AI Medical Care

Graduation Project, Part-II (CS482)

Computer Science Department

Faculty of Computer and Information

Luxor University

Project Supervisors

Dr. Mohamed Abdel-Hameid

Eng. Safynaz Abdel-Fattah

Submitted by

Abdelrahman Mahmoud Mohamed

Mohamed Ahmed Mohamed

Rabee Adel Beabesh

Soliman Ali Soliman

Youssof Harby Okeil

27th June 2021

ABSTRACT

The standard way of doing a medical diagnose enforces the patient and the doctor to exist together within a physical place such as a hospital or a clinic, which may be an obstacle especially for patients in remote places also boosts the possibility of infection-transmission among patients, also one of the difficulties that may struggle the process of diagnosing is the inefficient techniques to deal with the medical history of the patient which is mainly presented as an un-organized paper-work; that imposes the doctor to exert more effort to fully observe it, which decreases the productivity rate and increase the opportunities for a misdiagnosed patient or an in-proper treatment especially for the new and in-experienced doctors, which is dangerous.[1]

Our proposed system can makes advantage of the rapid growth of technology especially in the artificial intelligence field and natural language processing, developing text-based systems for health diagnosis where patients can submit a text or a voice-speech with a complaint of a disease he/she suffers through a mobile application or a web-based application, the system with his role analyzes the text to extract symptoms and specify a disease and a proper treatment, Taking into account the medical history of the patient in case he/she suffers chronic diseases or any on-going medication and report the final result to the doctor to confirm the process, in other words, it acts like a doctor-assistance to aid in the diagnosing process with the necessity of the doctor-existence to confirm the results.[2]

Contents List

List of Figures

List of Tables

Abbreviation

Keyword	Meaning
Natural Language Processing (NLP)	Natural language processing (NLP) is a field of artificial intelligence in which computers analyze, understand, and derive meaning from human language in a smart and useful way.
Part-Of-Speech (POS)	Part-of-speech (POS) is a process of converting a sentence to forms – list of words, list of tuples (where each tuple is having a form (word, tag)).
Classifier	Classifier is an algorithm that automatically orders or categorizes data into one or more of a set of "classes".
International Classification of Diseases (ICD)	The International Classification of Diseases (ICD) is a globally used diagnostic tool for epidemiology, health management and clinical purposes. The ICD is maintained by the World Health Organization (WHO), which is the directing and coordinating authority for health within the United Nations System. The ICD is originally designed as a health care classification system, providing a system of diagnostic codes for classifying diseases, including nuanced classifications of a wide variety of signs, symptoms, abnormal findings, complaints, social circumstances, and external causes of injury or disease

Medical Dictionary for Regulatory Activities (MedDRA)	The Medical Dictionary for Regulatory Activities (MedDRA) (link is external) is an internationally used set of terms relating to medical conditions, medicines and medical devices. It was created to assist regulators with sharing information. It is also used by industry, academics, health professionals and other organizations that communicate medical information.
WordNet	WordNet is a lexical database of semantic relations between words in more than 200 languages. WordNet links words into semantic relations including synonyms, hyponyms, and meronyms. The synonyms are grouped into sunsets with short definitions and usage examples.

Table 1. Abbreviation

Chapter 1

Introduction

1.1 Introduction

Remote diagnosis systems are becoming increasingly popular and accurate, with enormous advantages such as cost-effectiveness, fast and reliable decision support for medical diagnostics, and treatment and prevention of disease, illness, injury, and other physical and mental damages in human beings. The rise in remote health services (or telehealth) offered by healthcare institutions coincided with the evolution of assisted living systems and environments, aiming to widen the possibility for older and disadvantaged people to access appropriate healthcare services and thus improve their health status and clinical outcome. With the increase in the innovation of medical technologies, there is a need to adopt medical expert systems that will oversee and control diagnosis and treatment processes Medical diagnostic processes carried out with the aid of computer-related technology which is on the rise daily have improved the experience and capabilities of physicians to make an effective diagnosis of diseases while employing novel signal processing techniques for analysis of patient's physiological data and deep neural networks for decision support. With the rise of artificial intelligence (AI) techniques, the intelligent doctor-assistant have appeared as a promising direction to emphasize the accuracy of diagnosing diseases and do deep analysis on the health records and symptoms of the patient to provide a reliable health state of patients.

Artificial intelligence (AI) is increasingly being adopted across the healthcare industry, and some of the most exciting AI applications leverage natural language processing (NLP). Simply put, NLP is a specialized branch of AI focused on the interpretation and manipulation of human-generated spoken or written data which showed great potential in effective clinical decision support.

1.2 Background

Hospitals are the most widely used means by which a sick person gets medical check-ups, disease diagnosis, and treatment recommendation. People consider it as the most reliable means to check their health status. The proposed system is to create an alternative to this conventional method of visiting a hospital and making an appointment with a doctor to get a diagnosis. This research intends to apply the concepts of natural language processing and machine learning to create a medical care application. Executing this proposed framework can help people avoid the time-consuming method of visiting hospitals by using this free of cost application, wherever they are. Human-computer speech is gaining traction as a method of computer interaction. There has been a recent rise in speech-based search engines and assistants like Siri, Google Chrome, and Cortana.[1]

Natural language processing (NLP) techniques such as NLTK for Python can be applied to speech analysis, and intelligent responses can be found by designing an engine to deliver appropriate human responses. This type of program is called Medical care. In this project, we will deal with Natural Language Processing (NLP) techniques by sending a voice record that is converted into a text record, then analyzing speech and extracting symptoms, then disease diagnosis and treatment recommendation.[2]

1.3 Related work

Title	Authors	Year	Abstract
Text Messaging-Based Medical Diagnosis Using Natural Language Processing and Fuzzy Logic [1]	Omoregbe, Ndaman, Misra, Abayomi-Alli, and Damaševičius	Sep 2020	Ability to successfully build a text-based medical diagnosis system, which provides a personalized diagnosis utilizing self-input response from users to effectively suggest a disease diagnosis. The proposed system was able to combine NLP and machine learning algorithms for SMS and Telegram bot. The system was able to suggest a diagnosis using a direct approach of the question and answering technique to offer a diagnosis
A text based drug query system for mobile phones [2]	A. Langer, R. Banga, A. Mittal, L. V. Subramaniam, and P. Sondhi	July 2014	Used several NLP tools along with classification methods to process the drug-related questions. They developed a natural language-based interface that enables the users to phrase their queries and get an accurate result up to 81% in classifying drug-related questions
A text mining approach to automated healthcare for the masses [3]	V. S. Pendyala, Y. Fang, J. Holliday, and A. Zalzala	Oct. 2014	Presented an application that allows machines to take on the function of life support. The focus of the study was based on medical diagnosis, and an experiment was conducted to show the relationship of information retrieval and text mining to the medical diagnosis problem. The study concludes that the proposed system would help in improving the goals of providing a ubiquitous medical diagnosis.
A Laboratory Test Expert System for Clinical Diagnosis Support in Primary Health Care[4].	R. Carson-Stevens, J.-A. Medina-Merodio, Herraiz, and J.-M. Gutierrez-Martinez	4 August 2015	Presented a rule-based expert system using the list of likely diseases regarding laboratory test results for diagnosis. The authors concluded that the proposed system clinically gave a better accuracy and speed, thereby improving the efficiency and quality of service.

From narrative descriptions to MedDRA: automagically encoding adverse drug reactions [5].	C. Combi, M. Zorzi, G. Pozzani, U. Moretti, and E. Arzenton	August 2018	Proposed an NLP method for the transcoding of natural text descriptions of adverse drug reactions into MedDRA standard terms, reaching an average precision and recall of 91.8% and 86.9%.
Automated classification of primary care patient safety incident report content and severity using supervised machine learning (ML) approaches[6].	H. P. Kanegaye, A. Anastasiou, A. Edwards	March 2019	Analyzed patient safety incident reports written in free text to categorize incident type and the severity of outcome, reaching an accuracy of 0.891 and 0.708, respectively.
Improving Consumer Understanding of Medical Text: Development and Validation of a New Sub Simplify Algorithm to Automatically Generate Term Explanations in English and Spanish [7].	N. Kloehn, G. Leroy, D. Kauchak.	April 2018	Generate explanations for complex medical terms in Spanish and English using WordNet synonyms and summaries, as well as the word embedding vector as a source of knowledge.
An interpretable natural language processing system for written medical examination assessment [8]	A. Sarker, A. Z. Klein, J. Mee, P. Harik, and G. Gonzalez-Hernandez	October 2019	Used fuzzy logic and set theory-based methods for learning from a limited number of annotated examples of unstructured answers in health examinations for recognizing correct concepts with an average F1-measure of 0.89.

Table 2. Related work

Chapter 2

Domain Analysis and Techniques

2.1 Domain Analysis

The main issue that we have witnessed recently due to Corona pandemic highlighted our need for a remote systems Especially when it comes to medical care; a lot of patients live in a remote places with nearly any doctors nearby, Most of the patients treat their medical records lightly which may cause a great dangerous, also in-experienced doctors and misleading medical-records may yield to misdiagnose and in-proper treatment which may come with a great harm for the patient , all of the mentioned above represents an opportunity for intelligent systems and natural language processing to step up for the healthcare industry to break down antique silos and plug gaps in the care delivery system to make progress for the patient segment.

Natural Language Processing (NLP) can be stated as the automatic processing of the natural human language by a machine. It is a specialized branch of Artificial Intelligence which primarily focuses on interpretation as well as human generated data – text or speech based. The technology has various sub-disciplines, including Natural Language Query, Natural Language Generation, and Natural Language Understanding, the main use case of NLP is Comprehending human speech and extracting its meaning.[2]

Speech Recognition – NLP has matured its use case in speech recognition over the years by allowing clinicians to transcribe notes for useful Medical Records data entry. Front-end speech recognition eliminates the task of physicians to dictate notes instead of having to sit at a point of care, while back-end technology works to detect and correct any errors in the transcription before passing it on for human proofing, Machine Learning in healthcare has touched clinical documentation, freeing up physicians from the manual and complex structure of Electronic

Medical-Records, allowing them to focus more on care delivery. This has been possible because of speech-to-text dictation and formulated data entry that capture structures data at the point of care.

NLP solutions can help bridge the gap between complex medical terms and patients' understanding of their health. NLP can be an excellent way to combat the Medical Record distress. Many clinicians utilize NLP as an alternative method of typing and handwriting notes, NLP algorithms can extract vital information from large datasets and provide physicians with the right tools to treat patients with complex issues.[3]

Applying Machine learning algorithms and classifiers, have shown promising results in classifying free text and make a decision to assist medical experts in their diagnosis would serve as a boost in successfully improving healthcare services through effective analysis of narrative text of symptoms provided by a patient.

The proposed system serve as a doctor-assistant by combining the NLP and machine learning algorithms to aid the doctor in the diagnosing and treatment process which make the life easier for the patients and doctors in addition to the exponential growth of productivity and accuracy.[4]

2.2 Techniques

We provide an AI Model based on NLP and ML techniques to hear patients' complaints to extract symptoms to diagnose disease finally decision for the doctor.

- NLP model: to convert audio to text and do preprocessing extract symptoms exchange it to a vector of a feature.
- ML model: take a vector of symptoms and classify this to diagnose and get this specific disease.

We will implement this as a back-end and host it on serve and provide web service to can use it on android or desktop or web or other organization.

So… in android:

Application components:

- Login page: This page has the user name and password, and it also has a button for the registration page if the user is not registered.
- Registration page: A page that enables the user to register, to be able to use the application as a doctor or normal user.

After the user logs in:

- The Doctor will see a page containing:
  - Diagnose page: this contains options to scan QR code for the patient to see his medical history, Start and stop recording patient complaint to upload recorded audio to diagnose and option to write symptoms as text to diagnose and option to add new medical record and note for this patient, and option to confirm diagnose' result.
- The Normal user (Patient) will see a page containing:

Diagnose page: this contains the options to start and stop recording his complaint to upload recorded audio to diagnose and option to write symptoms as text to diagnose and option to see his medical history and add new medical record and note for his medical history.

2.3 Risk/Constraints

2.3.1 Risks

Table 3. Risks

Risk	Strategy	Strategy Type	Priority	Probability
Patient may not know how to use the application	Providing Manual Guide to the whole system features.	Avoidance Strategy	High	Moderate
Malfunction issues with patient`s microphone	Report an error with the microphone and providing a text-based method	Minimization Strategy	High	Low
Malfunction with the server or the API hosting the Machine Learning Model	Hosting the model and the API on several servers	Contingency Plan	High	Low
Malfunction with the servers that host the database and the medical records	Hosting the database and the several servers	Contingency Plan	High	Low

2.3.2 Constraints

a. User should have the application on his device
b. User must have internet on his device and always run when using application
c. User should speak in a clear voice
d. User should talk about symptoms in a clear and direct way
e. User should know how to deal with the application.

2.4 Project Plan

2.4.1 Project Plan

phase	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun
Gathering Information									
Define Requirements									
analysis									
design									
implementation									
Develop AI System									
Testing and Final Discussion									

Table 4. Project Plan

2.5 Quality Assurance Plan

- Black box:
  - In this stage we use test dataset as input to our system to ensure the accuracy of output.

- White box:-
  - Unity testing
    In this stage of testing, we will take every components of our AI health care system such as web service, NLP learning model, machine learning model, android application to test them separately.

  - Integration testing:
    In this stage of testing, we will take every components of our AI health care system such as web service, NLP learning model, machine learning model, android application to test them separately

  - Validation testing:
    Validation testing is the process of ensuring if the tested and developed application satisfies its functionality requirements. The business requirement logic or scenarios have to be tested in detail. All the critical functionalities of an application must be tested here.

  - Alpha:
    In this part, a group of testers in our team test the product in a laboratory environment to ensure efficiency of product and fix errors.

  - Beta:
    In this stage of testing the application has been sent to some doctors to be tested on clinical and hospital and test the system efficiency and outputs is correct or not, retrieve feedback to our team.

2.6 Requirements

2.6.1 Functional Requirements

1. A fully functional mobile app that controls work-flow with servers, APIs, doctors, and patients in organized, accurate, and quick methods.
2. The System must store all required data about doctors (ids, info, and assigned patients), patients (ids, info, medical history, medical reports, and treatments).
3. A Speech Recognition System to record the conversation between doctor and patient.
4. A Server that stores data and Machine Learning Models.
5. API to guarantee communication between server and application.
6. A fully functional and tested Machine Learning Model that analyzes texts to extract symptoms and diagnose diseases.
7. A system to specify proper treatments based on diagnosing.
8. A user interface that enables patients and doctors to create accounts, add medical records, check their medical history.
9. The doctor and the patient must be able to start a diagnosis process.
10. The Application must forward the description text of the symptoms to the model for analysis through an API
11. The doctor and the patient must be able to receive the generated report with the full diagnosis report and treatment and confirm it through an API.
12. Web service to be used as a website or/and mobile app.

2.6.2 Non-Functional Requirements

1- Fast APIs and data retrieval from servers within seconds.

2- Quick Analysis and Classifying Texts using efficient algorithms.

3- A clear, attractive, responsive, less-leggy, and efficient User-interface that guarantees a good user experience and handles a variety of tasks.

4- High security to prevent unauthorized access and guarantee the safety and integrity of data.

5- A well-organized and manageable system architecture that eases the process of maintenance, recoverability and scalability.

2.7 System Request

Machine learning in medical diagnosis has helped several healthcare organizations to improve the patient's health and reduce healthcare costs, using superior diagnostic tools and effective treatment plans. It is now used in healthcare to make an almost perfect diagnosis, predict readmissions, recommend medicines, and identify high-risk patients. These predictions and insights are drawn using patient records and data sets along with the symptoms exhibited by the patient so the proposed system eases the process of diagnosis for the doctor and patient through automation of the diagnose process to reduce the chance of in-proper diagnosis or treatments also there`s no longer need for a physical place or any travelling cost for the patient they can just do it online and the proposed system ensures the value of customer experience and customer satisfaction.[5]

- Functionality: -
  1- The Ability to record the text/speech description of the symptoms.

  2- The ability to analyze those records and predict disease and the proper treatment based on the medical history.

  3- The ability to generate full reports about the status of the patient.

  4- The ability to store medical histories for the patients as a reference for them.

- Expected Value: -
  1- Less-cost examinations process.

  2- More-fast diagnosis process.

  3- More organized reports and records.

  4- Less paper-work stuff.

  5- More accurate and reliable diagnosis.

  6- Ease-to-use application and friendly User-interfaces.

  7- Increase productivity.

8- Customer-satisfaction.

- Special Constraints: -
  - The ability of doctors and patients to deal with Technology.
  - The short development life-time.

Chapter 3

Proposed System and Methodology

3.1 System Architecture

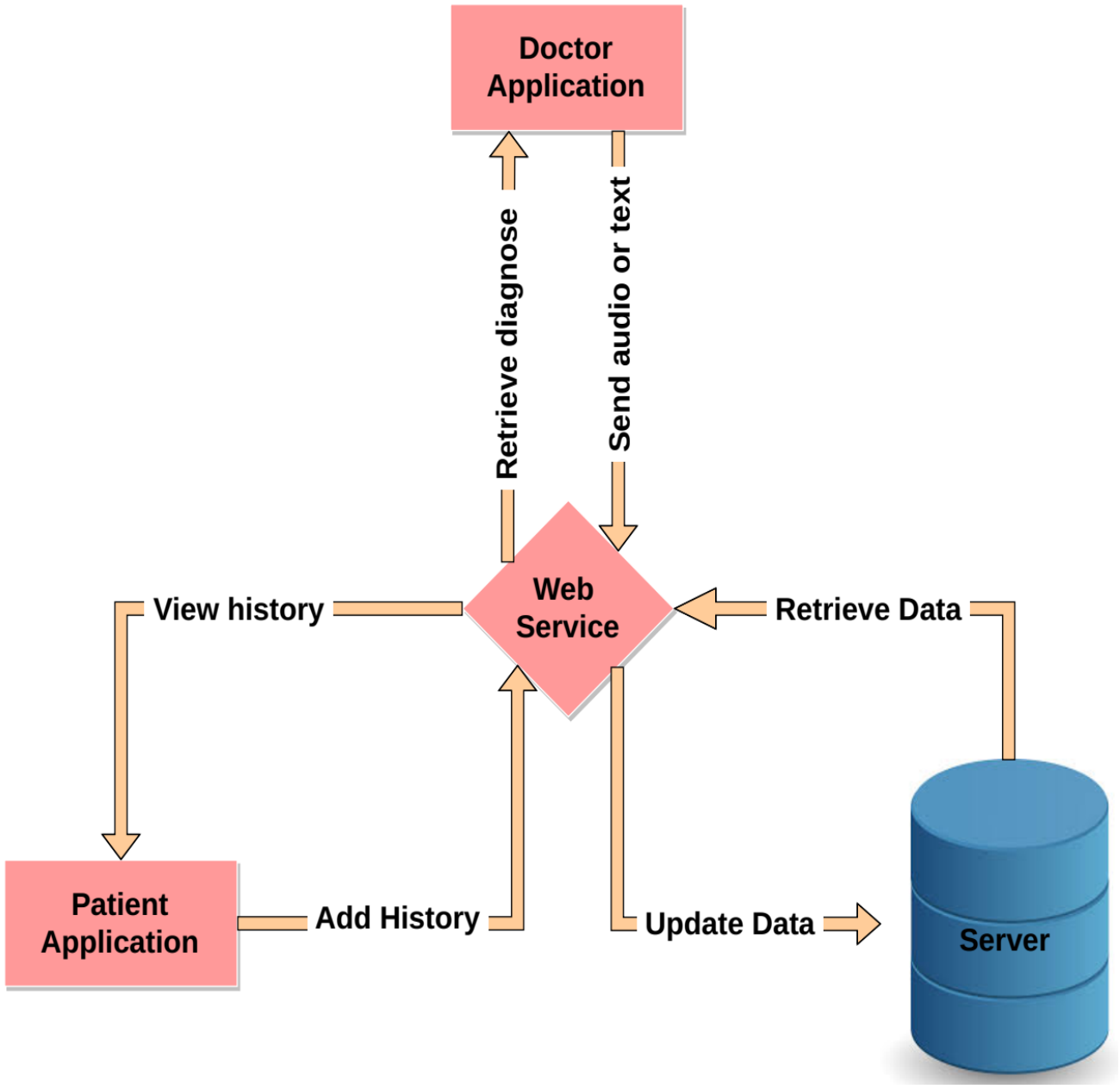


Figure 1. System architecture

System Architecture

- The main connection point is the Web Service (API) that connects all the system components
- Patient can update or retrieve medical records by sending a request to the API to retrieve/store records from/to the database server.
- Doctor sends a text-based description of symptoms to the hosted intelligent model through API
- And receives back the diagnose in addition to data retrieval from the database server.
- API forwards all the results of diagnose processes to the database server after confirmation to store them.

