

Heart Disease Prediction System Using Machine Learning



A project present to the National University in partial fulfilment of the requirements for the degree of B.Sc. (Hon's) in Computer Science & Engineering.

Supervised by:
Nusrhat Jahan Sarker
Lecturer

Department of Computer Science & Engineering
Daffodil Institute of IT (DIIT)

Submitted by:
Md Kamrul Islam
Registration No. 17502005078
Session: 2017-18



Department of Computer Science & Engineering
Daffodil Institute of IT (DIIT)
Under National University (NU) - Dhaka, Bangladesh
Submission Date: 04.09.2023

Approval

This Project titled “Heart Disease Prediction System Using Machine Learning” Is submitted to the Department of Computer Science & Engineering (CSE) of Daffodil Institute of IT (DIIT) Under National University (NU). It has been accepted satisfactory for the partial fulfillment of the requirements for the degree of Bachelor in Computer Science & Engineering (CSE) & approved as to its styles & contents.

.....

Examiner

.....

Examiner

.....

Nusrhat Jahan Sarker
Project Guide (Supervisor)
Lecturer
Department of CSE, DIIT

.....

Md. Imran Hossain
Head
Department of CSE,
Daffodil Institute of IT (DIIT)

Declaration

I hereby declare that the project worked entitled “Heart Disease Prediction System Using Machine Learning” submitted to the degree of BSc. (Hon’s) in Computer Science& Engineering (CSE) is a record of original work done by me. Except as acknowledged in the text and that the material has not been submitted, either in whole or in part, for a degree at this or any other university.

Submitted by:

.....
Md Kamrul Islam
Registration No. – 17502005078
Session – 2017-18

Acknowledgement

My sincere thanks to **Prof. Dr. Mohammed Shakhawat Hossain, Principal, DIIT** who has allowed me to do this project and encouragement given to me.

I would also like to thank my Project Supervisor, **Nusrhat Jahan Sarker**, Lecturer, Department of Computer Science & Engineering, DIIT, for his valuable guidance and support to meet the successful completion of my project.

I express my gratitude to **Md. Imran Hossain**, Head of Department Computer Science & Engineering, DIIT, Dhaka, for his patronage and giving me an opportunity to undertake this Project.

I express my gratitude to **Poly Bhoumik**, Senior Lecturer, DIIT, Dhaka, for having provided us the facilities to do the project successfully.

I also express my gratitude to **Saidur Rahman**, Senior Lecturer, DIIT, Dhaka, for having provided us the facilities to do the project successfully.

I also express my gratitude to **Safrun Nesa Saira**, Lecturer, DIIT, Dhaka, for having provided us the facilities to do the project successfully.

I also express my gratitude to **Mizanur Rahman**, Lecturer, DIIT, Dhaka, for having provided us the facilities to do the project successfully.

I also express my gratitude to **Moumita Akter**, Lecturer, DIIT, Dhaka, for having provided us the facilities to do the project successfully.

I also express my gratitude to **Md. Mushfiqur Rahman**, Lecturer, DIIT, Dhaka, for having provided us the facilities to do the project successfully.

Last but not the least I extend my sincere thanks to my family members and my friends for their constant support throughout this project.

Abstract

Cardiovascular diseases are the most common cause of death worldwide over the last few decades in the developed as well as underdeveloped and developing countries. Early detection of cardiac diseases and continuous supervision of clinicians can reduce the mortality rate. However, it is not possible to monitor patients every day in all cases accurately and consultation of a patient for 24 hours by a doctor is not available since it requires more sapience, time and expertise. In this project, we have developed and researched about models for heart disease prediction through the various heart attributes of patient and detect impending heart disease using Machine learning techniques like backward elimination algorithm, Gradient boosting classifier and the dataset available publicly in Kaggle Website, further evaluating the results using confusion matrix and cross validation. The early prognosis of cardiovascular diseases can aid in making decisions on lifestyle changes in high risk patients and in turn reduce the complications, which can be a great milestone in the field of medicine.

Table of Contents

Approval	ii
Declaration.....	iii
Acknowledgement.....	iv
Abstract.....	v
Table of Contents.....	vi

Contents

Chapter 01: Introduction	01-05
1.1 Introduction.....	02
1.2 Problem Definition.....	02
1.3 Motivation of the Work.....	03
1.4 Objective.....	04
1.5 Goal.....	04
1.6 Need of System.....	04
1.7 Predictable Attribute.....	04
1.8 Input Attributes.....	04
1.9 Business Prospect.....	05
Chapter 02: Proposed System	06-08
2.1 Related Work.....	07
2.2 Limitation of Existing System.....	07
2.3 Our Proposed System.....	08
2.4 Features.....	08
Chapter 03: Requirements and System Analysis	09-14
3.1 Performance Analysis.....	10
3.2 Project Scope.....	11
3.3 Python.....	11
3.4 HTML.....	12
3.5 Cascading Style Sheet (CSS).....	13

3.6 Javascript.....	13
3.7 Django.....	14
Chapter 04: Methodology	15-17
4.1 Our Used Methodology.....	16
4.2 Agile Software Development.....	16
4.3 Advantages of Agile Method.....	16
4.4 Why We Choose Agile Model.....	17
Chapter 05: Working of System	18-31
5.1 System Architecture.....	19
5.2 Machine Learning.....	19
5.2.1 Supervised Learning.....	20
5.2.2 Unsupervised learning.....	20
5.2.3 Reinforcement learning.....	20
5.3 Algorithm.....	21
6.3.1 Support Vector Machine (SVM).....	21
6.3.2 Naive Bayes Algorithm.....	23
6.3.3 Decision Tree.....	24
5.4 Datasets.....	25
5.5 Feature Selection and Reduction.....	26
5.6 Collection of Dataset.....	28
5.7 Pre-Processing of Data.....	28
5.8 Balancing of Data.....	29
5.9 Prediction of Disease.....	30
5.10 Software Requirement.....	31
5.10 Hardware Requirement.....	31
Chapter 06: System Design	32-44
6.1 Conceptual Design.....	33
6.1.1.1 Use Case Diagram between Admin and System.....	33
6.1.1.2 Use Case Diagram between User and System.....	33
6.1.1.3 Sequence Diagram for User and Administrator.....	34

6.2 Data Flow Diagram.....	34
6.2.1 Data Flow Diagram Lv0.....	35
6.2.2 Data Flow Diagram Lv1.....	36
6.2.3 Data Flow Diagram Lv2.....	36
6.3 Work Flow Diagram.....	37
6.4 Activity Diagram.....	38
6.5 Project Flow chart.....	39
6.6 E-R Diagram.....	40
Chapter 07: Implementation	41-45
7.1 Home page.....	42
7.2 Input Attribute Value page.....	42
7.3 All Patients Page.....	42
7.4 Patients and admin Feedback Page.....	43
7.5 View Prediction Result Page.....	43
7.6 All Doctors Page.....	43
7.7 Admin Dashboard Page.....	44
7.8 Result Page.....	44
7.9 User Profile Page.....	44
7.10 Doctor Suggestion Page.....	45
7.11 User Login Page.....	45
7.12 User Registration Page.....	45
Chapter 08: Conclusion	46-48
8.1 Limitation of My System.....	47
8.2 Future Enhancement.....	47
8.3 Conclusion.....	47
8.4 References.....	48
Appendix	49-54

List of Figures

1.2 Problem Definition.....	03
3.1 Confusion Matrix.....	10
3.1.1 Correlation Matrix.....	11
4.2 Agile Software Development Method.....	16
4.3 Advantage of Agile Method.....	17
5.1 System Architecture.....	19
5.3.1 Support Vector Machine (Svm).....	23
5.3.2 Naive Bayes Algorithm.....	24
5.3.3 Decision Tree.....	25
5.4 Dataset Used.....	26
5.6 Collection of Dataset.....	28
5.7 Pre-processing Of Data.....	29
5.8 Balancing Of Data.....	30
5.9 Prediction of Disease.....	30
6.1.1.1 Use Case Diagram between ADMIN and SYSTEM.....	33
6.1.1.2 Use Case Diagram between USER and SYSTEM.....	33
6.1.1.3 Sequence Diagram for User and Administrator.....	34
6.2.1 Data Flow Diagram Level-0.....	35
6.2.2 Data Flow Diagram Level-1.....	36
6.2.3 Data Flow Diagram Level-2.....	36
6.3 Work Flow Diagram.....	37
6.4 Activity Diagram.....	38
6.5 Project Flow chart.....	39
6.6 E-R Diagram.....	40

Chapter 1

Introduction

1.1 Introduction

According to the World Health Organization, every year 12 million deaths occur worldwide due to Heart Disease. Heart disease is one of the biggest causes of morbidity and mortality among the population of the world. Prediction of cardiovascular disease is regarded as one of the most important subjects in the section of data analysis. The load of cardiovascular disease is rapidly increasing all over the world from the past few years. Many researches have been conducted in attempt to pinpoint the most influential factors of heart disease as well as accurately predict the overall risk. Heart Disease is even highlighted as a silent killer which leads to the death of the person without obvious symptoms. The early diagnosis of heart disease plays a vital role in making decisions on lifestyle changes in high-risk patients and in turn reduces the complications.

Machine learning proves to be effective in assisting in making decisions and predictions from the large quantity of data produced by the health care industry. This project aims to predict future Heart Disease by analyzing data of patients which classifies whether they have heart disease or not using machine-learning algorithm. Machine Learning techniques can be a boon in this regard. Even though heart disease can occur in different forms, there is a common set of core risk factors that influence whether someone will ultimately be at risk for heart disease or not. By collecting the data from various sources, classifying them under suitable headings & finally analysing to extract the desired data we can say that this technique can be very well adapted to do the prediction of heart disease. An estimated 17.5 million deaths occur due to cardiovascular diseases worldwide. More than 75% deaths due to cardiovascular diseases occur in the middle-income and low-income countries. Also, 80% of the deaths that occur due to CVDs are because of stroke and heart attack.^[1]

1.2 Problem Definition

The major challenge in heart disease is its detection. There are instruments available which can predict heart disease but either it are expensive or are not efficient to calculate chance of heart disease in human. Early detection of cardiac diseases can decrease the mortality rate and overall complications. However, it is not possible to monitor patients, everyday in all cases accurately and consultation of a patient for 24 hours by a doctor is not available since it requires more sapience, time and expertise. Since we have a good amount of data in today's world, we can use various machine

learning algorithms to analyze the data for hidden patterns. The hidden patterns can be used for health diagnosis in medicinal data.

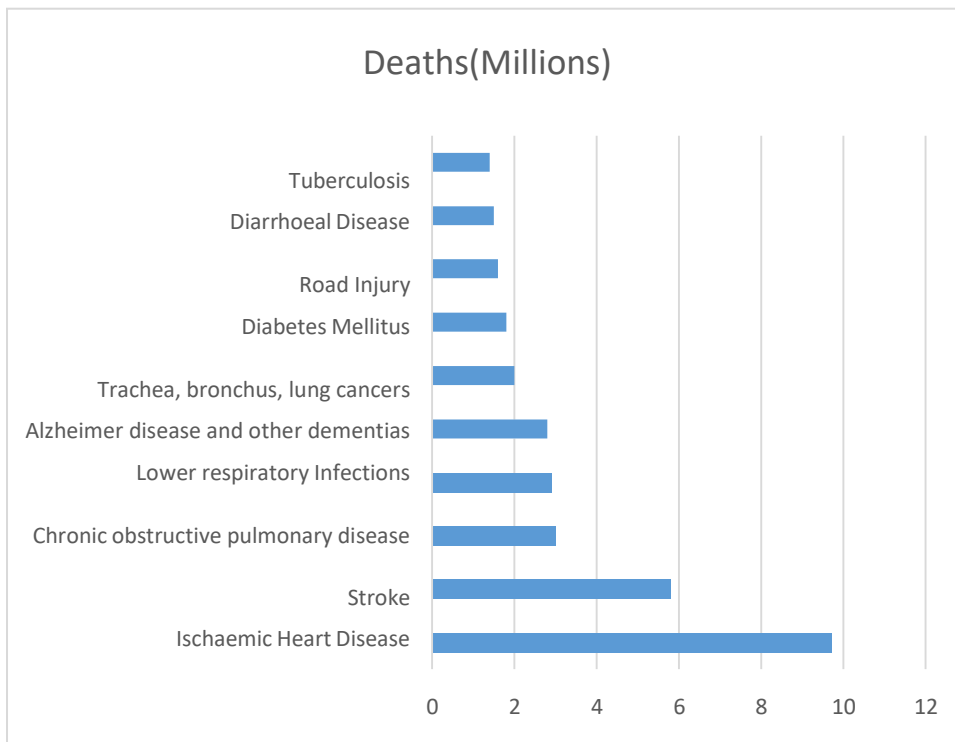


Figure 1.2: Top 10 Global causes of deaths (WHO, 2019)

1.3 Motivation of the Work

The main motivation of doing this research is to present a heart disease prediction model for the prediction of occurrence of heart disease. Further, this research work is aimed towards identifying the best classification algorithm for identifying the possibility of heart disease in a patient. This work is justified by performing a comparative study and analysis using three classification algorithms namely Naïve Bayes, Decision Tree, and Random Forest are used at different levels of evaluations. Although these are commonly used machine learning algorithms, the heart disease prediction is a vital task involving highest possible accuracy. Hence, the three algorithms are evaluated at numerous levels and types of evaluation strategies. This will provide researchers and medical practitioners to establish a better.

