

DOC(Doctor on call)

The project submitted to the
SRM University – AP, Andhra Pradesh
for the course project of

CSE-305L Software Engineering Lab

Submitted by

Revanth Gallam	AP21110010979
Bhanu Bharath Pyla	AP21110010995
Spurthi Kavuri	AP21110011040
Himabindhu Ch	AP21110011249



Under the Guidance of
Dr. Hema Kumar Yarnagula

SRM University-AP
Neerukonda, Mangalagiri, Guntur
Andhra Pradesh – 522 240
[May, 2024]

Table of Contents

Table of Contents	1
Abstract	2
System Requirements	2
DFD	11
E.R.Diagram:.....	12
Class Diagram:.....	15
Sequence diagram	16
USE CASE	17

Abstract

This project presents a comprehensive healthcare application designed to improve the interpretability, transparency, and trustworthiness of medical diagnoses through the use of explainable artificial intelligence models. As AI continues to play an important role in healthcare, this project addresses the critical need for transparent and justifiable decision-making processes in medical applications.

The project is focused on a specified healthcare use case, such as disease diagnosis and requires the collection and preprocessing of a diverse and high-quality dataset. To ensure transparency the project uses both naturally interpretable models, such as decision trees and linear models, and post-hoc explainability techniques, such as LIME and SHAP, for more complex models.

The healthcare application's user interface is designed to present not only predictions but also detailed explanations for each prediction, allowing healthcare professionals and end users to participate in AI driven decision-making process. Integration with electronic health records (EHR). improves prediction accuracy by incorporating relevant patient histories.

System Requirements

1. Introduction

The software **DOC** (Doctor On Call) version v1.0 will be created for a virtual medically trained AI health care consultant. The explainable AI developed in this software will assist customers with the customers medical diagnoses without the need to visit a physical hospital. Customers interact with **DOC** via a user-friendly interface, allowing them to access their medical data and receive a proper consultation with a detailed explanation of their current health condition.

1.1 Purpose

This SRS defines External Interface, Performance and Software System Attributes requirements of **DOC**. This document is intended for the following group of people:-

- ✓ Developers for the purpose of maintenance and new releases of the software.
- ✓ Reviewed by Ministry of HealthCare
- ✓ Documentation writers.
- ✓ Testers.

1.2 Scope

This document applies to Health Care chat bot software **DOC**. This software facilitates the user to get An AI medical consultant which helps the user to get the medical diagnosis with out going to an hospital. This software offers benefits such as reviewing the Blood

work and X-Ray scans, finding the medicine which is best suitable for a given diagnosis and it can also send notifications as a reminder to take medicine

The software takes as input the Name, Gender and other medical records of the user for analysis. The outputs then comprise of an interactive display that lets the user select the desirable service that he wants to perform like analysing his blood work or any discrepancy in blood sugar levels or Blood pressure.

1.3 Acronyms, and Abbreviations.

EHR	Electronic health record.
FAQ	Frequently Asked Questions.
HIV	Human Immunodeficiency Virus.
MRI	Magnetic Resonance Imaging.
PT/OT	Physical Therapy/Occupational Therapy.
ALS	Advanced Life Support.
EKG	Electrocardiogram.
CBC	Complete Blood Count.
ADHD	Attention Deficit Hyperactivity Disorder.
BP	Blood pressure.
BMI	Body Mass Index.
UTI	Urinary Tract Infection.
LIME	Local Interpretable Model-agnostic Explanations.
SHAP	Shapley Additive exPLANations.
XAI	Explainable AI

2. Overall Description

2.1 Product perspective

1. **Integration with Existing Infrastructure:** The healthcare application seamlessly integrates with Electronic Health Records (EHR) systems, enhancing its functionality and ensuring compatibility with established healthcare workflows.
2. **Utilization of Advanced AI Techniques:** Leveraging a combination of interpretable models and post-hoc explainability techniques, such as LIME and SHAP, the application provides transparent insights into complex AI-driven diagnostic processes.
3. **User-Centric Design:** With a focus on user experience, the application features an intuitive interface tailored to the needs of healthcare professionals and end-users. This design promotes ease of use and accessibility, facilitating informed decision-making.
4. **Enhanced Trust and Transparency:** By providing clear explanations for predictions and insights, the application fosters trust and confidence in AI-driven diagnostic capabilities, promoting acceptance and adoption within the medical community.
5. **Alignment with Ethical Standards:** The application adheres to ethical principles and guidelines governing the responsible use of AI in healthcare, ensuring the protection of patient privacy and confidentiality.