3.2 System Use-Cases

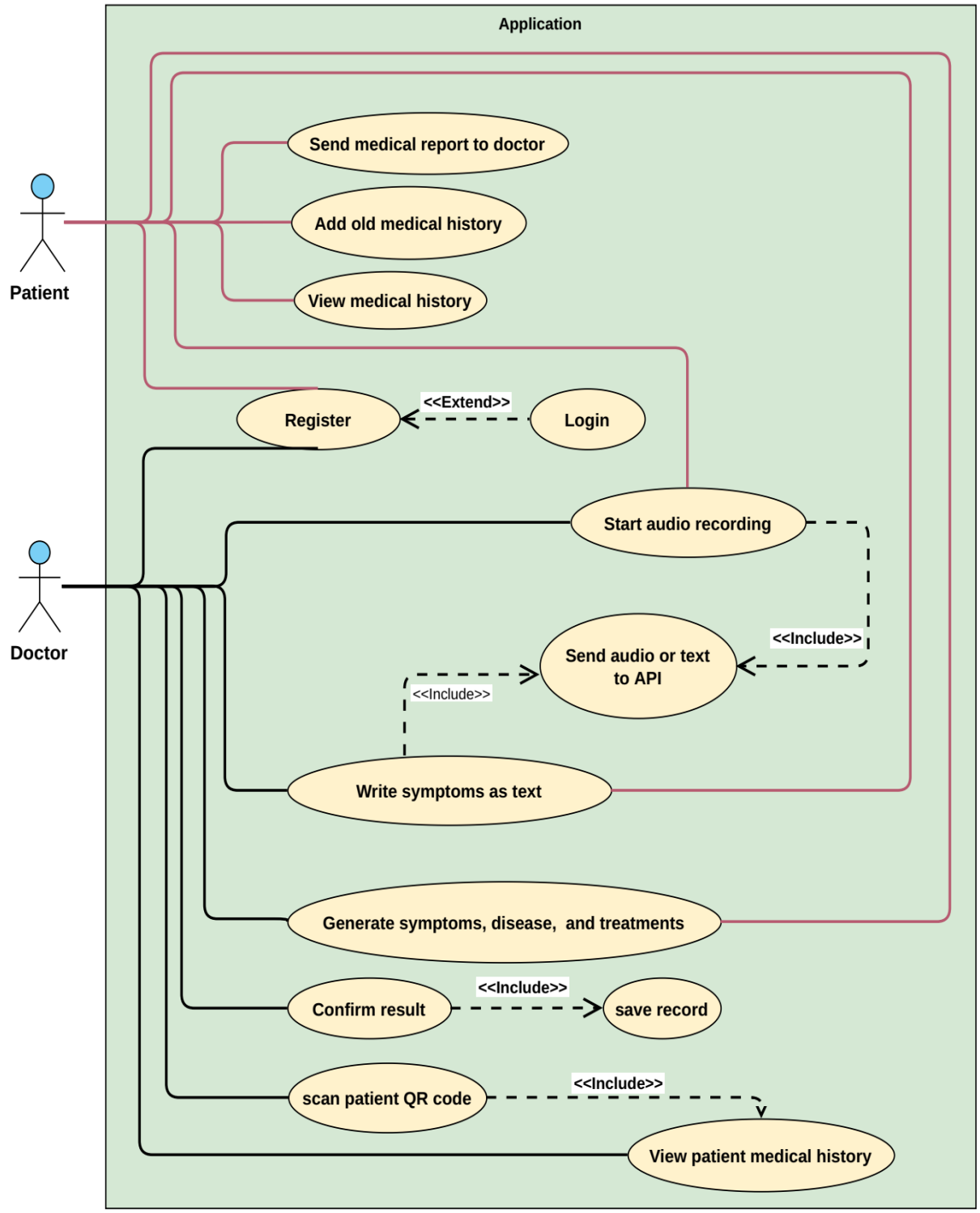


Figure 2. Application Use case

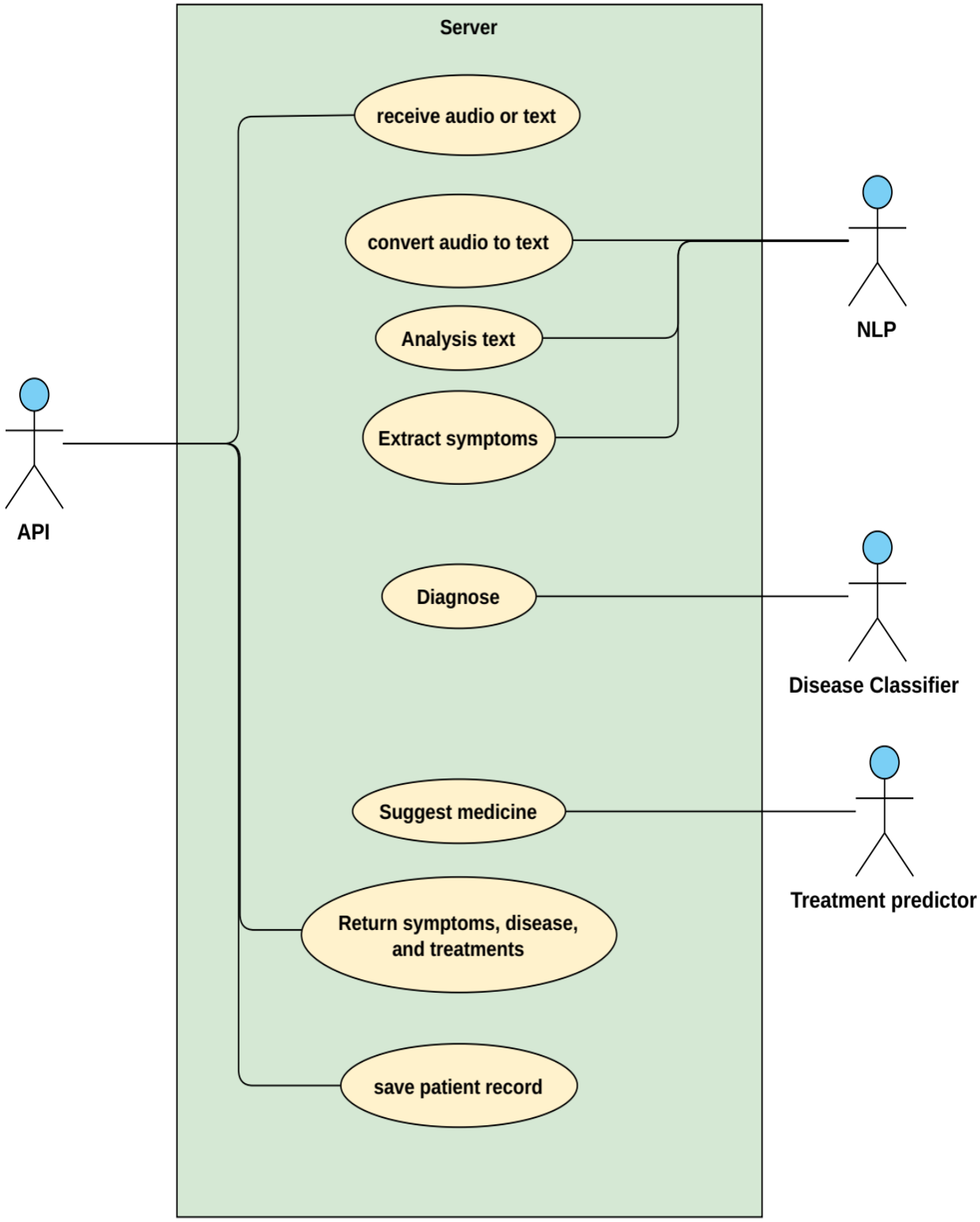


Figure 3. Server use case

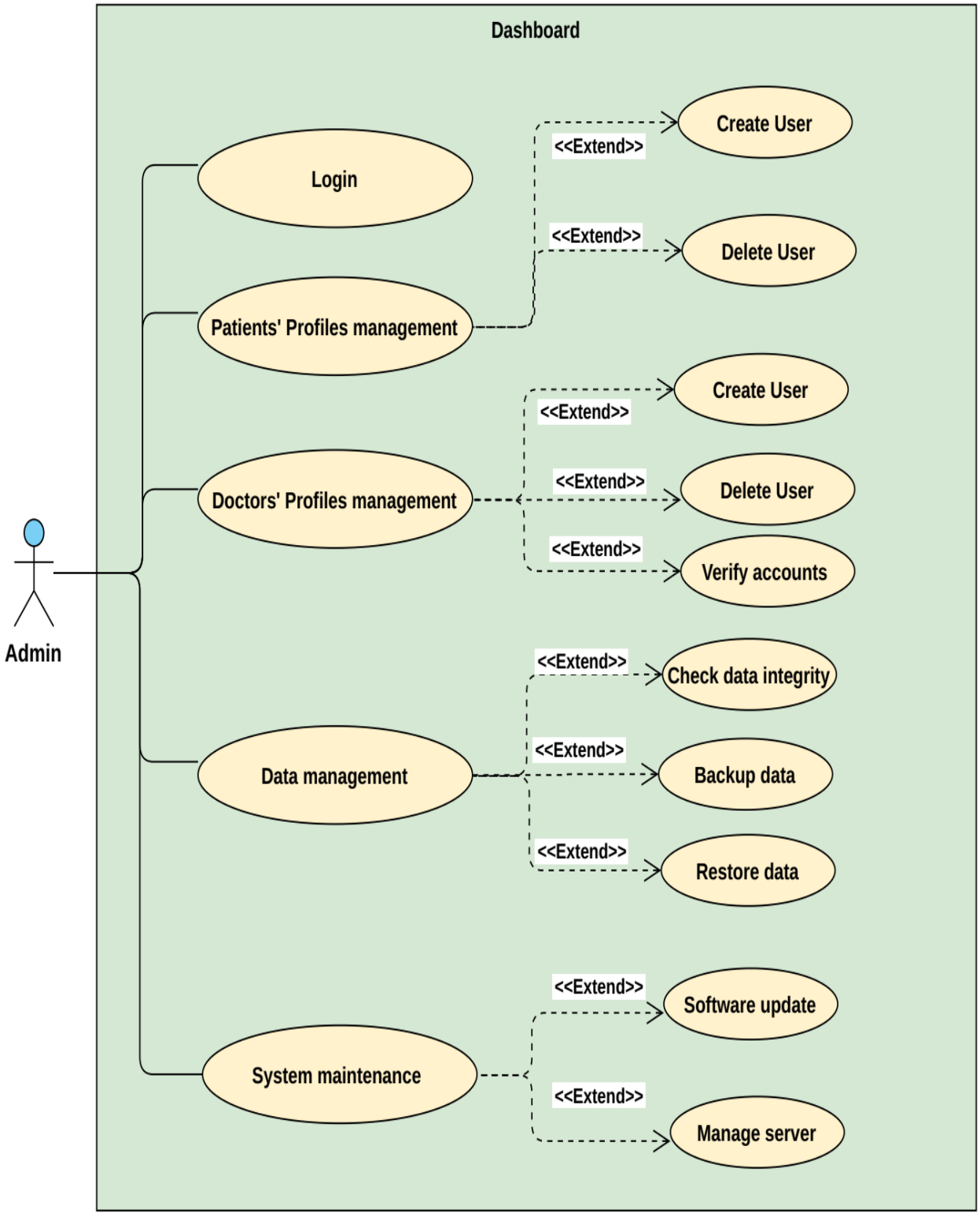
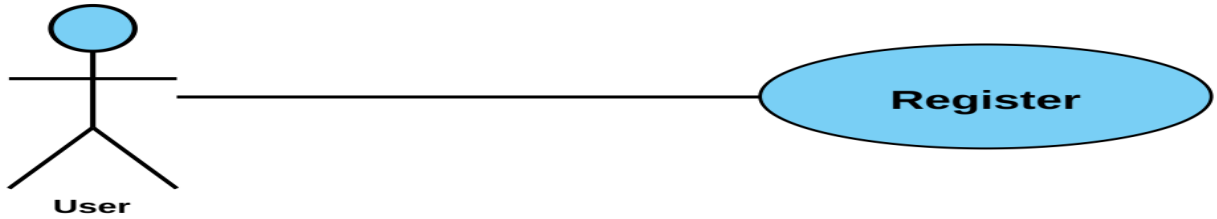


Figure 4. Admin dashboard use case

3.3 Use Case Description (Use case scenario)

Register: To use app user must have an account, and enters all required information



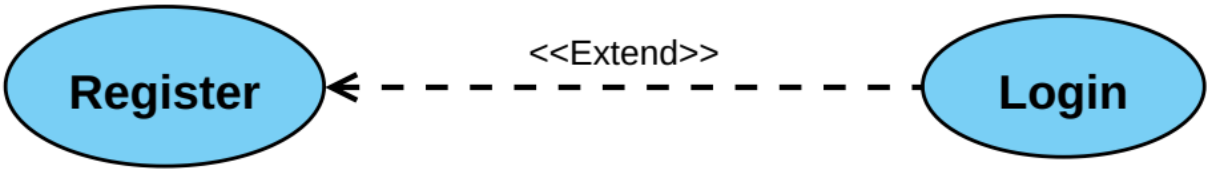
Use case name	Register
Unique ID	HealthCare-User-001
Area	Application
Actor(s)	User (Patient, or Doctor)
Description	User creates account
Triggering Event	User click “Register” button in the application
Preconditions	- The user needs to download application then open it - The user needs to have internet access
Postconditions	- User has successfully created account
Assumptions	- User have Medical Care application - A valid data

Steps Performed	**Information for Steps**
1- Open application 2- Choose if he is a patient or doctor 3- User enters his data 4- Click on “Create Account” button 5- Validation of entered data by application	Step 3: Name, Username, Password, E-mail, SSN, EMSN (Egyptian Medical Syndicate Number if he was a doctor)

Extensions (Alternative Flows)	- If the downloading interrupted for any reason, use should try again and download it - If user entered a non-valid data, a warning message should appear to him

Table 5. Register Use Case Description

Login: To log in the app the user must enter username and password



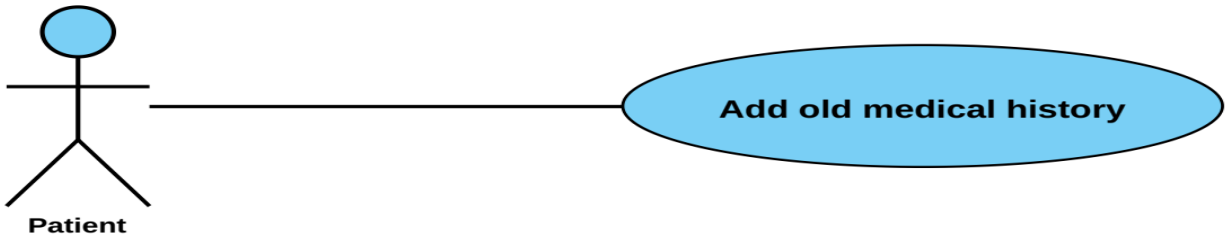
Use case name	Login
Unique ID	HealthCare-User-002
Area	Application
Actor(s)	User (Patient, or Doctor)
Description	User login to his account
Triggering Event	User click “Login” button in the application
Preconditions	- The user needs to download application then open it - The user needs to have internet access - The user needs to have account
Postconditions	- User has successfully logged in to his account
Assumptions	- User have Medical Care application - A valid data

Steps Performed	**Information for Steps**
1- Open application 2- User enters his data 3- Click on “Login” button 4- Validation of entered data by application	Step 2: E-mail, Password

Extensions (Alternative Flows)	- If user entered a non-valid data, a warning message should appear to him

Table 6. Login Use Case Description

Add Old Medical History: patient add his old medical records



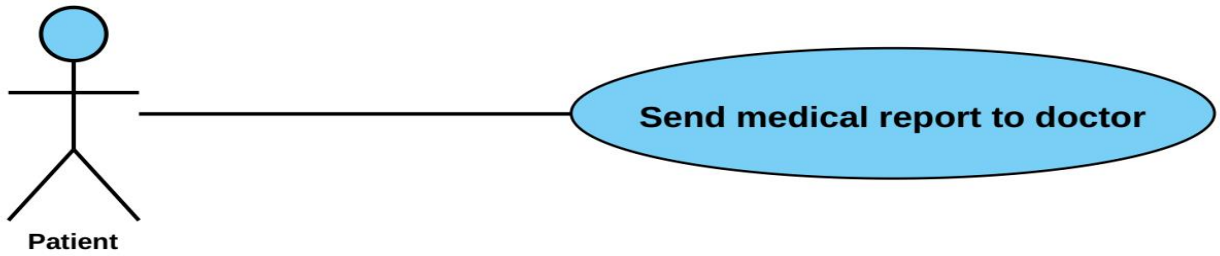
Use case name	Adding old medical history
Unique ID	HealthCare-Patient-001
Area	Application
Actor(s)	Patient
Description	Patient adds old medical history
Triggering Event	Patient click “Add old medical history” button in the application
Preconditions	- The Patient needs to login in to his account
Postconditions	- Patient has successfully added old medical history to his account
Assumptions	- Patient have Medical Care application - A valid data

Steps Performed	**Information for Steps**
1- Open application 2- Patient log in 3- Patient click “Add old medical history” button 4- Patient Enters his data 5- Patient clicks on “Save” button 6- Validation of entered data by application	Step 2: E-mail, Password Step 4: Symptoms, Disease, Treatment

Extensions (Alternative Flows)	- If user entered a non-valid data, a warning message should appear to him

Table 7. Adding Old Medical History Use Case Description

Send medical report to doctor: Patient after receive his Medical Care report he can send to his doctor



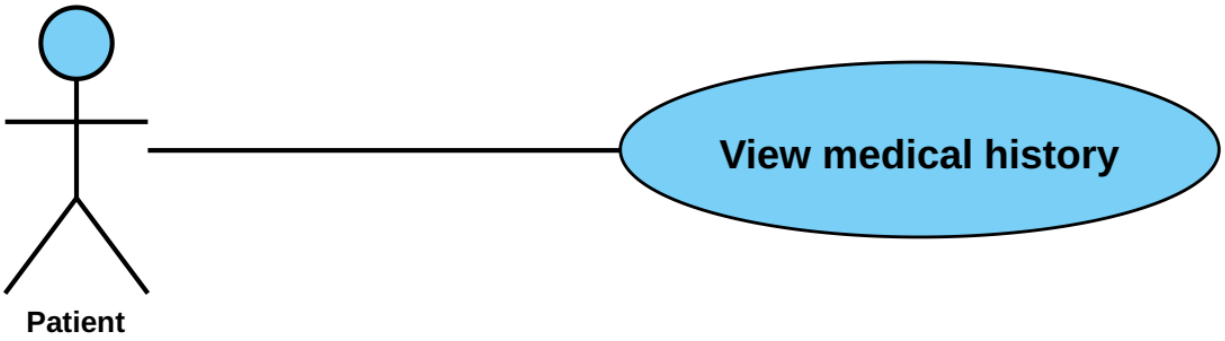
Use case name	Send medical report to doctor
Unique ID	Health care -Patient-002
Area	Application
Actor(s)	Patient
Description	Patient send medical report to doctor
Triggering Event	Patient click "send medical report to doctor" button in the application
Preconditions	- The Patient needs to login in to his account
Postconditions	- Patient has successfully sent medical report to doctor
Assumptions	- Patient have Medical Care application - A valid data