1.4 Objective

- This Research Is To Present a Heart Disease Prediction Model For The Prediction Of Occurrence Of Heart Disease.
- Due to lack of resources in the medical field, the prediction of heart disease occasionally may be a problem.
- Utilization of suitable technology support in this regard can prove to be highly beneficial to the medical fraternity and patient.
- Machine learning techniques can be very well adapted to the prediction of heart disease.

1.5 Goal

- User can search for doctor's help at any point of time.
- User can talk about their illness and get instant diagnosis.
- Inform the user about the type of disease or disorder it feels.
- Doctors get more clients online.

1.6 Need of System

- There is always a need of a system that will provide the disease prediction information according to parameter values shared by user.
- This system will help the user to find good doctors and medicines.

1.7 Predictable Attribute

- Diagnosis value 0:<50% diameter narrowing(no heart disease).
- Value 1:>50% diameter narrowing(has heart disease).

1.8 Input Attributes

- Gender.
- Chest Pain Type.
- Restecg.
- Exang.
- CA – number of major vessels colored by floursop .

- Thal.
- Rest Blood Pressure.
- Serum Cholesterol.
- Thalach – maximum heart rate achieved.
- Oldpeak.
- Age in Year.

1.9 Business Prospect

Medical health systems have been concentrating on artificial intelligence techniques for speedy diagnosis. However, the recording of health data in a standard form still requires attention so that machine learning can be more accurate and reliable by considering multiple features. The aim of this study is to develop a general framework for recording diagnostic data in an international standard format to facilitate prediction of disease diagnosis based on symptoms using machine learning algorithms. Efforts were made to ensure error-free data entry by developing a user-friendly interface.

Chapter 2

Proposed System

2.1 Related Work

With growing development in the field of medical science alongside machine learning various experiments and researches has been carried out in these recent years releasing the relevant significant papers. The paper [2] propose heart disease prediction using KStar, J48, SMO, and Bayes Net and Multilayer perceptron using WEKA software. Based on performance from different factor SMO (89% of accuracy) and Bayes Net (87% of accuracy) achieve optimum performance than KStar, Multilayer perceptron and J48 techniques using k-fold cross validation. The accuracy performance achieved by those algorithms are still not satisfactory. So that if the performance of accuracy is improved more to give batter decision to diagnosis disease.

In a research conducted using Cleveland dataset for heart diseases which contains 303 instances and used 10-fold Cross Validation, considering 13 attributes, implementing 4 different algorithms, they concluded Gaussian Naïve Bayes and Random Forest gave the maximum accuracy of 91.2 percent.^[3]

Using the similar dataset of Framingham, Massachusetts, the experiments were carried out using 4 models and were trained and tested with maximum accuracy K Neighbors Classifier: 87%, Support Vector Classifier: 83%, Decision Tree Classifier: 79% and Random Forest Classifier: 84%.^[4]

2.2 Limitation of Existing System

In this system, the input details are obtained from the patient. Then from the user inputs, using ML techniques heart disease is analyzed. Now, the obtained results are compared with the results of existing models within the same domain and found to be improved. The data of heart disease patients collected from the UCI laboratory is used to discover patterns with NN, DT, Support Vector machines SVM, and Naive Bayes. The results are compared for performance and accuracy with these algorithms. The proposed hybrid method returns results of 87% for F-measure, competing with the other existing methods.

- The prediction of cardiovascular disease results is not accurate.
- Data mining techniques do not help to provide effective decision-making.
- Cannot handle enormous datasets for patient records.

2.3 Our Proposed System

The working of the system starts with the collection of data and selecting the important attributes. Then the required data is preprocessed into the required format. The data is then divided into two parts training and testing data. The algorithms are applied and the model is trained using the training data. The accuracy of the system is obtained by testing the system using the testing data. This system is implemented using the following modules.

2.4 Features

- Predict most of the heart diseases which are discovered.
- Determine the condition of heart.
- Analyze the heart.
- Admin and user different page.
- Suggestion doctor by hearth condition.
- Accuracy rate will be high.

Chapter 3

Requirements

&

System Analysis

3.1 Performance Analysis

In this project, various machine learning algorithms like SVM, Naive Bayes, Decision Tree, Random Forest, Logistic Regression, Adaboost, XG-boost are used to predict heart disease. Heart Disease UCI dataset, has a total of 76 attributes, out of those only 14 attributes are considered for the prediction of heart disease. Various attributes of the patient like gender, chest pain type, fasting blood pressure, serum cholesterol, exang, etc are considered for this project. The accuracy for individual algorithms has to measure and whichever algorithm is giving the best accuracy, that is considered for the heart disease prediction. For evaluating the experiment, various evaluation metrics like accuracy, confusion matrix, precision, recall, and f1-score are considered. Accuracy- Accuracy is the ratio of the number of correct predictions to the total number of inputs in the dataset. It is expressed as:

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{FP} + \text{FN} + \text{TN})$$

Confusion Matrix: It gives us a matrix as output and gives the total performance of the system.

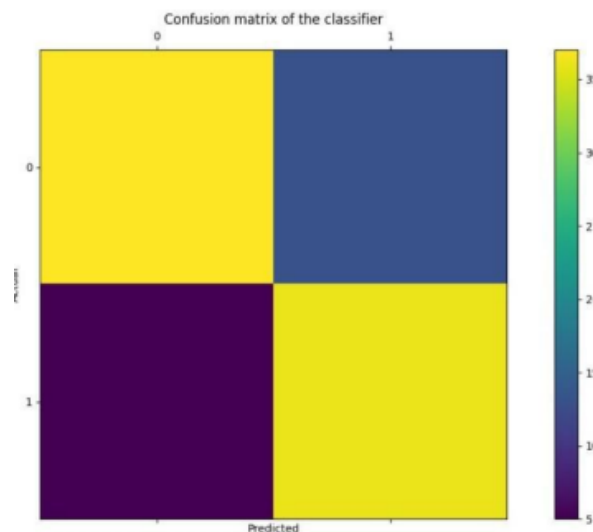


Figure 3.1: Confusion Matrix

Where

TP: True positive

FP: False Positive

FN: False Negative

TN: True Negative

Correlation Matrix: The correlation matrix in machine learning is used for feature selection. It represents dependency between various attributes.



Figure 3.1.1: Correlation Matrix

- **Precision:** It is the ratio of correct positive results to the total number of positive results predicted by the system.
- **Recall:** It is the ratio of correct positive results to the total number of positive results predicted by the system.

3.2 Project Scope

The project has a wide scope, as it is not intended to a particular organization. This project is going to develop generic software, which can be applied by any businesses organization. Moreover it provides facility to its users. Also the software is going to provide a huge amount of summary data.

3.3 Python

Python is a widely used general-purpose, high level programming language. It was initially designed by Guido van Rossum in 1991 and developed by Python Software Foundation. It was mainly developed for emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code. Python is a programming language that lets you work quickly and integrate systems more

efficiently. Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.



3.4 HTML

HTML (Hypertext Markup Language) is the set of markup symbols or codes inserted in a file intended for display on a World Wide Web browser page. The markup tells the Web browser how to display a Web page's words and images for the user. Each individual markup code is referred to as an element (but many people also refer to it as a tag). Some elements come in pairs that indicate when some display effect is to begin and when it is to end.



3.5 Cascading Style Sheet (CSS)

Cascading Style Sheets (CSS) are a collection of rules we use to define and modify web pages. CSS are similar to styles in Word. CSS allow Web designers to have much more control over their pages look and layout. For instance, you could create a style that defines the body text to be Verdana, 10 point. Later on, you may easily change the body text to Times New Roman, 12 point by just changing the rule in the CSS. Instead of having to change the font on each page of your website, all you need to do is redefine the style on the style sheet, and it will instantly change on all of the pages that the style sheet has been applied to. With HTML styles, the font change would be applied to each instance of that font and have to be changed in each spot. CSS can control the placement of text and objects on your pages as well as the look of those objects. HTML information creates the objects (or gives objects meaning), but styles describe how the objects should appear. The HTML gives your page structure, while the CSS creates the “presentation”. An external CSS is really just a text file with a .css extension. These files can be created with Dreamweaver, a CSS editor, or even Notepad. The best practice is to design your web page on paper first so you know where you will want to use styles on your page. Then you can create the styles and apply them to your page.



3.6 Javascript

JavaScript is a programming language commonly used in web development. It was originally developed by Netscape as a means to add dynamic and interactive elements to websites. While JavaScript is influenced by Java, the syntax is more similar to C and is based on ECMAScript, a scripting language developed by Sun Microsystems. JavaScript is a client-side scripting language, which means the source code is processed by the client's web browser rather than on the web server. This means JavaScript

functions can run after a webpage has loaded without COMMUNICATING with the server. For example, a JavaScript function may check a web form before it is submitted to make sure all the required fields have been filled out. The JavaScript code can produce an error message before any information is actually transmitted to the server. Like server-side scripting languages, such as PHP and ASP, JavaScript code can be inserted anywhere within the HTML of a webpage. However, only the output of server-side code is displayed in the HTML, while JavaScript code remains fully visible in the source of the webpage. It can also be referenced in a separate .JS file, which may also be viewed in a browser.

3.7 Django

Django is a web application framework written in Python programming language. It is based on MVT (Model View Template) design pattern. The Django is very demanding due to its rapid development feature. It takes less time to build application after collecting client requirement.

This framework uses a famous tag line: The web framework for perfectionists with deadlines.



Chapter 4

Methodology

4.1 Our Used Methodology

In our project, we will use the “Agile Software Development” model. There are a number of properties in the Agile Model that helps us to build our project “Blockchain-Based Digital Land Registration System”. We will discuss why it is best for our project throughout this chapter.

4.2 Agile Software Development

Agile software development comprises various approaches to software development under which requirements and solutions evolve through the collaborative effort of self-organizing and cross- functional teams and their customers/end users.

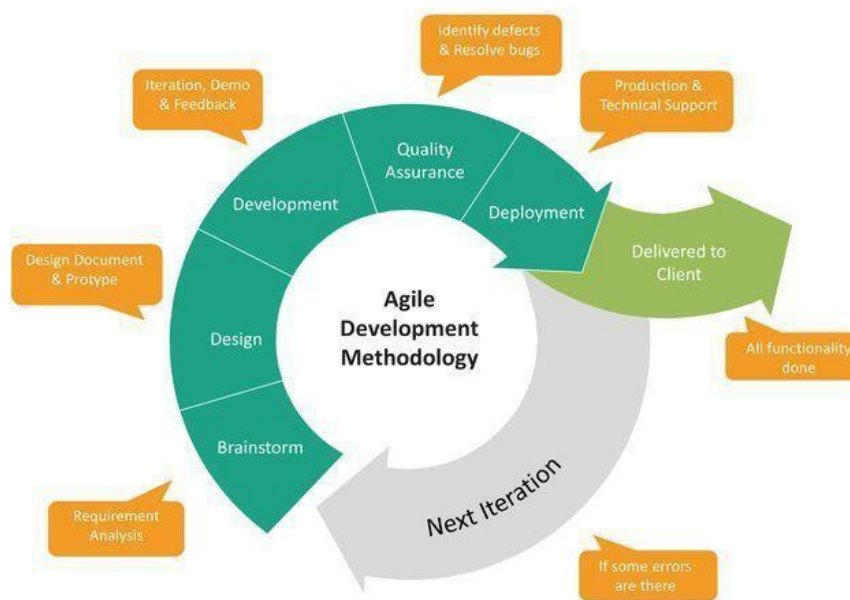


Figure 4.2: Agile Software Development Method

4.3 Advantages of Agile Method

- Customer satisfaction by rapid, continuous delivery of useful software.
- Effective for the dynamic development environment.
- Close, daily cooperation between business people and developers.
- Continuous attention to technical excellence and good design.
- Regular adaptation to changing circumstances.
- Even late changes in requirements are welcomed.