2.2 Product functions

The major functions that DOC performs are:

2.2.1 login page:

- ✓ Users are prompted to input their credentials (username and password) to access the application's dashboard. They can also utilize features like "Remember Me" for convenience, initiate password recovery via the "Forgot Password" link, or create a new account by clicking on "Sign Up."

2.2.2 profile :

- Each patient has a dedicated profile page displaying relevant information, including demographics, medical history, and current treatment plans.
- Tabs or sections organize data into categories such as demographics, medical history, lab results, and imaging studies.

2.2.3 Meal Logging:

- ✓ Users can log their meals and snacks throughout the day using a searchable database of foods and ingredients. For each item logged, the application displays nutritional information, including calories and nutrient breakdown

per serving size. Users can easily adjust portion sizes or servings to accurately reflect their consumption

2.2.4 Diagnostic Module:

- ✓ A dedicated section allows users to input symptoms or diagnostic test results for analysis.
- ✓ Predicted diagnoses and associated confidence levels are displayed, along with explanations highlighting key factors influencing the diagnosis.
- ✓ Following a diagnosis, the interface presents recommended treatment options, including medications, procedures, and lifestyle changes.
- ✓ Detailed explanations accompany each recommendation, outlining the rationale and potential outcomes

2.3 User classes and characteristics

The medical care application with AI accommodates different types of users, each with specific needs and levels of familiarity:

2.3.1 User A

Novice Healthcare Consumer, User A is new to using digital healthcare applications and may have limited experience with AI-driven tools. The application features intuitive interfaces with clear explanations of each function, tailored to User A's level of understanding. Interactive tutorials, accompanied by visual and audio aids, assist User A throughout the healthcare journey, ensuring accessibility and ease of use.

2.3.2 User B

Experienced Healthcare Consumer, User B is familiar with using digital healthcare applications and has previous experience with AI-driven healthcare tools. The application provides streamlined interfaces designed to expedite healthcare processes for User B. Help resources are available but are minimal, allowing User B to navigate the application efficiently and make informed healthcare decisions.

2.3.3 Medical Personnel

Healthcare Professionals, Medical Personnel, such as doctors and nurses, utilize the application for patient care and diagnosis support. These users have expertise in healthcare practices and are accustomed to utilizing AI-driven tools in their workflows. The application offers advanced features and diagnostic support tailored to the needs of medical professionals, enabling efficient patient management and decision-making.

2.4 Constraints

The major constraints that the project has are as follows:-

- ✓ The application allows only one user to access patient records at a time, preventing concurrent access to ensure data integrity.
- ✓ Users are allowed a maximum of three unsuccessful login attempts before their account is temporarily blocked for 30 minutes.
- ✓ Simultaneous access to a patient's medical records through both the application and other healthcare channels is prohibited to maintain data consistency.
- ✓ Users are restricted to a maximum of 100 diagnostic queries per session to manage server load and ensure timely responses. Treatment recommendations are limited to 10 per patient per day to prevent overloading healthcare providers with excessive information.
- ✓ *Before processing diagnostic queries or treatment recommendations, the application verifies the integrity of patient data using checksum algorithms to minimize errors.*
- ✓ *The application requires a minimum of 8GB RAM and a quad-core processor to execute AI algorithms effectively and ensure optimal performance.*
- ✓ *The application complies with HIPAA regulations and requires all user interactions to be logged for audit purposes, with logs retained for a minimum of seven years.*
- ✓ *Integration with EHR systems follows HL7 standards, ensuring compatibility and standardized data exchange protocols.*
- ✓ *The application requires NVIDIA GPUs with a minimum of 8GB VRAM for accelerated AI processing, along with the installation of TensorFlow v2.5 or higher as the primary AI framework.*
- ✓ *User support resources include a comprehensive user manual and video tutorials, with access to a 24/7 helpdesk for immediate assistance. All users are required to complete a mandatory training course before accessing the application.*
- ✓ *The application requires NVIDIA RTX series GPUs with a minimum of 16GB VRAM for AI acceleration, along with the installation of TensorFlow v2.6.0 or higher and CUDA Toolkit v11.4. Server infrastructure must be hosted on a redundant architecture with at least 99.99% uptime SLA (Service Level Agreement) to minimize downtime and ensure continuous availability.*
- ✓ *User support resources include a knowledge base with over 100 FAQs (Frequently Asked Questions) and a dedicated community forum for peer-to-peer support. All users are required to complete a 2-hour online training course on data privacy, security best practices, and application usage before gaining access to the system. Training refreshers are provided annually to reinforce knowledge retention.*

2.5 Assumptions and Dependencies

The document application requirements and functionality may be influenced by the following factors:

- ✓ The application's functionality may need to be updated in response to changes in organizational policies regarding document management. Timely incorporation of policy changes is critical to avoid disruption and ensure compliance with regulatory requirements.
- ✓ The application is dependent on the use of SQLAlchemy Database for data storage and management. Any changes or updates to the database system may impact the application's performance and functionality.
- ✓ Potential conflicts may arise if multiple users attempt to access and modify the same document simultaneously. The application must implement concurrency control mechanisms to prevent data corruption and ensure data integrity.
- ✓ While no specific quantitative measures are mandated for the application's speed and memory usage, it is assumed that all functions will be optimized to deliver efficient performance. Continuous monitoring and optimization of system resources are essential to maintain optimal performance.
- ✓ The scope of the document application is expected to expand significantly in the future to accommodate evolving user needs and technological advancements. Flexibility in the application's design and architecture is necessary to facilitate seamless scalability and adaptation to future requirements.

Addressing these assumptions and dependencies proactively ensures the robustness, flexibility, and adaptability of the document application to meet evolving demands and organizational objectives.

3. External Interface Requirements

3.1 User Interfaces

The interface provided to the user should be a very user-friendly one and it should provide an optional interactive help for each of the service listed. The interface provided is a menu driven one and the following screens will be provided:-

1. A login screen is provided in the beginning for entering the required username/ pin no. and account number.

2. An unsuccessful login leads to a reattempt (maximum three) screen for again entering the same information. The successful login leads to a screen displaying a list of supported languages from which a user can select any one.
3. In case of administrator, a screen will be shown having options to reboot system, shut down system, block system, disable any service.
4. In case of reboot/ shut down, a screen is displayed to confirm the user's will to reboot and also allow the user to take any backup if needed.
5. In case of blocking system, a screen is provided asking for the card no. By entering the card no of a particular user, system access can be blocked for him.
6. Administrator is also provided with a screen that enables him to block any service provided to the user by enter in the name of the service or by selecting it from the list displayed.
7. After the login, a screen with a number of options is then shown to the user. It contains all the options along with their brief description to enable the user to understand their functioning and select the proper option.
8. A screen will be provided for user to check his profil.
9. A screen will be provided that displays the diet management.
10. A screen will be provided for the user to check their diagnosis of Cancer.

3. system features

3.1 Virtual Medical Diagnosis

3.1.1 Description

The system is designed to provide the user with a Virtual health care application

Which helps the user to take care of his health with the help of his mobile or a pc with out consulting an actual doctor. The functioning of the system be as follows:

- At the start, the user is provided with a log in screen and he is required to enter his password which are then verified by the software. In case if an unsuccessful attempt a user is asked again for his credentials but the maximum number of attempt given to the user is limited to 3 only.
- After a successful log in, the user is presented with a display of the dashboard which displays a set of services along with their brief description, enabling the user to understand their functioning. The user can select any on the service and proceed with the process.
- If the user selects the medical diagnosis service, then he needs to provide the necessary Blood work and other records required by the software based on your symptoms.
- The software also provides the user with a number of miscellaneous services such as :
 - The software also provides meal logging which helps user to track their meals and keep track of their nutrition.

3.1.2 Sequencing Information

The information about the users and their account should be entered into the database prior to any of the transactions and the backup be maintained for all account information

3.1.3 Validity Checks

- In order to gain access to the system, the user is required to enter his/her correct Username and Password.
- The user can access only one account at a time
- Also if the user is an administrator, he is required to enter his login id in order to access and change the facilities provided by the system

3.1.4 Sequencing Information

The information about the users and their account should be entered into the database prior to any of the transactions and the backup be maintained for all account information

3.1.5 Error Handling/ Response to Abnormal Situations

If any of the above validation/sequencing flow does not hold true, appropriate error messages will be prompted to the user for doing the needful.

4. Other Nonfunctional Requirements

4.1 Performance Requirements

4.1.1 Capacity

- The DOC shall provide customers a 24hour service.

4.1.2 Quality

The primary objective is to produce quality software. As the quality of a piece of software is difficult to measure quantitatively, the following guidelines will be used when judging the quality of the software:

- Consistency – All code will be consistent with respect to the style. (This is implied when adhering to the standard).
- Test cases – All functionality will be thoroughly tested