Steps Performed	**Information for Steps**
1. Open application 2. Patient log in 3. Patient click "send medical report to doctor" button 4. Patient Enters his medical report 5. Patient clicks on "Save" button 6. Validation of entered medical report by application	Step 2: E-mail, Password Step 4: medical report

Extensions (Alternative Flows)	- If user entered a non-valid medical report, a warning message should appear to him

Table 8. Send Medical Report to Doctor Use Case Description

View Medical History: patient can view his medical records



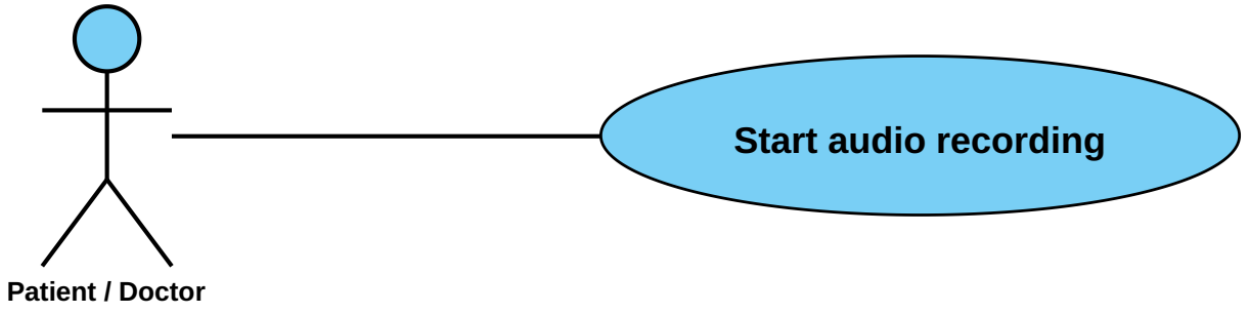
Use case name	View medical history
Unique ID	HealthCare -Patient-003
Area	Application
Actor(s)	Patient
Description	Patient views his medical history
Triggering Event	Patient click "View medical history" button in the application
Preconditions	- The Patient needs to login in to his account
Postconditions	- Patient has successfully viewed medical history
Assumptions	- Patient have Medical Care application - A valid data

Steps Performed	**Information for Steps**
1- Open application 2- Patient log in 3- Patient click "View medical history" button	Step 2: E-mail, Password

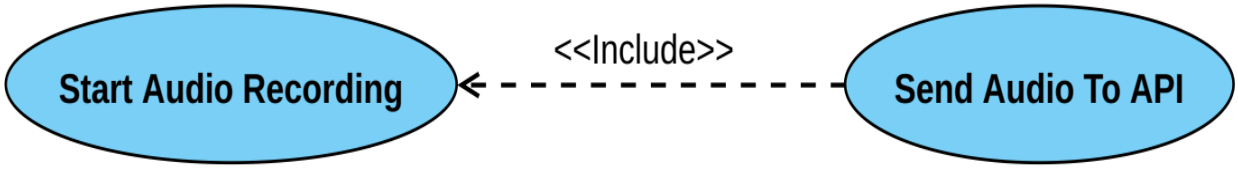
Extensions (Alternative Flows)	- If user entered a non-valid data, a warning message should appear to him

Table 9. View Medical History Use Case Description

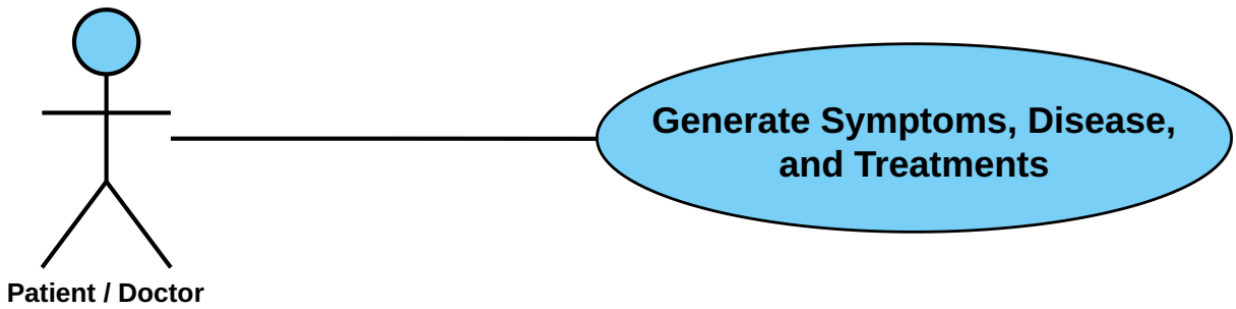
Start Audio Recording: Doctor or patient starts audio record to voice recording a patient's complaint



Send Audio To API: After complaint recorded it will be sent to model to start classification



Generate Symptoms, Disease, and Treatments: doctor or patient receives symptoms, diseases and treatments that can use to make decision for patient Illness



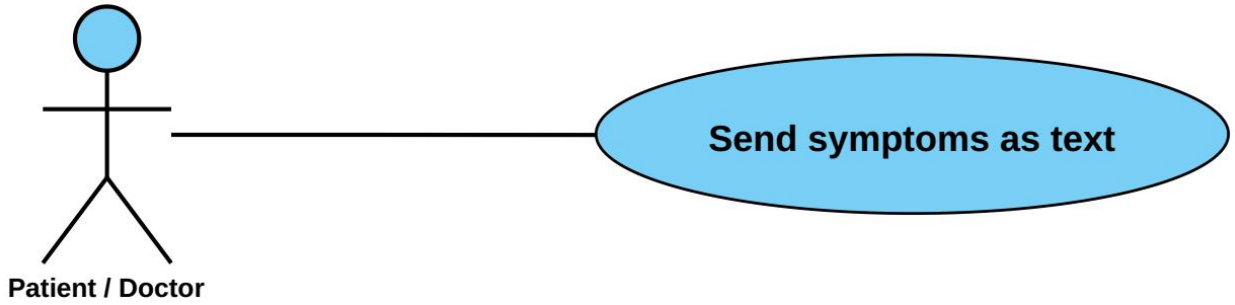
Use case name	Start audio recording
Unique ID	HealthCare-User-003
Area	Application
Actor(s)	User (Patient or Doctor)
Description	User start new diagnose
Triggering Event	User click “Start audio recording” button in the application
Preconditions	- The User needs to login in to his account
Postconditions	- Audio has successfully recorded, submitted to server and returned result
Assumptions	- User have Medical Care application - User should speak in a clear voice - User should talk about symptoms in a clear and direct way

Steps Performed	**Information for Steps**
1- Open application 2- User log in 3- User click “Start audio recording” button 4- User start talking and describes how he feels ‘symptoms’ 5- User clicks on “Finish recording” button 6- Validation of entered data by application and send data to server 7- Server responds with result 8- Doctor adds note if he wants 9- User click “Save” button	Step 2: E-mail, Password Step 4: Audio Step 8: Note Step 9: Symptoms, Disease, Treatment, Note

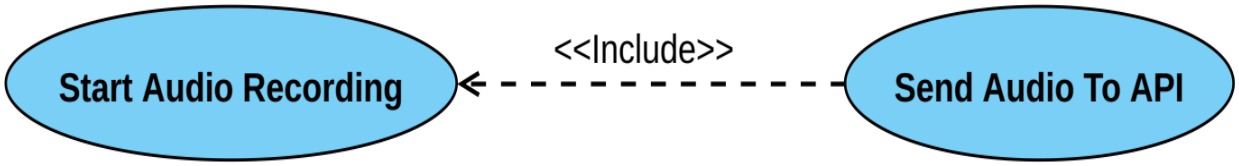
Extensions (Alternative Flows)	- If doctor entered a non-valid data in login, a warning message should appear to him

Table 10. Start Audio Recording Use Case Description

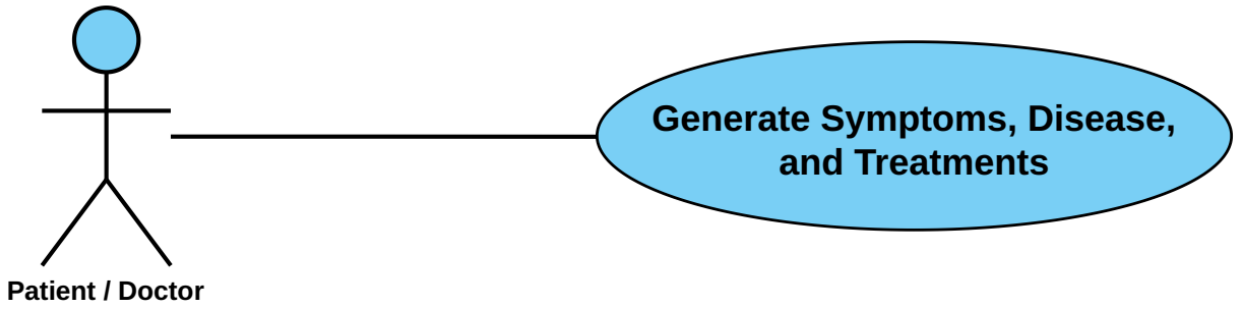
Send symptoms as text: Doctor or patient starts writing complaint



Send Audio To API: After complaint recorded it will be sent to model to start classification



Generate Symptoms, Disease, and Treatments: doctor or patient receives symptoms, diseases and treatments that can use to make decision for patient Illness.



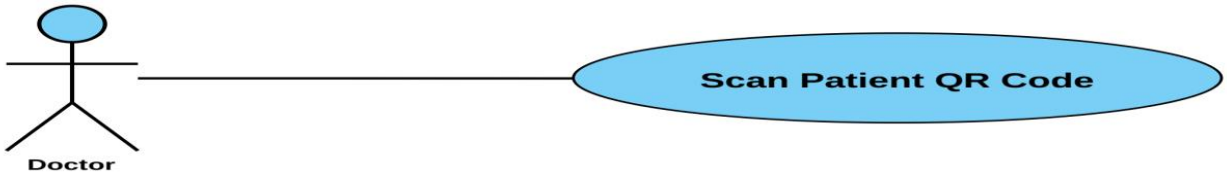
Use case name	Send symptoms as text
Unique ID	HealthCare-User-004
Area	Application
Actor(s)	User (Patient or Doctor)
Description	User start write about symptoms
Triggering Event	User click “Write symptoms as text” button in the application
Preconditions	- The User needs to login in to his account
Postconditions	- Text has successfully submitted to server and returned result
Assumptions	- User have Medical Care application - User should speak in a clear voice - User should talk about symptoms in a clear and direct way

Steps Performed	**Information for Steps**
1. Open application 2. User log in 3. User click “Write symptoms as text” button 4. User start writing and describes how he feels ‘symptoms’ 5. User clicks on “Send Symptoms” button 6. Validation of entered data by application and send data to server 7. Server responds with result 8. Doctor adds note if he wants 9. User click “Save” button	Step 2: E-mail, Password Step 4: Audio Step 8: Note Step 9: Symptoms, Disease, Treatment, Note

Extensions (Alternative Flows)	- If doctor entered a non-valid data in login, a warning message should appear to him

Table 11. Send Symptoms as Text Use Case Description

Scan Patient QR Code: Doctor scans Patient QR Code to be able view patient medical history



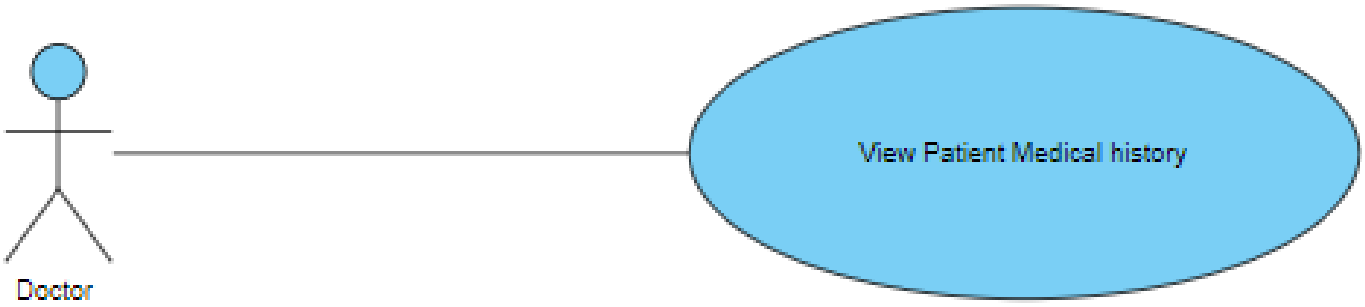
Use case name	Scan patient QR code
Unique ID	HealthCare -Doctor-001
Area	Care Bot Application, Server
Actor(s)	Doctor
Description	Get patient's old medical history
Triggering Event	Doctor click "Scan QR" button in the application
Preconditions	- Doctor needs to login in to his account - Patient needs to login in to his account - Patient must have QR code
Postconditions	- Doctor has successfully gotten patient's old medical history on his device
Assumptions	- Doctor have Care bot application - Patient should have QR code - Doctor phone have a camera

Steps Performed	**Information for Steps**
1- Open application 2- Doctor log in 3- Doctor click "Scan QR" button 4- Doctor scan QR code by his camera 5- Validation of entered data by application and send data to server 6- Server respond with result	Step 2: E-mail, Password Step 4: QR code Step 6: Old_Medical_History

Extensions (Alternative Flows)	- If doctor entered a non-valid data in login, a warning message should appear to him - If patient have non-valid QR code, then patient must generate new QR code in his account

Table 12. Scan Patient QR Code Use Case Description

View Patient Medical history: Doctor after scan QR code of patient now he can view patient medical record



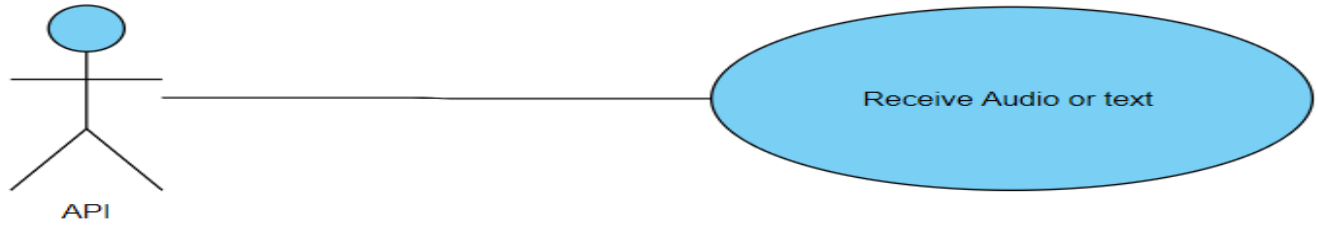
Use case name	View patient medical history	
Unique ID	HealthCare -Doctor-002	
Area	Care Bot	
Actor(s)	Doctor	
Description	Doctor views patient medical history	
Triggering Event	Doctor scan patient QR code	
Preconditions	- The Doctor needs to login in to his account	
Postconditions	- Doctor has successfully viewed patient medical history	
Assumptions	- Doctor have Care bot application - A valid data	
Steps Performed		**Information for Steps**
5- Open application 6- Doctor log in 7- Doctor scan patient QR code		Step 2: E-mail, Password
Extensions (Alternative Flows)	- If user entered a non-valid data, a warning message should appear to him	

Table 13. View Patient Medical History Use Case Description

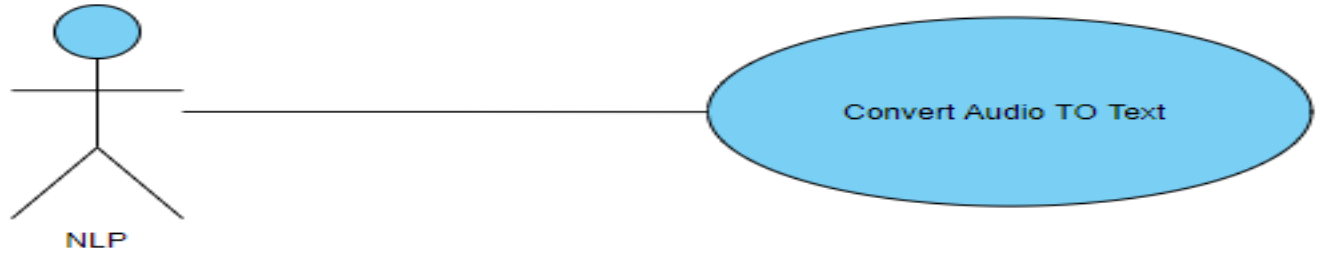
Use case name	Convert audio to text and extract symptoms	**Unique ID**	HealthCare -NLP-001
Area	Server		
Actor(s)	NLP		
Level	Blue		
Description	Convert audio received to text then extract symptoms		
Triggering Event	API send audio		
Preconditions	- A valid audio		
Postconditions	- NLP module has successfully converted audio to text and extracted symptoms		
Assumptions	- The audio has to be clear		
Steps Performed		**Information for Steps**	
1- Convert audio to text 2- Analyze text 3- Extract symptoms from text 4- Send symptoms		Step 1: Audio Step 2: Text Step 4: Symptoms	
Extensions (Alternative Flows)	- If audio wasn't clear, a warning message should appear to doctor to re-record audio - If language wasn't supported, a warning message should appear to doctor to re-record audio in supported language - If patient described an untrained disease for program, a warning message should appear to doctor that program couldn't extract symptoms - If module couldn't extract symptoms from text, a warning message should appear to doctor that program couldn't extract symptoms		

Table 14. Convert Audio to Text and Extract Symptoms Use Case Description

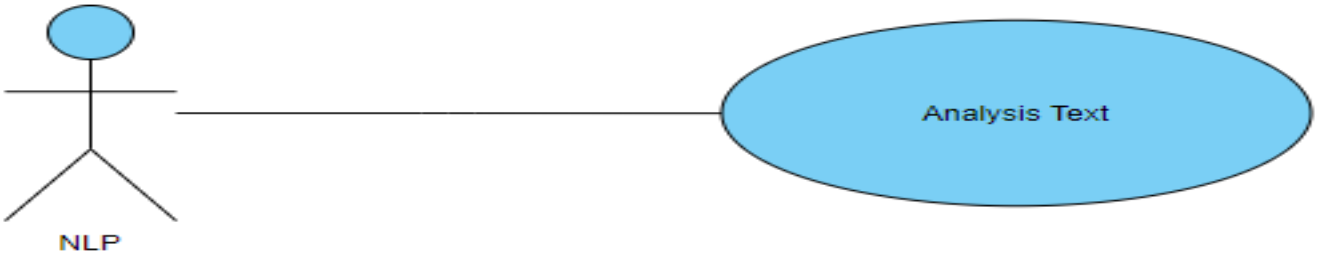
Receive Audio: server received the A patient's complaint



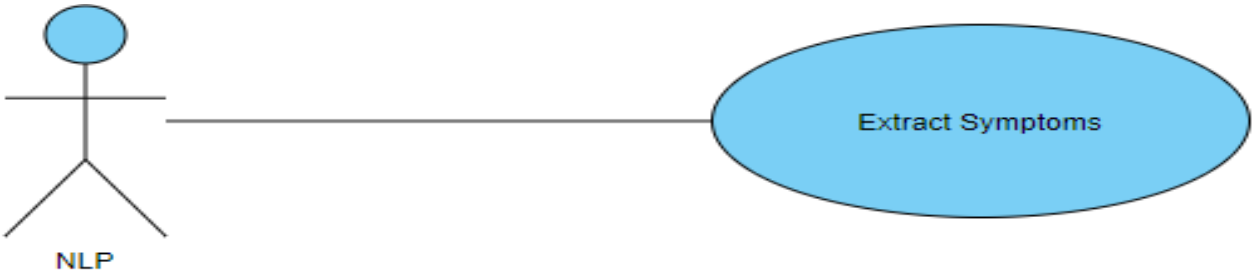
Convert Audio to Text: NLP start preprocessing text by convert Audio TO Text