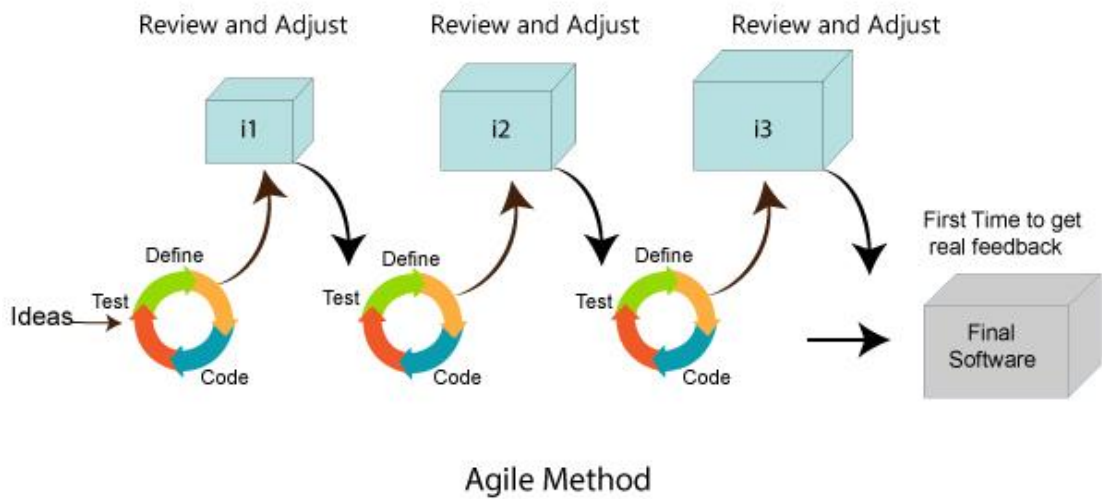


Figure 4.3: Advantage of Agile Method

4.4 Why We Choose Agile Model

Our system is a very sensitive system, and to increase the accuracy of our system we have to redo many parts of the program over and over. Agile model allows us to change and restructure our program at any time. That's why we chose the Agile Software Development Model.

Chapter 5

Working of System

5.1 System Architecture

The system architecture gives an overview of the working of the system.

The working of this system is described as follows:

Dataset collection is collecting data which contains patient details. Attributes selection process selects the useful attributes for the prediction of heart disease. After identifying the available data resources, they are further selected, cleaned, made into the desired form. Different classification techniques as stated will be applied on preprocessed data to predict the accuracy of heart disease. Accuracy measure compares the accuracy of different classifiers.^[5]

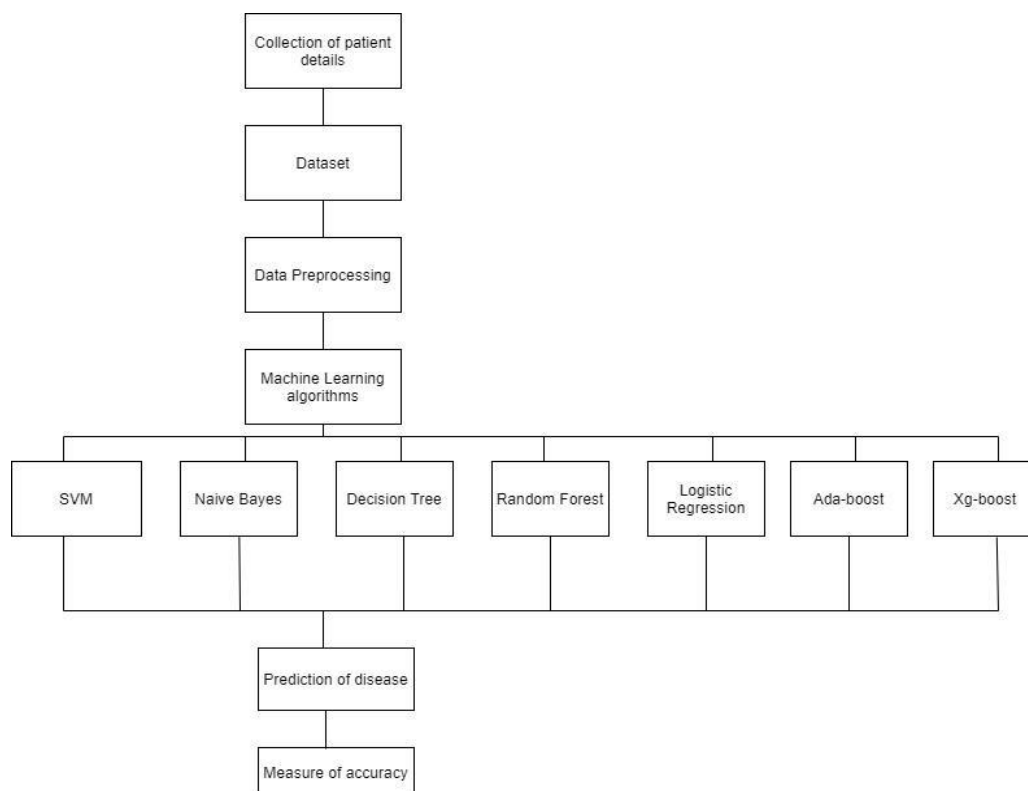


Figure 5.1: SYSTEM ARCHITECTURE

5.2 Machine Learning

In machine learning, classification refers to a predictive modeling problem where a class label is predicted for a given example of input data.

5.2.1 Supervised Learning

Supervised learning is the type of machine learning in which machines are trained using well "labelled" training data, and on the basis of that data, machines predict the output. The labelled data means some input data is already tagged with the correct output.

In supervised learning, the training data provided to the machines work as the supervisor that teaches the machines to predict the output correctly. It applies the same concept as a student learns in the supervision of the teacher.

Supervised learning is a process of providing input data as well as correct output data to the machine learning model. The aim of a supervised learning algorithm is to find a mapping function to map the input variable(x) with the output variable(y).

5.2.2 Unsupervised learning

- Unsupervised learning cannot be directly applied to a regression or classification problem because unlike supervised learning, we have the input data but no corresponding output data. The goal of unsupervised learning is to find the underlying structure of dataset, group that data according to similarities, and represent that dataset in a compressed format.
- Unsupervised learning is helpful for finding useful insights from the data.
- Unsupervised learning is much similar to how a human learns to think by their own experiences, which makes it closer to the real AI.
- Unsupervised learning works on unlabeled and uncategorized data which make unsupervised learning more important.
- In real-world, we do not always have input data with the corresponding output so to solve such cases, we need unsupervised learning.
- Reinforcement learning

5.2.3 Reinforcement learning

Reinforcement learning is an area of Machine Learning. It is about taking suitable action to maximize reward in a particular situation. It is employed by various software and machines to find the best possible behaviour or path it should take in a specific situation. Reinforcement learning differs from supervised learning in a way that in

supervised learning the training data has the answer key with it so the model is trained with the correct answer itself whereas in reinforcement learning, there is no answer but the reinforcement agent decides what to do to perform the given task. In the absence of a training dataset, it is bound to learn from its experience.

5.3 Algorithm

5.3.1 Support Vector Machine (SVM):

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane. SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called support vectors, and hence the algorithm is termed as Support Vector Machine. Support vector machines (SVMs) are powerful yet flexible supervised machine learning algorithms which are used both for classification and regression. But generally, they are used in classification problems. In the 1960s, SVMs were first introduced but later they got refined in 1990. SVMs have their unique way of implementation as compared to other machine learning algorithms. Lately, they are extremely popular because of their ability to handle multiple continuous and categorical variables.

The followings are important concepts in SVM -

- Support Vectors - Data Points that are closest to the hyperplane are called support vectors. Separating line will be defined with the help of these data points.
- Hyperplane - As we can see in the above diagram, it is a decision plane or space which is divided between a set of objects having different classes.

- Margin - It may be defined as the gap between two lines on the closest data points of different classes. It can be calculated as the perpendicular distance from the line to the

Types of SVM:

SVM can be of two types:

- Linear SVM: Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.
- Non-linear SVM: Non-Linear SVM is used for non-linearly separated data, which means if a dataset cannot be classified by using a straight line, then such data is termed as non-linear data and classifier used is called as Non-linear SVM classifier.

The advantages of support vector machines are:

- Effective in high dimensional spaces.
- Still effective in cases where the number of dimensions is greater than the number of samples.
- Uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.
- Versatile: different kernel functions can be specified for the decision function. Common kernels are provided, but it is also possible to specify custom kernels.

The disadvantages of support vector machines include:

- If the number of features is much greater than the number of samples, avoid over-fitting in choosing Kernel functions and regularization term is crucial.
- SVMs do not directly provide probability estimates, these are calculated using an expensive five-fold cross-validation.

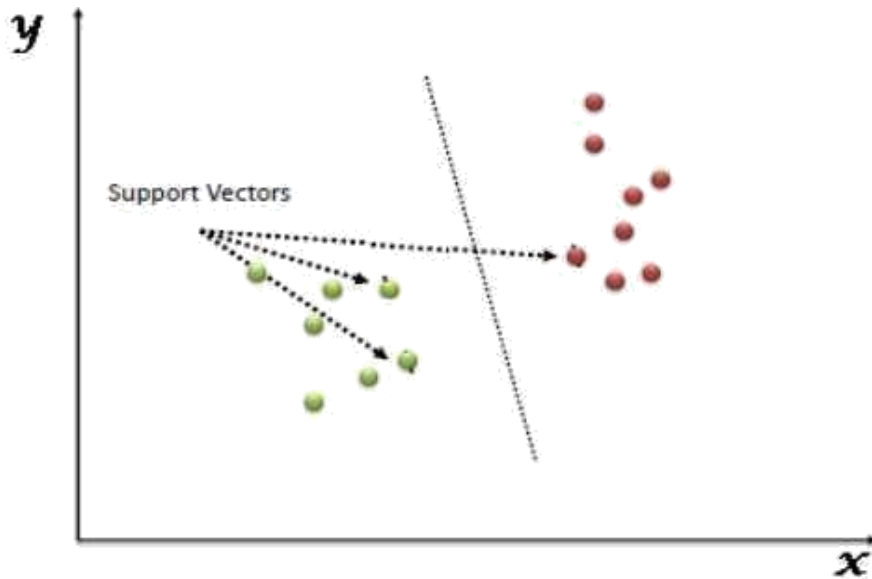


Figure 5.3.1: Support Vector Machine

5.3.2 Naive Bayes Algorithm:

It is a machine learning technique that works on the strategy of the Bayes' Theorem. It basically assumes that there would be no attributes dependent on each other. It is a group of algorithms that have a common principle that every feature is independent of the other. Bayes' Theorem tells us the probability of an event that will occur when another event has already occurred. The

mathematical equation is:

$$\mathbf{Probability(a|z) = (Probability(z|a) * Probability(a)) / Probability(z)}$$

Where

- **Probability(a|z):** Gives us the probability of a(the hypothesis)gives the data is.
- **Probability (z|a):** Gives the probability of the data when the hypothesis is true.
- **Probability (a):** Regardless of the data, the hypothesis is said to be true.
- **Probability (d):** Regardless if the data, the probability of the hypothesis is given.

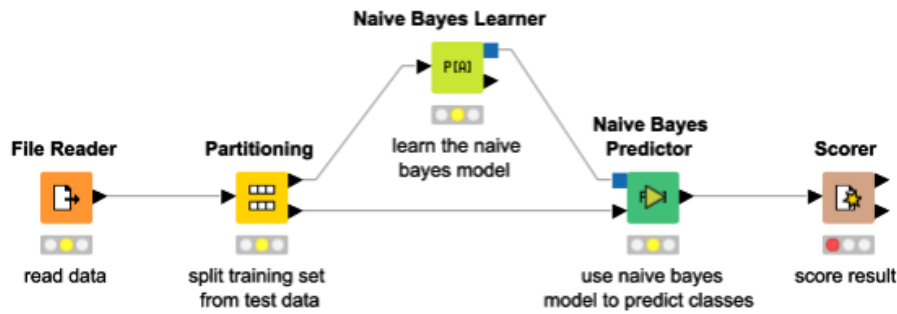


Figure 5.3.2: Naive Bayes Algorithm

5.3.3 Decision Tree:

Decision trees are treelike structures that are used to manage large datasets. They are often depicted as flowcharts, with outer branches representing the results and inner nodes representing the properties of the dataset. Decision trees are popular because they are efficient, reliable, and easy to understand. The projected class label for a decision tree originates from the tree's root. The following steps in the tree are decided by comparing the value of the root attribute with the information in the record. Following a jump on the next node, the matching branch is followed to the value shown by the comparison result. Entropy changes when training examples are divided into smaller groups using a decision tree node. The measurement of this change in entropy is information gain. An accuracy of 73.0% has been achieved by the decision tree.^[6] In a research by,^[7] 72.77% accuracy was achieved by the decision tree classifier. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches. The decisions or the test are performed on the basis of features of the given dataset. It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions. It is called a Decision Tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure. In order to build a tree, we use the CART algorithm, which stands for Classification and Regression Tree algorithm. A Decision Tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees

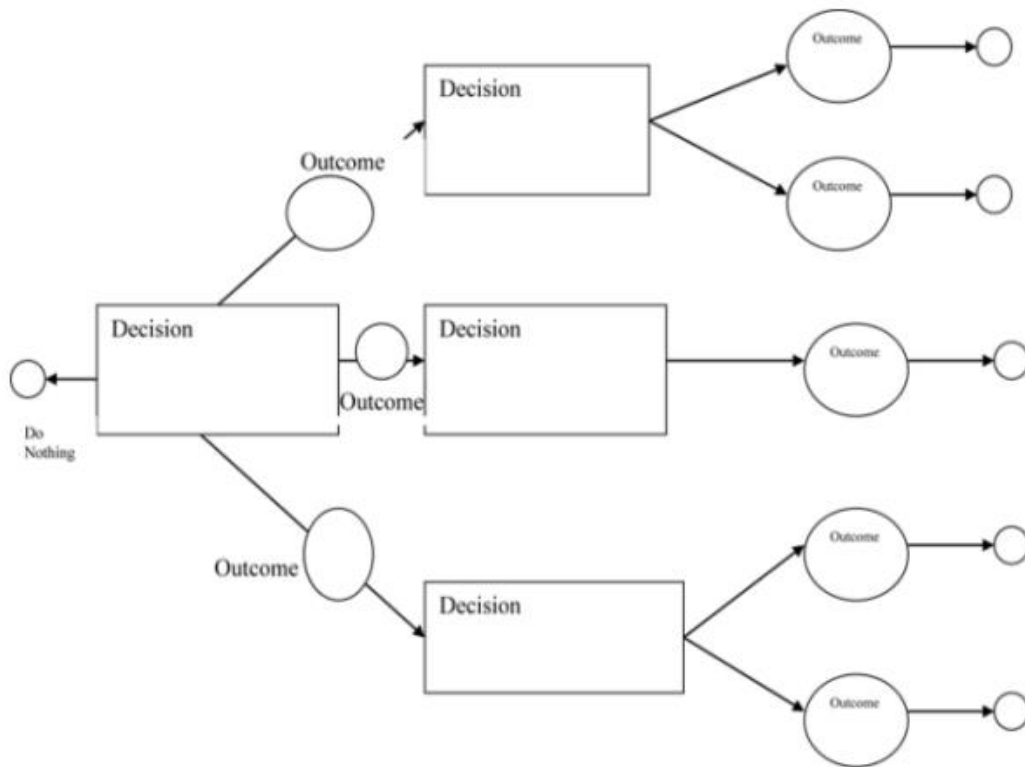


Figure 5.3.3: Decision Tree

5.4 Datasets

The dataset is publicly available on the Kaggle Website at.^[8] which is from an ongoing cardiovascular study on residents of the town of Framingham, Massachusetts. It provides patient information which includes over 4000 records and 14 attributes. The attributes include: age, sex, chest pain type, resting blood pressure, serum cholesterol, fasting, sugar blood, resting electrocardiographic results, maximum heart rate, exercise induced angina, ST depression induced by exercise, slope of the peak exercise, number of major vessels, and target ranging from 0 to 2, where 0 is absence of heart disease. The data set is in csv (Comma Separated Value) format which is further prepared to data frame as supported by pandas library in python.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target							
2	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1							
3	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1							
4	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1							
5	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1							
6	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1							
7	57	1	0	140	192	0	1	148	0	0.4	1	0	1	1							
8	56	0	1	140	294	0	0	153	0	1.3	1	0	2	1							
9	44	1	1	120	263	0	1	173	0	0	2	0	3	1							
10	52	1	2	172	199	1	1	162	0	0.5	2	0	3	1							
11	57	1	2	150	168	0	1	174	0	1.6	2	0	2	1							
12	54	1	0	140	239	0	1	160	0	1.2	2	0	2	1							
13	48	0	2	130	275	0	1	139	0	0.2	2	0	2	1							
14	49	1	1	130	266	0	1	171	0	0.6	2	0	2	1							
15	64	1	3	110	211	0	0	144	1	1.8	1	0	2	1							
16	58	0	3	150	283	1	0	162	0	1	2	0	2	1							
17	50	0	2	120	219	0	1	158	0	1.6	1	0	2	1							
18	58	0	2	120	340	0	1	172	0	0	2	0	2	1							
19	66	0	3	150	226	0	1	114	0	2.6	0	0	2	1							
20	43	1	0	150	247	0	1	171	0	1.5	2	0	2	1							
21	69	0	3	140	239	0	1	151	0	1.8	2	2	2	1							
22	59	1	0	135	234	0	1	161	0	0.5	1	0	3	1							
23	44	1	2	130	233	0	1	179	1	0.4	2	0	2	1							
24	42	1	0	140	226	0	1	178	0	0	2	0	2	1							
25	61	1	2	150	243	1	1	137	1	1	1	0	2	1							

Figure 5.4: Original Dataset Snapshot

5.5 Feature Selection and Reduction

We propose the use of binning as a method for converting continuous input, such as age, into categorical input in order to improve the performance and interpretability of classification algorithms. By categorizing continuous input into distinct groups or bins, the algorithm is able to make distinctions between different classes of data based on specific values of the input variables. For instance, if the input variable is “Age Group” and the possible values are “Young”, “Middle-aged”, and “Elderly”, a classification algorithm can use this information to separate the data into different classes or categories based on the age group of the individuals in the dataset.^[9]

Additionally, converting continuous input into categorical input through binning can also aid in the interpretability of the results, as it is easier to understand and interpret the relationship between the input variables and the output classes. On the other hand, continuous input, such as numerical values, can be more difficult to use in classification algorithms as the algorithm may have to make assumptions about where to draw boundaries between different classes or categories.^[10]

In this study, we applied the method of binning to the attribute of age in a dataset of patients. The age of patients was initially given in days, but for better analysis and prediction, it was converted to years by dividing it by 365. The age data were then divided into bins of 5-year intervals, ranging from 0–20 to 95–100. The minimum age in the dataset is 30 years, and the maximum is 65, so the bin 30–35 is labeled as 0, while the last bin 60–65 is marked as 6.

Furthermore, other attributes with continuous values, such as height, weight, ap_hi, and ap_lo, were also converted into categorical values. The results of this study demonstrate that converting continuous input into categorical input through binning can improve the performance and interpretability of classification algorithms.

In this comprehensive study of US individuals who were unencumbered by clinical CVD at the outset, the participants had a high lifetime risk for developing CVD, and even higher for those who were overweight or obese. In comparison with those with a normal BMI, obese people were shown to have an earlier beginning of incident CVD, a larger percentage of life spent with CVD morbidity (unhealthy life years), and a lower overall survival rate.^[11] This suggests that the attributes of height and weight can be converted to body mass index (BMI), which could improve the performance of our heart disease prediction model. The BMI values are then converted into categorical values for further analysis.

$$\text{BMI} = \text{weight (kg/lb)} / \text{height}^2 \text{ (m}^2\text{/in}^2\text{)}$$

The average blood pressure a person has during a single cardiac cycle is known as mean arterial pressure (MAP) in medicine. MAP is a measure of peripheral resistance and cardiac output, and has been shown to be linked to significant CVD events in the ADVANCE study.^[12,13] In a research including people with type 2 diabetes, it was shown that for every 13 mmHg rise in MAP, the risk of CVD rose by 13%. Additionally, if MAP raises the risk of CVD in people with type 2 diabetes, it should also result in a higher number of CVD hospitalizations.^[13] These findings suggest a direct relationship between MAP and CVD.

5.6 Collection of Dataset

Initially, we collect a dataset for our heart disease prediction system. After the collection of the dataset, we split the dataset into training data and testing data. The training dataset

is used for prediction model learning and testing data is used for evaluating the prediction model. For this project, 70% of training data is used and 30% of data is used for testing. The dataset used for this project is Heart Disease UCI. The dataset consists of 76 attributes; out of which, 14 attributes are used for the system.

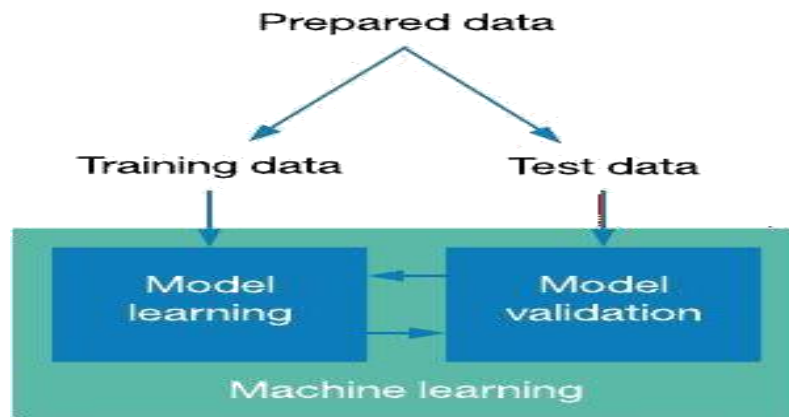


Figure 5.6: Collection of Data

5.7 Pre-Processing of Data

Data pre-processing is an important step for the creation of a machine learning model. Initially, data may not be clean or in the required format for the model which can cause misleading outcomes. In pre-processing of data, we transform data into our required format. It is used to deal with noises, duplicates, and missing values of the dataset. Data pre-processing has the activities like importing datasets, splitting datasets, attribute scaling, etc. Preprocessing of data is required for improving the accuracy of the model.

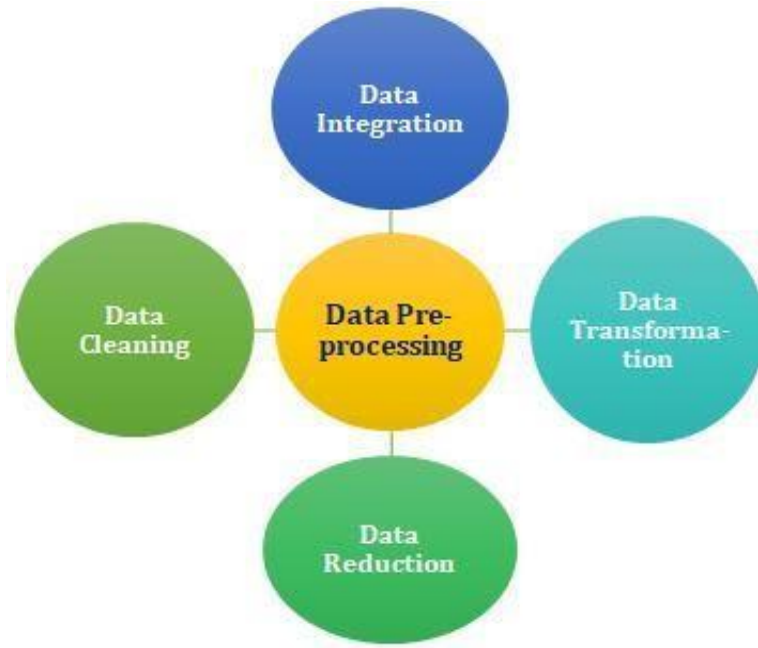


Figure 5.7: Data Pre-processing

5.8 Balancing of Data

Imbalanced datasets can be balanced in two ways. They are Under Sampling and Over Sampling

- **Under Sampling:**

In Under Sampling, dataset balance is done by the reduction of the size of the ample class. This process is considered when the amount of data is adequate.

- **Over Sampling:**

In Over Sampling, dataset balance is done by increasing the size of the scarce samples. This process is considered when the amount of data is inadequate.

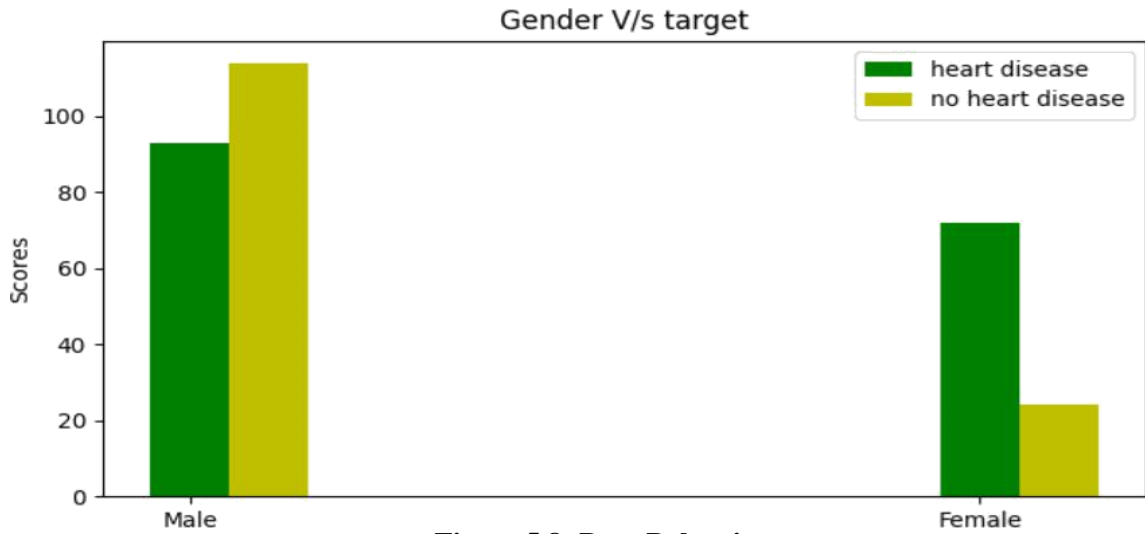


Figure 5.8: Data Balancing

5.9 Prediction of Disease

Various machine learning algorithms like SVM, Naive Bayes, Decision Tree, Random Tree, Logistic Regression, Ada-boost, Xg-boost are used for classification. Comparative analysis is performed among algorithms and the algorithm that gives the highest accuracy is used for heart disease prediction.