4.2 Software System Attributes

4.2.1 Reliability

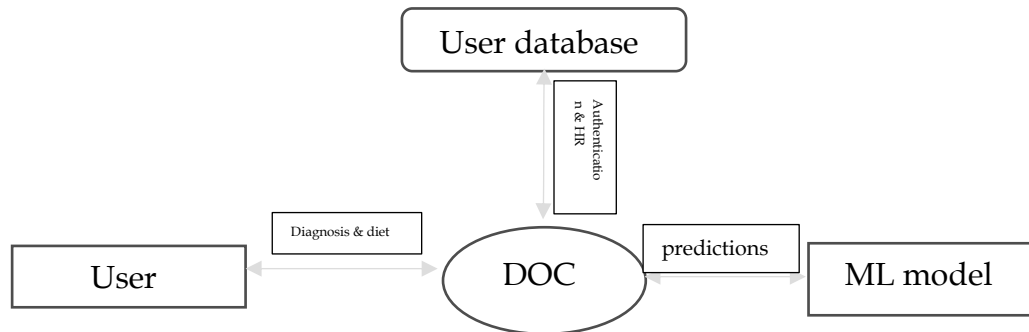
- That data communication protocol shall be such that it ensures reliability and quality of data
- The memory system shall be of non-volatile type

4.2.2 Security

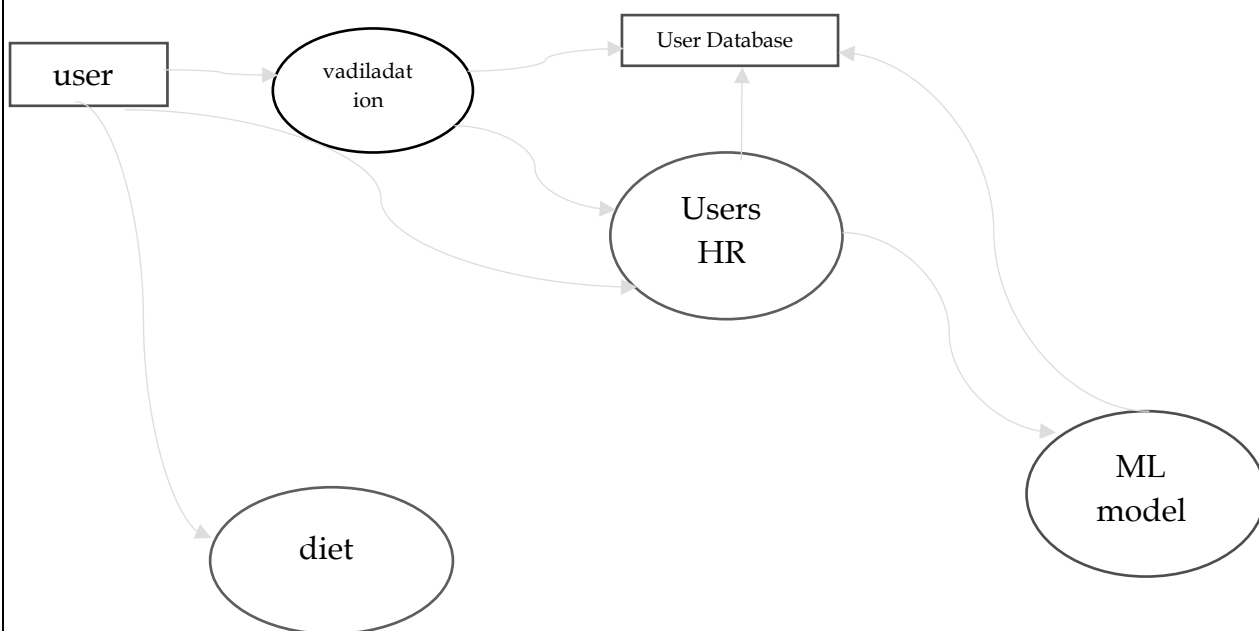
- The system ensure that personal health data is not disclosed inappropriately or accessed by unauthorized parties.
- Users enter a passcode each time they log in to prevent unauthorized access.
- Uses secure communication protocol HTTPS to protect data during transmission. Ensures that APIs and data exchanges between components are secure.
- Regularly conducts security code reviews and vulnerability assessments.
- The XAI system should provide transparent explanations for its predictions.
- Obtains informed consent from patients before using their data for model training. Clearly explains how their data will be used and the benefits of XAI.
- Conduct periodic security audits to identify vulnerabilities.
- Schedule of fixed assets.

