



Ministry of High Education

Culture and Science City C.S.C at 6th of October

The High Institute of Computer Science and Information Systems

(Graduation Project)

Intelligent Hospital Management System

Prepared by:

Name	ID
Youssef Sherif Hassan Al-Sayed	20183802
Mohamed Khaled Mohamed Abu-Youssef	20192918
Hesham Mohamed Shahat Mahfouz	20200068
Abdallah Fekry Mohamed Moustafa	20200192
Mahmoud Mohamed Ali Ibrahim Al-Sharkawy	20200247
Mohamed Ahmed Abdelazeem Ashmawy	20200594
Rahma Mohamed Kamel Ali	20201840

Supervised by:

Dr. Amira Mohammed Gaber

Eng. Abdelrahman Sherif



Ministry of High Education

Culture and Science City C.S.C at 6th of October

The High Institute of Computer Science and Information Systems

(Graduation Project)

Intelligent Hospital Management System

Prepared by:

الاسم	كود الطالب
يوسف شريف حسن السيد	20183802
محمد خالد محمد أبو يوسف	20192918
هشام محمد شحات محفوظ	20200068
عبدالله فكري محمد مصطفى	20200192
محمود محمد علي إبراهيم الشرقاوي	20200247
محمد احمد عبدالعظيم عشماوي	20200594
رحمة محمد كامل علي	20201840

Supervised by:

د. أميرة محمد جابر

م. عبد الرحمن شريف

Acknowledgement

Firstly, we would like to express our deepest gratitude to the esteemed faculty and administration of the institute, especially to the Dean, for their unwavering support and guidance throughout our academic journey.

Your leadership and dedication have been instrumental in shaping our educational experience.

A special thank you goes to our supervisors, whose expertise and encouragement have been invaluable. Your insights, patience, and commitment to excellence have greatly contributed to the success of this project. We are profoundly grateful for the time and effort you have invested in mentoring us.

Thank you all for believing in us and for providing an environment where we can thrive and achieve our goals.

Abstract

The Smart Medical and Healthcare Application will empower patients to access intelligent medical diagnoses and scan medical imaging scans or electrocardiography ECG by Artificial Intelligence techniques, it also provide to the patients booking an appointments, purchase a prescription, interact with doctors, in addition they can make a personality analysis test and information about their personality type that helps in making better decisions in their life, patients also will getting notification remainder on appointments time, and manage their healthcare needs seamlessly.

In the other side the web application provides clinic doctors, pharmacists, and laboratory specialists to managing and scheduling their patients' appointments.

The mobile and web applications leverage artificial intelligence models and user-friendly interfaces to deliver an exceptional user experience.

Table of Contents

Chapter 1: Introduction	1
1.1 Introduction	2
1.2 Problem Statement	3
1.3 Software Requirement Specification (SRS)	4
Chapter 2: System Analysis	14
2.1 System Models – Diagrams	15
2.1.1 Context Diagram	15
2.1.2 Use Case Diagram	15
2.1.3 Sequence Diagram	16
2.1.4 Activity Diagram	28
2.1.5 Entity Relationship Diagram ERD	30
2.1.6 Class Diagram	31
Chapter 3: System Design	32
3.1 User Interface (UI) Design	33
3.1.1 Mobile App UI	33
3.1.2 Web App UI	48
3.2 User Experience (UX)	52
Chapter 4: Implementation	62
4.1 Artificial Intelligence	63
4.2 Mobile Application Development	72
4.3 Fron-end Development	83
4.4 Backend Development	87
Chapter 5: Conclusion and Future Improvements	92
5.1 Conclusion	93
5.2 Future Improvements	94
References	95

List of Figures

Figure 1.1 Context Diagram	15
Figure 1.2 Use Case Diagram	15
Figure 1.3 Sequence Diagram Login	16
Figure 1.4 Sequence Diagram Signup	17
Figure 1.5 Sequence Diagram Chatbot	18
Figure 1.6 Sequence Diagram ECG Scan	19
Figure 1.7 Sequence Diagram Medical Imaging Scan	20
Figure 1.8 Sequence Diagram Personality Analysis Test	21
Figure 1.9 Sequence Diagram Chatting	22
Figure 1.10 Sequence Diagram Laboratory Reservation	23
Figure 1.11 Sequence Diagram Order Medication	24
Figure 1.12 Sequence Diagram Clinic Reservation	25
Figure 1.13 Sequence Diagram Accept Reservation	26
Figure 1.14 Sequence Diagram Refuse Reservation	27
Figure 1.15 Activity Diagram Patient	28
Figure 1.16 Activity Diagram Doctors	29
Figure 1.17 Entity Relationship Diagram ERD	30
Figure 1.18 Class Diagram	31
Figure 2.1 UI Mobile Bash Screen	32
Figure 2.2 UI Mobile Login Screen	33
Figure 2.3 UI Mobile Signup Screen	34
Figure 2.4 UI Mobile Home Page Screen	35
Figure 2.5 UI Mobile Chatbot Blank Screen	36
Figure 2.6 UI Mobile Chatbot Diagnosis Screen	37
Figure 2.7 UI Mobile Medical Imaging Scan Blank Screen	38
Figure 2.8 UI Mobile Medical Imaging Diagnosis Screen	39
Figure 2.9 UI Mobile Booking Doctors' Appointment Screen	40
Figure 2.10 UI Mobile Booking Laboratory' Appointment Screen	41
Figure 2.11 UI Mobile Ordering Pharmacy' Medicine Screen	42
Figure 2.12 UI Mobile Search Screen	43
Figure 2.13 UI Mobile Scheduling Appointments Screen	44
Figure 2.14 UI Mobile Personality Analysis Test Screen	45
Figure 2.15 UI Mobile Edit Profile Screen	46
Figure 2.16 UI Web Scrolling Identification Screens	47
Figure 2.17 UI Web FAQ, Contact-us Screens	48
Figure 2.18 UI Web Login & Sign-up Screens	49

List of Figures

Figure 2.19 UI Web Doctors Appointments Scheduling Screen	50
Figure 3.1 UX Persona 1: Patient	57
Figure 3.2 UX Persona 2: Doctor	58
Figure 3.3 UX Persona 3: Laboratory Specialist	58
Figure 3.4 UX Empathy Map	59
Figure 3.5 UX Empathy Map	59
Figure 3.6 UX Empathy Map	60
Figure 3.7 UX User Flow Patient	60
Figure 3.8 UX User Flow Doctors	61
Figure 4.1 AI Phases	63
Figure 4.2 AI VGG-16 Model Architecture	67
Figure 4.3 AI LSTM Model Architecture	67
Figure 4.4 AI Medical Imaging Scan Models Sequence	68
Figure 4.5 AI Training History	69
Figure 4.6 AI Sample outputs Heatmaps	70
Figure 4.7 AI Model Deployment and API Requests	71
Figure 5.1 Mobile App API Request	77
Figure 5.2 Mobile App View Model	81
Figure 5.3 Frontend with Backend Connection	85
Figure 6.1 Backend API requests	87

Chapter 1: Introduction

Introduction

The project consists of three main modules

Module 1: An AI based smart chatbot called "Caroline" talking to the patient and taking its disease symptoms, then diagnosing the disease and recommend making some tests as x-ray, MRI ... in addition, given information about the predicted disease as an overview, symptoms, and treatments. It can predict 30 diseases such as (Breast Cancer, Influenza, Covid 19, Stroke, ...)

Module 2: A sequence of AI Computer Vision models for scanning medical imaging and tests it can scan (X-ray, MRI, CT, OCT, ECG, or Food image), detect the image type (Image Recognition), if it is medical imaging image, applying anatomical recognition, disease evaluation, disease diagnosis. It can predict 25 disease types such as (Bone Fracture, Brain Tumor, Covid 19, Breast Cancer, ...)

It also can recognize 101 food types from images and shows the approximate number of calories per gram.

Module 3: A mobile and web applications (native).

Patients after applying the smart diagnosis, chatbot can orientate them to booking an appointment in a laboratory, clinic, or even to purchase the prescription medicines.

The patient can search for clinics, Laboratories, or pharmacies each have its page, and filter them according to the nearest, cheapest, or according to its health insurance.

The doctors, pharmacists, and laboratory specialists can create accounts on the website and schedule patients' appointments.

The application will notify the patient to remind him for a doctor's appointment or for its medication dose.

1.1. Problem Statement

Sometimes the patient needs to get a medical advice in a fast way, but it have to wait a lot of time to get the diagnosis that happens because of large crowds and queues in the hospitals, laboratory or clinics places system every day, also doctors need to schedule their reservations considering the priority of which one will start first when there is a large number of patients at the clinic or the hospital.

So, they need a common system to handle these problems by helping patients to make a smart diagnosis, access the critical information fastly and book an appointment. And helping doctors to manage and schedule their reservations.

1.2. Software Requirement Specification SRS

1.2.1 Introduction

The following section provides an overview for the software requirements specifications (SRS) for the comprehensive medical system “I Care”.

1.2.1.1 purpose

AI based smart application that mimics a doctor.

The main purpose for the system is to make a closed circuit around the main users for the medical sector which are patient, doctor or clinic, laboratory, and pharmacy by providing the appropriate needs for each of them.

The application will help patients diagnose diseases and give them all information about the disease, easily book a doctor's appointment, reminder of medication appointments, sending the prescription to the nearest pharmacy to order medicines, scan complex X-ray, MRI, CT, OCT scans, and ECG with click of a button.

The web application will help doctors to schedule reservations.

In addition, canceling queues and crowds by providing “easy healthcare for anyone anywhere”.

1.2.1.2 Project Scope

The application will help all the medical sector members as patients, doctors, clinics, laboratories and pharmacies in all their medical needs as diagnosis diseases, disease information retrieval, medical imaging scanning, personality analysis, scheduling the appointments for the patient's side or the doctors' side.

1.2.1.3 SRS References

<https://youtube.com/playlist?list=PL4mqzqquSRgaQzbRMiVNzVI0UHeJrS9q&si=u9fbVkJthmLg-W>

<https://youtube.com/playlist?list=PLQUxWr2rTswnKJ6N6WI9ae1gkYLhGH2TU&si=lKkX4BAvxYPmHwnr>

https://web.cs.dal.ca/~hawkey/3130/srs_template-ieee.doc

1.2.2 Overall Description

1.2.2.1 Product Perspective

The comprehensive medical system consists of hardware in the shape of server and devices and software mobile application and web application.

1.2.2.2 Product Features

Patients can talk with the chatbot to make a fast smart diagnosis and get information about the disease.

Patients can book a doctor's appointment.

Patients can send the prescription to the nearest pharmacy to order medicines.

Patients can get reminders of medication appointments.

Patients can scan complex X-ray scan, Magnetic Resonance Imaging (MRI) scan, Computerized Tomography (CT) scan, Optical Coherence Tomography (OCT) scan, or Electrocardiography (ECG) and diagnosis its disease then get information about it.

Patients can scan a food image to identify the number of calories in it.

Doctors can schedule their reservations.

1.2.2.3 User Classes and Characteristics

The system has four main users which are patients, doctors or clinics, pharmacists, and laboratory specialists.

Patient: will book appointments, and access medical information.

Doctors: will add clinics and schedule reservations.

Pharmacists: will receive prescriptions then send medicines to the patient.

Laboratory specialists: will schedule reservations and send the results to the user on the application.

1.2.2.4 Operating Environment

The mobile application will run on android operating systems, so it will need a mobile device or PC simulator to be run.

While the web application can run on any web browser like Google Chrome, Fire Fox, Microsoft Edge, Opera, and even Internet Explorer so it does not require a specific operating system can run on personal computers or desktop with any operating systems as Windows, Linux, or MacOS and on mobile devices that supports Android or IOS.

1.2.2.5 Assumptions and Dependencies

The software will always be used on desktop computers, laptops, or mobile phones so all the actors must be aware of using computers and English language.

1.2.3 System Features

1.2.3.1 Functional Requirements

1.2.3.1.1 User Class 1 - Patient

1.2.3.1.1.1 Functional Requirement 1.1

ID: FR1

TITLE: Download the mobile application

DESC: The user should be able to download the mobile application from a trustworthy place as “App Store”. The application should be free to download.

1.2.3.1.1.2 Functional Requirement 1.2

ID: FR2

TITLE: Create account for new users

DESC: The user can register with a valid email address and password.

1.2.3.1.1.3 Functional Requirement 1.3

ID: FR3

TITLE: Manage personal information

DESC: The user can add or edit personal information during or after the sign-up process.

1.2.3.1.1.4 Functional Requirement 1.4

ID: FR4

TITLE: Smart digital diagnosis with AI Chatbot

DESC: The user can make a fast smart medical diagnosis by adding its symptoms of the disease to an AI based chatbot, then the bot detects the illness type then give him information about it and recommend imaging scans and tests to be made by the patient then automatically orientate the patient to the laboratory screen and auto filtering the laboratories according to the category based on that illness.

1.2.3.1.1.5 Functional Requirement 1.5

ID: FR5

TITLE: Scan a medical imaging image

DESC: The user can upload an X-ray scan, Magnetic Resonance Imaging (MRI) scan, Computerized Tomography (CT) scan, Optical Coherence Tomography (OCT) scan, or Electrocardiography (ECG), then detected by artificial intelligence AI and diagnosis the illness, then give information about then automatically orientate the patient to the clinics screen and auto filtering the doctors according to the category based on that illness.

1.2.3.1.1.6 Functional Requirement 1.6

ID: FR6

TITLE: Searching for a clinic

DESC: The user can search for all doctor's clinics and choose according to the nearest, cheapest, or highest rate (rating system), he can also search for doctors in its work union.

1.2.3.1.1.7 Functional Requirement 1.7

ID: FR7

TITLE: Booking a clinic appointment

DESC: The user can book a clinic appointment.

1.2.3.1.1.8 Functional Requirement 1.8

ID: FR8

TITLE: Order medicines from pharmacies with prescriptions

DESC: The user can send the prescriptions and order the medication from the pharmacies category.

1.2.3.1.1.9 Functional Requirement 1.9

ID: FR9

TITLE: Get reminder notification for appointments

DESC: The user can add its medication appointments and the application will make an alarm to remind the patient to take its potion.

1.2.3.1.1.10 Functional Requirement 1.10

ID: FR10

TITLE: Make a personality analysis test

DESC: The user can make a personality analysis test and get information about his personality to help him take the best decision in case he is confused about something like department or career track.

1.2.3.1.1.11 Functional Requirement 1.11

ID: FR11

TITLE: Chatting with real doctors

DESC: The user can chat with anyone of the doctors, pharmacists, or laboratory specialists for asking about any information needed.

1.2.3.1.2 User Class 2 - Doctor

1.2.3.1.2.1 Functional Requirement 2.1

ID: FR1

TITLE: Create account for new users

DESC: The user can register with a valid email address and password.

1.2.3.1.2.2 Functional Requirement 2.2

ID: FR2

TITLE: Add or edit personal information

DESC: The user can add or edit personal information during or after the sign-up process such as clinic category or location.

1.2.3.1.2.3 Functional Requirement 2.3

ID: FR3

TITLE: Scheduling patient's appointments

DESC: The user can schedule its reservations (accept or refuse).

[1.2.3.1.2.4 Functional Requirement 2.4](#)

ID: FR4

TITLE: Chatting with patient

DESC: The user can chat with the patient to give him medical advice.

[1.2.3.1.3 User Class 3 - Pharmacist](#)

[1.2.3.1.3.1 Functional Requirement 3.1](#)

ID: FR1

TITLE: Create account for new users

DESC: The user can register with a valid email address and password.

[1.2.3.1.3.2 Functional Requirement 3.2](#)

ID: FR2

TITLE: Add or edit personal information

DESC: The user can add or edit personal information during or after the sign-up process such as clinic category or location.

[1.2.3.1.3.3 Functional Requirement 3.3](#)

ID: FR3

TITLE: Receive prescription orders

DESC: Can receive a prescription from patients orders and send the medications to the specified location.

[1.2.3.1.3.4 Functional Requirement 3.4](#)

ID: FR4

TITLE: Chatting with patient

DESC: The user can chat with the patient in case there is information needed.

1.2.3.1.4 User Class 4 – Laboratory Specialist

1.2.3.1.4.1 Functional Requirement 4.1

ID: FR1

TITLE: Create account for new users

DESC: The user can register with a valid email address and password.

1.2.3.1.4.2 Functional Requirement 4.2

ID: FR2

TITLE: Add or edit personal information

DESC: The user can add or edit personal information during or after the sign-up process such as clinic category or location.

1.2.3.1.4.3 Functional Requirement 4.3

ID: FR3

TITLE: Scheduling patient's appointments

DESC: The user can schedule reservations (accept or refuse).

1.2.3.1.4.4 Functional Requirement 4.4

ID: FR4

TITLE: Assign scan results to the patient

DESC: Can send the scan results or the x-rays to the patient on the app when it has been done.

1.2.3.1.4.5 Functional Requirement 4.5

ID: FR5

TITLE: Chatting with patient

DESC: The user can chat with the patient in case there is information needed.

1.2.4 External Interface Requirements

1.2.4.1 User Interfaces

Using easy graphical user interfaces for both mobile and web applications.

All the design colors should be consistent with logo colors which are white and green.

1.2.4.2 Hardware Interfaces

Using a large storage as possible to provide enough space for artificial intelligence needs and the several large data for several users.

Using strong computing power for processing to perform artificial intelligence tasks in a fast way.

Upload database on a web server to be accessed anytime by the authorized users.

The used mobile phones which will run the application must contain a GPS sensor to identify their locations.

1.2.4.3 Software Interfaces

The system needs to access google maps to calculate the nearest locations to the patient to recommend them.

The system needs to access google calendar for patient's appointments.

It also needs to access google translate because the chatbot is in English and the user may use the Arabic language, so we need to translate the user's Arabic into English then give it to the chatbot and get the output then send it to translate the English output into Arabic then send it to the user.

1.2.4.4 Communications Interfaces

Communication between the different parts of the system is important since they depend on each other's.

Communication ways as below:

- Chatting between the mobile application by the patient and each of the doctors, the pharmacists, and the laboratory specialists on the web application.
- Application reminder notification.

1.2.5 Nonfunctional Requirements

1.2.5.1 Performance Requirements

The system needs to work on fast as possible for providing a smooth fast comfortable usage for the users, especially for the artificial intelligence processing or predictions, we can provide that by using strong computing power devices for processing, using a fast retrieval memory in internal ram memory, and fast hard disks for the storage for information retrieval.

1.2.5.2 Safety Requirements

For safety patients mustn't rely on the artificial intelligence disease diagnosis 100% during the trial period until the outputs are reviewed from specialists for a period of time to guarantee that the outcomes are accurate as possible to avoid disease misclassification.

1.2.5.3 Security Requirements

- The system should use the protocol HTTPS instead of HTTP for transferring media files and messages as it is more secure.
- The messages should be encrypted for login communications using a strong asymmetric cypher algorithm, so others cannot get username or password from these messages.
- The conversation chats should be end to end encrypted to ensure confidentiality of the conversation between patient and doctors.
- The database must be protected from unauthorized access.

Chapter 2: System Analysis

System Analysis

2.1. System Models

Context Diagram:

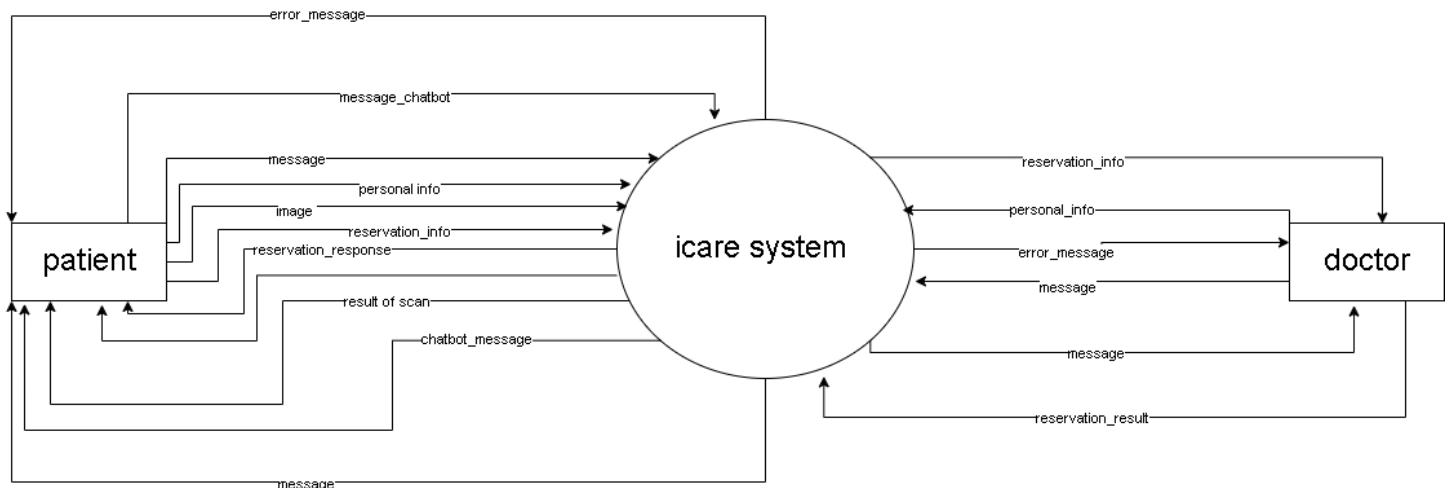


Figure 1.1

Use Case Diagram:

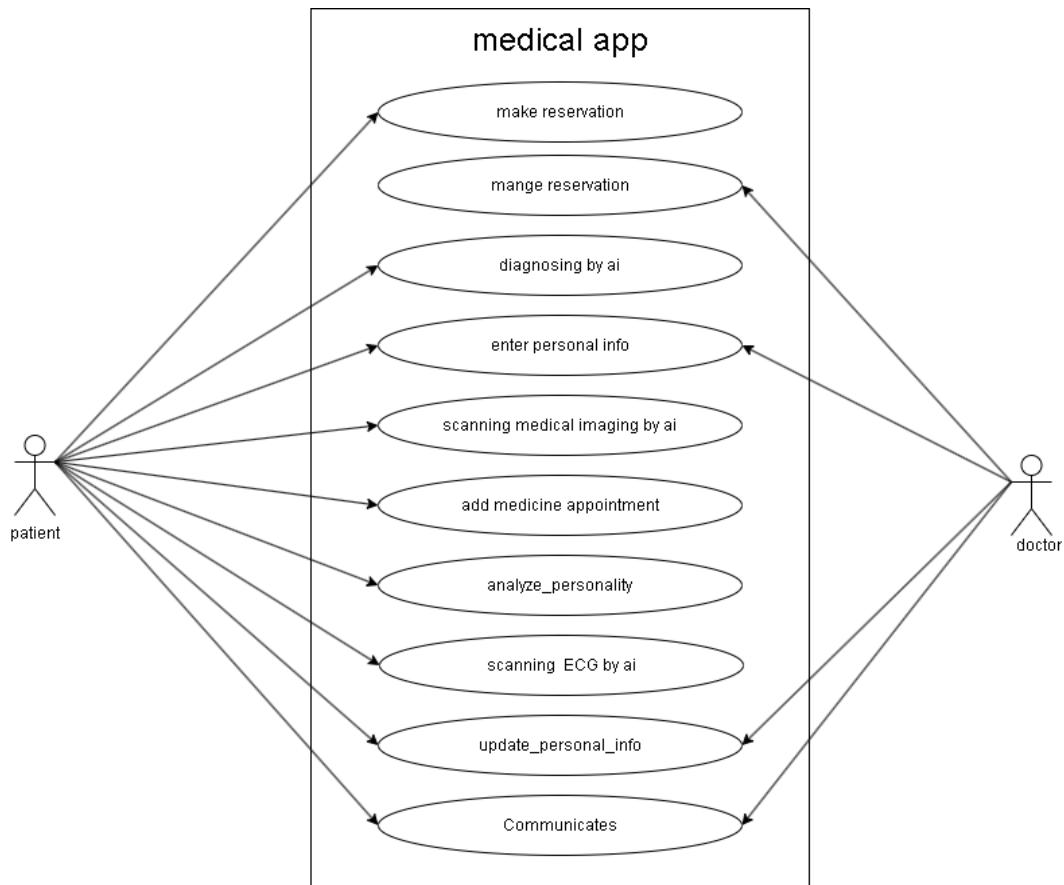


Figure 1.2

Sequence Diagram Log in:

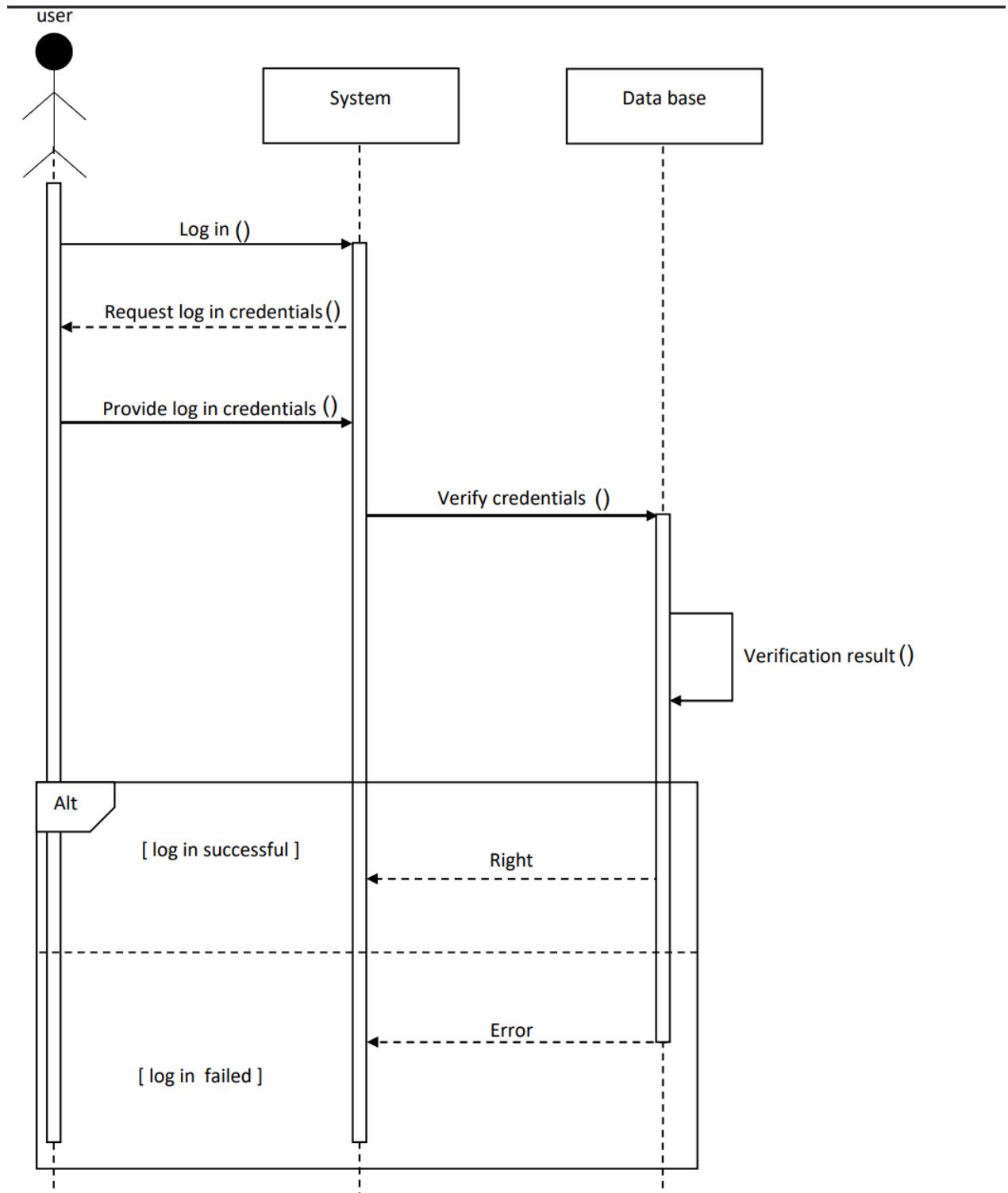


Figure 1.3

Sequence Diagram Sign up:

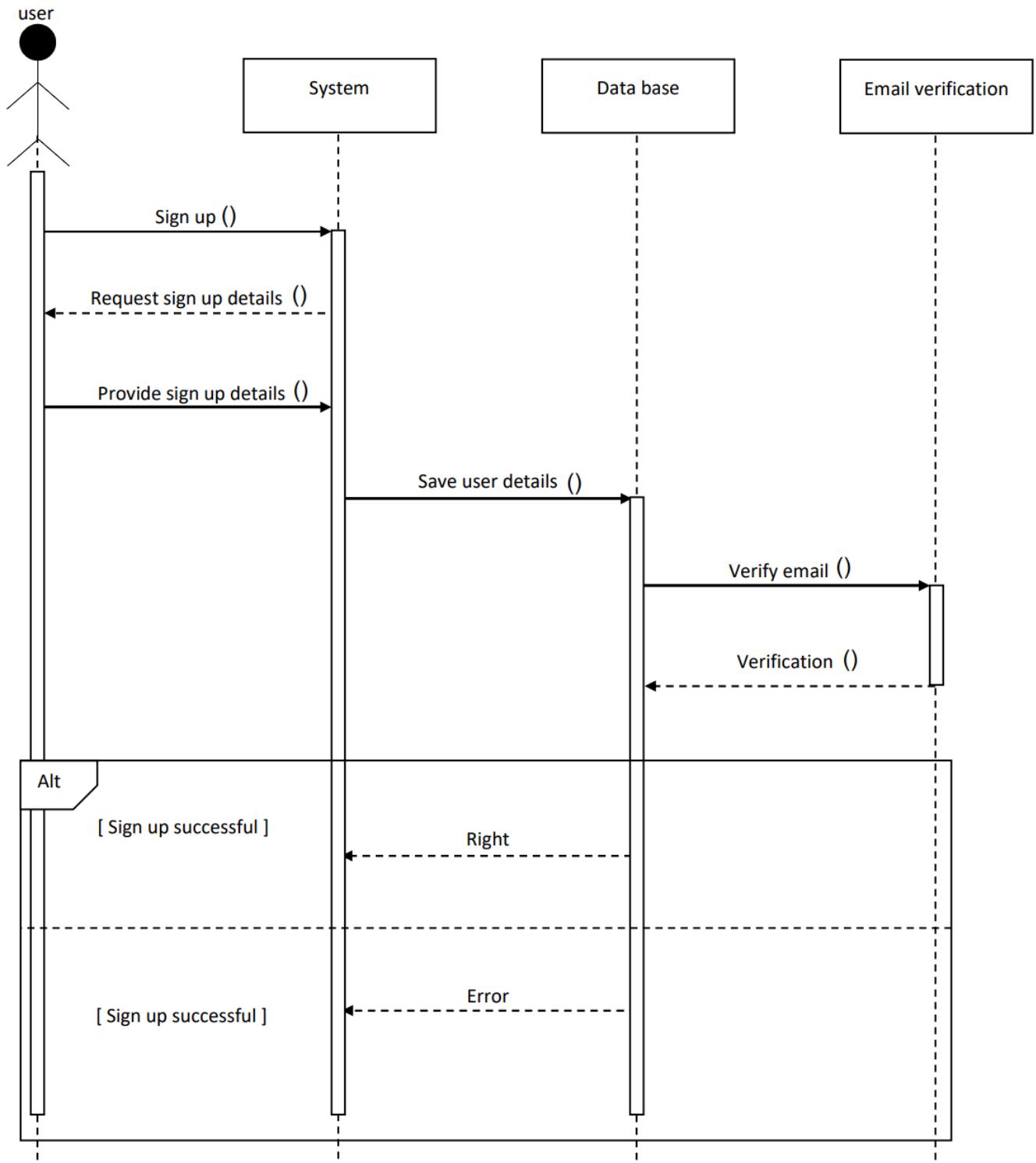


Figure 1.4

Sequence Diagram chat bot:

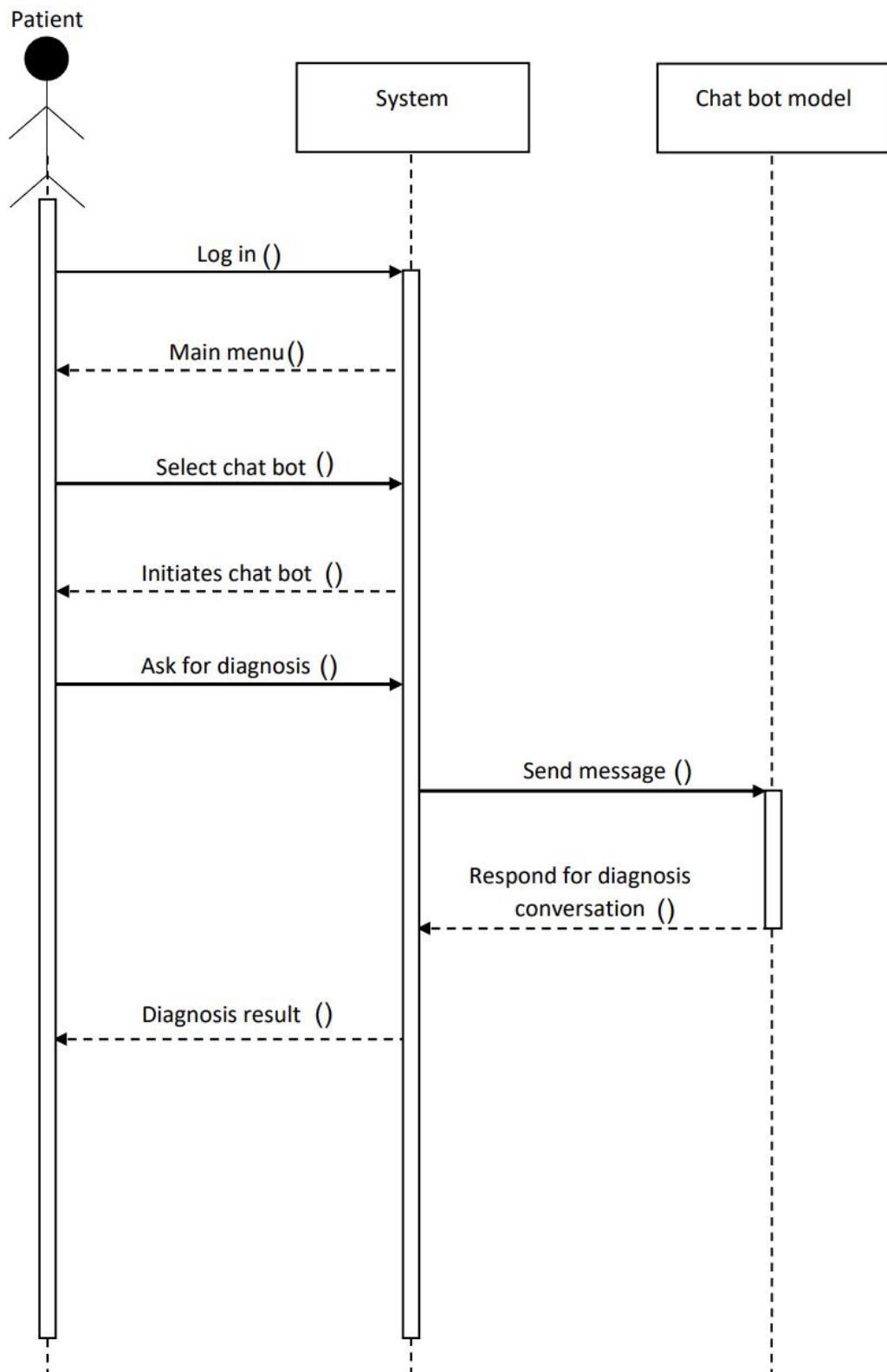


Figure 1.5

Sequence Diagram ECG Scan:

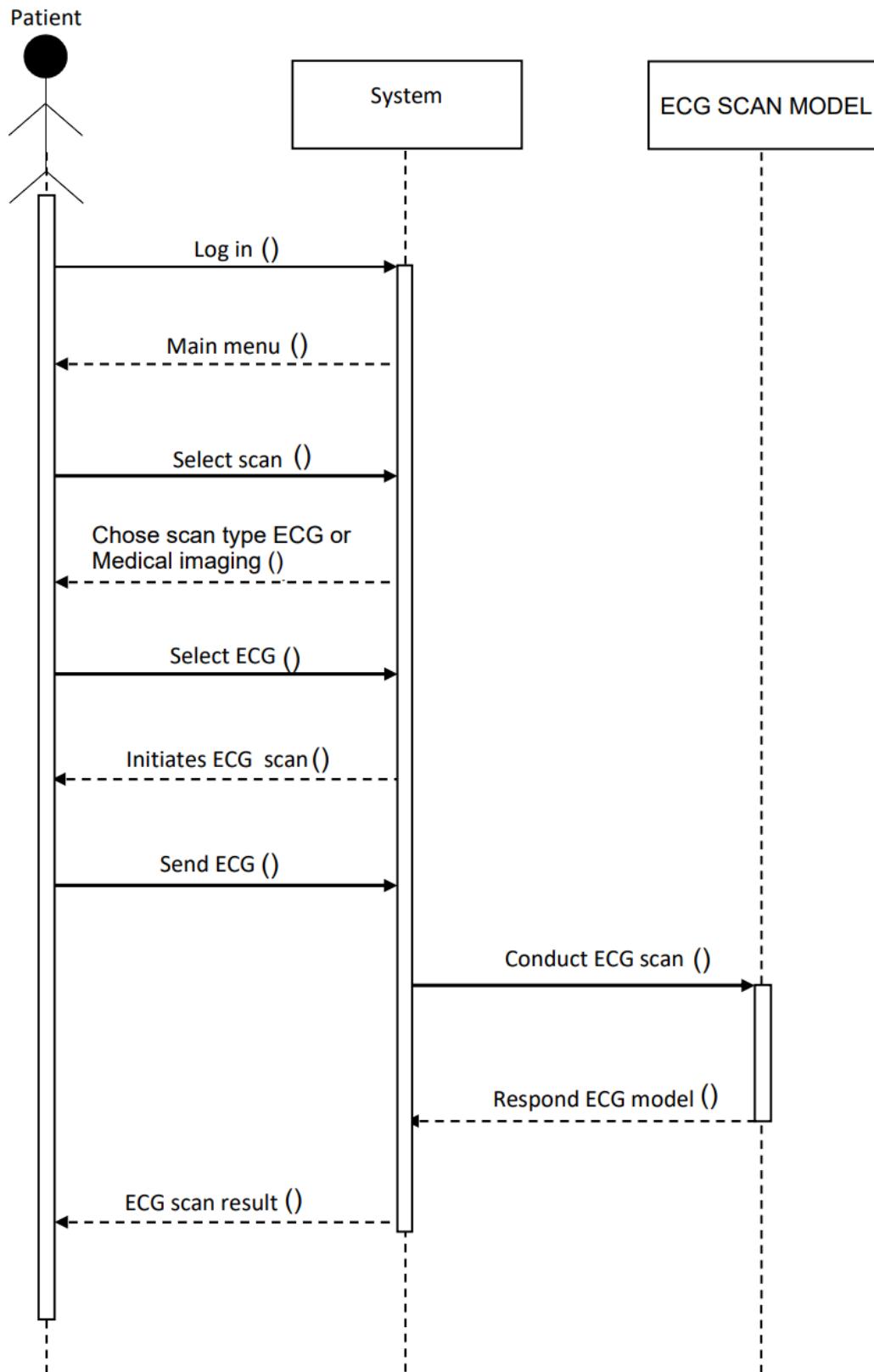


Figure 1.6

Sequence Diagram Medical Imaging Scan:

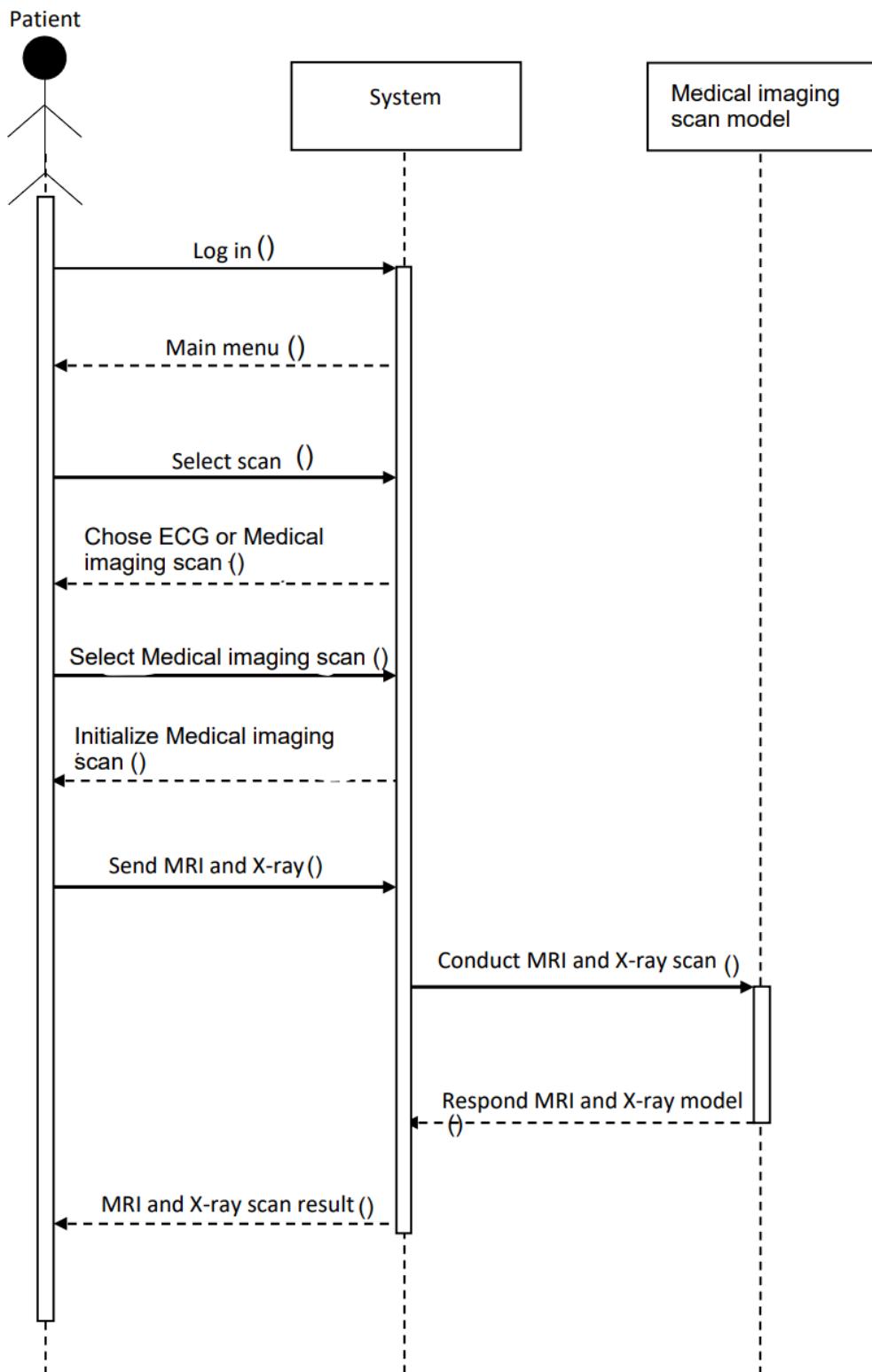


Figure 1.7

Sequence Diagram Personality Analyses:

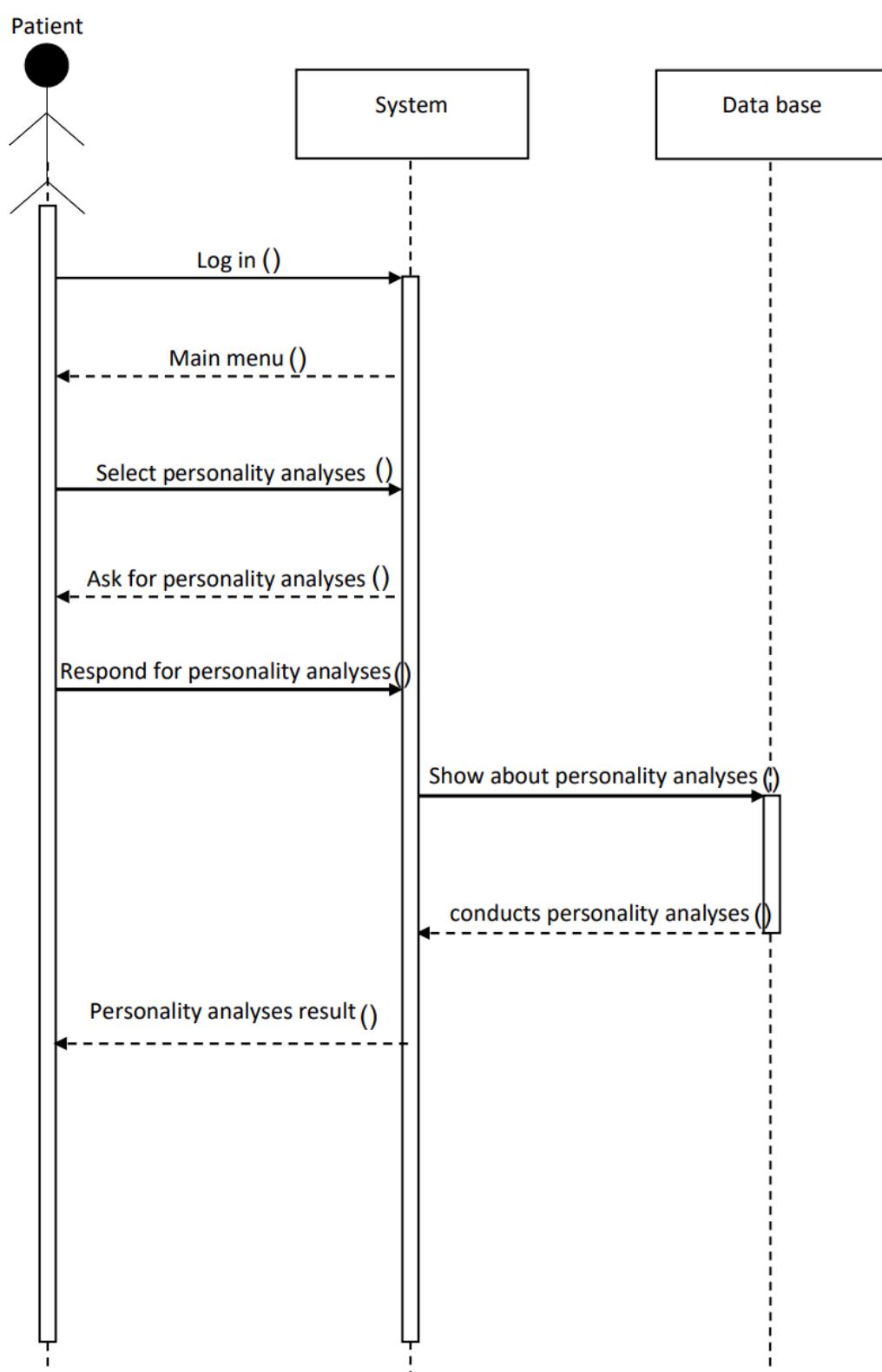


Figure 1.8

Sequence Diagram chat:

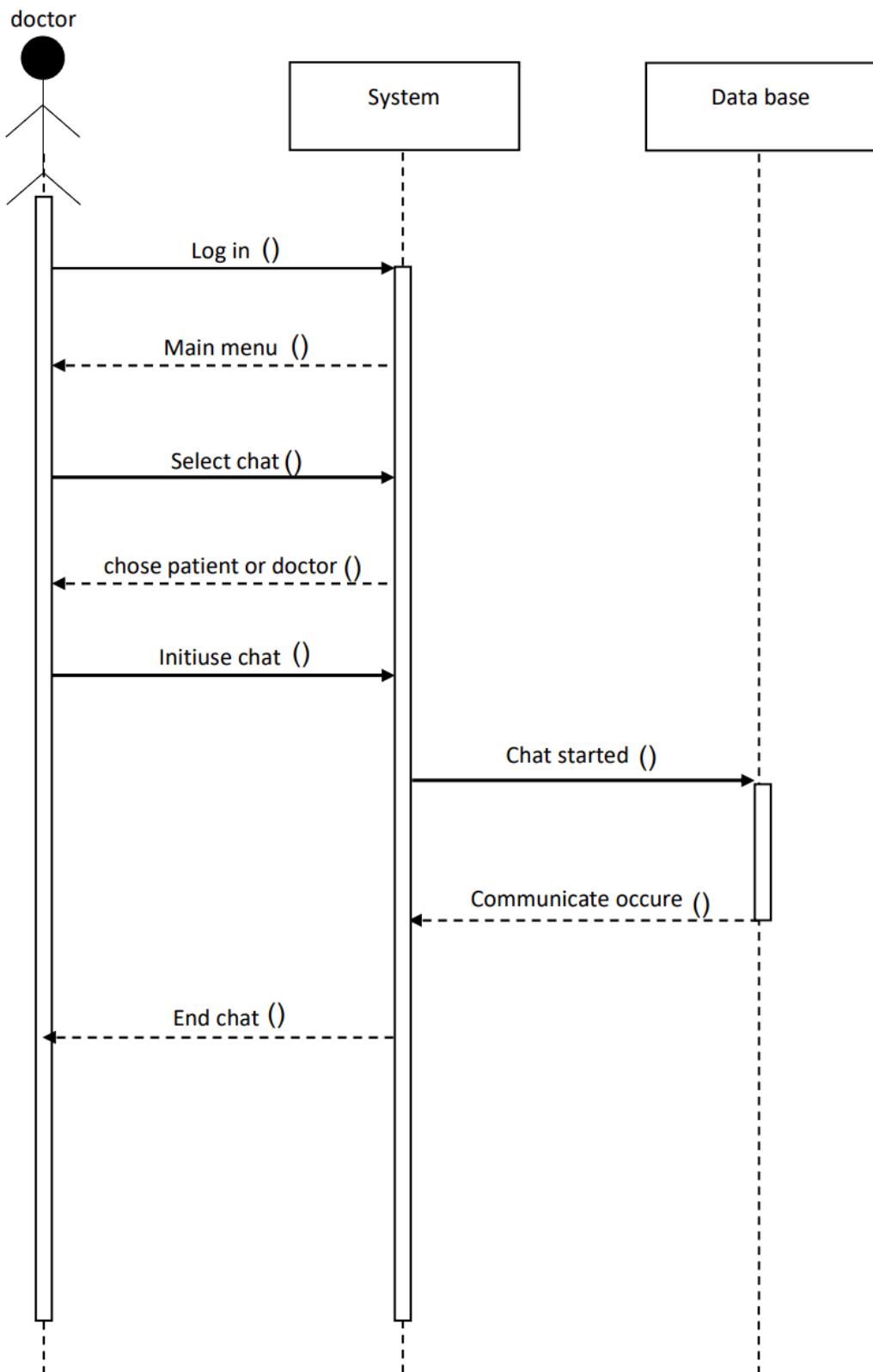


Figure 1.9

Sequence Diagram Laboratory Reservation:

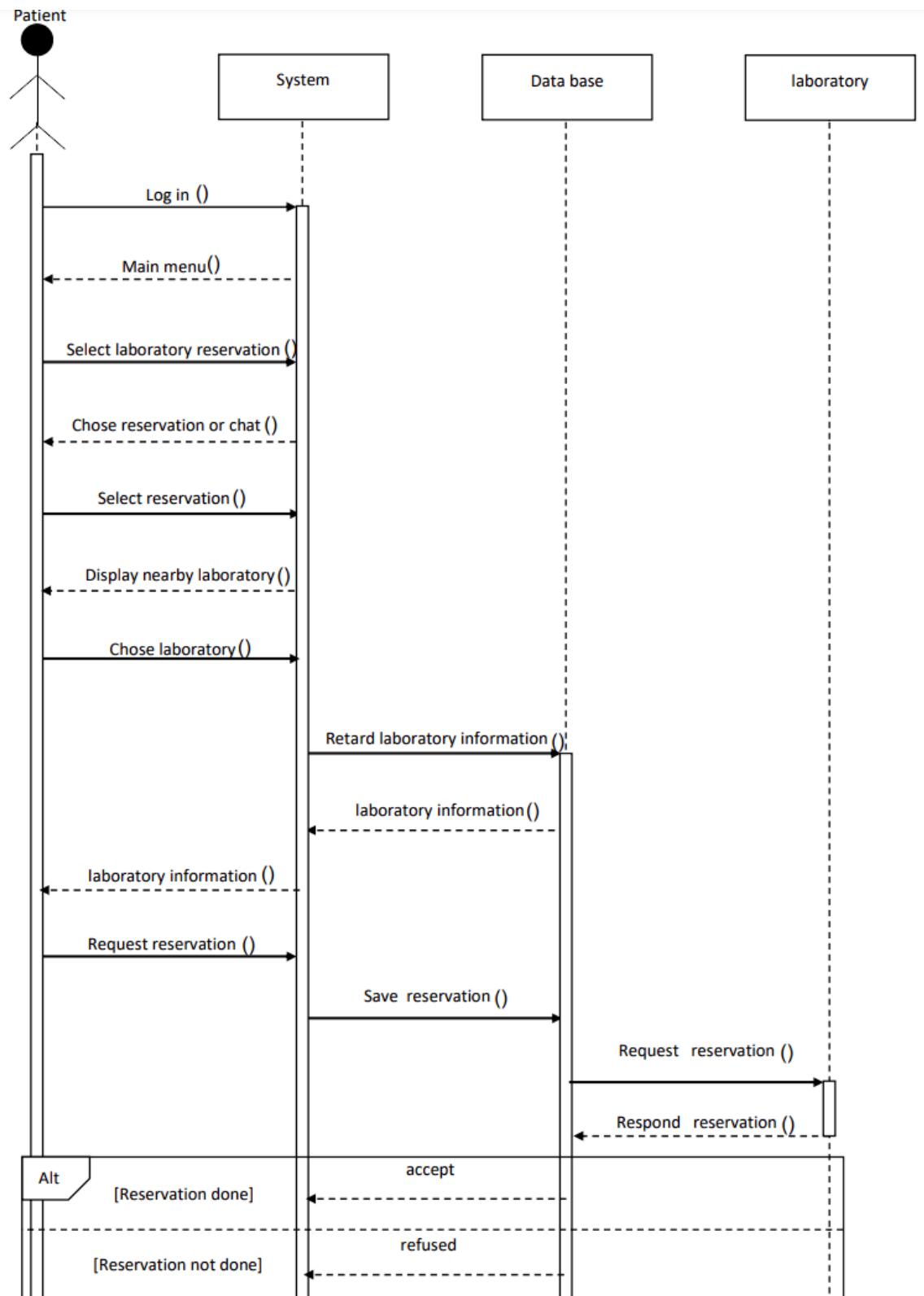


Figure 1.10

Sequence Diagram chat with pharmacy (Order medication):

