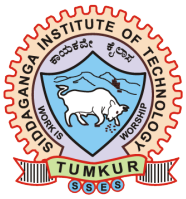
Prediction of heart disease and diabetes using machine learning

A project report submitted to
Visvesvaraya Technological University. Belgaum, Karnataka
in the partial fulfillment of the requirements for the award of degree of

Bachelor of Engineering
in
Computer Science and Engineering
by

Akshat Agarwal	**1SI16CS010**
Akarsh Singh	**1SI16CS007**
Ayush Bhargava	**1SI16CS131**
Srijan Yadav	**1SI16CS109**

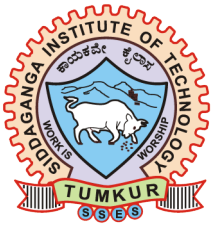
under the guidance of
H.K Vedamurthy
Assistant Professor



Department of Computer Science and Engineering
Siddaganga Institute of Technology, Tumakuru

May, 2020

Department of Computer Science and Engineering
Siddaganga Institute of Technology
Tumakuru - 572103



CERTIFICATE

Certified that the Project Report entitled **"Prediction of Heart disease and diabetes using machine learning"** is a bonafide work carried out by **Akshat Agarwal (1SI16CS010), Akarsh Singh (1SI16CS007), Ayush Bhargava (1SI16CS131) and Srijan Yadav (1SI16CS109)** in the partial fulfillment of the requirement for the award of the degree of Bachelor of Engineering in Computer Science and Engineering , Visvesvaraya Technological University, Belagavi during the year 2015-16. It is certified that all corrections/suggestions indicated for the internal assessment have been incorporated in the report. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the Bachelor of Engineering Degree.

.....	.....
Guide	**Group Convener**
H.K Vedamurthy	**Dr. Shreenath K N**
Asst. Professor	**Professor**
Dept of CSE, SIT	**Dept of CSE, SIT**
.....	.....
Dr. R. Sumathi	**Dr. Shivananda K P**
Professor and Head	**Principal**
Dept of CSE, SIT	**SIT, Tumakuru**

Name of the Examiners — Signature with Date

1. Prof.
2. Prof.

Department of Computer Science and Engineering
Siddaganga Institute of Technology
Tumakuru - 572103

DECLARATION

I hereby declare that the entire work embodied in this dissertation has been carried out by me at **Siddaganga Institute of Technology** under the supervision of **H.K Vedamurthy**. This dissertation has not been submitted in part or full for the award of any diploma or degree of this or any other University.

Name of the student with USN

- Akashat Agarwal 1SI16CS010
- Akarsh Singh 1SI16CS007
- Ayush Bhargava 1SI16CS131
- Srijan Yadav 1SI16CS109

Department of Computer Science and Engineering
Siddaganga Institute of Technology
Tumakuru - 572103

Acknowledgements

With reverential pranams, we express our sincere gratitude and salutation to His Holiness Dr. Sree Sivakumara Swamigalu of Sree siddaganga Mutt for his unlimited blessings. First and foremost, we wish to express our deep sincere feelings of gratitude to our institution, Siddaganga Institute of Technology for providing us for completing our project successfully. We are grateful to Dr. M.N. Channabasappa, Director, Siddaganga Institute of Technology, Tumakuru for his cooperation and encouragement. We express our kind thanks to Dr. Shivananda K P, principal, Siddaganga Institute of Technology Tumakuru for his encouragement towards student's attitude.

We express our heartfelt thanks to Dr. R. Sumathi, Professor and Head, Department of Computer Science and Engineering, Siddaganga Institute of Technology, Tumakuru for her suggestions and advice. We express our gratitude and humble thanks to our project guide Mr. H.K Vedamurhty, Assistant Professor, Department of Computer Science and Engineering, Siddaganga Institute of Technology, Tumakuru for guiding and facilitating to complete our Major-Project successfully.

We are conscious of the fact that we have received cooperation in many ways from the Teaching, Technical and supporting staffs of the Department of Computer Science and Engineering and we are grateful to all for their cooperation.

We express heartfelt gratitude to our parent and friends for their constant moral support and encouragement throughout this work.

Abstract

1. **Objectives of the project** Paragraph on motivation to do the current project
   - The proposed system predicts heart diseases as well as the chances of diabetes
   - There are no proper methods to handle semi structured and unstructured data. The proposed system is expected to work well with both structured and unstructured data.
   - The secondary target of the project is to build a web application which permits clients to anticipate diabetes and coronary illness by using prediction engine.
2. **Description** With the advance of big data analytics equipment, more devotion has been paid to disease expectation from the perception of big data inquiry, various explores have been conducted by choosing the features mechanically from a large number of data to improve the truth of menace classification rather than the formerly selected physiognomies. However, those prevailing work mostly measured structured data. Number of inquires about has been led to choosing the characteristics of a disease prediction from an enormous volume of an data. A large portion of the current work is based on structured data. For the unstructured data one can utilize a convolutional neural network. Convolutional neural network are comprised of a neurons, every neuron gets a few sources of info and performs activities and the entire network expresses a single differentiable score methods.
   The system analyses the structured and unstructured data in healthcare field to assess the risk of disease. First, it uses Decision tree map algorithm to generate the pattern and causes of disease. Second, by using Map Reduce algorithm for partitioning the data such that a query will be analyzed only in a specific partition, which will increase the operational efficiency but reduce query retrieval time. Map reducing algorithm is used for partitioning the medical data based on the output of Decision Tree map algorithm. Compared to several

typical prediction algorithms, the prediction accuracy of our proposed algorithm increases.

3. **Validation of Test Results**

   - **K-Fold Cross-Validation:** In K fold cross validation, the data set is separated into K equivalent size of parts, in which K less one groups are utilized to train the classifiers and remaining part is utilized for checking the out-performance in each progression. The procedure of this new valid ation is repeated K number of times. The classifier performance is computed dependent on K results..
   - **Manual Validation:** The test results predicted by the model can be cross verified with the observed patient medical reports

Contents

Acknowledgements iii

Abstract iv

List of Figures ix

1 Introduction 1

1.1 Background Study . . . 2
1.1.1 Motivation . . . 2
1.1.2 Social Impact . . . 4
1.2 Related Work . . . 5
1.2.1 Effective Heart Disease Prediction System . . . 6
1.2.2 Analysis and Prediction the Occurrence of Coronary Illness Using Information Mining Techniques . . . 6
1.2.2.1 Particle swarm optimization(PSO) . . . 7
1.2.2.2 Naive Bayes Classifier . . . 7
1.2.3 Datasets . . . 8
1.2.3.1 Cleveland Heart Dataset . . . 8
1.2.3.2 Pima Indians Diabetes Dataset . . . 11
1.3 Summary of Gaps identified . . . 13
1.4 Project problem statement and Objective . . . 13
1.4.1 Problem Statement . . . 13
1.4.2 Objectives of the project . . . 14
1.5 Organization of the Report . . . 14

2 High-level Design 15

2.1 Software Development Methodology . . . 15
2.1.1 Stage 1: Planning and Requirement Analysis . . . 15
2.1.2 Stage 2: Defining Requirements . . . 16
2.1.3 Stage 3: Designing the Product Architecture . . . 16
2.1.4 Stage 4: Building or Developing the Product . . . 17

2.1.5 Stage 5: Testing the Product . . . 17
2.1.6 Stage 6: Deployment in the Market and Maintenance . . . 17
2.2 Architecture . . . 17
2.3 Incremental Model . . . 19
2.4 Agility and Scrum . . . 21
2.4.1 Agility and the cost of change . . . 21
2.5 Scrum . . . 22
2.5.1 Activities performed by the team in the scrum . . . 22
2.6 Functional Requirements . . . 24
2.7 Non-Functional Requirement . . . 24
2.8 Feasibility Analysis . . . 24
2.8.1 Technical feasibility . . . 24
2.8.2 Economic feasibility . . . 25

3 Detailed Design **26**
3.1 Interface Design . . . 26
3.2 Data Structures and Algorithms . . . 30
3.2.1 Naive Bayes Classifier . . . 30
3.2.2 Decision Tree . . . 34
3.2.3 Support vector machine(SVM) . . . 36
3.2.4 Logistic Regression . . . 38
3.2.4.1 Cost Function . . . 40
3.2.4.2 Gradient Descent . . . 42
3.2.4.3 Sigmoid Function . . . 42
3.2.5 K-Nearest Neighbour . . . 43
3.2.6 Neural Network . . . 43
3.2.6.1 Multilayer Perceptron Neural Network (MLPNN) . . . 43
3.2.6.2 Backpropagation network . . . 46
3.3 UML diagrams with discussions . . . 48
3.4 Data Source and Formats . . . 49
3.4.1 Heart Disease DataSet . . . 49
3.4.2 Diabetes DataSet . . . 52

4 Implementation **54**
4.1 Tools and Technologies . . . 54
4.1.1 Django . . . 54
4.1.2 Python . . . 54
4.1.3 SqlLite . . . 55
4.1.4 IntelliJ IDEA . . . 55

4.1.5 Machine Learning . . . 56
4.1.6 HTML . . . 56
4.1.7 Cascading Style Sheets . . . 57
4.2 Experimental Setup . . . 57
4.3 Coding Standards followed . . . 58
4.4 Code Integration details . . . 58
4.5 Implementation work flow . . . 58
4.5.1 Data cleaning . . . 59
4.5.1.1 Sources of Missing Values . . . 59
4.5.1.2 Non-Standard Missing Values . . . 59
4.5.2 Handling unstructured and structured Data . . . 59
4.6 Execution Results and Discussions . . . 60
4.7 Non-functional requirements results . . . 61

5 Testing **62**
5.1 Test workflow . . . 62
5.1.1 Integration Testing . . . 62
5.1.2 Unit Testing . . . 62
5.1.3 Validation Testing . . . 63
5.2 Test case details . . . 63
5.2.1 Test case 1: . . . 63
5.2.2 Test case 2: . . . 64
5.2.3 Test case 3: . . . 65
5.2.4 Test case 4: . . . 65
5.2.5 Test case 5: . . . 66

6 Conclusions and Future Scope **67**

List of Figures

3.1 Home page 27
3.2 Home 27
3.3 Home 28
3.4 SignUp and Login feature 29
3.5 options of Prediction Engine 30
3.6 Heart diseases prediction form for patient 31
3.7 Heart diseases prediction result for patient 32
3.8 Diabetes prediction form for patient 33
3.9 Diabetes prediction result for patient 34
3.10 Profile of a patient showing the past results 35
3.11 Implementation Flow of Naive Bayes Algorithm 36
3.12 Flowchart for Decision Tree 37
3.13 Flowchart For SVM 39
3.14 FlowChart For Logistic Regression 41
3.15 FlowChart for KNN 44
3.16 Neural Network 45
3.17 BackPropagation 46
3.18 Flowchart for Neural Network 47
3.19 UML Sequence Diagram 48

4.1 making a list of missing values 60

Chapter 1

Introduction

With the development of big data analytics equipment, more commitment has been paid to disease desire from the impression of the big data request, different analyses have been directed by picking the highlights precisely from an enormous number of data to improve the reality of danger characterization instead of the in the past chose physiognomies. Be that as it may, those overall work, for the most part, estimated structured data. Various looks into have been led to choose the attributes of a disease forecast from a huge volume of data. The vast majority of the current work depends on structured data. For the unstructured data, one can utilize a convolutional neural system. Convolutional neural networks are comprised of a neuron, every neuron gets a few information sources and performs activities and the entire system communicates a single differentiable score function.

The framework examines the data in the medical field to evaluate the danger of disease. It utilizes methods to clean and change the data. Second, by utilizing different machine learning algorithms, it investigations the new approaching data point and orders the point into one of the two groups to be specific whether the individual is experiencing disease or not experiencing the disease. Different investigation procedures have been utilized to clean and change the data to fit the data into the machine learning model successfully. Contrasted with a few run of the mill forecast algorithms, the expected accuracy of our proposed algorithm framework is the most elevated

The essential point of this undertaking is to break down the "Pima Indian Diabetes Datasets" and "Heart Disease Dataset" and utilize Logistic Regression, Support Vector Machine, Naïve Bayes, K -Nearest Neighbors and Multi-Layer Perceptron (Neural Network) for forecast and build up an expectation motor and a straightforward UI which is simple and basic for new clients or patients to utilize. As far as we could know in the territory of clinical data analytics, none of the current work centres around the equivalent.

1.1 Background Study

1.1.1 Motivation

The principle inspiration for doing this project is to introduce a prediction model for the prediction of the occurrence of diabetes and heart diseases. Further more, this undertaking work is pointed toward distinguishing the best classification method for recognizing the chance of heart disease or diabetes in a patient. This work is justified by playing out a similar report and analysing, utilizing some machine learning algorithms for classification namely Naive Bayes, Decision Tree, K -Nearest Neighbors, Logistic Regression, Support Vector Classifier and Neural Networks. Despite the fact that these are normally utilized machine learning algorithms, disease prediction is anvital task including the most highest possible exactness. Subsequently, the three algorithmic methods are assessed at various levels and sorts of classification strategies. This will give scientists and clinical experts to set up a superior understanding and assist them with recognizing an answer for distinguish the best technique for anticipating heart sicknesses as well as the chance of diabetes.

A key challenging task confronting healthcare organisation (emergency clinics, clinical focuses) is the facility of qualities administrations at vary reasonable costs. Quality amenities recommend diagnosing patents precisely and controlling medications which are compelling. Poor clinical decisions can result in deplorable outcomes, that are as such un satisfactory. Medical clinics should limit the expense of clinical tests. They can achieve this results by using fitting PCs based information and additional choice emotionally supportive networks. The heart is a essential bit of our body. Life is itself dependent on successful working of the heart. In the task in which the undertaking of the heart's not real, it will impact other body part of human, for instance, cerebrum, kidney, etc. Coronary ailment is an sickness that impacts on the action of the heart.

There are a few components which assemble the threat of heart disease. Some of them are recorded below:

- The family history of heart disease.
- The family history of diabetes.
- Smoking .
- Cholesterol .
- High blood pressure .

- Obesity.
- Lack of physical exercises.

