبي لمساعدة وتوجيه المرضى. كما يوفر النظام إدارة ذكية للمواعيد الطبية، مما يسهل الحجز ويقلل الأعباء الإدارية. يتيح النظام خدمات آمنة وفي الوقت الحقيقي عبر تطبيقات الويب والأندرويد، مما يعزز التواصل بين الأطباء والمرضى والصيدليات والإدارة. يبرز هذا العمل إمكانات تطبيقات الذكاء الاصطناعي في تحسين جودة الخدمات الصحية وزيادة تفاعل المرضى وفعالية التشغيل

الكلمات المفتاحية:

، روبوت الدردشة، التطبيق الطبي، المنصة، معالجة اللغة الطبيعية ، التعرف الضوئي على الحروف ,الذكاء الاصطناعي الصحة الإلكترونية , الصحة المتنقلة

Table of Contents

Table of Figures

Chapter 1 : General Introduction

1.1. Context and Motivation

The rapid advancement of digital technologies has profoundly impacted various sectors, with healthcare experiencing significant transformation. Mobile health (mHealth) applications, in particular, have emerged as powerful tools for improving healthcare access, patient engagement, and overall service efficiency. Recognizing this potential, the "Dawa2y" application was initially developed during our bachelor's degree project. This foundational version aimed to address the challenges of managing paper-based prescriptions and enhancing medication adherence through digital tracking and basic doctor-patient communication functionalities.

However, the evolving landscape of healthcare demands even more sophisticated and intelligent solutions. The integration of Artificial Intelligence (AI) offers unprecedented opportunities to augment the capabilities of mHealth platforms, leading to more personalized, proactive, and efficient patient care. This master's thesis builds directly upon the foundational work of the "Dawa2y" application. It embarks on a significant expansion by introducing advanced Artificial Intelligence (AI) capabilities and a comprehensive appointment management system. The motivation behind this advanced development is to further revolutionize healthcare delivery by empowering both patients and healthcare professionals with intelligent tools that streamline workflows, enhance decision-making, and improve the overall healthcare experience. This work seeks to leverage AI to tackle more complex challenges within the medical field, moving beyond basic management to intelligent assistance and interaction.

1.2. Problem Statement

While the initial version of "Dawa2y" addressed fundamental issues such as inefficiencies in prescription management and medication tracking, the complexities of modern healthcare present further challenges that AI-enhanced systems are uniquely positioned to solve. Healthcare professionals often face information overload when dealing with extensive patient records, and manual data entry from physical documents remains a time-consuming and error-prone process. Patients, on the other hand, may lack immediate access to support for common queries or face cumbersome, manual processes for scheduling medical appointments.

This Master's project specifically addresses these prevailing issues by enhancing "Dawa2y" to:

- Mitigate the cognitive burden on doctors by providing AI-powered assistance in analyzing medical documents and patient data.
- Reduce inefficiencies and errors associated with manual data input through an intelligent AI-based scanning and document understanding system.
- Improve patient support and accessibility by developing an interactive AI chatbot capable of handling queries and assisting with tasks.
- Streamline the **appointment booking and management process** for both patients and healthcare providers, reducing administrative overhead and improving access to care.

1.3. Objectives of the Master's Project

The primary goal of this master's thesis is to design, implement, and conceptually evaluate the integration of advanced AI functionalities and an appointment management system into the "Dawa2y" platform. The specific objectives are:

1. To design and develop an **AI-powered Document Aid System** that assists doctors in interpreting and extracting key information from medical documents (e.g., lab reports, patient histories).
2. To implement an **AI-driven Scanning Feature** capable of accurately digitizing and understanding information from physical medical documents using OCR and NLP techniques.
3. To create and integrate an **Intelligent Medical Chatbot** within the patient-facing "Dawa2y" application to provide instant support, answer frequently asked questions, and guide users through application features, including appointment booking.
4. To design and implement a robust and user-friendly **Appointment Management System** enabling patients to schedule, view, and manage appointments, and allowing doctors to manage their availability and appointment schedules efficiently.
5. To ensure seamless integration of these new AI modules and the appointment system with the existing "Dawa2y" backend and frontend infrastructure.

1.4. Scope and Limitations of the Master's Project

The scope of this Master's project encompasses the design, development, and integration of the AI-powered document aid and scanner, the patient-facing chatbot, and the appointment management system within the "Dawa2y" framework. The project will focus on processing medical prescriptions, patient health summaries, and scanned handwritten notes, while the chatbot will assist patients with medication inquiries, appointment scheduling, and general health information, and the appointment system will enable real-time booking, notifications, and doctor-side schedule management.

Limitations of this project include the AI models that will be trained on a limited dataset and may require further refinement for broader generalization, and extensive clinical validation is beyond the scope of this master's thesis. Additionally, the current implementation focuses on the Android platform for patient applications, with plans for cross-platform support in future work. The project will primarily focus on functional implementation and conceptual business modeling rather than large-scale deployment or exhaustive performance benchmarking against all commercial alternatives.

1.5. Thesis Outline

This thesis is structured into 4 chapters to systematically present the research, design, implementation, and analysis of the enhanced "Dawa2y" application:

- **Chapter 1: General Introduction** provides the context, problem statement, objectives, scope, and outline of the thesis.
- **Chapter 2: State of the Art and Literature Review** reviews mHealth platforms, AI in healthcare (document processing, chatbots, CDSS), and appointment systems.
- **Chapter 3: System Design and Architecture** details system design, UML diagrams (Use Case, Class, Sequence, Activity)
- **Chapter 4: Implementation** describes development tools, languages, user interfaces, technical enhancements (encryption, bitmask booking, OCR-LLM), and validation metrics.

- **General Conclusion** summarizes the work undertaken, highlights key achievements and contributions, discusses limitations, and proposes directions for future research and development.

Chapter 2 : State of the Art and Literature Review

2.1. Introduction

The rapid evolution of digital technologies has profoundly influenced the healthcare sector, giving rise to new models of care delivery, data management, and patient engagement. Mobile health (mHealth) and electronic health (eHealth) solutions have become integral to improving healthcare accessibility, reducing administrative burden, and enhancing the quality of care provided to patients.

This chapter aims to explore the current state of technological advancement in the domains most relevant to the development of the Dawa2y application. It provides an overview of existing medical applications, their functionalities, and the challenges they present. Furthermore, it examines the growing role of artificial intelligence in healthcare, especially in areas such as document processing, patient communication via chatbots, and appointment scheduling systems.

By analyzing existing platforms and technologies, this chapter identifies the strengths and shortcomings of current solutions and highlights the need for a more integrated, intelligent system—an area where Dawa2y aims to contribute significantly.

2.2. Digital Transformation in Healthcare

2.2.1. Role of Digital Tools in Modern Healthcare

The integration of digital technologies has significantly transformed healthcare delivery, enhancing efficiency, accessibility, and patient-centered care. Key technologies driving this transformation include the Internet of Things (IoT), Artificial Intelligence (AI), big data analytics, cloud computing, and social media. These tools have been instrumental in streamlining administrative processes, improving clinical decision-making, and facilitating real-time patient monitoring. For instance, AI and machine learning algorithms are increasingly utilized to analyze vast amounts of healthcare data, leading to improved diagnostic accuracy and personalized treatment plans [1].

A study by Mauro et al. (2024) employed Porter's value chain model and the technology–organization–environment framework to assess the impact of digital technologies on managerial

support processes in healthcare. The findings revealed that IoT and AI have the most significant impact on administrative support processes, while blockchain technology was found to be less relevant in this context. Moreover, the study highlighted that the skills and competencies of employees are crucial determinants for the successful adoption of digital technologies in healthcare organizations [1] .

2.2.2. Rise of mHealth and eHealth Platforms

Mobile Health (mHealth) and electronic Health (eHealth) platforms have emerged as pivotal components of digital healthcare, offering innovative solutions for remote monitoring, patient engagement, and chronic disease management. mHealth applications encompass a wide range of tools, including telemedicine platforms, mobile apps for chronic disease management, and wearable devices. These applications have been shown to improve patient outcomes, increase access to healthcare services, and reduce healthcare costs. However, challenges such as privacy and security concerns, lack of technical infrastructure, and regulatory issues need to be addressed to ensure the successful adoption and utilization of mHealth applications [2].

A comprehensive review by Singh Jat and Grønli (2024) emphasized the transformative potential of mHealth applications in healthcare delivery. The study highlighted that while mHealth applications offer numerous benefits, their successful implementation requires addressing challenges related to data accuracy, user adherence, and the digital divide. Furthermore, the integration of mHealth data with electronic health records and the development of AI-driven mHealth applications are identified as future directions to enhance healthcare delivery [2].

2.3. Existing Medical Applications and Their Limitations

2.3.1. Doctolib

Doctolib is a French company founded in 2013 by Stanislas Niox-Chateau, Steeve Abou Rjeily, Ivan Schneider and Jessy Bernal.

Doctolib distributes an online patient-friendly appointment booking service in France, Italy, Germany and the Netherlands. In addition to this feature, the website offers software services to healthcare professionals [3].

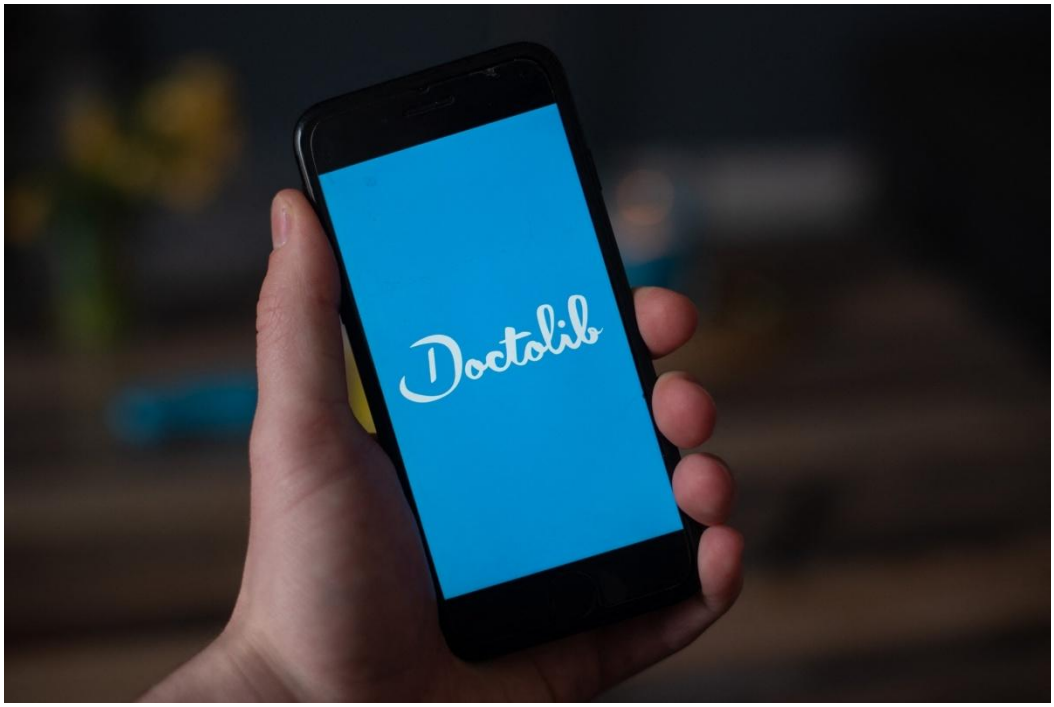

Figure 2-1. doctolib application

Key Features:

- User-friendly interface for both patients and healthcare providers
- Online booking and calendar synchronization.
- Automated appointment reminders via email or SMS.
- Secure video consultation capabilities.

Limitations:

- Limited customization options for healthcare providers.
- Dependence on stable internet connectivity; issues may arise in areas with poor connectivity.

2.3.2. NHS App

The **NHS App** allows patients using the National Health Service in England to book appointments with their General Practitioner, order repeat prescriptions and access their GP record. Available since late 2018, the app was developed by NHS Digital and NHS England. The health ministers Jeremy Hunt and Matt Hancock both stressed their support for the project. Hancock presented it as key to a radical overhaul of NHS technology. Hunt said it would mark "the death-knell of the 8am scramble for GP appointments that infuriates so many patients".

The app can also be used to access NHS 111, set patients' data sharing preferences, and record organ donation preferences and end-of-life care preferences. All GPs in England are required to give their patients access to their health record via the NHS app. The app allows a user to manage healthcare services for others, such as parents or spouses. [4]

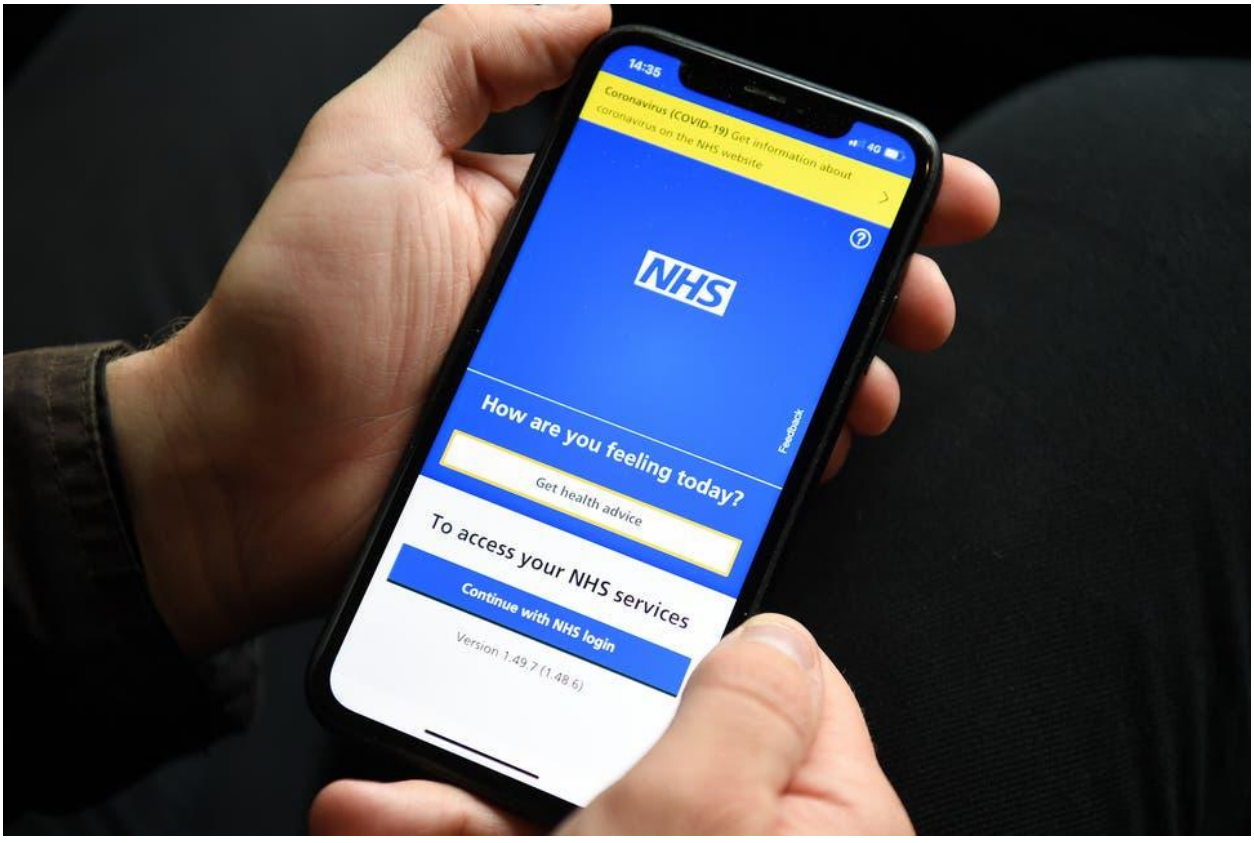

Figure 2-2. NHS app

Key Features:

- **Prescription Management**: Order repeat prescriptions and choose a pharmacy.
- **GP Appointment Booking**: Book and cancel appointments at your registered GP surgery.
- **Medical Record Access**: View your GP health record, including allergies and medications.
- **Hospital Referrals**: View and manage hospital referrals and outpatient appointments.
- **Health Advice**: Access NHS 111 online, symptom checkers, and health information.
- **Linked Profiles**: Manage health services for children or someone you care for.
- **Notifications**: Receive messages and updates from your GP surgery.
- **NHS Number**: View and share your NHS number.
- **Organ Donation**: Register and manage your organ donation preferences.
- **Data Sharing**: Manage consent for sharing your health data for research and planning.

Limitations

- **Feature Availability**: Not all features are available in every area; availability depends on local GP practices.
- **Access to Medical Records**: Some users report limited access to detailed medical records due to GP practice settings.
- **Technical Issues**: Users may experience difficulties with registration, login, and accessing certain features.
- **Digital Literacy**: Some patients, especially older adults, may find the app challenging to use without assistance.
- **Language Support**: The app is primarily available in English, which may pose challenges for non-English speakers.
- **Device Compatibility**: Certain features may not function optimally on all devices or operating systems.

2.3.3. Healthily

Your.MD also known as **Healthily**, is a digital health tech company that uses artificial intelligence to provide users with personalized health information via a chatbot.

It was founded in Oslo, Norway in 2013 by Henrik Pettersen and now has headquarters in London, England. In June 2017, Your.MD raised $10 million in funding from its series A round, bringing total funding to $19 million. Its current investors include: Smedvig Capital AS, Orkla Group and a number of angel investors. [5]

Figure 2-3. Healthily logo

Key Features:

- **AI-Powered Symptom Checker (DOT):** An interactive chatbot that assesses user symptoms and provides potential causes and next steps.
- **Personalized Health Content:** Offers tailored health information and articles based on user profiles and interests.
- **Self-Care Guidance:** Provides actionable recommendations for managing minor health issues effectively.
- **Health Trackers:** Includes 28 pre-set trackers and the ability to create custom trackers for monitoring aspects like sleep, mood, activity, and pain.
- **Integration with Health Data:** Syncs with other health and fitness apps to provide a comprehensive overview of the user's health metrics.

Limitations

- Not a Substitute for Professional Medical Advice
- Limited Scope for Certain Populations
- Subscription Model
- User Experience Issues

2.3.4. WebMD

WebMD is an American corporation which publishes online news and information about human health and well-being. The WebMD website also includes information about drugs and is an important healthcare information website and the most popular consumer-oriented health site. [6]

WebMD was started in 1998 by internet entrepreneur Jeff Arnold. In early 1999, it was part of a three-way merger with Sapient Health Network (SHN) and Direct Medical Knowledge (DMK). SHN began in Portland, Oregon, in 1996 by Jim Kean, Bill Kelly, and Kris Nybakken, who worked together at a CD-ROM publishing firm, Creative Multimedia. Later, in 1999, WebMD merged with Healtheon, founded by Netscape Communications founder James H. Clark. [6]

Figure 2-4. WebMD logo

Key Features:

- **Symptom Checker**: Allows users to input symptoms and receive a list of potential conditions.
- **Comprehensive Health Library**: Offers medically reviewed articles on various health topics, including conditions, treatments, and wellness.
- **Medication Information**: Provides detailed information on drugs, including uses, side effects, and interactions.
- **Physician Directory**: Helps users locate nearby doctors, hospitals, and pharmacies based on their location.
- **Health Tools**: Includes tools like pill identifiers, medication reminders, and health trackers.