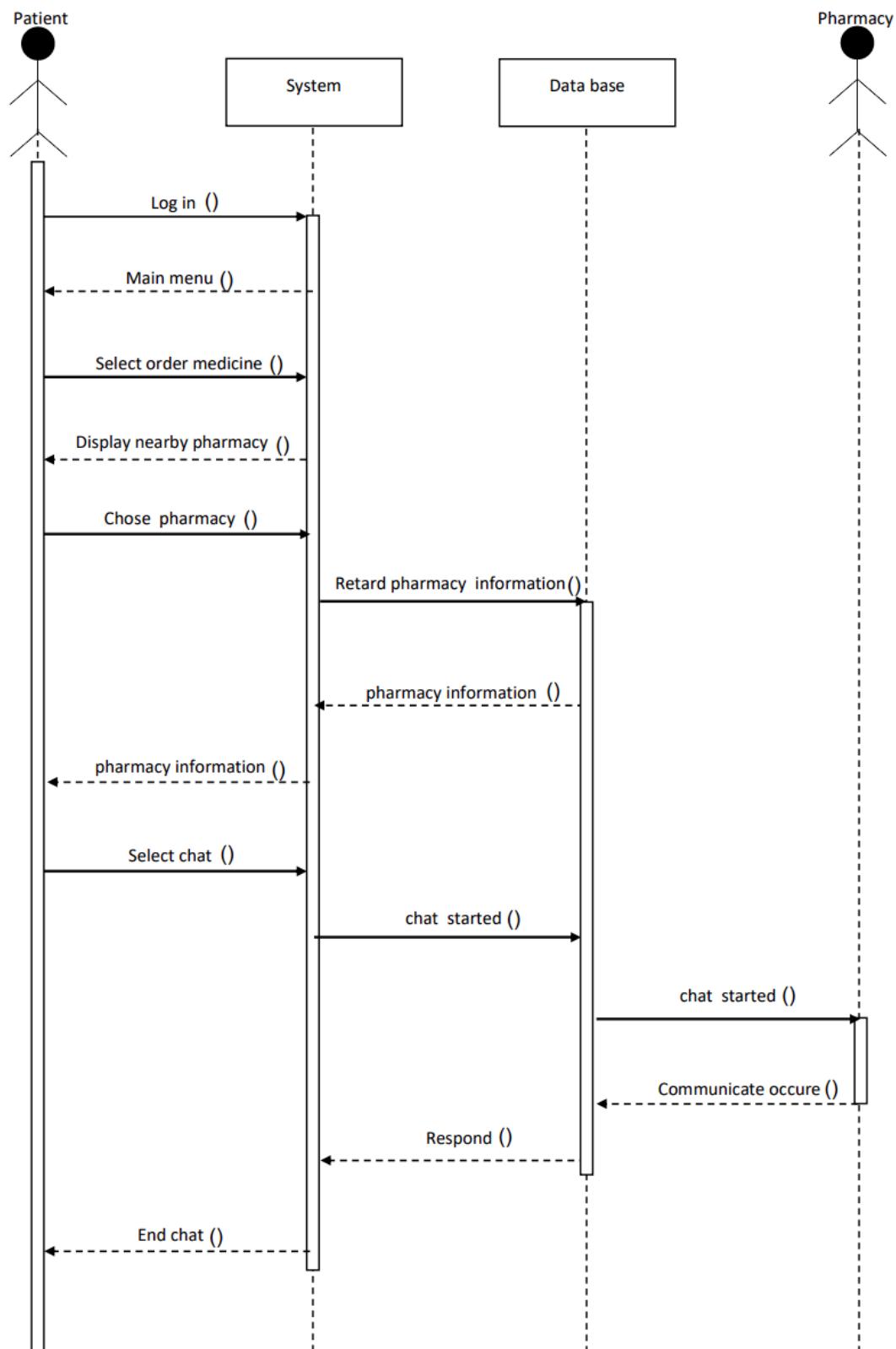


Figure 1.11

Sequence Diagram Clinic Reservation:

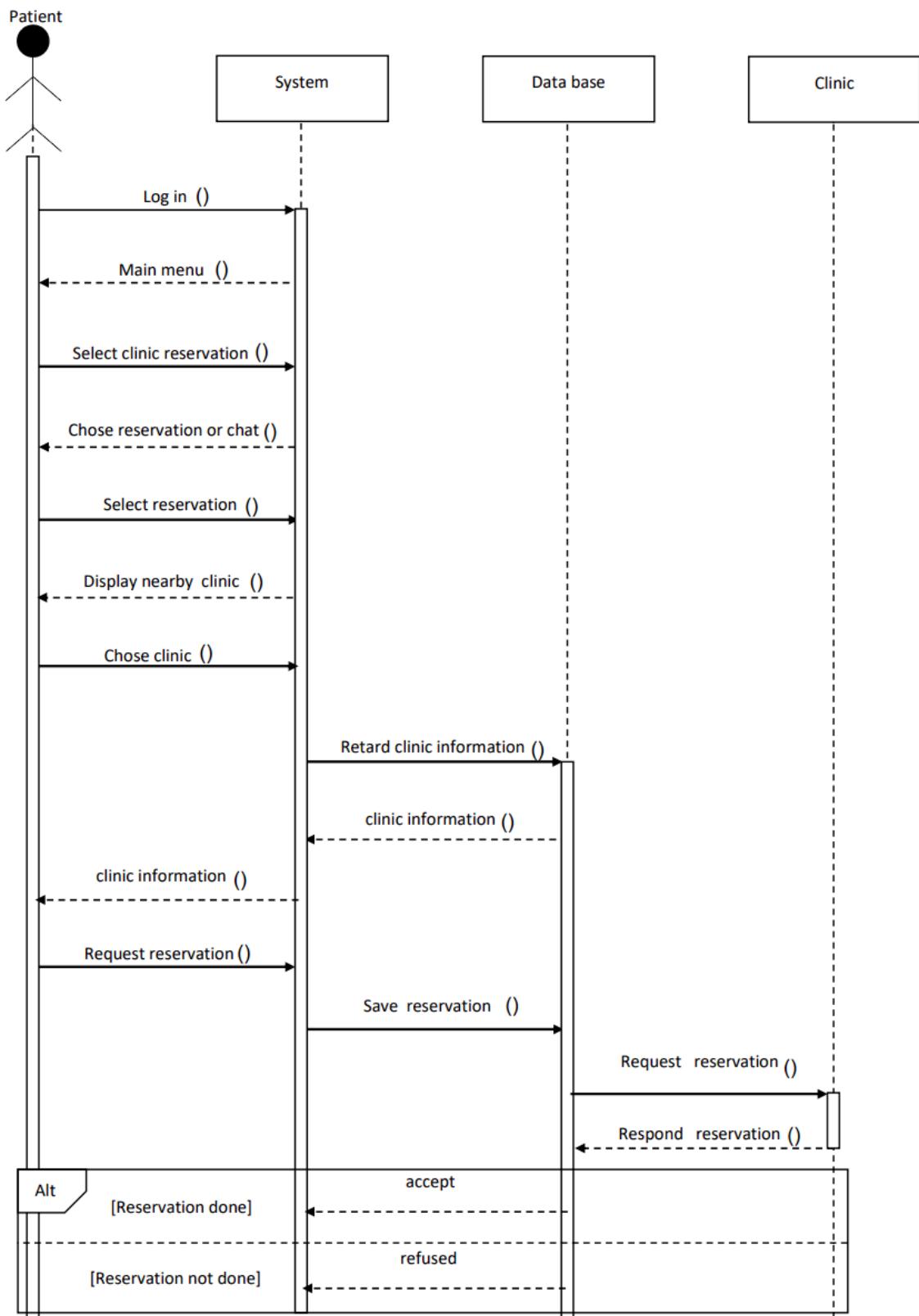


Figure 1.12

Sequence Diagram Accept Reservation:

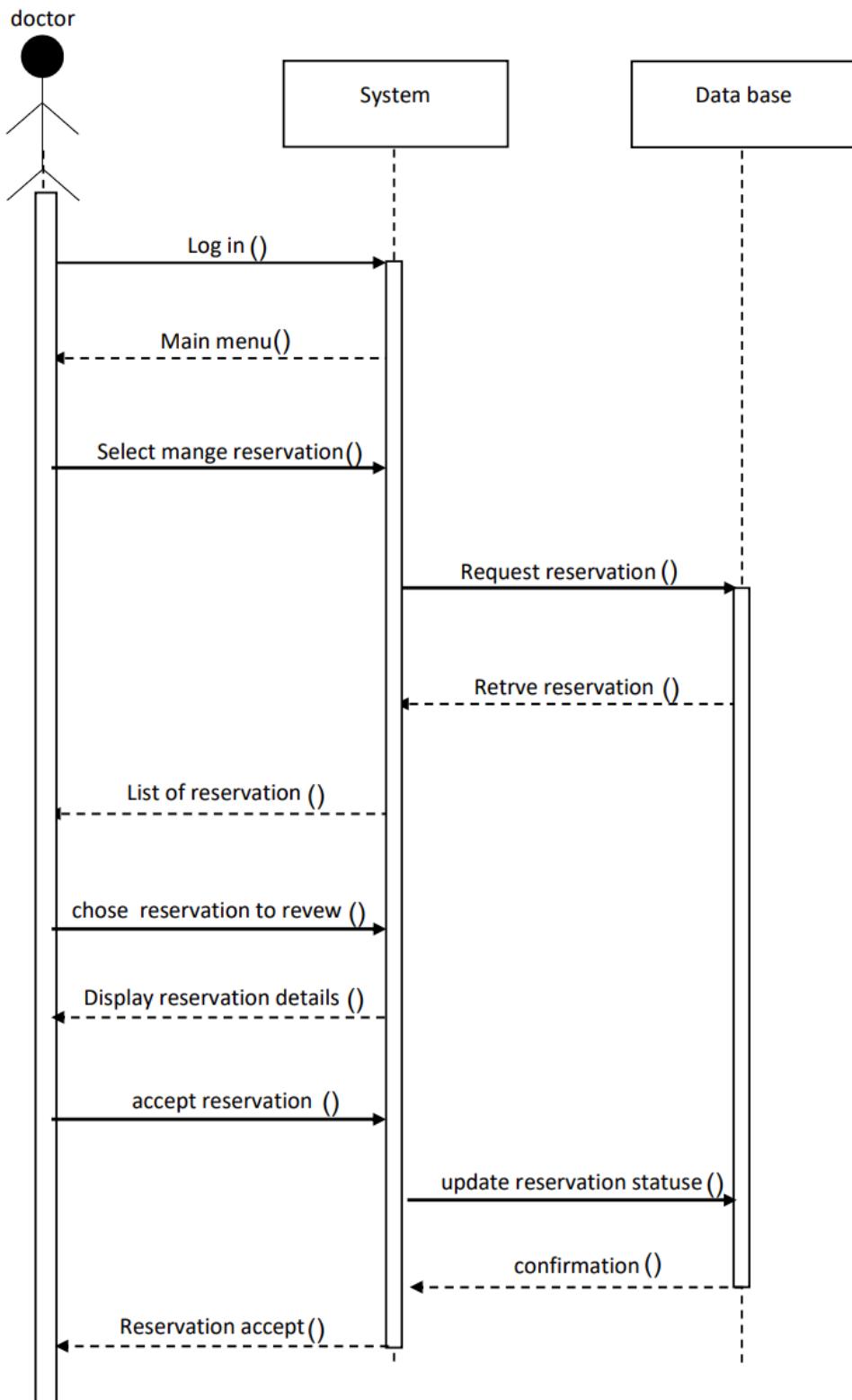


Figure 1.13

Sequence Diagram Refuse Reservation:

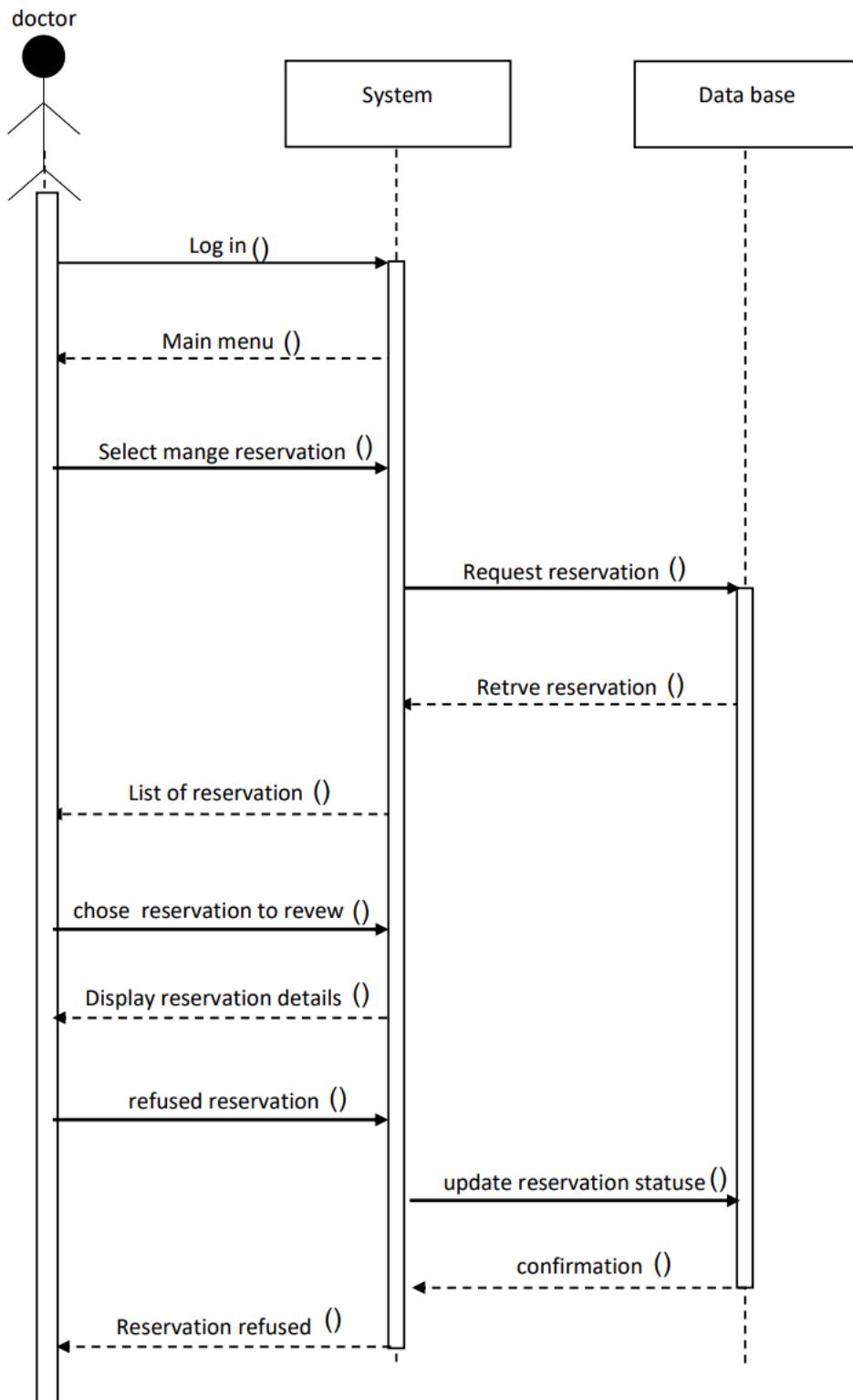


Figure 1.14

Patient Activity Diagram:

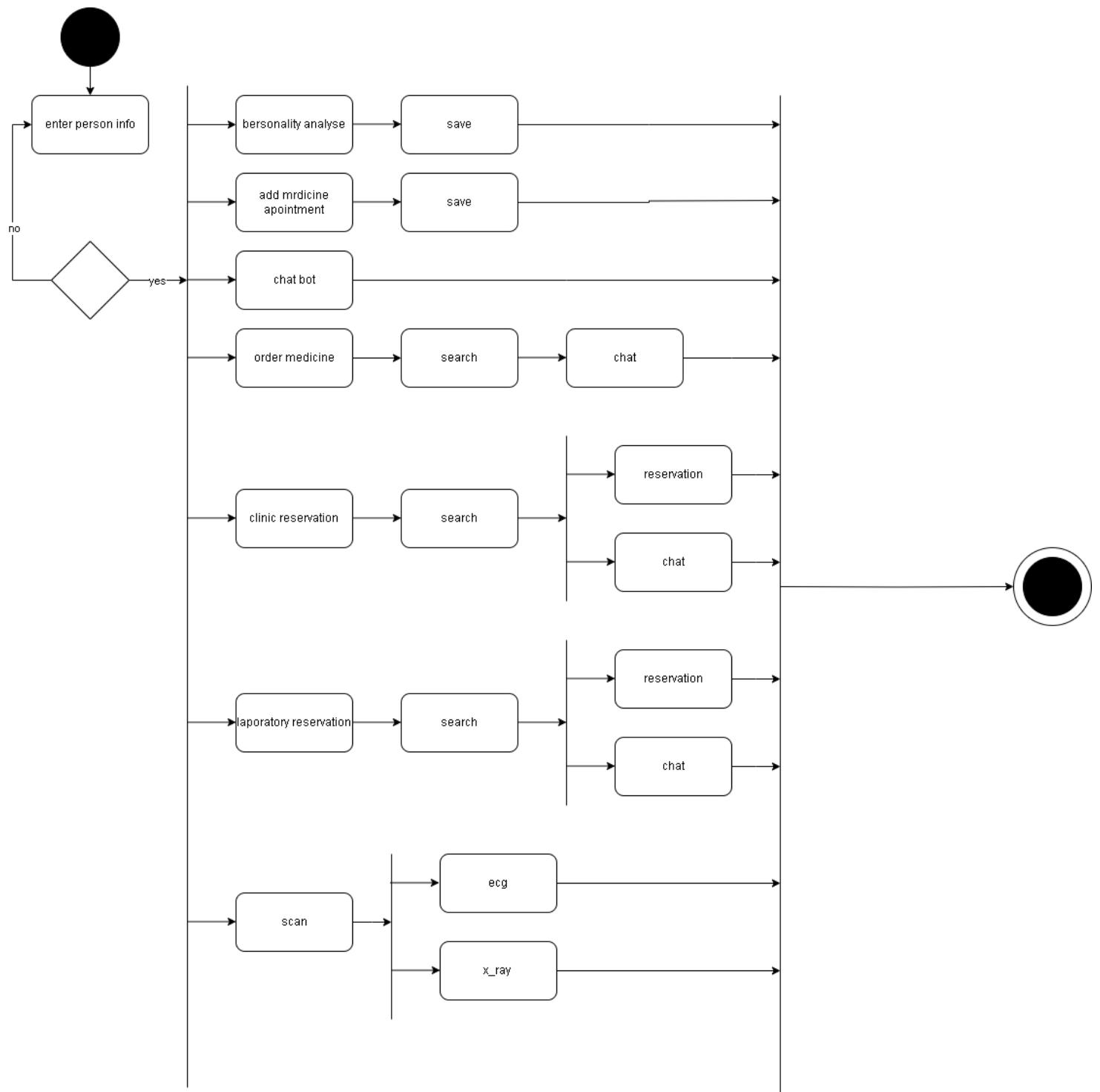


Figure 1.15

Doctors Activity Diagram:

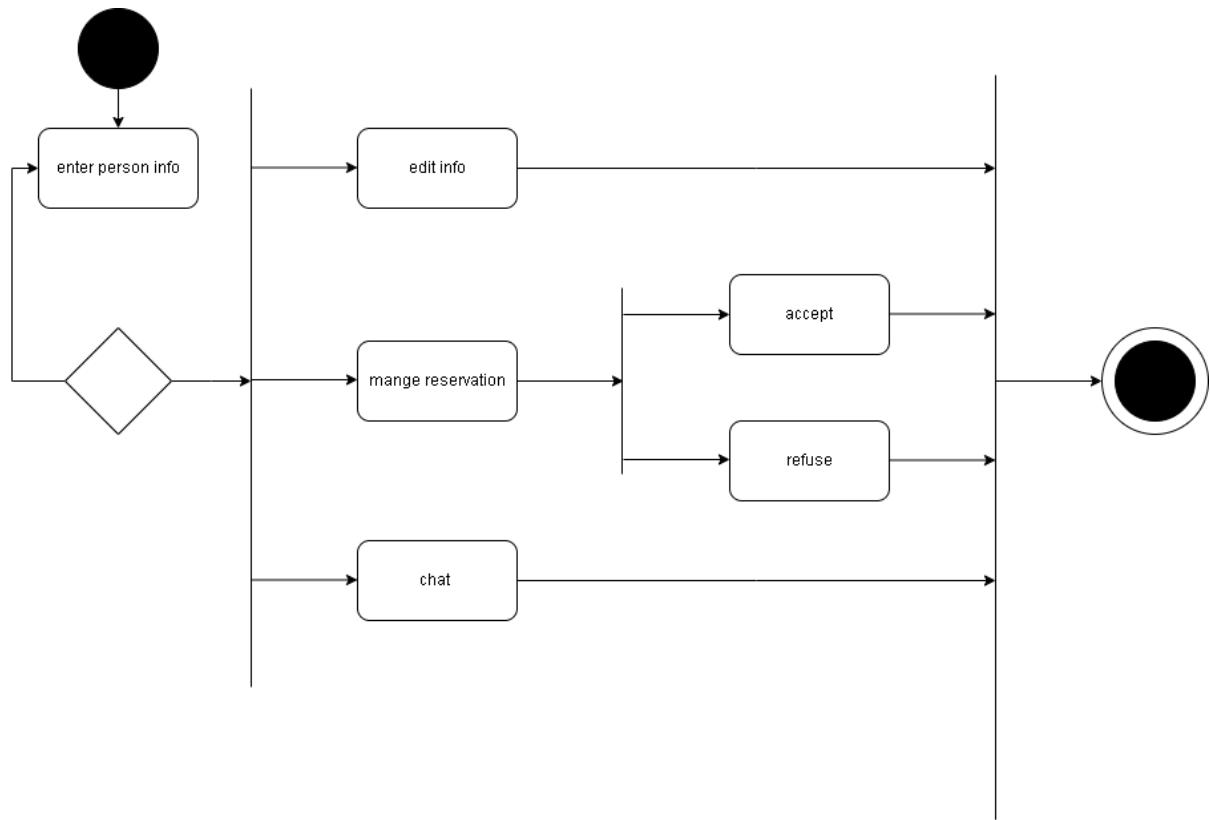


Figure 1.16

Entity Relationship Diagram ERD:

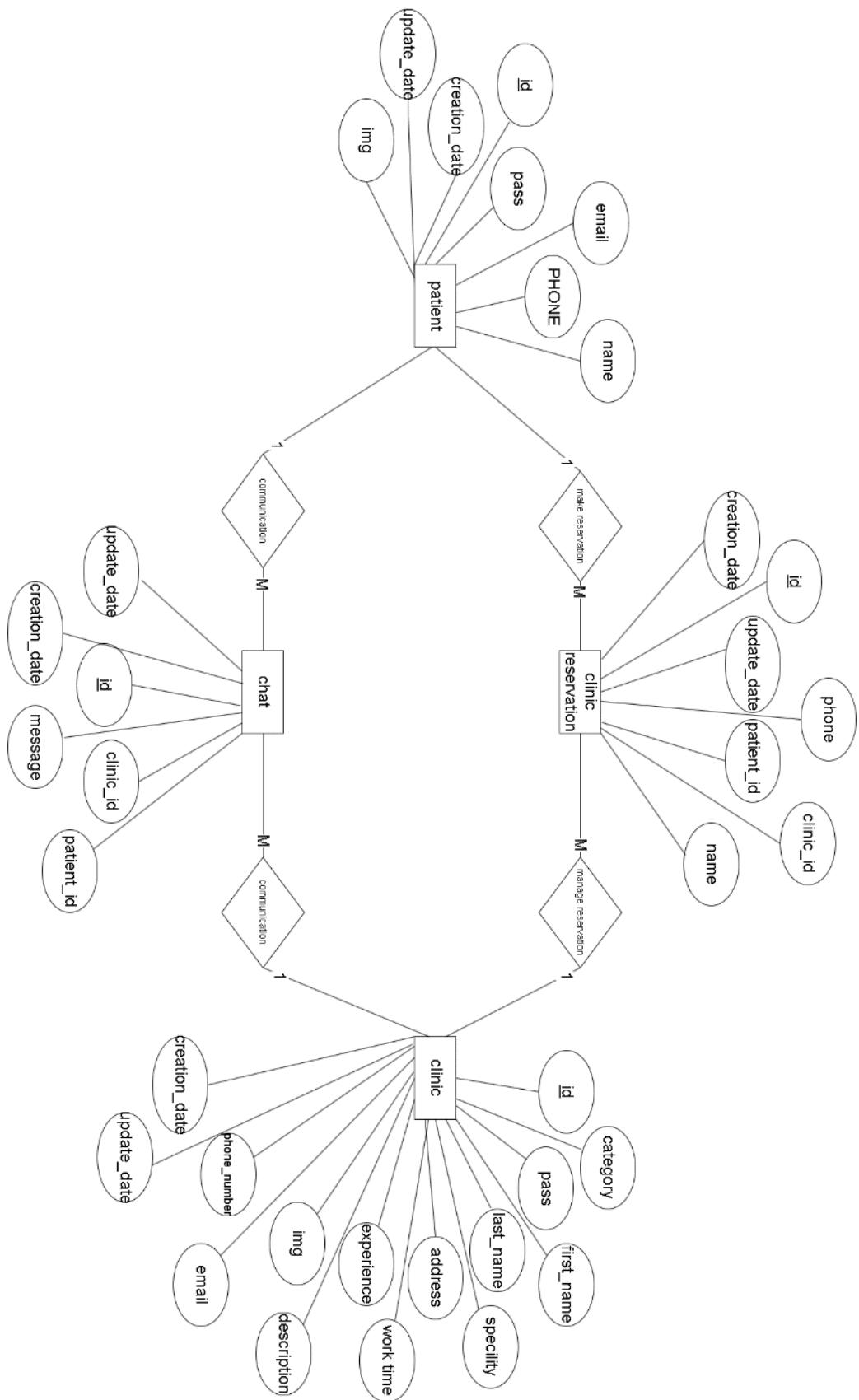


Figure 1.17

Class Diagram:

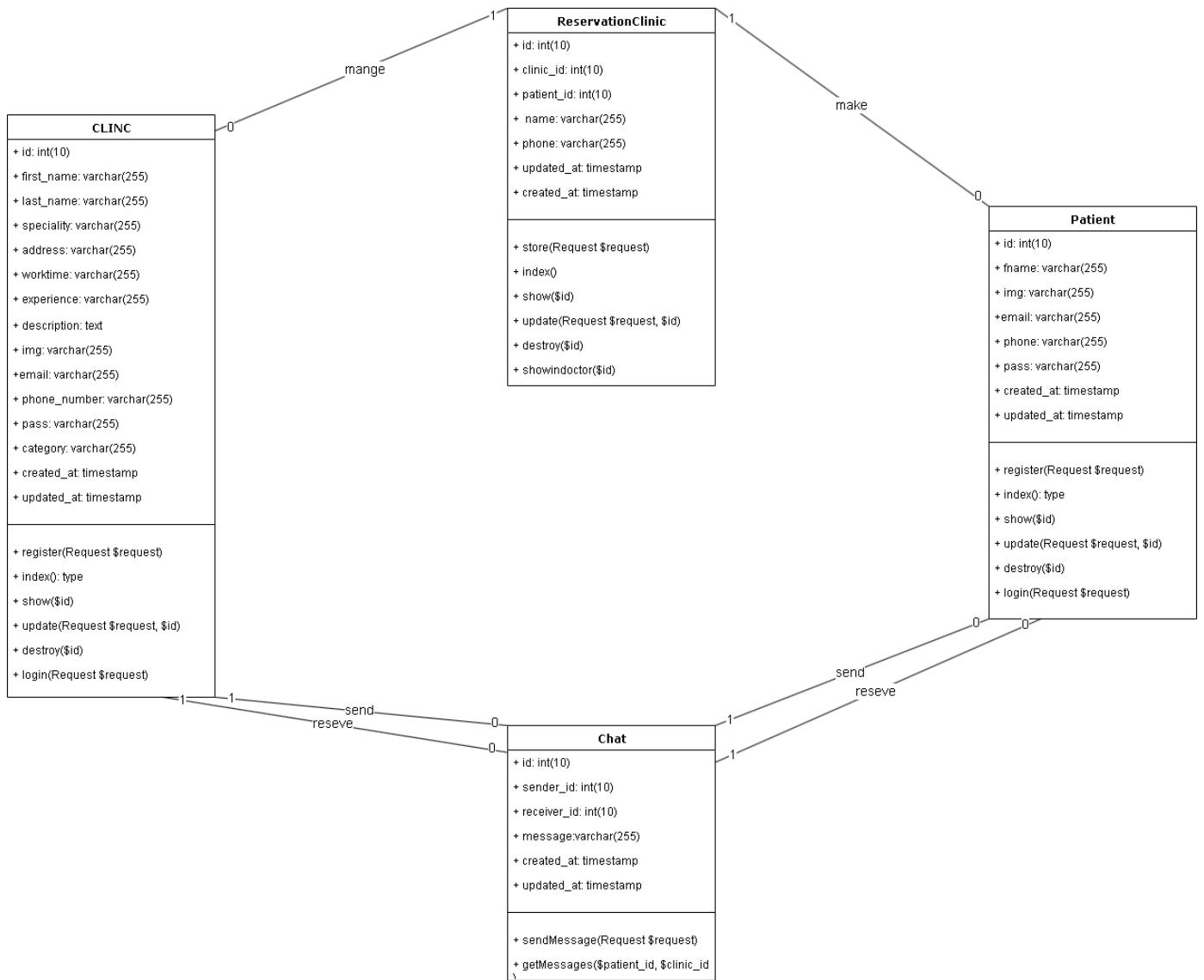


Figure 1.18

Chapter 3: System Design

2. System Design

2.1 User Interfaces: Mobile application



Figure 2.1: Bash screen

An introduction screen appears while loading the application

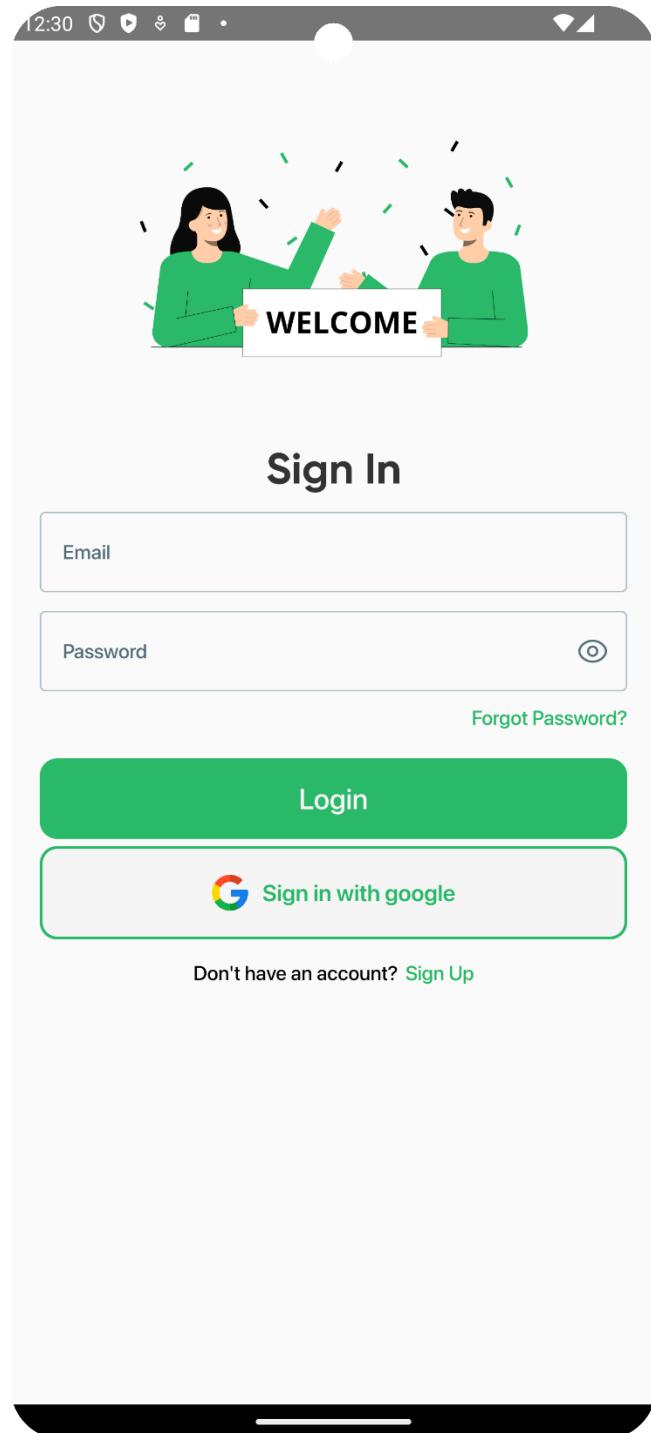


Figure 2.2: Login screen

The welcome page has sections for your login information, your login and register also a forgotten password option in case the user forgets its password he can to verify itself by sending a verification code to its email.

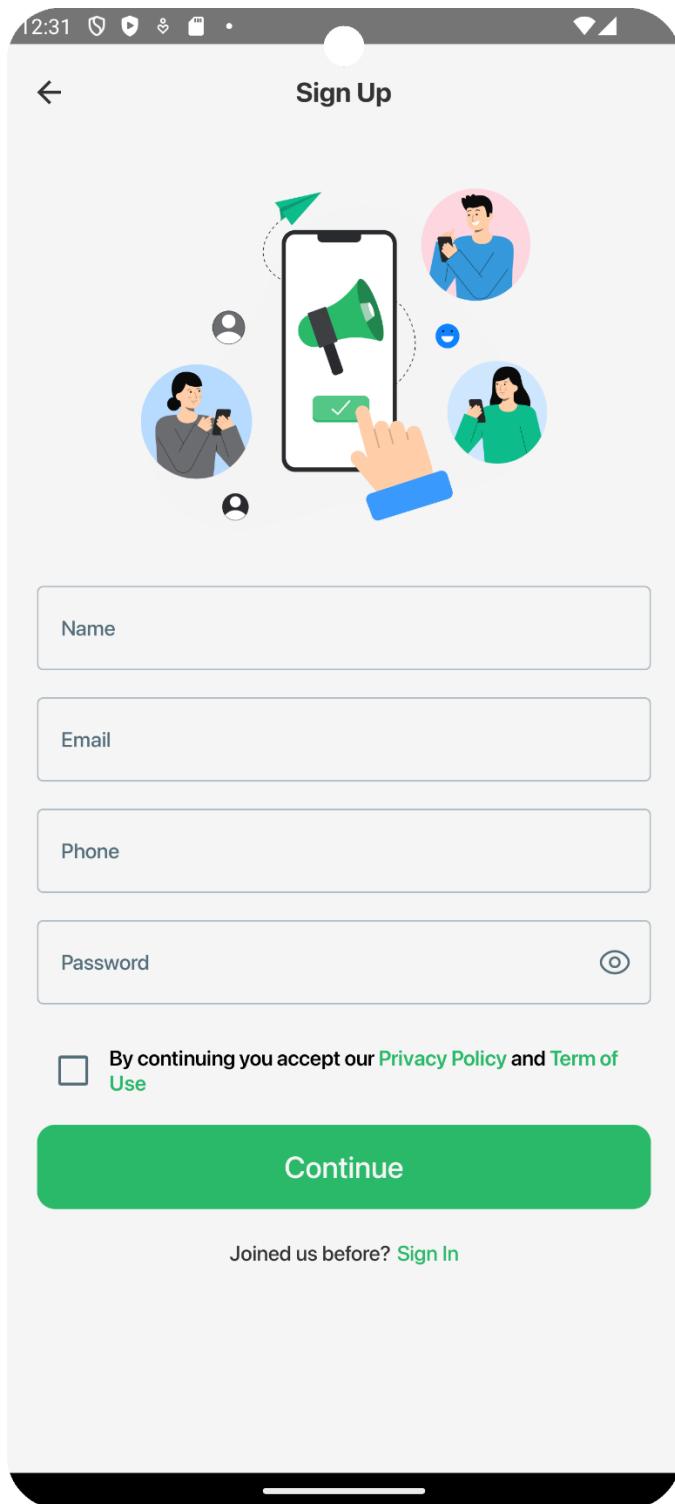


Figure 2.3: Sign up screen

The user can register into the application for the first time on the Sign-Up page. He or she must enter their name, email address, and password confirmation. Next, select "Continue" to handle his or her account.

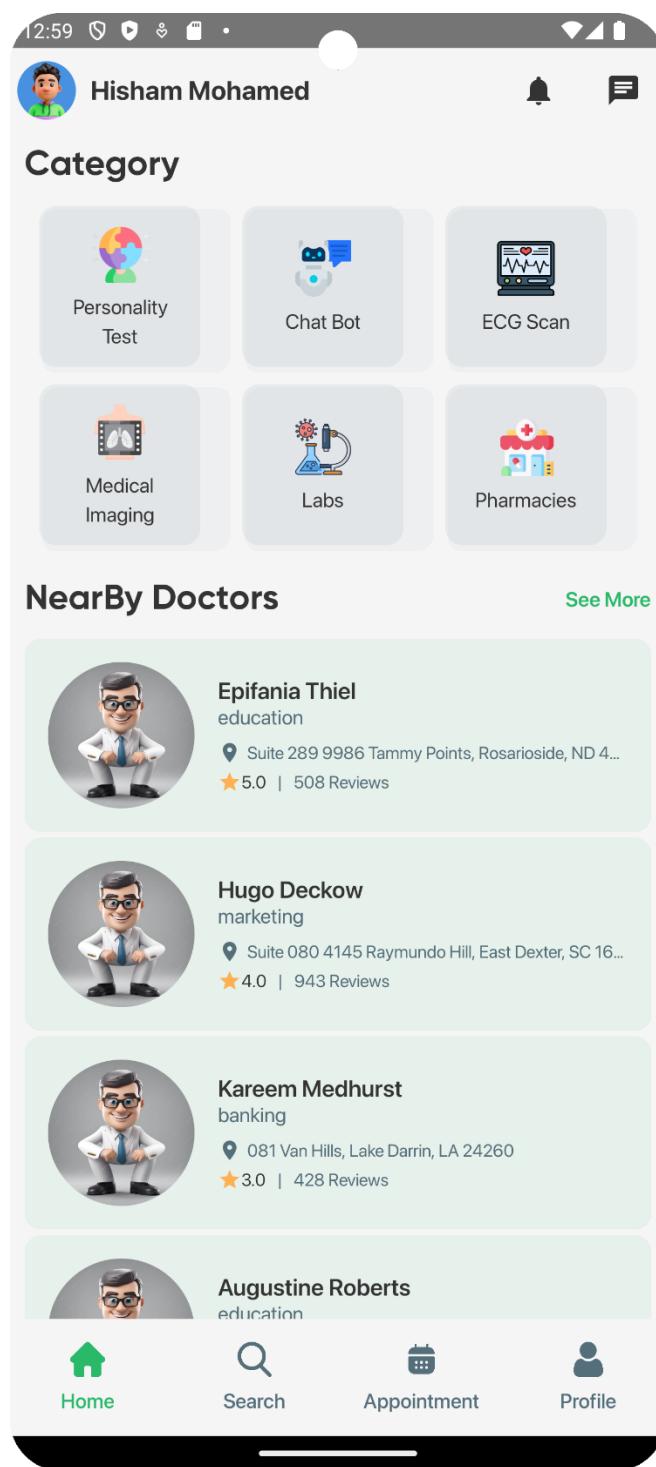


Figure 2.4: Home page

The home page that contains several categories and features as making a personality test, or make a smart diagnosis with chatbot, or scan a medical imaging scan image, booking an appointment with clinics, pharmacies, or a purchase medication from a nearby pharmacy with the ability for searching

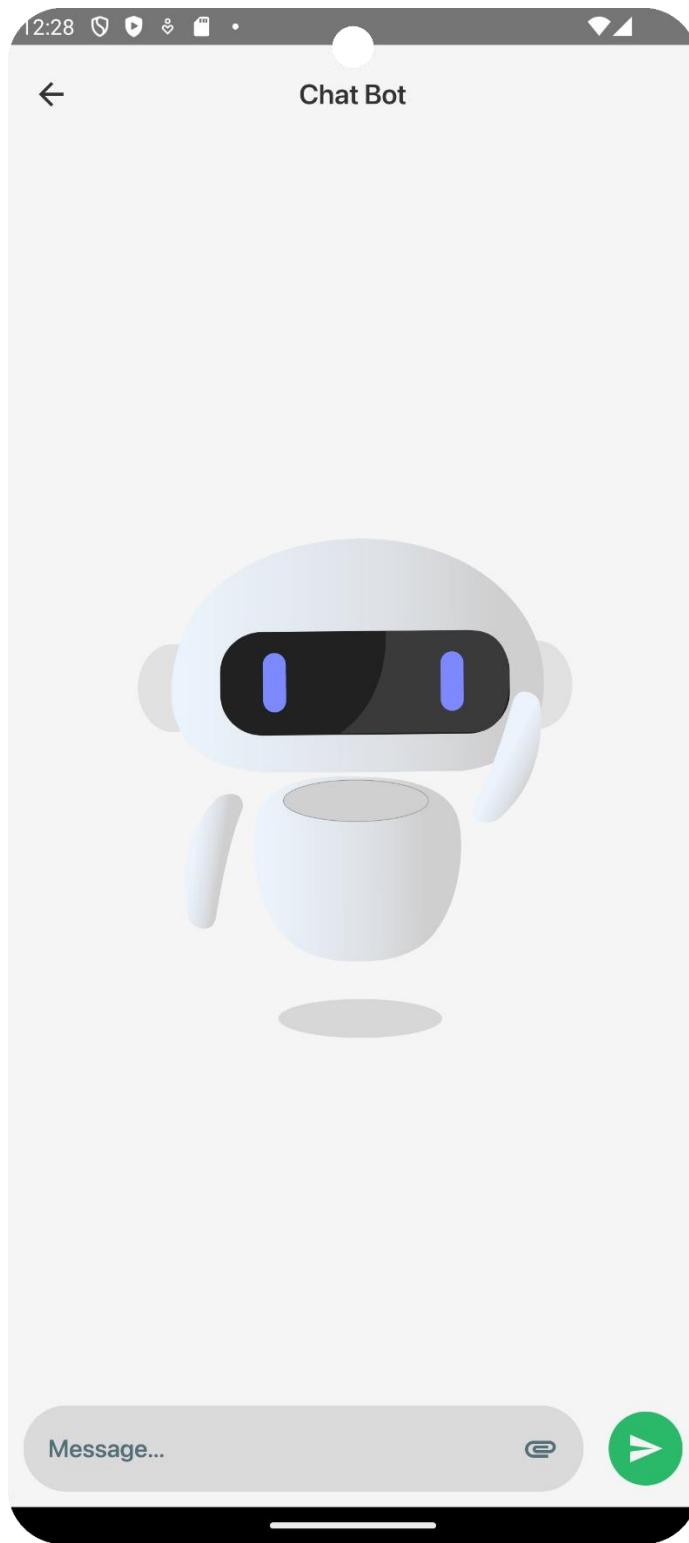


Figure 2.5: Chatbot for diagnosis blank screen

Patient can send text messages and get the response from the chatbot

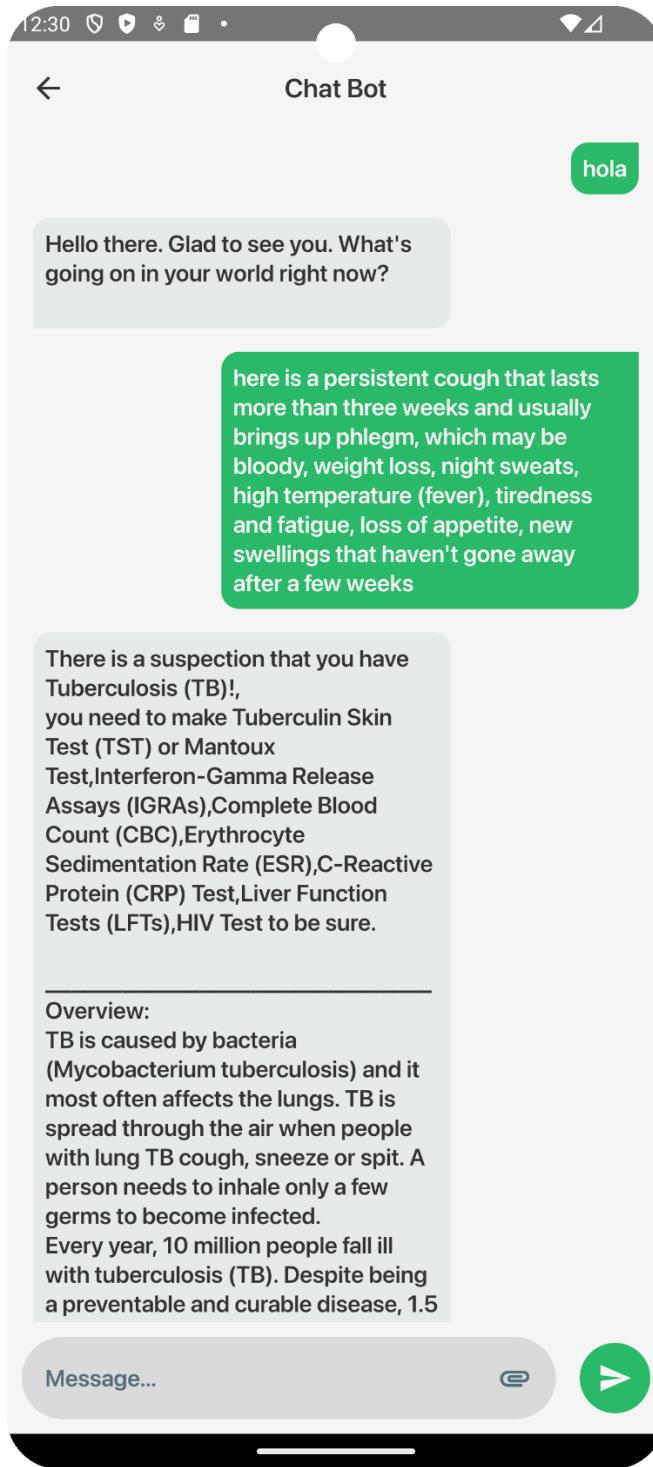


Figure 2.6: Chatbot for diagnosis screen

The patient applying a medical diagnosis by sending his disease' symptoms for the chatbot and the chatbot detect the diseases and response with information about it and recommend tests and scans to ensure that the diseases diagnosis was accurate.

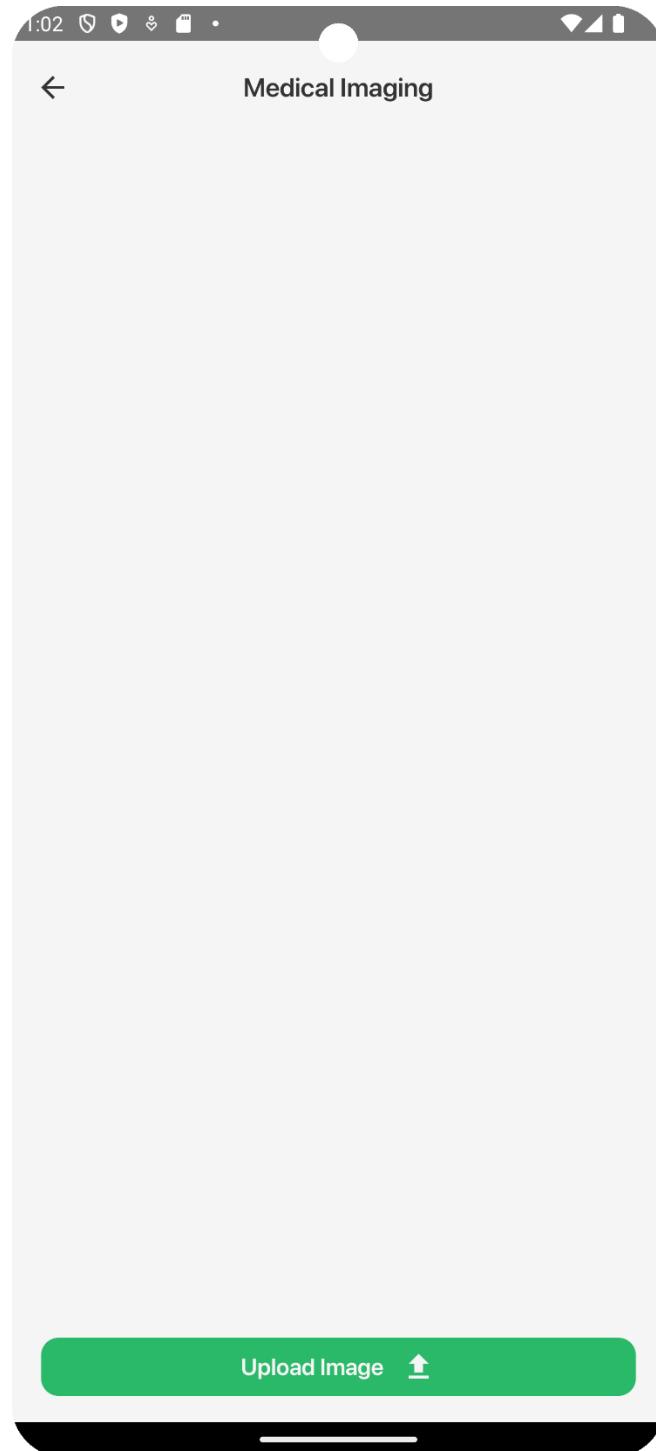


Figure 2.7: Medical imaging scan blank screen

Patient can upload an image and the scan models will send response

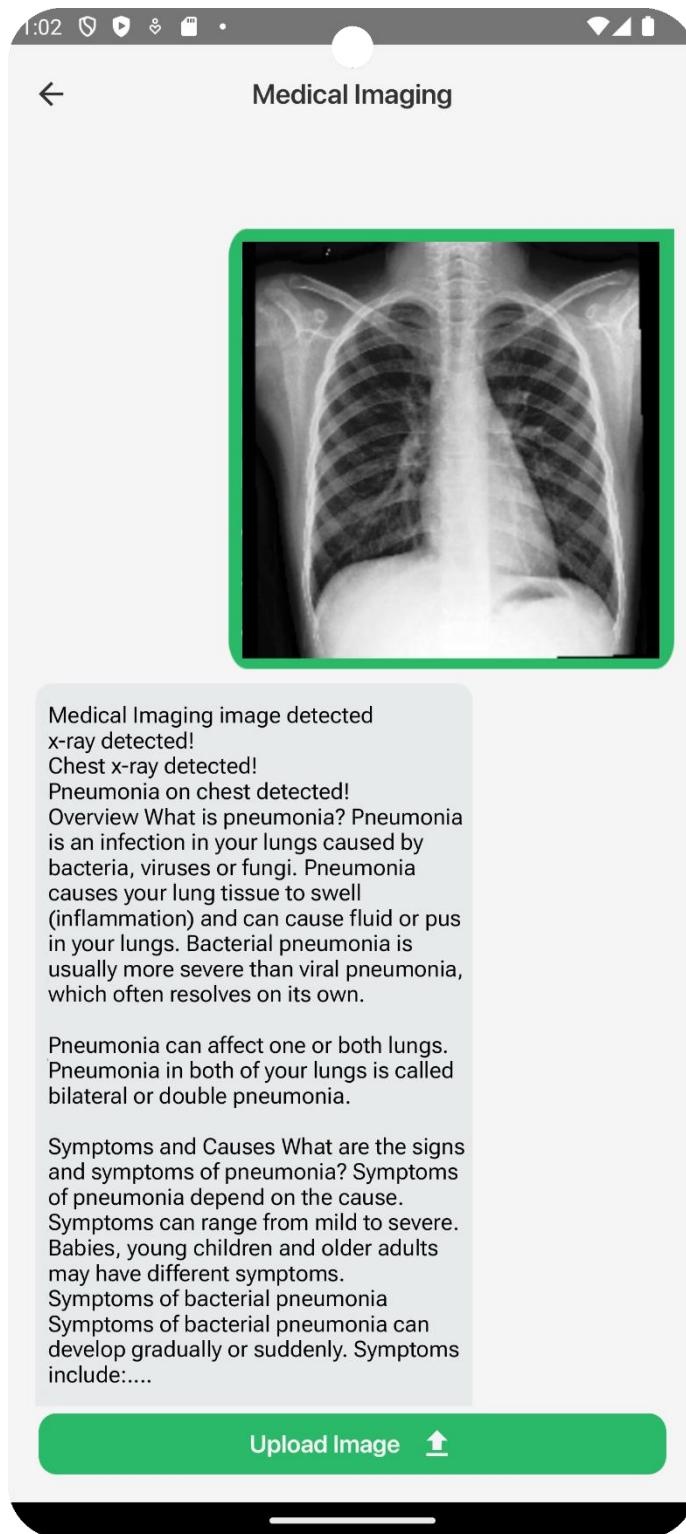


Figure 2.8: Medical imaging scan screen

Patient upload a medical imaging scan image and the scan models detect the image, imaging type, on which body part and diagnose the disease, and give him the diagnosis response