Analysis Text: NLP receive text after converting and start analysis text



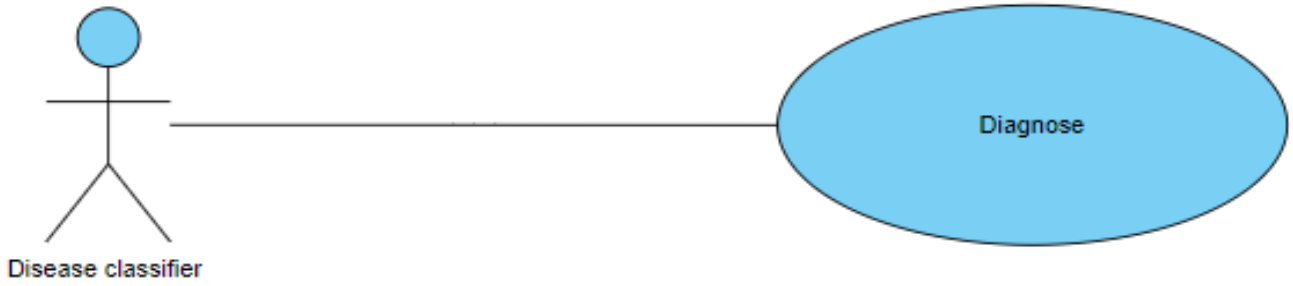
Extract Symptoms: NLP will Determines Symptoms after make Analysis to Text



Use case name	Classification of disease (Diagnose)	**Unique ID**	HealthCare - DiseaseClassifier-001
Area	Server		
Actor(s)	Disease Classifier		
Description	Getting symptoms and predict disease		
Triggering Event	NLP module sent symptoms to disease classifier		
Preconditions	- There is no preconditions		
Postconditions	- Disease classifier module has successfully predicted disease		
Assumptions	- NLP module must send symptoms		
Steps Performed		**Information for Steps**	
1- Disease classifier analyze symptoms and predict disease 2- Return result		Step 1: Symptoms Step 2: Disease	
Extensions (Alternative Flows)	- If module couldn't analyze symptoms to predicate disease, a warning message should appear to doctor that program couldn't predict disease		

Table 15. Classification of Disease (Diagnose) Use Case Description

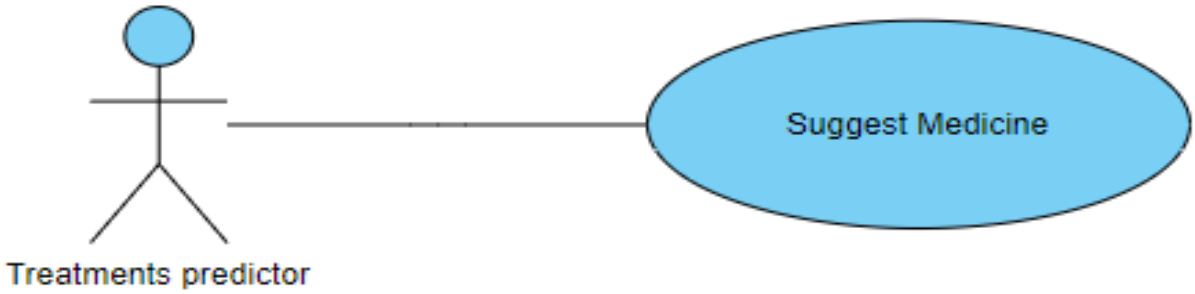
Classification Disease: after Extract Symptoms Disease Classifier will receive Symptoms and predict Diseases



Use case name	Suggest medicine	**Unique ID**	HealthCare - TreatmentPredictor-001
Area	Server		
Actor(s)	Treatment predictor		
Description	Suggest medicine		
Triggering Event	Disease classifier sent predicted disease to treatment predictor		
Preconditions	- Disease classifier sent predicted disease		
Postconditions	- Treatment predictor provided appropriate treatment		
Assumptions	- Disease classifier must send predicted disease		
Steps Performed		**Information for Steps**	
1- Treatment predictor analyze disease and search for the best medicine for it 2- Return result		Step 1: Disease Step 2: Medicines	
Extensions (Alternative Flows)	- If module couldn't analyze disease to predicate medicine, a warning message should appear to doctor that program couldn't predict appropriate medicine		

Table 16. Suggest Medicine Use Case Description

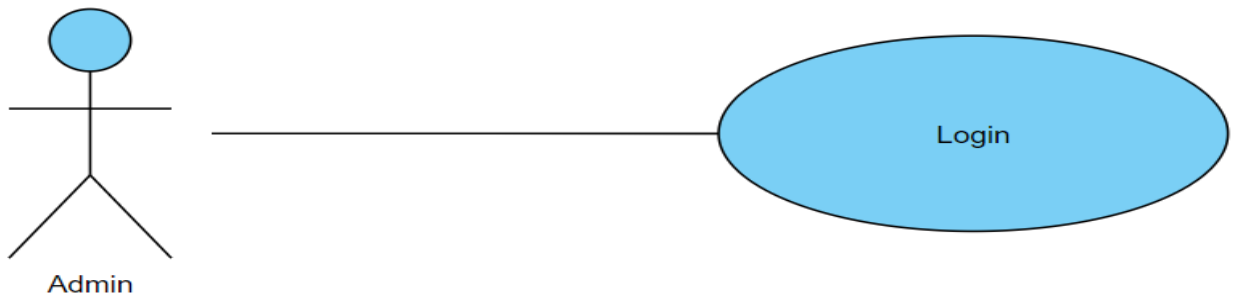
Suggest Medicine: after predict Diseases Treatments predictor will receive Diseases and suggest medicine.



Use case name	Login	**Unique ID**	HealthCare -Admin-003
Area	System Back-End		
Actor(s)	System Administrator		
Description	Admin logging to an administrative account.		
Triggering Event	Admin click “Login” button in the application		
Preconditions	- The admin needs to download application then open it. - The admin needs to have internet access The user needs to have the authorities to log in as an admin.		
Postconditions	- Admin has successfully logged in to his account		
Assumptions	- The admin has the medical-care application A valid data		
Steps Performed		**Information for Steps**	
1- Open application 2- Admin enters his data 3- Click on “Login” button 4- Validation of entered data by application		Step 2: E-mail, Password	
Extensions (Alternative Flows)	- If user entered a non-valid data, a warning message should appear to him.		

Table 17. Dashboard Login Use Case Description

Login: admin login in dashboard to management

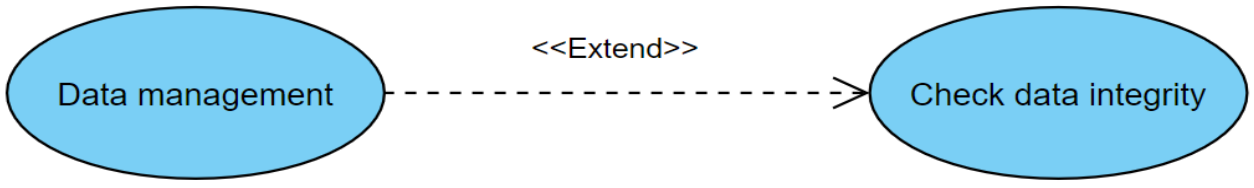


Use case name	Check Data-Integrity (Data management)	**Unique ID**	HealthCare -Admin-002
Area	System Back-End		
Actor(s)	Administrator		
Description	The administrator verifies the integrity of data.		
Triggering Event	Logging into the database and check the records.		
Preconditions	The user must be verified as an admin and at least 1 new record available.		
Postconditions	- Data integrity confirmed.		
Assumptions	- A functional database to retrieve data and check integrity techniques.		

Steps Performed	**Information for Steps**
1- Open application 2- Log in as an admin 3- Check new records. 4- Apply integrity investigations. 5- Confirm data.	Records.
Extensions (Alternative Flows)	- No data to check.

Table 18. Check Data-Integrity (Data management) Use Case Description

Check data integrity: admin check data that come from users

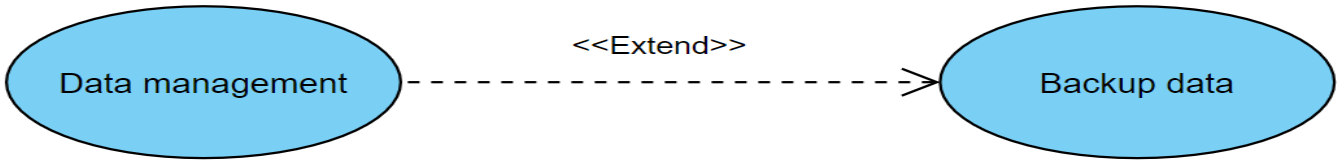


Use case name	Backup and Restore Data	**Unique ID**	HealthCare -Admin-003
Area	System Back-End		
Actor(s)	Administrator		
Description	The administrator perform back-up or restore operation.		
Triggering Event	Initiate back-up or Restoration process.		
Preconditions	There must be a data to back-up or a backed-up data to restore.		
Postconditions	- Data has been successfully backed-up or restored.		
Assumptions	Available database section to perform back-up or restore and an authorized user.		
Steps Performed		**Information for Steps**	
5- Open application 6- Log in as an admin 7- Open database. 8- Back-up a piece of data (if desired). 9- Restore a certain piece of data (if desired). 10-Confirm the process and save changes.		Database.	
Extensions (Alternative Flows)	- No data to back-up or restore.		

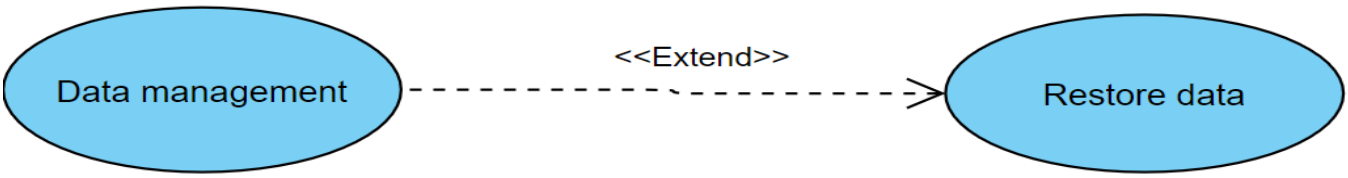
Table 19. Backup and Restore Data Use Case Description

Data management: admin make management to data that come from all users

Backup data: admin make back up data



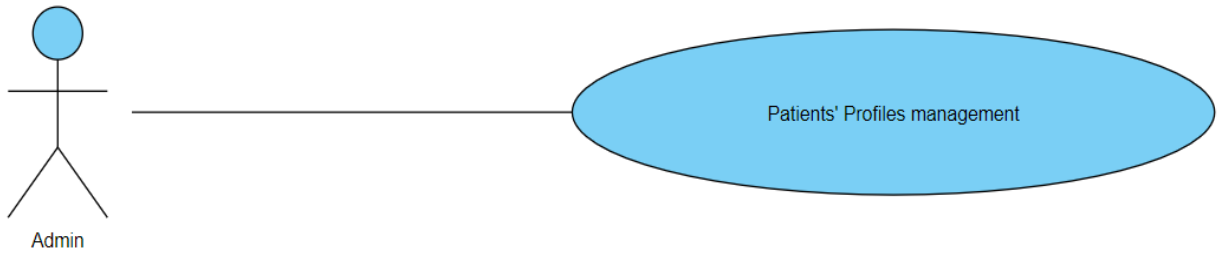
Restore data; admin make restore data



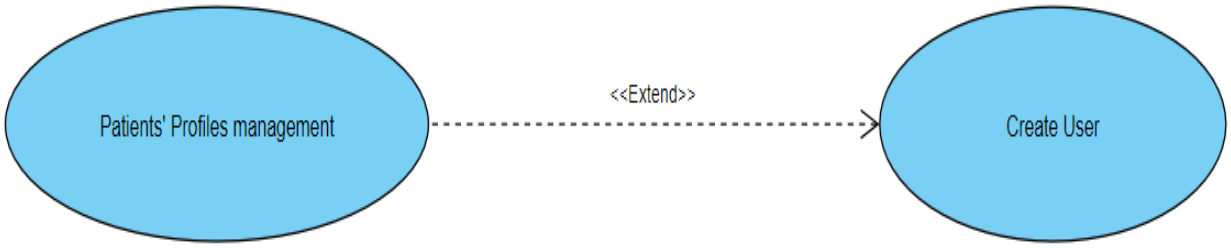
Use case name	(Patients' Profiles management) Create user	**Unique ID**	HealthCare - Admin-004
Area	Dashboard		
Actor(s)	admin		
Description	The admin manages patient's profiles and create it		
Triggering Event	Allow for add patient's profiles		
Preconditions	- The admin login		
Postconditions	- patient's profiles is created already		
Assumptions	- admin must make login		
Steps Performed		**Information for Steps**	
1- admin get of data user 2- admin can create new user		Step 1: data user	
Extensions (Alternative Flows)	- If Admin don't make login, a warning message should appear to admin that program couldn't create new user.		

Table 20. (Patients' Profiles management) Create user Use Case Description

Patients' Profiles: management: admin management to patient profiles



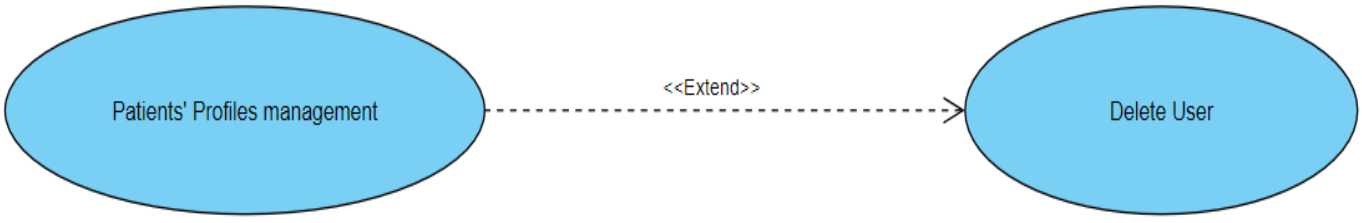
Create User: admin can create new patient account



Use case name	(Patients' Profiles management) Delete user	**Unique ID**	HealthCare - Admin-005
Area	Dashboard		
Actor(s)	admin		
Description	The admin manages patient's profiles and delete it		
Triggering Event	Allow for delete patient's profiles		
Preconditions	- The admin login		
Postconditions	- patient's profiles are deleted already		
Assumptions	- admin must make login		
Steps Performed		**Information for Steps**	
1- admin get of data user 2- admin can delete it		Step 1: data user	
Extensions (Alternative Flows)	- If Admin don't make login, a warning message should appear to admin that program couldn't delete user.		

Table 21. (Patients' Profiles management) Delete user Use Case Description

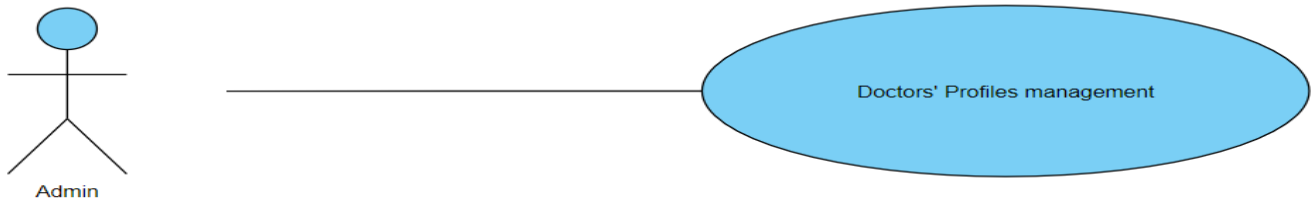
Delete User: admin can delete patient account



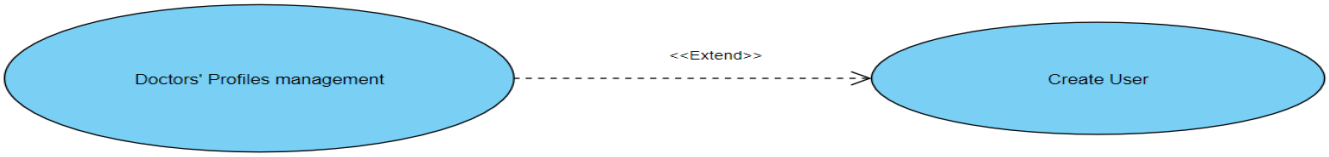
Use case name	(Doctors' Profiles management) Create user	**Unique ID**	HealthCare-Admin-006
Area	Dashboard		
Actor(s)	admin		
Level	Blue		
Description	The admin manages doctor's profiles and create it		
Triggering Event	Allow for add doctor's profiles		
Preconditions	- The admin login		
Postconditions	- Doctor's profiles are created already		
Assumptions	- admin must be login		
Steps Performed		**Information for Steps**	
1- admin get of data user 2- admin can create new user		Step 1: data user	
Extensions (Alternative Flows)	- If Admin don't make login, a warning message should appear to admin that program couldn't create new user.		

Table 22. (Doctors' Profiles management) Create user Use Case Description

Doctors' Profiles: management: admin management to doctors' profiles



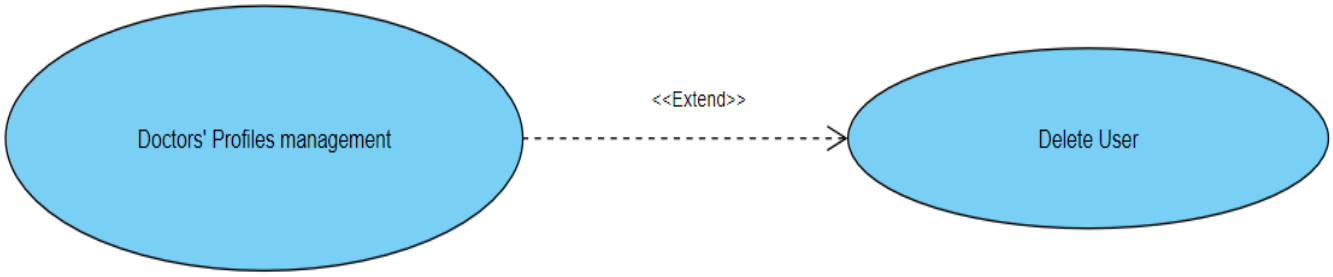
Create User: admin can create new doctor account



Use case name	(Doctors' Profiles management) Delete user	**Unique ID**	HealthCare-Admin-007
Area	Dashboard		
Actor(s)	admin		
Level	Blue		
Description	The admin manages doctor’s profiles and delete it		
Triggering Event	Allow for delete doctor’s profiles		
Preconditions	- The admin login		
Postconditions	- doctor’s profiles are deleted already		
Assumptions	- admin must make login		
Steps Performed		**Information for Steps**	
1- admin get of data user 2- admin can delete it		Step 1: data user	
Extensions (Alternative Flows)	- If Admin don’t make login, a warning message should appear to admin that program couldn’t delete user.		

