Chapter 1

Introduction

CHAPTER 1
INTRODUCTION

1.1 Introduction

According to research, there are 40% of people ignore the general disease which leads to harmful diseases later. The main reason for ignorance is laziness to consult a doctor and time concern people have involved themselves so much that they have no time to make an appointment and consult a doctor which later results in fatal disease. According to research, 25% of people face death due to early ignorance. So we propose to develop this project where a user can sit at their places and have a checkup of their health. The UI is designed in a simple way that everyone can easily operate on it and have a checkup.

1.2 Background

A **web application** (or **web app**) is application software that runs on a web server, unlike computer-based software programs that are run locally on the operating system (OS) of the device. Web applications are accessed by the user through a web browser with an active network connection. Our application is completely done with the help of Machine Learning and Python programming language for developing the backend, HTML, CSS, JavaScript for designing the front end. To predict the diseases we have used datasets that are previously available by the hospitals.

1.3 Motivation

Nowadays many people go to hospitals to know about their health conditions. But they have to travel to get to know their answers and sometimes the patients may not get the results based on various factors such as doctor might be on leave. So to avoid all these confusions and problems we are developing a project which will help all those people who are in need to know the condition of their health.

1.4 Purpose

The purpose of developing this project called “Medico” is to predict the accurate disease of the patient using all their general information and also the symptoms. Using this information, we will compare with our datasets to predict the disease. If this prediction is done at the early stage of the disease with the help of this project then the disease can be cured and in general, this prediction system can also be very useful in the health industry.

1.5 Proposed System

Disease prediction using machine learning is a web application that predicts the disease based on the information provided by the user. If the user wants to consult a doctor, there will be a list of recommended doctors. The user can easily find the doctor’s details. They chat with the doctors. They can also rate doctors based on their consultation.

1.6 Objectives

- ✓ To predict diseases based on the symptoms using datasets
- ✓ To prevent patients from various aspects
- ✓ To consult recommended doctors

1.7 Features of The Proposed System

Our system will have the following features

- ❖ **Secure :** Our system will use an email authentication system to authenticate users. Our system can define whether the user is a patient or a doctor.

- **Accuracy :** Our system will use previous datasets of the hospitals so after comparing it can provide up to 80% of accurate results.
- **Online Chatting :** Our system will have online chatting features. The patients and doctors can easily chat with each other using our system.
- **Rating :** Our system will have a rating system. The users can provide feedback about doctors based on their consultation. Users can rate doctors and write a review about them.

1.8 Project Overview

The Medico application is created for predicting diseases efficiently.

- ✓ Signup
- ✓ Login
- ✓ Check Diseases
- ✓ View Recommended Doctors
- ✓ Chat
- ✓ Review
- ✓ Update Profile

Signup & Login :

At first, the patient or doctor needs to register themselves. After registration, only registered candidates can log in by giving a username and password. Each user has a unique username and password.

Check Diseases :

After successfully logging in, the patients can have a check-up by entering symptoms.

View Recommended Doctors : The patients can view recommended doctors based on their disease.

Chat :

The patient can send messages to doctors and can get suggestions from doctors.

Review :

The user can give a review based on the doctor's consultation.

Update Profile :

The users can update their profiles to modify their personal information.

Chapter 2

Literature Review

CHAPTER 2
LITERATURE REVIEW

2.1 Machine Learning

Arthur Samuel, a pioneer in the field of artificial intelligence and computer gaming, coined the term "Machine Learning". He defined machine learning as – "Field of study that gives computers the capability to learn without being explicitly programmed".

In a very layman manner, Machine Learning(ML) can be explained as automating and improving the learning process of computers based on their experiences without being programmed i.e. without any human assistance. The process starts with feeding good quality data and then training our machines(computers) by building machine learning models using the data and different algorithms. The choice of algorithms depends on what type of data do we have and what kind of task we are trying to automate.

Machine learning implementations are classified into three major categories, depending on the nature of the learning "signal" or "response" available to a learning system which is as follows:-

- **Supervised Learning :**

When an algorithm learns from example data and associated target responses that can consist of numeric values or string labels, such as classes or tags, in order to later predict the correct response when posed with new examples comes under the category of Supervised learning. This approach is indeed similar to human learning under the supervision of a teacher. The teacher provides good examples for the student to memorize, and the student then derives general rules from these specific examples.

- **Unsupervised Learning :**

Whereas when an algorithm learns from plain examples without any associated response, leaving to the algorithm to determine the data patterns on its own. This type of algorithm tends to restructure the data into something else, such as new features that may represent a class or a new series of un-correlated values. They are quite useful in providing humans with insights into the meaning of data and new useful inputs to supervised machine learning algorithms.
As a kind of learning, it resembles the methods humans use to figure out that certain objects or events are from the same class, such as by observing the degree of similarity between objects.

Some recommendation systems that you find on the web in the form of marketing automation are based on this type of learning.

❖ **Reinforcement Learning :**

When we present the algorithm with examples that lack labels, as in unsupervised learning. However, one can accompany an example with positive or negative feedback according to the solution the algorithm proposes comes under the category of Reinforcement learning, which is connected to applications for which the algorithm must make decisions (so the product is prescriptive, not just descriptive, as in unsupervised learning), and the decisions bear consequences. In the human world, it is just like learning by trial and error.

Errors help one learn because they have a penalty added (cost, loss of time, regret, pain, and so on), teaching you that a certain course of action is less likely to succeed than others. An interesting example of reinforcement learning occurs when computers learn to play video games by themselves.

In this case, an application presents the algorithm with examples of specific situations, such as having the gamer stuck in a maze while avoiding an enemy. The application lets the algorithm know the outcome of actions it takes, and learning occurs while trying to avoid what it discovers to be dangerous and to pursue survival. You can have a look at how the company Google DeepMind has created a reinforcement learning program that plays old Atari video games. When watching the video, notice how the program is initially clumsy and unskilled but steadily improves with training until it becomes a champion.

2.2 Characteristics of Machine Learning

Here are seven key characteristics of machine learning

❖ **The ability to perform automated data visualization :**

Machine learning offers several tools that provide rich snippets of data that can be applied to both unstructured and structured data. With the help of user-friendly automated data visualization platforms in machine learning, businesses can obtain a wealth of new insights to increase productivity in their processes**.**

❖ **Automation at its best :**

One of the biggest characteristics of machine learning is its ability to automate repetitive tasks and thus, increasing productivity. A huge number of organizations are already using machine learning-powered paperwork and email automation.

❖ **Accurate data analysis :**

Traditionally, data analysis has always been encompassing the trial and error method, an approach that becomes impossible when we are working with large and heterogeneous datasets. Machine learning comes as the best solution to all these issues by offering effective alternatives to analyzing massive volumes of data. By developing efficient and fast algorithms, as well as, data-driven models for processing data in real-time, machine learning can generate accurate analysis and results.