In light of the wide openness of superlative proportion of data and need to change over this available tremendous proportion of information to supportive data require the use of data mining procedures. Data Mining and KDD (learning disclosure in the database) has ended up being non-prominent as of late. The popularities of data mining and KDD (data disclosure in the database) should not be wonder since the proportion of the data expands that are open are very broad to be analyzed physically and even the procedures for programmed data examination taking into account built up bits of knowledge and machine adjusting as often as possible compromise issues while planning huge, dynamic information increases comprising of complex item. Data Mining is the highlight of Knowledge Discovery Database (KDD). Various people view Data Mining as an identical word for KDD since it is a key bit of the KDD procedure. There are certain phases of data mining that u should know, and these are exploration , design recognizable proof, and deployment . Data mining is an iterative methodology that regularly incorporates the going with stage.

- About 1 among every 4 deaths in India occur due to heart disease.
- Coronary illness is the main source of death in India. More than 50 percent of the demise due to coronary disease in the year 2009 were in men.
- In India , someone has a heart attack every 41 seconds.
- 1% of women of age 41 or more who participate in the continuos screening have heart problem.
- A great deal of cash is spent by the administration on the patients determined to have heart illnesses. The amount spent incorporates the expense of health-care insurance services, meds, and lost profitability.

1.1.2 Social Impact

In everyday life, a few elements affect a human heart. A few issues are happening at a quick pace and new heart ailments are quickly being recognized. In this day and age of pressure, Heart, being one of the most significant organs in a human body that siphons blood through the body for the blood dissemination is basic and its wellbeing is to be safeguarded for a solid living. The wellbeing of the heart acknowledges on the encounters in an extremely individual's life and is absolutely reliant on the expert and individual practices of an individual. There may likewise be a few hereditary factors through which a sort of coronary illness is passed down from ages. As indicated by the World Health Organization, consistently in excess of 12 million passings are happening worldwide because of the different kinds of heart illnesses which are additionally known by the term cardiovascular sickness. The term Heart ailment incorporates numerous infections that are different and explicitly influence the heart and the veins of a person. Indeed, even youthful matured individuals around their 20-30 years of life expectancy are getting influenced by heart maladies. The expansion in the chance of coronary illness among youngsters might be because of the terrible dietary patterns, absence of rest, anxious nature, wretchedness and various different factors, for example, stoutness, horrible eating routine, family ancestry, hypertension, high blood cholesterol, inactive conduct, family ancestry, smoking and hypertension. The determination of heart ailments is significant and is itself the most confounded undertaking in the clinical field. All the referenced elements are mulled over when breaking down and understanding the patients by the specialist through manual registration at customary interims of time.

The side effects of coronary illness significantly rely on which of the uneasiness felt by a person. A few side effects are not normally recognized by the average people.

The common symptoms include chest pain , breath less ness, and heart palpitations. The chest pain common to numerous sorts of the heart disease is known as angina , or angina pectoris, and happens when a part of the heart doesn't get enough oxygen. angina might be activated by distressing occasions or physical effort and typically endures under 10 minutes. Heart failures can likewise happen because of various sorts of heart diseases. The indications of a respiratory failure resemble anginal discomfort aside from that they can happen during rest and will, in general, be increasingly serious. The manifestations of heart failure can some of the time take after heartburn. Acid reflux and a stomach hurt can happen, just like an overwhelming pain in the chest. Different symptoms of a respiratory failure incorporate agony that movements through the body, for instance from the chest to the arms, neck, back, midregion, or jaw, dazedness and dizzy sensations, profuse sweating, nausea and vomiting.

Heart failure is likewise a result of heart disease, and breathlessness can happen when the heart turns out to be too weak to circulate blood. Some heart conditions happen without any symptoms by any stretch of the imagination, particularly in more seasoned grown-ups and people with diabetes. the term 'inborn heart disease' covers a scope of conditions, however, the general side effects incorporate sweating, elevated levels of weakness, fast heartbeat and breathing, shortness of breath, chest pain. Notwithstanding, these side effects probably won't develop in an individual until he/she is younger than 13 years. In these kinds of cases, the analysis turns into a mind-boggling task requiring extraordinary experience and high aptitude. The danger of a heart attack or the chance of heart disease whenever recognized early can enable the patients to play it safe and take administrative measures. As of late, the human services industry has been producing colossal measures of information about patients and their disease conclusion reports are in effect particularly taken for the forecast of heart assaults around the world. At the point when the information about heart disease is enormous, AI strategies can be executed for the investigation.

1.2 Related Work

The healthcare industry gathers a tremendous amount of human health information that, unfortunately, are not "mined" to discover the hidden data for successful decision making. The revelation of hidden pattern and relationship regularly goes un-exploited . The healthcare industry is still 'data-rich' but 'information poor'. Their is abundance of information accessible inside the medicial services framework. However, there is an absence of successful investigation apparatuses to find hidden relationships in the information. Today medical administrations have made some amazing progress to treat

patients with different diseases. Among the most deadly one is the heart disease issue which can't be seen with an unaided eye and comes in a flash when its limitations are reached. Today diagnosing patients accurately and regulating compelling medications have become a significant test. This area gives the details of the previous works and researchs performed.

1.2.1 Effective Heart Disease Prediction System

- **Author** Mr. Purushottam Sharma
- **Year** 2015

In the research paper , the authors have introduced an Efficient Heart Disease Prediction System utilizing data mining . This framework is useful to the clinical professional and is proficient and successful in decision making depending on the given parameters. The framework is trained and tested utilizing 10 overlap strategy and the last accuracy score acquired in the testing stage is 0.86 and 0.87 in the training stage.This model exhibits better outcomes and helps the region authorities and even individual related with the field to prepare for a better decide and give the patient than have early assurance results as it performs reasonably well even with no retraining .

1.2.2 Analysis and Prediction the Occurrence of Coronary Illness Using Information Mining Techniques

- **Author** Mr. Chala Beyene
- **Year** 2018

The principle goal of the proposed methodology in this research paper is to foresee the event of heart disease for an early programmed finding of the disease inside recovering outcomes in a brief timeframe. This assumes imperative jobs for medical field specialists to treat their patients dependent on precise dynamic and give characteristics of administrations to the individuals. The proposed methodology in the previously mentioned research paper is likewise basic in human services Organization with specialists that have no more information and ability. One of the primary impediments of the current methodology is the capacity to give precise outcome varying. The significant advantages of the study paper are the improved existing methodology for better dynamic by utilizing various algorithms and highlight determination strategies. The proposed methodology utilizes the Naive Bayes algorithm for anticipating the event of coronary illness for early programmed finding and the brief timeframe result recovery

that assists with giving the characteristic of administration and lessen expense to spare the lives of people.

1.2.2.1 Particle swarm optimization(PSO)

PSO is an Evolutionary Computation strategy proposed by Kenedy in 1995. PSO is roused by social practices, for example, bird running and fish schooling. In PSO , population swarm comprises of "n" particle, and the situation of every molecule represents the potential arrangement in D-dimensional space . The particles change its condition dependent on three perspectives: To keep its idleness; To change the condition as indicated by its most self-assured person position; To change the condition as per the multitude's most optimistic position. In PSO , a population is encoded as particle in the pursuit space dimensionality. PSO begins with random initialization of populace of particles. In light of the best understanding of one molecule and its neighboring particles, PSO looks for optimal solution by refreshing the speed and the situation of every molecule at equal time intervals.
PSO is used as feature subset selection method due to following advantages:

- Simple and very easy to build .
- Continuous and optimised approach.

1.2.2.2 Naive Bayes Classifier

Naive Bayes classifiers are a group of basic probabilistic classifiers based by utilizing Bayes theorem with solid (credulous) autonomy presumptions between the highlights. Naive Bayes classifiers are profoundly versatile by requiring a few parameter direct for number of highlights or indicators as a variable in a learning issue. It is least difficult and quickest probabilistic classifier, particularly for the training stage.

Feature selection - It is a procedure of expelling the insignificant and repetitive features from the dataset dependent on evaluation criterion that is utilized to improve precision. There are two methodologies as individual assessment and the other one is subset assessment . The procedure of feature selection is ordered into three expansive classes. One is 'filter', another is 'wrapper' and the third one is an embedded method based on how the feature selection is deployed by a supervised learning algorithm. In this paper, they proposed a model which utilizes Naive Bayes as classifier and PSO as Feature subset selection measure for prediction of coronary illness.

Proposed system - In this section , we propose a methodology to improve performance of Bayesian classifier for prediction of heart disease . Algorithm for proposed model is: .

Algorithm 1: Heart disease prediction by using Bayes classifier and PSO .

Input: Heart disease dataset .

Output: Classify patient dataset into heart disease or not (normal).

Step 1: Read the dataset .

Step 2: Apply particle swarm optimization for feature selection.

Step 3: Remove the features with low value of PSO .

Step 4: Apply the Naive Bayes classifier on relevant feature.

Step 5: Evaluate performance of NB+PSO model .

The above calculation isolated into two segments, section 1 (step 2 and step 3) performs handling and feature subset determination. In section 2 (step 4 and step 5) Naive Bayes is applied on relevant feature information and assess the performance as far as exactness. Cross-validation procedure used to split into training and testing information.

$$Accuracy = \frac{Number\ of\ objects\ correctly\ classified}{Total\ numberof\ objects\ in\ test\ set}$$

1.2.3 Datasets

For this project we have used The Cleveland Heart Dataset from the UCI Machine Learning Repository and the Pima Indians Diabetes Dataset as they are widely used by the pattern design community.

1.2.3.1 Cleveland Heart Dataset

The Cleveland heart dataset comprises of 303 individual clinical reports in which 164 don't have any illness. In this dataset there is are aggregate of 97 female patients in which 25 individuals are the confirmed case, likewise there are 206 male patients in which 114 are determined to have the sickness. There are 6 missing values in this dataset and every single numeric values are perceived as numeric. We have 13 feature that are applicable to the particular infection with respect to the dataset listed as follows:

- Age
- Sex

- Chest Pain Type
- Resting Blood Pressure
- Serum Cholesterol in mg/dl
- Fasting Blood Sugar
- Resting electrocardiographic result
- Maximum heart rate achieved
- Exercised-induced angina
- Old peak, ST depression induced by exercise relative to rest
- Number of major vessels colored by fluoroscopy
- Thal: Normal=3, fixed defect=6, reversible defect=7

The involvement of each attribute with respect to number of instances is as shown in the histogram below:

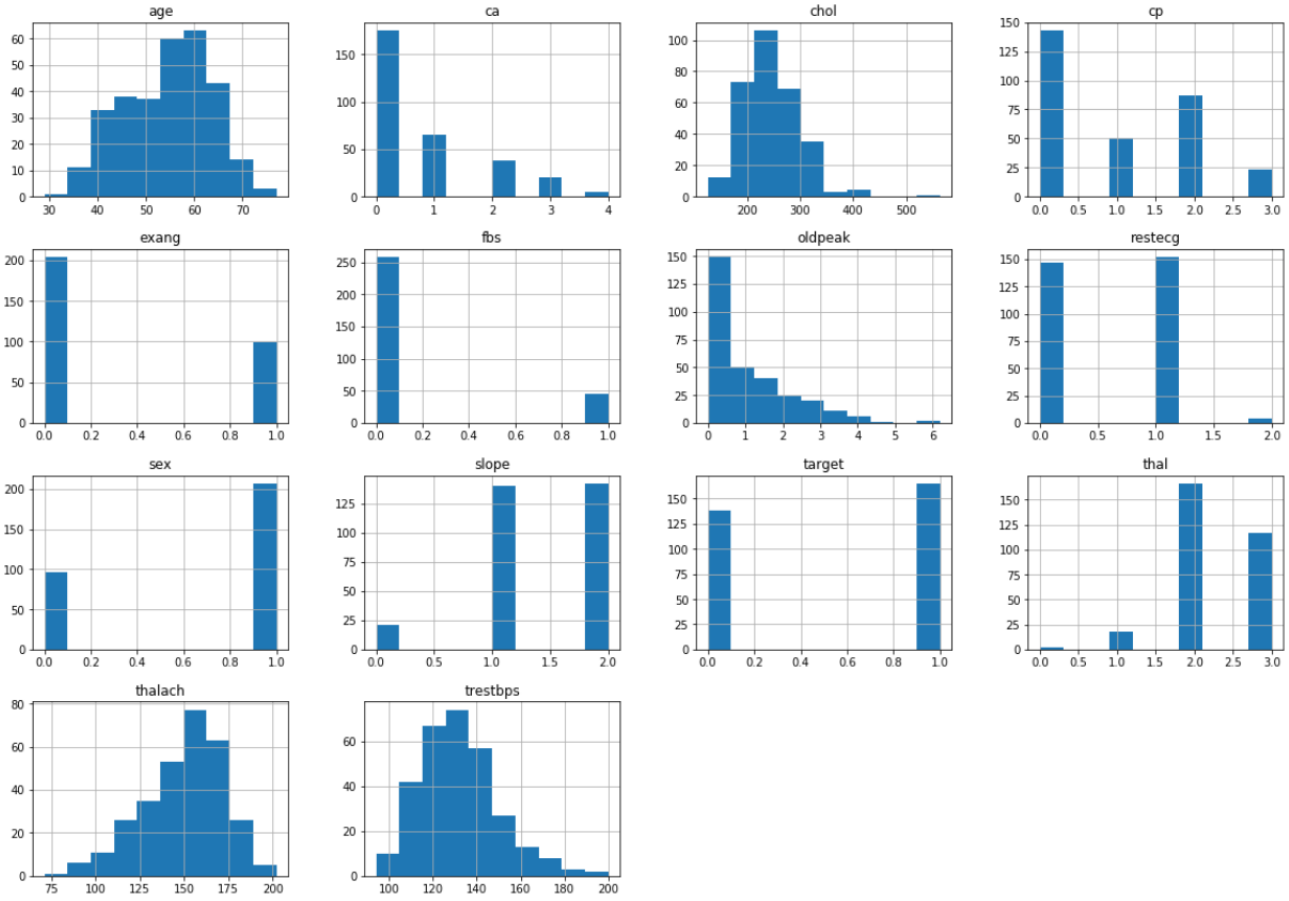


Figure 1.1 Histograms - Cleveland Heart Disease Dataset

Figure 1.2 Frequency Count of the Target Class

The Count of each target class for the given dataset is as depicted below. The two target classes are:

- 0: The instances that don't have heart disease.
- 1: The instances that have heart disease.