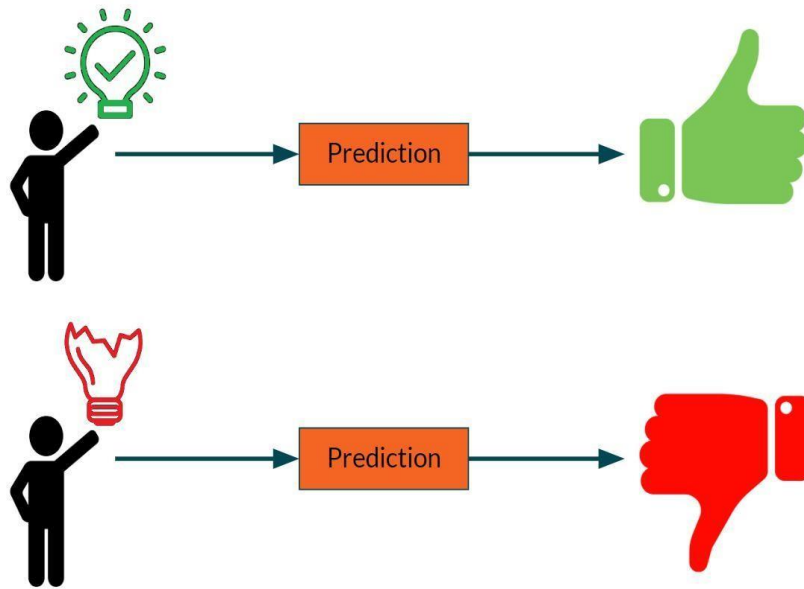


Figure 5.9: Prediction of Disease

5.10 Software Requirement

- Technology: Python Django.
- IDE: Pycharm /Atom.
- Client Side Technologies: HTML, CSS, JavaScript, Bootstrap.
- Server Side Technologies: Python.
- Data Base Server: Sqlite.
- Operating System: Microsoft Windows/Linux

5.11 Hardware Requirement

- Processor: Pentium-III (or) Higher.
- Ram: 64MB (or) Higher.
- Hard disk: 80GB (or) Higher.

Chapter 6

System Design

6.1 Conceptual Design

6.1.1 Use Case Diagram

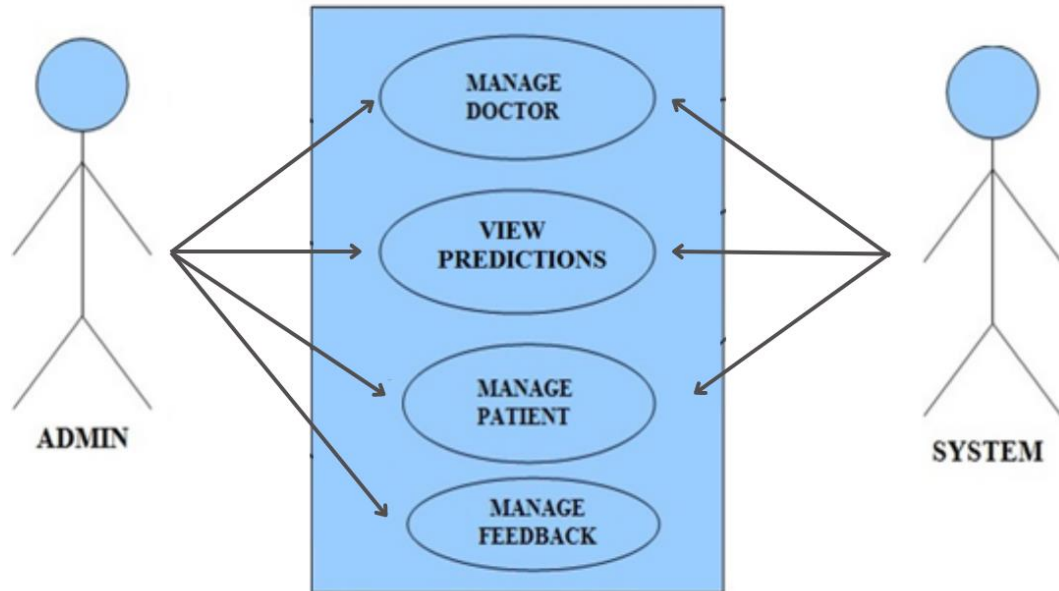


Figure 6.1.1.1: Use Case Diagram between ADMIN and SYSTEM

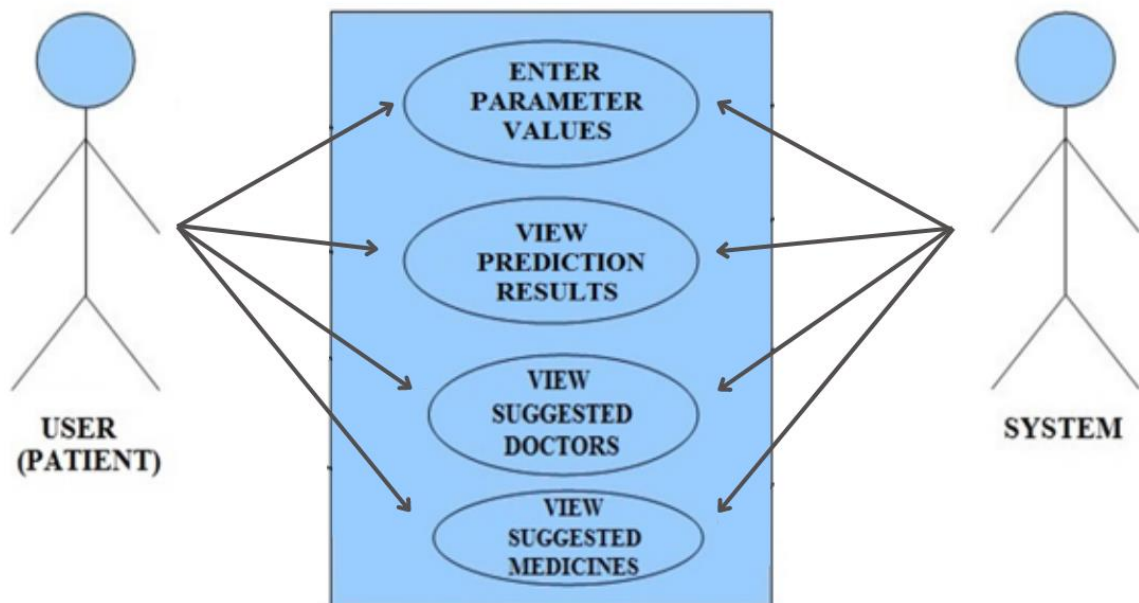


Figure 6.1.1.2: Use Case Diagram between USER and SYSTEM

6.1.1.3 Sequence Diagram for User and Administrator

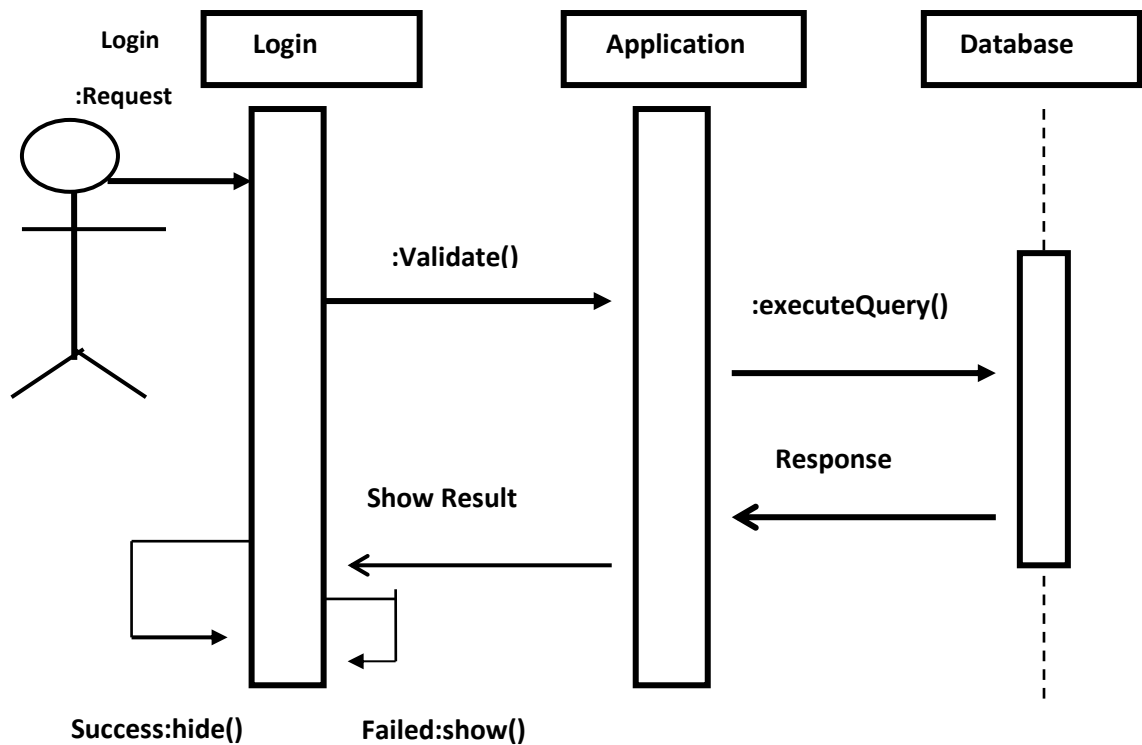


Figure 6.1.1.3: Sequence Diagram for User and Administrator

6.2 Data Flow Diagram

A Data Flow Diagram (DFD) is a graphical representation of the "flow" of data through an Information System. A data flow diagram can also be used for the visualization of Data Processing. It is common practice for a designer to draw a context-level DFD first which shows the interaction between the system and outside entities. This context-level DFD is then "exploded" to show more detail of the system being modeled. A DFD represents flow of data through a system. Data flow diagrams are commonly used during problem analysis. It views a system as a function that transforms the input into desired output. A DFD shows movement of data through the different transformations or processes in the system. Dataflow diagrams can be used to provide the end user with a physical idea of where the data

they input ultimately has an effect upon the structure of the whole system from order to dispatch to restock how any system is developed can be determined through a dataflow diagram. The appropriate register saved in database and maintained by appropriate authorities.