Limitations:

- **Lack of Personalization**: Information is generalized and may not account for individual health histories or contexts, potentially leading to misinterpretation.
- **Advertising Influence**: Reliance on advertising revenue, particularly from pharmaceutical companies, may raise concerns about content bias.
- **Information Overload**: The vast amount of available information can be overwhelming, making it challenging for users to find relevant content.

2.3.5. Ava Industries

Ava EMR is a modern solution designed to simplify and streamline medical clinic workflows. It is a platform that empowers healthcare professionals with seamless task management, efficient patient communication, and flexibility to work from anywhere.

Founded in 2016 by Dr. Mike Forseth and Dr. Matt Henschke, Ava EMR was created from their experience running a medical clinic in Calgary, Alberta. They identified a need for an EMR system that was intuitive, efficient, and designed to optimize clinic operations. Their goal was to develop a system that enhances connectivity, boosts efficiency, and elevates. patient care [7]

Figure 2-5. Ava Industries

Key Features:

- Scheduling, charting, billing, and clinic management tools.
- Emphasis on interoperability with existing systems.
- AI-powered medical scribe for documentation automation.

Limitations:

- As a relatively newer entrant, it may lack some advanced features found in more established systems.
- Limited adoption may result in fewer user reviews and community support resources.

2.4. Artificial Intelligence in Healthcare

Artificial Intelligence (AI) is rapidly transforming healthcare, offering powerful tools to enhance accuracy, efficiency, and patient-centered care. By leveraging techniques like Optical Character Recognition (OCR), Natural Language Processing (NLP), machine learning, and conversational interfaces, AI enables automated document processing, augmented clinical decision-making, and personalized patient interaction. This section explores these advancements and their implications for health systems.

2.4.1. Document Processing

AI-driven document processing uses OCR to digitize medical forms and handwritten notes, while NLP interprets and extracts critical information. For instance, medical organizations like Omega Healthcare partnered with UiPath to automate document processing—reducing administrative effort by 40% and speeding up workflows with 99.5% accuracy, saving over 15,000 work-hours monthly [8]

NLP systems streamline clinical documentation by extracting key data from unstructured notes—freeing clinicians from manual data entry that can take up to six hours per day. Furthermore, when integrated with clinical decision support systems (CDSS), NLP helps identify prescription hazards by flagging drug–drug interactions and unsafe dosages

2.4.2. Chatbots

A . Adoption of Chatbots in Healthcare Industry

The adoption of chatbots in the healthcare industry has been gradual, with varying levels of enthusiasm among stakeholders. According to the **2024 Generative AI in Healthcare Survey**, 35% of healthcare companies are not actively considering AI solutions, while 21% are exploring potential use cases. Interestingly, 19% are experimenting with AI models but haven't yet moved them into production. A smaller percentage, 14%, are in the early stages of adoption with a solution running in production, while 11% are in the mid-stage, with multiple AI-driven systems actively in use. [9]

Chatbots are among the leading applications of AI in healthcare, with 21% of companies focusing on answering patient questions, followed closely by 20% using chatbots for patient engagement, and 19% leveraging AI for information extraction and data management. These use cases highlight how AI technology streamlines communication and enhance patient care. [9]

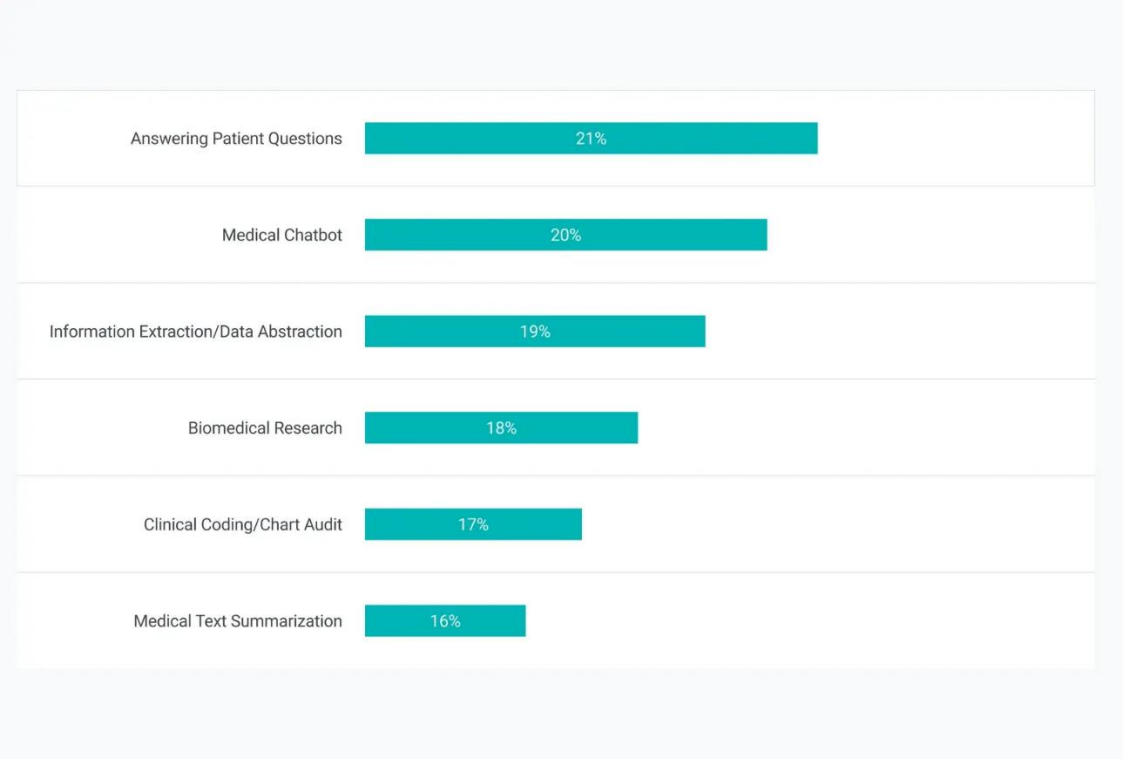

Figure 2-6. top Use Cases for AI in Healthcare

B . Chatbot Use Cases in Healthcare

From scheduling appointments to offering mental health support, these AI-powered chatbots tackle some of the most common healthcare challenges. Let's explore how they're reshaping care delivery across various use cases.

- Appointment Scheduling and Reminders
- Patient Triage and Symptom Checking
- Medication Management and Adherence
- Mental Health Support
- Chronic Disease Management
- Telehealth and Virtual Consultations

2.4.3. Clinical Decision Support Systems (CDSS)

A clinical decision support system (**CDSS**) is a health information technology that provides clinicians, staff, patients, and other individuals with knowledge and person-specific information to help health and health care. CDSS encompasses a variety of tools to enhance decision-making in the clinical workflow. These tools include computerized alerts and reminders to care providers and patients, clinical guidelines, condition-specific order sets, focused patient data reports and summaries, documentation templates, diagnostic support, and contextually relevant reference information, among other tools. CDSSs constitute a major topic in artificial intelligence in medicine. [10]

A . Real-World CDSS Example: SMASH Dashboard

A notable implementation is the **Salford Medication Safety Dashboard (SMASH)**, which leverages prescribing safety indicators within EHRs to identify and highlight patients at risk of medication-related harm across 43 general practices. The system, coupled with a pharmacist-led review model, achieved:

- ✓ A **40.7% reduction** in potentially hazardous prescribing over 12 months
- ✓ Sustained, consistent use across practices, with median monthly interactions of ~6–12 sessions

B . Integration of NLP in CDSS

Natural Language Processing (NLP) is increasingly integrated into CDSS to convert unstructured clinical narrative into actionable data. Reviews have shown that NLP systems can automatically extract adverse symptom patterns and abnormal lab trends, enabling early interventions and proactive alerts. Additionally, pharmacological CDSS use algorithms to flag potential allergies and drug–drug interactions, guiding safer prescribing decisions

2.4.4. Existing Appointment Management Systems (Web/Mobile)

Efficient appointment scheduling is critical for both patient access to care and clinic operational efficiency. Modern appointment systems have evolved significantly from manual phone-based booking.

Web-Based Portals: Many clinics and hospitals offer web portals (e.g., Zocdoc, Doctolib in some regions, or proprietary portals) where patients can search for doctors, view availability, and book appointments online. These often integrate with the clinic's practice management software.

Mobile Applications: Dedicated mobile apps provide on-the-go convenience for appointment booking, reminders, and management. They often include features like:

- ✓ Doctor search and filtering (by specialty, location, insurance).
- ✓ Real-time availability display.
- ✓ Automated reminders (SMS, push notifications).
- ✓ Options for rescheduling or canceling appointments.

- ✓ Integration with calendars.

A . Limitations of Current Systems:

- Lack of Intelligent Scheduling: Many systems still rely on simple slot-filling and may not optimize schedules for doctors or offer intelligent suggestions to patients based on urgency or doctor workload.
- Limited Integration with Pre/Post-Appointment Workflows: Appointment booking is often a standalone function, not deeply integrated with pre-visit form filling, post-visit follow-ups, or communication facilitated by other mHealth features.
- User Experience Inconsistencies: The user experience can vary widely, with some systems being clunky or difficult to navigate.
- Manual Confirmation Overheads: Some systems still require manual confirmation by clinic staff, reducing the benefits of automation.

The appointment system in the enhanced "Dawa2y" aims to overcome some of these limitations by providing a more integrated, intelligent, and user-friendly experience, potentially leveraging the chatbot for a conversational booking interface.

2.5. Image Recognition And Its Medical Usages

Recent advances in computer vision have introduced powerful tools for analyzing both textual and visual medical data. Technologies like Optical Character Recognition (OCR), image classification, and medical image segmentation have significantly improved the efficiency and accuracy of healthcare systems. This section presents key image-based techniques and their roles in medical contexts.

2.5.1. OCR and Its Medical Usage

Optical Character Recognition (OCR) is a technology that enables the automatic conversion of printed or handwritten text into machine-readable data. Initially developed for digitizing books and documents, OCR has evolved significantly with advancements in computer vision and deep learning, making it a powerful tool across various domains—including healthcare.

OCR helps in converting scanned images or pictures into machine-readable text characters, making it easier to extract and analyze data from medical documents. [11]

2.5.2. Image Classification for X-rays and MRI Scans

Image classification is used to automatically label medical images such as X-rays and MRIs. Deep learning models, particularly Convolutional Neural Networks (CNNs), are trained to detect patterns related to diseases like pneumonia, tumors, fractures, or brain abnormalities.

For example, chest X-rays can be automatically classified as normal or abnormal, flagging cases for faster review. Similarly, MRI scans can be analyzed to detect stroke regions, brain tumors, or degenerative conditions. Pretrained models like ResNet, DenseNet, and Vision Transformers (ViT) are widely used in medical imaging due to their accuracy and robustness.

These systems reduce diagnostic workload, assist radiologists, and increase early detection rates, especially in under-resourced hospitals.

2.5.3. Medical Image Segmentation

Beyond classification, image segmentation identifies and isolates specific anatomical structures or regions of interest within medical images. For instance, segmenting a tumor in an MRI helps in tracking its size and location over time. U-Net and its variants are the dominant architectures in this domain due to their ability to work well even with limited annotated data.

Segmentation is critical for surgical planning, radiation therapy, and monitoring treatment progress. Combined with classification systems, segmentation improves the precision of computer-aided diagnosis.

2.6. Retrieval-Augmented Generation

Retrieval-Augmented Generation (RAG) is the process of optimizing the output of a large language model, so it references an authoritative knowledge base outside of its training data sources before generating a response. Large Language Models (LLMs) are trained on vast volumes of data and use billions of parameters to generate original output for tasks like answering questions, translating languages, and completing sentences. RAG extends the already powerful capabilities of LLMs to specific domains or an organization's internal knowledge base, all without the need to retrain the model. It is a cost-effective approach to improving LLM output so it remains relevant, accurate, and useful in various contexts. [12]

2.6.1. RAG Vs Fine Tuning

Retrieval-Augmented Generation (RAG) and **fine-tuning** are two different approaches to adapting large language models to specific tasks or domains. Fine-tuning involves updating the model's internal weights using domain-specific data, which can improve fluency and performance for specific tasks. However, it's resource-intensive, harder to update, and often acts like a black box making it difficult to trace or control what the model has "learned."

RAG takes a different path. Instead of changing the model itself, it keeps the model frozen and retrieves relevant information from an external knowledge base during each query. The model then uses this retrieved data as context to generate a response. This makes RAG easier to update, more transparent, and safer for use cases like healthcare, where facts must be reliable and patient-specific.

A . When should I use RAG and when should I fine-tune the model?

RAG is the right place to start, being easy and possibly entirely sufficient for some use cases. Fine-tuning is most appropriate in a different situation, when one wants the LLM's behavior to change, or to learn a different "language." These are not mutually exclusive. As a future step, it's possible to consider fine-tuning a model to better understand domain language and the desired output form — and also use RAG to improve the quality and relevance of the response. [13]

2.6.2. How RAG Works

When a user asks a question to a language model, the system first transforms the question into a numerical format that machines can understand. This numerical representation, known as an embedding or vector, captures the meaning of the query in a way that allows the model to process and compare it efficiently.

The embedding model then compares these numeric values to vectors in a machine-readable index of an available knowledge base. When it finds a match or multiple matches, it retrieves the related data, converts it to human-readable words and passes it back to the LLM.

Finally, the LLM combines the retrieved words and its own response to the query into a final answer it presents to the user, potentially citing sources the embedding model found. [14]

2.6.3. Benefits of RAG on LLMs

A . Up-to-date and accurate responses

RAG ensures that an LLM's response is not limited to outdated training data by retrieving relevant, up-to-date information from external sources at the time of the query. This improves accuracy, relevance, and adaptability, especially in fields like healthcare where current information is essential.

B . Providing domain-specific, relevant responses

Using RAG, the LLM will be able to provide contextually relevant responses tailored to an organization's proprietary or domain-specific data by retrieving and incorporating information from a custom knowledge base during inference. This approach enables the model to generate answers that are not only accurate and up-to-date but also aligned with the organization's unique context, terminology, and informational needs, without the need to retrain the model itself.

In medical assistant chatbots case, RAG can help by applying variance in responses and less generalization, so the responses will be a bit various and different from a patient to another,

depending on their health status, history, diseases they have and medicines they are currently consuming.

A . Being efficient and cost-effective

Compared to other approaches to customizing LLMs with domain-specific data, RAG is simple and cost-effective. Organizations can deploy RAG without needing to customize the model. This is especially beneficial when models need to be updated frequently with new data.

2.7. Conclusion

The literature review reveals that mobile health (mHealth) and electronic health (eHealth) platforms, such as Doctolib, NHS App, Healthily, WebMD, and Ava EMR, have significantly advanced healthcare delivery by improving access, patient engagement, and administrative efficiency. However, these systems face limitations, including lack of personalization, connectivity issues, and limited AI integration. Artificial Intelligence, particularly in document processing, chatbots, and clinical decision support systems, offers transformative potential for automating workflows and enhancing decision-making, yet challenges like data accuracy, user adherence, and regulatory compliance persist. Current appointment management systems often lack intelligent scheduling and integration with broader healthcare workflows. The enhanced "Dawa2y" application addresses these gaps by integrating AI-driven document analysis, an intelligent chatbot, and a streamlined appointment system, aiming to provide a more cohesive, efficient, and user-centric healthcare solution.

Chapter 3 : Conceptual Study

3.1. Introduction

The realization of a platform must be imperatively preceded by an analysis and design methodology which aims to formalize the preliminary stages of the development of a site in order to make this development more faithful to the needs of the customer. The present chapter is the subject of the implementation of the conceptual study of the proposed platform.

This chapter will present the comprehensive design of the enhanced "Dawa2y" application, detailing its overall system architecture which now incorporates advanced AI services for document understanding and chatbot interactions, alongside a new appointment management module. We will utilize the Unified Modeling Language (UML) as a standard graphical notation to illustrate various aspects of the system, including actor interactions through Use Case diagrams, the static structure via Class diagrams, and dynamic behaviors through Sequence and Activity diagrams.

3.2. Definition and History of UML

The analysis phase makes it possible to list the expected results, in terms of functionalities and the design phase makes it possible to describe in an unambiguous way, most often using a modeling language, the operation, the future operation of the system, in order to facilitate its realization, the following sections describe the UML modeling language [15].

Figure 3-1. UML Logo

The UML language (Unified Modeling Language) is a modeling language that aims to facilitate transactions during the development of a project, from the original need to the implementation phase. This language is defined as a graphical and textual modeling language intended to understand and define needs, specify and document systems, sketch software architectures, design solutions and communicate points of view. UML models all the data and the treatments by elaborating different diagrams [15].

3.3. Project presentation

Dawa2y is a platform that combines both a mobile and a web application, built to modernize and automate several traditional healthcare processes. The application brings together multiple stakeholders of the healthcare system—**patients**, **doctors**, **pharmacies**, and **administrators**—on one unified platform. The application is provided with multiple services like notifications, and real-time database that makes it interactive with the users, and we lately added new AI-powered features that help patients and professionals.

The project is divided into several integrated modules:

- **AI Document Aid and Scanner**: Assists doctors in writing and reading prescriptions, using Optical Character Recognition (OCR) and Natural Language Processing (NLP) techniques.
- **Patient Chatbot**: Provides 24/7 support for patient inquiries, medication reminders, symptom-based suggestions, and appointment guidance.
- **Appointment System**: Enables real-time scheduling and management of doctor appointments, with notifications and availability tracking.
- **Mobile Patient Application (Android)**: Offers user-friendly features like QR scanning, reminders, chat interface, and medical history.
- **Doctor Web Interface**: Designed for creating prescriptions, accessing patient records, and managing incoming appointment requests.
- **Admin and Pharmacy Panels**: Allow system monitoring, user validation, and pharmacy stock tracking based on prescriptions.

3.3.1. General objectives of Dawa2y

The main objective of this project is to design and develop a comprehensive medical application—**Dawa2y**—that leverages modern technologies, including **Artificial Intelligence**, to improve healthcare delivery and communication between patients, doctors, and pharmacies.

More specifically, the project aims to:

- ✓ Provide a **secure and centralized digital platform** for medical prescriptions and healthcare interactions.
- ✓ Assist **doctors** in creating and managing prescriptions using **AI-based document scanning and aid tools**.
- ✓ Empower **patients** by offering access to personalized medical information, medication reminders, and a smart **chatbot** for health-related questions.
- ✓ Facilitate **appointment booking and management**, allowing patients to schedule visits and doctors to organize their agendas efficiently.

- ✓ Enhance **pharmacy services** by enabling access to validated prescriptions and availability of medications.

3.4. Analysis and Conception

3.4.1. Use case Diagram

The Use Case Diagram (UCD) is a UML diagram used to provide an overview of the functional behavior of a software system. It illustrates the interactions between actors (users or external systems) and the system, showcasing the various use cases and their relationships.

A . Roles of Use case Diagram

The main purpose of a use case diagram is to portray the dynamic aspect of a system. It accumulates the system's requirement, which includes both internal as well as external influences. It invokes persons, use cases, and several things that invoke the actors and elements accountable for the implementation of use case diagrams. It represents how an entity from the external environment can interact with a part of the system [16].