1.2.3.2 Pima Indians Diabetes Dataset

This dataset is initially from the National Institute of Diabetes and Digestive and Kidney Disease. The goal of the dataset is to analytically foresee whether a patient has diabetes, in view of certain diagnostic estimations included in the dataset. The dataset comprises of 768 individual clinical reports in which 500 don't have any sickness. In this dataset all the patients are females of atleast 21 years old of Pima Indian Heritage. The dataset consists of 8 features shown below:

- Pregnancies
- Glucose
- Blood Pressure
- Skin Thickness
- Insuline

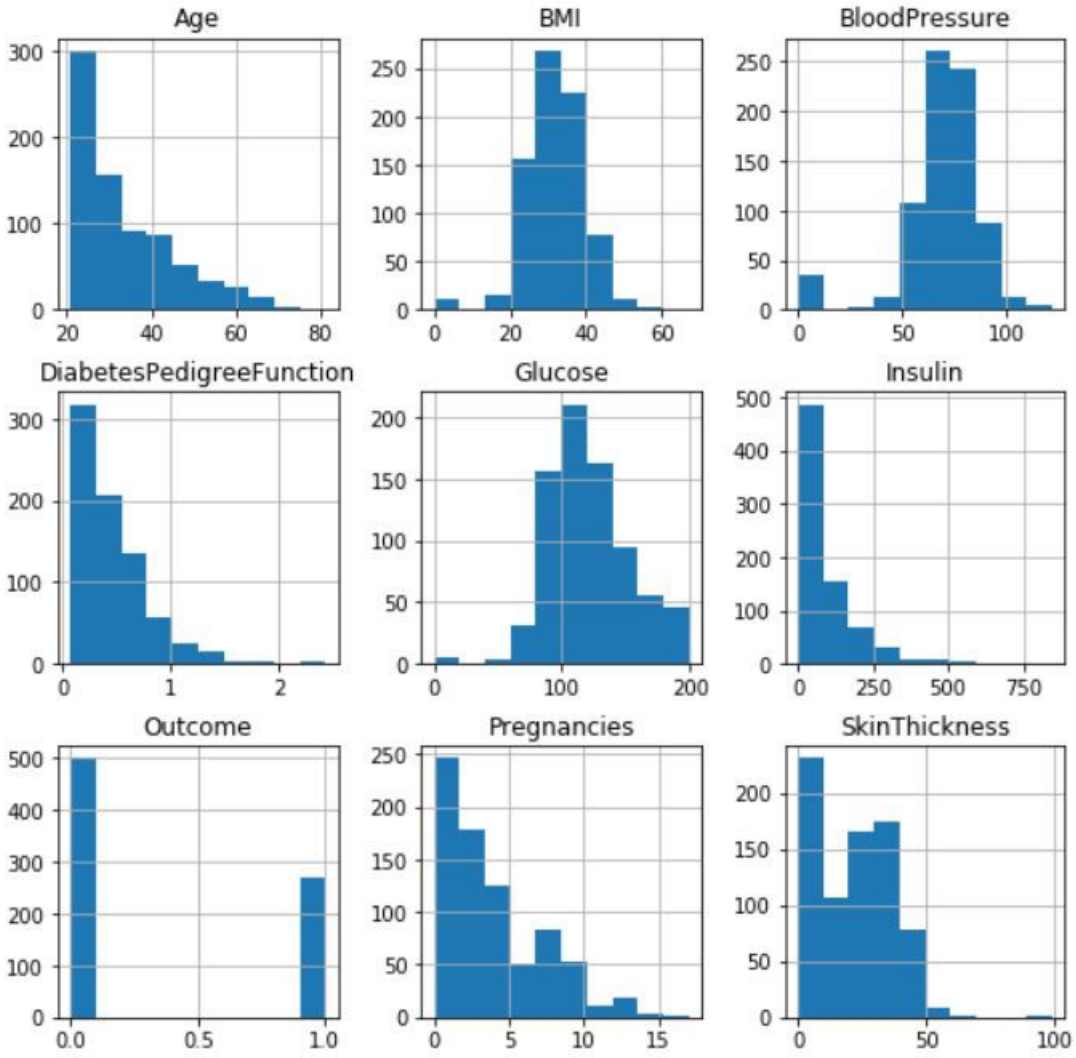


Figure 1.3 Histograms - Pima Indian Diabetes Dataset

- BMI
- Diabetes Pedigree Function
- Age

The involvement of each attribute with respect to the number of instances are shown in Figure 1.3:

The count of each target class for the given dataset is as depicted in Figure 1.4.

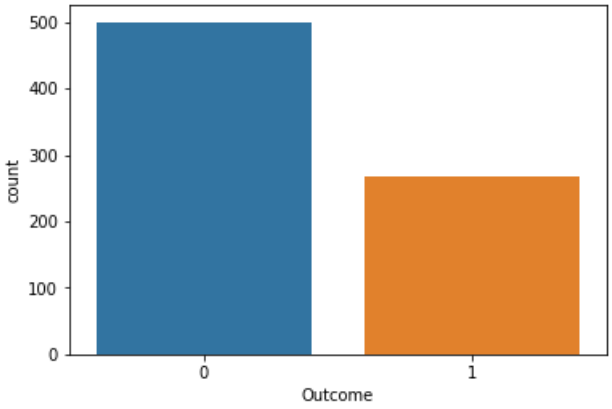


Figure 1.4 Frequency Count of the Target Class

The two target classes are:

- 0: The instances that don't have diabetes.
- 1: The instances that have diabetes.

1.3 Summary of Gaps identified

Medical diagnosis is considered as a noteworthy yet unpredictable errand that should be done accurately and proficiently. The robotization of the equivalent would be exceptionally helpful. Clinical choice are frequently made dependent on doctor's instinct and experience as opposed to the knowledge of rich information covered up in the database . This training prompt undesirable inclinations, mistakes and extreme medical cost which influences the nature of administration gave to patients. Information mining can create an information-rich condition which can help to essentially improve the nature of clinical choices.

1.4 Project problem statement and Objective

1.4.1 Problem Statement

Doctor depends on common knowledge for treatment . When known facts is lacking, studies are recalled after some cases have been read again. But this method takes some time, where as if we use machine learning, the symptoms can be identified at early state . For using machine learning , a huge amount of data is required. There is a very limited amount of data available depending on the disease . Also, the number of

sample having 0 diseases is large when compared with number of samples with positive disease .

1.4.2 Objectives of the project

1. The proposed system predicts heart diseases as well as the chances of diabetes.
2. Currently, there is no platform available which helps the users to predict the chances of heart disease and diabetes. We aim to build a powerful platform(web app) which helps the users to predict diabetes and heart disease.

1.5 Organization of the Report

The current chapter deals with the detailed Introduction of the project followed by the social and economical impacts of the project. The chapter also contains with the details of all the related research work carried out in this field.
The organization of the remainder of the report is as per the following:

- **Chapter 2**: This section contains the high-level structure of the proposed model alongside the software development methodologies utilized by the project developers during the advancement of this undertaking.
- **Chapter 3**: This chapter contains the design details and UML diagrams of the model along with the data structures and algorithms used in this project.
- **Chapter 4**: This chapter includes the implementation level information of the aforementioned project.
- **Chapter 5**: The testing details of the final model is included in this chapter.
- **Chapter 6**: This part of the report contains the final conclusion drawn along with the future scope of this project.

Chapter 2

High-level Design

This chapter covers the engineering modules on which this project is build. First, we briefly define the incremental model , which includes several cycles after which the current version of the web app has been obtained. Then, there is the definition of agility that means regular status check of the project by the faculty panel and the project guide. We then briefly describe the Scrum, namely the regular meetings that we had with the team members.

2.1 Software Development Methodology

Software Development Life Cycle (SDLC) is a technique for setting up, creating and testing available programming from merchants (as in Figure 2.1). Sending better programming from the SDLC means that the client meets or exceeds the client's expectations. SDL is a technology used for a product's project, a product's association. This includes creating, maintaining, abusing, and improving programming features. The figure below is a graphic example of the various stages of a typical SDLC.

The detailed SDLC outline (Figure 2.1) shows how all the steps have been contributed to make the proposed work accurate and precise. A typical software development lifecycle consists of the following steps:

2.1.1 Stage 1: Planning and Requirement Analysis

The simulation test is the most important and central stage in the SDLC. This is done by the elderly people in the group with significant participation from clients, business offices, advertising studies, space specialist in this business. This data was later used to adjust the required operational procedures and to keep the devil's ability to understand effective, operational and specific zones. The process of standardized

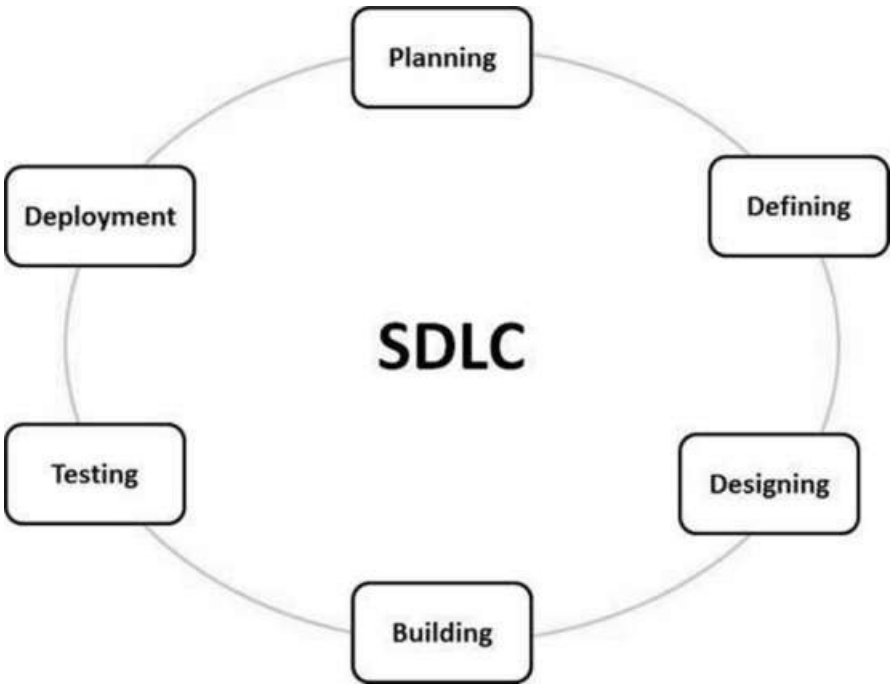


Figure 2.1 Software Development Life Cycle

certification requirements is tailored to the curriculum and evidence of risk-related risks similar to that in the Master Mining stage. The result considers the outcome of a very specific decision to describe important specific ways of insight, which can be done after a minute's careful understanding of the task.

2.1.2 Stage 2: Defining Requirements

Once the required test is completed the stage will then be able to image and report the needs of the product and get help from customers or marketing agents. This is done through an SSR (Software Requirements Explanation) report, which contains the essentials to be had and is created during the life cycle.

2.1.3 Stage 3: Designing the Product Architecture

A systematic approach clearly defines all the plan modules of the item accordingly with the appearance of the data flow and the external and non-modular models (to expect one). Within the framework of the significant number of proposed design models, the unconditional components in the DDS must be minimized.

2.1.4 Stage 4: Building or Developing the Product

Real change begins in this period of SDLC, and things are made. DDS generates programming code during these stage . If the arrangement is done in a positive way, the code can be a long life problem. The developers should go after their representation of these code rules and use their affiliation and programming gadgets such as code compilers, middlemen, checker debuggers, etc. to generate the code. Separate state-of-the-art programming programmers for coding, for example C, C ++, Pascal , Java, and Python are used. The programming language is used to select the type of computer programs to write.

2.1.5 Stage 5: Testing the Product

This stage is usually a part of a wider number of stages, as in the front-line SDLC model, testing processes are generally two-connected with each period of the SDLC. However, this phase checks that the bus, after representing something, has grown, settled and re-examined, until it meets the quality measures shown in the SRS.

2.1.6 Stage 6: Deployment in the Market and Maintenance

Once this item is tried and arranged to pass it is regularly issued in the right market. The remodeling now and again begins in phases, as evidenced by this affiliate business strategy. The first thing may be issued in a limited area and tried in a valid business state (UAT-User affirmation testing) then in view of the information, the item can be released as it is or parcel of promotion. With the proposed changes to focus on. After releasing the item into the market, its upbringing has improved the condition of existing customers.

2.2 Architecture

This report will provide a result that is distributed into three phases:

1. **Analysis Phase (Based on the Dataset):** At this stage, the main concentration is to examine the information from the data set and to illuminate the patient's medical data. At this stage, we try to analyze which medical data or medical values have the greatest impact on disease prognosis and which features have the least impact.