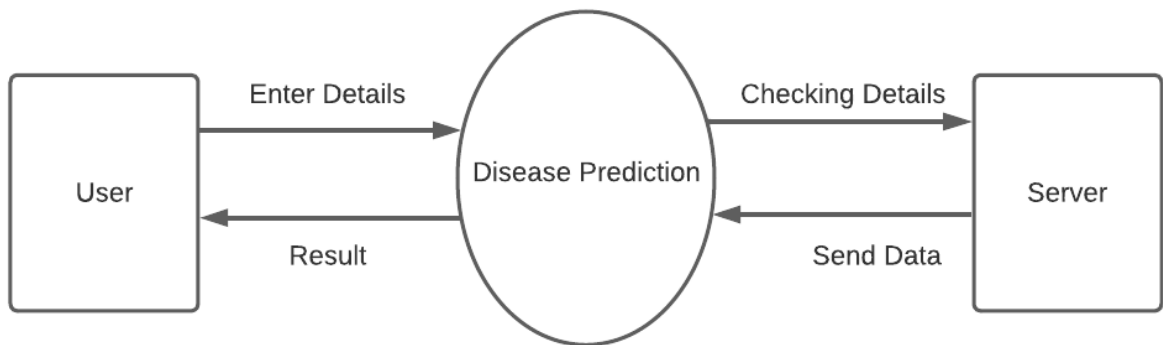


Figure 6.2.1: DFD L0

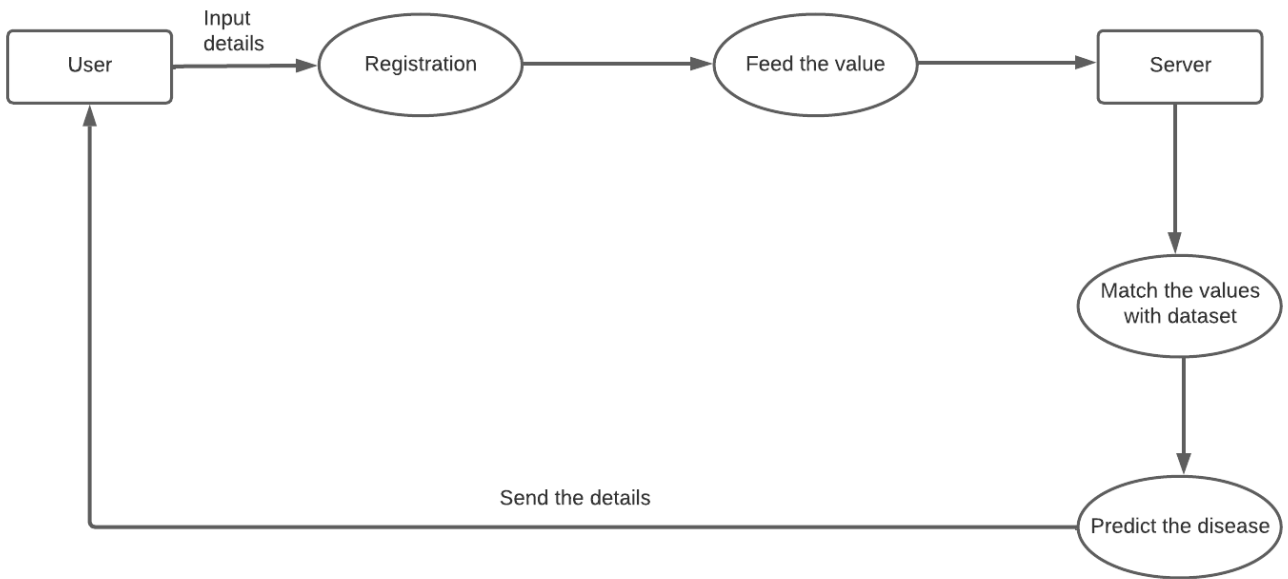


Figure 6.2.2: DFD L1

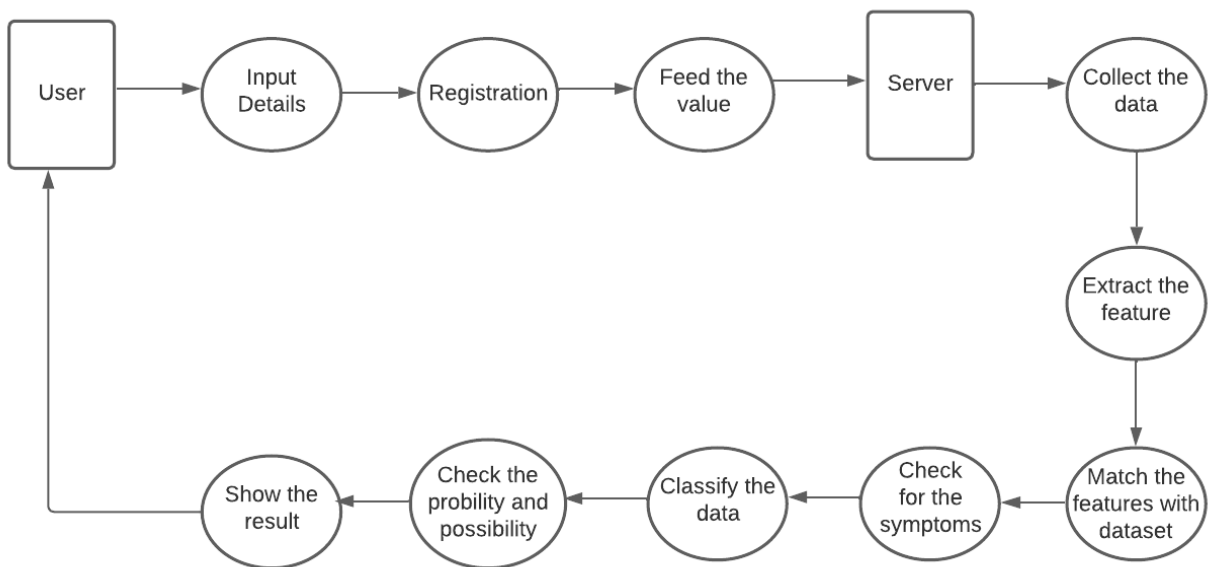


Figure 6.2.3: DFD L2

6.3 Work Flow Diagram

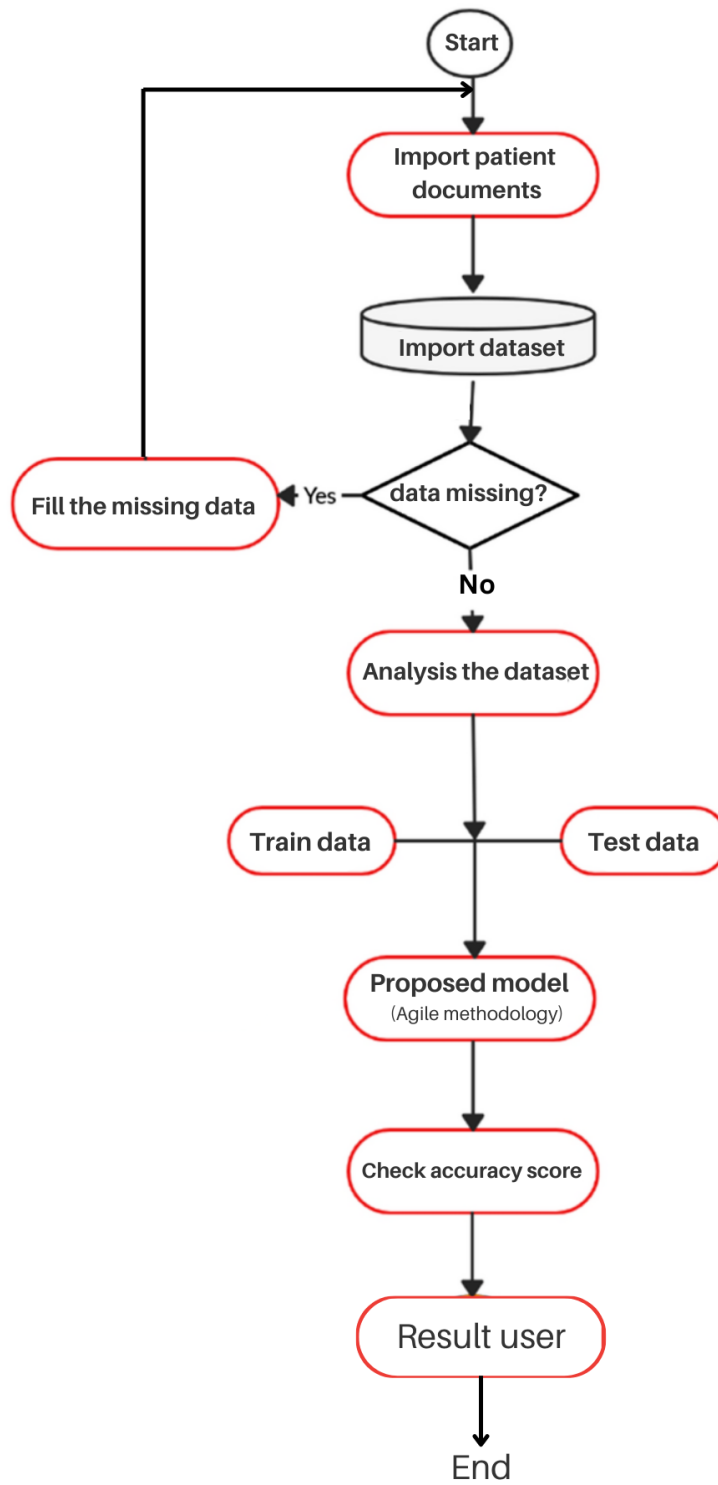


Figure 6.3: Work flow diagram

6.4 Activity Diagram

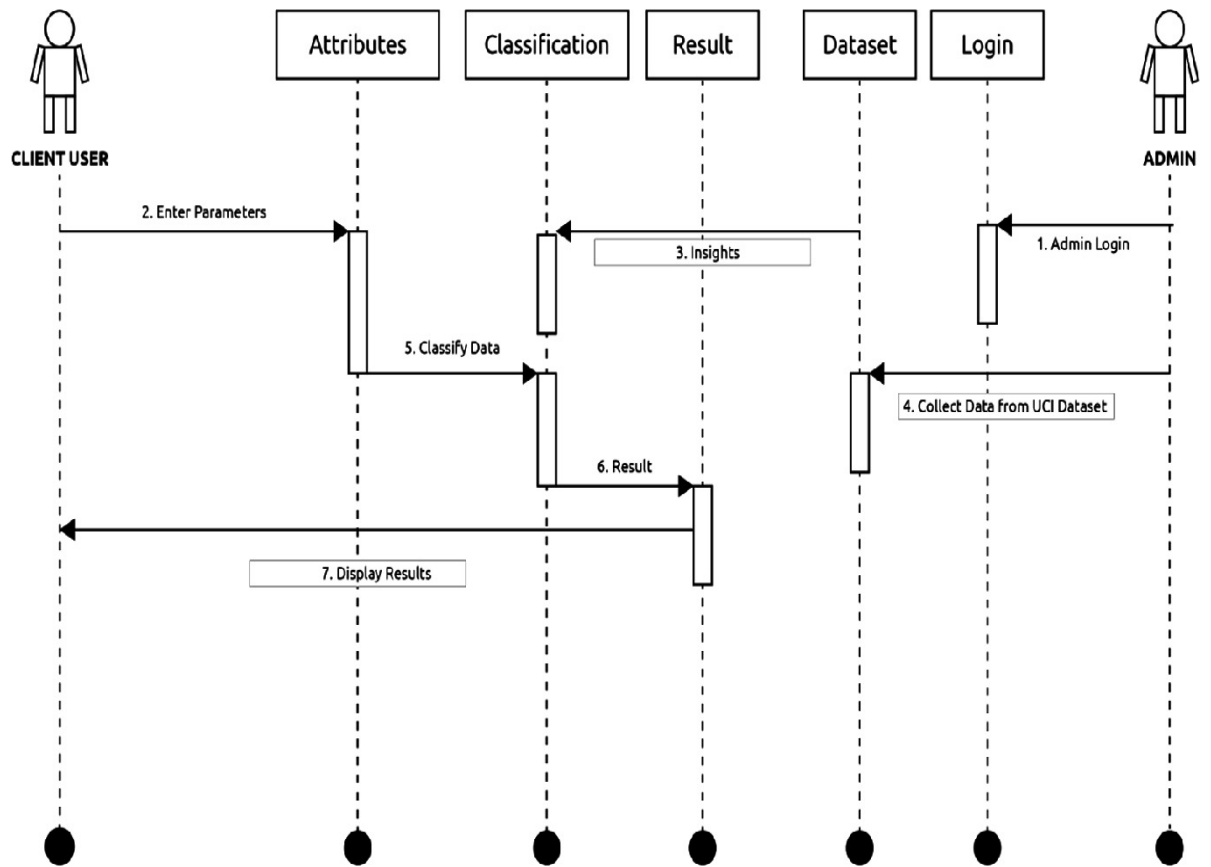


Figure 6.4: Activity Diagram

6.5 Project Flow chart

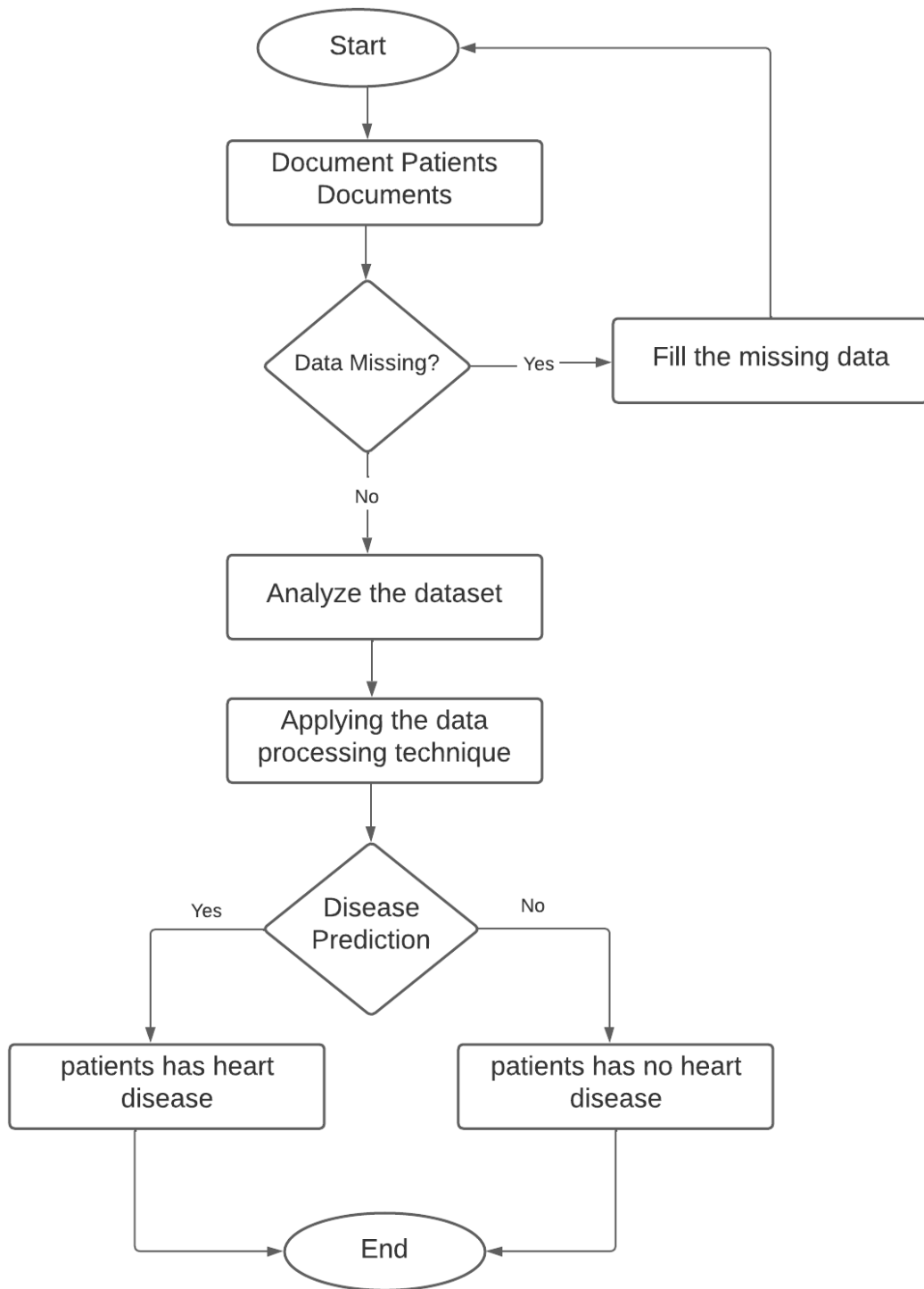


Figure 6.5: Flowchart

6.6 E-R Diagram

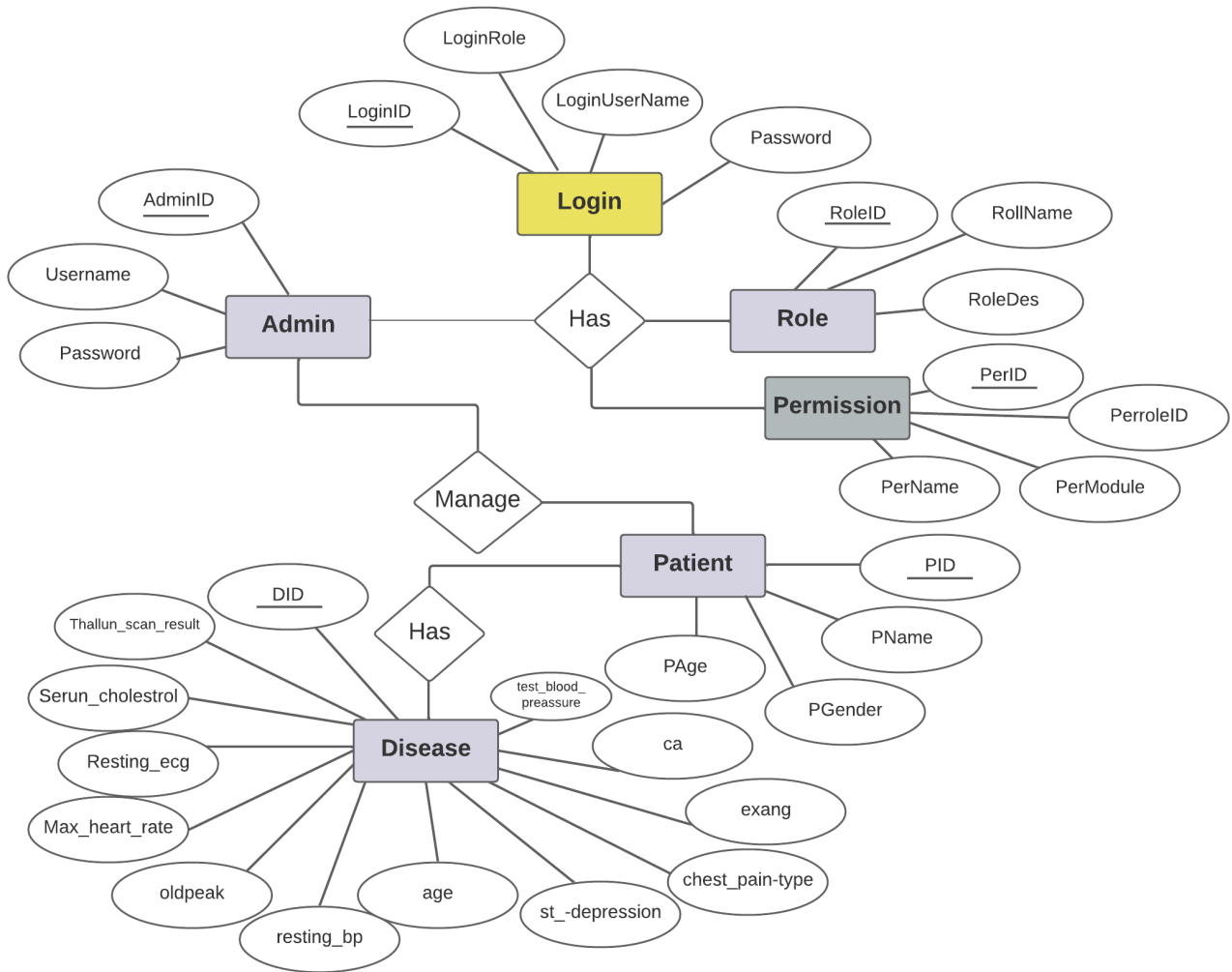
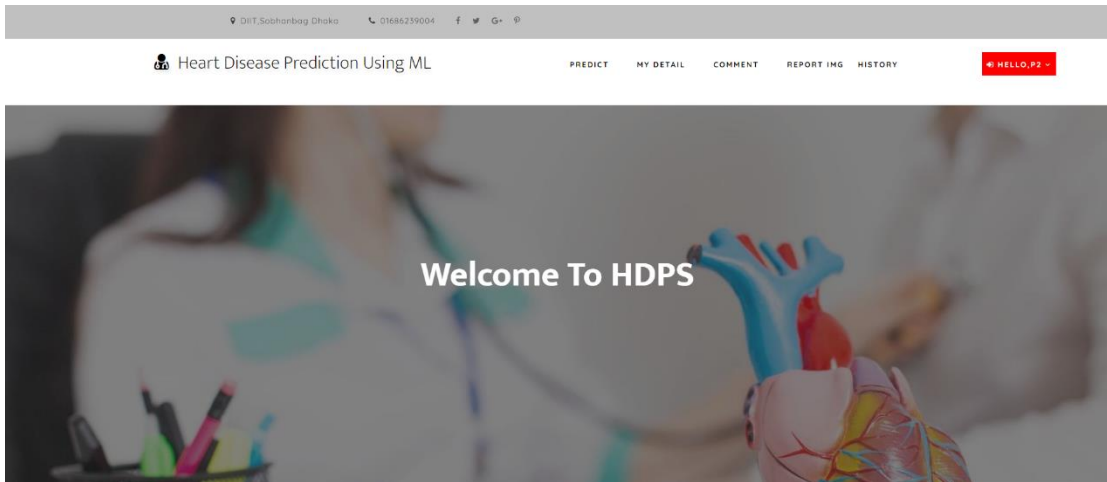


Figure 6.6: E-R Diagram

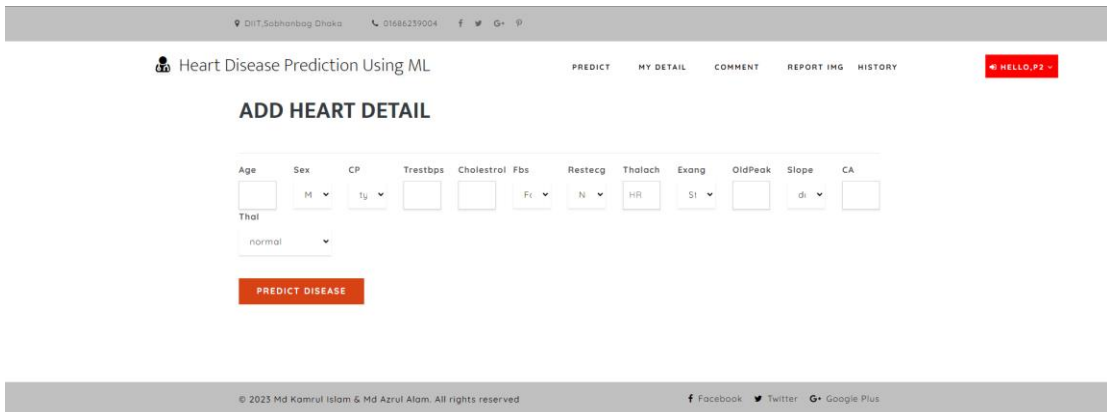
Chapter 7

Implementation

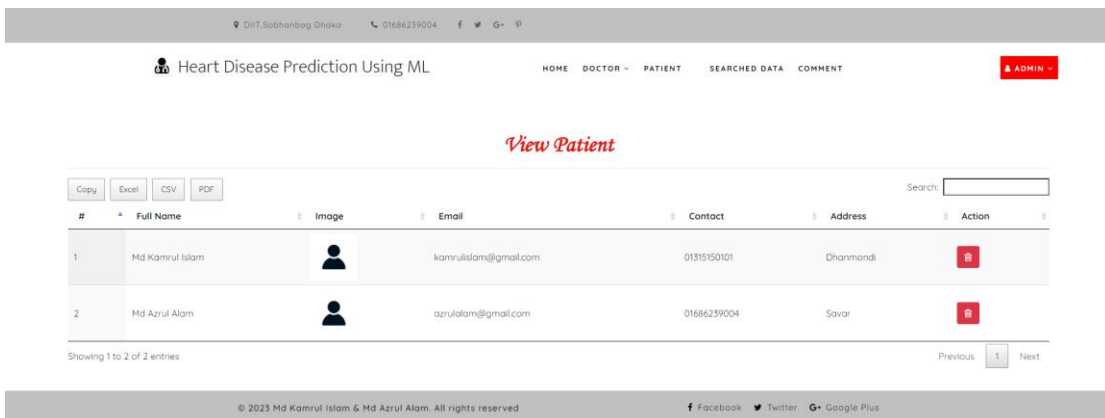
7.1 Home Page:



7.2 Input Attribute Value page:



7.3 All Patients Page:



7.4 Patients and admin Feedback Page:

© DIT,Sobhanbag Dhaka 01686239004 f w G+ P

Heart Disease Prediction Using ML HOME DOCTOR PATIENT SEARCHED DATA COMMENT ADMIN

View Feedback

Copy Excel CSV PDF Search:

#	Full Name	Email	Contact	Messages	Action
1	Md Azrul	azrulalam@gmail.com	01686239004	hi	

Showing 1 to 1 of 1 entries Previous 1 Next

© 2023 Md Kamrul Islam & Md Azrul Alam. All rights reserved Facebook Twitter Google Plus

7.5 View Prediction Result Page:

© DIT,Sobhanbag Dhaka 01686239004 f w G+ P

Heart Disease Prediction Using ML HOME MY DETAIL SEARCHED DATA ADMIN

View Searched Data

Copy Excel CSV PDF Search:

#	Patient Name	Phone	Accuracy	Result	HeartValue	Action
1	Md Azrul Alam	01686239004	88.52459016593442	Healthy	[24, 0, 2, 200, 123, 0, 0, 123, 0, 1, 1, 1, 1]	
2	Md Azrul Alam	01686239004	88.52459016593442	Unhealthy	[24, 0, 2, 120, 214, 0, 0, 98, 0, 1, 1, 1, 1]	
3	Md Azrul Alam	01686239004	88.52459016593442	Unhealthy	[46, 0, 0, 130, 214, 0, 0, 168, 0, 2, 1, 0, 3]	