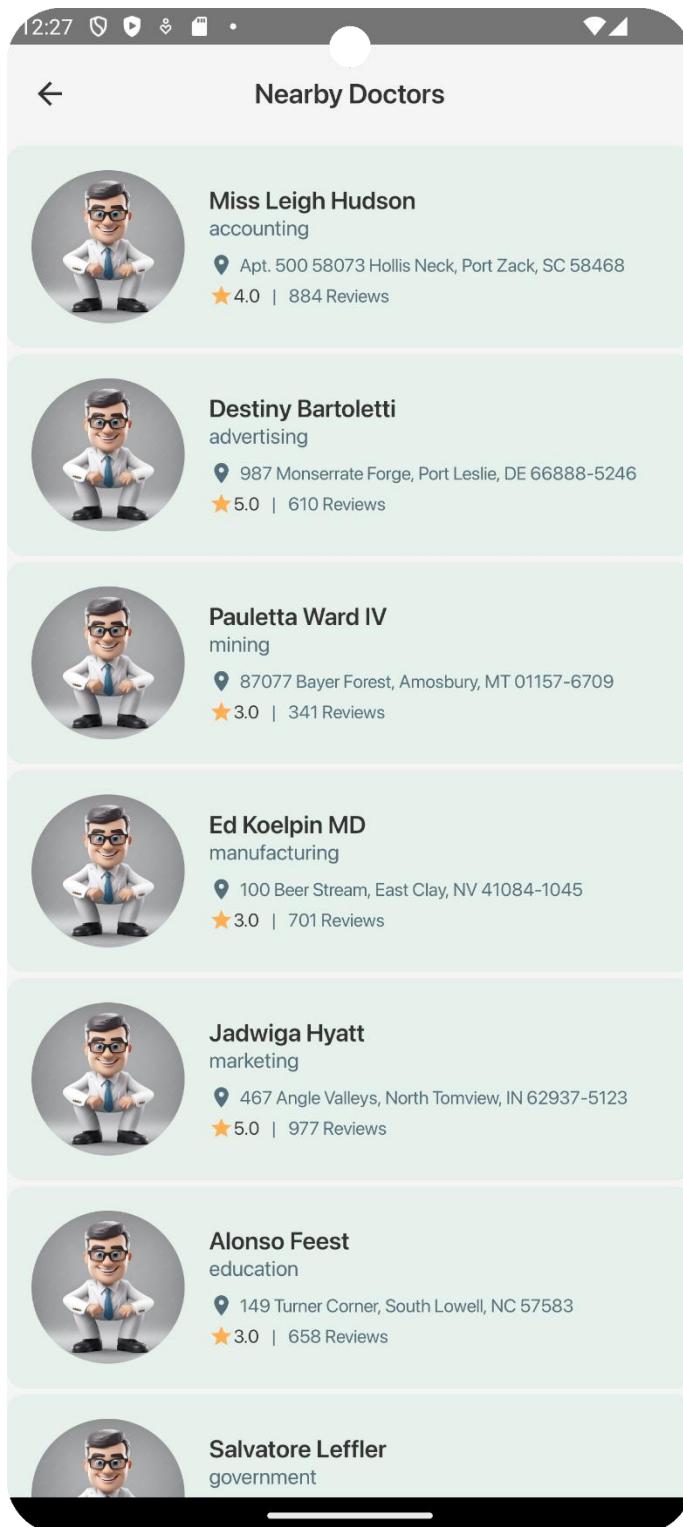


Figure 2.9: Booking doctor' appointment screen

Patient looking for the nearby doctors' clinics arranged by the nearest location

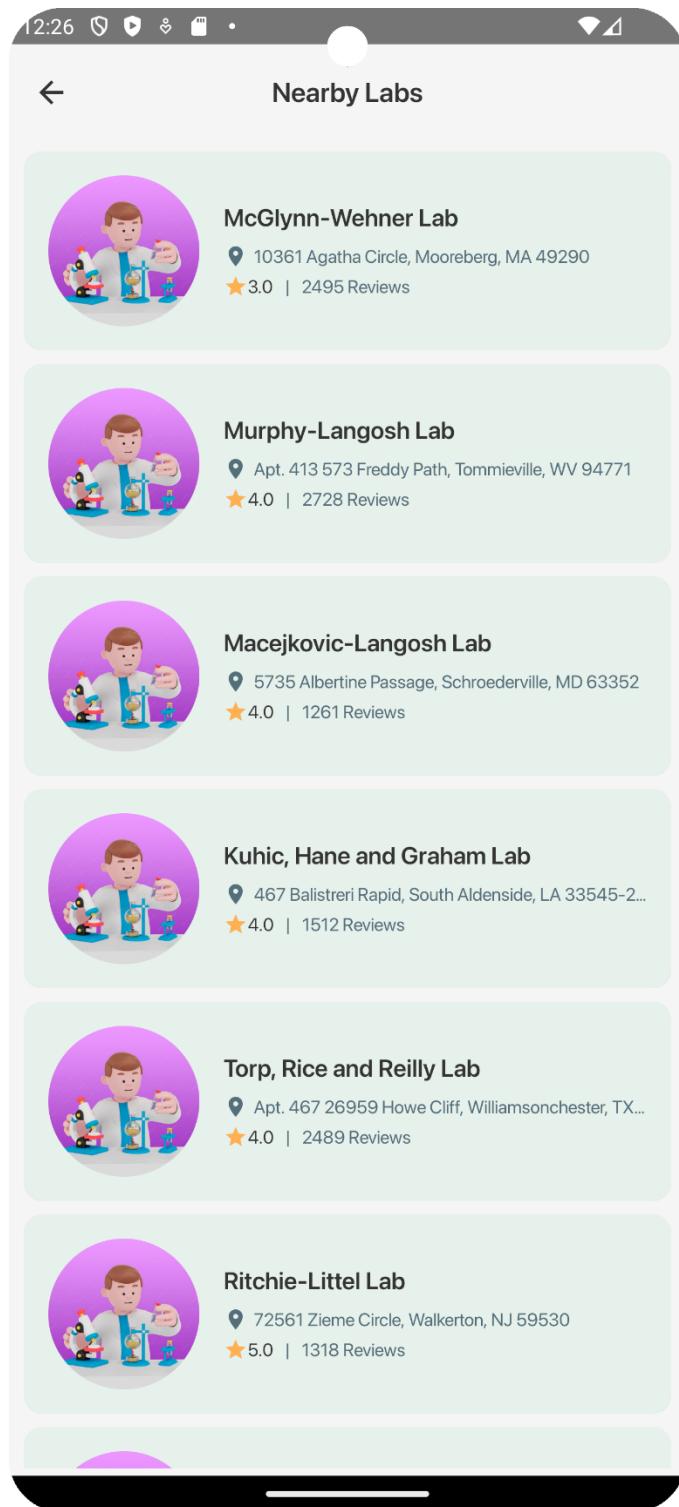


Figure 2.10: Booking laboratory' appointment screen

Patient looking for the nearby doctors' laboratories arranged by the nearest location

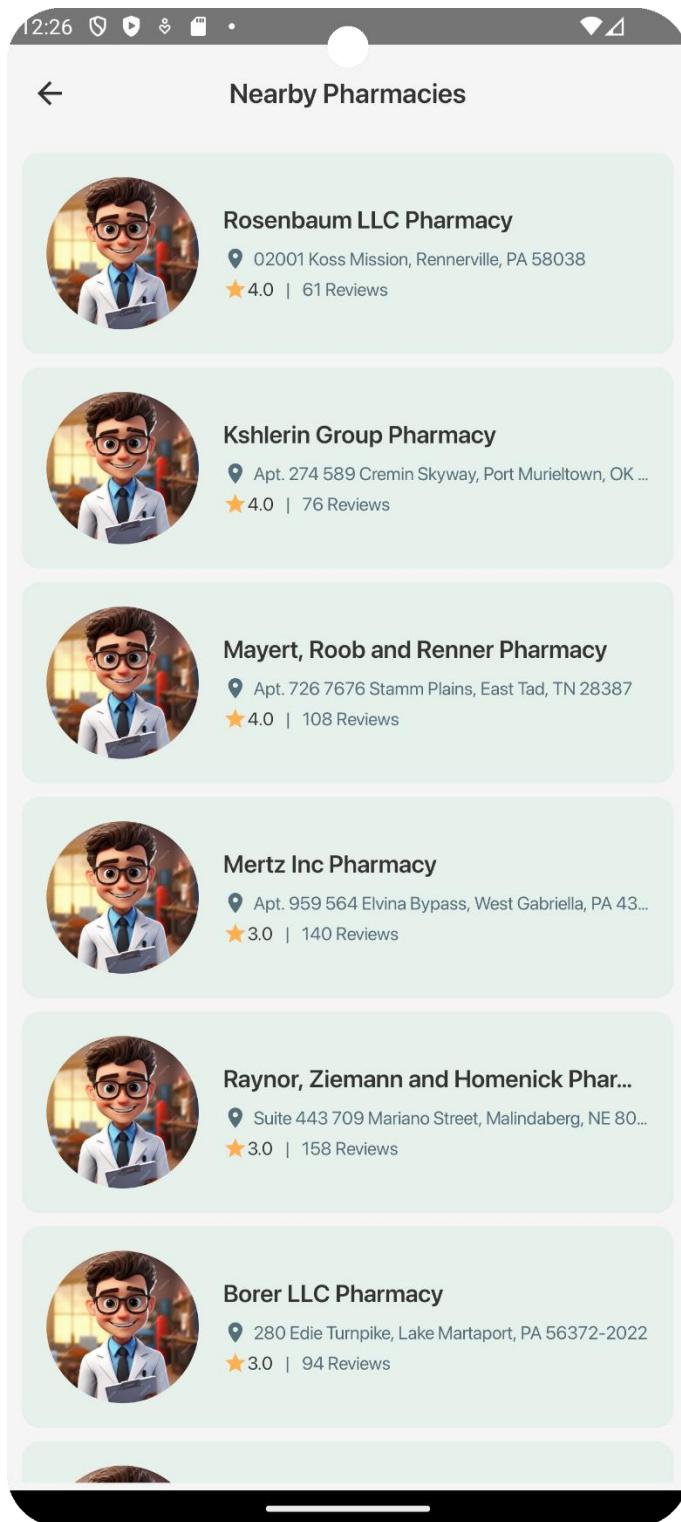


Figure 2.11: Booking pharmacy' appointment screen

Patient looking for the nearby doctors' pharmacies arranged by the nearest location

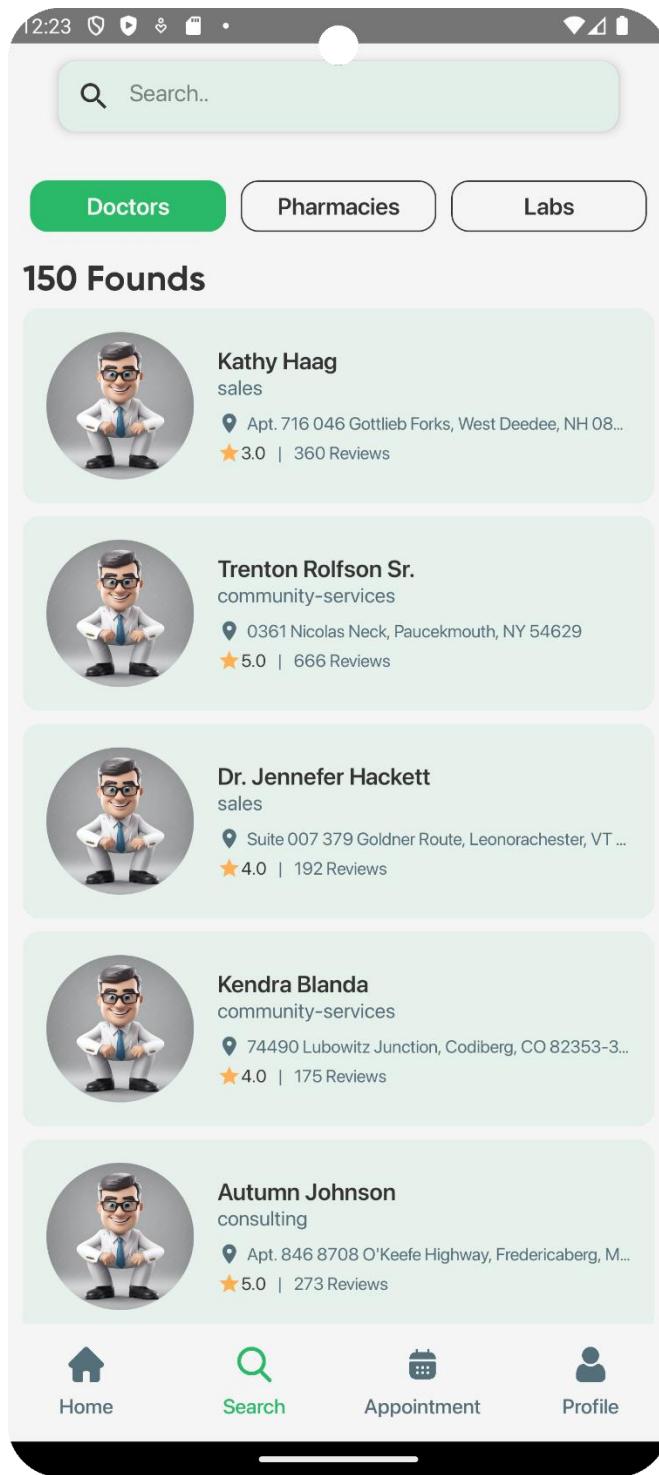


Figure 2.12: Search screen

Patient searching for a specific doctors' clinic

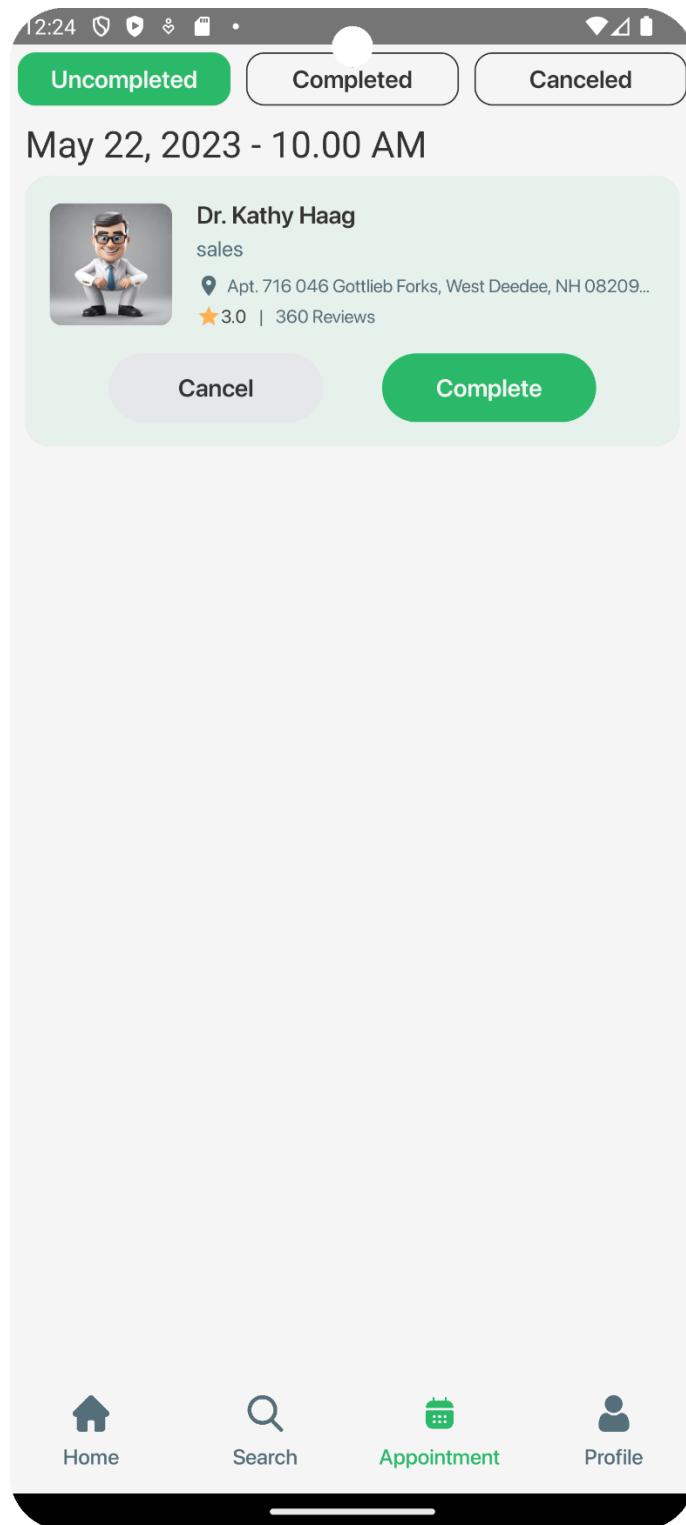


Figure 2.13: Appointments schedule screen

Patient managing his appointments with doctors

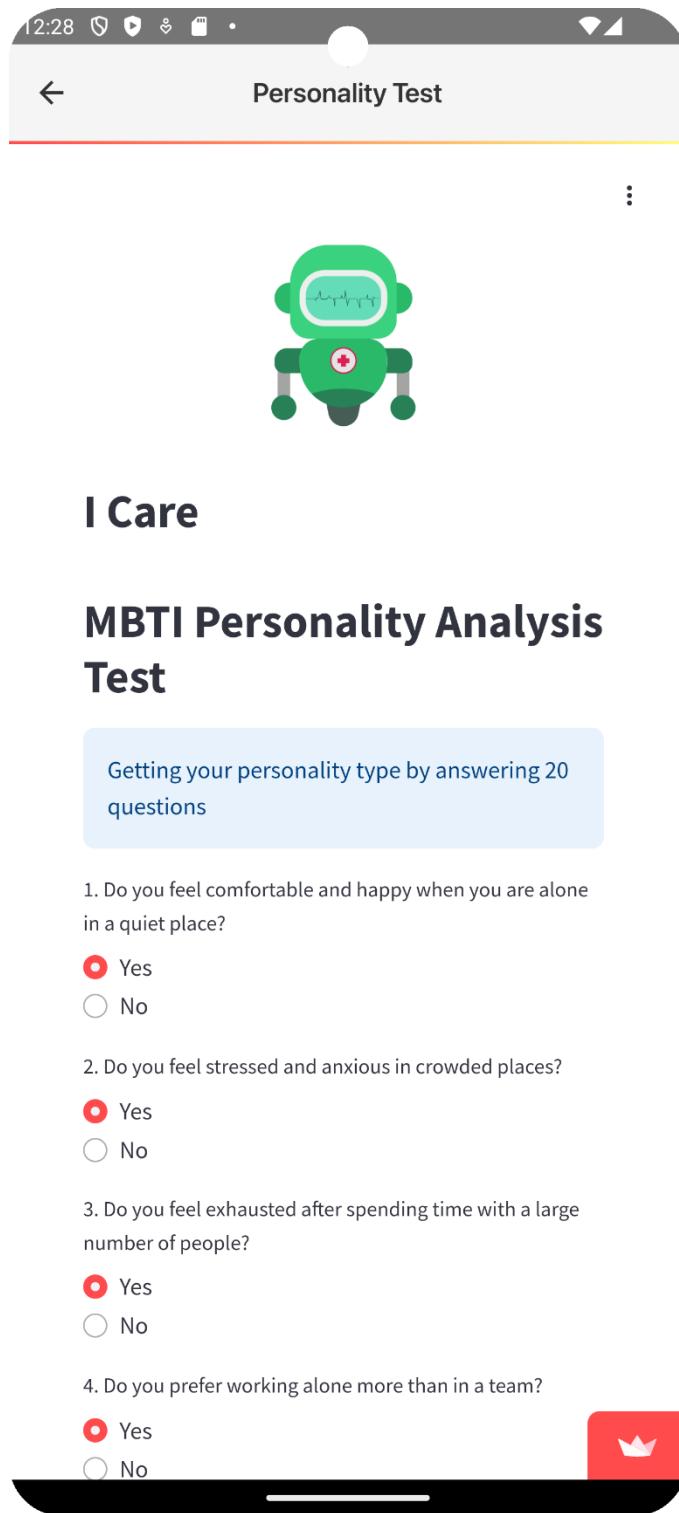


Figure 2.14: Personality analysis test screen

A personality analysis test called MBTI according to the famous psychologist “Myers Briggs” that divides all people into 16 personality types

After doing the test it gives the user his personality and information about it that can help him to make better decisions

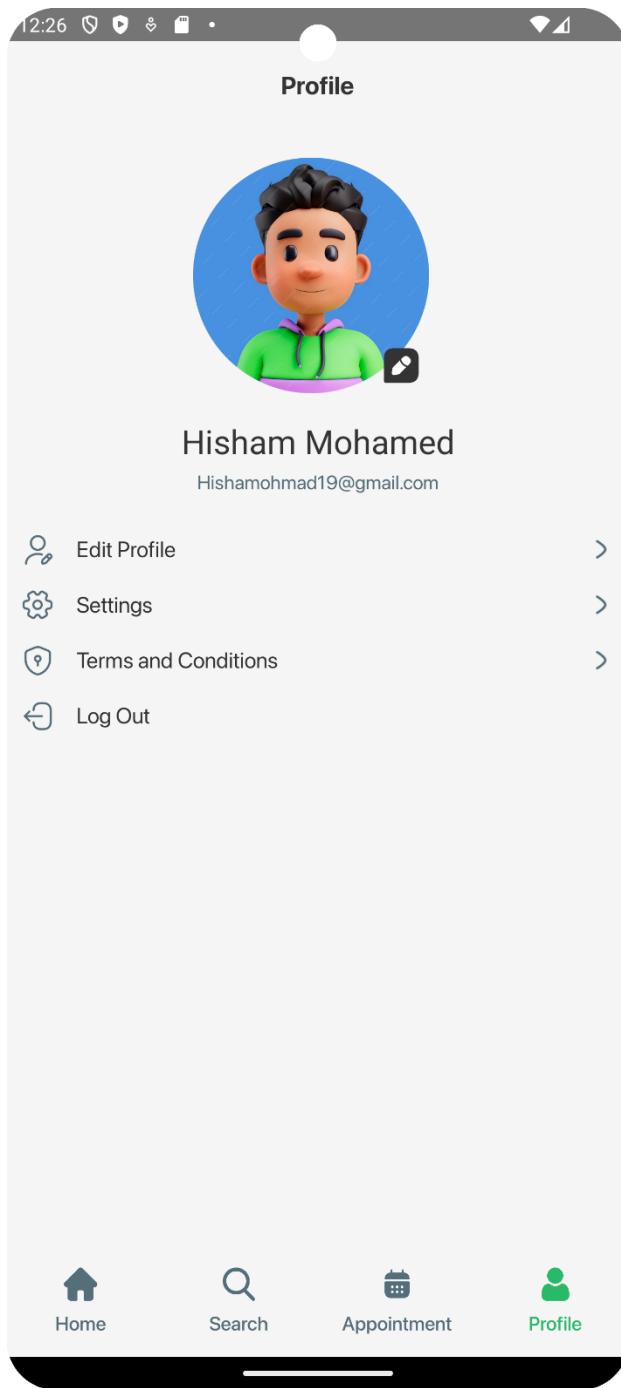


Figure 2.15: Edit profile screen

Patient can edit his profile information as his profile picture, name, address or location. He can also choose the settings options or log out from his account in case he need to sign in with another account

2.2 User Interfaces: Web application

Helps the doctor in the management of reservations
Welcome

Efficient medical management system

[Sign up](#) [Login →](#)

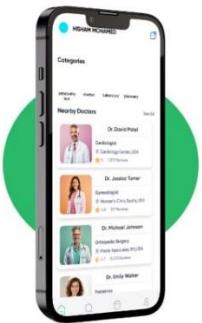
About us



We are dedicated to providing an efficient clinic management system, streamlining healthcare processes for enhanced patient care.

Our focus is on innovation, precision, and excellence in delivering comprehensive solutions for clinics, labs, and pharmacies.

Download our app



Are you a patient?
Stay healthy with our app

Easy healthcare for anyone anytime

[Download Now](#)

Activate Windows
Go to Settings to activate Windows.

Figure 2.16: Identification scrolling screens

services



Doctor

A medical facility providing healthcare services to patients. Clinics serve as spaces for disease diagnosis, treatment, and initial examinations. They encompass a variety of medical specialties to meet diverse patient needs.



Pharmacy

A facility that dispenses medications and medical supplies. Pharmacists offer guidance on medication usage, determine appropriate dosages, and ensure the safety of medications. Pharmacies play a vital role in patient care, facilitating access to essential treatments.



laboratory

A medical laboratory is a facility where blood tests, urine analyses, and other diagnostic procedures are conducted. Laboratories aim to provide accurate results for disease diagnosis and ongoing monitoring of patients' health. Labs are considered cornerstone in healthcare delivery.

FAQ

How secure is the platform for storing patient data? +

Can I easily access patient records and history through the platform? +

Is there a feature for real-time communication with patients? +

Contact us

📍 1234 Street Name, City, Country
📞 +123 456 7890
✉️ info@example.com

Name:

Email:

Message:

Send Message

[Home](#) [About US](#) [Services](#) [FAQ](#) [Contact](#)
© 2024 I Care. All rights reserved.

Activate Windows
Go to Settings to activate Windows.

Figure 2.17: FAQ common questions with answers and contact us option in case the users need a technical support

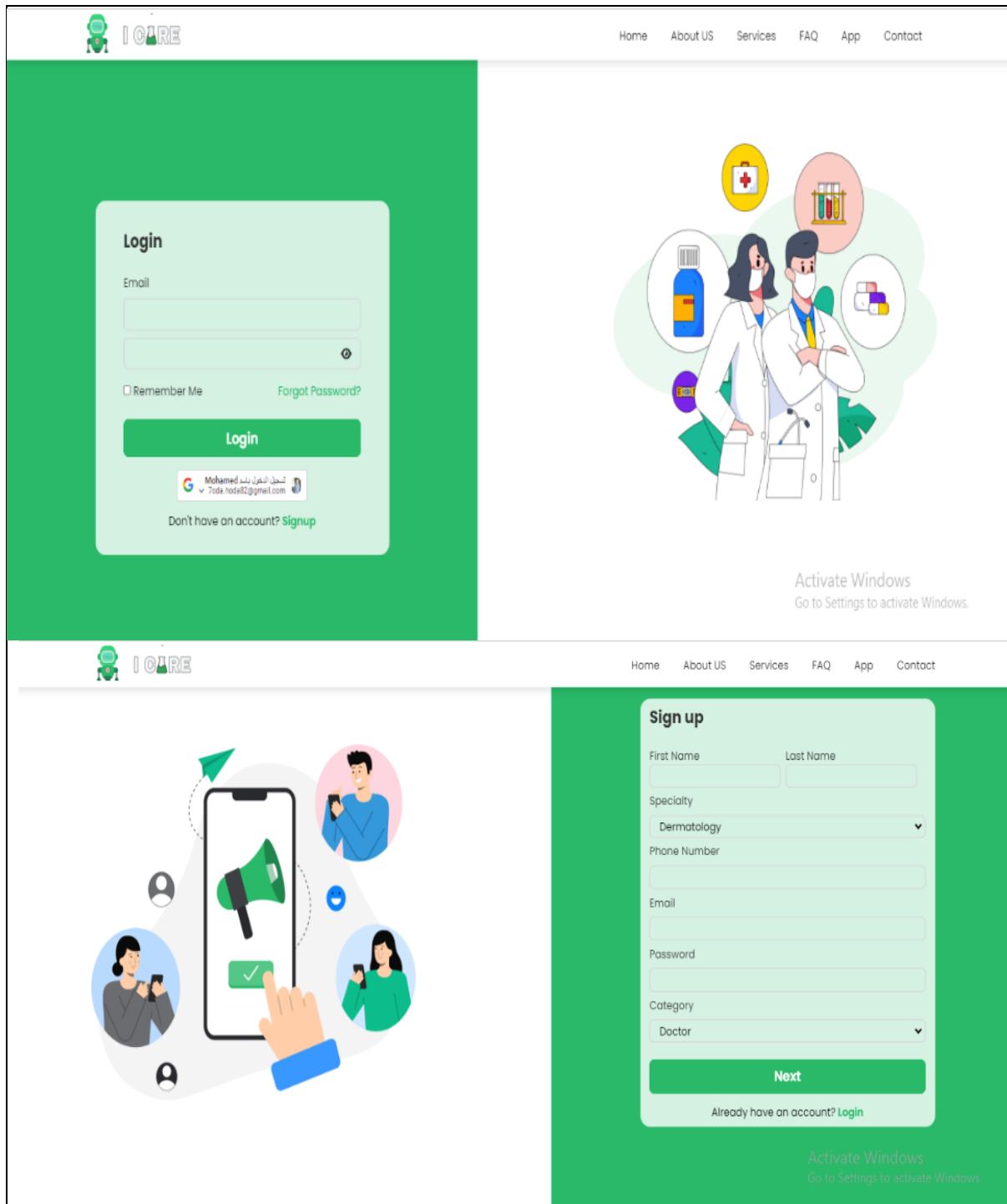


Figure 2.18: Login and sign-up screens

login where users can enter their email and password, with an option to show or hide the password. users can select "Remember Me" to stay logged in. There's also an option to log in with a Google account.