DFD

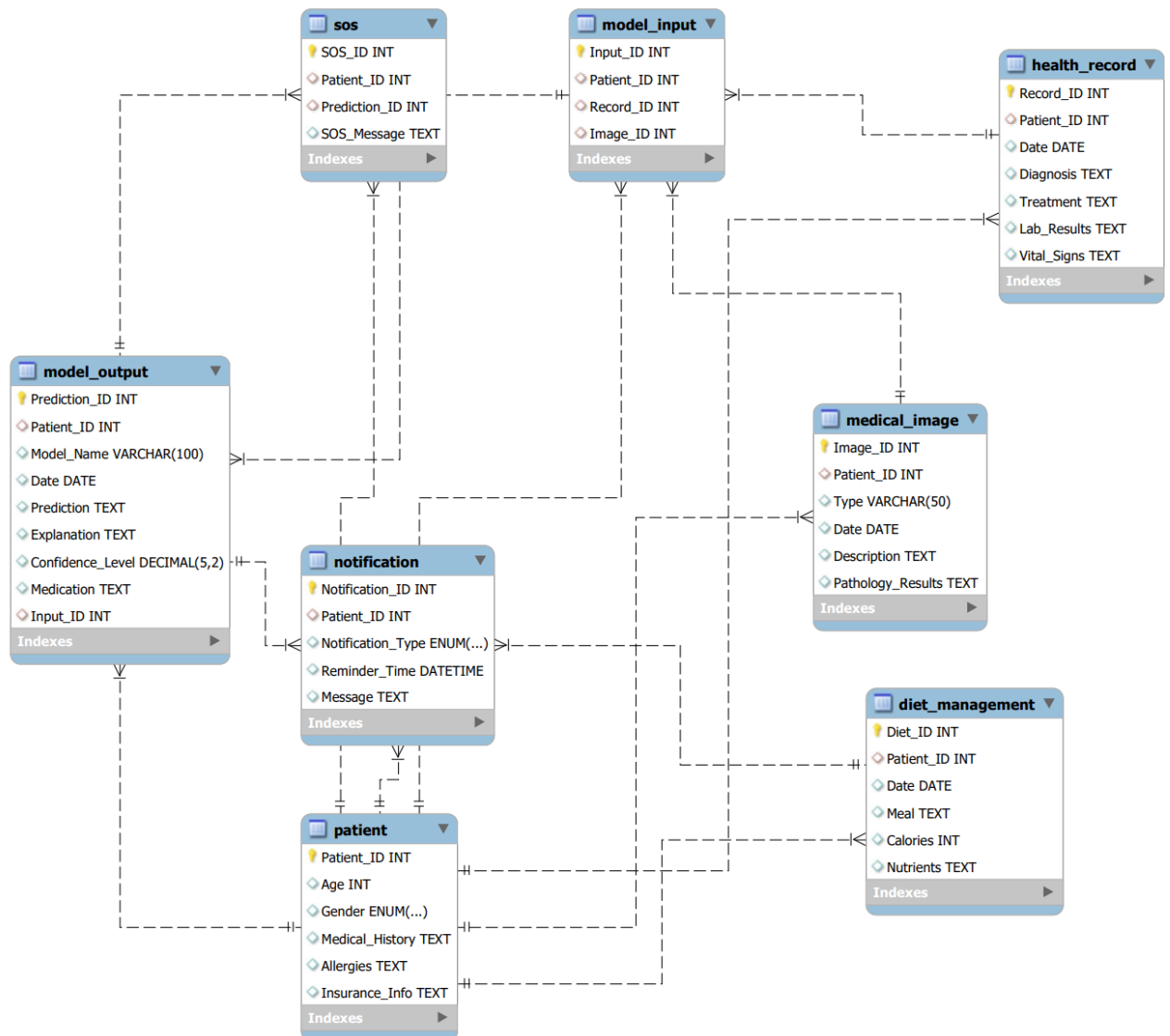
1.1.Login Management(Level-0)



1.2. Level-1



E.R.Diagram:



Data Dictionary:

User table:

Attribute Name	How It's Used	Length	Format	Description
UserID	Primary Key	INT	numeric	Unique identifier for each user
First name		VARCHAR(255)	Alphanumeric	Username for login
Password		VARCHAR(150)	Alphanumeric	Password for login
Email		VARCHAR(255)	Alphanumeric	User's email address

Health Record Table

Attribute Name	How It's Used	Length	Format	Description
HR ID	Primary Key	INT	Numeric	Unique identifier for each HR record
UserID	Foreign Key	INT	Alphanumeric	References UserID in the user table
GENDER		VARCHAR(10)	Alphanumeric	Gender of a patient
Medical conditions		VARCHAR(1000)	Alphanumeric	Medical Conditions of the user
date		Date	DateTime	Date of the HR

Diagnosis Table

Attribute Name	How It's Used	Length	Format	Description
Diagnosis ID	Primary Key	INT	Numeric	Unique identifier for diagnosis schedule
UserID	Foreign Key	INT	Numeric	References UserID in the user table
Gender		DATE	Date	Gender of the user
Age at diagnosis		TIME	Time	Age of the patient during diagnosis
Idh1		VARCHAR(255)	Alphanumeric	Blood results
Tp53		VARCHAR(255)	Alphanumeric	Blood results
Atrx		VARCHAR(50)	Alphanumeric	Blood results
pten		VARCHAR(50)	Alphanumeric	Blood results
egfr		VARCHAR(50)	Alphanumeric	Blood results