4	Md Azrul Alam	01686239004	88.52459016593442	Unhealthy	[46, 0, 0, 130, 214, 0, 0, 168, 0, 2, 1, 0, 3]	
5	Md Azrul Alam	01686239004	88.52459016593442	Unhealthy	[46, 0, 1, 130, 214, 0, 0, 168, 0, 2, 1, 0, 3]	
6	Md Azrul Alam	01686239004	88.52459016593442	Unhealthy	[46, 1, 1, 130, 214, 0, 0, 168, 0, 2, 1, 0, 3]	
7	Md Azrul Alam	01686239004	88.52459016593442	Unhealthy	[46, 1, 2, 130, 214, 0, 0, 168, 0, 2, 1, 0, 2]	
8	Md Azrul Alam	01686239004	88.52459016593442	Unhealthy	[41, 0, 1, 130, 204, 0, 1, 172, 0, 14, 0, 2, 2]	

7.6 All Doctors Page:

© DIT,Sobhanbag Dhaka 01686239004 f w G+ P

Heart Disease Prediction Using ML HOME DOCTOR PATIENT SEARCHED DATA COMMENT ADMIN

View Doctor

Copy Excel CSV PDF Search:

#	Full Name	Image	Email	Contact	Address	Category	Status	Assign	Action
1	Dr. Abdullah Al-Safi MqJumder		majumder@gmail.com	01672803232	Dhanmondi	None	Authorized	Cancel	
2	Prof. Dr. SM Ahsan Habib		habib@gmail.com	01972803232	Savar	None	Authorized	Cancel	

Showing 1 to 2 of 2 entries Previous 1 Next

© 2023 Md Kamrul Islam & Md Azrul Alam. All rights reserved Facebook Twitter Google Plus

7.7 Admin Dashboard Page:

DIIT, Sobhanbag Dhaka | 01686239004 | f | G+ | P

Heart Disease Prediction Using ML | HOME | DOCTOR | PATIENT | SEARCHED DATA | COMMENT | ADMIN

ADMIN DASHBOARD

2 Total Doctors All access are given .	2 Total Patients All access are given .	1 Total Feedback All access are given .	8 Searched Quantity All access are given .
---	--	--	---

© 2025 Md Kamrul Islam & Md Azrul Alam. All rights reserved | Facebook | Twitter | Google Plus

7.8 Result Page:

DIIT, Sobhanbag Dhaka | 01686239004 | f | G+ | P

Heart Disease Prediction Using ML | PREDICT | MY DETAIL | COMMENT | REPORT IMG | HISTORY | HELLO, P2

HEART CONDITION RESULT

Heart Condition

Accuracy (%) is : 88.52459016393442
Result is : Your heart are fine. No problem found.