2.3 Traditional Programming Versus Machine Learning

Traditional programming is a manual process — meaning a person (programmer) creates the program. But without anyone programming the logic, one has to manually formulate or code rules. We have the input data, and someone (programmer) coded a program that uses that data and runs on a computer to produce the desired output. Machine Learning, on the other hand, the input data and output are fed to an algorithm to create a program.

This is the basic difference between traditional programming and machine learning. Without anyone programming the logic, In Traditional programming, one has to manually formulate/code rules while in Machine Learning algorithms automatically formulate the rules from the data, which is very powerful.

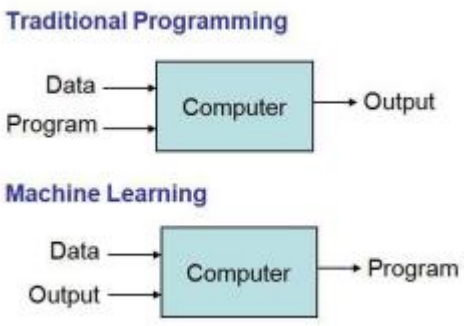

Figure 2.1 : Traditional Programming versus Machine Learning

2.4 Applications

Application of machine learning include-

Web Search Engine :

One of the reasons why search engines like google, bing, etc work so well is because the system has learned how to rank pages through a complex learning algorithm.

Photo tagging Applications :

Be it Facebook or any other photo tagging application, the ability to tag friends makes it even more happening. It is all possible because of a face recognition algorithm that runs behind the application.

Spam Detector :

Our mail agent like Gmail or Hotmail does a lot of hard work for us in classifying the mails and moving the spam mails to the spam folder. This is again achieved by a spam classifier running in the back end of the mail application.

2.5 Data in Machine Learning

It can be any unprocessed fact, value, text, sound, or picture that is not being interpreted and analyzed. Data is the most important part of all Data Analytics, Machine Learning, Artificial Intelligence. Without data, we can't train any model and all modern research and automation will go in vain.

In machine learning, we split data in the following way-

Training Data :

The part of data we use to train our model. This is the data that your model actually sees(both input and output) and learns from.

Validation Data :

The part of data that is used to do a frequent evaluation of the model, fits on the training dataset along with improving involved hyperparameters (initially set parameters before the model begins learning). This data plays its part when the model is actually training.

Testing Data :

Once our model is completely trained, testing data provide an unbiased evaluation. When we feed in the inputs of Testing data, our model will predict some values(without seeing actual output). After prediction, we evaluate our model by comparing it with the actual output present in the testing data. This is how we evaluate and see how much our model has learned from the experiences fed in as training data, set at the time of training.

2.6 Existing System

Prediction using traditional methods and models involves various risk factors and it consists of various measures of algorithms such as datasets, programs, and much more to add on. High risk and low risk patient classification is done based on the tests that are done in a group. But these models are only valuable in clinical situations and not in the big industry sector.

Limitations :

- Time-consuming
- Can not detect in early stage
- Need to conduct tests
- Costly

So to include disease prediction in various health-related industries, we have used the concepts of machine learning and supervised learning methods to build the prediction system.

Chapter 3

Analysis & Requirements

CHAPTER 3
ANALYSIS & REQUIREMENTS

3.1 Proposed Methodology

The proposed work predicts disease by exploring the machine learning classification algorithms and does performance analysis. The objective of this study is to effectively predict if the patient suffers from any disease. The user enters the symptoms as input. The data is fed into a model which predicts the probability of having heart disease. The entire process is given below-

Data Collection :

The dataset used was the human disease Dataset which is a combination of 4 different databases. This database consists of a total of 76 attributes but we have used the already processed UCI Cleveland dataset available on the Kaggle website for our analysis.

Classification :

The input dataset is split into 80% of the training dataset and the remaining 20% into the test dataset. The training dataset is the dataset that is used to train a model. A testing dataset is used to check the performance of the trained model. Classification of the data model is done with the help following algorithms.

Random Forest algorithms are used for classification as well as regression. It creates a tree for the data and makes predictions based on that. Random Forest algorithm can be used on large datasets and can produce the same result even when large sets of record values are missing. The generated samples from the decision tree can be saved so that they can be used on other data. In the random forest there are two stages, firstly create a random forest then make a prediction using a random forest classifier created in the first stage.

The Decision Tree algorithm is in the form of a flowchart where the inner node represents the dataset attributes and the outer branches are the outcome. Decision Tree is chosen because they are fast, reliable, easy to interpret, and very little data preparation is required. In the Decision Tree, the prediction of class labels originates from the root of the tree. The value of the root attribute is compared to the record's attribute. On the result of the comparison, the corresponding branch is followed to that value, and a jump is made to the next node.

Logistic Regression is a classification algorithm mostly used for binary classification problems. In logistic regression instead of fitting a straight line or hyperplane, the logistic regression algorithm uses the logistic function to squeeze the output of a linear equation between 0 and 1. 13 independent variables make logistic regression good for classification.

Naïve Bayes algorithm is based on the Bayes rule[]. The independence between the attributes of the dataset is the main assumption and the most important in making a classification. It is easy and fast to predict and holds best when the assumption of independence holds. Bayes' theorem calculates the posterior probability of an event (A) given some prior probability of event B represented by P(A/B)[10] as shown in equation :

$P(A|B) = (P(B|A)P(A)) / P(B)$

Software Development Model:

We are using the Agile software development model to implement this application.

Agile:

Agile is a time-bound, iterative approach to software delivery that builds software incrementally from the start of the project, instead of trying to deliver all at once. It was specially designed to curate the needs of the rapidly changing environment by embracing the idea of incremental development and developing the actual final product.

Principles:

- The highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- It welcomes changing requirements, even late in development.
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference for the shortest timescale.
- Build projects around motivated individuals. Give them the environment and the support they need, and trust them to get the job done.
- Working software is the primary measure of progress.
- Simplicity the art of maximizing the amount of work not done is essential.
- The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

Development in Agile:

- In Agile development, Design and Implementation are considered to be the central activities in the software process.
- The design and Implementation phase also incorporates other activities such as requirements elicitation and testing.
- In an agile approach, iteration occurs across activities. Therefore, the requirements and the design are developed together, rather than separately.
- The allocation of requirements and the design planning and development as executed in a series of increments. In contrast with the conventional model, where requirements gathering needs to be completed to proceed to the design and development phase, it gives Agile development an extra level of flexibility.
- An agile process focuses more on code development rather than documentation.

3.2 Feasibility Study

A feasibility study is an analysis done to determine the viability of a project from an economic, legal, and technical perspective. Simply put, it gives us an insight into whether a project is doable or not.