The following are the purposes of a use case diagram given below:

- ✓ It gathers the system's needs.
- ✓ It depicts the external view of the system.
- ✓ It recognizes the internal as well as external factors that influence the system.
- ✓ It represents the interaction between the actors.

B . Identification of actors in our application

- Patient
- Doctor
- Pharmacy
- Admin
- AI-Agent

C . Actors of our application

Patient: A user who interacts with the system to manage their health and medications.

His use cases:

- Authentication.
- Get notification for medication (Secondary Actor).
- Display prescription: Access to all prescriptions.
- Create costume alarm: With custom title and time.
- Navigate medical library: Search about Medicines and get detailed information about them.
- Scan prescription to show more details : Scan QR code of the prescription to show its details.
- Navigate doctors profiles: Get doctors contact information.
- Navigate the coming tasks: Get remaining time of the current medications.
- Navigate orders: Get states of orders of medications in pharmacies.

Doctor: A healthcare provider who manages patients' profiles and treatment plans.

His use cases:

- Authentication.
- Create patient profile
- Verify patient status and history.
- Add new documents to patient profile (blood tests, radiology images…).
- Create new prescription.
- Request Ai document analyses.

Pharmacy: A medicine provider responsible for processing prescriptions.

His use cases:

- Authentication.
- Scan prescription: Get the ID of the prescription to show its details.
- Activate the medicines in the prescriptions.
- Make a command for the medicine.

Admin: A system administrator who manages platform access for other actors.

His use cases:

- Authentication.
- create doctors and pharmacies accounts.

AI Agent: An intelligent assistant that enhances user experience with smart features.

His use cases:

- Analyze patient documents and generate reports.
- Interact with patients via chatbot for guidance.
- Check for medicine interactions.
- Book appointments for patients when needed.

D . Use case diagram in our application

The use case diagram in our application (android/web) is structured as follow:

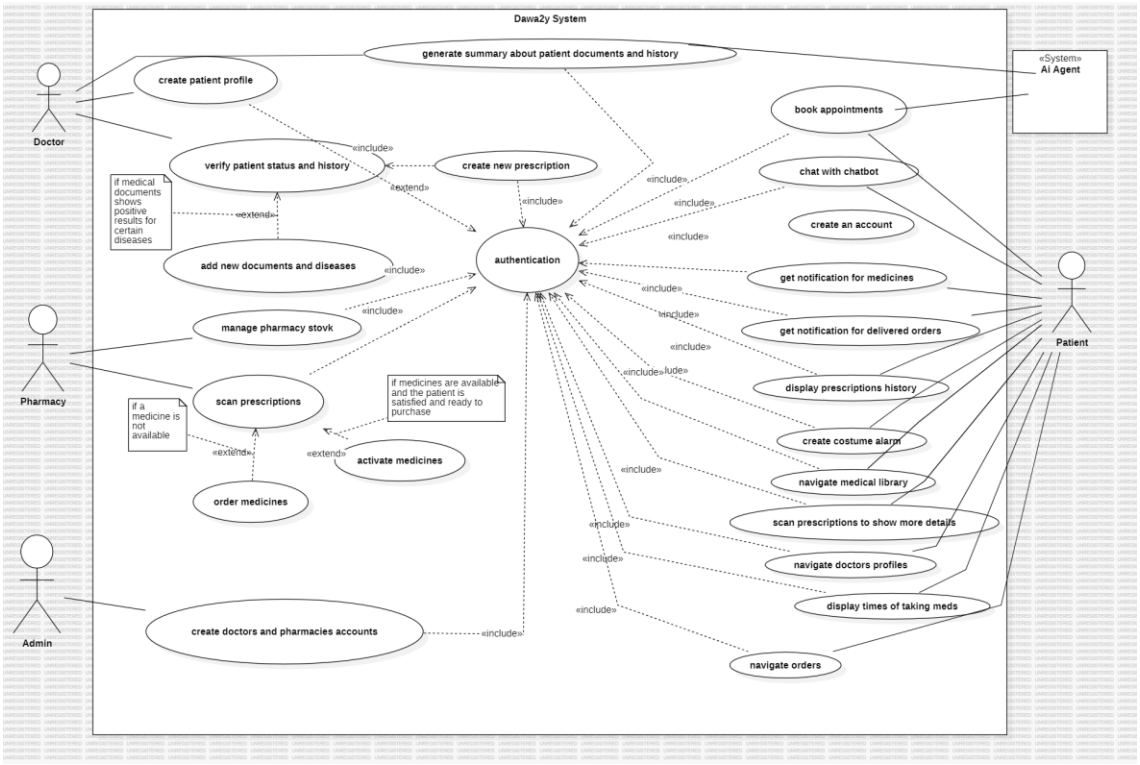

Figure 3-2 : Use case diagram of "Dawa2y" application (web/android)

E . Textual Description of use cases:

a. Use case "create an account"

Identification

- **Use case name:** create a patient account.
- **Goal:** creation of a new patient account.
- **Actors:** Patient.

Sequencing

- **Preconditions:**
  1. Authentication.
  2. User should have an Identity card.
  3. Internet connection.

- **Main Flow:**
  1. The user enters the personnel information including name, email, ID card number...
  2. The app will verify the integrity of the inputs and also verify the existence of the user in the database.
  3. If the app found that the user already exists in the database then it will notify him, and the user will choose to login instead.
  4. If the app found that the user does not exist in the database then it will create a user account in Firebase authentication and a patient node in real-time database.
- **Postconditions:**
  1. Patient account is created successfully.

This table briefly explain the use case "create a patient account":

b. Use case "create a new Prescription"

This table briefly explain the use case "create a new Prescription":

Identification

- **Use case name :** create new Prescription.
- **Goal :** creation of a new Prescription.
- **Actors :** Doctor.

Sequencing

- **Preconditions:**
  1. Authentication.
  2. verify patient status and history.

- **Main Flow:**
  1. The doctor search for the patient profile.
  2. The doctor set the date of the Prescription and the duration of the medication.
  3. The doctor add medicines to the Prescription and for each medicine several medication tasks (hour of day and an optional annotation).
  4. If the website detects that there's a medicine in the Prescription that can not be used or may cause some problems if the patient uses it with another medicine that he's currently consuming, or a medicine that is forbidden for the patient due to a certain disease, the website will alert the doctor.
  5. If Prescription meets all required conditions, it will be added to the patient database.
- **Postconditions:**
  1. Prescription is created successfully.

c. Use case "activate the medicines in the Prescription"

This table briefly explain the use case "activate the medicines in the Prescription":

Identification

- **Use case name :** activate the medicines in the Prescription.
- **Goal :** activation of the medicines in a Prescription.
- **Actors :** Pharmacy.

Sequencing

- **Preconditions:**
  1. QR Code of the Prescription or its ID.
  2. The medicines in the Prescription are not activated already or expired.

- **Main Flow:**
  1. The pharmacist enters the ID code of the Prescription in the website or scan the Prescription QR Code.
  2. Activate the medicines.
  3. The activation date will be added to the attributes of the Prescription in the database.
- **Postconditions:**
  1. Medicines are activated.
  2. Tasks appears to the patient in his android application.

d. Use case "make a command for the medicine"

This table briefly explain the use case "make a command for the medicine":

Identification

- **Use case name :** make a command for the medicine.
- **Goal :** adding a command for a medicine, so the patient will be notified when the medicine is available in the pharmacy.
- **Actors :** Pharmacy.

Sequencing

- **Preconditions:**
  1. Authentication
  2. QR Code of the Prescription or its ID.
  3. The medicine is not available in the pharmacy.

- **Main Flow:**
  1. The pharmacist enters the ID code of the Prescription in the website or scan the Prescription QR Code.
  2. Add an order for the medicine with a specific date to notify the patient for the availability of the medicine in the pharmacy.
- **Postconditions:**
  1. Add the order to the patient profile.
  2. Patient get a notification when the medicine is supposed to be available.

e. Use case "get notification for medication"

This table briefly explain the use case "get notification for medication":

Identification

- **Use case name :** get notification for medication.
- **Goal :** notify the patient to take his medication in time.
- **Actors :** Patient (Secondary).

Sequencing

- **Preconditions:** none

- **Main Flow:**
    1. When the specific time of the medication task comes, an alarm will be triggered showing a notification for the patient in the application.
- **Postconditions:**
    1. Notification displayed with the annotation of the doctor.

f. Use case "create doctors and pharmacies accounts"

This table briefly explain the use case "create doctors and pharmacies accounts":

Identification

- **Use case name :** create doctors and pharmacies accounts.
- **Goal :** creating a doctor or a pharmacy account .
- **Actors :** Admine.

Sequencing

- **Preconditions:**
  1. The admine should have the email and the password of the email linked to the Firebase database of "Dawa2y" application.
  2. The doctor or the pharmacy should provide the admine with all required information.

- **Main Flow:**
  1. The admine creates a user account in Firebase authentication for the doctor or the pharmacy with the email and a default password.
  2. The admine creates a new reference in Firebase real-time database for the doctor or the pharmacy with all the required information of the new user.
- **Postconditions:**
  1. New reference in the database representing the doctor or the pharmacy.
     The new

g. Use case "generate a report about patient documents and history"

This table briefly explain the use case "generate a report about patient documents and history":

Identification

- **Use case name :** generate a report about patient documents and history.
- **Goal :** provide the doctor with summary about patient's documents to give him an overview for continuity of care.
- **Actors :** Doctor, Ai agent.

Sequencing

- **Preconditions:**
  1. The doctor should have access to the Ai features.
  2. The patient should already have documents in his account such as blood tests and x-ray images.

- **Main Flow:**
  1. The doctor request a report.
  2. The Ai agent uses the OCR to read the content of the documents.
  3. Transform the content to an LLM that analyse the documents and returns a report.
- **Postconditions:**
  1. A report is generated and displayed on the Doctors app.

h. Use case "book appointments"

This table briefly explain the use case "book appointments":

Identification

- **Use case name :** book appointments.
- **Goal :** book appointments for patients with doctors.
- **Actors :** Patient, Ai agent.

Sequencing

- **Preconditions:**
  1. Authntication.

- **Main Flow:**
  1. The patient will choose between the manual booking or booking with the help of the Ai agent.
  2. The patient will explore the available days and hours.
  3. The patient can choose any free time he would like.
- **Postconditions:**
  1. Appointment booked.

3.4.2. Sequence Diagram

A . Purpose of sequence diagram

A sequence diagram helps us see how different parts of a system like users or components interact with each other and in what order. It is useful for understanding how the system behaves, and it also helps us check if the system meets its requirements correctly.

B . Sequence diagram for "Authentication"

Authentication is a fundamental component of any application. It ensures that only authorized individuals can access, view, or modify sensitive information. As such, authentication serves as a critical layer in maintaining the overall security and integrity of data within a system.

Firebase Authentication provides backend services, easy-to-use SDKs, and ready-made UI libraries to authenticate users to your app. It supports authentication using passwords, phone numbers, popular federated identity providers like Google, Facebook and Twitter, and more. [17]

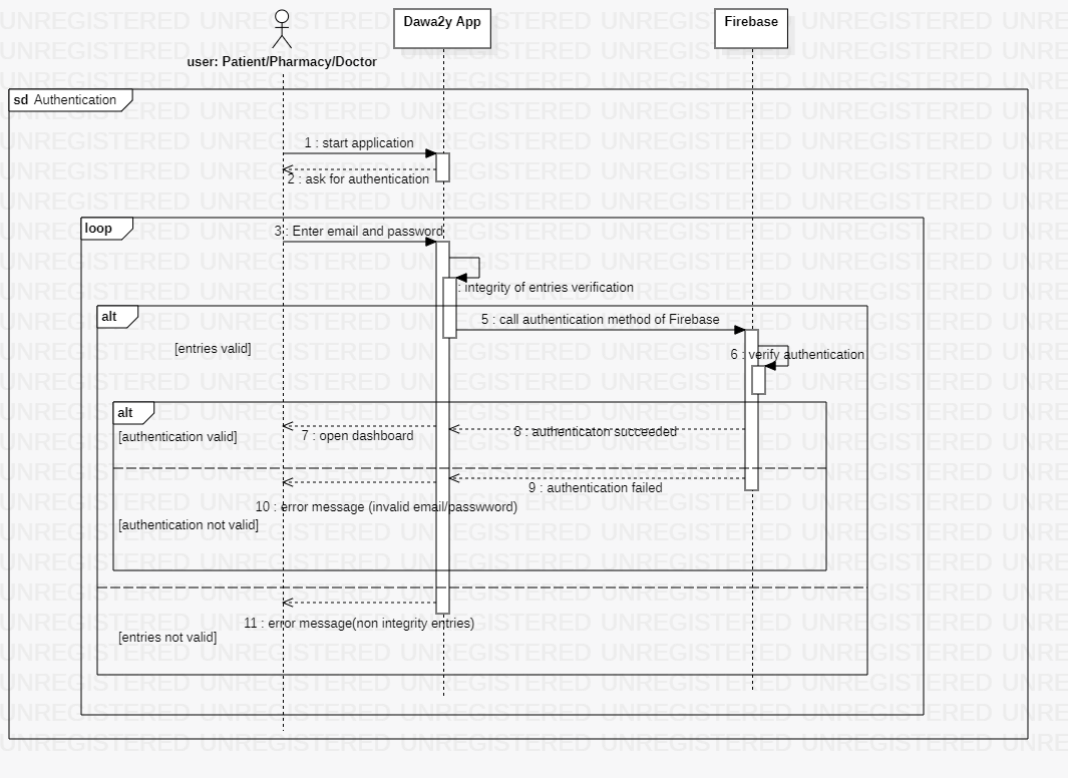

Figure 3-3 : Sequence diagram for Authentication

C . Sequence diagram for "Create patient account"

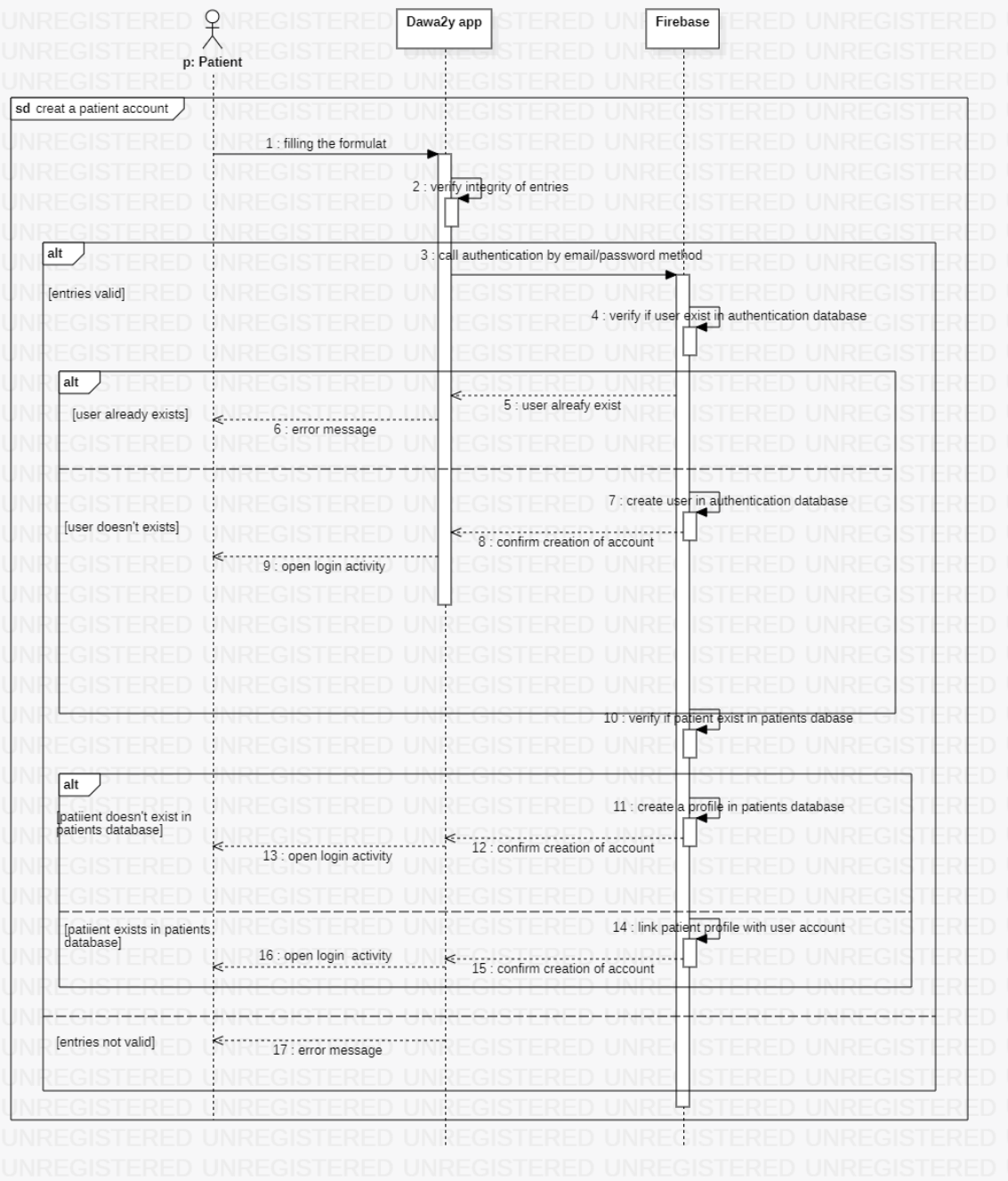

Figure 3-4 : Sequence diagram for creating an account

D . Sequence diagram for "Create new Prescription"