© 2025 Md Kamrul Islam & Md Azrul Alam. All rights reserved | Facebook | Twitter | Google Plus

7.9 User Profile Page:

DIIT, Sobhanbag Dhaka | 01686239004 | f | G+ | P

Heart Disease Prediction Using ML | HOME | MY DETAIL | SEARCHED DATA | OR3

My Detail

Full Name :	Prof. Dr. SM Ansan Habib
Image :	
Email Id :	habib@gmail.com
Date of Birth :	Dec. 22, 1979
Contact :	01972805232
City :	Savar
Specialist :	None

UPDATE DETAIL

7.10 Doctor Suggestion Page:

DIIT, Sobhanbag Dhaka 01686239004 f G+ P

Heart Disease Prediction Using ML PREDICT MY DETAIL COMMENT REPORT IMG HISTORY HELLO, P3


HEART CONDITION RESULT

Heart Condition

Accuracy (%) is : 88.52459016393442
Result is : Your Heart has a problem. You Need to Checkup.

Your City Best Doctor is here

Copy Excel CSV PDF Search:

#	Image	Full Name	Email	Contact	Address
1		Prof. Dr. SM Ahsan Habib	habib@gmail.com	01972805252	Savar

Showing 1 to 1 of 1 entries Previous 1 Next

7.11 User Login Page:

DIIT, Sobhanbag Dhaka 01686239004 f G+ P

Heart Disease Prediction Using ML HOME ABOUT US REPORT IMG

LOGIN NOW

Username

We'll never share your Detail with anyone else.

Password

LOGIN

7.12 User Registration Page:

DIIT, Sobhanbag Dhaka 01686239004 f G+ P

Heart Disease Prediction Using ML HOME ABOUT US REPORT IMG

REGISTER NOW

Full Name <input type="text" value="First Name"/>	Institute Name (Optional) <input type="text" value="Only Doctors"/>
Username <input type="text" value="Username"/>	Password <input type="password" value="Password"/>
Email <input type="text" value="Enter Email"/>	Contact <input type="text" value="Enter Contact"/>
Date Of Birth <input type="text" value="mm/dd/yyyy"/>	Image <input type="button" value="Choose File"/> No file chosen
City <input type="text" value="Dhanmondi"/>	User Type <input type="radio"/> Patient <input type="radio"/> Doctor

REGISTER

By clicking Register, I agree to your terms

Chapter 8

Conclusion

8.1 Limitation of Our System

- Some features is not optimize for the mobile users.
- Doctors cannot see the patients address
- Patients cannot send comment to the doctors

8.2 Future Enhancement

- Patients and Doctors Real time chatting features.
- Patients can easily give there appointment to their selected doctors
- Mobile application

8.3 Conclusion

It has been a great pleasure for me to work on this exciting and challenging project. This project proved good for me as it provided practical knowledge of not only programming in Python and Sqlite web based application. It also provides knowledge about the latest technology used in developing web enabled application and client server technology that will be great demand in future. This will provide better opportunities and guidance in future in developing projects independently.future research. Future research could focus on addressing the limitations of this study by comparing the performance of the k-modes clustering algorithm with other commonly used clustering algorithms, such as k-means.^[14] or hierarchical clustering.^[15] to gain a more comprehensive understanding of its performance. Additionally, it would be valuable to evaluate the impact of missing data and outliers on the accuracy of the model and develop strategies for handling these cases. Furthermore, it would be beneficial to evaluate the performance of the model on a held-out test dataset in order to establish its generalizability to new, unseen data. Ultimately, future research should aim to establish the robustness and generalizability of the results and the interpretability of the clusters formed by the algorithm, which could aid in understanding the results and support decision making based on the study's findings.

8.4 References

1. WHO. (2023). *World Health Organization*. Retrieved February 2, 2023, from World Health Organization website: <https://www.who.int/health-topics/cardiovascular-diseases/en>
2. Sultana, M. (2017, March 06). IEEE Xplore. *Analysis of Data Mining Techniques for Heart Disease Prediction*, 76, 1-20. Retrieved February 2, 2023, from <https://ieeexplore.ieee.org/document/7873142>
3. K, M. I., I., A., & Ali, S. M. (2019, june). Heart Disease Prediction Using Machine Learning Algorithms. *Journal of marketing*, 1-19. Retrieved February 10, 2023
4. Bhanot, K. (2019, February 13). *Predicting presence of Heart Diseases using Machine Learning*. Retrieved February 13, 2023, from [towardsdatascience.com](https://towardsdatascience.com/predicting-presence-of-heart-diseases-using-machinelearning-36f00f3edb2c): <https://towardsdatascience.com/predicting-presence-of-heart-diseases-using-machinelearning-36f00f3edb2c>
5. B, P. L., Davis B R, C. J., & Cushman W C, W. J. (2002, Nov 14). Validation of heart failure events in the Antihypertensive and Lipid Lowering Treatment to Prevent Heart Attack Trial (ALLHAT) participants assigned to doxazosin and chlorthalidone. *National Library Of medicine*. Retrieved February 14, 2023, from <https://pubmed.ncbi.nlm.nih.gov/12459039/>
6. Shorewala. (2021, june 28). Early detection of coronary heart disease using ensemble techniques. *Science Direct*. Retrieved March 1 2023, from <https://www.sciencedirect.com/science/article/pii/S235291482100143X>
7. Waigi, R., Choudhary, S., Fulzele, P., & Mishra. (2020, 07). G. Predicting the risk of heart disease using advanced machine learning approach. *Eur. J. Mol. Clin. Med*, 1638–1645. Retrieved March 05, 2023
8. SINGH, U. (2018, July). *Kaggle*. Retrieved March 10, 2023, from Kaggle web site: <https://www.kaggle.com/datasets/utkarshx27/heart-disease-diagnosis-dataset>
9. Han, J., & Kamber. (2011). *Data Mining: Concepts and Techniques*. *Google Scholars*. Retrieved March 24, 2023, from https://scholar.google.com/scholar_lookup?title=Data+Mining:+Concepts+and+Techniques&author=Han,+J.A.&author=Kamber,+M.&publication_year=2011
10. Rivero, R., & Garcia, P. (2009). A Comparative Study of Discretization Techniques for Naive Bayes Classifier. *IEEE Trans. Knowl.*, 674–688. Retrieved April 01, 2023, from https://scholar.google.com/scholar?q=A+Comparative+Study+of+Discretization+Techniques+for+Naive+Bayes+Classifiers&hl=en&as_sdt=0&as_vis=1&oi=scholar

11. Khan, S., Ning, H., Wilkins, J., Allen, N., Carnethon, M., Berry, J., . . . Lloyd-Jones, D. (2018, May). Association of body mass index with lifetime risk of cardiovascular disease and compression of morbidity. *JAMA Cardiol*, 280–287. Retrieved April 11, 2023, from <https://jamanetwork.com/journals/jamacardiology/article-abstract/2673289>
12. Kengne, A.-P., Czernichow, S., Huxley, R., Grobbee, D., Woodward, M., Neal, B., . . . al., e. (2009). Blood Pressure Variables and Cardiovascular Risk. *Hypertension*, 399–404. Retrieved April 23, 2023, from <https://www.ahajournals.org/doi/full/10.1161/HYPERTENSIONAHA.109.133041>
13. Yu, D., Zhao, Z., & Simmons, D. (2016, Aug 3). Interaction between Mean Arterial Pressure and HbA1c in Prediction of Cardiovascular Disease Hospitalisation: A Population-Based Case-Control Study. *J. Diabetes Res*. Retrieved May 1, 2023, from <https://www.hindawi.com/journals/jdr/2016/8714745/>
14. Hassan, C., Iqbal, J., Irfan, R., Hussain, S., Algarni, A., Bukhari, S., . . . Ullah, S. (2022, jun). Effectively Predicting the Presence of Coronary Heart Disease Using Machine Learning Classifiers. *Sensors*, 01-35. Retrieved May 11, 2023, from <https://www.mdpi.com/1424-8220/22/19/7227>
15. Subahi, A., Khalaf, O., Alotaibi, Y., Natarajan, R., Mahadev, N., & Ramesh, T. (2022). Modified Self-Adaptive Bayesian Algorithm for Smart Heart Disease Prediction in IoT System. *Sustainability*, 01-14. Retrieved Jun 1, 2023, from <https://www.mdpi.com/2071-1050/14/21/14208>

Appendix

Add_heartdetails.html

```
{% extends 'index.html' %}
{% load static %}
{% block body %}
<!-- register -->
<section class="logins py-5">
<div class="container py-xl-5 py-lg-3">
<div class="title-section mb-md-5 mb-4">
<h6 class="w3ls-title-sub"></h6>
<h3 class="w3ls-title text-uppercase text-dark font-weight-bold">Add Heart
Detail</h3>
</div><hr/>
<div class="login px-sm-12" style="width:100%">
<form action="" method="post" enctype="multipart/form-data">
{% csrf_token %}
<div class="form-group row">
<div class="col-md-1">
<label>Age</label>
<input type="text" class="form-control" name="age" required="">
</div>
<div class="col-md-1">
<label>Sex</label>
<select type="text" class="form-control" name="sex" required="">
<option value="M">Male</option>
<option value="F">Female</option>
</select>
</div>
<div class="col-md-1">
```

```
<label>CP</label>
<select type="text" class="form-control" name="cp" required="">
<option value="0">typical angina</option>
<option value="1">
```

Views.py

```
from django.contrib import messages
from django.contrib.auth.decorators import login_required
from django.shortcuts import render, redirect
import datetime
from sklearn.ensemble import GradientBoostingClassifier
from .forms import DoctorForm
from .models import *
from django.contrib.auth import authenticate, login, logout
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

sns.set_style('darkgrid')
from sklearn.preprocessing import StandardScaler, MinMaxScaler, RobustScaler
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC
from sklearn.neural_network import MLPClassifier
from django.http import HttpResponse
# Create your views here.

doctor.status = 2
messages.success(request, 'Selected doctor are successfully withdraw his approval.')
else:
```

```

doctor.status = 1
messages.success(request, 'Selected doctor are successfully approved.')
doctor.save()
return redirect('view_doctor')

else:
login(request, user)
error="notmember"

else:
error="not"
d = {'error': error}
return render(request, 'login.html', d)

def Login_admin(request):
error = ""
if request.method == "POST":

u = request.POST['uname']
p = request.POST['pwd']
user = authenticate(username=u, password=p)
if user.is_staff:
login(request, user)
error="pat"
else:
sign = Patient.objects.get(user=user)
if sign:
error = "pat"
except:
sign = Doctor.objects.get(user=user)
terror = ""
if request.method=="POST":

```

```

n = request.POST['pwd1']

c = request.POST['pwd2']
o = request.POST['pwd3']
if c == n:
    u = User.objects.get(username__exact=request.user.username)
    u.set_password(n)
    pred = "<span style='color:red'>You are Unhealthy, Need to Checkup.</span>"
    return redirect('predict_desease', str(rem), str(accuracy))
doc = Search_Data.objects.get(id=pid)
doc.delete()
return redirect('view_search_pat')

@login_required(login_url="login")
def View_Doctor(request):
    doc = Doctor.objects.all()
    d = {'doc':doc}
    return render(request,'view_doctor.html',d)

@login_required(login_url="login")
def View_Patient(request):
    patient = Patient.objects.all()
    d = {'patient':patient}
    return render(request,'view_patient.html',d)

@login_required(login_url="login")
def View_Feedback(request):
    dis = Feedback.objects.all()
    d = {'dis':dis}
    return render(request,'view_feedback.html',d)

```

```

@login_required(login_url="login")
def View_My_Detail(request):
    terror = ""
    user = User.objects.get(id=request.user.id)
    error = ""
    try:
        sign = Patient.objects.get(user=user)
        error = "pat"
    except:
        sign = Doctor.objects.get(user=user)
    d = {'error': error,'pro':sign}
    return render(request,'profile_doctor.html',d)

```

```

@login_required(login_url="login")
def Edit_Doctor(request,pid):
    doc = Doctor.objects.get(id=pid)
    error = ""
    # type = Type.objects.all()
    if request.method == 'POST':
        f = request.POST['fname']
        l = request.POST['lname']
        e = request.POST['email']

```