Table 23. (Doctors' Profiles management) Delete user Use Case Description

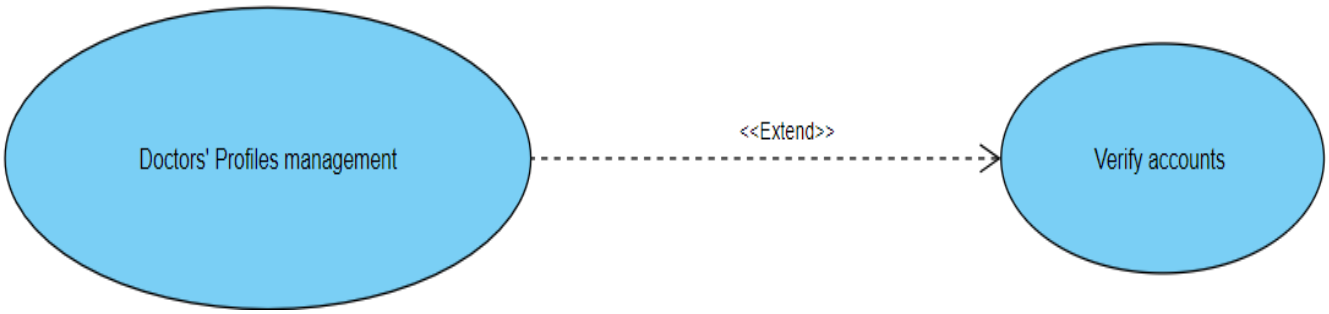
Delete User: admin can delete doctor account



Use case name	(Doctors' Profiles management) verify account	**Unique ID**	HealthCare-admin-008
Area	Dashboard		
Actor(s)	admin		
Level	Blue		
Description	The admin manages doctor's profiles and verify account it		
Triggering Event	Allow for verify account doctor's profiles		
Preconditions	The admin login		
Postconditions	Doctor's profiles are created already and it is correct		
Assumptions	admin must be login		
Steps Performed		**Information for Steps**	
1- admin get of account user 2- admin can verify it		Step 1: account user	
Extensions (Alternative Flows)	- If Admin don't make login, a warning message should appear to admin that program couldn't make verify account.		

Table 24. (Doctors' Profiles management) verify account Use Case Description

Verify accounts admin accept and verify new doctor account:

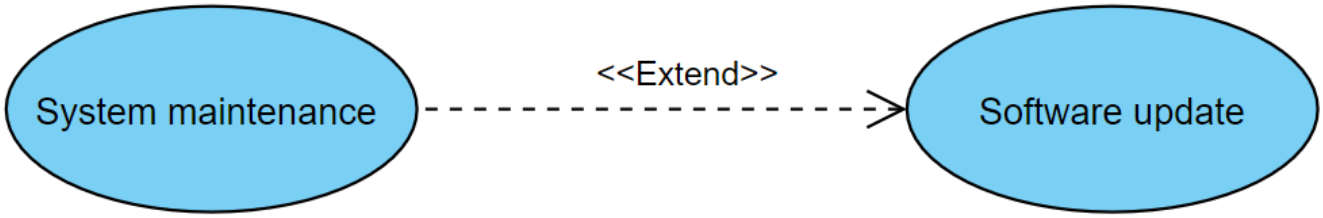


Use case name	System maintenance	**Unique ID**	HealthCare-admin-008
Area	Health Care		
Actor(s)	Admin		
Description	Admin update system and manage server		
Triggering Event	Admin open dashboard on web app or C-panel on server		
Preconditions	- The admin needs to have internet access - The admin needs to has user name and password then open it - The admin needs to have account		
Postconditions	- admin has successfully logged in to dashboard - admin update and manage system		
Assumptions	- admin have user name and password - A valid data		
Steps Performed		**Information for Steps**	
1- Open login form of dashboard 2- Admin enters his data 3- Validation of entered data by system 4- Start maintenance, update and manage system		Step 2: E-mail, Password	
Extensions (Alternative Flows)	- If admin entered a non-valid data, a warning message should appear to him		

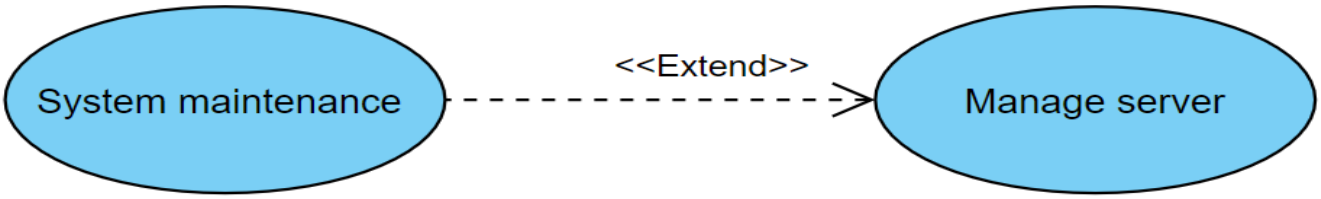
Table 25. System Maintenance Use Case Description