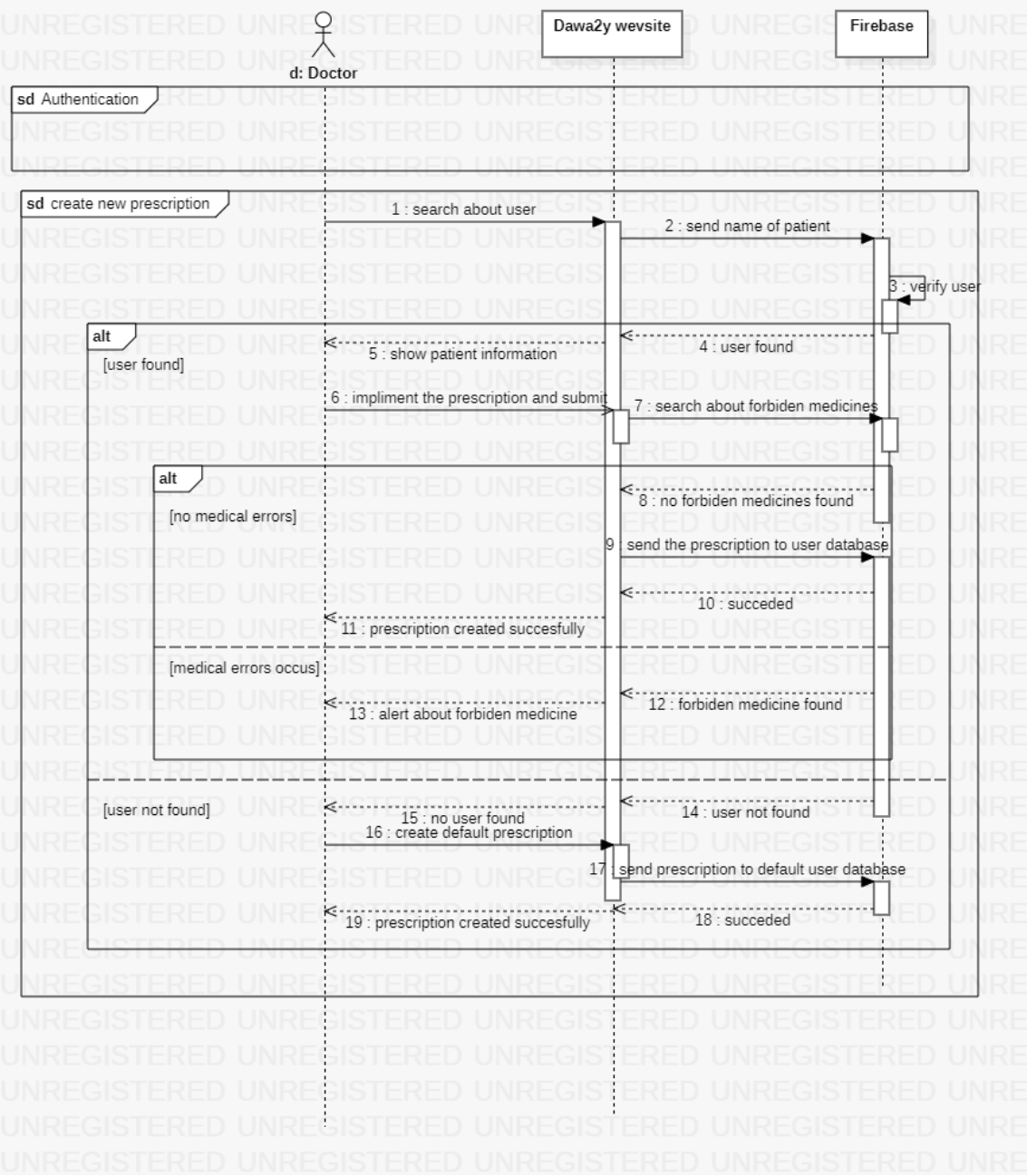

Figure 3-5 : Sequence diagram for creation of a new Prescription

E . Sequence diagram for "Activate a medicine in the Prescription"

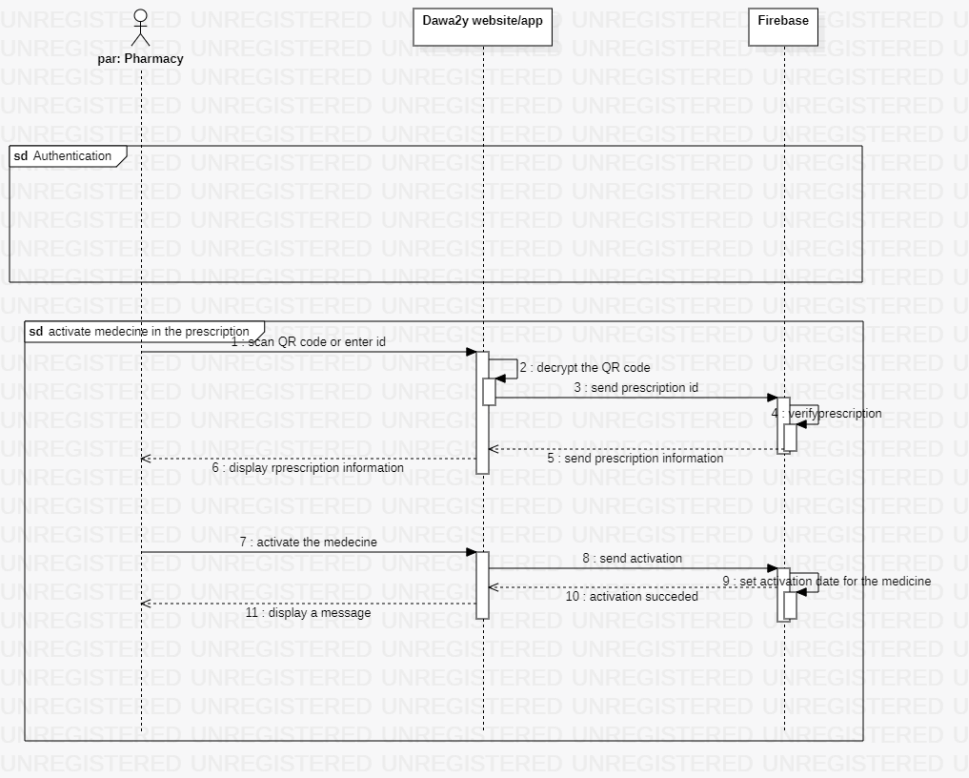

Figure 3-6 : Sequence diagram for activating of a medicine

F . Sequence diagram for "Order a medicine"

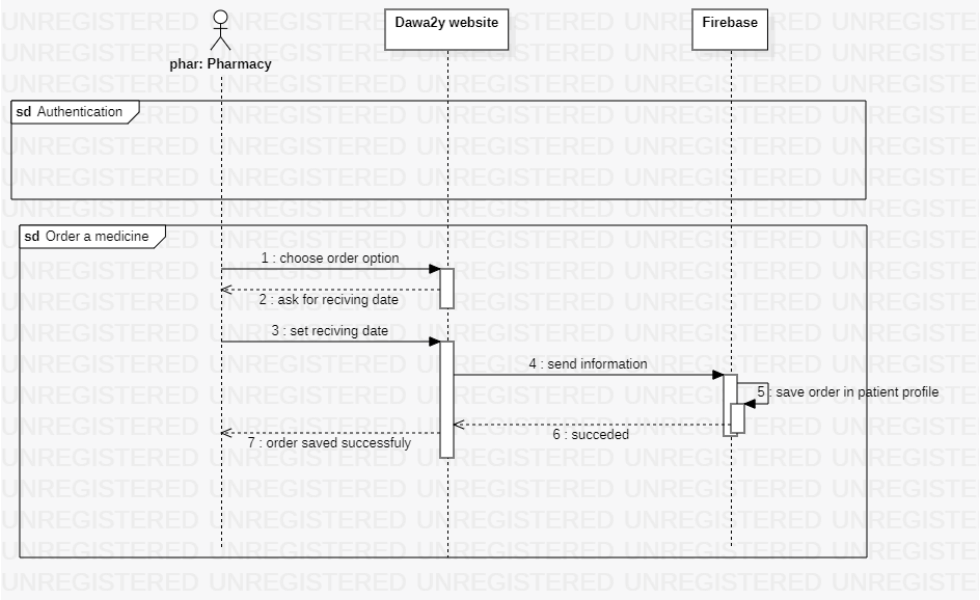

Figure 3-7 : Sequence diagram for ordering a medicine

G . Sequence diagram for "Create a doctor/pharmacy account"

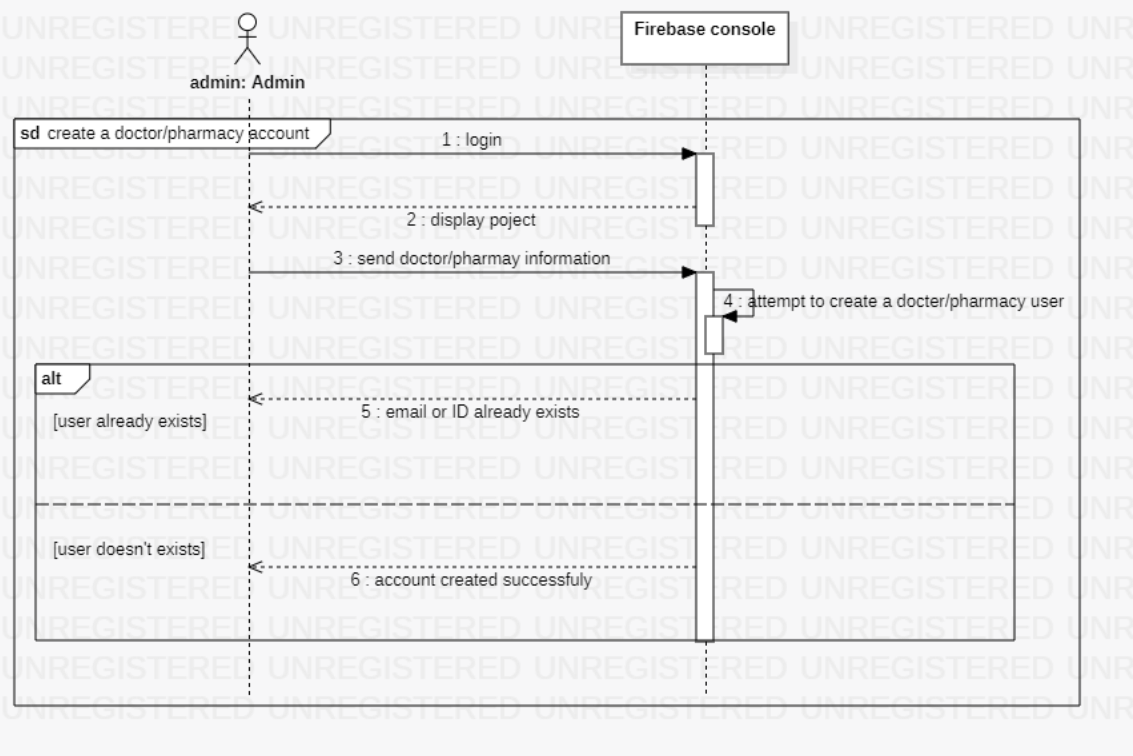

Figure 3-8 : Sequence diagram for creating an account for a doctor or pharmacy

H . Sequence diagram for "Generate a patient report"

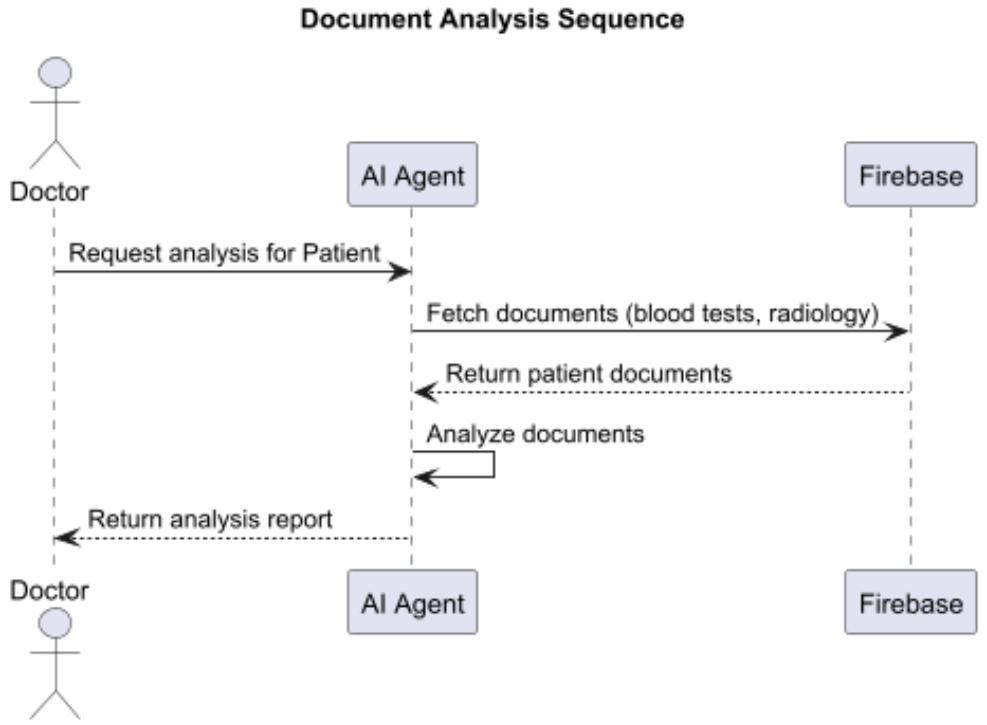

Figure 3-9. Sequence diagram for generate a patient report

A . Sequence diagram for "Generate a chatbot assistant message"

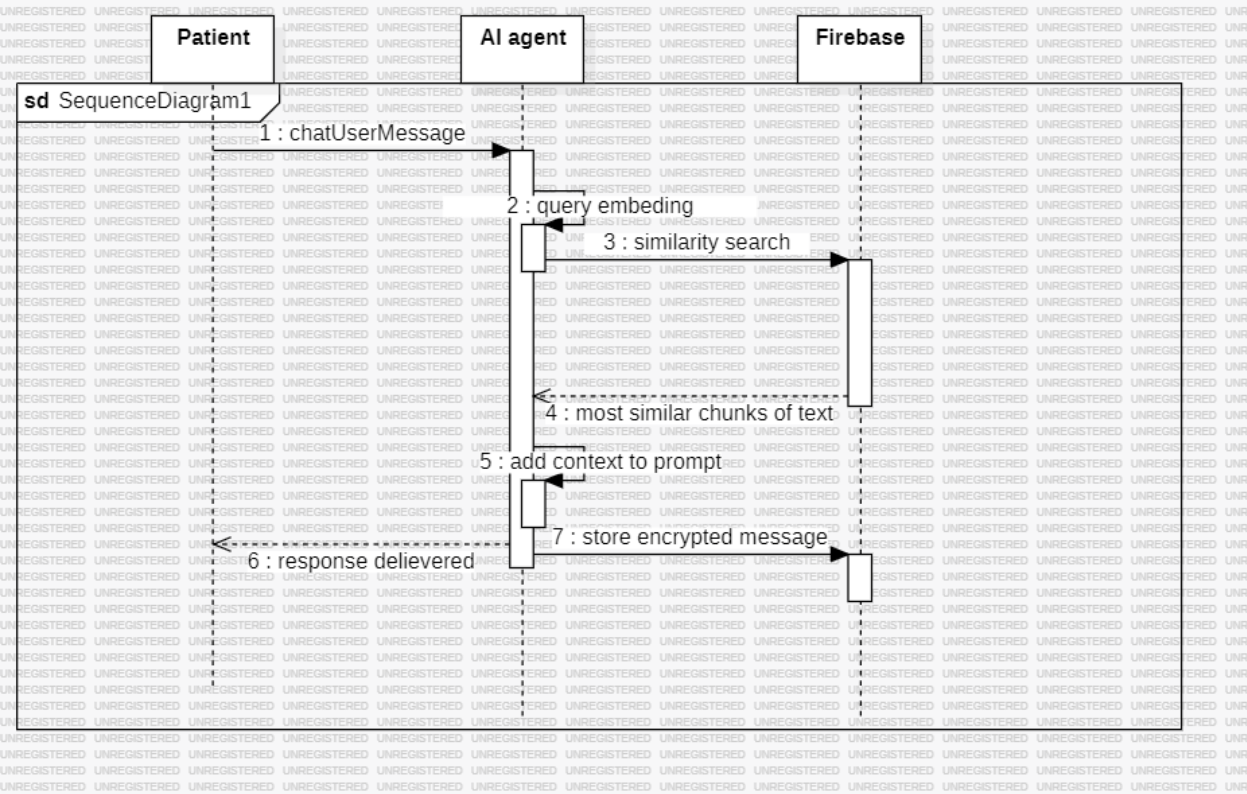

Figure 3-10. Sequence diagram for generate a chatbot assistant message

3.4.3. Class Diagram

Class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects. [18]

A . Purpose of class diagram

- Modeling the structure: Class diagrams depict the static structure of a system, showing classes, their attributes (data), and methods (operations). They provide an overview of the system's components and their relationships, serving as a blueprint for the implementation.
- Identifying classes and their relationships: Class diagrams help identify the classes needed in a system and their associations, aggregations, inheritances, and dependencies. This helps in understanding the entities involved and how they interact with each other.
- Visualizing the system's architecture: Class diagrams provide a high-level view of the system's architecture, including its major components and their relationships. They help in understanding the system's organization and the interactions between different parts.

B . Identification of classes of class diagram in our application

Person:

- **Attributes**: id, name, email, phoneNumber.
- **Methods**: login(), logout.

Patient (inherits from Person):

- **Attributes**: birthDate , sex , insurance, diseaseList.
- **Methods**: checkLibrary(), scanQrCode(), manageAlarms(), checkDoctors(), registr().

Doctor (inherits from Person) :

- **Attributes**: speciality ,adress ,location.
- **Methods**: createRecipe(), consultePatient(),checkHistory().

Pharmacy (inherits from Person):

- **Attributes**: adress.
- **Methods**: order(), activateMed().

Admine (inherits from Person):

- **Attributes**: none.
- **Methods**: createUserAccount(), updateAccount(), deleteAccount().

Ai Agent:

- **Attributes**: .
- **Methods**: createUserAccount(), updateAccount(), deleteAccount().

Alarm:

- **Attributes**: id, title, isActive, time.
- **Methods**: notify().

Order:

- **Attributes**: id, orderDate, deliveryDate, medName.
- **Methods**: none.

Prescription:

- **Attributes**: id, qrCode, duration, drName, date.
- **Methods**: none.

Medicine:

- **Attributes**: name, laboName, forbMeds, medicalPaper, INN, dose, form.
- **Methods**: none.

Task:

- **Attributes**: id, sequence, times, notes.
- **Methods**: notify(), incrementNextDate().

C . Class diagram for "Dawa2y" application (web/android)

After a detailed study of the use cases, we have derived the overall class diagram of the system, This diagram is considered the final stage of the theoretical design of our system and will serve as a reference for the software development process and the writing of the source code of our application.