Diet

Attribute Name	How It's Used	Length	Format	Description
Diet ID	Primary Key	INT	Numeric	Unique identifier for each Diet recorded
UserID	Foreign Key	INT	Numeric	References UserID in the user table
Topic		VARCHAR(255)	Alphanumeric	Topic of the quiz
Marks		FLOAT	Numeric	Marks obtained in the quiz
Subject		VARCHAR(255)	Alphanumeric	Subject of the quiz

Class Diagram:

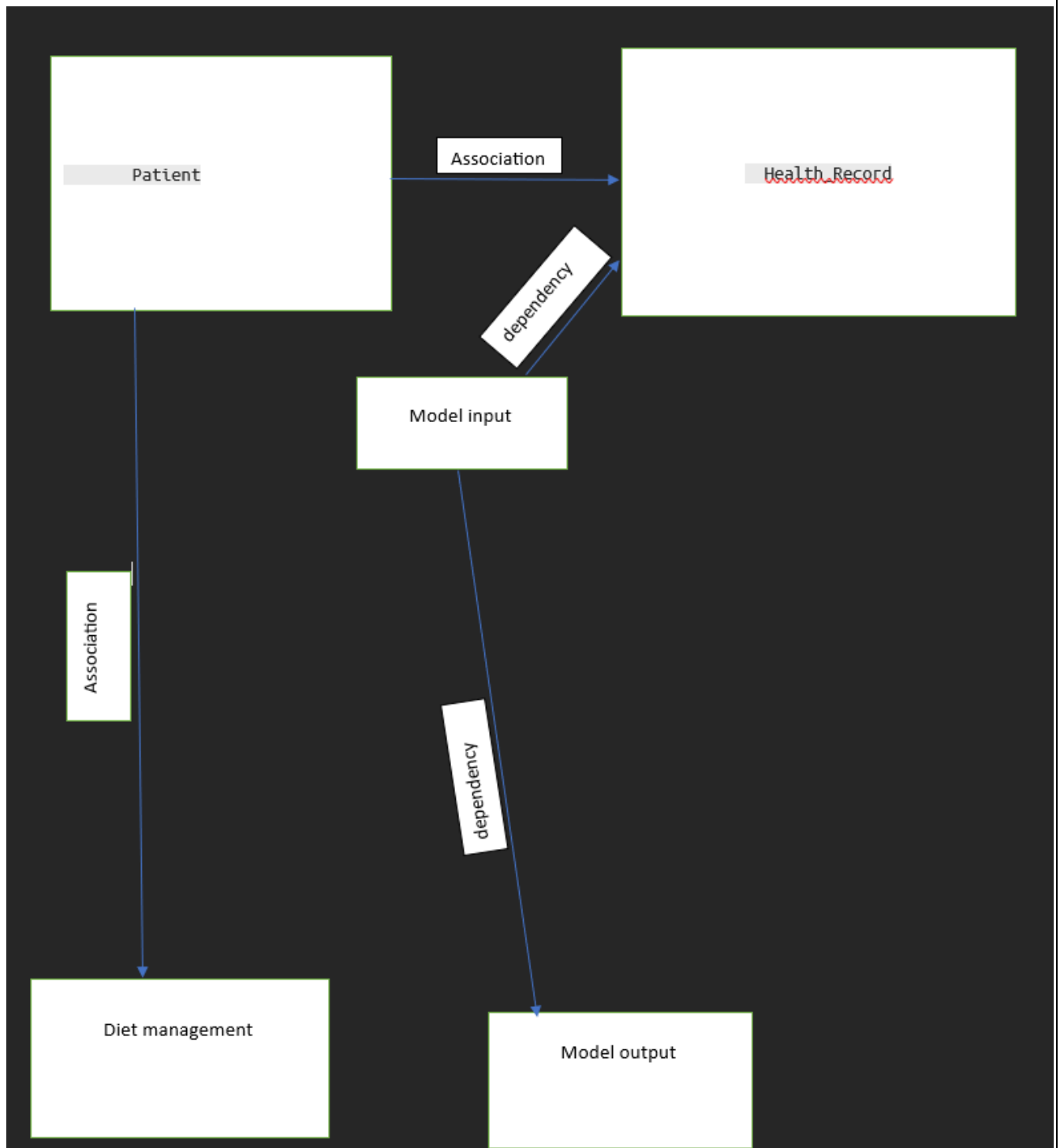
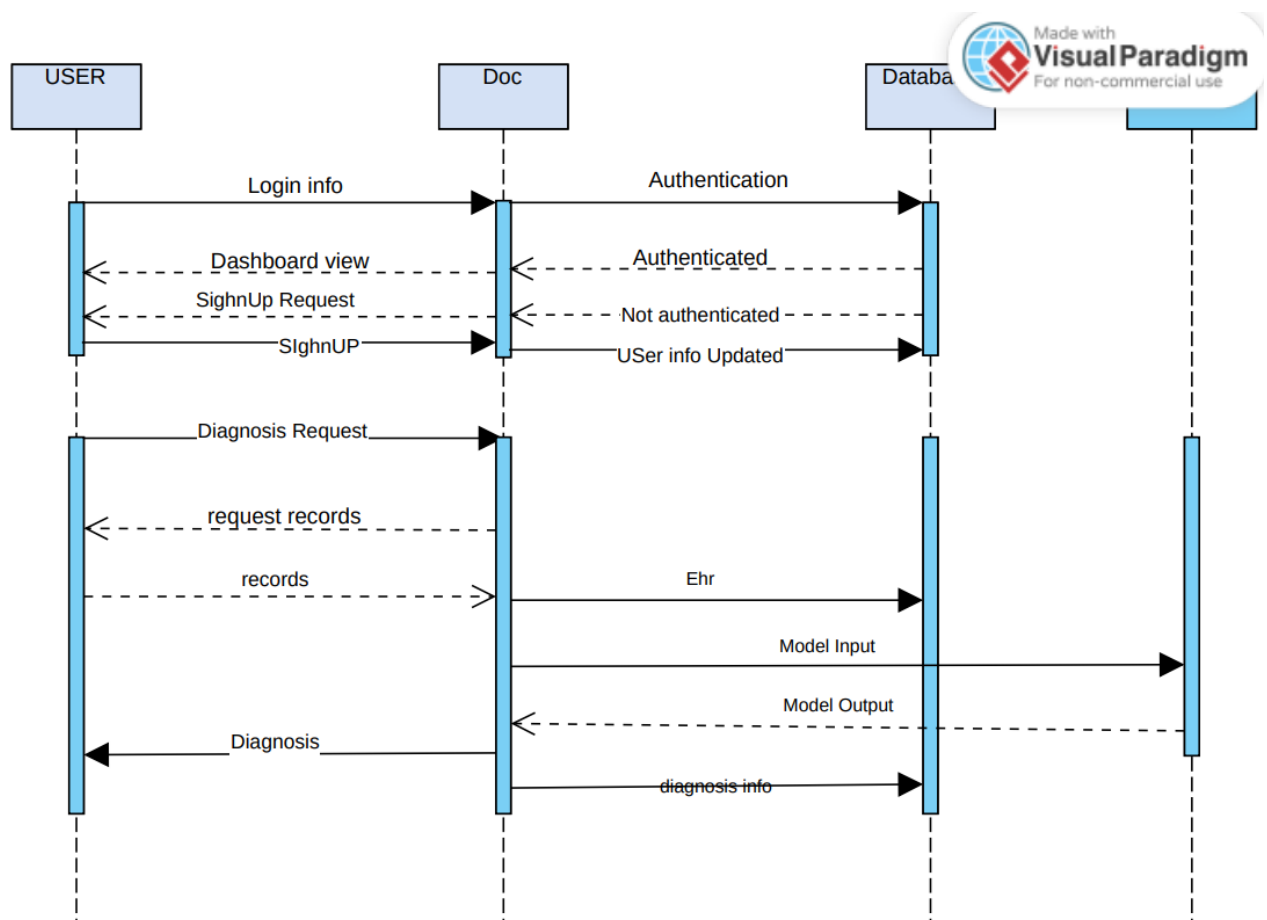