2. **Combining analysis stage with the parameters:** At this stage, we give some conditions based on the patient's condition (whether the patient is suffering from

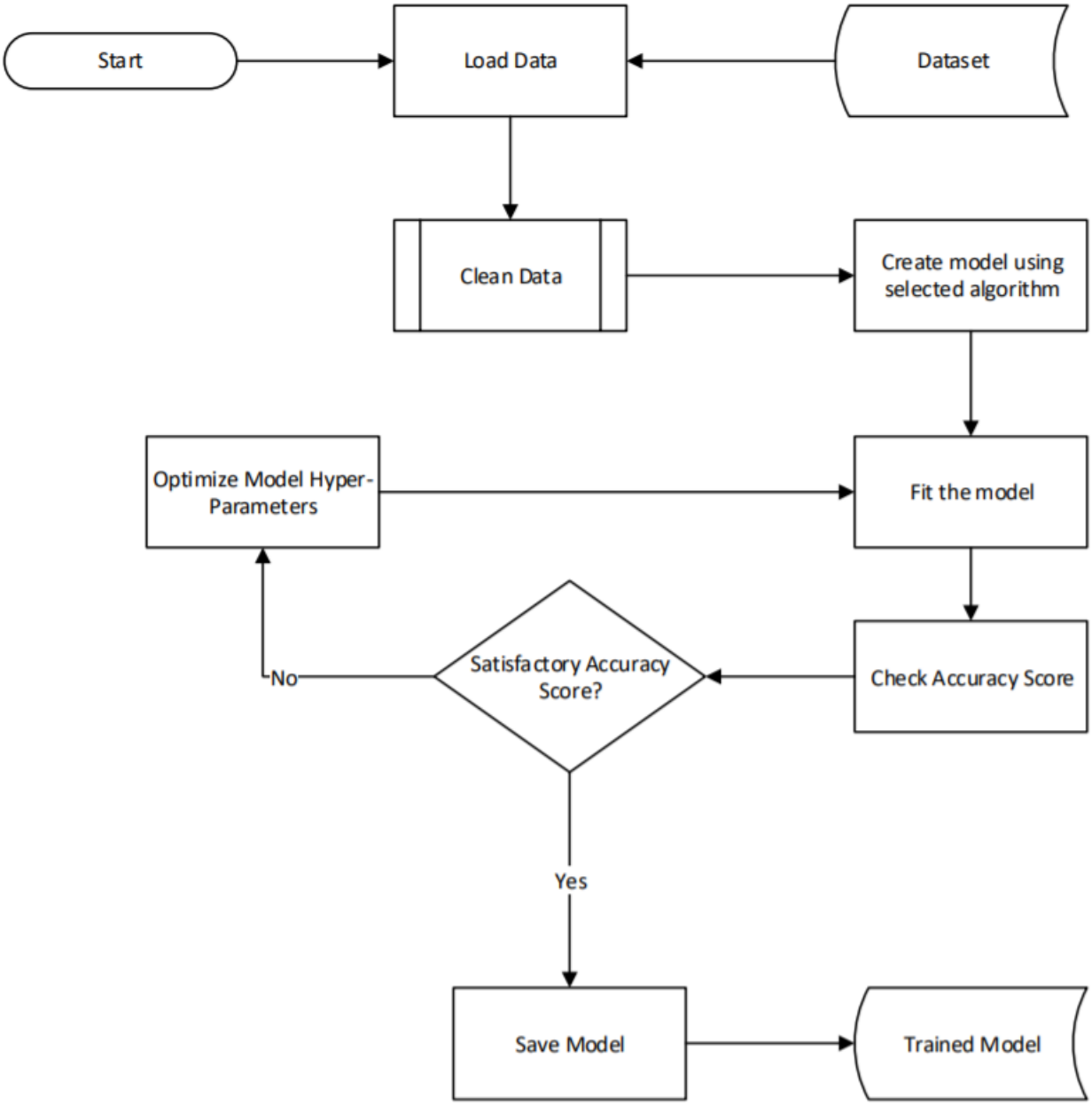


Figure 2.2: Initial design of the proposed system

heart disease and diabetes). The application applies various machine learning algorithms to generate a module, which in turn is used in the prediction process.

3. **Prediction Phase:** At this stage, we disclose the results and declare the possibility that the person is suffering from heart disease or diabetes. The results were predicted using different machine learning algorithms from user-defined values.

2.3 Incremental Model

The incremental frames represent a strategy for programming movement where something is designed, processed and maximized (until some degree more solid each time) until the task is completed. 2.2). It both turns and comes back down. Things get caught up when he caters to the more prominent part of his needs. This indicates that the datasets of the waterfall appeared on a civilizational basis of the prototype. The item is delivered in separate sections, each of which is organized and structured in an anonymized way (together with the name). Each part is sent to the client when it is completed. This serprise avoids the use of products and maintains a strategic space from long growth times. It likewise guarantees the removal of an important starting capital and the shortage period. This shows that growth spurt is at the same time promoting the frightening effects of modern systems at the same time.

Characteristics of the Incremental Model:

- Systems are isolated into different small units.
- Partial system are meant to deliver the ultimate framework.
- Firstly the required procedures are performed.
- What is required is cemented when an extended bit is produced.

After each cycle, backward testing is facilitated. In the midst of this test, the damaging parts of the product can be seen lightly so that some changes can be made internally at any one stress. It is, in general, easy to examine and explain the different approaches to the programming movement in light of the ways in which the eight-part changes are made today between each section. It focuses more on the center and takes the internal scrutiny of each section seriously. The customer seems to respond to highlighting and evaluating the product being used for any desired changes. The essential thing is the faster and lower cost of the accelerator model.

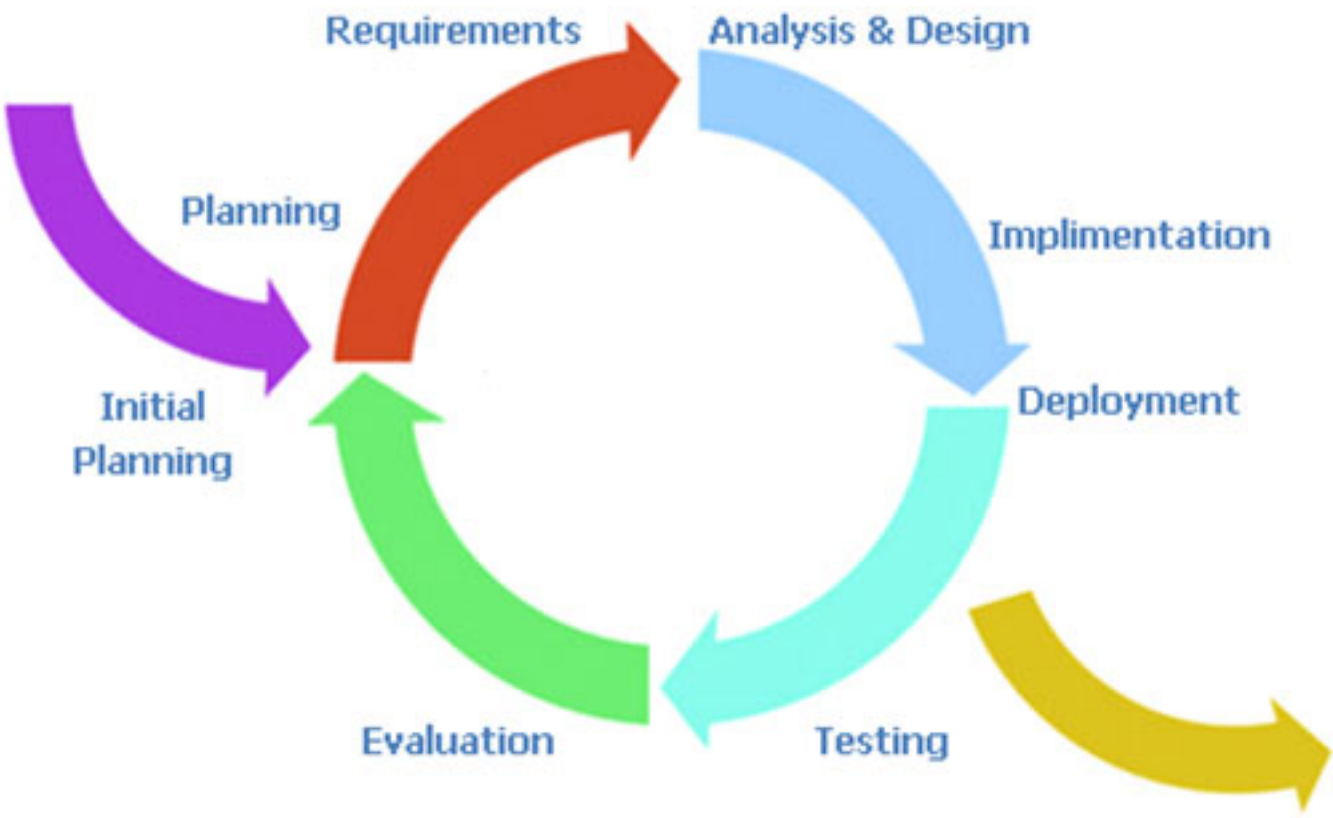


Figure 2.3: Incremental model for project

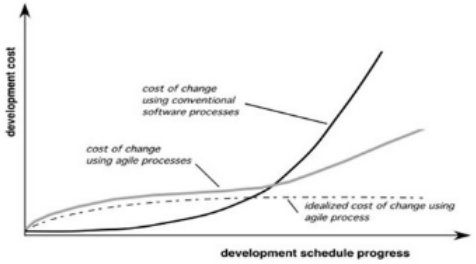


Figure 2.4: Agility and the cost of change

2.4 Agility and Scrum

Advanced programming can be a strategy for creating programming techniques (such as another programming reverse systems - waterfall model, V-model, incremental shows etc.). In English, detectors mean the ability to move quickly and easily, and reacting quickly is often an important piece of well-structured programming reversals. In traditional expert programming such as Waterfalls model, an outage can take several months and a long time because the client may not get the opportunity to see the end result of that commitment. In the exceptional case, the non-Agile assignments determine the timeframe for submission, arrangement, development, testing and client acceptance testing failures, spray work done sprints or attributes that are in term today (sprint) / Squares can move from 2 weeks to 2 months) in the midst of selected skills.

2.4.1 Agility and the cost of change

The general method of undertaking in programming movement is the advance increment nonlinearly as an endeavour advances (Figure 2.3). It is fairly fundamental to oblige and adjust when a thing pack is gathering essentials (exactly on time in a meander). A pre-owned circumstance must be changed, the rundown of the breaking point likely could be broadened, or made express could be changed. The expense of achieving this work is unimportant, and the exertion required won't ominously influence the outcome of the undertaking. Nevertheless, envision a condition where we fast forward different months. The group is amidst support testing (something that happens commonly late inside the wander), and a basic partner is requesting an essential reasonable change. The modify requires a change to the compositional organize of the thing, the diagram and movement of three present-day parts, alterations to another five segments, the game plan of unused tests, and so on.

Advantages of Agile Methodology:

- In Dexterous framework the advancement of making PC program is unremitting.
- The clients are fulfilled considering reality that after each Sprint, the working fragment of the thing is given to them.
- Clients can see the working part which fulfilled their needs.
- in the event that the customers have any analysis or any change inside the bit by then it tends to be obliged inside the show area of the thing.
- In Spry framework, the bit by bit affiliations are required between the bosses and the creators.
- In this structure thought is paid to the extraordinary chart of the thing.

2.5 Scrum

Scrum could be a fast structure for sorting out work with an element on programming improvement. It is sorted out gatherings of three to nine fashioners who break their work into works out that should be conceivable inside time-boxed cycles, called runs (routinely fourteen days) and track advance and re-graph in 15-minute stand-up social undertakings, called a tiny bit at a time scrums. Ways to deal with manage sifting through made by contrasting scrum packages in progressively basic affiliations concrete Large-Scale Scrum, Scaled Dexterous System (SAF) and Scrum of Scrums, among others.

2.5.1 Activities performed by the team in the scrum

- At the initial stages, extension and plan of the project are chosen.
- Regular commitment on a week by week premise to keep a mind all the exercises are in a state of harmony with one another.
- Engagement with the project guide assisted with doing the necessary adjustment of the project.
- Requirement gathering was done in a gradual manner.
- project was separated into various components with the goal that legitimate appropriation of work should be possible in the group.
- Each colleague is answerable for its doled out work.

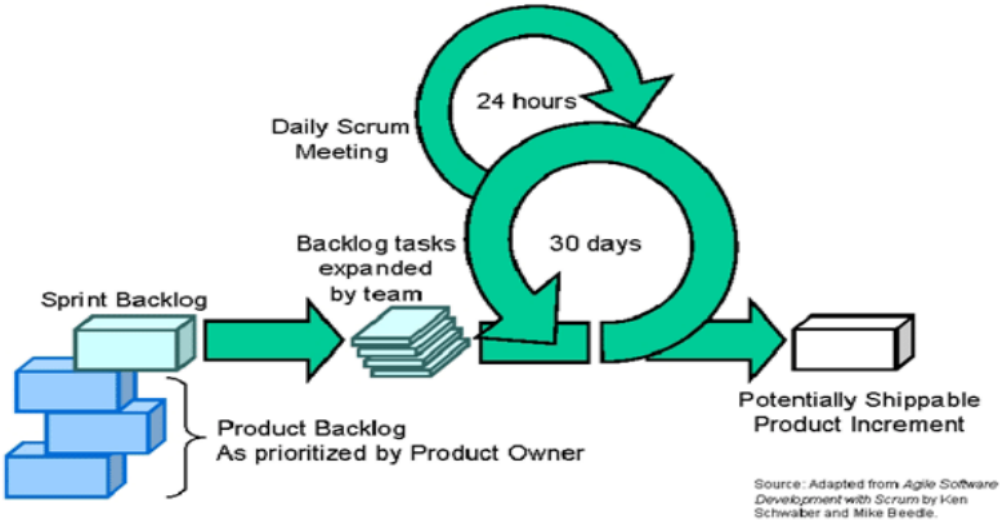


Figure 2.5: Scrum design of the project

- Every 14 days progress was checked and new objectives of the project were characterized to concentrate on.
- After the development of each module testing was done to ensure the best possible working of the module.
- All the components were coordinated to ensure everything functions admirably together.
- Report composing was done on a nonstop premise to catch all the outcomes and conversations.

Scrum consist of three roles: Product Owner, Scrum Master, and Team:

- **Item Owner:** The Item Owner ought to be cautious with exchange with vision, star, and responsiveness. The Item Proprietor is at risk for perseveringly giving the vision and necessities to the change gathering. It's every so often hard for Product Proprietors to strike the most ideal difference in thought. Since Scrum respects self-relationship among get-togethers, an Item Proprietor must battle the need. to humbler scale direct. Meanwhile, Item Proprietors must be available to answer ask from the get-together.
- **Scrum Master:** The Scrum Master goes round as helper for the Item Proprietor and the social gathering. The Scrum Ace doesn't deal with the social event.

The Scrum Ace attempts to rinse any obstructions that are blocking the social issue from satisfying its run goals. This secures in the social event to stay innovative and valuable while ensuring its triumphs are obvious to the Item Proprietor.

- **Team:** As showed up by Scrum's facilitator, "the social occasion is absolutely self-managing." The progression hard and fast is fit for self-sorting out to signify work. A Scrum advancement store up contains around seven completely dedicated individuals (authoritatively 3-9), preferably in one social event room ensured from outside redirections. For programming meanders, a standard social affair solidifies a blend of programming engineers, modelers, programming engineers, auditors, QA experts, analysers, and UI organizers.

2.6 Functional Requirements

- Predict the probability of Heart Disease and Diabetes with given user inputs.
- Contribute to the dataset or request to add functionalities.

2.7 Non-Functional Requirement

Non-rational necessities are prerequisites that exhibit rules that can be utilized to pass judgment on the activity of a structure, as opposed to explicit practices. This could be showed up particularly in association with important prerequisites that portray explicit direct or cutoff points.

Non-practical necessities are a noteworthy piece of the time called characteristics of a structure. Unmistakable verbalizations for utilitarian necessities are restrictions, quality attributes, quality targets, nature of association prerequisites and nonbehavioral fundamentals.