While the new doctors can sign up.

The screenshot shows a web application interface for managing reservations. At the top left is the logo 'I CARE'. On the right, there is a user profile for 'Carter Smith' with a dropdown arrow. Below the header is a navigation bar with icons for home, search, and other functions. The main content area is titled 'Reservations' and contains a table of appointment details.

First Name	Last Name	Phone Number	Appointment Date & Time	Status	Action
Jane	Cooper	+91 9876543210	13-Aug-2023 at 10:00 AM	Open	Accept Refuse
Wade	Warren	+91 9876543210	13-Aug-2023 at 10:00 AM	Booked	Pay Now Book Now
Brooklyn	Simmons	+91 9876543210	13-Aug-2023 at 10:00 AM	Completed	Pay Now Re-Block
Cameron	Williamson	+91 9876543210	13-Aug-2023 at 10:00 AM	Open	Pay Now Book Now
Leslie	Alexander	+91 9876543210	13-Aug-2023 at 10:00 AM	Open	Pay Now Book Now
Savannah	Nguyen	+91 9876543210	13-Aug-2023 at 10:00 AM	Open	Pay Now Book Now
Darlene	Robertson	+91 9876543210	13-Aug-2023 at 10:00 AM	Completed	Pay Now Book Now
Ronald	Richards	+91 9876543210	13-Aug-2023 at 10:00 AM	Open	Pay Now Book Now
Kathryn	Murphy	+91 9876543210	13-Aug-2023 at 10:00 AM	Open	Pay Now Book Now
Darrell	Steward	+91 9876543210	13-Aug-2023 at 10:00 AM	Open	Pay Now Book Now

Figure 2.19: Doctors managing scheduling screen

This page appears to be a reservation management system. The main section of the page is titled "Reservations" and it displays a table with information about various reservation appointments.

The table includes columns for the first name, last name, phone number, appointment date and time, status, and actions. Each row in the table represents a separate reservation, with details like the person's name, contact information, scheduled appointment time, and the current status of the reservation (open, completed, or booked).

The actions column provides buttons for the user to accept, refuse, or delete the reservation. This suggests that the user viewing this page likely has some level of administrative or management access to handle these reservation requests.

2.3 User Experience (UX)

Problem

In modern healthcare, access to timely medical advice, diagnosis, and treatment remains a challenge for a significant portion of the population. Patients often face obstacles such as limited access to healthcare professionals, escalating healthcare costs, and an overwhelming demand for minor illness consultations, which can lead to delayed treatment and unnecessary strain on healthcare systems.

Solution

To address these challenges, we aim to develop a medical web and app solution that leverages artificial intelligence (AI) to provide patients with free treatment recommendations for minor illnesses without the need for an immediate consultation with a healthcare provider. This innovative approach can potentially empower patients, alleviate the burden on doctors, laboratories, and pharmacies, and enhance the efficiency of healthcare delivery.

Goals

1. Enhance Access to Healthcare
2. Empower Patients
3. Alleviate Healthcare Provider Workload
4. Improve Healthcare Collaboration
5. Ensure Safety and Accuracy

Competitive analysis

APP NAME	Target Audience	Key Features	User Experience	Monetization Strategy	Strengths	Weaknesses
COMPETITOR 1	Patients	- Symptom checker	User-friendly	Subscription fee	- Large user base	- Limited AI capabilities
		Appointment booking	- Intuitive	In-app ads	- Extensive symptom database	- No integration with labs
		- Medication reminders	Fast response	- In-app purchases	Established reputation	- No pharmacy partnerships

APP NAME	Target Audience	Key Features	User Experience	Monetization Strategy	Strengths	Weaknesses
COMPETITOR 2	Patients	- User reviews and ratings	Simplified	Freemium model	- Strong AI capabilities	- Limited user base
		- AI-driven diagnosis	Minimalistic	- In-app ads	- Quick response times	- Limited language support
		- Medication info	- Easy navigation	telehealth fees	- User-friendly interface	- Limited speciality support

APP NAME	Target Audience	Key Features	User Experience	Monetization Strategy	Strengths	Weaknesses
COMPETITOR 3	Doctors	- Doctor network	- Professional	Subscription fee	- Strong doctor network	- Limited patient base
		Appointment scheduling	Customizable	In-app ads	Comprehensive features	No direct AI capabilities

		- Patient records	- Secure	- In-app purchases	- Reputation among doctors	- Limited lab integration
--	--	-------------------	----------	--------------------	----------------------------	---------------------------

APP NAME	Target Audience	Key Features	User Experience	Monetization Strategy	Strengths	Weaknesses
COMPETITOR 4	Laboratories	- Test booking	User-friendly	Subscription fee	- Large user base	- Limited AI capabilities
		Test results tracking	Informative-	- In-app purchases-	Comprehensive test menu-	Limited integration with other apps
		Home sample purchases	- User-friendly	- In-app ads	- Fast report delivery	- Limited outreach efforts

APP NAME	Target Audience	Key Features	User Experience	Monetization Strategy	Strengths	Weaknesses
COMPETITOR 5	Pharmacies	Medication ordering	Convenient	Transaction fees	- Extensive pharmacy network	Limited medication database
		Prescription upload	- User-friendly	- In-app ads	Quick delivery options	- Limited healthcare support
		Medication reminders	- Secure	- In-app purchases	Loyalty and discount programs	-- Limited pharmacy partnerships

User Research (interview & survey)

Patient Survey Questions:

How often do you seek medical advice for minor illnesses or symptoms?

Rarely

Occasionally

Frequently

Always

What challenges have you faced when seeking medical advice? (Open-ended)

Would you be interested in a medical web and app that provides free treatment recommendations based on AI analysis?

Very interested

Somewhat interested

Not interested

What features would you expect from such an app? (Open-ended)

Do you have any concerns about relying on AI for medical advice? (Open-ended)

How user-friendly should the app be for you to consider using it regularly?

Extremely user-friendly

Moderately user-friendly

Not very user-friendly

Doctor Survey Questions:

How often do you see patients seeking medical advice for minor illnesses or symptoms?

Rarely

Occasionally

Frequently

Always

What challenges do you encounter when patients seek medical advice for minor illnesses? (Open-ended)

Would you support the use of a medical web and app that offers free treatment recommendations to patients based on AI analysis?

Strongly support

Somewhat support

Do not support

What features do you believe are important for such an app to be effective for patients? (Open-ended)

Do you have any concerns about patients relying on AI for medical advice? (Open-ended)

How user-friendly should the app be for you to consider recommending it to your patients?

Extremely user-friendly

Moderately user-friendly

Not very user-friendly

Laboratory and Pharmacy Survey Questions:

How often do patients visit your laboratory or pharmacy for minor illnesses or symptoms?

Rarely

Occasionally

Frequently

Always

What types of medical tests or treatments do you commonly provide for minor illnesses or symptoms?

Would you be interested in collaborating with a medical web and app that offers free treatment recommendations to patients based on AI analysis?

Very interested

Somewhat interested

Not interested

What benefits or challenges do you anticipate with such a collaboration?
(Open-ended)

Do you have any concerns about the accuracy and safety of the AI-driven recommendations provided to patients?

How user-friendly should the app be for patients to facilitate their interaction with your laboratory or pharmacy services?

Extremely user-friendly

Moderately user-friendly

Not very user-friendly

Define phase:

Persona1: patient

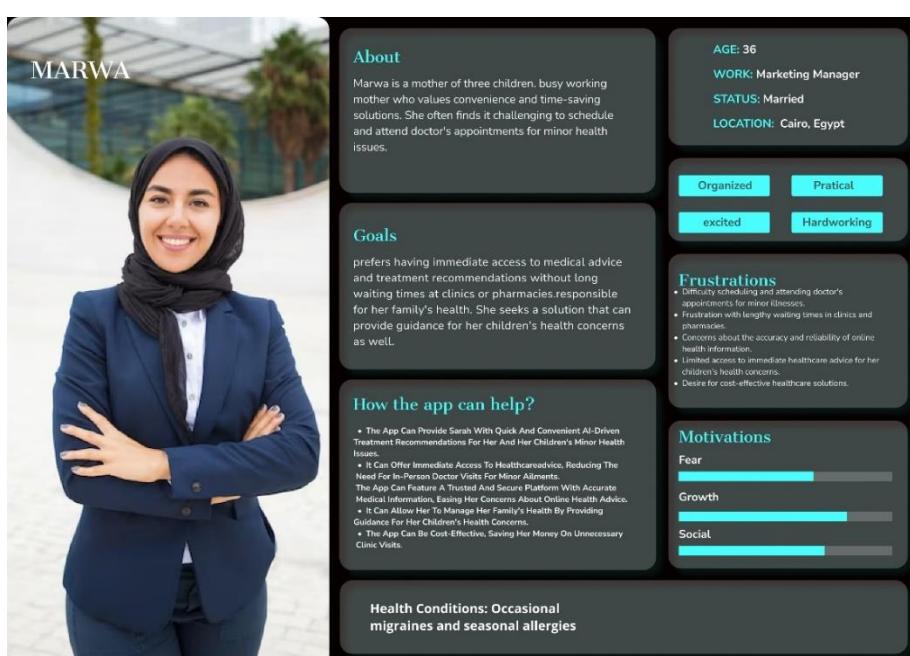


Figure 3.1

Persona 2: Doctor



Dr. Ahmed Mahmoud

About

Dr. Ahmed Mahmoud has been a dedicated general practitioner for the past 15 years, serving the community in Alexandria, Egypt. After completing his medical degree at Cairo University, he pursued a specialization in general medicine and developed a passion for providing comprehensive healthcare to his patients.

Goals

- Efficient Workload: Dr. Ahmed values tools that can help him efficiently manage patient consultations, especially for minor illnesses, allowing him to focus on more complex cases.
- Patient Care: He wants to provide accessible and timely healthcare advice to patients, ensuring they receive the guidance they need without unnecessary delays.
- Collaboration: Dr. Ahmed values tools that facilitate collaboration with laboratories and pharmacies, streamlining patient care.
- Technology Integration: He is open to adopting technology that seamlessly integrates with his practice, making it easier to manage patient records and communication.

How the app can help?

- The App Can Offer An AI-Driven Symptom Checker, Allowing Patients To Receive Initial Advice For Minor Issues Without A Direct Consultation.
- It Can Provide A Platform For Easy Appointment Scheduling And Virtual Consultations, Offloading The Burden On Dr. Ahmed's Schedule.
- The App Can Facilitate Seamless Communication With Laboratories And Pharmacies For Efficient Patient Care.
- Integration With Existing Electronic Health Record Systems Can Make It Easier For Dr. Ahmed To Manage Patient Information.

AGE: 45
WORK: General Practitioner

- Location: Alexandria, Egypt
- Years of Experience: 15 years
- Specialization: General Medicine

Characteristics:
 Organized, Practical, excited, Hardworking

Frustrations

- Overwhelming patient load, particularly for minor health issues.
- Time constraints and challenges in efficiently managing patient appointments.
- Limited collaboration tools with laboratories and pharmacies, leading to delays in patient care.
- Resistance to adopting new technologies without clear benefits.

Motivations

Fear: [Progress bar]

Growth: [Progress bar]

Social: [Progress bar]

Limited collaboration tools with laboratories and pharmacies, leading to delays in patient care

Figure 3.2

Persona 3: Laboratory Specialist



Aya Hassan

About

Aya Hassan, a 28-year-old laboratory technician, has been actively contributing to the field of clinical pathology for the past five years. She earned her degree from Ain Shams University and quickly developed a keen interest in the intricate world of diagnostic testing. Aya has worked in various laboratories across Giza, gaining experience in handling a wide array of medical tests.

Goals

- Efficient Workflow: Aya aims to streamline the laboratory testing process, ensuring quick and accurate results for patients.
- Comprehensive Data: She wants to connect with doctors and pharmacies to provide comprehensive and timely healthcare services.
- Technology Integration: Aya is interested in tools that integrate seamlessly with laboratory equipment and processes, minimizing manual data entry. She deeply understands the importance of her role in contributing to patient care and seeks tools that enhance this aspect.

How the app can help?

- The App Can Offer An Integrated Platform For Receiving And Managing Test Requests, Reducing Manual Data Entry.
- It Can Facilitate Seamless Communication With Doctors And Pharmacies, Ensuring Timely Delivery Of Test Results.
- A User-Friendly Interface Can Enhance The Efficiency Of Managing A High Volume Of Specimens.
- Automation With Laboratory Equipment Can Automate Certain Processes, Minimizing The Risk Of Errors.
- These Personas Provide A Glimpse Into The Needs, Goals, And Pain Points Of Doctors And Laboratory Professionals, Helping Guide The Design And Features Of Your Medical Web And App For A Comprehensive User Experience.

Organized, Practical, excited, Hardworking

Frustrations

- Overwhelming patient load, particularly for minor health issues.
- Time constraints and challenges in efficiently managing patient appointments.
- Limited collaboration tools with laboratories and pharmacies, leading to delays in patient care.
- Resistance to adopting new technologies without clear benefits.

Motivations

Fear: [Progress bar]

Growth: [Progress bar]

Social: [Progress bar]

• Overwhelming patient load, particularly for minor health issues.

Figure 3.3

Empathy Map

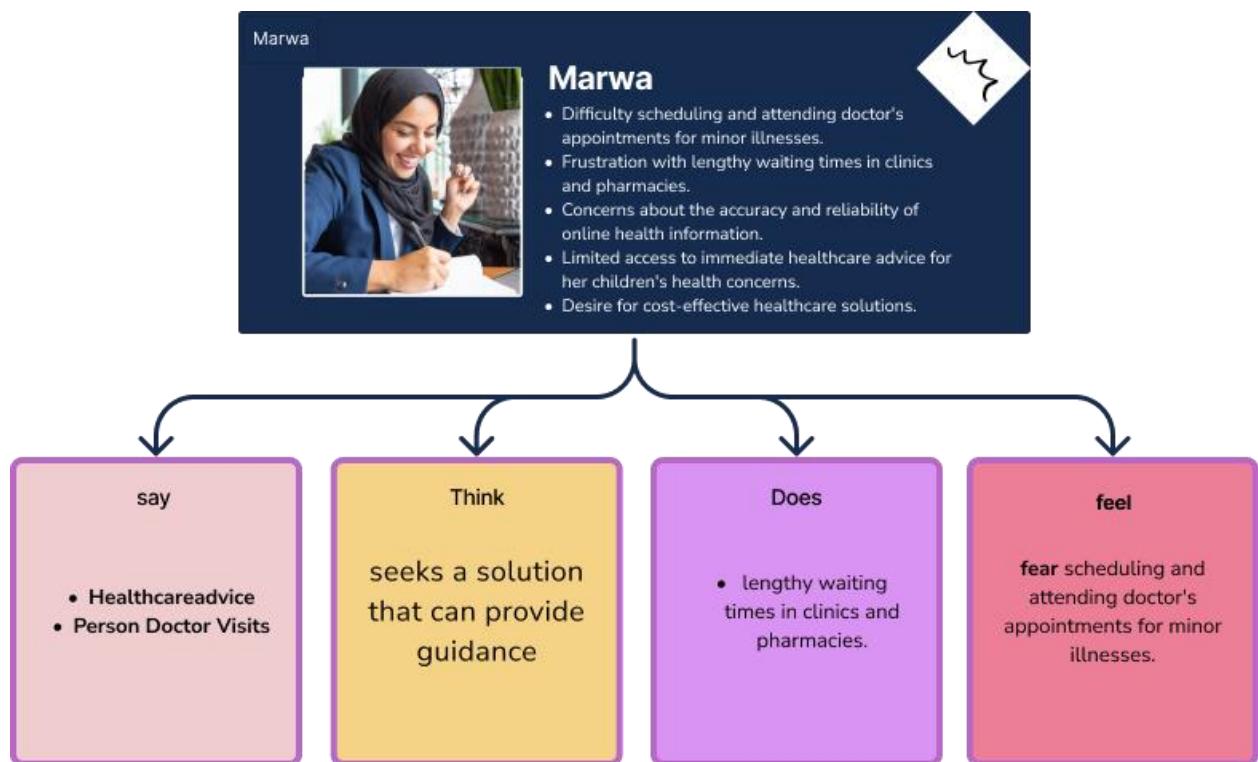


Figure 3.4

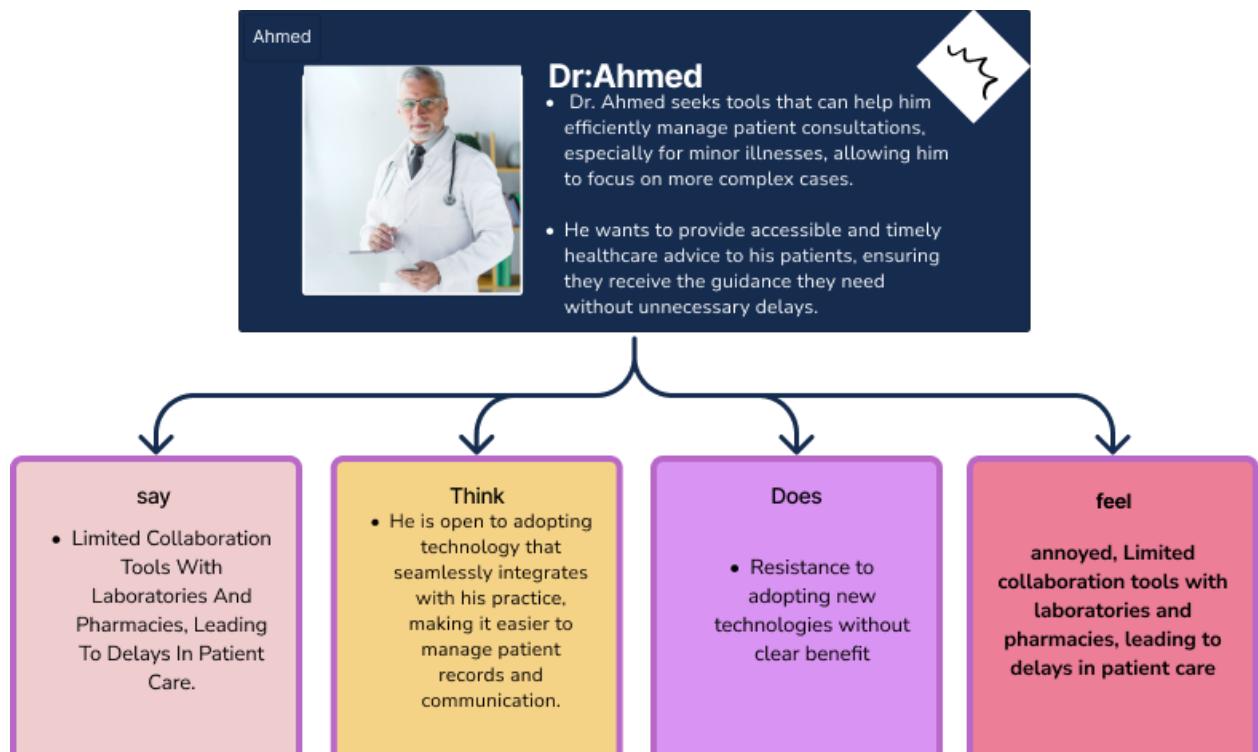


Figure 3.5



Figure 3.6

User Flow:

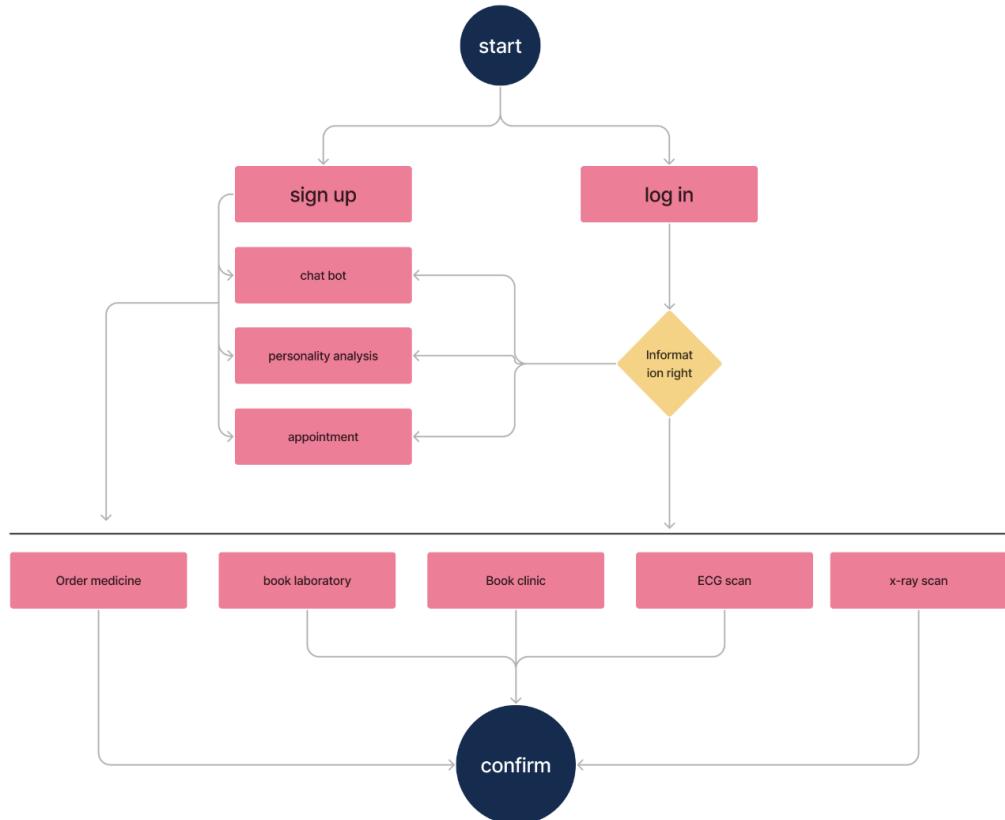


Figure 3.7

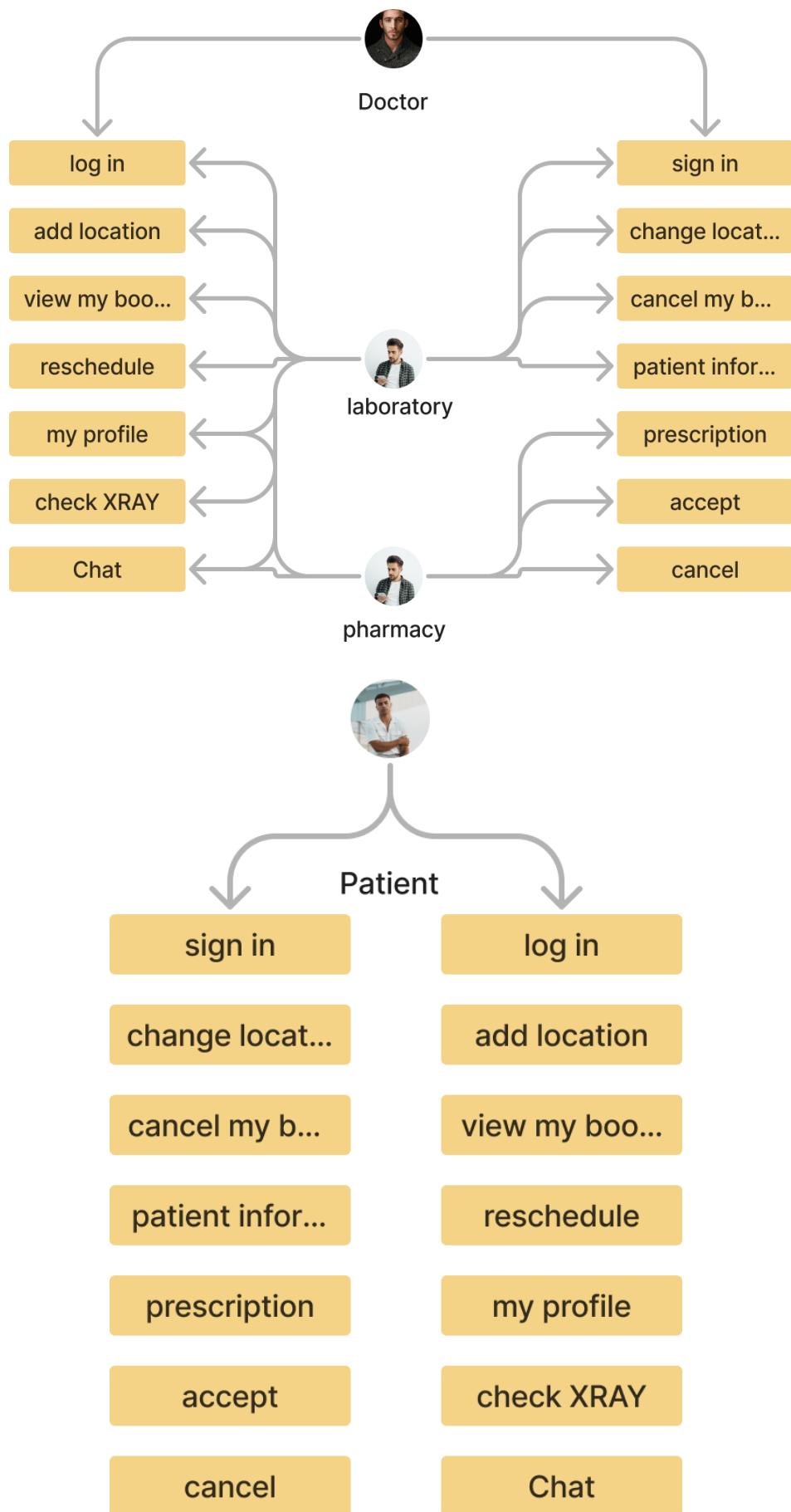


Figure 3.8

Chapter 4: Implementation

4. Implementation

4.1 Artificial Intelligence AI

The project contains 17 AI models, each one has a specific function and role.

After finishing and testing each one of them individually, combining them to give the final product which consists of several models connected as a series one to another to get the individual final accurate output.

To make that we need to follow the Artificial Intelligence Development Life Cycle which are:

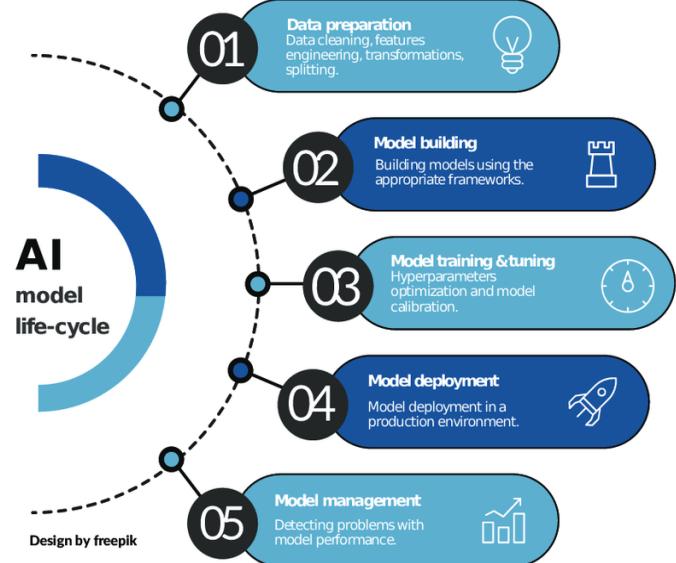


Figure 4.1

Using transfer learning to get a pre-trained models on a huge dataset image net and customize their input layer shape to be suitable with the images and customize the output layer structure and activation function as need, so the models have an initial value for the parameters then they train faster and gives better accuracy.

The AI module is designed in different parts. There are a Natural Language Processing NLP and Computer Vision models that mimic a doctor.