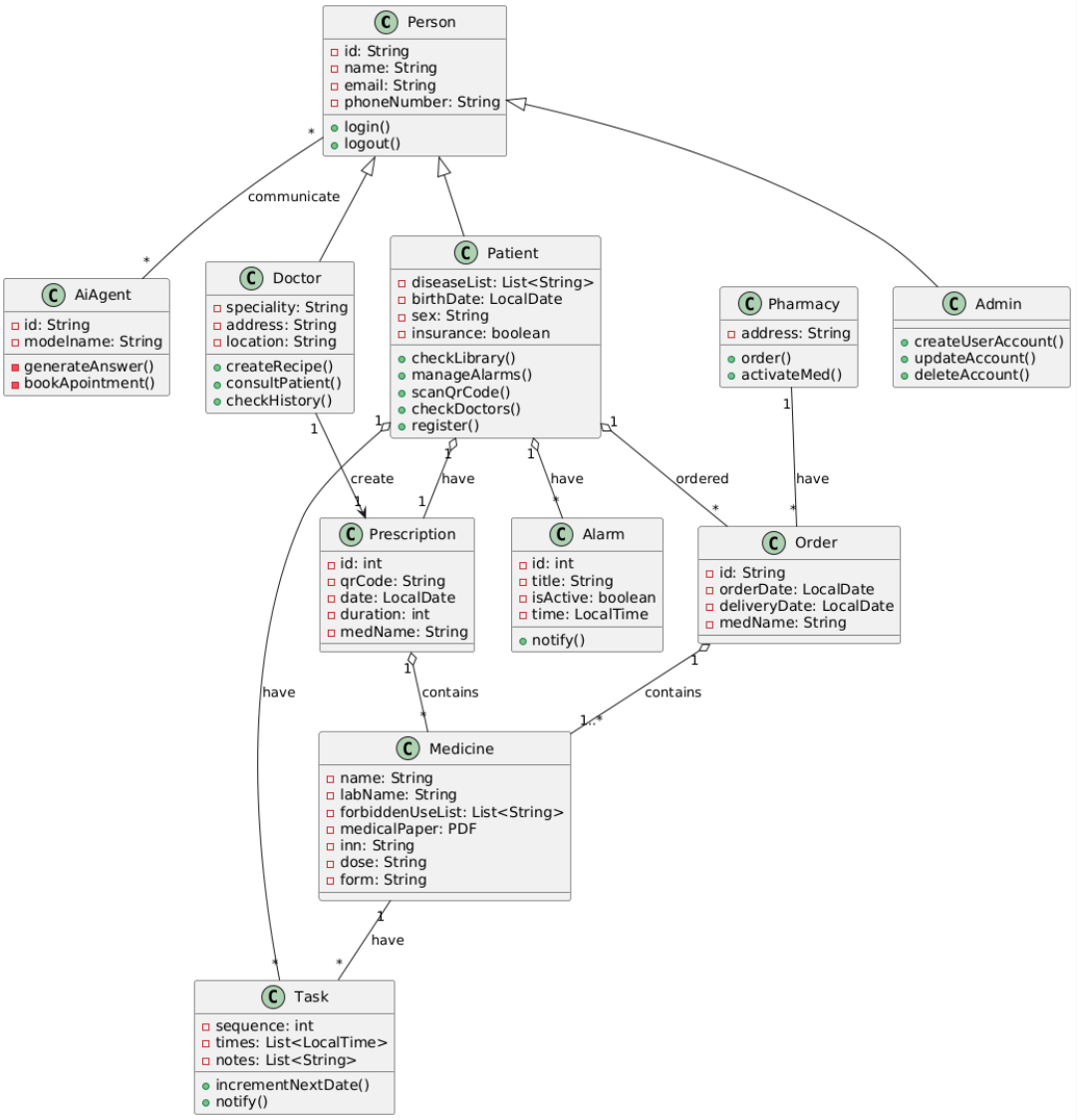

Figure 3-11 : Class Diagram of Dawa2y application

3.4.4. Activity Diagram

Activity Diagrams describe how activities are coordinated to provide a service which can be at different levels of abstraction. Typically, an event needs to be achieved by some operations, particularly where the operation is intended to achieve a number of different things that require coordination, or how the events in a single use case relate to one another, in particular, use cases where activities may overlap and require coordination. It is also suitable for modeling how a collection of use cases coordinate to represent business workflows [19].

An activity edge is a sequencing relationship between a source activity node and a target activity node. The target node cannot execute until the source node has completed execution and emitted a token onto the activity edge. If a node has multiple edges of which it is the target, it cannot execute until all of them have tokens, unless the rules for the particular kind of activity node specify that a subset of input edges may enable execution. An edge may represent simple flow of control (control flow) or it may represent the flow of data (data flow), including the implicit flow of control that indicates that the data value has been produced [20].

A . Activity diagram for "Authentication" in our application

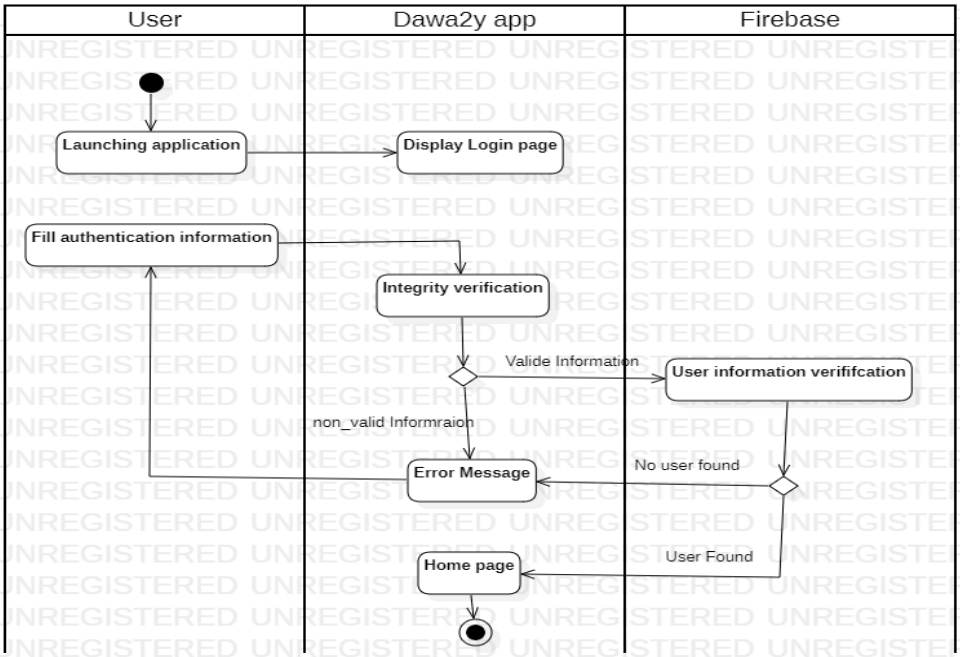

Figure 3-12 : Activity diagram for Authentication

B . Activity diagram for "Registration" in our application

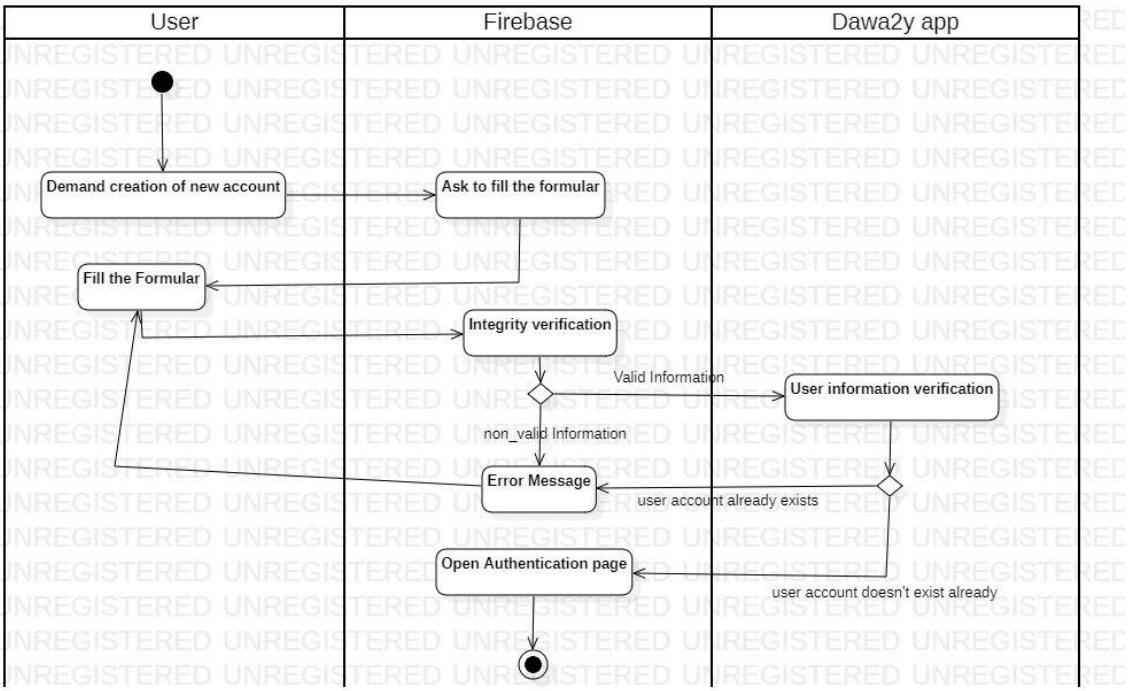

Figure 3-13 : Activity diagram for Registration

3.5. Conclusion

In this chapter, we have addressed the analysis and design phase, which is carried out through various UML diagrams such as use case diagrams, class diagrams, sequence diagrams, and activity diagrams. These modeling tools have allowed us to clearly define the system's functional requirements, visualize the interaction between different components, and ensure coherence between the user's needs and the system's behavior.

By translating real-world requirements into abstract representations, this phase helps bridge the gap between the problem domain and the technical implementation. The use of UML not only improves communication between stakeholders but also lays a solid foundation for the development phase by reducing ambiguities and enhancing system understanding.

Through this structured approach, we ensure that the system is well-conceived, scalable, and aligned with the intended objectives.

Chapter 4 : Implementation and Interface

4.1. Introduction

Every research work consists of two parts: a theoretical part explained in the thesis, and the practical part.

In this chapter, we will briefly present the structure of our web application, the programming languages, and the tools used for its development, as well as the description of different interfaces.

4.2. Work environment

4.2.1. Software environment and tools

A . Visual Studio Code

Visual Studio Code is a lightweight but powerful source code editor that runs on your desktop and is available for Windows, macOS, and Linux. It comes with built-in support for JavaScript, TypeScript, and Node.js, and has a rich ecosystem of extensions for other languages (such as C++, C#, Python, Java) and runtime environments. [21]

B . Firebase

Firebase is a comprehensive platform-as-a-service (PaaS) that offers a range of cloud-based services and tools to aid developers in building, testing, and deploying mobile and web applications. It provides features such as real-time database, authentication, hosting, cloud functions, cloud messaging, and more, which enable developers to focus on app development without worrying about server infrastructure and backend management [22]

C . Android SDK

Android SDK is a collection of libraries and Software Development tools that are essential for Developing Android Applications. Whenever Google releases a new version or update of

Android Software, a corresponding SDK also releases with it. In the updated or new version of SDK, some more features are included which are not present in the previous version. Android SDK consists of some tools which are very essential for the development of Android Application. These tools provide a smooth flow of the development process from developing and debugging. Android SDK is compatible with all operating systems such as Windows, Linux, macOS, etc. [23]

D . Node Package Manager (NPM)

npm stands for Node Package Manager. It's a library and registry for JavaScript software packages. npm also has command-line tools to help you install the different packages and manage their dependencies. npm is free and relied on by over 11 million developers worldwide. You could say it's kind of a big deal. They're open-source and have become the center of JavaScript code sharing. There are more than a million packages available on npm. [24]

E . React

React (also known as React.js or ReactJS) is a free and open-source front-end JavaScript library that aims to make building user interfaces based on components more "seamless". It is maintained by Meta (formerly Facebook) and a community of individual developers and companies. [25]

React can be used to develop single-page, mobile, or server-rendered applications with frameworks like Next.js and Remix. Because React is only concerned with the user interface and rendering components to the DOM, React applications often rely on libraries for routing and other client-side functionality. A key advantage of React is that it only re-renders those parts of the page that have changed, avoiding unnecessary re-rendering of unchanged DOM elements. [25]

F . Tailwind CSS

Tailwind CSS is a utility-first CSS framework that simplifies web development by providing a set of pre-designed utility classes. These utility classes enable you to build custom designs without writing any custom CSS, promoting consistency, scalability, and efficiency. [26]

Tailwind shifts the focus from traditional CSS components to functional utility classes, empowering you to build responsive and visually appealing interfaces with ease and speed. [26]

4.2.2. Programming Languages Used

A . HTML

HTML (Hypertext Markup Language) is the most basic building block of the Web. It defines the meaning and structure of web content. Other technologies besides HTML are generally used to describe a web page's appearance/presentation (CSS) or functionality/behavior (JavaScript).

"Hypertext" refers to links that connect web pages to one another, either within a single website or between websites. [27]

B . CSS

Cascading Style Sheets (CSS) is a stylesheet language used to describe the presentation of a document written in HTML or XML (including XML dialects such as SVG, MathML or XHTML). CSS describes how elements should be rendered on screen, on paper, in speech, or on other media. [28]

C . JavaScript

JavaScript (JS) is a lightweight, interpreted, or just-in-time compiled programming language with first-class functions. While it is most well-known as the scripting language for Web pages, many non-browser environments also use it, such as Node.js, Apache CouchDB and Adobe

Acrobat. JavaScript is a prototype-based, multi-paradigm, single-threaded, dynamic language, supporting object-oriented, imperative, and declarative (e.g. functional programming) styles. [29]

D . Java

Java is a programming language and computing platform first released by Sun Microsystems in 1995. It is the underlying technology that powers state-of-the-art programs, including utilities, games, and business applications. Java runs on more than 3 billion devices worldwide, making it one of the most popular programming languages. [30]

E . Tesseract

Tesseract is an open source text recognition (OCR) Engine, can be used directly via command line, or (for programmers) by using an API to extract printed text from images. It supports a wide variety of languages. Tesseract doesn't have a built-in GUI, but there are several available from the 3rdParty page. External tools, wrappers and training projects for Tesseract are listed under AddOns. [31]

F . Azure

Azure is Microsoft's cloud computing platform, offering a massive range of hosted services including compute, storage, databases, networking, and critically for us, AI and machine learning tools.

G . N8N

n8n is an extendable workflow automation tool. With a fair-code distribution model, n8n will always have visible source code, be available to self-host, and allow you to add your own custom functions, logic and apps. n8n's node-based approach makes it highly versatile, enabling you to connect anything to everything. [32]

4.3. Interfaces of our application

The graphical interfaces of the application are very important, because they allow to facilitate the dialogue between man and machine as well as to improve the performance of the application.

In this part we present the main functionalities of our application by describing some interfaces From the website and the android application

4.3.1. Website interfaces

A . Doctors Interfaces

- **Doctors login Interface**

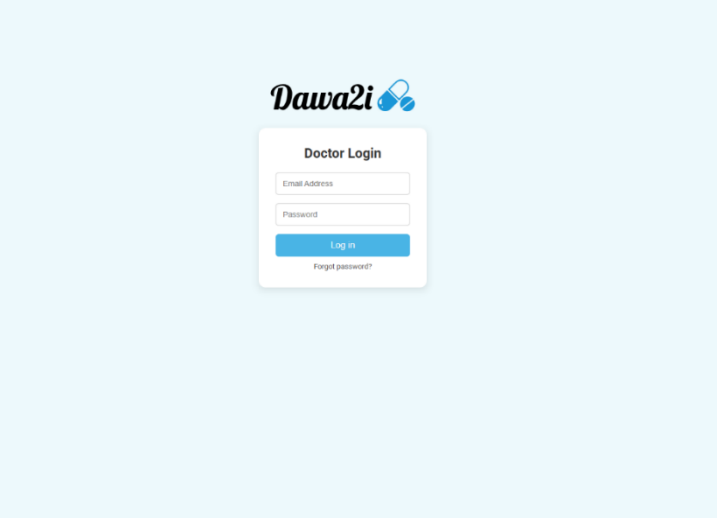

Figure 4-1. Doctor Login Interface

- **Doctors main Interface**

The following figure shows the main interface that the doctor faces where he will be able to create recipes and see the patient information

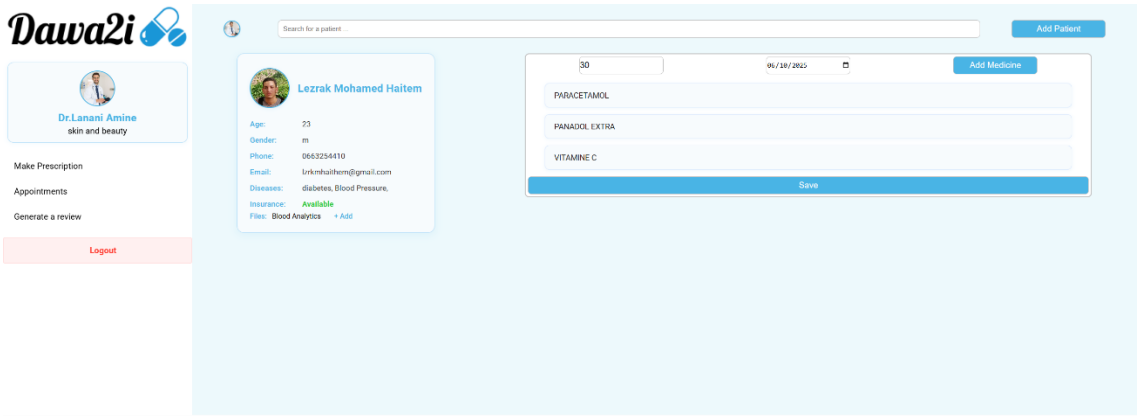

Figure 4-2. Doctor main Interface

Patient profile interface

The patient information that the doctor can access includes chronic diseases , prescription History and patient profile . like shown in the previous figures.