A feasibility study, that's well-designed should offer insights on the description of the project, resource allocation, accounting statements, financial data, legal requirements, and tax obligations. It helps to determine whether the project is both possible and profitable for the company to undertake. Hence, this study is mandatorily done before technical development and project execution.

There are five types of feasibility studies based on the area that is examined
1) Technical Feasibility, 2) Economic Feasibility, 3) Legal Feasibility 4) Operational Feasibility and 5) Scheduling Feasibility.

Here we will only discuss the technical feasibility.

3.3 Technical Feasibility

Technical feasibility is concerned with the design and development part of the project. It is concerned with hardware, software, and platform-related issues. The following are the technical specifications of the project.

3.3.1 Hardware Requirements

The following configuration is needed.

- ✓ Microsoft Windows 7/8/10 (32 or 64bit)
- ✓ 4 GB RAM minimum, 8 GB recommended
- ✓ Minimum 2 GB of available disk space
- ✓ 1280 X 800 minimum screen resolution

3.3.2 Software Requirements

The following software is required

- ✓ Pycharm IDE
- ✓ Jupyter Nootbook
- ✓ Anaconda Package Manager
- ✓ Web Browser

3.3.3 Technology Used

Python :

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy-to-learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse.

The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms and can be freely distributed.

Postgre SQL :

PostgreSQL is also known as Postgres, is a free and open-source relational database management system (RDBMS) emphasizing extensibility and SQL compliance. It was originally named POSTGRES, referring to its origins as a successor to the Ingres database developed at the University of California, Berkeley. In 1996, the project was renamed PostgreSQL to reflect its SQL support. After a review in 2007, the development team decided to keep the name PostgreSQL and the alias Postgres.

HTML :

HTML stands for HyperText Markup Language. HTML is the standard markup language for creating Web pages.HTML describes the structure of a Web page. HTML consists of a series of elements.HTML elements tell the browser how to display the content.HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

CSS :

CSS stands for Cascading Style Sheets. It is the language for describing the presentation of Web pages, including colors, layout, and fonts, thus making our web pages presentable to the users.

Chapter 4

Design

CHAPTER 4
DESIGN

4.1 Design

The design phase of a system is a key stage. I will start by introducing our app in a much more detailed and specific way. Then we will demonstrate the web application interface. The interface is the portal through which a user interacts with the software. No matter how good the software intentionally is, if the interface is not intuitive and hard to use by a user then people will not use the software or most people will not. Now keep those in mind we tried to make the interface as simple and intuitive as possible. Then come to the data flow diagram that presents the flow of data over the system. Next, come to the database structure that presents the general structure of the database, tables, and relationships among these tables. Next is the flowchart that shows the sequence of our system.

4.2 System Architecture

A system architecture or systems architecture is the conceptual model that defines the structure, behavior, and more views of a system An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviors of the system. The following figure shows the system architecture of the proposed system.

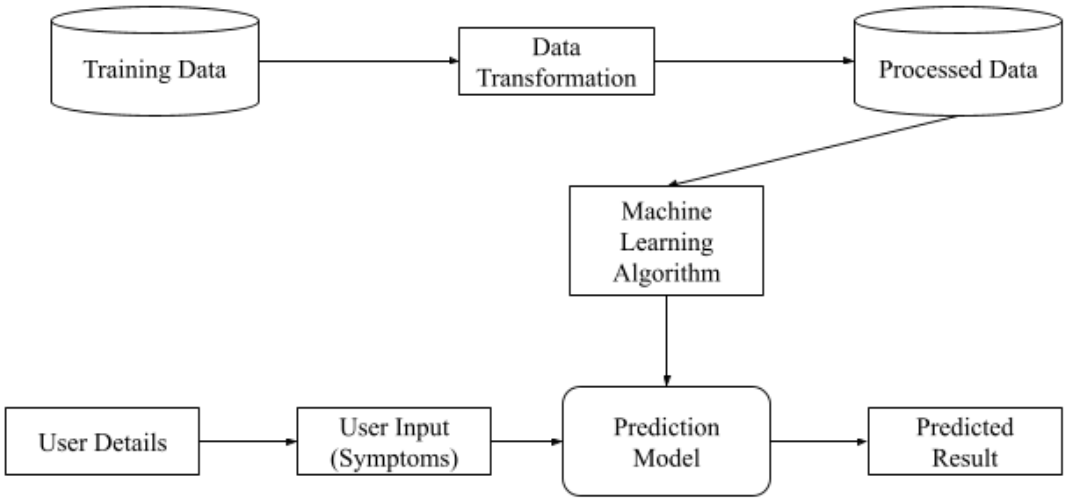

Figure 4.1 : System Architecture

4.3 Flowchart

A flowchart is a formalized graphic representation of a logic sequence, work or manufacturing process, organization chart, or similar formalized structure. The purpose of a flow chart is to provide people with a common language or reference point when dealing with a project or process.

The flow chart of the Medico app is as follows:

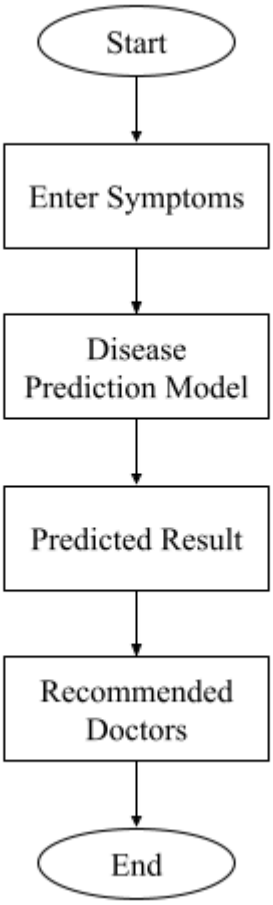

Figure 4.2 : FlowChart

4.4 Use Case Diagram

A use case model is a graphic description of the interactions among the elements of a system. A use case is a methodology used in system analysis to identify, clarify and organize system requirements. Use case diagram can be used for the following purposes-

- ✓ To represent scenarios in which the system interacts with people
- ✓ To represent goals of the system
- ✓ To represent the scope of the system