System maintenance: this use case includes all system maintenance processing

Software update: system maintenance processing includes software update process



Manage server: system maintenance processing includes Manage server



3.4 Analysis Class

3.4.1 Context Diagram

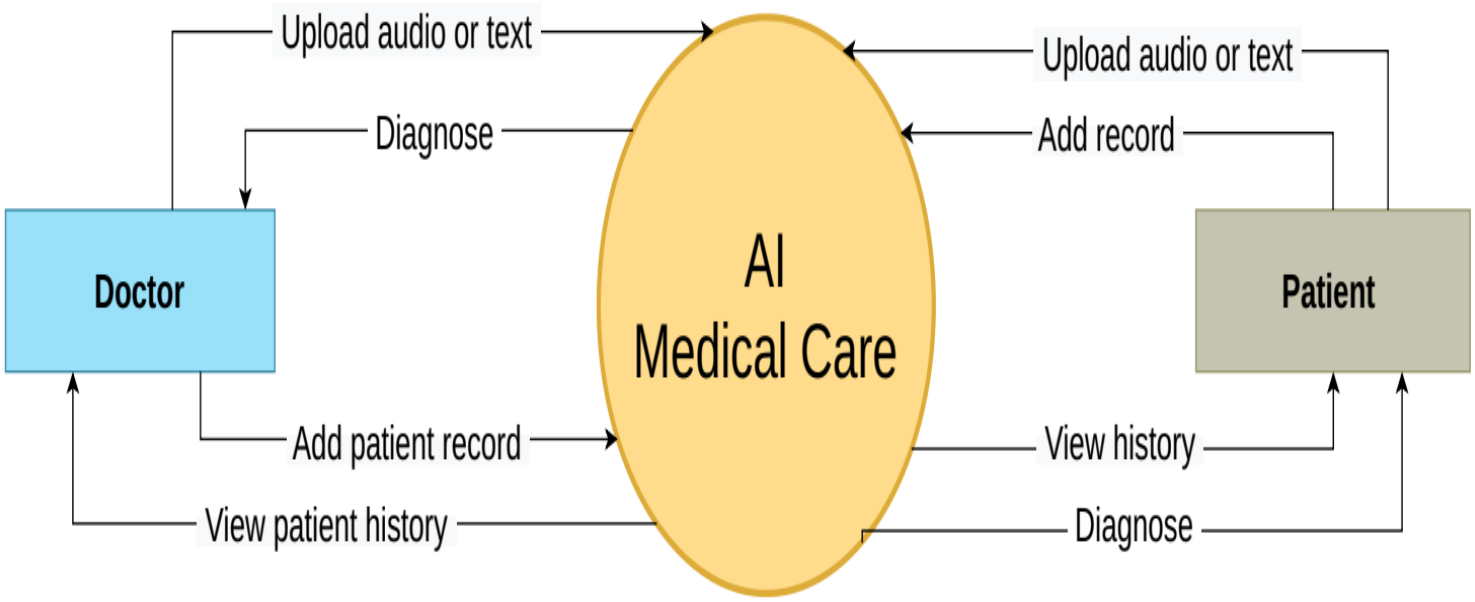


Figure 5. Context diagram

3.4.2 State Diagram

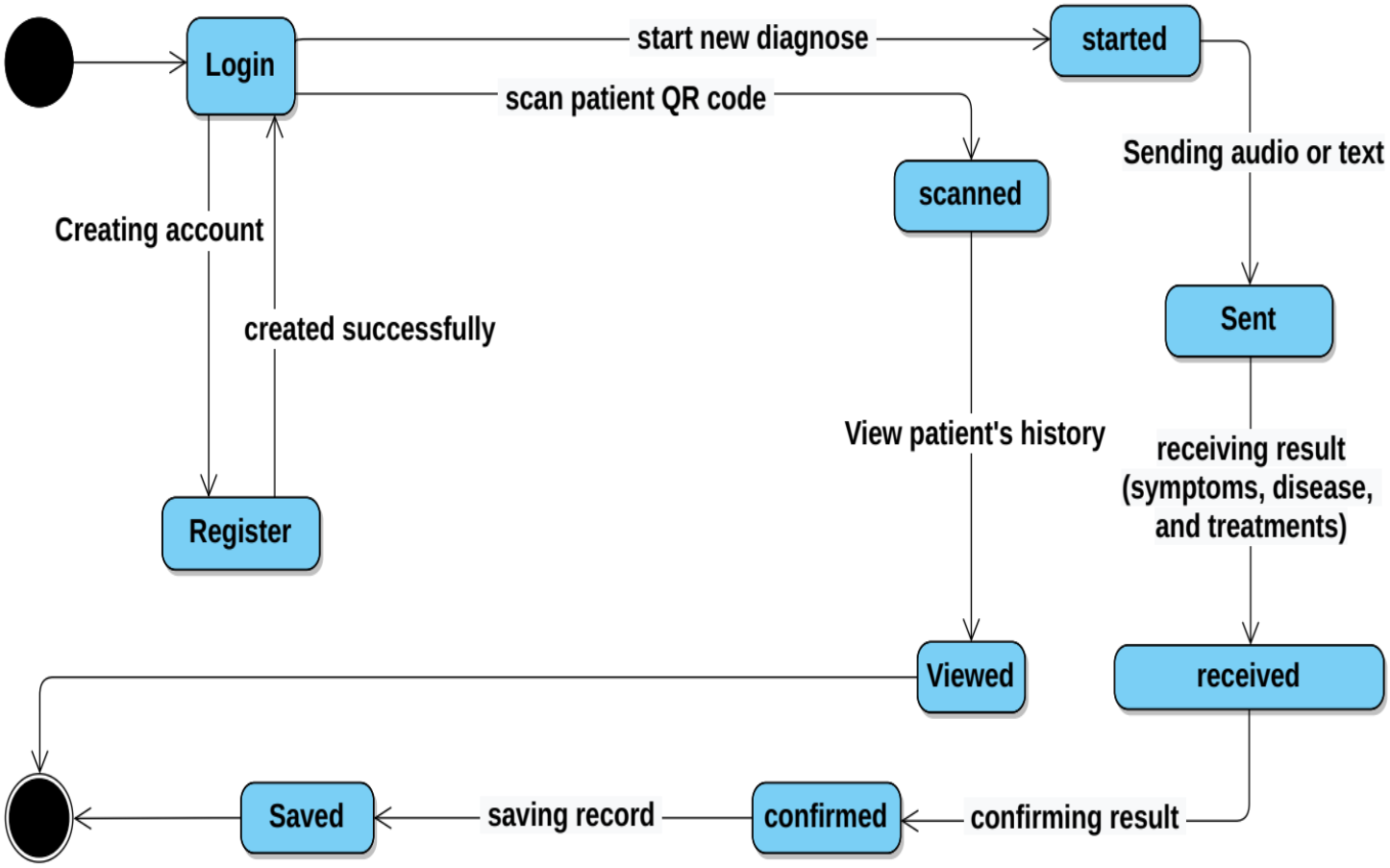


Figure 6. Doctor state diagram

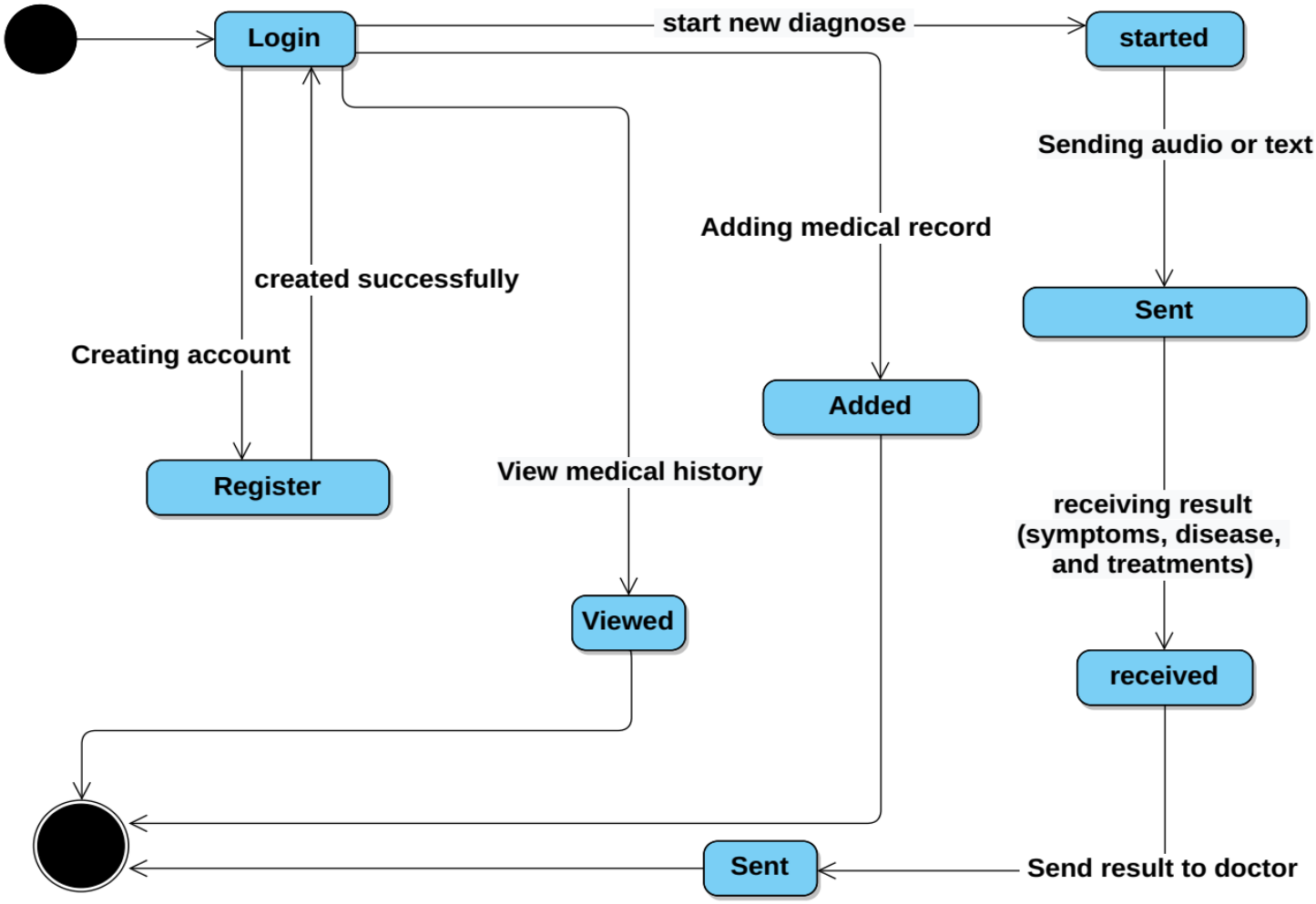


Figure 7. Patient state diagram

3.5 Interaction Diagram (Sequence Diagram)

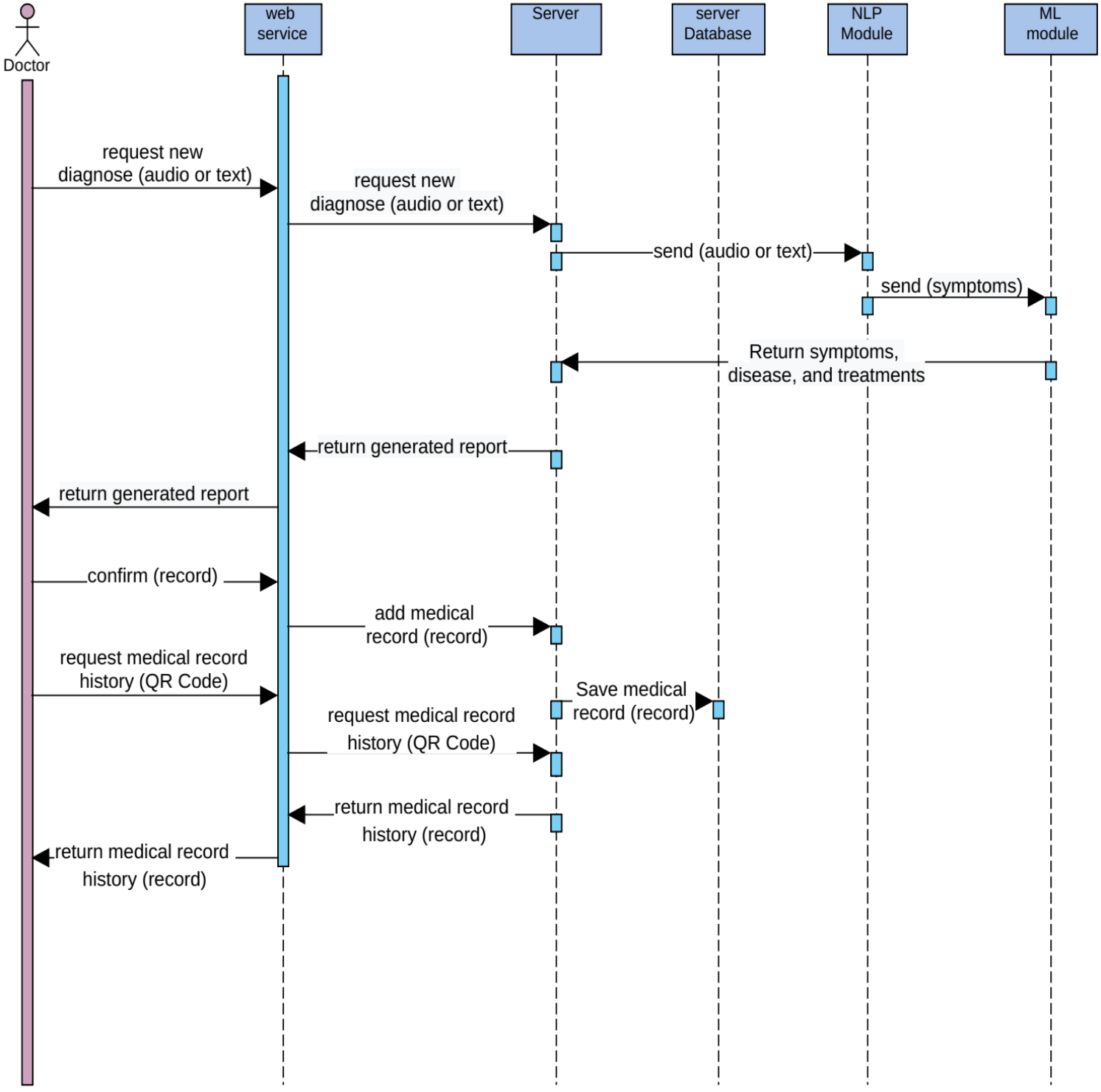


Figure 8. Doctor Sequence diagram

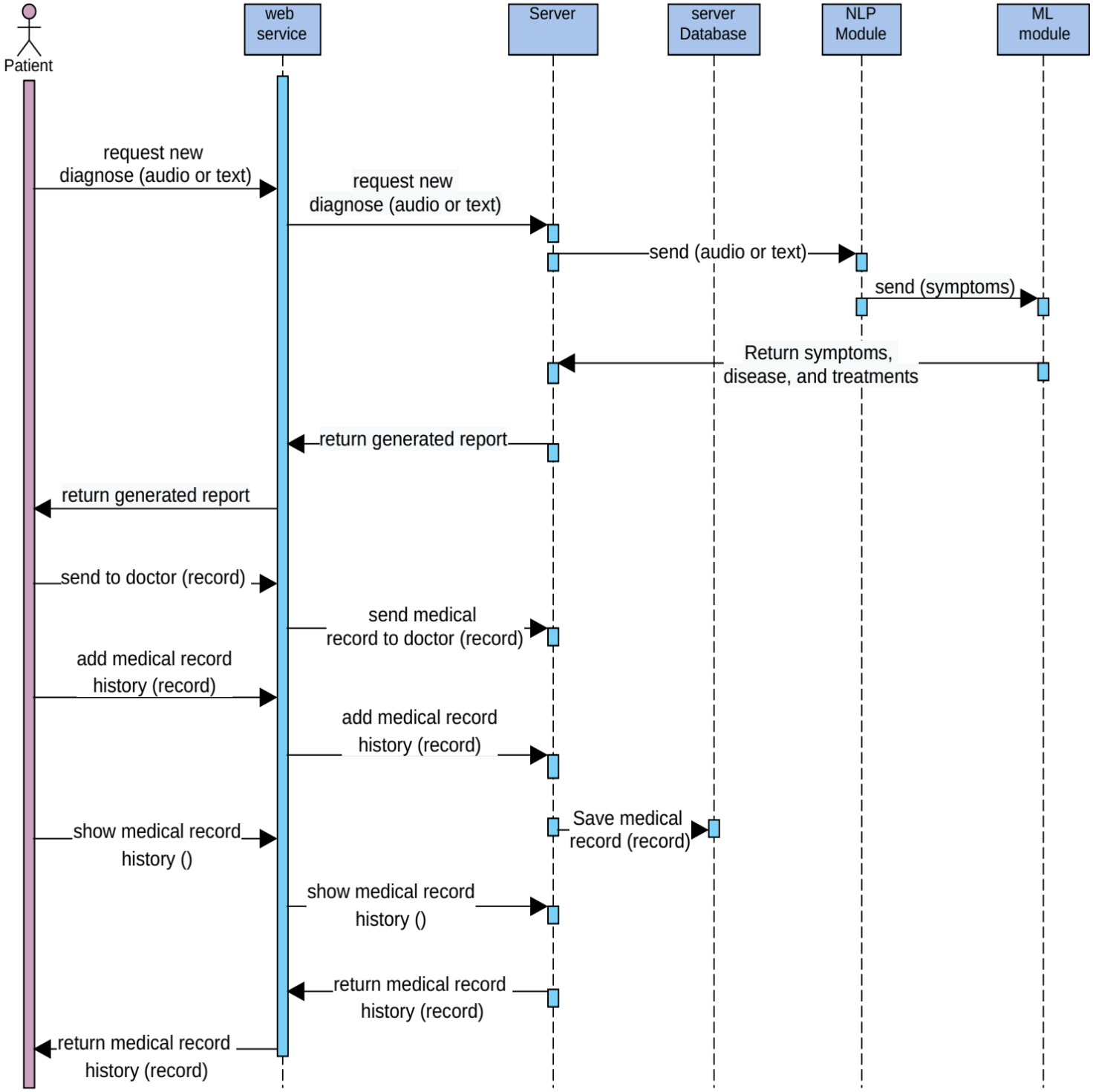


Figure 9. Patient Sequence diagram

3.6 Design Class

3.6.1 Class Diagram

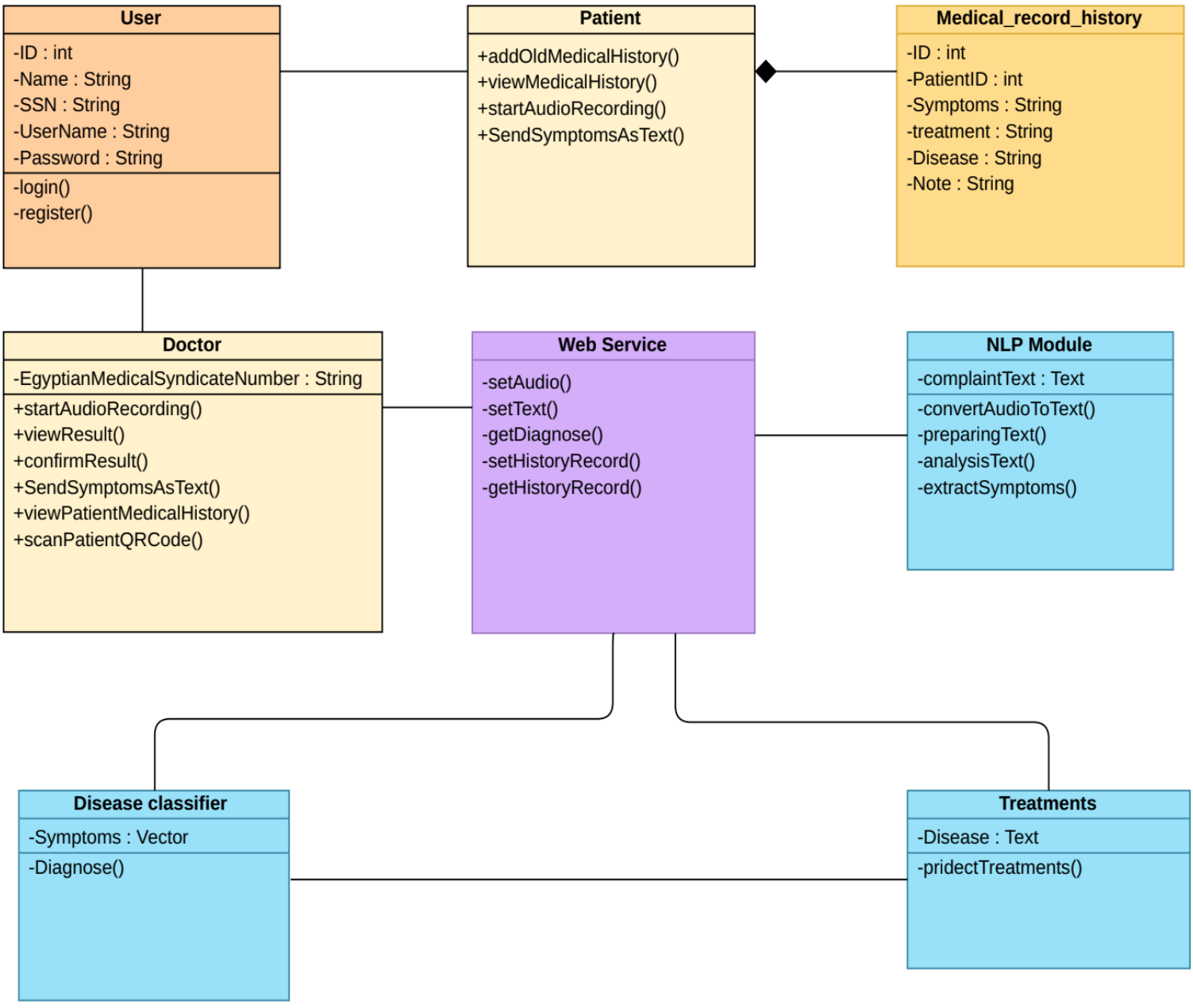


Figure 10. Class diagram

3.6.2 Domain Diagram

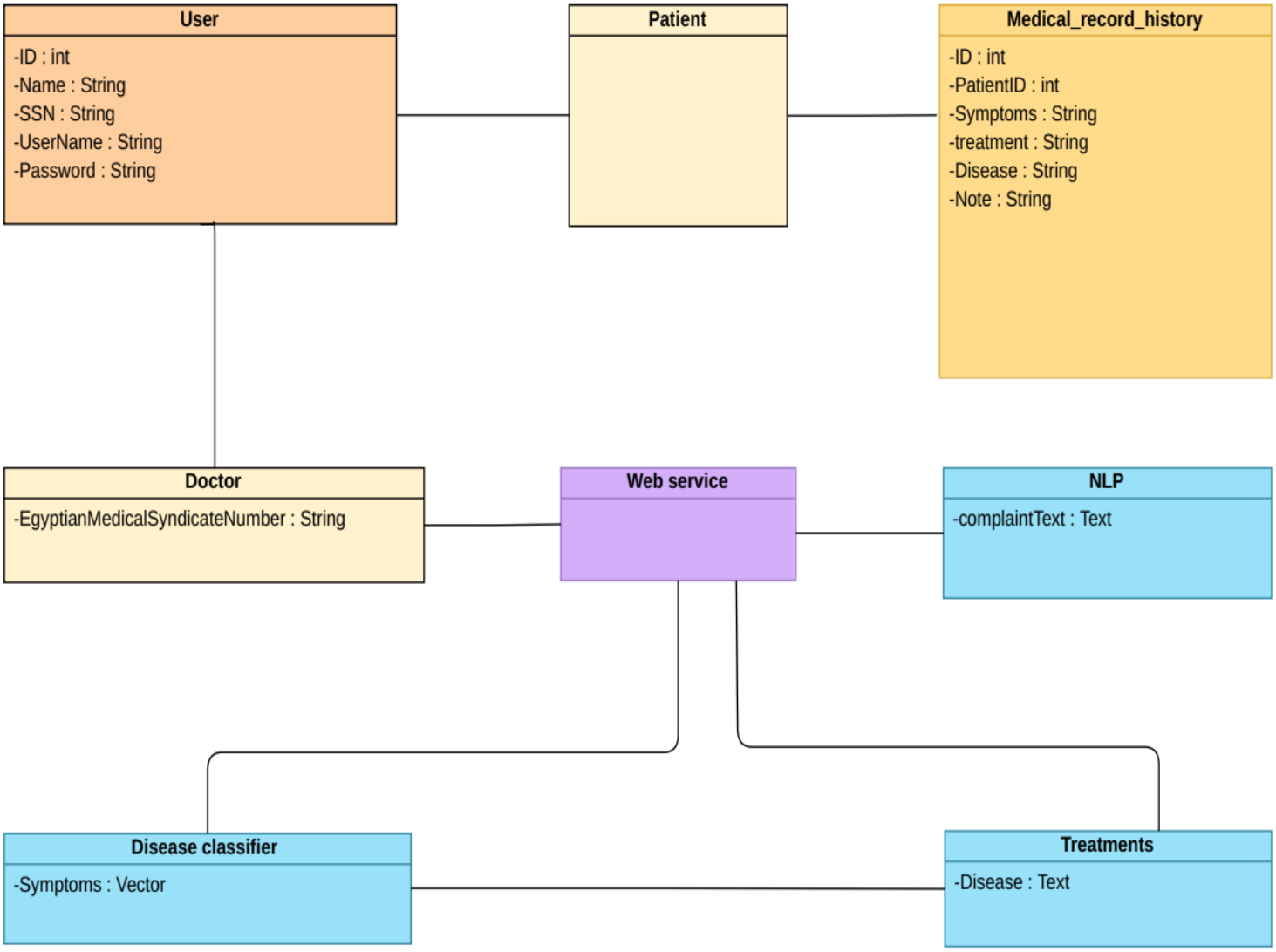