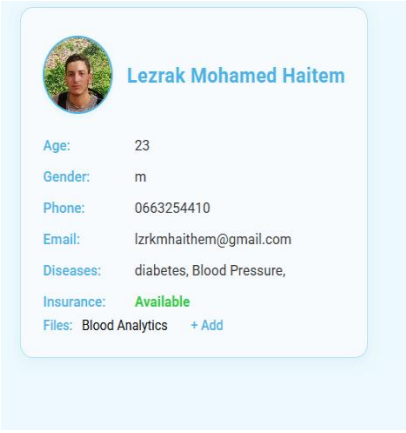

Figure 4-3 : Profile interface

- **AI document analysis interface**

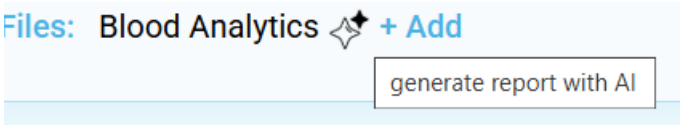

Figure 4-4 : Profile interface

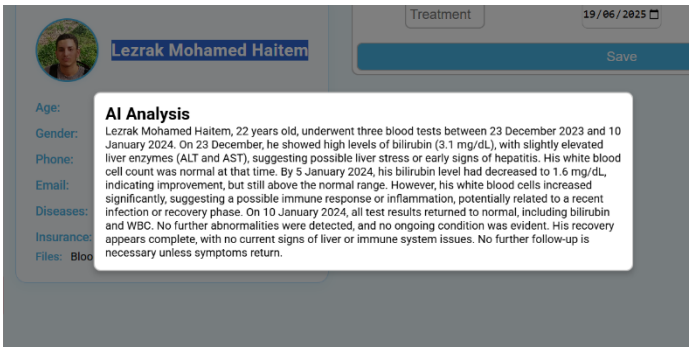

Figure 4-5 : Profile interface

- **Doctors prescription Interface**

Here, the doctor can add medicine and which dose to buy , and quantity , times of consuming with a note of additional instructions

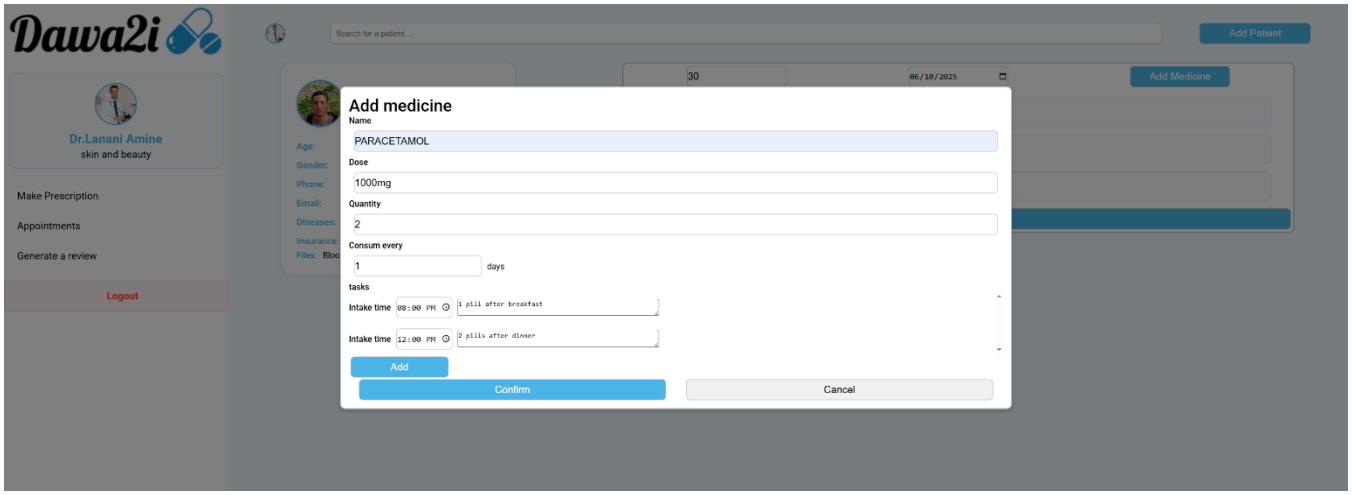

Figure 4-6 : Doctors prescription Interface

- **Saving and printing prescription**

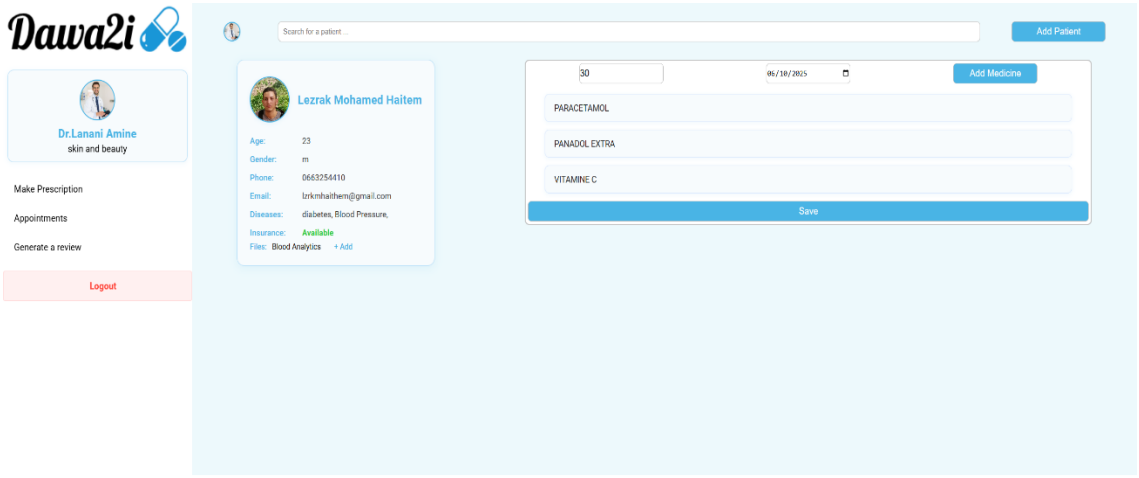

Figure 4-7 : implementation of prescription

After completing it the doctor clicks on confirm to save it, then he chooses whether he will print it or not.

In case he chose to print it, the following pdf file will be downloaded with a QR code containing the id of the prescription.

Figure 4-8 : printed prescription

- **Add a patient interface**

In case the patient doesn't have an account , the doctor can add him , the first figure shows the verification of the id to make sure that the id does not exist

Figure 4-9 : Id interface

After making sure that the id does not exist the doctor start to fill patient information

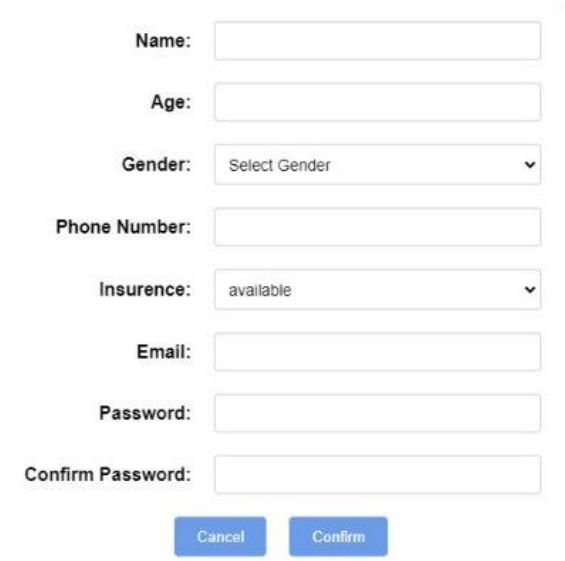

Figure 4-10 : Add patient interface

B . Pharmacist Interfaces

- **pharmacist main Interface**

When the patient arrives at the pharmacy with a digital prescription, the pharmacist opens the system, selects the patient's profile, and checks the prescribed medications against the available stock. If the medication is in stock, the pharmacist can activate and process the prescription. If not, they can order it

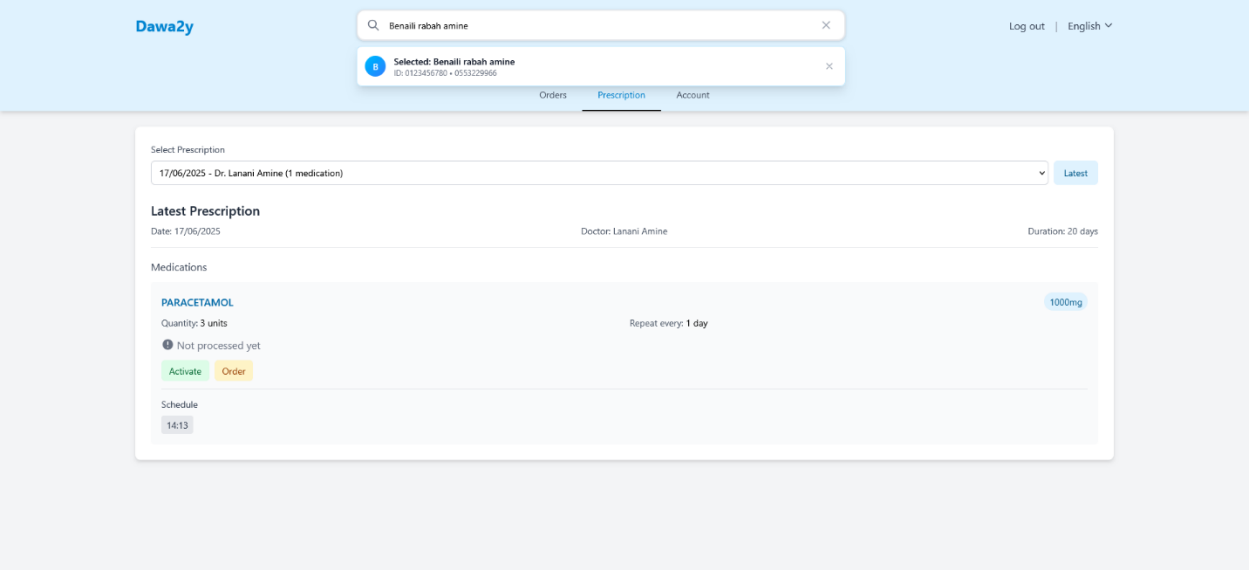

Figure 4-11. Pharmacist main page

- **pharmacist order interface**

When a patient requests a medication that is not currently available in the pharmacy, the pharmacist uses the "Orders" section to place a restock request. Each request is linked to the patient's profile and includes the medication name, quantity, and order date. Once the medication arrives, the pharmacist updates the status by marking it as "Arrived" and can then notify the patient through the system. This ensures smooth tracking of out-of-stock medications and keeps the patient informed without requiring repeated visits or phone calls.

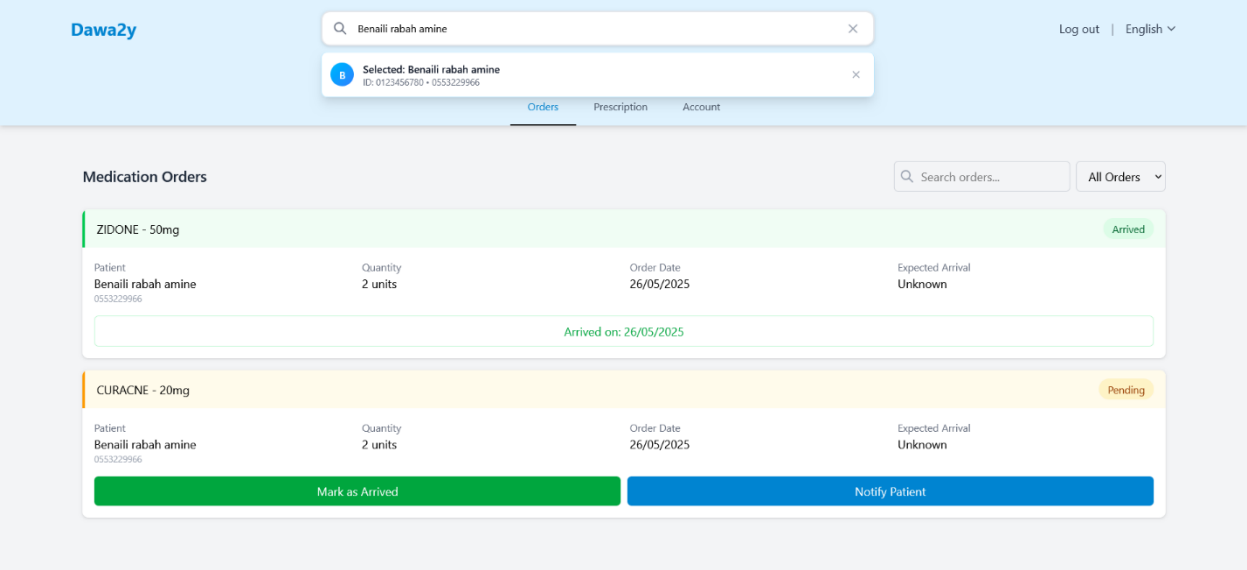

Figure 4-12. pharmacist order page

- **pharmacist account information interface**

When the pharmacist registers or logs into the system, they can access the "Account" section to manage their pharmacy's profile. From here, they can view and edit essential information such as the pharmacy name, phone number, location, email address, and license number. Keeping this information up to date ensures accurate identification within the platform, facilitates communication with patients and doctors, and allows smooth operation of prescription validation, order processing, and notification systems.

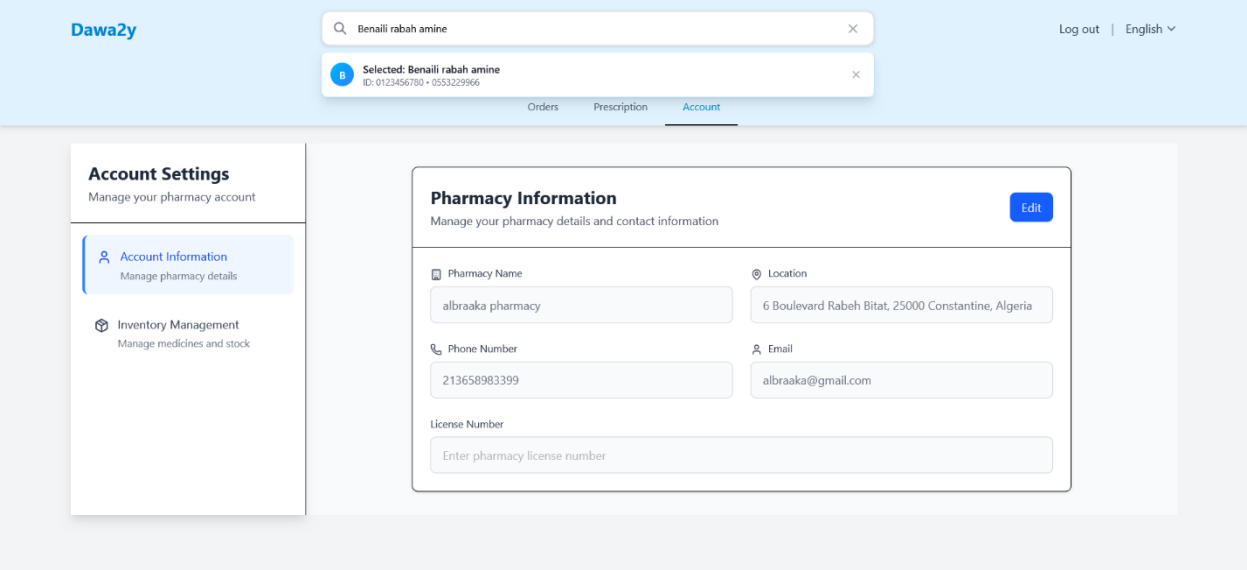

Figure 4-13. account information page

- **view inventory interface**

When the pharmacist needs to review or update the stock of medications, they navigate to the Inventory Management section under the Account tab. The system displays a detailed table listing all available medicines

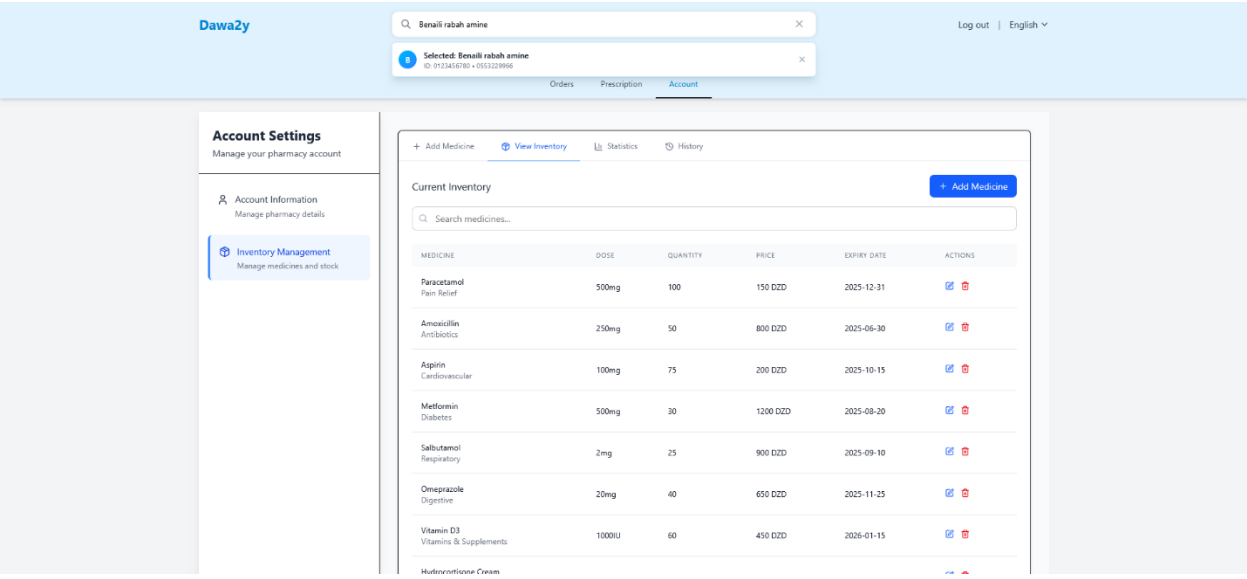

Figure 4-14. view inventory tab

- **statistics interface**

The pharmacist can navigate to the Statistics tab to get a comprehensive overview of the pharmacy's stock performance and inventory status.

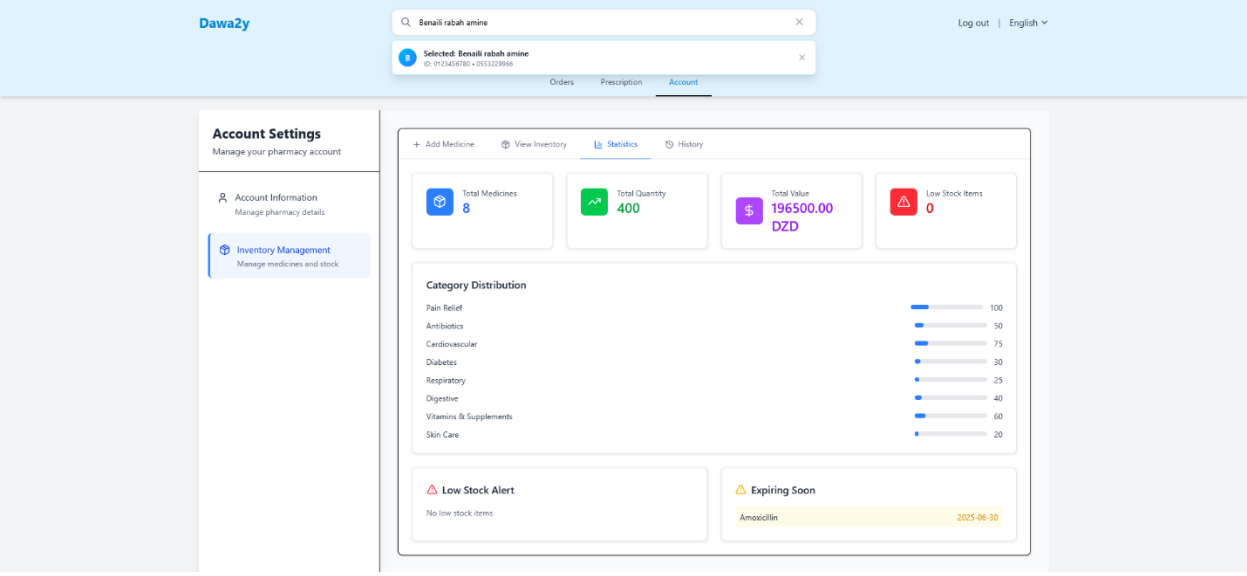

Figure 4-15. statistics tab

4.3.2. Android App interfaces

- **Login interface**

when the user opens the application for the first time, he will find the Login interface, which offers a secured email and password authentication, after filling the required fields, the application will verify the integrity of the inputs (the email input must have the format of an email, password with a minimum length of 6 characters). If the inputs meet the required conditions it will verify user's information to allow access to the home interface.

If the user doesn't have an account already, he can click on "create an account ?" link, it will take him to the interface of "Registration"

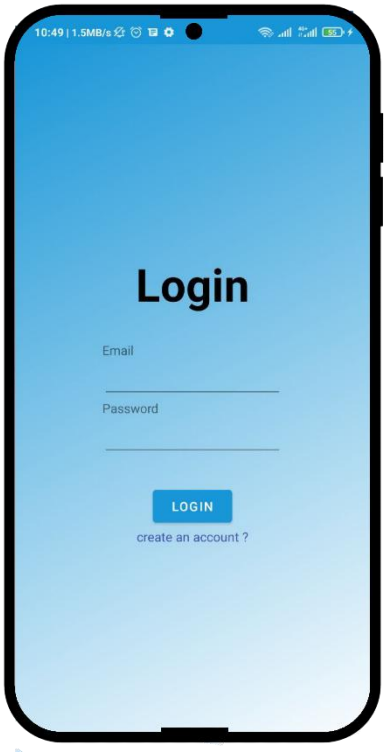

Figure 4-16 : Login interface

- **Registration Interface**

In this interface, user is able to create his own account by entering his personal information. after he fill all the required fields, the application will verify the integrity and the existence of a possible account that already has the same email or Identity number, after that it will confirm the creation of the account and forward the user to the login page to Login.