The Use case diagram of the Medico application is as follows-

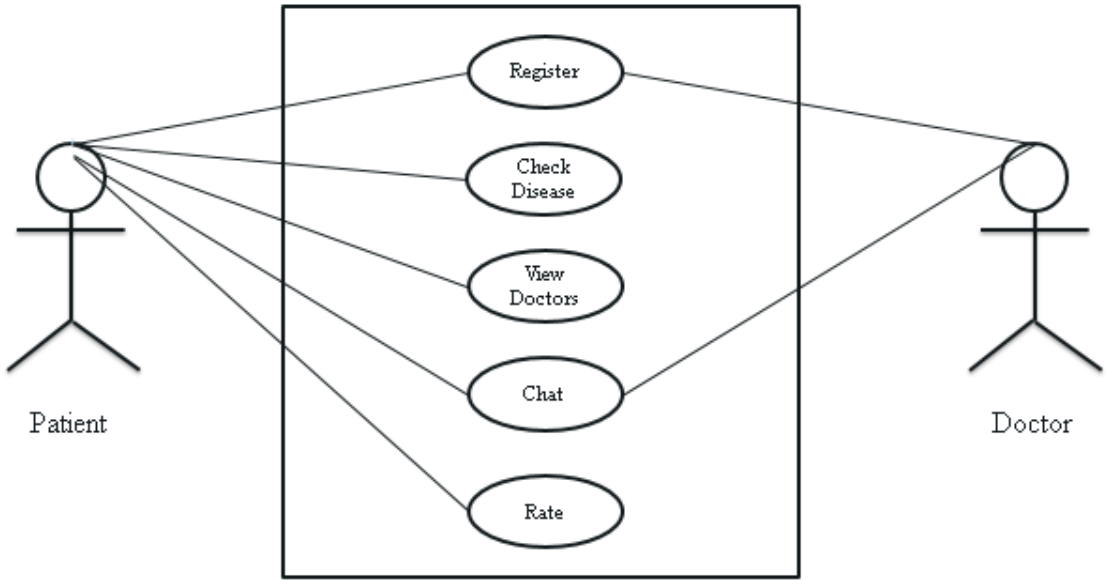

Figure 4.3 : Use Case Diagram

4.5 E-R Diagram

An entity-relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is an object, a component of data. An entity set is a collection of similar entities. These entities can have attributes that define their properties.

By defining the entities, their attributes and showing the relationship between them, an ER diagram shows the logical structure of databases.

An E-R diagram is used to sketch out the design of a database.

E-R diagram of the Medico application is as follows-

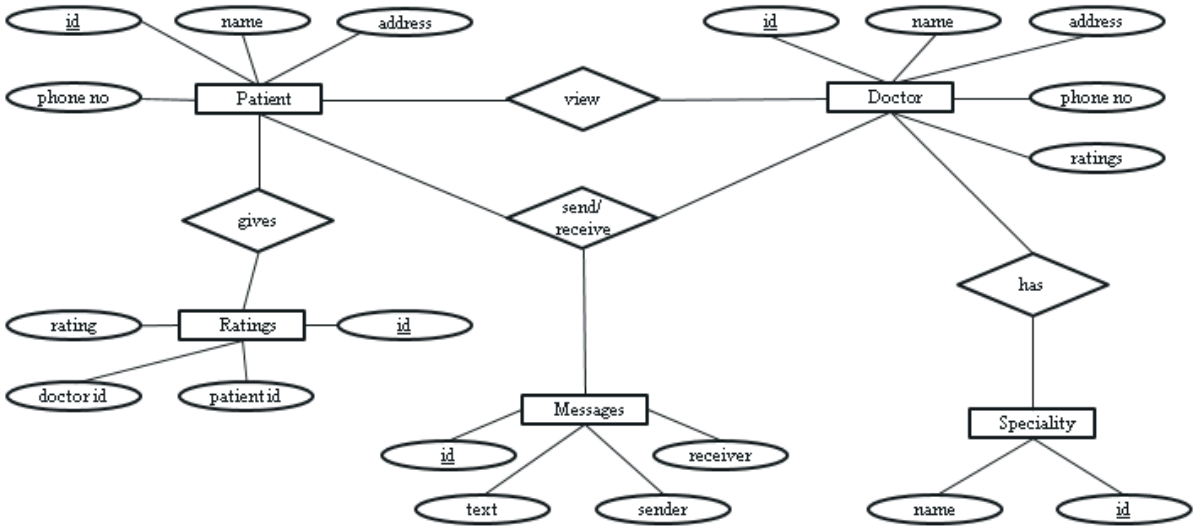

Figure 4.4 : E-R Diagram

4.6 Front-End Design

In any application front-end design is the visual part of an application. By which the user can interact with the application. In the sense of a user, the front-end design is the main thing. In the sense of a developer, it means the language to set the UI/UX of an application.

We have used HTML, CSS to design the user interface of different modules of the system

4.6.1 Landing Page

This page is rendered when the application is visited. It is called the landing page. Following is the landing page of the Medico application.

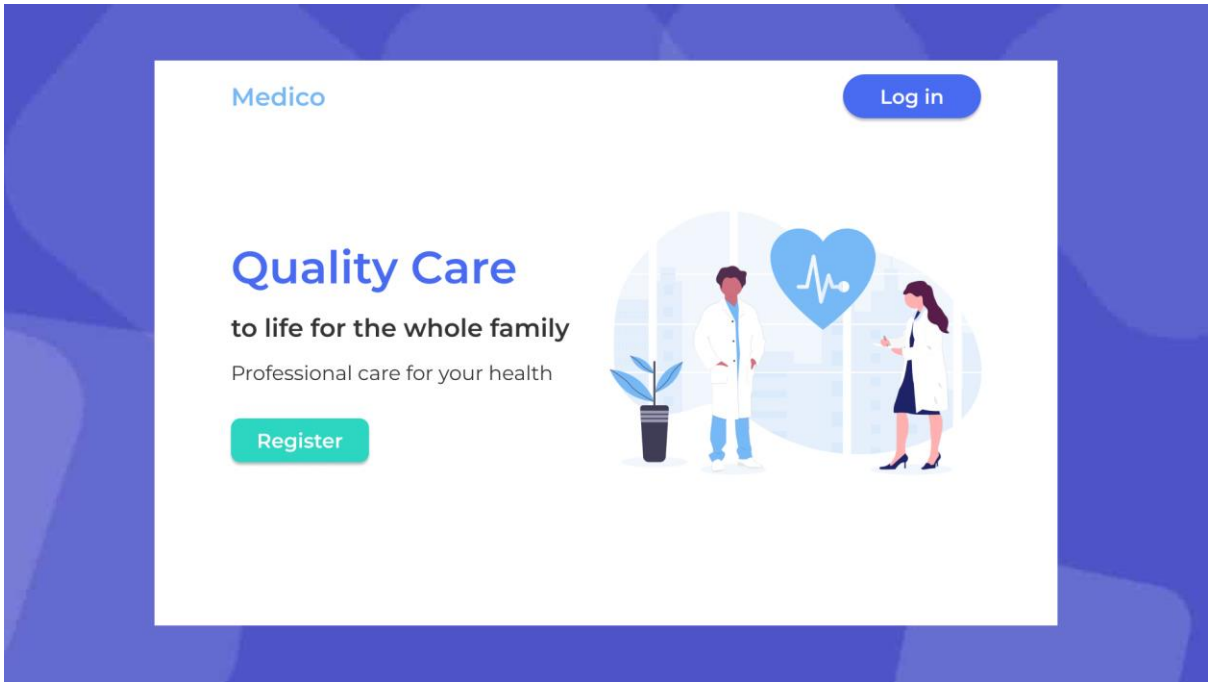

Figure 4.5 : Landing Page

4.6.2 Register Page

Using this module users can create an account after filling up the details. The register page of the Medico application is as follows-