Figure 11. Domain diagram

3.7 ER Diagram

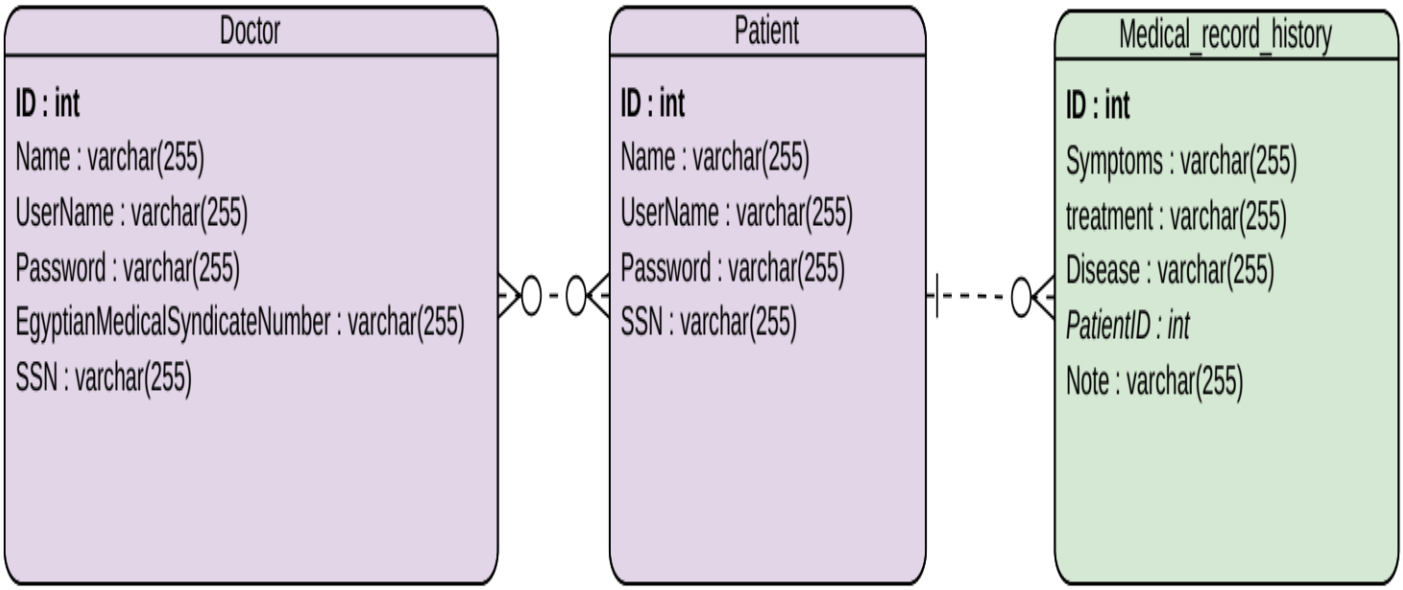


Figure 12. Entity relationship diagram

3.8 Database Schema

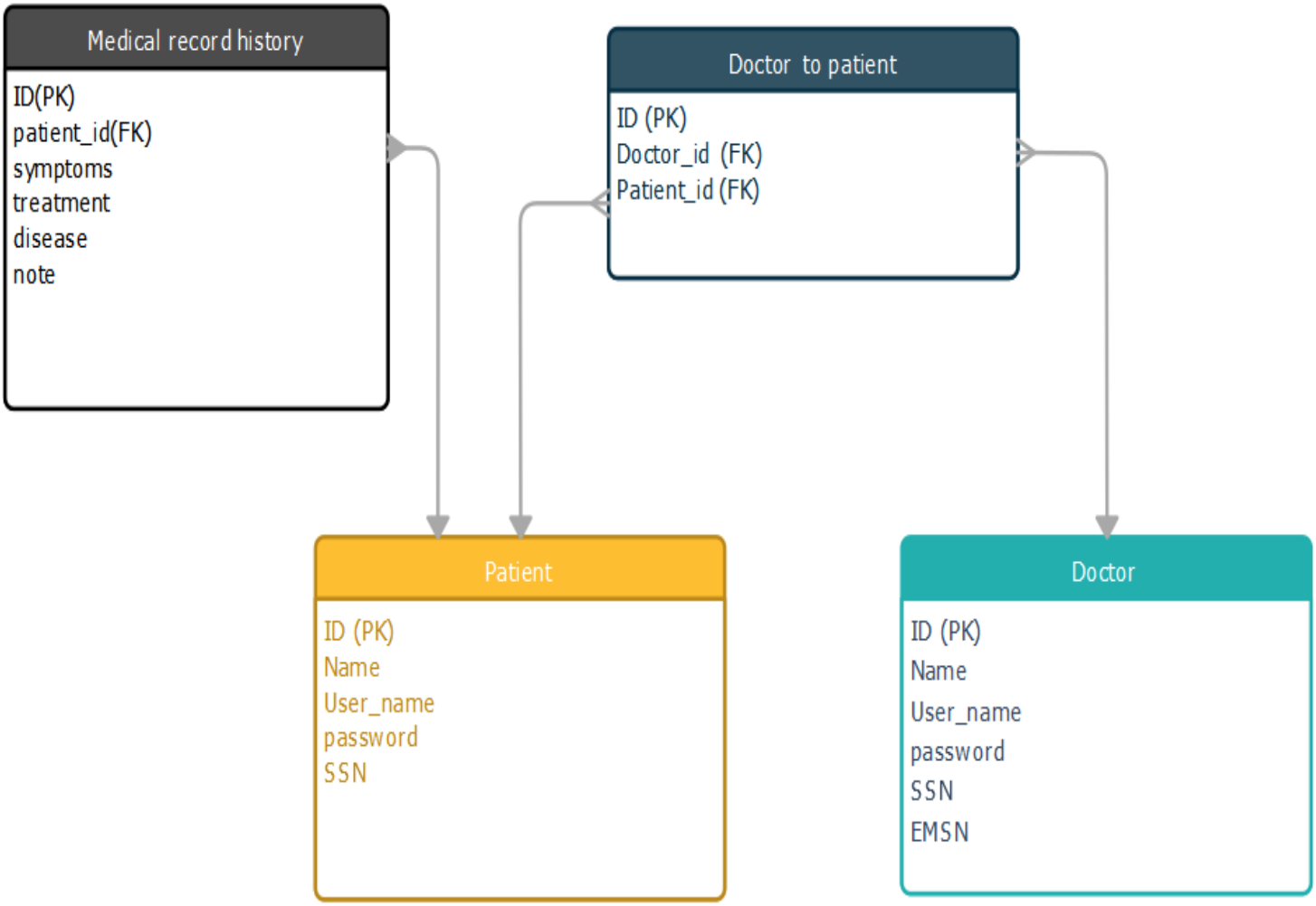


Figure 13. Database Schema

3.9 Design Mockup

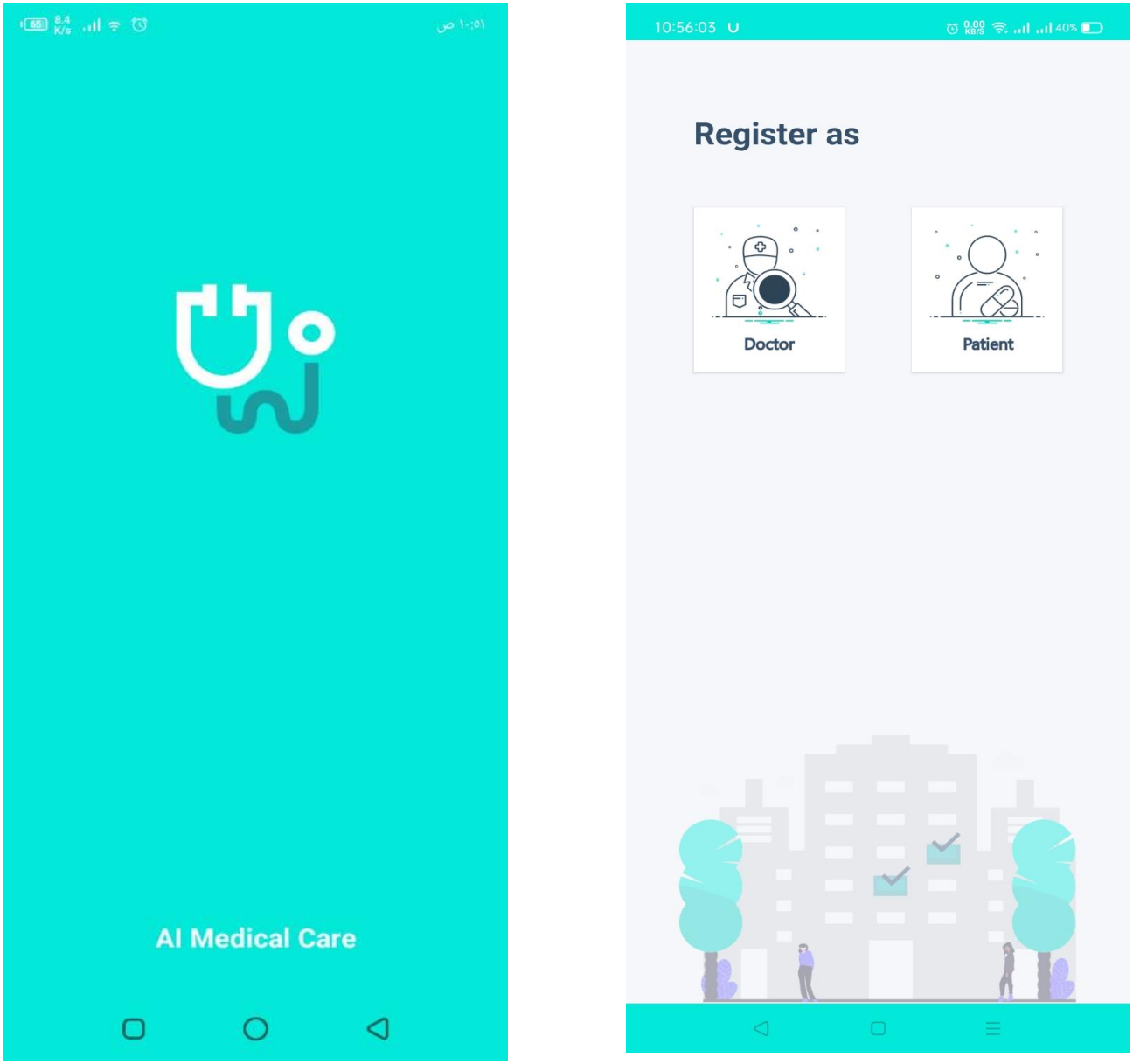
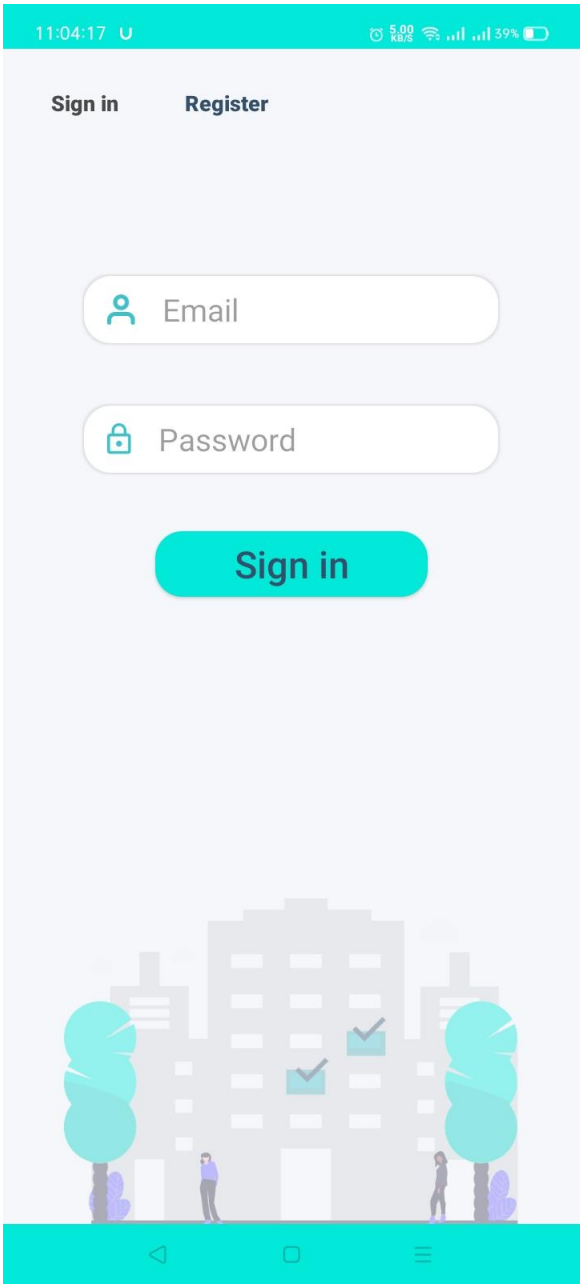
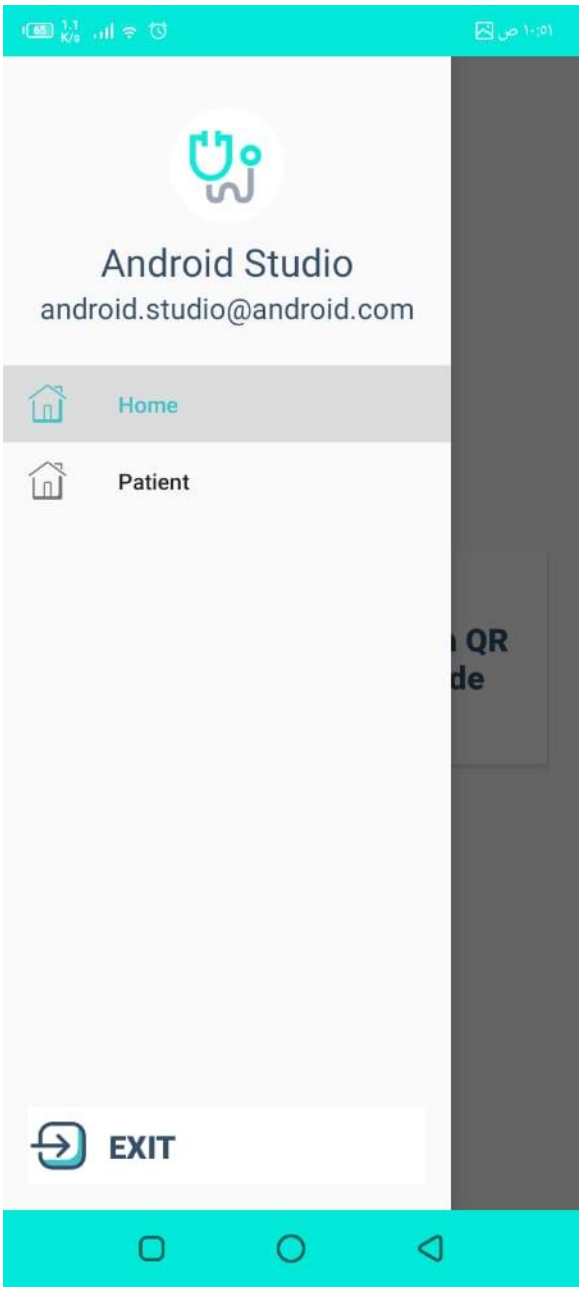
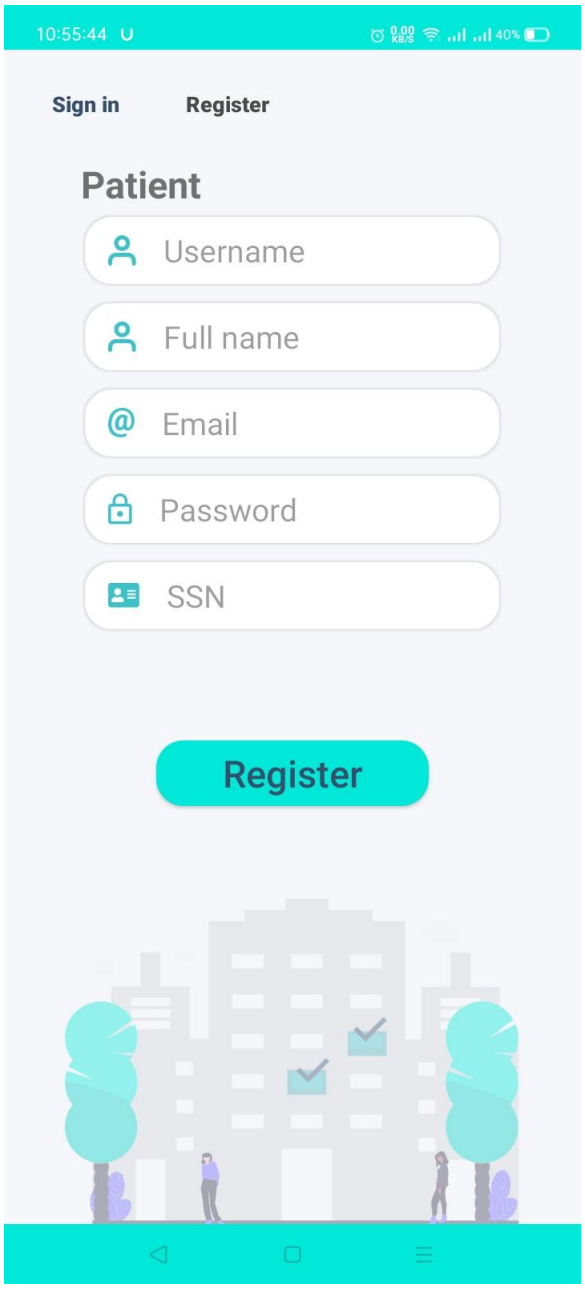
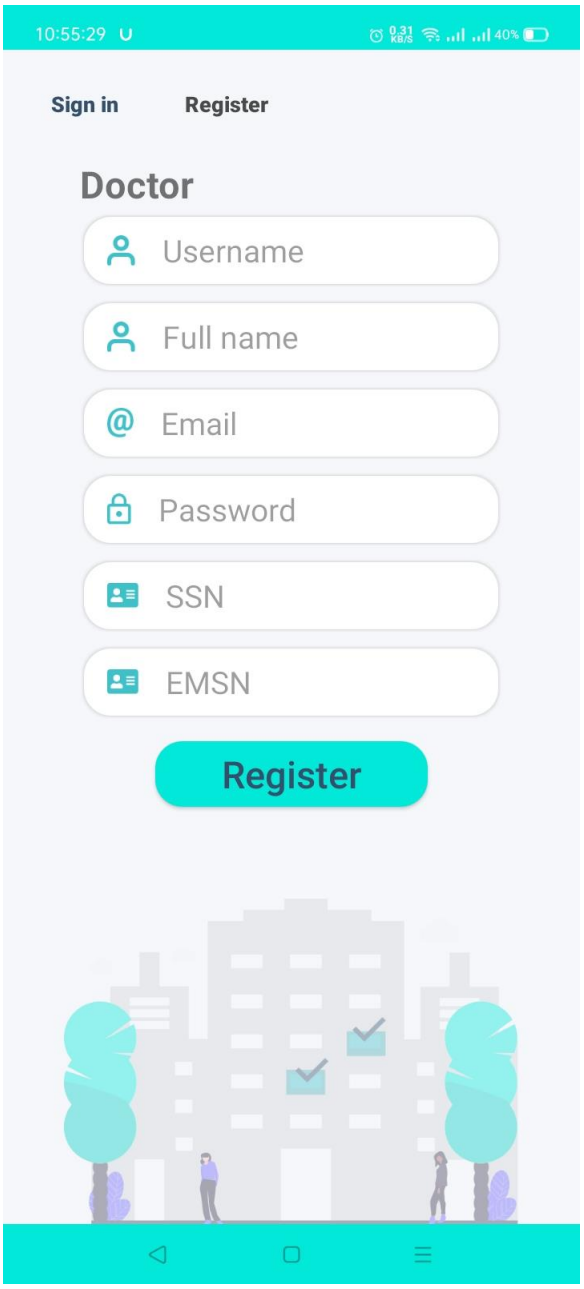


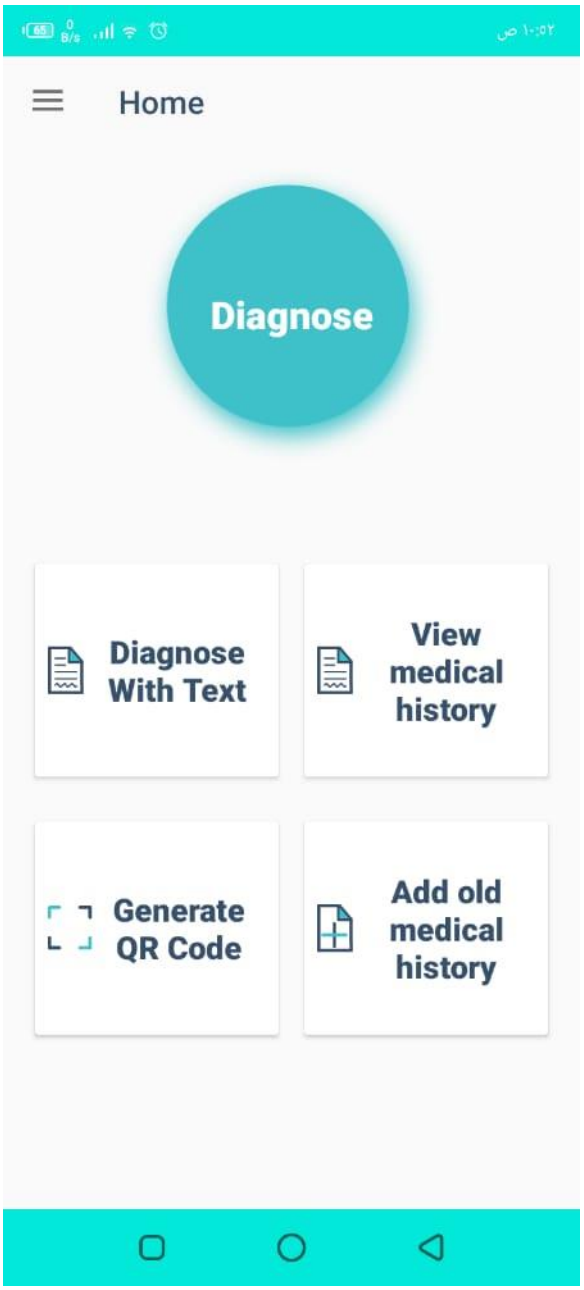
Figure 14. Design Mockup

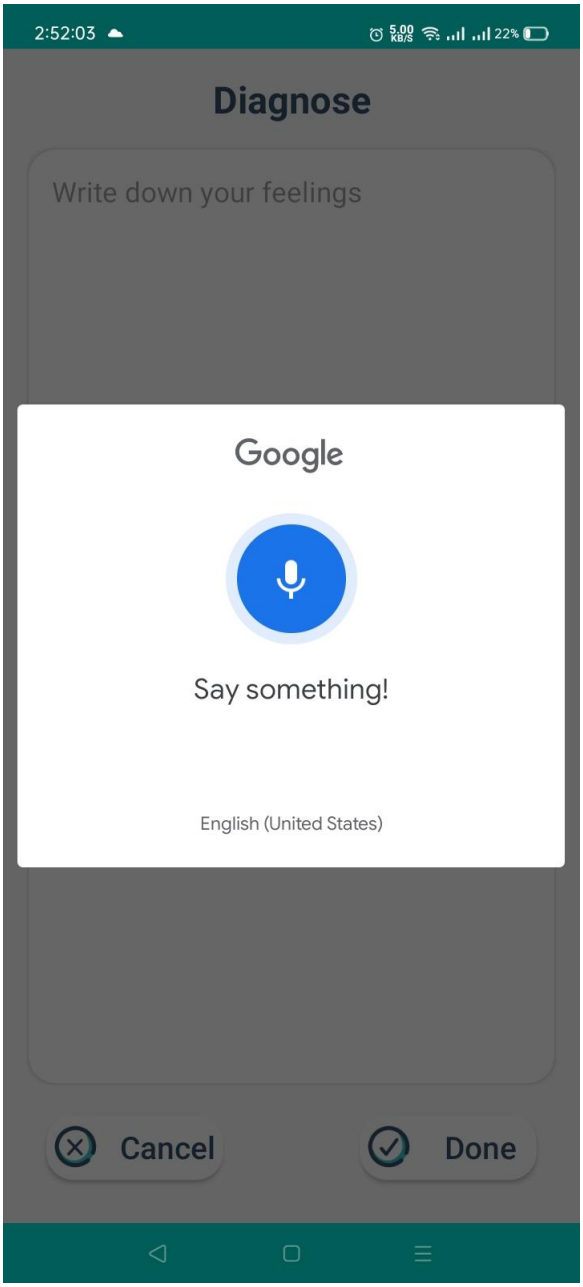

11:04:17
Sign in
Register
Email
Password
Sign in

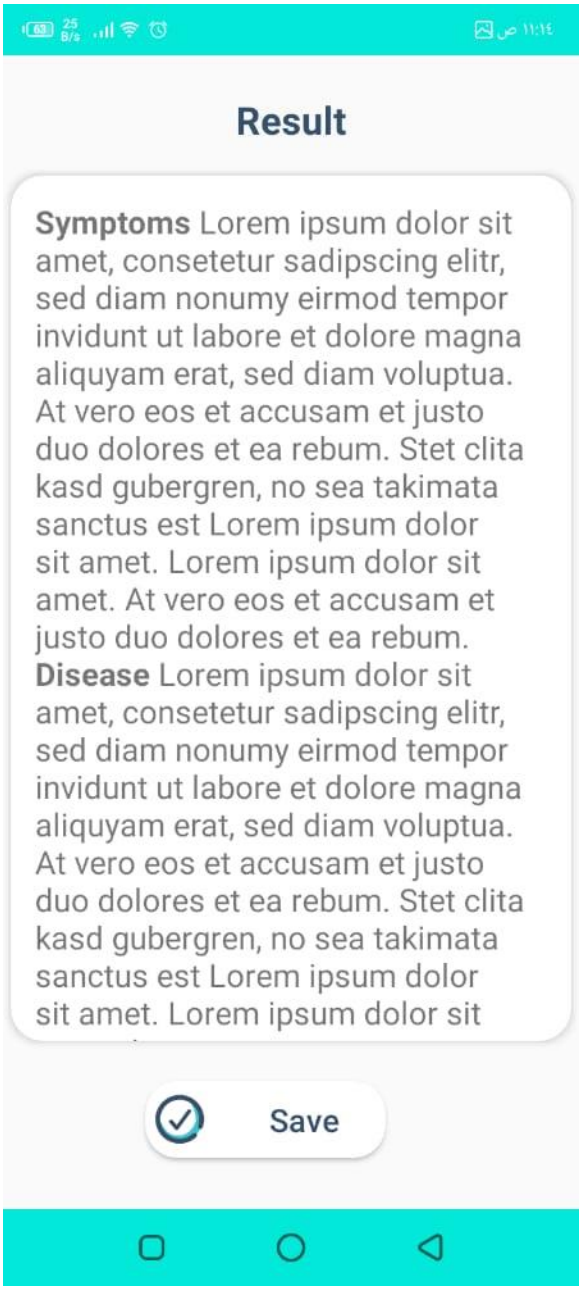

Android Studio
android.studio@android.com
Home
Patient
EXIT


10:55:44
Sign in
Register
Patient
Username
Full name
Email
Password
SSN
Register

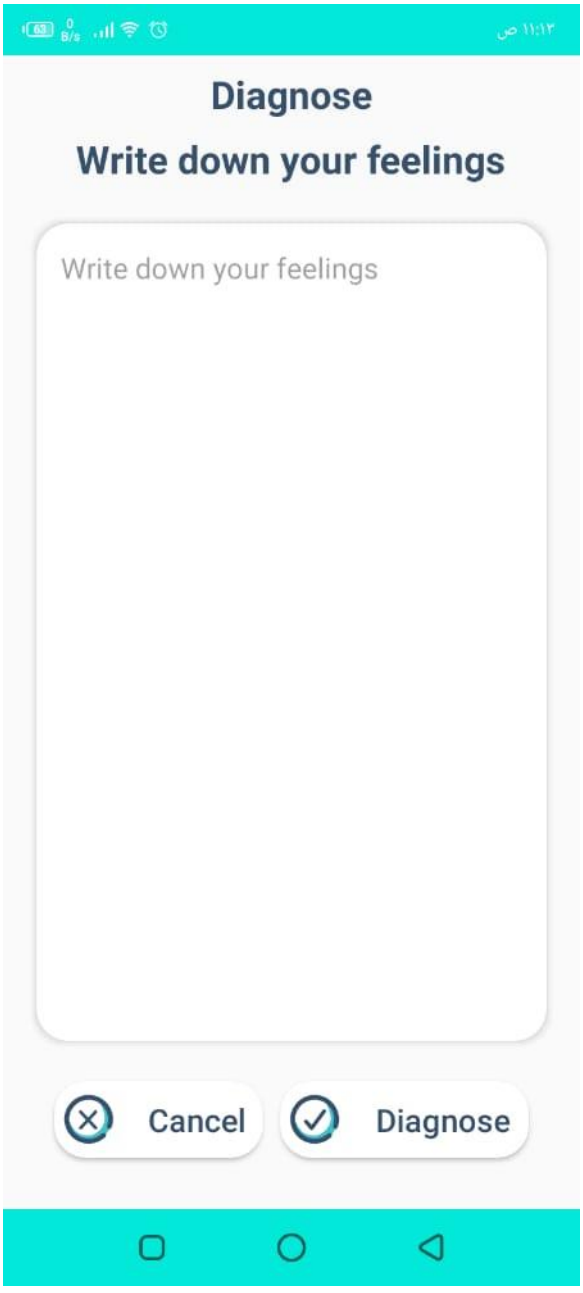

10:55:29
Sign in
Register
Doctor
Username
Full name
Email
Password
SSN
EMSN
Register