2.8 Feasibility Analysis

2.8.1 Technical feasibility

The project is technically feasible as it very well may be manufactured utilizing the currently accessible advances. It is an electronic application that utilizes the Grails Framework. The innovation required by Disease Predictor is accessible and consequently, it is technically feasible.

2.8.2 Economic feasibility

The project is economically feasible as the cost of the project is included uniquely in the deployment of the web-app. As the information tests expands, which devour additional time and handling power. All things considered, a superior processor may be required.

Chapter 3

Detailed Design

This section will cover the plan of our model in detail. Right off the bat with an interface plan that will give a nitty gritty clarification about the interface plan, and afterward with the Data Structure and Algorithms that have been utilized in the project. The entire point by point system plan with use case has been appeared in Fig 3.1.

3.1 Interface Design

This section depicts the client's connection with the interface. The between face plans/screen-shots have been included a request to give a superior perspective on the user interface.

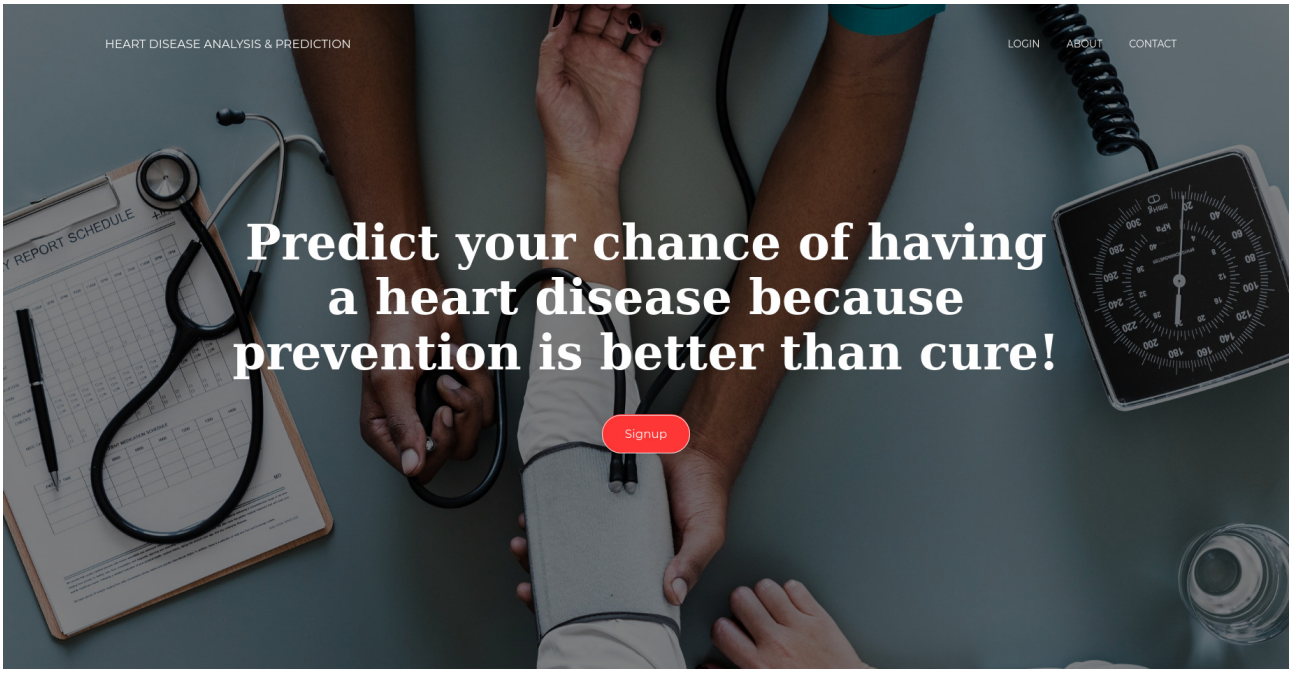
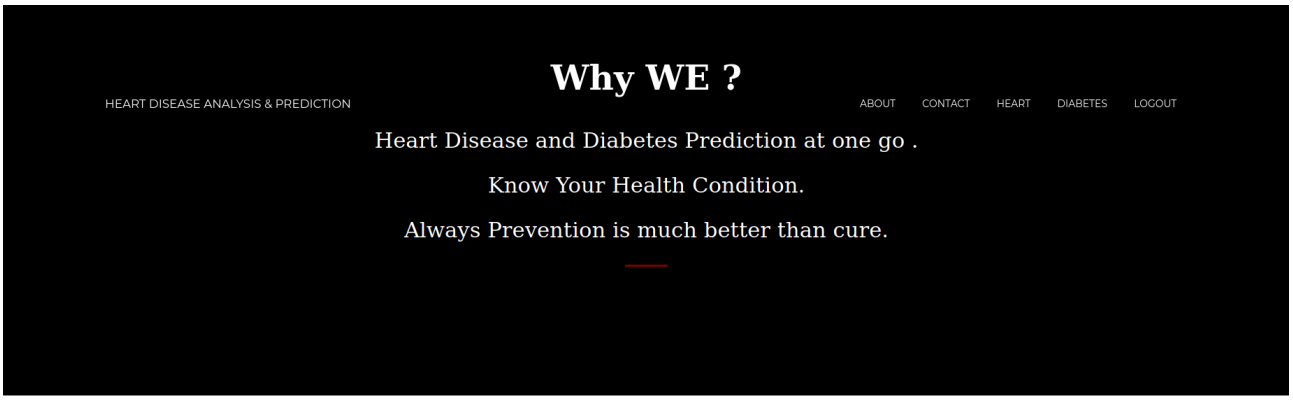