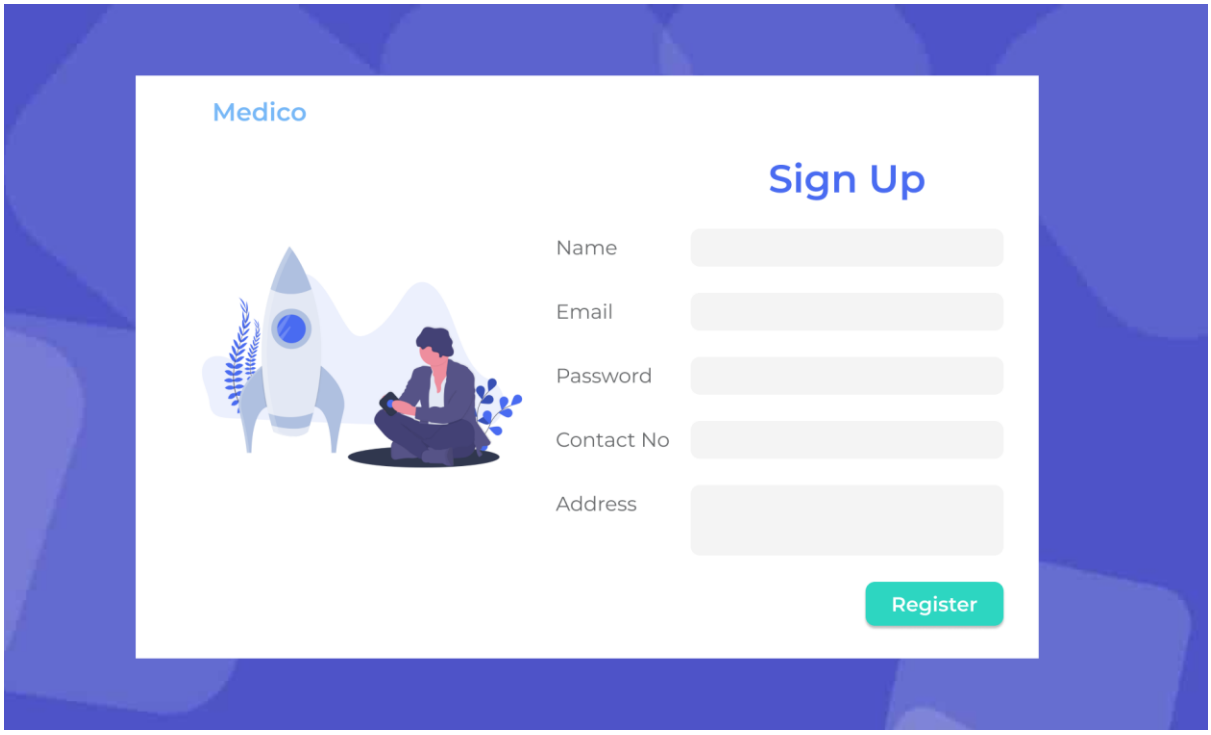

Figure 4.6 : Register Page

4.6.3 Login Page

This module is used to authenticate users. The login page of the Medico application is as follows:

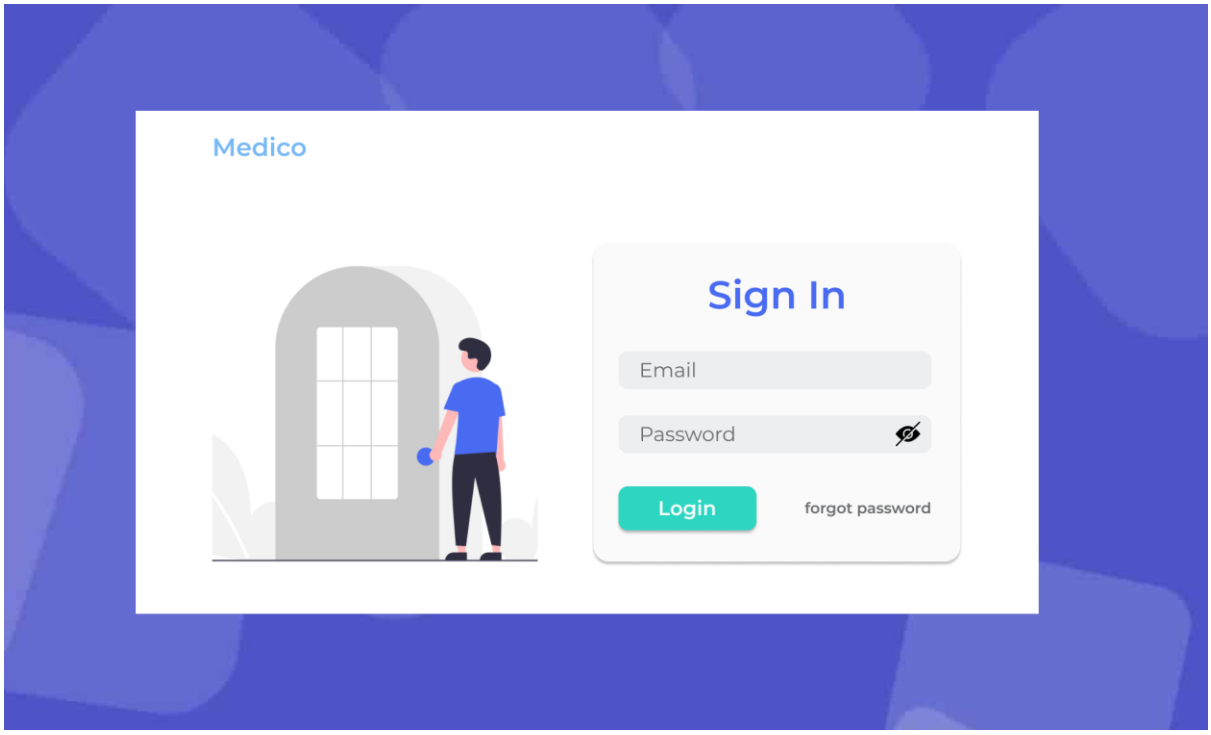

Figure 4.7 : Login Page

4.6.4 Home Page

This is the page that users can see after successfully logging into our system. The home module of the Medico application is as follows-