Home
Diagnose
Diagnose With Text
View medical history
Generate QR Code
Add old medical history

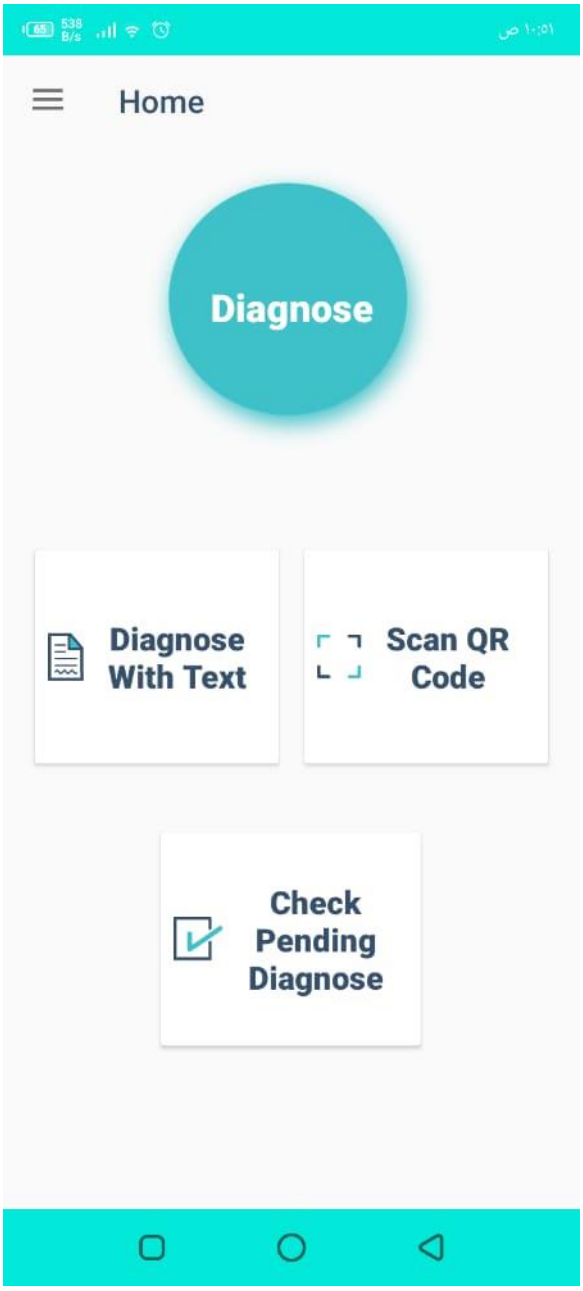

2:52:03
Diagnose
Write down your feelings
Google
Say something!
English (United States)
Cancel
Done

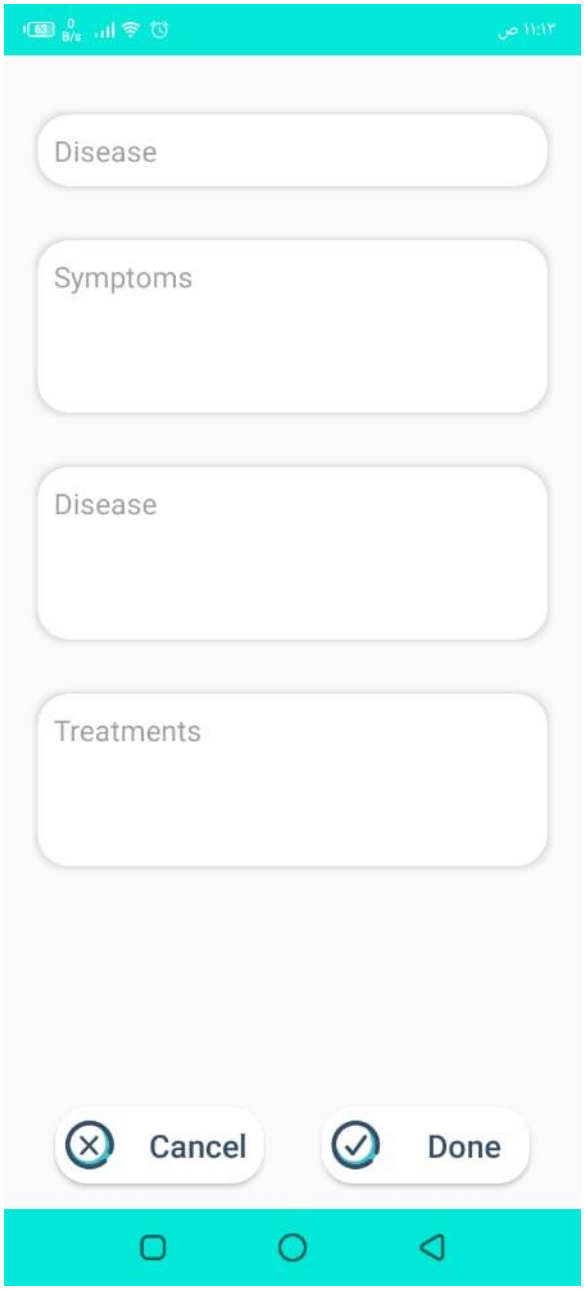

Result
Symptoms Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet. At vero eos et accusam et justo duo dolores et ea rebum.
Disease Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit
Save


Cancel


Diagnose
Write down your feelings
Write down your feelings
Cancel
Diagnose


Medical History
Commodo vestibulum ultricies
2020-07-13
Enim quis lorem
2020-06-18
Pharetra consequat
2020-02-23
Commodo vestibulum ultricies
2020-07-13
Enim quis lorem
2020-06-18
Pharetra consequat
2020-02-23
Commodo vestibulum ultricies
2020-07-13
Enim quis lorem
2020-06-18


Home
Diagnose
Diagnose With Text
Scan QR Code
Check Pending Diagnose


Disease
Symptoms
Disease
Treatments
Cancel
Done

Chapter 4

Experimental Results and Discussion

4.1 Expected Deployment

Global Overview

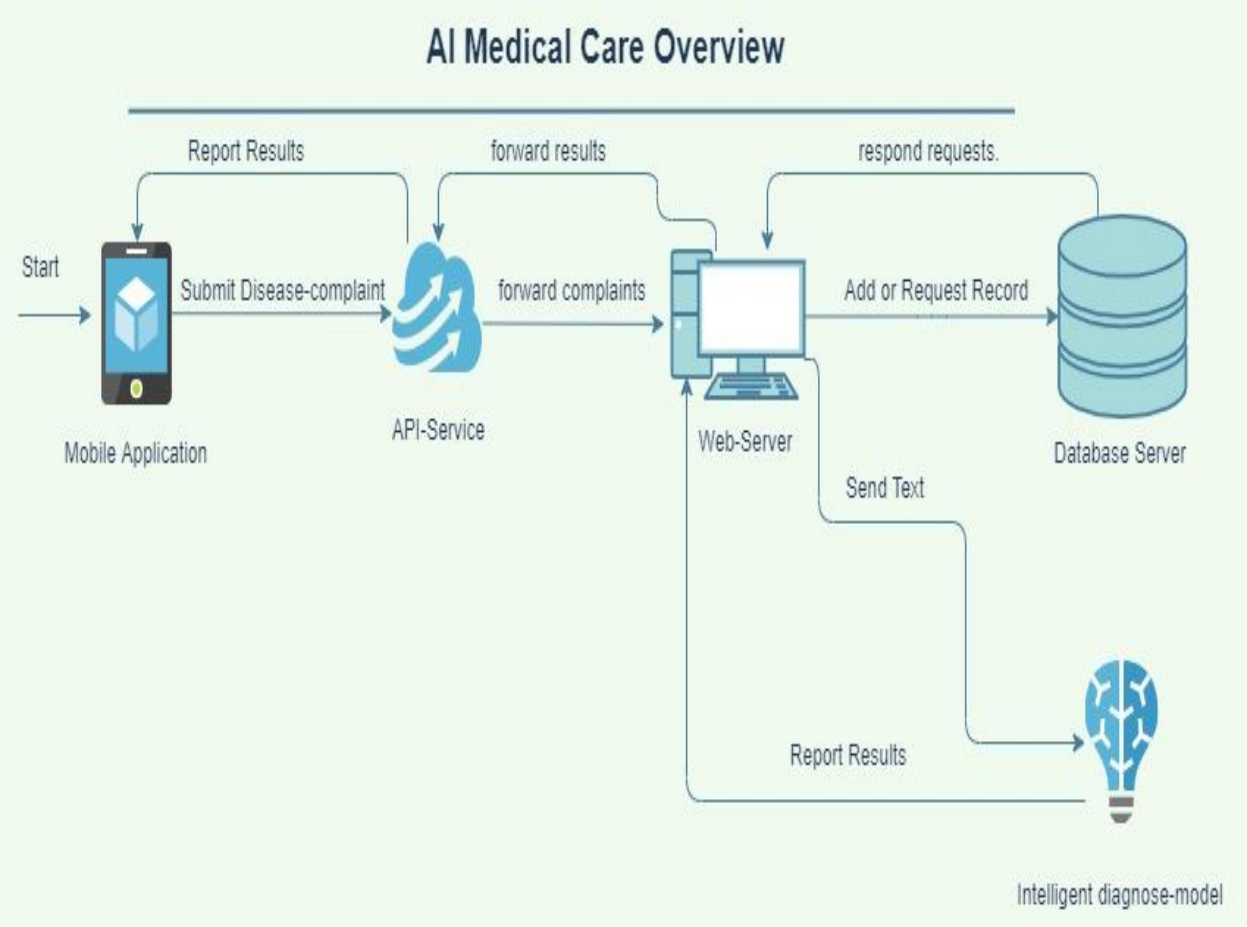


Figure 15. System deployment diagram

4.2 Algorithms:

NLP:

Natural language processing (NLP) refers to the branch of computer science—and more specifically, the branch of artificial intelligence or AI—concerned with giving computers the ability to understand text and spoken words in much the same way human beings can.[7]

Several NLP tasks break down human text

- **Part of speech tagging**, also called grammatical tagging, is the process of determining the part of speech of a particular word or piece of text based on its use and context
- **Word sense disambiguation** is the selection of the meaning of a word with multiple meanings through a process of semantic analysis that determine the word that makes the most sense in the given context
- **Named entity recognition,** or NEM, identifies words or phrases as useful entities. NEM identifies 'Kentucky' as a location or 'Fred' as a man's name.
- **Sentiment analysis** attempts to extract subjective qualities—attitudes, emotions, sarcasm, confusion, suspicion—from text.

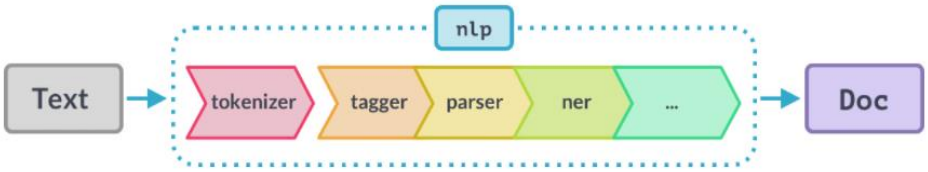


Figure 16. NLP Pipeline

NLTK:

The Natural Language Toolkit, or more commonly NLTK, is a suite of libraries and programs for symbolic and statistical natural language processing (NLP) for English written in the Python programming language.

Our Custom Symptoms and disease NER :
Accuracy:

```
=====================================
Interation = 0
Losses = {'ner': 12413.5455307262}
==============VALID DATA=========================
F1-score = 0.6959393136506714
Precision = 0.621686450512136
Recall = 0.865552460538533
==============TEST DATA=========================
F1-score = 0.6997005348310342
Precision = 0.6271857804405306
Recall = 0.8627726952850105
=====================================
```

```
=====================================
Interation = 12
Losses = {'ner': 2834.2658785035437}
==============VALID DATA=========================
F1-score = 0.9432400932704061
Precision = 0.952574612017509
Recall = 0.949021900929979
==============TEST DATA=========================
F1-score = 0.915692015779982
Precision = 0.9289043420025123
Recall = 0.9240625314165075
=====================================
```

```
=====================================
Interation = 29
Losses = {'ner': 1273.7032793850776}
==============VALID DATA=========================
F1-score = 0.9367737179043095
Precision = 0.9439656181021083
Recall = 0.9413520802935845
==============TEST DATA=========================
F1-score = 0.8947205152342379
Precision = 0.9054416945162899
Recall = 0.9038372820392525
=====================================
```

Figure 17. Our Custom Symptoms and disease NER

TF-IDF Vectorizer:

Is the base building block of many NLP pipelines. It is a simple technique to vectorize text documents — i.e. transform sentences into arrays of numbers — and use them in subsequent tasks.

accuracy_score : 0.994748687171793

accuracy_report :

	precision	recall	f1-score	support
0	0.98	1.00	0.99	49
1	0.98	0.98	0.98	61
2	1.00	0.98	0.99	45
3	1.00	1.00	1.00	70
4	1.00	0.98	0.99	42
5	1.00	0.98	0.99	63
6	1.00	1.00	1.00	39
7	0.98	1.00	0.99	61
8	1.00	0.96	0.98	52
9	1.00	1.00	1.00	64
10	1.00	1.00	1.00	61
11	1.00	1.00	1.00	54
12	1.00	1.00	1.00	51
13	1.00	0.98	0.99	43
14	1.00	1.00	1.00	63
15	0.98	1.00	0.99	46
16	1.00	1.00	1.00	40
17	0.96	1.00	0.98	47
18	1.00	1.00	1.00	45
19	1.00	1.00	1.00	67
20	0.98	1.00	0.99	52
21	1.00	1.00	1.00	51
22	1.00	1.00	1.00	51
23	1.00	1.00	1.00	64
24	1.00	1.00	1.00	52
accuracy			0.99	1333
macro avg	0.99	0.99	0.99	1333
weighted avg	0.99	0.99	0.99	1333

Figure 18. Symptom's Accuracy

Multinomial Naive Bayes:

This a supervised learning technique that classifies every new document by assigning one or more class labels from a fixed or predefined class. It uses the bag of words approach, where the individual words in the document constitute its features, and the order of the words is ignored.

```
from sklearn.naive_bayes import MultinomialNB

nb = MultinomialNB()
naive_bayes_model = nb.fit(X_train, y_train)
```

```
from sklearn.metrics import accuracy_score

y_pred =naive_bayes_model.predict(X_test)
nb_test = accuracy_score(y_test, y_pred)

print("Test accuracy :{:.2f}".format(nb_test))

```

```
Test accuracy :0.95
```

Figure 19. Disease Classifier

4.3 Methodology

4.1.1 Main Work:

- We can divide our application's main work into main four parts:
  - Learning android developing basics.
  - Before developing our application, we needed to read about android because knowledge about android developing was not sufficient enough to start developing the project.
  - Learning Machine learning techniques.
  - Learning Natural Languages processing techniques.
  - Apply text preprocessing on text after convert it from audio.
  - Apply NLP techniques on text
  - Apply machine learning on the data set.

- Applications and languages used:
  - Languages used:
    - kotlin language.
    - Python language.
    - SQLite language.

  - Applications used:
    - Android Studio.
    - Anaconda Platform.

4.1.2 Sample Scenarios

1. Doctor and Patient (Diagnose Process)
   1.1. The doctor opens the application and asked to enter a username and password to authenticate his/her identity and scan the QR-Code of the patient.
   1.2. The doctor starts a diagnose process and ask the patient to describe his/her symptoms in details, the recorded audio is then converted to a text for the doctor and the patient to review then the doctor submit it for the machine learning model to analyze it.
   1.3. The doctor receives the generated report that describe symptoms and diseases.
   1.4. The doctor confirms the report and the report is saved at the patient medical history.

2. Patient (Diagnose Process):
   2.1. The patient opens the application and authenticate his/her identity by providing a username and password.
   2.2. The patient starts a stand-alone diagnose process and start describing all his/her symptoms in details in text or voice then the recorded text is shown to be reviewed.
   2.3. The patient receives a report with the symptoms and the diagnose.

3. Patient or Doctor (First-Use):
   3.1. The doctor/patient opens the application for the first time and click on Sign up button
   3.2. The doctor/patient starts filling the required information to complete the registration process
   3.3. A registration confirmation notification is shown and the user is ready to log on into his own account.

Chapter 5

Conclusion and Future work

5.1 Conclusion

The timely access to healthcare avoiding unnecessary time wastage of patients and doctors is a major issue in the whole world especially in the Middle-East and Egypt where a lot of patients get in worse health conditions due to lack of healthcare-ness. However, considering the exponential growth of mobile users and the need for a real-time medical diagnosis assistance tool, it is therefore important to explore the need for a cost-effective telehealthcare platform, which allows the earlier detection of diseases and effective communication with patients (users) to a doctor with the assistance of a diagnose and treatment system.[8]

Based on the highlighted needs, this study was able to successfully build mobile-based health care and diagnosis system, which aids the doctor in the diagnosis process to avoid any misdiagnose and in-proper given treatment based on the medical history in addition to the ability of the doctor to scan and explore the medical history of the patients to take considerations and a dedicated account where patients and doctors can check their profiles to do the mentioned above.

We were able to combine NLP and machine learning algorithm and integrate them into Mobile-based Application to offer diagnosis, treatments based on a recorded speech conversation between the doctor and the patient and the medical history of the patient.

5.2 Future work

Integrate more features and machine learning models to improve the accuracy of the diagnose process and the diversity of the diseases that being diagnosed (i.e.) a computer vision model to scan medical X-Rays. Add support Arabic language in diagnose process.

5.3 References

1. A. Langer, R. Banga, A. Mittal, L. V. Subramaniam, and P. Sondhi, “A text based drug query system for mobile phones,” International Journal of Mobile Communications, vol. 12, no. 4, pp. 411–429, 2014.
2. V. S. Pendyala, Y. Fang, J. Holliday, and A. Zalzala, “A text mining approach to automated healthcare for the masses,” GHTC, vol. 8, pp. 28–35, 2014
3. R. Carson-Stevens, J.-A. Medina-Merodio, R. Plata, J.-J. Martinez-Herraiz, and J.-M. Gutierrez-Martinez, “A laboratory test expert system for clinical diagnosis support in primary health care,” Applied Sciences, vol. 5, no. 3, pp. 222–240, 2015.
4. A. Atutxa, A. D. de Ilarraza, K. Gojenola, M. Oronoz, and O. Perez-de-Viñaspre, “Interpretable deep learning to map diagnostic texts to ICD-10 codes,” International Journal of Medical Informatics, vol. 129, pp. 49–59, 2019.
5. C. Combi, M. Zorzi, G. Pozzani, U. Moretti, and E. Arzenton, “From narrative descriptions to MedDRA: automagically encoding adverse drug reactions,” Journal of Biomedical Informatics, vol. 84, pp. 184–199, 2018.
6. H. P. Kanegaye, A. Anastasiou, A. Edwards et al., “Automated classification of primary care patient safety incident report content and severity using supervised machine learning (ML) approaches,” Health Informatics Journal, vol. 23.
7. N. Kloehn, G. Leroy, D. Kauchak et al., “Improving consumer understanding of medical text: development and validation of a new subsimplify algorithm to automatically generate term explanations in English and Spanish,” Journal of Medical Internet Research, vol. 20, no. 8, p. e10779, 2018.
8. A. Sarker, A. Z. Klein, J. Mee, P. Harik, and G. Gonzalez-Hernandez, “An interpretable natural language processing system for written medical examination assessment,” Journal of Biomedical Informatics, vol. 98, 2019.