- NLP part “Chatbot”:

An AI based smart chatbot called "Caroline" talking to the patient and taking its disease symptoms, then diagnosing the disease and recommend making some tests as x-ray, MRI ... in addition, given information about the predicted

disease as an overview, symptoms, and treatments. It can predict 30 diseases such as (Breast Cancer, Influenza, Covid 19, Stroke, ...)

- Computer Vision part “Medical Imaging Scan”:

A sequence of AI Computer Vision models for scanning medical imaging and tests it can scan (X-ray, MRI, CT, OCT, ECG, or Food image), detect the image type (Image Recognition), if it is medical imaging image, applying anatomical recognition, disease evaluation, disease diagnosis.

It can predict 25 disease types such as (Bone Fracture, Brain Tumor, Covid 19, Breast Cancer, ...)

It also can recognize 101 food types from images and shows the approximate number of calories per gram.

In addition, there is an Ensemble Machine Learning model with Random Forest algorithm for Electrocardiography ECG for predicting heart diseases.

The table below explains the name, input, and the functionality for output of each AI model.

Model number	Input	Functionality	Output
Model 1	Unknown image	Image recognition	Medical imaging - Food - Other
Model 2	Medical imaging	identify medical imaging type	CT - MRI - Xray - OCT
Model 3	x-ray image	Anatomical recognition (identify body parts)	Skull - Hand - Feet - Chest - Other
Model 4	Chest x-ray image	chest diseases diagnosis	Covid 19 - Pneumonia - Normal
Model 5	x-ray image	bone fracture evaluation	Fractured - Not fractured
Model 6	Fractured bone x-ray	bone fracture diagnosis (identify fracture type)	Avulsion - Spiral - Comminuted ,...12
Model 7	MRI image	Anatomical recognition (identify body parts)	Brain - Breast
Model 8	Brain MRI	brain tumor evaluation	Tumored - Not tumored
Model 9	Tumored brain MRI	brain tumor diagnosis (identify what type)	Glioma - Meningioma - Pituitary
Model 10	Breast MRI	breast cancer evaluation	Cancered - Not cancered
Model 11	Cancered breast MRI	breast cancer diagnosis	Benign - Malignant
Model 12	CT image	CT body disease diagnosis	Cyst - Normal - Stone - Tumor
Model 13	OCT image	eye diseases diagnosis	CNV - DME - Drusen - Normal
Model 14	Food image	food recognition	Fries - Ice cream - Falafel, ...101
Model 15	ECG file	heart diseases diagnosis	Normal - Supraventricular - Premature, ...
Model 16	Comment text	identify is the comment positive or negative	Positive - Negative - Neutral
Model 17	Patient's message	Symptoms diseases diagnosis	Influenza - Malaria - Heart attack, ...30

4.1.1 Data Gathering

The datasets that are used vary according to the functionality variety as in the chatbot we need to collect text data.

Start with searching and collecting the dataset from the famous web sites as Kaggle, Google Dataset Search, COCO dataset, TCIA, NIH, Figshare, ...

Making a dataset from scratch:

- Chatbot text dataset:

The dataset consists of three main parts “Tag”, “Pattern”, and the “Response”, the Tag is the labels of the class it can be greeting or gratitude or a disease name. The Pattern is a list containing all possible questions under the same label as disease symptoms messages for the same disease. The response is a list containing all possible outcomes for this label or Tag.

- Images datasets:

The image dataset is simply considered as a folder containing sub-folders that contain image samples from the same class according to the problem.

For example, the 1st model is used for image identification as we must consider that the user may add a non-medical imaging or food image, so the first model is filtering and orientating the images for the suitable next model in the sequence. It contains three classes Medical imaging, Food, and Other.

So, i have developed a code to access many medical imaging images with different types to the medical imaging class folder, and to add a food image from the Food-101 dataset to the food class folder, finally add images from the image-net dataset which contains 1000 different classes as cars, dogs, horses, ... into the other class folder.

4.1.2 Data Analysis and Preprocessing

The preprocessing process varies according to the used data type and the needs. We will discuss each type of the used data and the suitable preprocessing processes on it.

4.1.2.1 Text Preprocessing:

Firstly, the chatbot uses the text data so the input is a string of text or sequence of related words, here the arrangement of the words is important, so we need to consider these while the modeling process.

- Start with removing the unwanted characters as colons, quotes, dots, question marks, spaces before and after the text, and any special character that is doesn't effect on the meaning of the word.
- Converting all characters in the text to lowercase to ensure uniformity.
- Stemming the text to return each word to its root form for example make “writing” into “write” and “plays” into “play”
- Apply tokenization split text into individual words or tokens and indexing them.

4.1.2.2 Image Preprocessing

- Resize the image to the needed size that the models train on it which is (224 x 224), so we can customize the target size of all images while being read by the code into this size.
- Next step is normalization for the images by divide the pixels' values by 255, as the pixels' values is between [0:255], so we normalize these values to be between [0:1]
- Convert the image into a matrix by Numpy array rather than a list as the Numpy arrays run faster by around 30 times than lists.

4.1.3 Modeling and Training

After preprocessing the datasets, the next phase is to choose the suitable model architecture, build it, then train it on the preprocessed data.

The model architecture in the computer vision part for the imaging image detection is using the deep learning convolutional neural networks CNN architecture that is mainly developed for unstructured data mainly for image detecting purpose.

It is suitable for the simple tasks with a small number of classes as for the evaluation process which answer does the patient have a disease or not, and classification of small number of classes.

When the number of classes increases as in the food images a different architecture is used as Resnet and Inception v3 and LSTM for the chatbot.

Models Architecture:

The Convolutional Neural Networks CNN architecture has three main kinds of layers which are Convolutional layer, Pooling layer, and fully connected layer, the output layer with activation function vary according to the problem using the optimizers Adam and Gradient Dissent Algorithm.

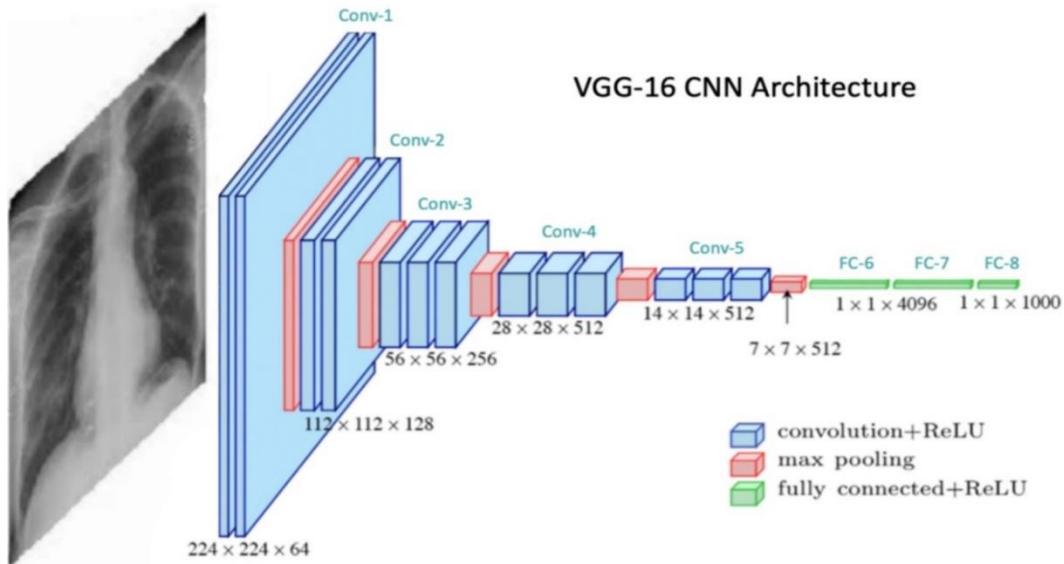


Figure 4.2

Long Short-Term Memory LSTM – Chatbot model architecture after the embedding layer and the final layer is a layer with 46 neurons that is the number of classes, and the activation function is SoftMax.

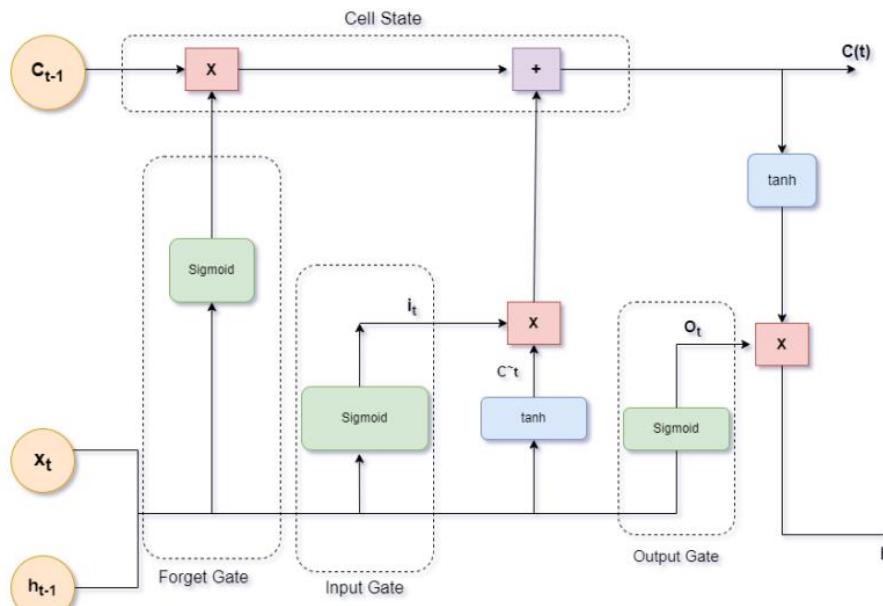


Figure 4.3

AI scan models sequence

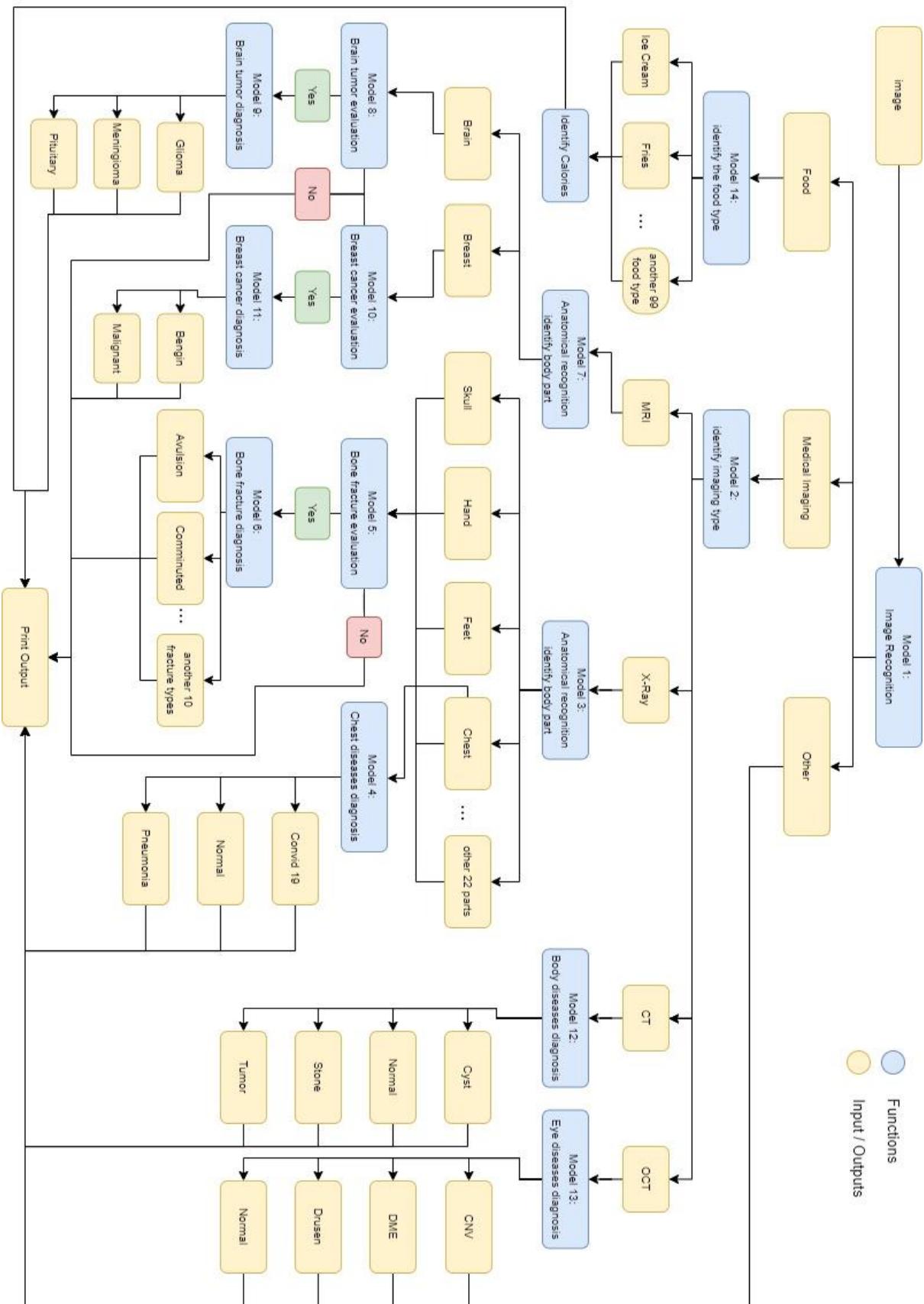


Figure 4.4

Models training history:

There are a sample of four models validation accuracy and loss value history while they are training.

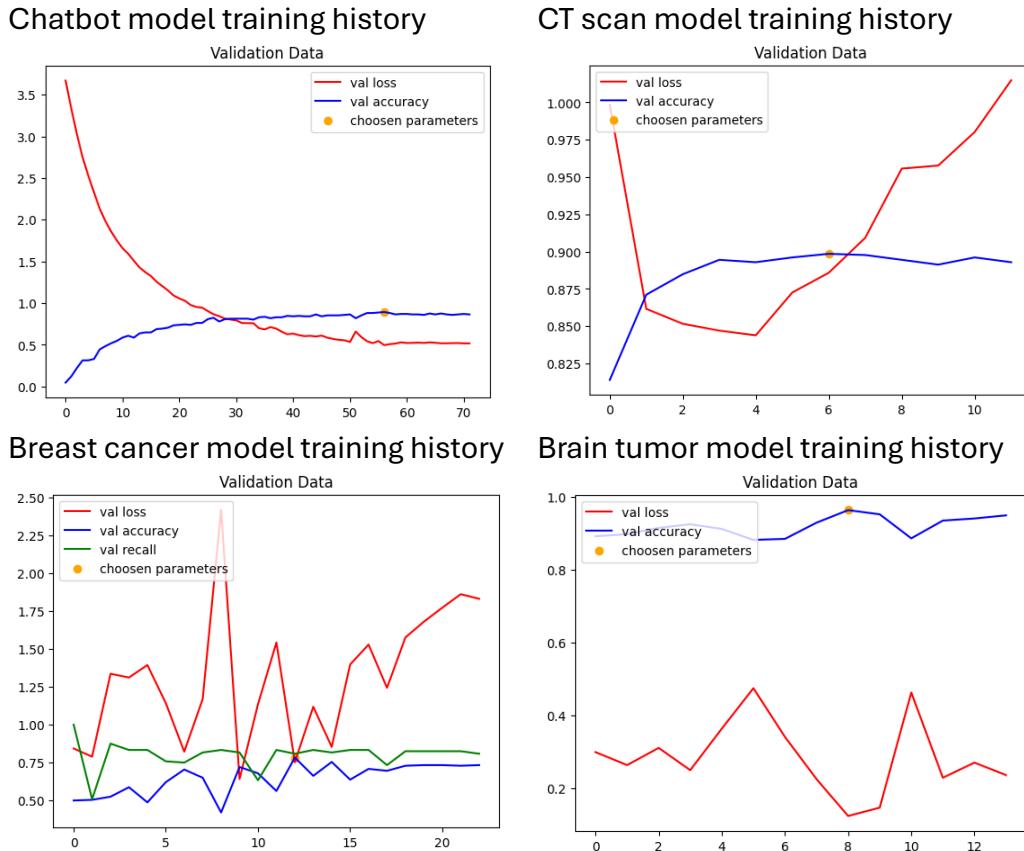


Figure 4.5

3.1.4 Testing and Validation

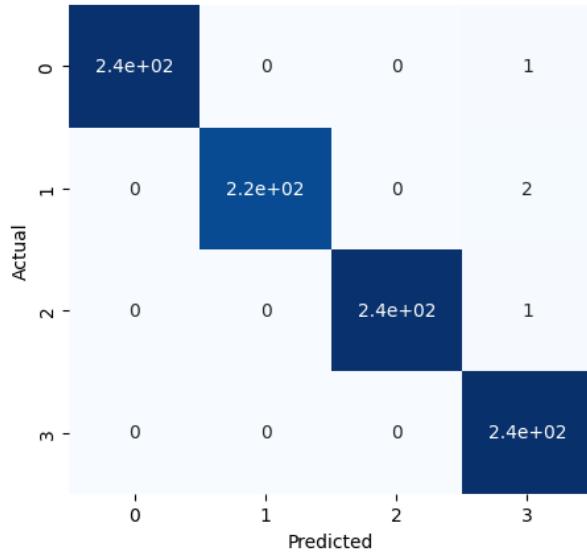
After the continuity process in training, testing, and improvement the final accuracy for all the models is in the table below:

AI models input, output, functionality, and accuracy

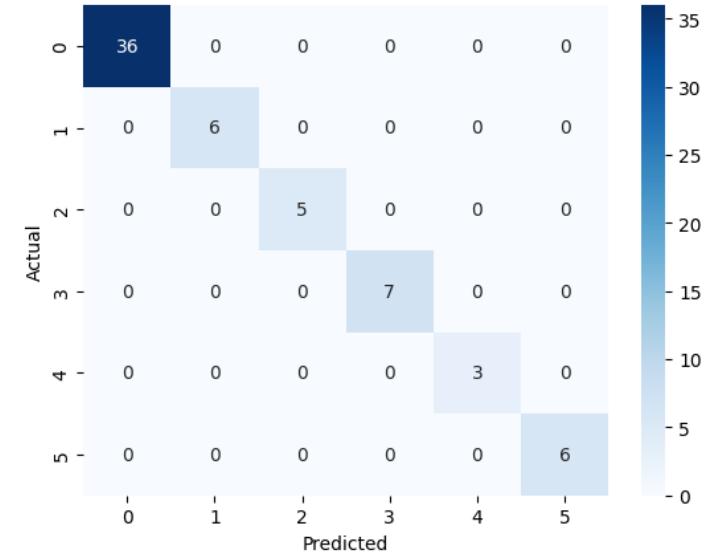
Model number	Input	Functionality	Train accuracy	Test accuracy	Recall	Output
Model 1	Unknown image	Image recognition	98.7%	93.8%	-	Medical imaging - Food - Other
Model 2	Medical imaging	identify medical imaging type	100%	99.5%	-	CT - MRI - Xray - OCT
Model 3	x-ray image	Anatomical recognition (identify body parts)	99.2%	100%	-	Skull - Hand - Feet - Chest - Other
Model 4	Chest x-ray image	chest diseases diagnosis	98.3%	100%	-	Covid 19 - Pneumonia - Normal
Model 5	x-ray image	bone fracture evaluation	98.3%	77.8%	81%	Fractured - Not fractured
Model 6	Fractured bone x-ray	bone fracture diagnosis (identify fracture type)	85.8%	52.1%	-	Avulsion - Spiral - Comminuted ,...12
Model 7	MRI image	Anatomical recognition (identify body parts)	99.27%	98.8%	-	Brain - Breast
Model 8	Brain MRI	brain tumor evaluation	98.2%	95.8%	100%	Tumored - Not tumored
Model 9	Tumored brain MRI	brain tumor diagnosis (identify what type)	98.4%	96.4%	-	Glioma - Meningioma - Pituitary
Model 10	Breast MRI	breast cancer evaluation	100%	94.7%	98%	Cancered - Not cancered
Model 11	Cancered breast MRI	breast cancer diagnosis	96%	78.3%	80%	Bengin - Malignant
Model 12	CT image	CT body disease diagnosis	100%	93.3%	-	Cyst - Normal - Stone - Tumor
Model 13	OCT image	eye diseases diagnosis	94.3%	88.4%	-	CNV - DME - Drusen - Normal
Model 14	Food image	food recognition	99.9%	77.4%	-	Fries - Ice creame - Falafel, ...101
Model 15	ECG file	heart diseases diagnosis	98%	97.7%	-	Normal - Supraventricular - Premature, ...
Model 16	Comment text	identify is the comment positive or negative	88.4%	86.2%	-	Positive - Negative - Neutral
Model 17	Patient's message	Symptoms diseases diagnosis	98.8%	94.3%	-	Influenza - Malaria - Heart attack, ...30

Sample models output heatmaps.

Model 2: medical imaging scan type heatmap



Model 3: anatomical recognition heatmap



Model 17: chatbot heatmap

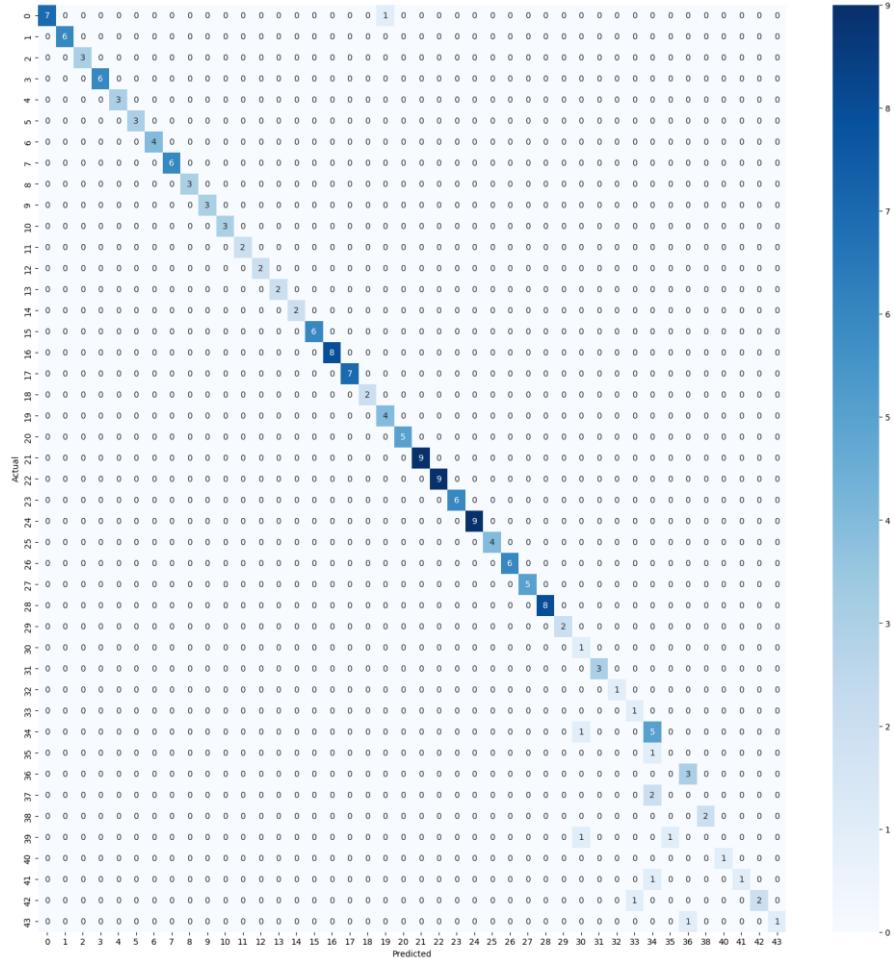


Figure 4.6

4.1.5 Deployment

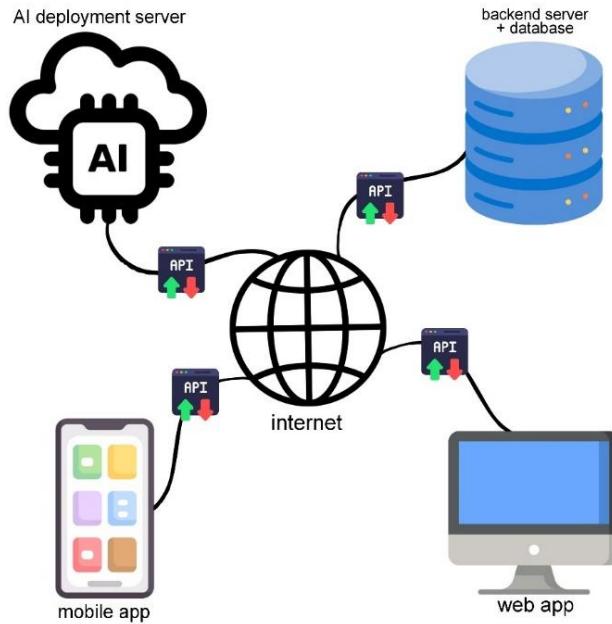


Figure 4.7

After the testing and evaluation processes the models were ready for the next step to collect and use.

The models simplified using the TensorFlow lite library to be suitable to run on mobile phones then deployed using a Flask application to make a route for the prediction functions and then uploaded as a GitHub repository and connect them with a server get from the apps an API links to be embedded in mobile application.

So, the user did not need to download the AI models with the application as they are consuming large storage but when the user start using the mobile app to make a smart diagnosis the input travels from the app as a json body to the AI server predict the output then it as the response to the application then the application forward the response to the backend server to store it.

4.1.6 The used tools:

- Requirements for the AI environment:

Python >= 3.9, Numpy, Pandas, Matplotlib, Seaborn, TensorFlow == 2.15.0, Keras, Scikit-Learn, Pillow, Flask, Streamlit, and Gunicorn.

4.2 Mobile Application

1. Intro of android application
2. Purpose of app
3. Key functionality
4. Target audience
5. Development environment
 - a. Kotlin
 - b. IDE
 - c. Library
6. Hardware Requirements
7. Software Requirements
8. Installation
9. Design Pattern: MVVM
 - a. Concept
 - b. Explain why
 - c. Benefits
10. Overview
 - d. Model
 - e. View
 - f. View Model

4.2.1 Introduction

Welcome to the documentation for our groundbreaking mobile application designed for a transformative medical ecosystem designed to empower seamless collaboration between doctors, pharmacies, labs, and patients. Icare fosters a patient-centric environment, promoting informed decision-making and ultimately, improved health outcomes for all.

4.2.2 Purpose of the App

I care serves a multifaceted purpose, catering to the distinct needs of each user group within the distinct Needs of each user group within the healthcare hand scape

- **Doctors:** Experience streamlined patient communication, efficient appointment and reservation management, and access to cutting-edge AI-powered diagnostic tools that enhance patient care.
- **Pharmacies & Labs:** Connect directly with patients for prescription refills and lab test scheduling, fostering stronger patient engagement.
- **Patients:** Gain convenient access to a comprehensive suite of healthcare services. Book appointments with doctors and labs with ease, and leverage AI-powered features for self-assessment and diagnosis support.

4.2.3 Key functionality

- **Effortless Reservation System:** Simplify appointment booking for doctors, pharmacies, and labs, prioritizing patient convenience.

- **AI-powered Diagnostics:** Harness the power of cutting-edge AI to analyze X-rays, ECG scans (and potentially other medical data), providing valuable insights and diagnostic support to doctors.
- **Cognitive Imaging & Personality Tests:** Offer patients access to AI-powered assessments designed to cultivate self-awareness and enhance overall well-being.
- **AI-powered Chatbot for Medical Inquiries:** Empower patients to ask medical questions and receive informative responses from a sophisticated AI chatbot.
- **Comprehensive Communication Tools:** Facilitate seamless communication between patients, doctors, pharmacies, and labs, fostering a collaborative healthcare ecosystem.