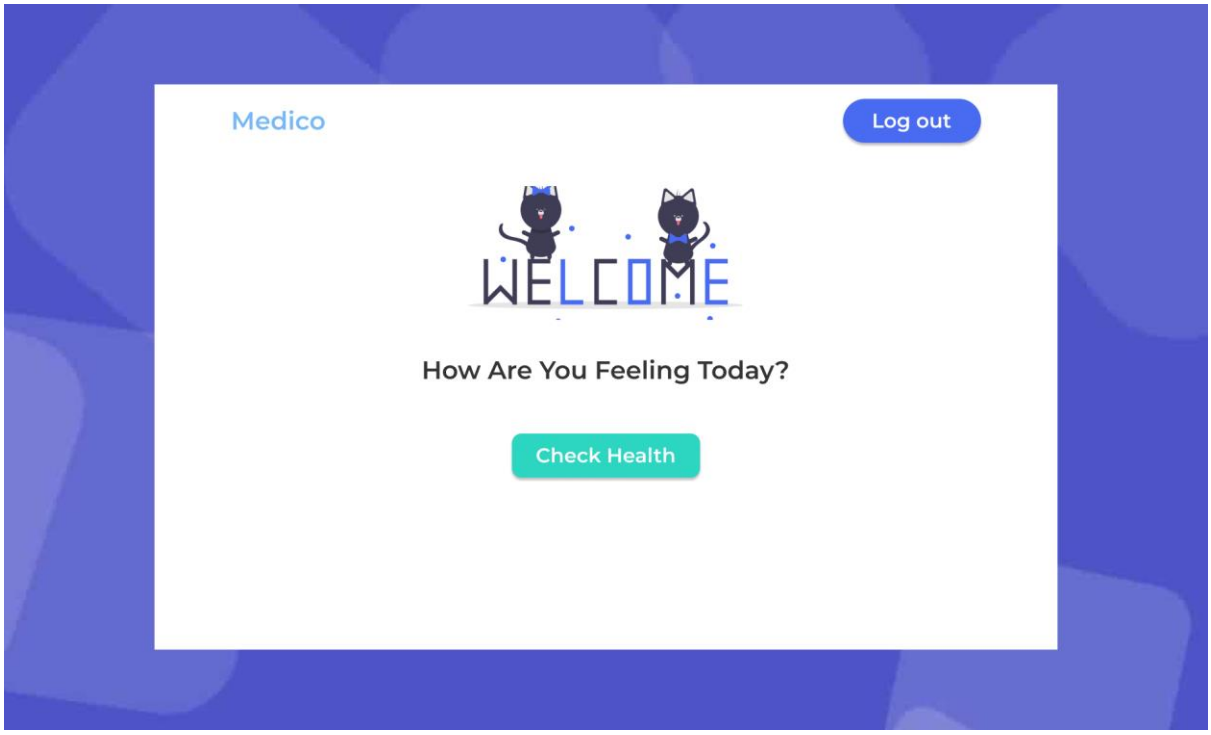

Figure 4.8 : Home Page

4.6.5 Check Health Page

Users can check health conditions using this module. The check health module of the Medico application is as follows-

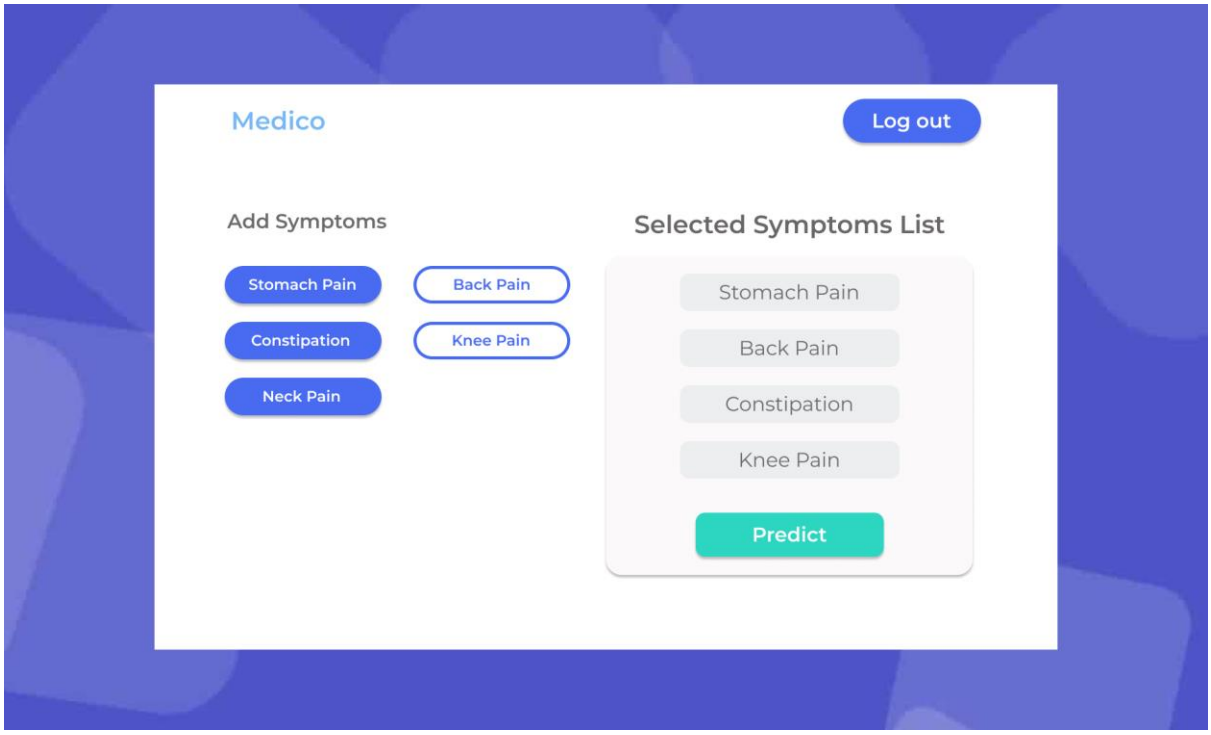

Figure 4.9 : Check Health Page

4.6.6 Result Page

Users can view results based on their symptoms using this module. The result module of the Medico application is as follows-