Figure 3.1: Home page



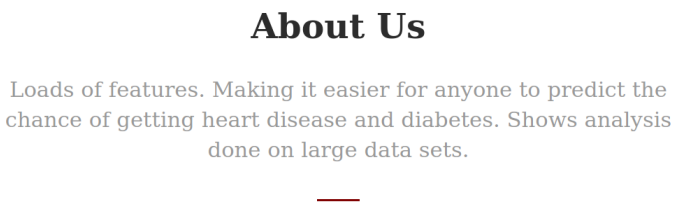


Figure 3.2: Home

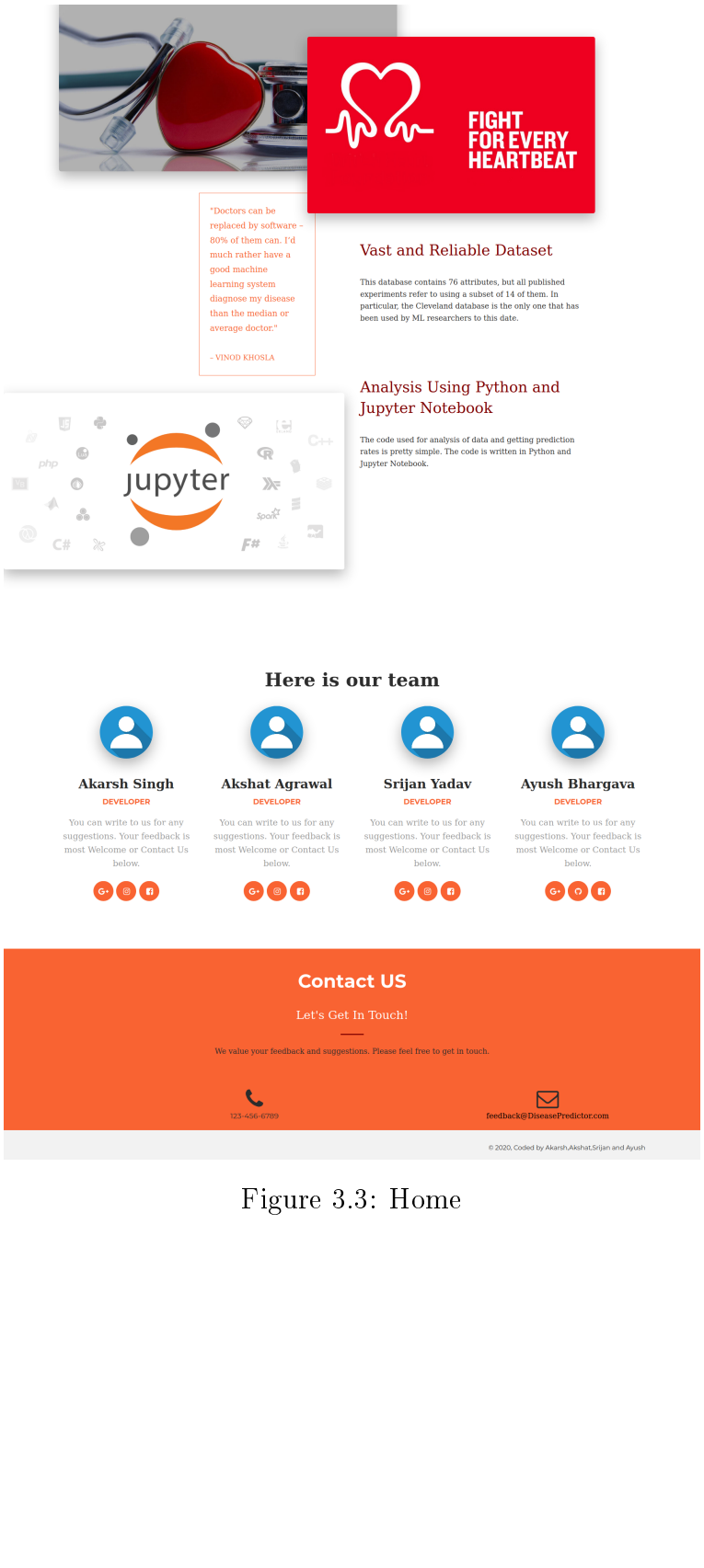


Figure 3.3: Home

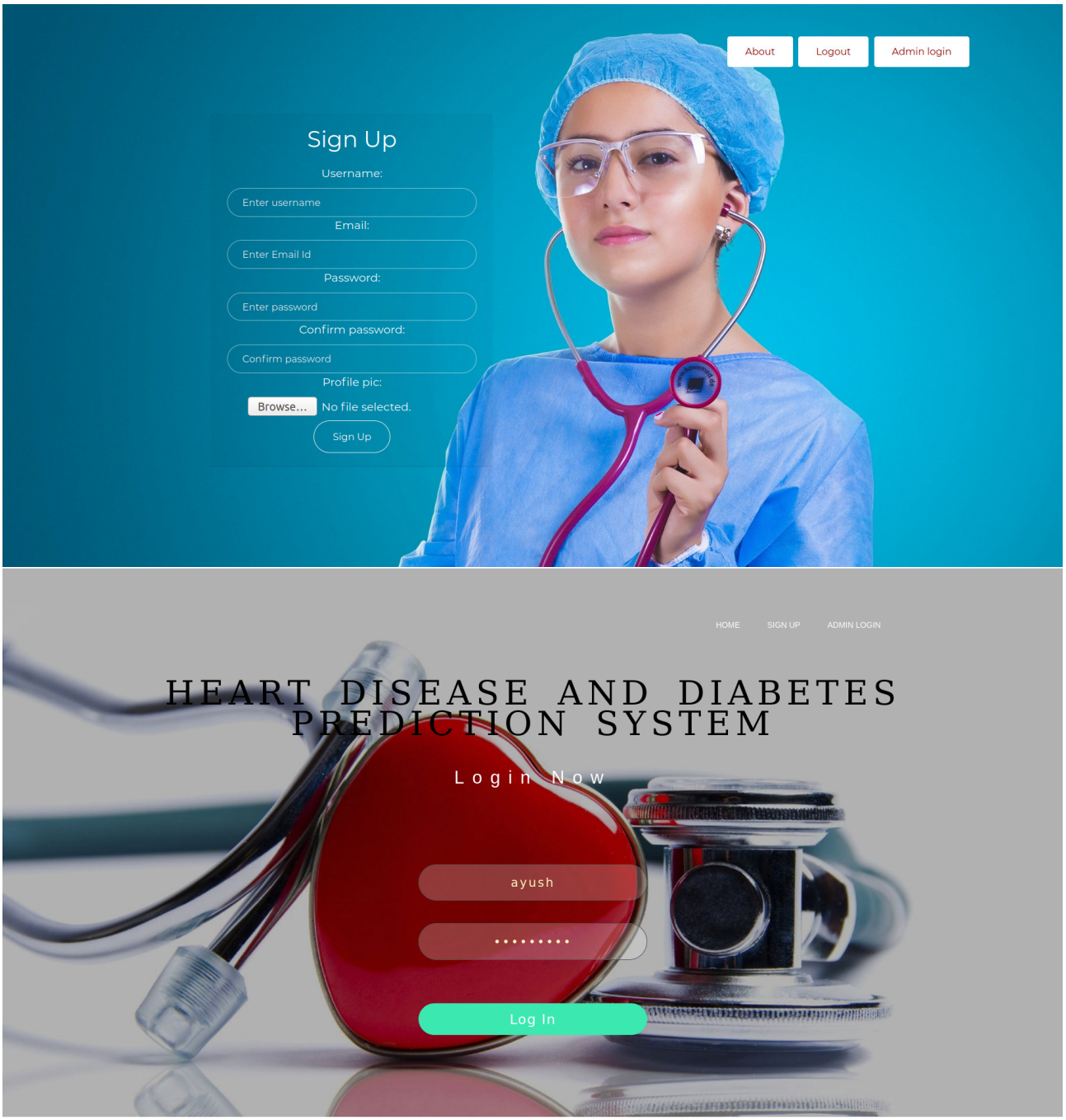


Figure 3.4: SignUp and Login feature

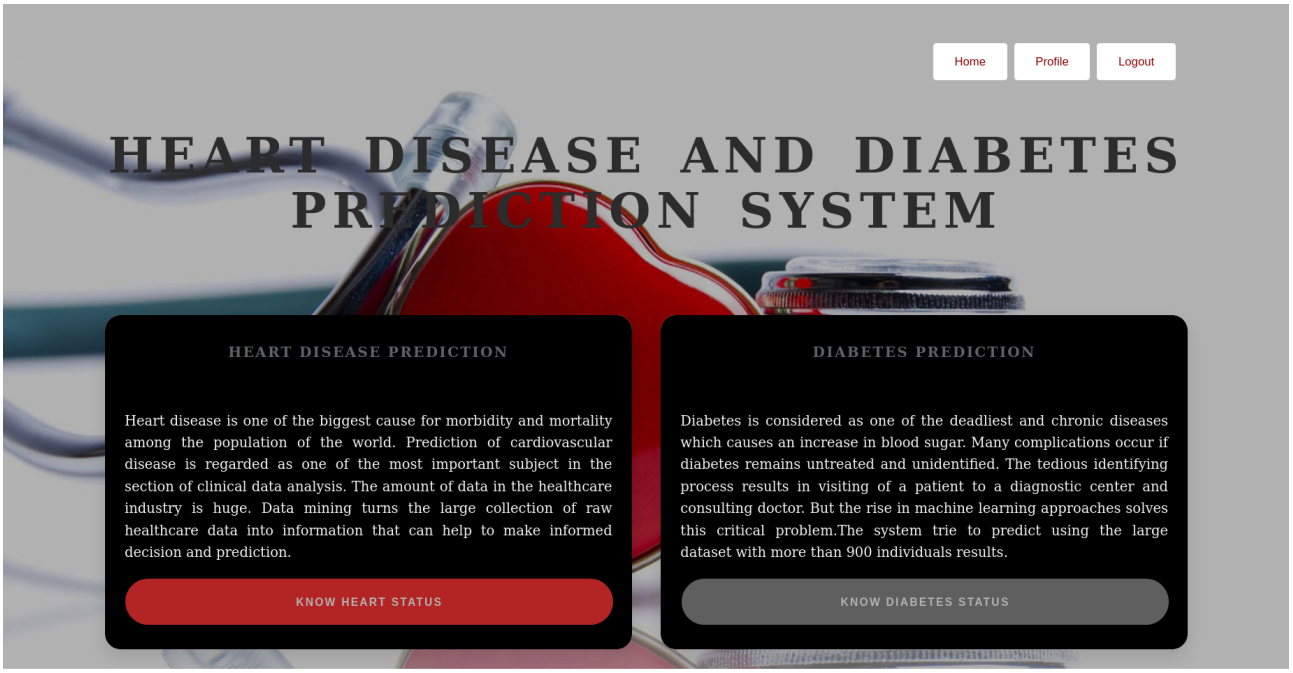


Figure 3.5: options of Prediction Engine

3.2 Data Structures and Algorithms

This section manages data structure and algorithms we have utilized in our project.

3.2.1 Naive Bayes Classifier

Naive Bayes classifiers are a gathering of straightforward probabilistic classifiers based by using Bayes theorem with strong (naive) freedom suppositions between the features. Naive Bayes classifiers are incredibly flexible by requiring different parameters direct for the number of features or pointers as a variable in a learning issue. It is the least perplexing and the snappiest probabilistic classifier, especially for the planning stage.

Naive Bayes classifier depends on Bayes theorem. This classifier utilizes restrictive autonomy wherein characteristic worth is autonomous of the estimations of different qualities. The Bayes theorem is as per the following:

Let X= x1, x2, ..., Xn be a lot of n qualities. In Bayesian learning, X is considered as proof and H be some speculation implies, the data of X has a place with explicit class C. We need to decide P (H|X), the likelihood that the speculation H holds given proof, for example, data test X. As indicated by Bayes theorem the P (H|X) is communicated as:

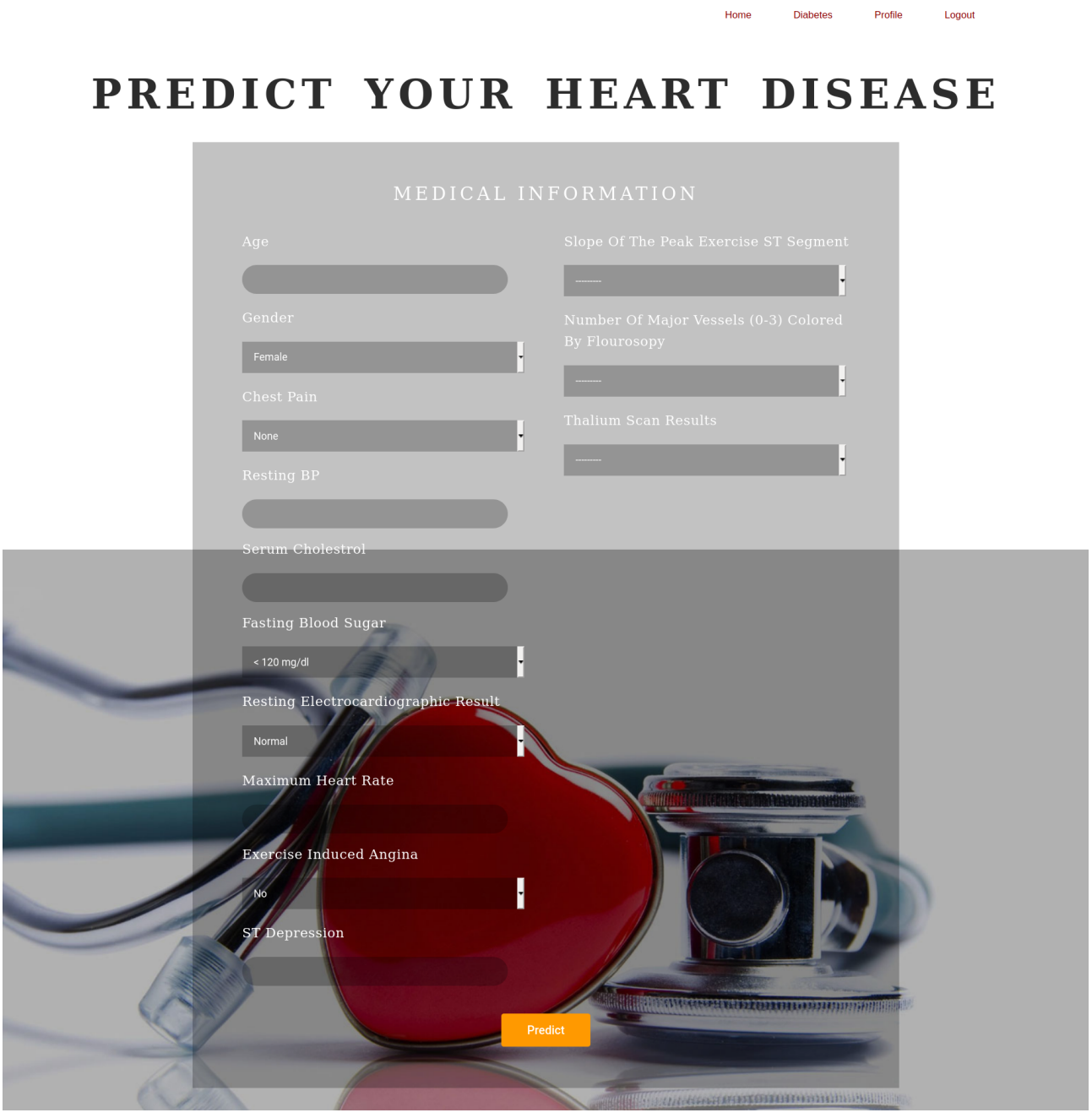


Figure 3.6: Heart diseases prediction form for patient

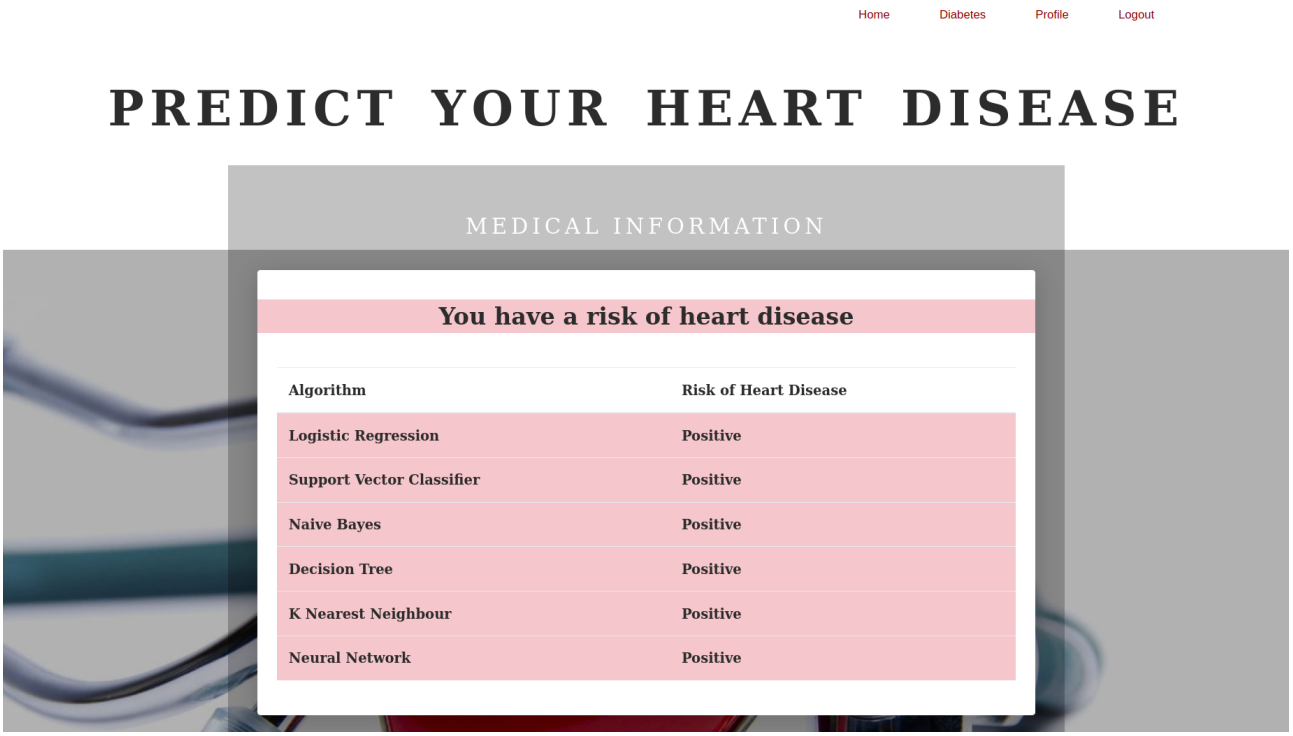


Figure 3.7: Heart diseases prediction result for patient

$$P(H|X) = \frac{P(X|H) * P(H)}{P(X)}$$

.

Utilizing Bayesian classifiers, the framework will find the hidden information related to diseases from authentic records of the patients having heart disease. Bayesian classifiers anticipate the class participation probabilities, such that the likelihood of a given example has a place with a specific class factually. A Bayesian classifier depends on Bayes' theorem. We can utilize Bayes theorem to decide the likelihood that a proposed analysis is right, given the perception. A basic probabilistic, the naive Bayes classifier is utilized for grouping dependent on which depends on Bayes' theorem. As per Naive Bayesian classifier, the event or an event of a specific element of a class is considered as independent in the occurrence or nonoccurrence of some other element. At the point when the element of the sources of info is the high and progressively productive outcome is normal, the boss Naive Bayes Classifier method is appropriate. The Naive Bayes model distinguishes the physical attributes and highlights of patients experiencing heart disease. For each info, it gives the chance of a property of the worthy state. Naive Bayes is a measurable classifier which expects no reliance between properties. This classifier calculation utilizes conditional independence, implies it ac-