Fig: Class diagram

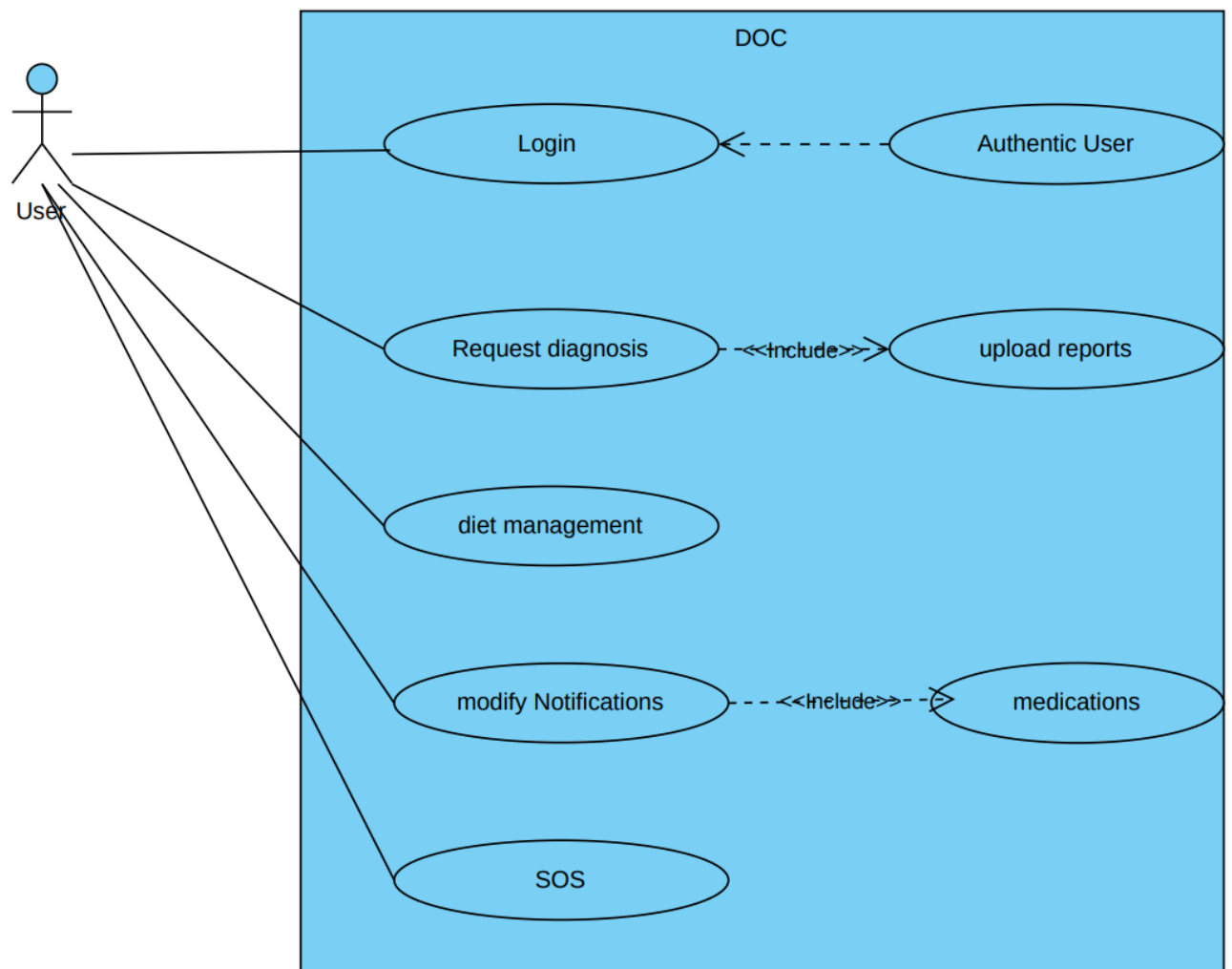
Sequence diagram

2.1. Login



USE CASE

USE CASE DIAGRAM



USE CASE SCENORIO:

Login

Use Case ID	UC-101
Use Case	Login Page
Actors	User
Description	Logging into the System with valid credentials.
Pre-Condition	<ul style="list-style-type: none"> None
Flow of Events	<ol style="list-style-type: none"> 1. User enters the User name and Password. 2. System verifys the credentials. 3. If credentials are valid,the system logs the user in. 4. If credentials are invalid,Aneror message is displayed.
Post-Condition	<ul style="list-style-type: none"> User is logged into the System.
Alternative Flows	None.

diagnosis:

Use Case ID	UC-102
Use Case	diagnosis
Actors	user
Description	User enters his unique user id and password to login, choose diagnosis upload HR's and receive diagnosis.
Pre-Condition	<ul style="list-style-type: none"> User is logged in.
Flow of Events	<ol style="list-style-type: none"> 1. User open's application 2. Login into the application 3. User selects diagnosis

	<ol style="list-style-type: none"> 4. System asks user to upload blood results 5. Data is sent to ML model and predicts the diagnosis
Post-Condition	<ul style="list-style-type: none"> • Redirected to home page.
Alternative Flows	None

Diet management:

Use Case ID	UC-103
Use Case	Diet Management
Actors	User
Description	After user authentication, user uploads diet and receive its analysis.
Pre-Condition	<ul style="list-style-type: none"> • User should already be logged in.
Flow of Events	<ol style="list-style-type: none"> 1. User logs in. 2. Redirect to the home page. 3. Systems asks to enter the food and calories and exercise and calories burned.
Post-Condition	<ul style="list-style-type: none"> • Redirected to the home page.
Alternative Flows	None

Link For the Project:

https://drive.google.com/drive/folders/1MytTMHdUqRA-a-a2dzwxjUANUCeVtdRf?usp=drive_link