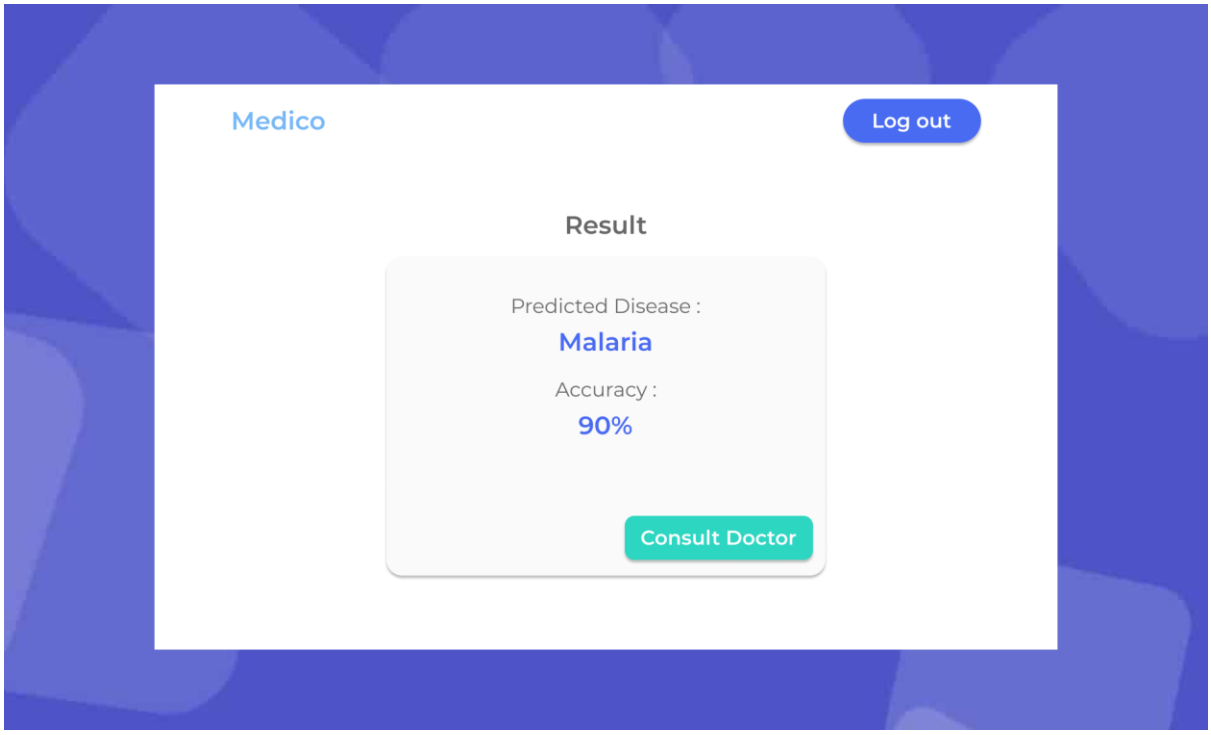

Figure 4.10 : Result Page

4.6.7 Doctors List Page

Users can view recommended doctors list using this page. The doctors list module of the Medico application is as follows-

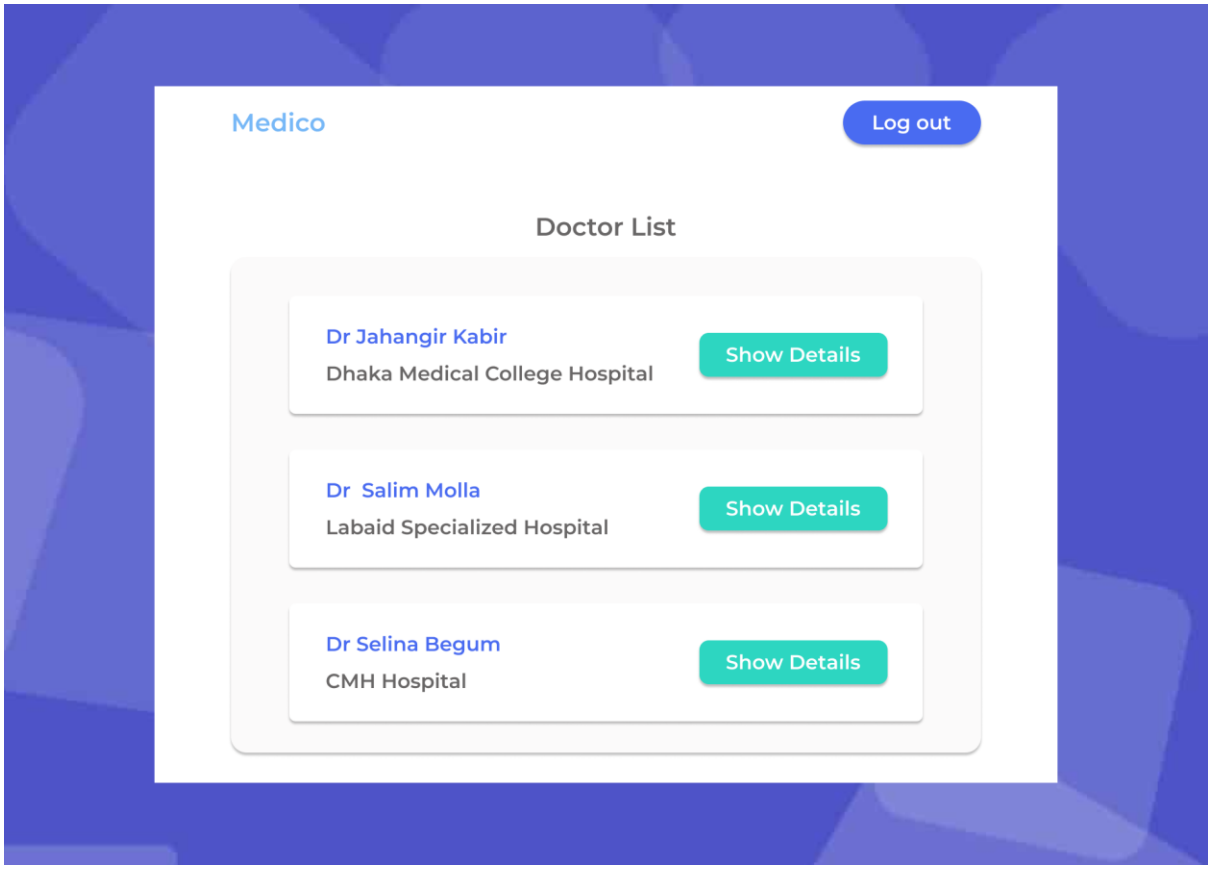

Figure 4.11 : Doctors List Page

4.6.8 Doctor's Profile Page

Users can view the doctor's profile using this module. The doctor's profile module of the Medico application is as follows-