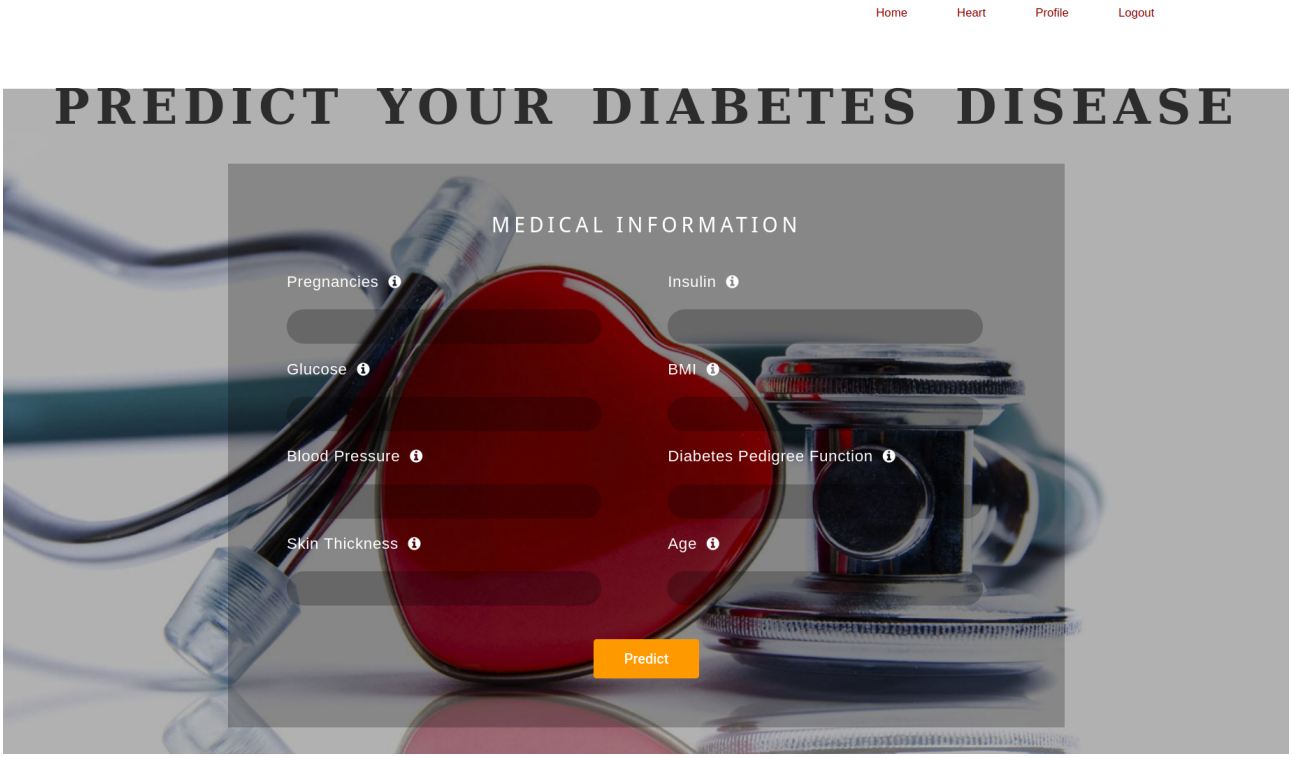


Figure 3.8: Diabetes prediction form for patient

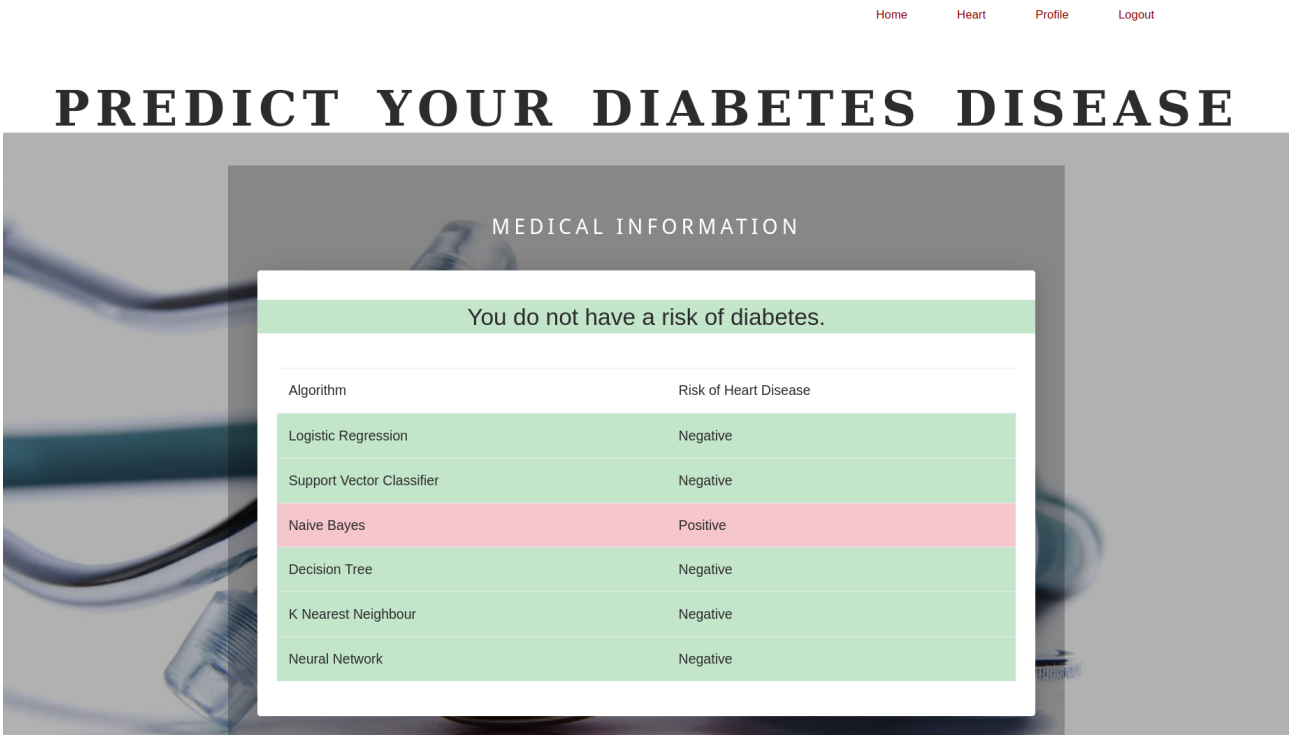


Figure 3.9: Diabetes prediction result for patient

cept that a quality estimation of a given class is free of the estimations of different qualities. The advantage of using Naive Bayes is that one can work with the Naive Bayes model without utilizing any Bayesian strategies. (Brownlee, 2016).

P (Disease|symptom1, symptom2, ..., symptomn) P(Disease)P(symptom1, ..., symptomn|Disease) = P(symptom1, symptom2, ...symptomnN).

3.2.2 Decision Tree

Decision tree learning uses a decision tree as a prescient model which maps discernments about a thing to decisions about the thing's objective. It is one of the prescient demonstrating approaches used in estimations, data mining and Artificial Intelligence. Tree models where the target variable can take a limited arrangement of values are called characterization trees. In these tree structures, leaves address class stamps and branches address conjunctions of features that lead to those class names. Decision trees where the target variable can take ceaseless values (customarily real numbers) are called regression trees. In decision tree analysis, a decision tree can be used to apparently and explicitly address decisions and decision making. In data mining, a

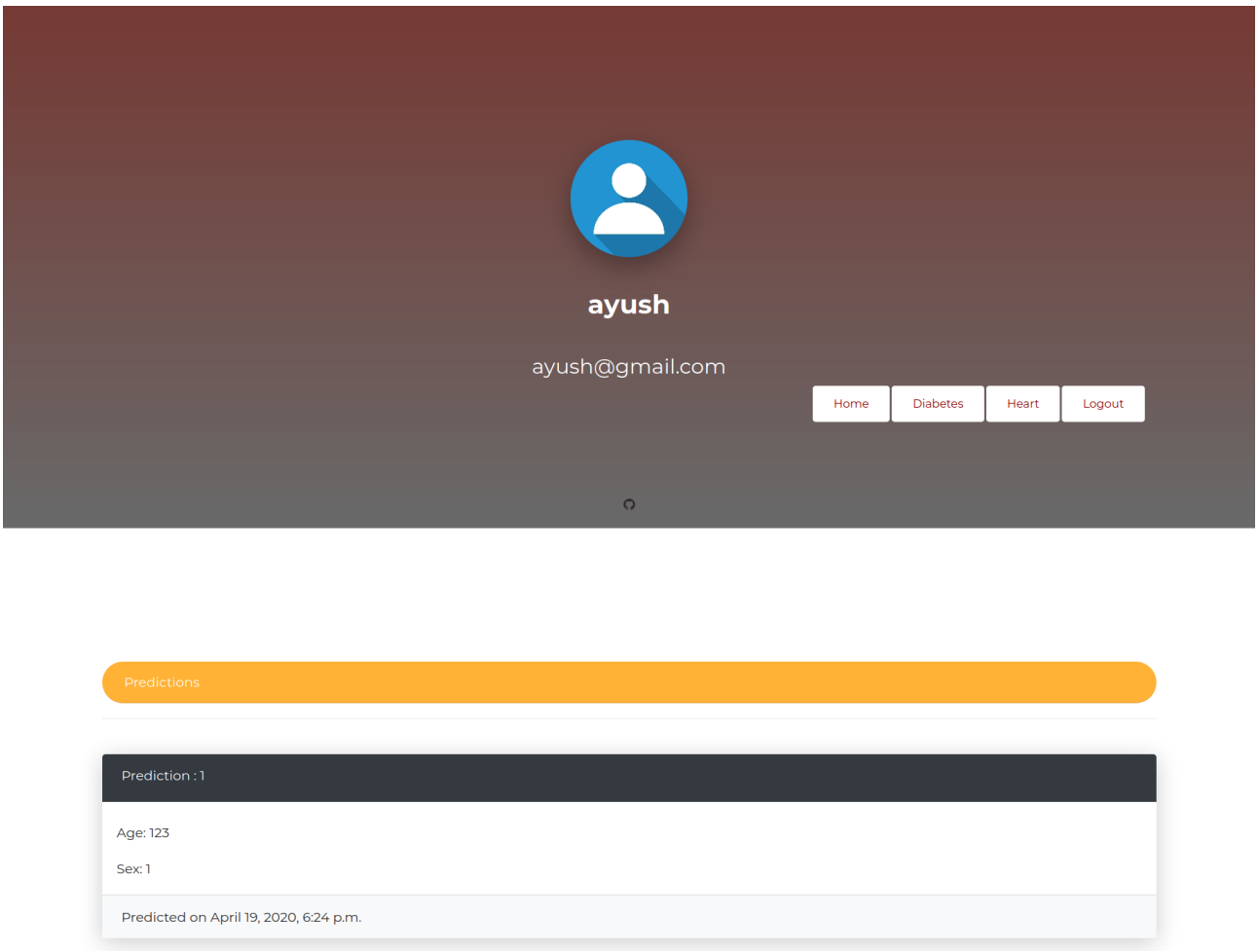


Figure 3.10: Profile of a patient showing the past results

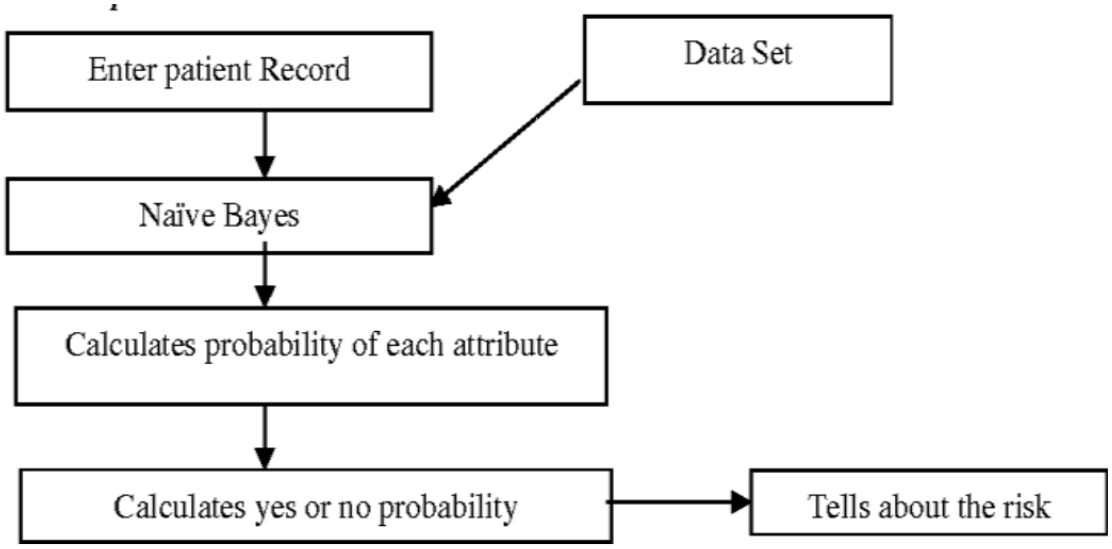


Figure 3.11: Implementation Flow of Naive Bayes Algorithm

decision tree portrays data yet not decisions; rather the resulting characterization tree can be a commitment for decision making.

The order tree makes a tree with branches, nodes, and leaves that let us take an obscure data point and plunge the tree, applying the attributes of the data point to the tree until a leaf is reached and the obscure yield of the data point can be resolved. To make a not too bad grouping tree model, we must have a current educational file with known yield from which we can manufacture our model. We also parcel our enlightening assortment into two sections: a preparation set, which is used to construct the model, and a test set, which is used to watch that the model is exact and not overfitted.

This classifier makes a decision tree dependent on which it doles out the class values to every datum point. Here, we can fluctuate the most extreme number of highlights to be thought of while making the model.

3.2.3 Support vector machine(SVM)

SVM was developed by Vladimir Vapnik at AT&T Bell Labs. It is based on the concept of decision planes that define decision boundaries. A decision plane is a hyperplane that separates the objects having different class memberships. SVM classifiers separate the observations into two or more classes in such a way that maximum separation is achieved. A hypothetical hyperplane is the separator in SVM classification problems. In other words, SVM constructs a hyperplane that separates the two sets so as to

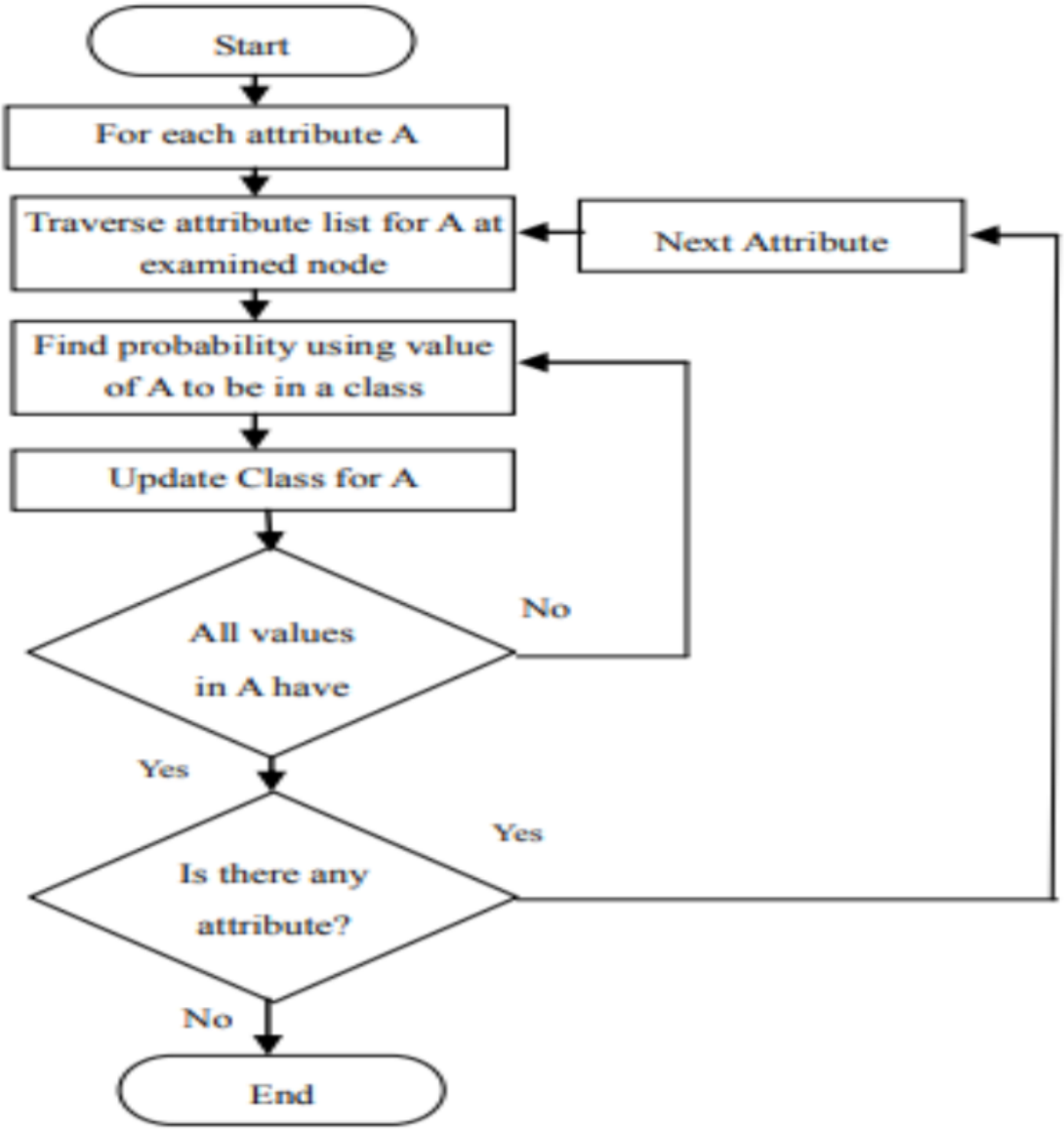


Figure 3.12: Flowchart for Decision Tree

minimize the number of misclassified points. Generally, there are two types of SVM models: linear and nonlinear. Linear SVM works better on linearly separable datasets but nonlinear SVM model works well even on hardly separable datasets. Since we are dealing with hardly separable data in our experiments we use nonlinear SVM. The dual formulation of the nonlinear SVM function can be formulated as

$$MaxW(\alpha) = \sum_{i=1}^{m} \alpha_i - 0.5 \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j K(x_i.x_j)$$

subject to:

$$\sum_{i=1}^{m} \alpha_i y_i = 0,$$

$$0 \leq \alpha_i \leq C$$

Input vectors $x_i \in R^m$, i= 1, 2, 3, ..., m, which are called features or attributes are extracted from the database. Associated with every particular input we have a corresponding label ($y_i = \pm 1$) which is called the target value or output in the database. The variable α_i is the Lagrange multiplier in the dual formulation and C is a user-specified parameter representing the penalty for misclassification K(xi ,xj) is the kernel function and maps the original data points to another space. One of the popular choices for the kernel is Gaussian kernel which is also known as Radial Basis Function (RBF) in the literature. The formulation for this kernel is

$$K(x_i, x_j) = e^{-\frac{\|x_i - x_j\|^2}{2\sigma}}$$

where parameter σ is known as the kernel width.