4.2.4 Target Audience

I care caters to a diverse range of users within the healthcare system, aiming to bridge the gap between patients and healthcare providers:

- **Doctors:** I care is designed to be a valuable tool for doctors of all specialties. It streamlines communication with patients, simplifies appointment management, and offers AI-powered diagnostic support to enhance patient care.
- **Pharmacies & Labs:** This app empowers pharmacies and labs to connect directly with patients for prescription refills and lab test scheduling, improving efficiency and fostering stronger patient engagement.
- **Patients:** I care is designed to be user-friendly and accessible for patients of all ages and technical backgrounds. It provides convenient access to a

variety of healthcare services, including appointment booking, AI-powered self-assessment tools, and a medical chatbot for basic inquiries.

4.2.5 Development Environment

4.2.5.1 Programming Language: Kotlin

Kotlin is the official language recommended by Google for Android development. It offers several advantages, including:

Why Kotlin?

While Java remains a strong option for Android development, Kotlin offers several advantages that make it the preferred choice for this project:

- **Modern and Concise:** Kotlin's syntax is more concise and easier to read compared to Java. This reduces boilerplate code, improves maintainability, and allows developers to express their ideas more clearly.
- **Null Safety:** One of Kotlin's key strengths is its built-in null safety system. This helps prevent null pointer exceptions, a common source of crashes and errors in Java applications. By ensuring type safety, Kotlin promotes more robust and reliable code.
- **Interoperability:** Kotlin seamlessly integrates with existing Java libraries and frameworks. This allows developers to leverage the vast ecosystem of Java libraries while still enjoying the benefits of Kotlin's features.
- **First-Class Support from Google:** As the official language recommended by Google for Android development, Kotlin receives

ongoing support and improvements. This ensures compatibility with the latest Android features and development tools.

4.2.5.2 IDE (Integrated Development Environment):

Android Studio (version [Koala | 2024.1.1])

Android Studio is the official IDE for Android app development. Built on IntelliJ IDEA, it provides a comprehensive set of tools specifically tailored for creating Android apps, including:

- **Code Completion:** Provides suggestions and auto-completion for code snippets, speeding up development.
- **Debugging Tools:** Allows for efficient debugging of your application to identify and fix issues.
- **Built-in Emulator:** Enables testing your app on various virtual devices without needing a physical device.
- **Integration with Android SDK and Build System:** Streamlines the development and build process.

4.2.5.3 Libraries :

Jetpack Compose:

Jetpack Compose is a modern UI framework developed by Google for building declarative and composable user interfaces in Android apps. It offers a significant departure from the traditional XML-based approach, providing several advantages:

- **Declarative:** With Compose, you describe the desired UI state, and the framework automatically recomposes the UI whenever the state changes. This leads to a more concise and readable codebase.
- **Composable:** UI elements in Compose are built from composable functions, allowing for modular and reusable code. This promotes maintainability and easier UI creation.
- **Modern and Expressive:** Compose offers a clean and expressive API for building complex and dynamic user interfaces. It leverages Kotlin's features to provide a smooth development experience.

Retrofit:

- Retrofit is a powerful library that simplifies network communication in Android apps. It provides a type-safe API for building RESTful APIs, allowing you to define interfaces that represent your API endpoints. Retrofit takes care of making network calls, parsing JSON responses into your defined data models (using libraries like Gson), and handling errors. This significantly reduces boilerplate code and improves code maintainability.

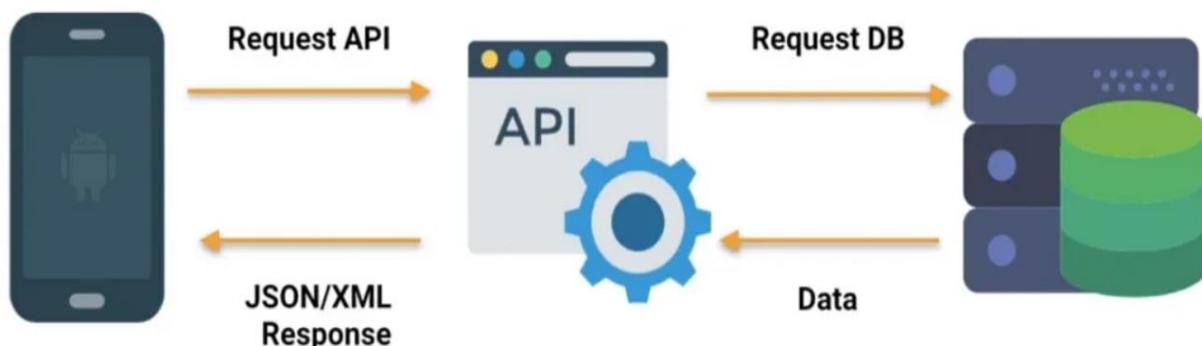


Figure 5.1

4.2.6 Hardware Requirements

- Operating System: Android 12.0 (API level 31) or later
- Processor: Recommended processor architecture (e.g., ARMv8, x86_64)
- RAM: 1 GB or more
- Storage: 500 MB or more free space

4.2.7 Software Requirements

- A device running a compatible version of Android (as specified above)
- A stable internet connection (for features that require internet access)

4.2.8 Installing

Since I care is currently unavailable on app stores like Google Play or App Store, obtaining the app involves downloading the source code and building it on your device. This process requires some technical knowledge and experience with Android development tools.

4.2.8.1. Downloading the Source Code:

Head over to our official GitHub repository and download the source code for I care. You can access the repository here:

https://github.com/thisishisham1/i_care

There are several ways to download the code:

- **Using the GitHub website:** On the GitHub repository page, click the "Code" button. You can then choose to download the ZIP file or clone the repository using Git commands (if you're familiar with Git).

- **Using Git:** If you have Git installed on your computer, open a terminal and navigate to the desired directory. Then, use the `git clone` command followed by the repository URL to clone the code locally.

4.2.8.2. Building and Installing the App:

Once you have downloaded the source code, follow the detailed instructions provided within the repository itself. These instructions will guide you through the process of building and installing the app on your Android device using Android Studio or other development tools.

4.2.9 Design Pattern: MVVM(Model-View-ViewModel)

4.2.9.1 Concept:

The Model-View-ViewModel (MVVM) architectural pattern is a popular approach to building user interfaces, particularly in mobile app development. It promotes a clean separation of concerns by dividing the application into three distinct layers: Model, View, and ViewModel. Each layer has its own specific responsibilities, leading to a more modular, maintainable, and testable codebase.

4.2.9.2 Why Use MVVM ?

Traditional approaches to UI development often mix data logic, presentation logic, and user interface elements within the same codebase. This can lead to several challenges:

- **Spaghetti Code:** Over time, the code becomes intertwined and difficult to understand or modify.

- **Testing Difficulties:** Testing becomes complex as changes in one part of the code can have unintended consequences in other areas.
- **UI inflexibility:** Modifying the UI often requires changes to the underlying logic.

4.2.9.3 Benefits :

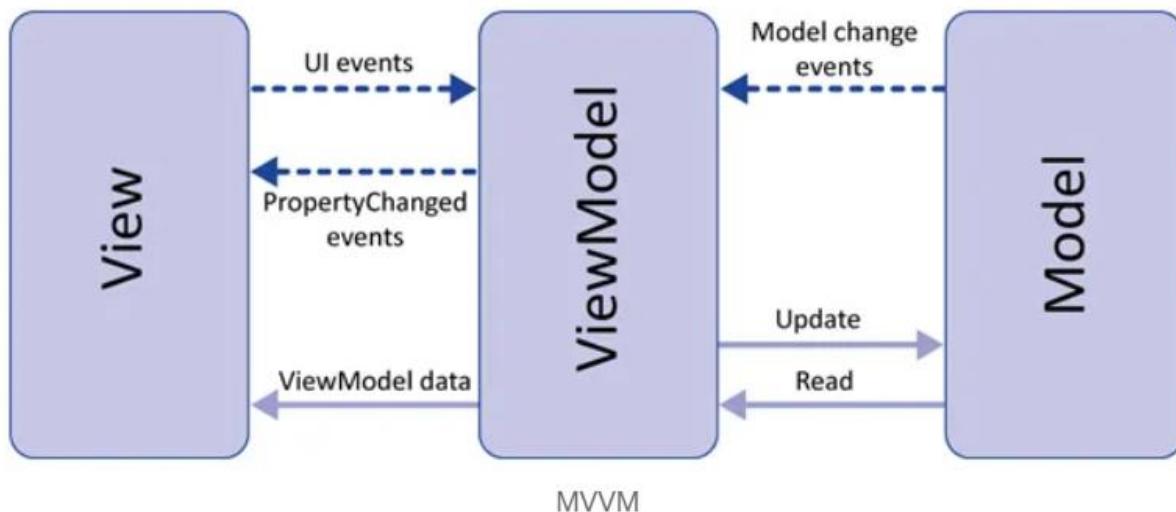
MVVM addresses these challenges by separating responsibilities:

- **Improved Maintainability:** Clear separation of concerns makes the code easier to understand, maintain, and modify. Developers can focus on specific layers without worrying about dependencies on other parts of the application.
- **Enhanced Testability:** Each layer (Model, View, and ViewModel) can be tested independently, allowing for faster bug detection and better overall code quality.
- **Flexible UI Design:** Changes to the UI can be made independently of the underlying data or business logic. This promotes a more flexible and adaptable user interface.
- **Reusability:** ViewModels can be potentially reused across different Views, promoting code reusability and reducing development time.

The Model-View-ViewModel (MVVM) architectural pattern is a cornerstone of modern mobile app development. It promotes a clean separation of concerns by dividing the application into distinct layers, each with specific responsibilities. This approach offers numerous advantages, including improved maintainability, enhanced testability, and a more flexible user interface (UI) design.

Understanding the Layers:

Figure 5.2



1. Model

- Represents the core data of your application.
- Encapsulates the data logic and business rules, independent of the UI.
- This layer can take various forms, including plain data objects, databases, or any source that holds the app's data.

2. View:

- Represents the user interface, the visual layer that users interact with.
- Responsible for displaying information and handling user interactions with UI elements like buttons, text fields, and lists.
- The View typically doesn't contain any application logic and relies on the ViewModel for data and updates.

3. ViewModel:

- Acts as the intermediary between the Model and the View.
- Prepares the data from the Model for presentation in the View,
- Handles user interactions (like button clicks), and updates the Model as needed.
- The ViewModel doesn't own any UI elements directly but communicates with the View through data binding or other mechanisms.

4.3 Frontend Development

4.3.1 Introduction

Welcome to I Care, the ultimate platform designed to streamline the booking management process for doctors, laboratories, and pharmacies. Our user-friendly interface allows healthcare professionals to efficiently manage patient appointments, track schedules, and enhance communication. By centralizing all booking activities, I Care ensures a seamless experience for both medical staff and patients.

4.3.2 Purpose of the App

- doctor: The doctor reviews reservation requests to ensure that they meet medical standards and are compatible with available resources. This helps the approval process coordinate patient care and manage clinic capacity effectively.
- pharmacy: offers an online chat service and a dedicated phone number for patients to communicate directly with pharmacists, enhancing treatment outcomes and overall healthcare quality. These multiple communication channels ensure patients receive appropriate pharmaceutical care through direct interaction with the pharmacy team.
- The lab: reviews reservation requests to ensure they meet medical standards and align with available resources, facilitating the approval process and coordinating patient care. This approach helps manage lab capacity effectively.

4.3.3 Key functionality

- Authentication:
Secure login and registration for doctors, laboratories, and pharmacies.
Support for Google account integration.
- Appointment Management:
Schedule and manage patient appointments with ease. Automated reminders and notifications for upcoming appointments.

- patient Records:
Maintain and access comprehensive patient records. Secure storage and retrieval of medical history and test results.
- Consultation Services:
Offer online medical consultations via video or chat. Efficiently manage consultation schedules and patient queues.
- Laboratory Integration:
Seamless integration with laboratories for test orders and results.
Real-time updates on test status and results.
- Pharmacy Services:
Manage prescription orders and inventory. Coordinate with patients for prescription pickups or deliveries.
- Reporting and Analytics:
Generate detailed reports on appointments, consultations, and prescriptions. Analyze data to improve service efficiency and patient satisfaction.

4.3.4 Software tools

- HTML: is a markup language used to create the structure and content of web pages. It consists of a series of elements that define the different parts of a webpage.
- CSS: is a styling language used to control the presentation and layout of HTML elements on a web page.
- Font-Awesome: A library that provides a vast collection of scalable vector icons, which can be easily customized and included in web projects.
- Bootstrap: is a popular front-end framework used to design and develop responsive and mobile-first websites. It provides a collection of pre-built CSS and JavaScript components, such as navigation bars, buttons, forms, and carousels, that can be easily customized and integrated into web projects.
- JavaScript: is a versatile programming language primarily used for enhancing interactivity within web pages. It enables developers to manipulate elements on a webpage dynamically.

- React.js: is a JavaScript library developed by Facebook for building user interfaces, emphasizing component-based architecture for reusable UI elements.

Here's an overview in four points:

1. Component-Based: React.js encourages the creation of reusable UI components, each managing its own state and rendering logic, which simplifies development and maintenance.
2. Virtual DOM: React uses a virtual DOM to optimize rendering performance. It updates only the necessary parts of the actual DOM when data changes, leading to faster and more efficient UI updates.
3. Single-Page Applications: React is commonly used to build single-page applications (SPAs) where most or all of the application's functionality is contained within a single HTML page, enhancing user experience by providing a smoother and more responsive interface.
- Axios: It is a standalone library that is used to fetch data from API in a more complex way but offers many features like better error management and handles responses in an easier way.



Figure 5.3

4.3.5 Implementation

Front-end:

The front-end of our website is developed using ReactJS, a popular JavaScript library for building interactive user interfaces. ReactJS is selected for its efficiency, flexibility, and strong community support. This section provides an overview of the key concepts, tools, and used in our ReactJS implementation.

Key Concepts and Tools:

1. Component-Based Architecture: ReactJS is built around the concept of reusable components. Each component represents a part of the user interface, such as a header, footer, or form. This modular approach allows for better organization and reusability of code.
2. JSX Syntax: (JavaScript XML) is a syntax extension that allows us to write HTML-like code within JavaScript. This makes it easier to visualize and create the structure of the user interface.
3. Props: are used to pass data between components. They enable components to be more dynamic and reusable by allowing them to receive data from their parent components.
4. React Router : is used to handle navigation between different pages of the application. It allows for the creation of single-page applications (SPAs) where the page does not reload as the user navigates through the site.
5. Styling: Styling in ReactJS can be done using CSS, CSS-in-JS libraries, or pre-processors like SASS. This project utilizes CSS modules to scope styles locally to components, preventing conflicts and improving maintainability.

4.4 Backend Development

Introduction

As a user of Laravel, you benefit from its intuitive features, secure development environment, and robust ecosystem. Whether you're building a simple website or a complex web application, Laravel empowers you to create high-quality solutions efficiently and effectively.

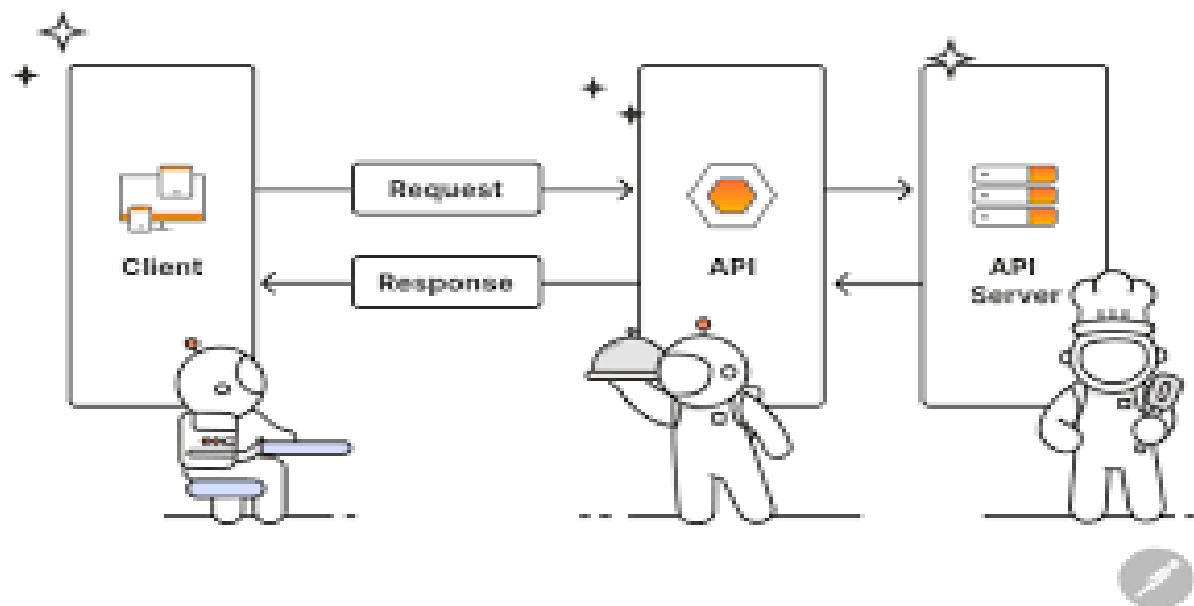


Figure 6.1

Clinic Management System API Documentation

Introduction

The Clinic Management System API is designed to facilitate the management of clinics, patients, reservations, and communication between them. Built on the Laravel framework, it provides a robust set of endpoints to perform CRUD (Create, Read, Update, Delete) operations securely and efficiently.

Base URL

The base URL for accessing the API is:

<https://icare48.000webhostapp.com/>

Authentication

Authentication within the API is handled using Laravel Sanctum. Each request (excluding registration and login endpoints) requires a valid API token to be included in the headers for authorization.

Endpoints Overview

Clinics

- **Register a New Clinic**
 - Endpoint: POST /clinics/register
 - Purpose: Allows clinics to register in the system with necessary details such as name, specialty, address, contact information, etc.
- **Login**
 - Endpoint: POST /clinics/login
 - Purpose: Authenticates a registered clinic to obtain an access token for subsequent API calls.
- **Get Clinic Details**
 - Endpoint: GET /clinics/{id}
 - Purpose: Retrieves detailed information about a specific clinic based on its unique identifier (id).
- **Update Clinic Details**
 - Endpoint: PUT /clinics/{id}
 - Purpose: Allows updating of clinic information such as contact details, working hours, and description.
- **Delete Clinic**
 - Endpoint: DELETE /clinics/{id}
 - Purpose: Deletes the clinic from the system permanently.

Patients

- **Register a New Patient**
 - Endpoint: POST /patients/register
 - Purpose: Enables patients to register with the system by providing personal details and creating a patient profile.
- **Login**
 - Endpoint: POST /patients/login
 - Purpose: Authenticates a registered patient to obtain an access token for subsequent API calls.
- **Get Patient Details**
 - Endpoint: GET /patients/{id}
 - Purpose: Retrieves detailed information about a specific patient based on its unique identifier (id).

Reservations

- **Create a Reservation**
 - Endpoint: POST /reservations
 - Purpose: Allows patients to make appointments with a specific clinic by providing necessary details such as patient ID and clinic ID.
- **Get Reservation Details**
 - Endpoint: GET /reservations/{id}
 - Purpose: Retrieves details of a reservation based on its unique identifier (id).
- **Delete Reservation**
 - Endpoint: DELETE /reservations/{id}
 - Purpose: Cancels and removes a reservation from the system.

Chats

- **Send a Message**
 - Endpoint: POST /chats
 - Purpose: Facilitates communication between clinics and patients by enabling them to send messages to each other.
- **Get Chat History**
 - Endpoint: GET /chats/{sender_id}/{receiver_id}
 - Purpose: Retrieves the messages exchanged between a specific clinic and patient.

Error Handling

The API returns appropriate HTTP status codes and JSON-formatted error messages in case of any errors or invalid requests. Error responses include details such as error codes and descriptions to assist developers in troubleshooting.

Security

All API endpoints are secured using Laravel Sanctum for authentication, ensuring that only authorized users can access the system. Secure coding practices are followed to protect against common web vulnerabilities.

Rate Limiting

To prevent abuse and ensure fair usage, rate limiting is applied to API requests. Exceeding the allowed limit results in HTTP 429 (Too Many Requests) responses, prompting clients to retry after a specified time.

Conclusion

The Clinic Management System API offers a comprehensive suite of endpoints to manage clinics, patients, reservations, and communication effectively. It is designed to be scalable, secure, and easy to integrate into existing healthcare management systems.

Key Features of Laravel

1. **Expressive Routing:** Laravel's routing system allows developers to define clean and readable application routes, making it easier to manage and navigate through different parts of your web application.
2. **Eloquent ORM:** Laravel's Eloquent ORM simplifies database interactions by allowing you to work with database tables using PHP syntax. This makes querying databases and defining relationships between tables straightforward.
3. **Blade Templating Engine:** Blade provides a fast and efficient templating engine that allows you to write clean and reusable templates using straightforward PHP syntax. It helps you separate your application's logic from its presentation layer.
4. **Artisan CLI:** Laravel includes Artisan, a command-line interface that automates repetitive tasks such as database migrations, generating boilerplate code, and managing application components. It saves you time by streamlining common development tasks.
5. **Authentication and Authorization:** Laravel provides built-in mechanisms for user authentication and authorization, making it easy for you to implement secure user login, registration, and access control features in your application.
6. **Testing Support:** Laravel supports unit testing out of the box with PHPUnit, allowing you to write and run tests to ensure your application functions correctly. This helps you maintain code quality and identify issues early in the development process.
7. **Security Features:** Laravel prioritizes security with features like hashed password storage, protection against SQL injection attacks, cross-site request forgery (CSRF) protection, and secure authentication mechanisms, ensuring your application's data remains safe.
8. **Community and Resources:** Laravel has a vibrant community of developers and extensive documentation. It provides tutorials, forums, and third-party packages via Composer, enhancing your ability to learn, troubleshoot, and extend Laravel's functionality.

Chapter 5: Conclusion and Future Work

5.1. Conclusion

The Smart Medical and Healthcare Application stands as a transformative innovation in the healthcare sector, empowering patients to access intelligent medical diagnoses and scan medical imaging or their food calories by AI, book appointments, purchase a prescription, interact with doctors, make a personality analysis test, get notification remainder on appointments time, and manage their healthcare needs seamlessly.

The web application provides clinic doctors, and laboratory specialists to manage and scheduling patients' appointments and helping pharmacists to manage their prescription and medication orders.

The mobile and web applications leverage artificial intelligence models and user-friendly interfaces to deliver an exceptional user experience.

In conclusion, the smart medical and healthcare application leverages cutting-edge AI technology to deliver a seamless and exceptional user experience for both patients and healthcare providers. This project represents a significant step forward in making healthcare more accessible, personalized, and efficient, ultimately improving patient outcomes and healthcare service delivery.

5.2. Future Improvements

- Support all world languages.
- Enhance the AI models accuracy as much as possible.
- Adding an NLP AI based model trained on Arabic dataset.
- Adding all types of medical imaging scans to detect all possible diseases.
- Train the food identifier model on the Arabian famous food.
- Adding an AI option to read the medical tests and discuss it.
- Adding spam detection on comments.
- Adding an AI bot that works as a psychiatrist to diagnose mental illnesses.

References

Datasets used for trained the artificial intelligence models:

X-ray body type dataset:

<https://www.kaggle.com/datasets/ibombonato/xray-body-images-in-png-unifesp-competition>

ECG dataset:

<https://www.kaggle.com/datasets/shayanfazeli/heartbeat>

COVID19+PNEUMONIA+NORMAL Chest X-Ray Image Dataset:

<https://www.kaggle.com/datasets/sachinkumar413/covid-pneumonia-normal-chest-xray-images>

bone fracture detection using x-rays dataset:

<https://www.kaggle.com/datasets/vuppalaadithyasairam/bone-fracture-detection-using-xrays>

Chest X-ray - 17 Diseases dataset:

https://www.kaggle.com/datasets/trainingdatapro/chest-xray-17-diseases?select=xray_chest.csv

Brain Tumor Classification (MRI) dataset:

<https://www.kaggle.com/datasets/sartajbhuvaji/brain-tumor-classification-mri>

Brain tumors 256x256 dataset:

<https://www.kaggle.com/datasets/thomasdubail/brain-tumors-256x256>

Bone Break Classification Image Dataset:

<https://www.kaggle.com/datasets/pkdarabi/bone-break-classification-image-dataset>

Breast Cancer Diagnosis dataset:

<https://www.kaggle.com/datasets/faysalmiah1721758/breast-cancer-diagnosis>

RSNA Breast Cancer Detection - 256x256 pngs dataset:

https://www.kaggle.com/datasets/theoviel/rsna-breast-cancer-256-pngs?select=10042_495770405.png

Breast Cancer Patients MRI's dataset:

<https://www.kaggle.com/datasets/uzairkhan45/breast-cancer-patients-mris>

Bone Fracture Atlas dataset:

https://figshare.com/articles/dataset/The_dataset/22363012

MRI image Breast cancer dataset:

<https://www.kaggle.com/datasets/raselislambabu/mri-image-breast-cancer>

Medical information sites:

<https://www.ao.org/>

<https://www.who.int/>

<https://my.clevelandclinic.org/>

Courses:

Huawei HCIA AI v3.5:

https://e.huawei.com/en/talent/outPage/#/sxz-course/home?courseId=LEr4w0UtMYfpbhkgY_Yd5QqFAO0

Machine learning, deep learning, and chatbot courses:

https://www.youtube.com/playlist?list=PL6-3IRz2XF5X-1zMZdmkvGAx1a3kIm7_I

<https://www.youtube.com/playlist?list=PL6-3IRz2XF5VAZOahFLIj2k4WK-8TRc5I>

https://www.youtube.com/playlist?list=PL6-3IRz2XF5Ua2KG_Fl3lbZ-kKi3-Np0

GitHub Project Link

[GitHub \(click me\)](#)



"نظام ذكي لإدارة المستشفيات"

يهدف المشروع إلى عمل دائرة مغلقة حول جميع شخصيات القطاع الطبي كالمريض والأطباء في العيادات الخاصة والصيدليات والمعامل.

بالإضافة إلى اتاحة سهولة استخدام لفحص طبي سريع باستخدام نماذج الذكاء الاصطناعي فتبدأ العملية بفتح المريض التطبيق والتحدث مع الذكاء الاصطناعي الشات بوت لإعطائه اعراض المرض له ثم سيتوقع النموذج المرض مع ذكر معلومات عنه كالأعراض والأسباب وما إلى ذلك ويطلب منه التحاليل او الاشعارات الطبية اللازمة لعملية التشخيص ويوجه المريض إلى صفحة حجوزات المعامل لحجز موعد لإجراء التحليل المطلوب من ثم يستلم المريض النتائج على التطبيق ويوجهها إلى نماذج الذكاء الاصطناعي الخاص بقراءة الصور للتشخيص النهائي.

بإمكان المريض أيضا البحث عن العيادات والمعامل والصيدليات القريبة لحجز موعد أو طلب جرعات الدواء.

بإمكان المريض أن يجري اختبار لتحليل الشخصية والحصول على نوع شخصيته نسبا إلى عالم النفس الشهير "مايرز بريجز" وتلقي معلومات عن شخصيته تساعد في اختيارات كالوظيفة القسم المناسب.

بإمكان الأطباء الدخول على موقع الويب بالإضافة عيادة، معمل، أو صيدلية على النظام وتنظيم مواعيد حجوزات المرضى لديه.