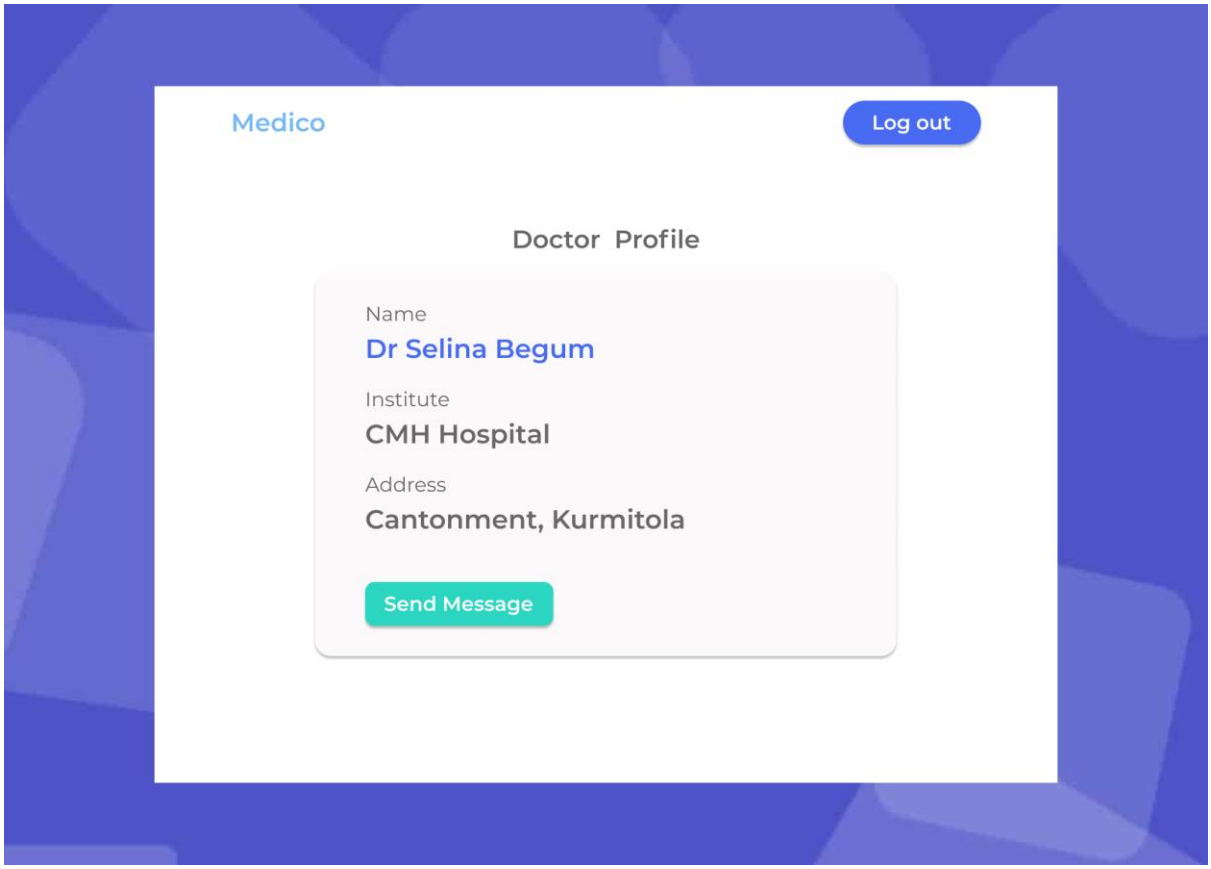

Figure 4.12 : Doctor's Profile Page

4.6.9 Chat Page

Users can send and view messages using this module. The chat module of the Medico application is as follows-

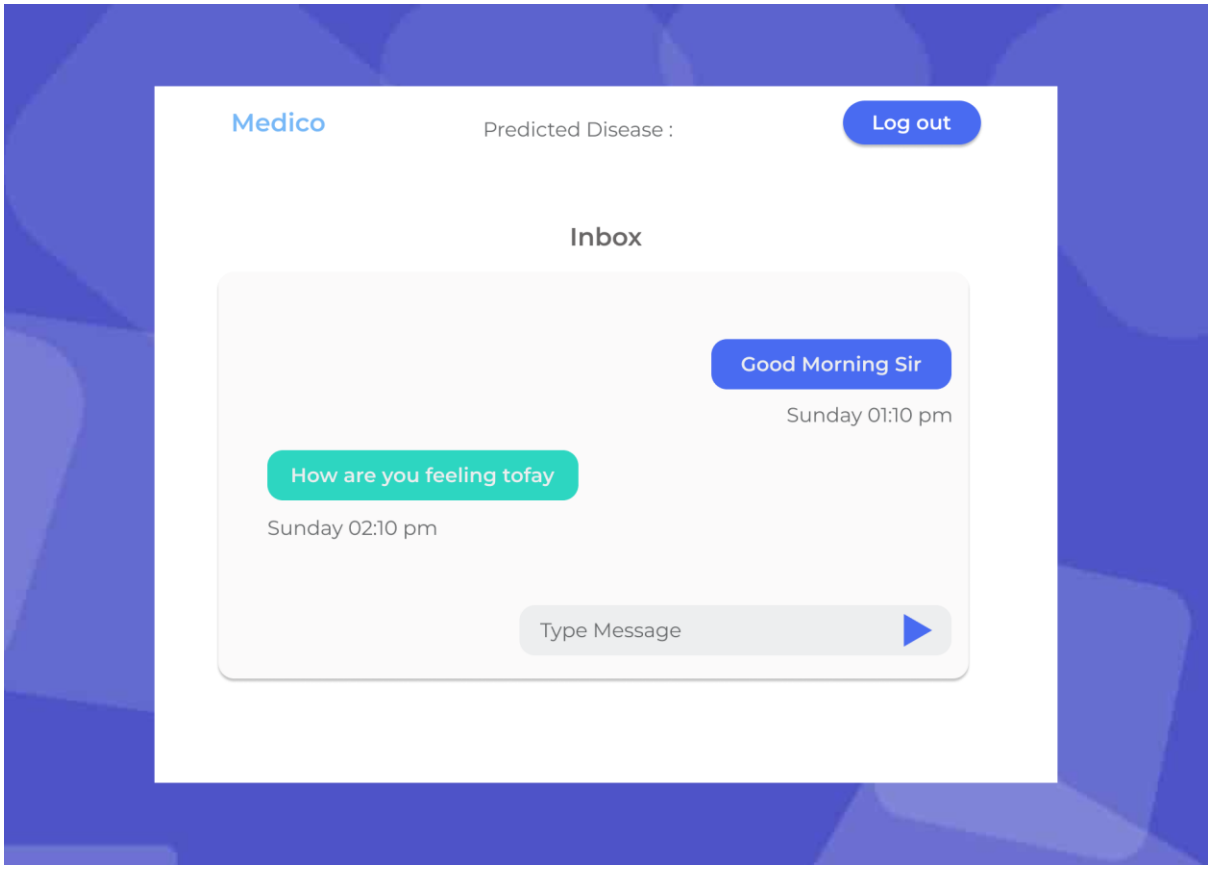

Figure 4.13 : Chat Page

REFERENCES

[1] Introduction to Machine Learning with Python (Andreas C. Muller & Sarah Guido)

[2] https://www.python.org/docs

[3] Machine Learning Algorithms for Classification using Scikit-Learn
https://scikit-learn.org/stable/

[4] Data Sets for human diseaseas
https://www.kaggle.com/

[5] Embedding a Machine Learning Model into a Web Application
https://flask.palletsprojects.com/en/2.0.x/

[6] https://www.postgresql.org/

[7] https://www.w3schools.com/