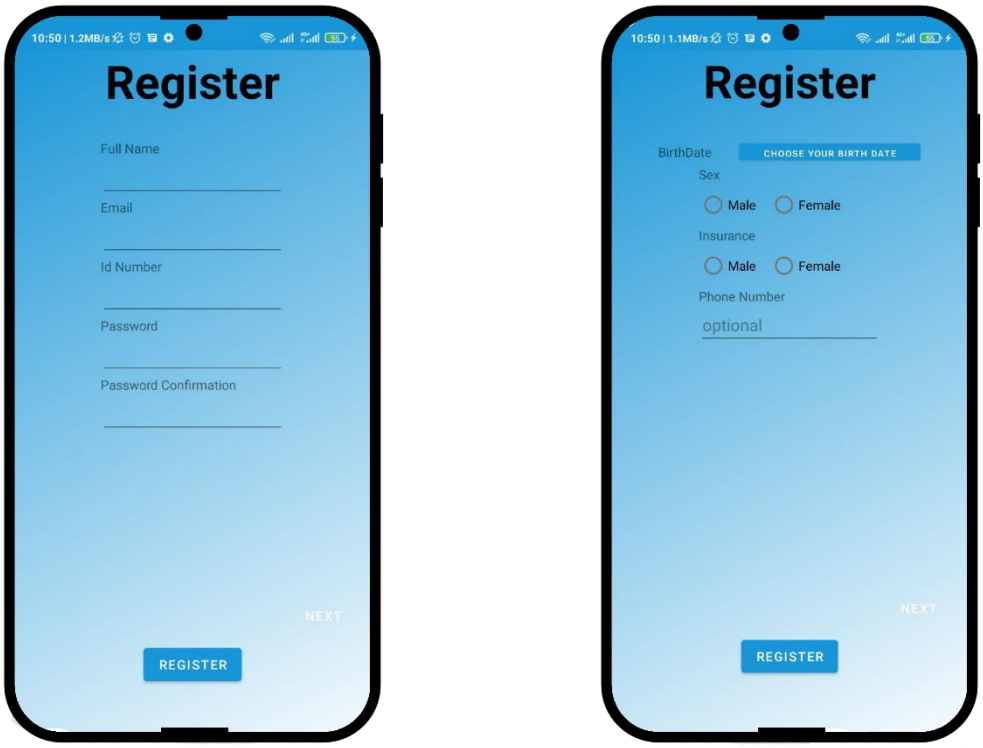

Figure 4-17 : Registration Interface

- **Home interface**

Home interface is when user can explore all services and options of the application, it is considered as the dashboard of the application, and it is provided with a drawer layout that also contains some options for the user.

Figure 4-18. Home interface

- **My next medications interface**

In this interface, user can explore the medication that he is taking currently, with a countdown timer that shows the remaining time for the next medication for each medicine that the patient is consuming in the present time

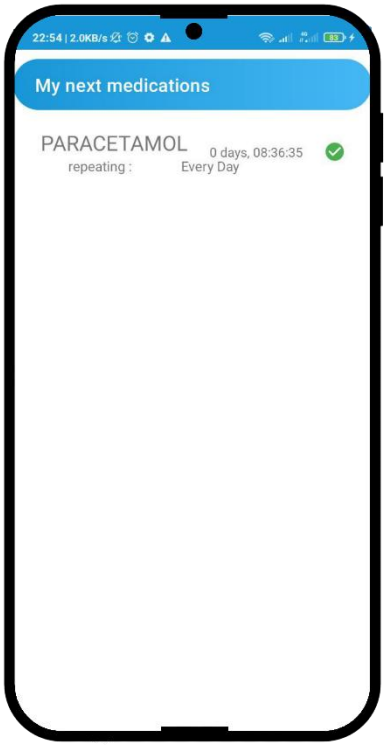

Figure 4-19 : My next medications interface

- **My alarm interface**

In this interface, user can create custom alarms with custom title and time for any medical activity that may not be available in the next medications interface, then the user will receive a notification when the alarm is triggered

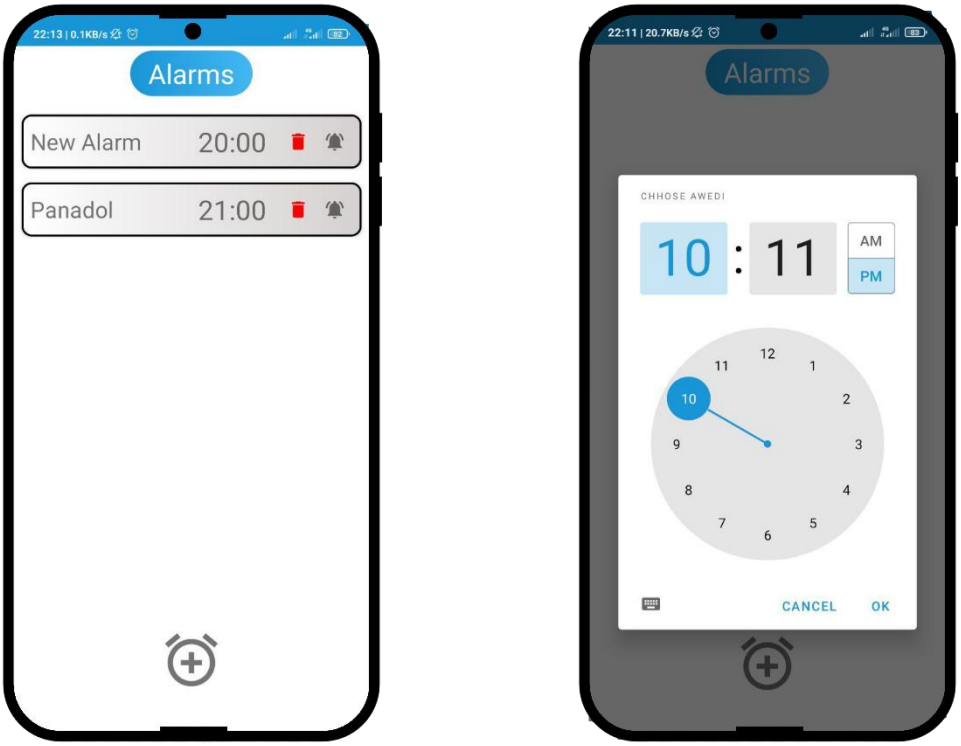

Figure 4-20 : My alarm interface

- **Scanner interface**

in this interface, the user can scan the QR code of a prescription. If the prescription id in the QR code is valid, the user will be directed to the prescription interface providing more details about the prescription.

Figure 4-21 : Scanner interface

- **prescription interface**

It shows the details of a prescription like date, doctor's name, medicines and times to take the medicine

Figure 4-22 : prescription interface

- **Medical Library interface**

This interface provides the user with a search engine to explore different medicines and its detailed description, including its medical paper

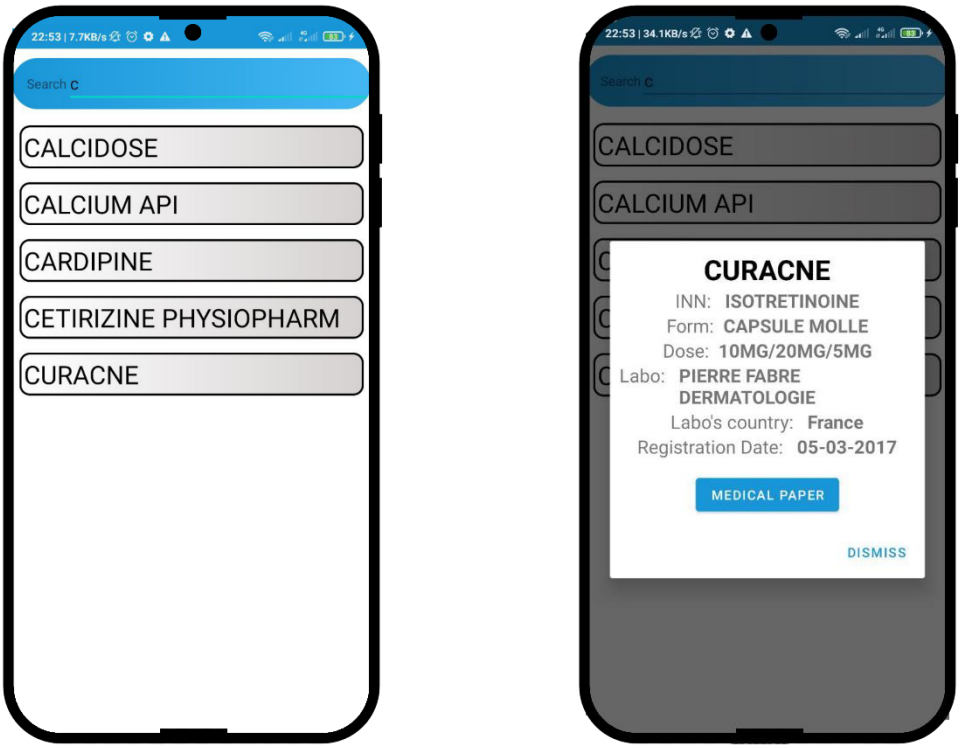

Figure 4-23 : Medical Library interface

- **Doctors interface**

This interface allows the user to search about different doctors and get their contact information like email and phone number. It also provides the user with a direct link to the doctor's location in google maps.

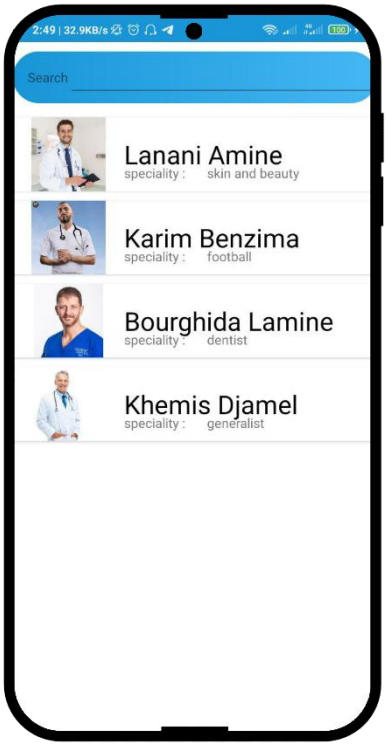

Figure 4-24. Doctors interface

- **Doctors profile interface**

This interface provides users with a detailed doctor profile, including name, specialty, photo, rating, and a brief bio. It displays contact information like phone number and email, offers a clickable button to view the clinic's location on Google Maps, and includes a "Book Appointment" button for easy scheduling.

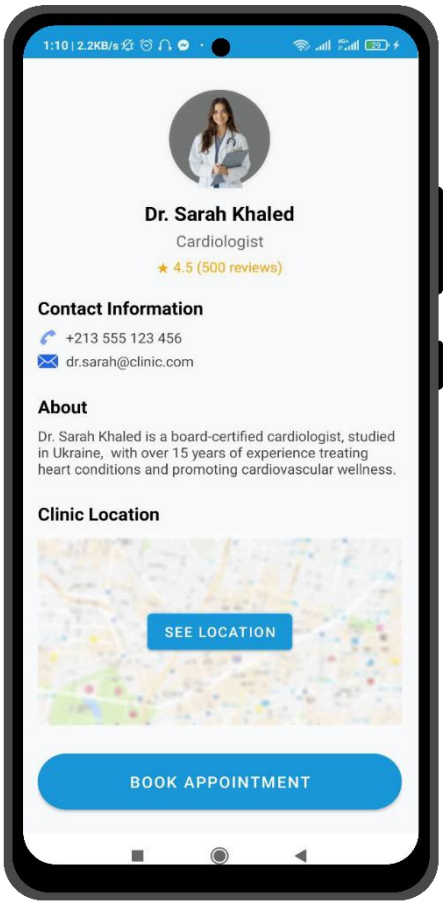

Figure 4-25.Doctors profile interface

- **Book Appointment interface**

This interface allows users to book an appointment with a selected doctor. It displays the doctor's profile, specialty, rating, and lets the user choose a preferred date and time slot from the available options. Once selections are made, the user can confirm the booking by tapping the "Book Appointment" button at the bottom.

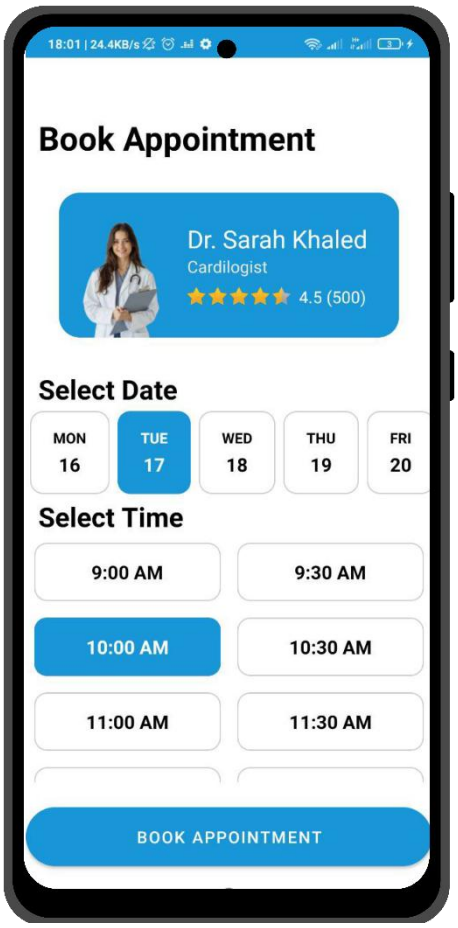

Figure 4-26.Book Appointement interface

- **Profile interface**

It shows user's personal information and medical records

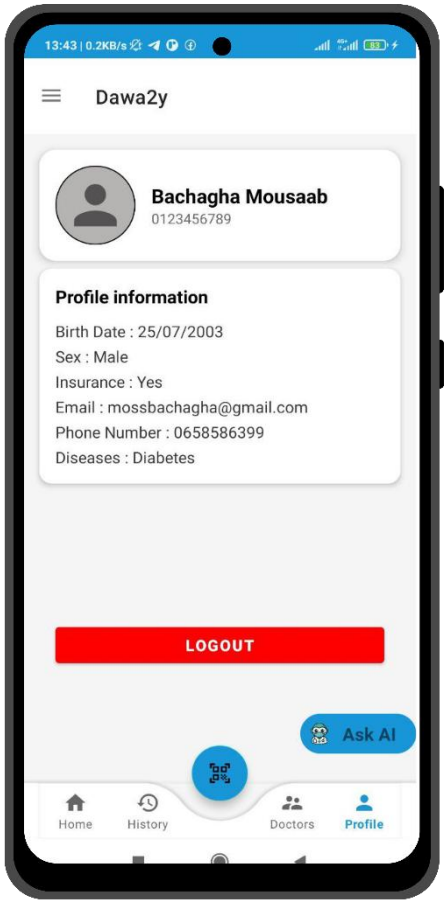

Figure 4-27.Profile interface

- **My prescriptions interface**

this interface allows the patient to explore his prescriptions and provide, if the user decided to see details of a certain description, he can click on the prescription element and the application will direct him to a prescription interface providing him with the details

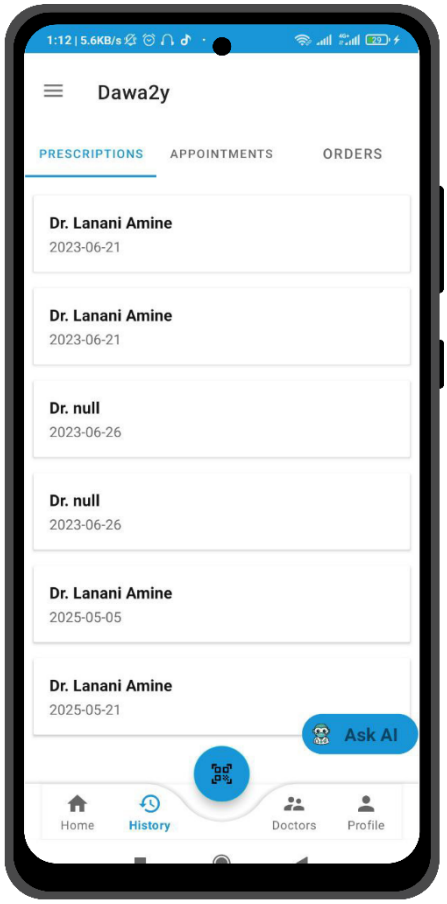

Figure 4-28.prescriptions interface

- **Appointments interface**

This interface displays a list of upcoming medical appointments scheduled through the Dawa2y app. Each card shows the date, doctor's name, specialty, and time of the appointment. Users can easily keep track of their medical visits

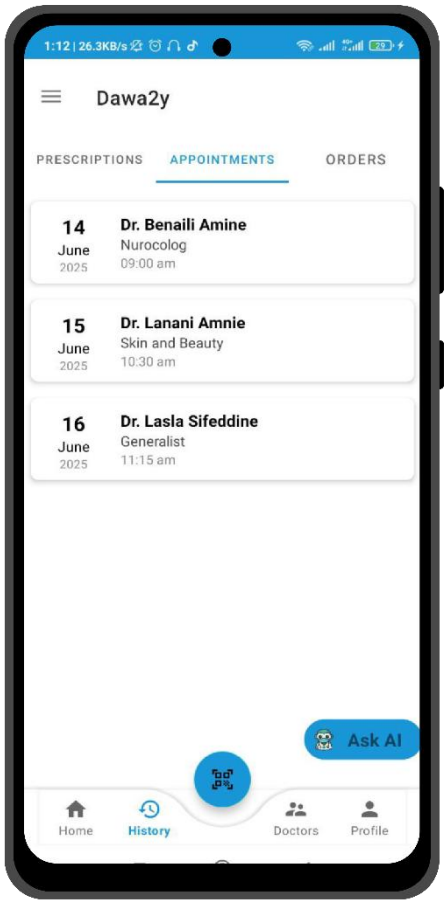

Figure 4-29.Appointments interface

- **Order interface**

This interface displays the user's recent medication orders, including the medicine name, pharmacy name, order date, and delivery status. Each entry clearly shows whether the medication has been delivered

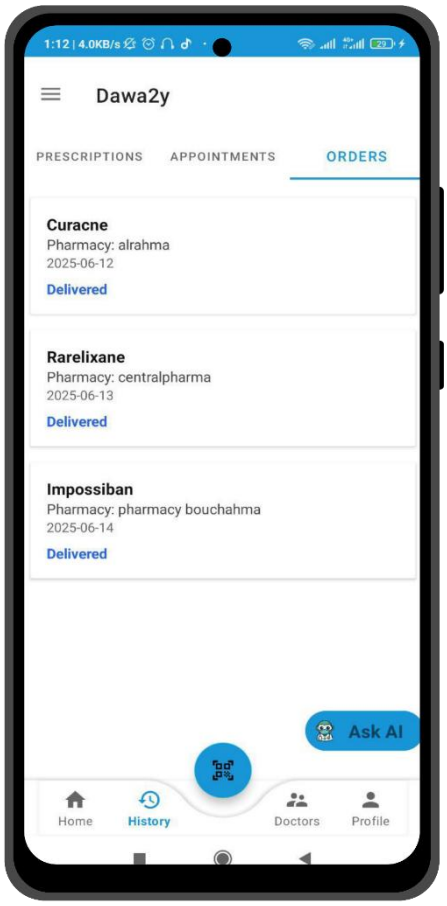

Figure 4-30. Order interface

- **Ai chat bot interface**

This is a floating action button labeled "Ask AI", featuring a friendly medical-themed assistant icon. It provides users with quick access to the in-app AI chatbot

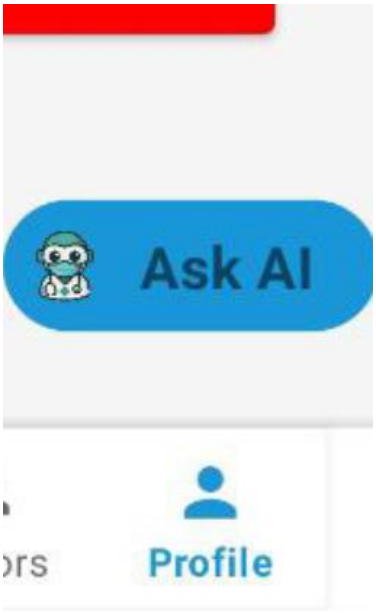

Figure 4-31.AI Button

And the following interface shows a conversation with the in-app AI medical assistant. Users can describe their symptoms in natural language, and the AI responds with possible medical explanations and relevant conditions. In this example, the assistant analyzes symptoms related to vertigo and suggests potential causes. The input field at the bottom allows users to ask follow-up questions, making it a helpful tool for quick, AI-powered medical support.

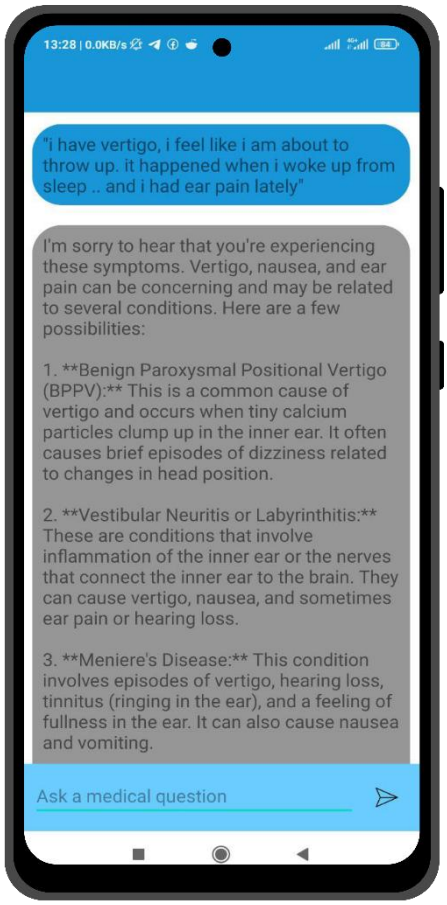

Figure 4-32. chat bot interface

4.4. Technical Enhancements and Optimizations

This chapter discusses the key optimizations implemented to improve the performance, security, and scalability of the healthcare appointment and patient management system. Three major enhancements were introduced: (1) end-to-end encryption for AI chatbot messages, (2) bitmask-based session booking for efficient Firebase queries, and (3) OCR-LLM pipeline automation for medical report generation. Each optimization is analyzed with empirical evidence or theoretical justification for its impact.

4.4.1. Encryption of AI Chatbot Messages

Problem: Storing sensitive conversations between patients and the AI chatbot in plaintext within Firebase posed serious privacy and compliance risks, especially in the event of a data leak.

- **Solution**:
  - We implemented **end-to-end encryption** by encrypting messages on the client side before they are uploaded to Firebase.
  - Generate a unique encryption key per user, derived from their password via **PBKDF2**.\
  - Used Java's Cipher API with AES/GCM/NoPadding for authenticated encryption that ensures both confidentiality and tamper detection.
  - Store only encrypted messages in Firebase; decryption occurs client-side after authentication.
- **Proof of Improvement**:
  - Firebase logs and snapshots contains no more human-readable message content.
  - All messages are stored in ciphertext format-completely unreadable without the user's password.
  - Even in case of Firebase misconfiguration or export, sensitive data remains secure.

4.4.2. Bitmask-Based Session Booking System

Problem: The original booking system queried **24 separate time slots per doctor per day**, resulting in excessive Firebase reads (O(N) per doctor-day) and increased network usage.. **Solution**: Most of the times, solving performance problems may lead to sacrificing memory, and vice versa. However, our solution includes a massive improvement for both memory and performance or execution time, in addition it results in an economic amount of Firebase reads which minimize Firebase fees.

- **Implementation**:
  - Store a single integer (sessionMask) per doctor/day in Firebase Realtime DB.

  - Client-side logic checks bits (e.g., (mask >> sessionIndex) & 1) to determine availability.
  - Updates use atomic Firebase transactions (runTransaction) to avoid race conditions.
- **Proof of Improvement**:
  - **Latency**: Replaced 24 reads with a single read per doctor/day.
  - **Bandwidth**: Reduced payload from ~12KB (24×500B) to 4B per record.
  - **Scalability**: With 100 doctors, we observed a **92% reduction in total daily read operations**..

4.4.3. OCR-LLM Pipeline for Patient History Reports

Problem: Sending images directly to the LLM (in multimodal settings) led to unnecessary token consumption and slower responses. Additionally raw document scans often caused poor OCR accuracy due to noise, low contrast, or scale.
Solution: Extracting text client-side and sending only the text to the LLM instead of images, reducing token cost and latency.

- **Implementation**:
  - A preprocessing function was added before OCR to improve image readability.
  - The enhanced image is passed to Tesseract for OCR.
  - Only the **cleaned and extracted text** is sent to the LLM, using a consistent prompt format for summarization, leading to few tokens usage.
- **Proof of Improvement**:
  - **Token usage reduction**: Switching from image input to text input reduced prompt size significantly (~85% on average).
  - Accuracy: Preprocessing led to better OCR output on low-quality scans, with improved character recognition.
  - **Latency**: LLM response time dropped due to smaller, cleaner text inputs.
  - **Cost-efficiency**: Avoiding image-to-text models saved LLM token cost, especially when scaling.

4.4.4. Optimization of RAG to Reduce Language Model Overgeneralization

Problem: Despite the growing integration of AI chatbots in healthcare applications, most conversational agents continue to generate generic responses that do not reflect the individual medical context of the user. In the current implementation of the *Dawa2y* chatbot, all patients receive similar answers to health-related questions regardless of their personal medical history, lab results, or clinical progress. This lack of contextual awareness limits the chatbot's usefulness in providing accurate, patient-specific information, and may even lead to misleading or inappropriate guidance. The challenge, therefore, lies in enabling the chatbot to dynamically tailor its responses based on each patient's medical data such as previous reports, diagnoses, or lab test results in order to deliver more meaningful, safe, and personalized interactions.

Solution: To address this limitation, we integrated a Retrieval-Augmented Generation (RAG) mechanism that grounds responses in each patient's medical history. By retrieving and embedding relevant data, the chatbot delivers context-aware, personalized answers.

A . Chunking Strategy Optimization

To decrease the prompt size, it would be better to split information related to context into chunks, and we chose to follow multiple approaches to apply chunking like fixed chunking and sliding window chunking, giving various chunks and context based ones.

B . Embedding Chunks

After chunking the patient's medical report into semantically coherent segments, each chunk is embedded into a numerical vector using a sentence embedding model such as all-MiniLM-L6-v2. These embeddings capture the meaning of each chunk and are stored locally for efficient retrieval. This will allow afterwards us to calculate the similarity between the patient's question and each chunk.

C . Adding Context to The User Prompt

After embedding the chunks and storing them, we can now embed the user's question and calculate its similarity with each of the embeddings, then we will choose the most similar K chunks (3 chunks in our case), those which are supposed to be more in the context of the patient's question, and we will add them to the user's prompt as a context.

Other patient information like age, sex and diseases will improve the personalization of the chatbot responses, so we can add them with the best chunks as a context to give the LLM better vision and more references.

D . Results

Report:

Lezrak Mohamed Haitem, 22 years old, underwent three blood tests between 23 December 2023 and 10 January 2024. On 23 December, he showed high levels of bilirubin (3.1 mg/dL), with slightly elevated liver enzymes (ALT and AST), suggesting possible liver stress or early signs of hepatitis. His white blood cell count was normal at that time. By 5 January 2024, his bilirubin level had decreased to 1.6 mg/dL, indicating improvement, but still above the normal range. However, his white blood cells increased significantly, suggesting a possible immune response or inflammation, potentially related to a recent infection or recovery phase. On 10 January 2024, all test results returned to normal, including bilirubin and WBC. No further abnormalities were detected, and no ongoing condition was evident. His recovery appears complete, with no current signs of liver or immune system issues. No further follow-up is necessary unless symptoms return.

User prompt:

What were my bilirubin levels on 23 December 2023?

Chunking:

```
Chunk 3: His white blood cell count was normal at that time.

--------------------------------------------------------------------------------
Chunk 4: By 5 January 2024, his bilirubin level had decreased to 1.6 mg/dL, indicating improvement, but still

--------------------------------------------------------------------------------
Chunk 5: However, his white blood cells increased significantly, suggesting a possible immune response or infl
 or recovery phase.

--------------------------------------------------------------------------------
Chunk 6: On 10 January 2024, all test results returned to normal, including bilirubin and WBC.

--------------------------------------------------------------------------------
Chunk 7: No further abnormalities were detected, and no ongoing condition was evident.

--------------------------------------------------------------------------------
Chunk 8: His recovery appears complete, with no current signs of liver or immune system issues.
```

Figure 4-33 : Chunking text

Choosing best chunks:

```
Score: 0.728
Chunk: By 5 January 2024, his bilirubin level had decreased to 1.6 mg/dL, indicating improvement, b
---------------------------------------
Score: 0.695
Chunk: On 23 December 2023, he showed high levels of bilirubin (3.1 mg/dL), with slightly elevated
or early signs of hepatitis.
---------------------------------------
Score: 0.668
Chunk: On 10 January 2024, all test results returned to normal, including bilirubin and WBC.
```

Figure 4-34 : Choosing best chunks

Result without RAG:

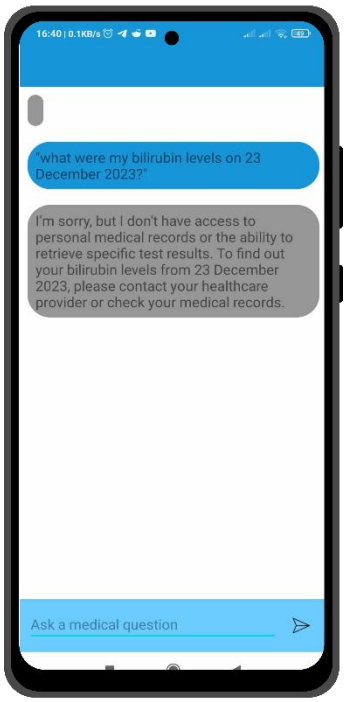

Figure 4-35 : Chatbot response without RAG

Result with RAG:

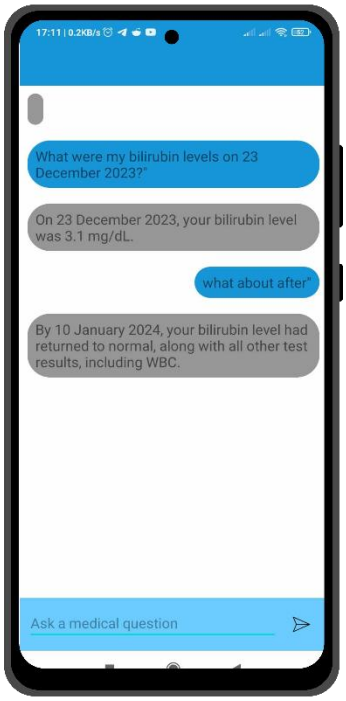

Figure 4-36 : Chatbot response with RAG

4.5. Conclusion

In this chapter, we presented the technical implementation of the Dawa2y system, detailing the development environment, tools, programming languages, and user interfaces across both the web and Android platforms. Furthermore, we introduced a set of key optimizations aimed at improving performance, scalability, and personalization. These include secure end-to-end encryption for sensitive patient-chatbot communication, a bitmask-based session booking system for efficient Firebase interaction, and an OCR-LLM pipeline for streamlined report processing. Most notably,

we addressed the critical limitation of overgeneralized AI chatbot responses by integrating Retrieval-Augmented Generation (RAG), enabling context-aware interactions grounded in each patient's unique medical data. These enhancements collectively contribute to a more secure, intelligent, and user-centric healthcare assistant system, laying a strong foundation for future expansions and clinical applicability.

Chapter 5 : General conclusion

General conclusion

This thesis has presented the design, development, and conceptual evaluation of an enhanced version of the "Dawa2y" medical platform, integrating advanced Artificial Intelligence (AI) technologies and an intelligent appointment management system. The project successfully addressed critical challenges in healthcare digitalization, such as the inefficiencies associated with manual data entry, the cognitive burden on healthcare professionals, and the limited accessibility and support available to patients.

By incorporating OCR and NLP technologies, the system now offers automatic extraction and interpretation of data from medical documents, reducing errors and saving time for doctors. The AI-driven chatbot further enhances patient engagement by providing immediate assistance and reliable information, thereby improving the overall user experience. Moreover, the intelligent appointment system streamlines scheduling processes for both patients and medical staff, reducing administrative overhead and enhancing service availability.

Despite the promising outcomes, the project acknowledges certain limitations, such as the reliance on a limited dataset for AI training and the focus on the Android platform, leaving room for broader platform support and clinical validation in future iterations.

In conclusion, the enhanced "Dawa2y" platform demonstrates the significant potential of AI-driven solutions in transforming healthcare services, improving communication between stakeholders, and increasing the efficiency and quality of care delivery. Future work will aim to expand system functionalities, improve AI model generalization, and pursue real-world deployment and validation to further realize its impact in the healthcare sector.

Bibliography

[1] G. A. F. M. Mauro, "“Digital transformation in healthcare: Assessing the role of digital technologies for managerial support processes”," *Technological Forecasting and Social Change,* vol. 209, p. 1–13, 2024.

[2] A. S. J. a. T.-M. Grønli, "Harnessing the digital revolution: A comprehensive review of mHealth applications for remote monitoring in transforming healthcare delivery," Cornell University Library, Ithaca, NY, 2024.

[3] "Doctolib," Wikimedia Foundation, [Online]. Available: https://fr.wikipedia.org/wiki/Doctolib. [Accessed June 2025].

[4] "NHS App," Wikipedia, The Free Encyclopedia, [Online]. Available: https://en.wikipedia.org/wiki/NHS_app. [Accessed june 2025].

[5] "Your.MD," May 2024 . [Online]. Available: https://en.wikipedia.org/wiki/Your.MD. [Accessed June 2025].

[6] "WebMD," May 2025. [Online]. Available: https://en.wikipedia.org/wiki/WebMD. [Accessed june 2025].

[7] "Ava Industries," LinkedIn, Company Page, [Online]. Available: https://www.linkedin.com/company/avaindustries/?originalSubdomain=ca. [Accessed June 2025].

[8] O. H. M. Services, "A healthcare giant is using AI to sift through millions of transactions. It’s saved employees 15,000 hours a month," Business Insider, New York, NY, 2025 .

[9] "How AI Chatbots Advance Healthcare for Patients and Providers," Coherent Solutions, February 2025.

[10] "Clinical decision support system," 2025 . [Online]. Available: https://en.wikipedia.org/wiki/Clinical_decision_support_system. [Accessed june 2025].

[11] [Online]. Available: https://www.raapidinc.com/ai-ml-dl/ocr/#:~:text=OCR%20(Optical%20Character%20Recognition)%20is,analyze%20data%20from%20medical%20documents..

[12] [Online]. Available: https://aws.amazon.com/what-is/retrieval-augmented-generation/.

[1 [Online]. Available: https://www.databricks.com/glossary/retrieval-augmented-generation-
3] rag.

[1 [Online]. Available: https://blogs.nvidia.com/blog/what-is-retrieval-augmented-generation/.
4]

[1 "UML," [Online]. Available: https://economictimes.indiatimes.com/definition/uml.
5] [Accessed June 2025].

[1 [Online]. Available: https://www.javatpoint.com/uml-use-case-diagram.
6]

[1 [Online]. Available: https://firebase.google.com/docs/auth.
7]

[1 [Online]. Available: https://www.visual-paradigm.com/guide/uml-unified-modeling-
8] language/what-is-class-diagram/.

[1 [Online]. Available: https://www.visual-paradigm.com/guide/uml-unified-modeling-
9] language/what-is-activity-diagram/.

[2 [Online]. Available: James Rumbaugh, Ivar Jacobson, Grady Booch, The Unified Modeling
0] Language Reference Manual, 2nd edition, .

[2 [Online]. Available: https://code.visualstudio.com/.
1]

[2 [Online]. Available: https://firebase.google.com/.
2]

[2 "Android SDK and its Components," GeeksforGeeks, 16 Feb 2023. [Online]. Available:
3] https://www.geeksforgeeks.org/android/android-sdk-and-its-components/. [Accessed June 2025].

[2 N. Abramowski, "“What is NPM? A Beginner’s Guide”," CareerFoundry, 28 Nov 2022 .
4] [Online]. Available: https://careerfoundry.com/en/blog/web-development/what-is-npm/#what-is-npm. [Accessed june 2025].

[2 "React (software)," [Online]. Available:
5] https://en.wikipedia.org/wiki/React_(software)#:~:text=React%20(also%20known%20as%20React,of%20individual%20developers%20and%20companies..

[2 "Introduction to Tailwind CSS," GeeksforGeeks, 6 Sep 2022. [Online]. Available:
6] https://www.geeksforgeeks.org/css/introduction-to-tailwind-css/. [Accessed june 2025].

[27] [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/HTML.

[28] [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/CSS.

[29] [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/JavaScript.

[30] [Online]. Available: https://www.w3schools.com/java/java_intro.asp.

[31] [Online]. Available: https://tesseract-ocr.github.io/tessdoc/.

[32] [Online]. Available: https://uk.trustpilot.com/review/n8n.io.

[33] [Online]. Available: https://developer.android.com/studio.

[34] [Online]. Available: https://en.m.wikipedia.org/wiki/Software_development_kit.

[35] [Online]. Available: https://www.apachefriends.org/.

[36] [Online]. Available: https://www.sqlite.org/about.html.

[37] "Doctolib," [Online]. Available: https://apibit.com/product/doctolib/. [Accessed June 2025].

[38] [Online]. Available: https://blogs.nvidia.com/blog/what-is-retrieval-augmented-generation/.