3.2.4 Logistic Regression

Logistic Regression is a statistical analysis procedure that is utilized for foreseeing the data esteem dependent on the earlier perception of the data set. The logistic regression model predicts the needy data variable by examining the connection between at least one existing free factors. Logistic Regression is one of the significant instruments for forecast, which can likewise be utilized for characterizing and foreseeing the data dependent on the historical data. The actualized model is a twofold Logistic model that has subordinate factors with just two potential results i.e., one is a positive worth and another is the negative worth which is having 0 or 1 as a class mark.

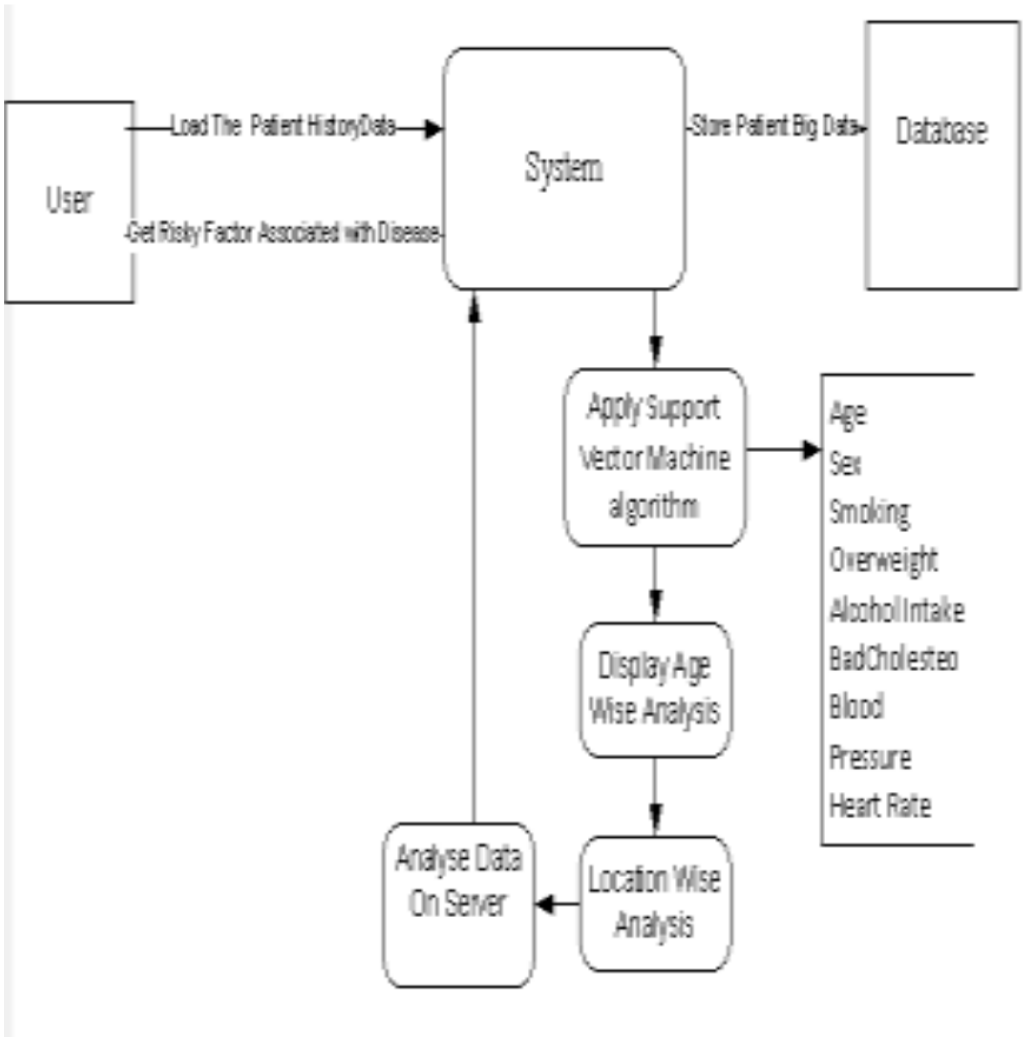


Figure 3.13: Flowchart For SVM

It for the most part comprises of two significant stages: regularized cost work and regularized angle plummet. Cost Function is utilized for ascertaining the greatest probability estimation. Angle plummet is an iterative procedure for getting coefficients from preparing

data. The procedure is rehashed until we get the ideal parameters of train data. The model is prepared with the ideal coefficient.Whenever a test data has been passed to the model dependent on the parameters can recognize whether the individual is having coronary illness or not, it tests the data utilizing the sigmoid capacity. The cost work is the technique that is utilized for decreasing the mistakes of the anticipated name and the genuine mark. Slope plunge work is the technique that is utilized for computing the coefficient until we get a base estimation of the class mark.

3.2.4.1 Cost Function

A minimisation work is used that is the cost work. It uses the Log Loss, for instance, the logarithmic misfortune which evaluates the presence of the model where the figure input regard is the probability between the zero and one. The Log loss is the vulnerability of the forecast which relies upon the sum it changes from the genuine name. Cost work which urges the student to address or change the direction to limit the mistakes. The cost capacity can be surveyed by iteratively running the model to take a gander at the anticipated worth and the known or real worth. The regularized cost work is a method that is used for enduring the risk of overfitting. Lamda is the parameter which controls the regularization term.
The cost function is calculated by the following:

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} y^{(i)} \log[h_\theta(x^{(i)}) - (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)})] + \frac{\lambda}{2m}\sum_{i=1}^{n} {\theta_j}^2$$

m = The number of instances
n = The number of attributes
y = The Class label
x = Features of the training data
θ = coefficients
λ = learning rate

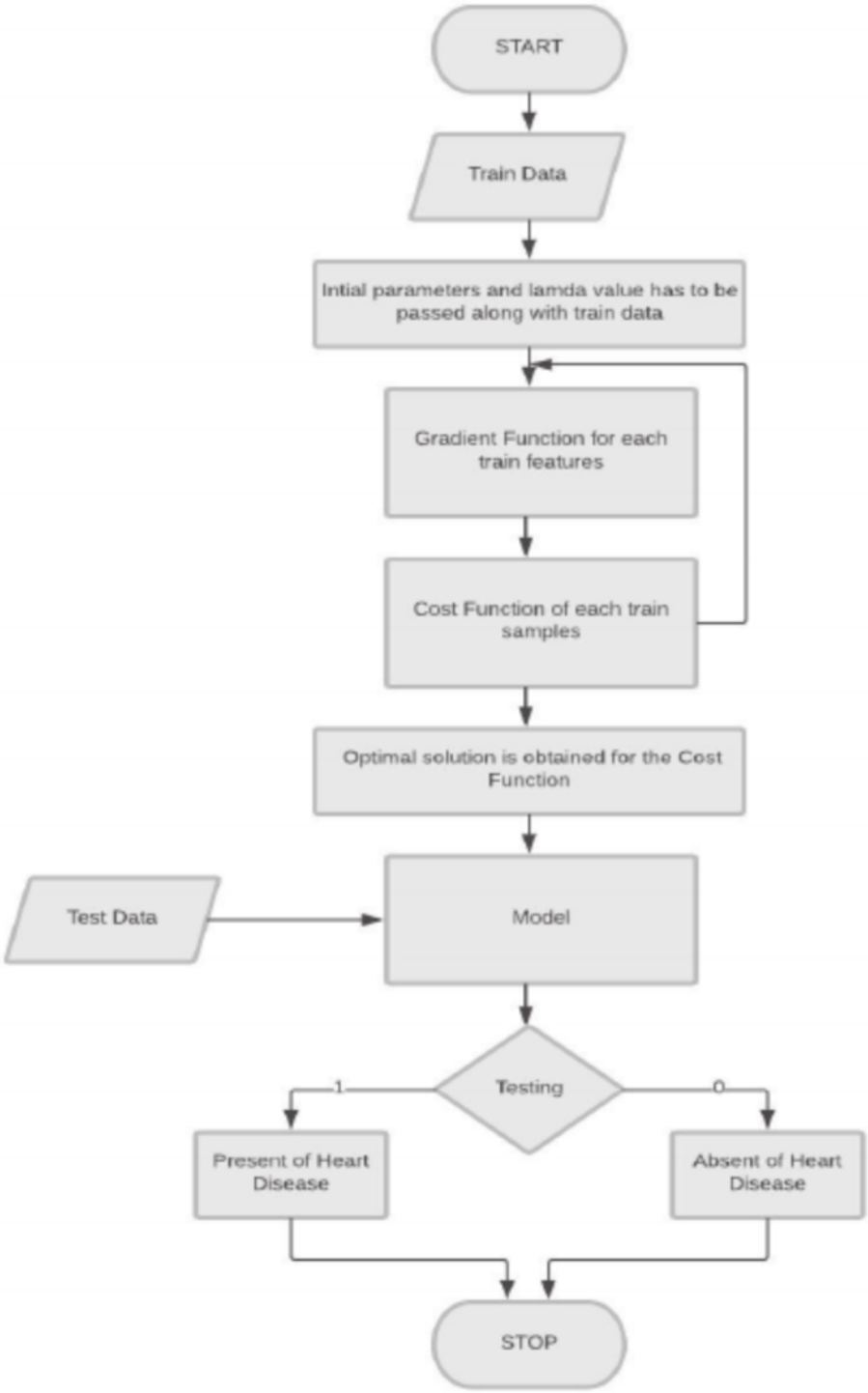


Figure 3.14: FlowChart For Logistic Regression

3.2.4.2 Gradient Descent

Gradient descent is an advanced strategy which is utilized to discover the parameters or the coefficient of the cost function. Gradient the descent is a rehashed procedure so as to get the coefficients to limit the cost function. The Gradient descent is determined for both the classes to get the pair of a coefficient for both classes marks. The objective here is to proceed with the strategy to attempt the unique esteem for the coefficient, assessing their cost and choosing the new coefficient that is having the somewhat lower cost. Thinking about this coefficient and putting away them in the model. Gradient descent is calculated as follows:

$$\theta_{ji} = \theta_j - \frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^i) - (y^i)){x_j}^i + \frac{\lambda}{m}\theta_j$$

m = The number of instances
x = Features of the training data
y = The class label
θ = coefficients
λ = Learning rate

3.2.4.3 Sigmoid Function

The sigmoid function is the logistic function between. This takes genuine information vales and yield values between the 0 and 1 for logistic function [12]. This is deciphered as taking log chances and having the yield probability. For the most part, the sigmoid function is utilized to delineate to the probability it is characterized as:

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^{T}.x}}$$

x = test data features
θ = coefficients
At whatever point a test data is passed it figures the worth dependent on the parameters put away in the model. It computes the probability of each class label. We return the most extreme probability estimation of the class label x_i.

The test data contains the thirteen ascribes that we have to pass and compute for both the classes it will restore the two values we take the most extreme estimation of two values we will restore the class with a label which is having the greatest probability.

3.2.5 K-Nearest Neighbour

K-Nearest Neighbor (KNN) is a straightforward, lazy and nonparametric classifier. KNN is favoured when all the highlights are consistent. KNN is additionally called case-based thinking and has been utilized in numerous applications like example acknowledgement, statistical estimation. Classification is acquired by distinguishing the nearest neighbour to decide the class of an obscure example. KNN is favoured over other classification algorithms because of its high combination speed and simplicity.

KNN classification has two phases:

1. Find the k number of instances in the dataset that is closest to instance S
2. These k number of instances then vote to determine the class of instance S

The Accuracy of KNN relies upon separation metric and K esteem. Different methods of estimating the separation between the two cases are cosine, Euclidean separation. To assess the new obscure example, KNN processes its K nearest neighbours and dole out a class by greater part voting.

With the KNN algorithm, we have the opportunity to change the parameter's weight. It implies that we may accept that a few parameters are more significant or having more effect than others. Among 8 parameters we use, we can classify them our data into 2 classifications, one is "non-clinical" parameters (Age and Sex) and the other is "clinical" parameters (CP, Trestbps, Trestbpd and so forth). We may feel that clinical parameters are a higher priority than non-clinical, which we will see in test results. Alongside weighting, we should discover the estimation of "k" so it gives the best classification result. Since it is a 2-decision classification ("yes") and ("No") k will be an odd number.

3.2.6 Neural Network

3.2.6.1 Multilayer Perceptron Neural Network (MLPNN)

One of the most important models in Artificial Neural Network is Multilayer Perceptron (MLP). The type of architecture used to implement the system is Multilayer Perceptron Neural Network (MLPNN).

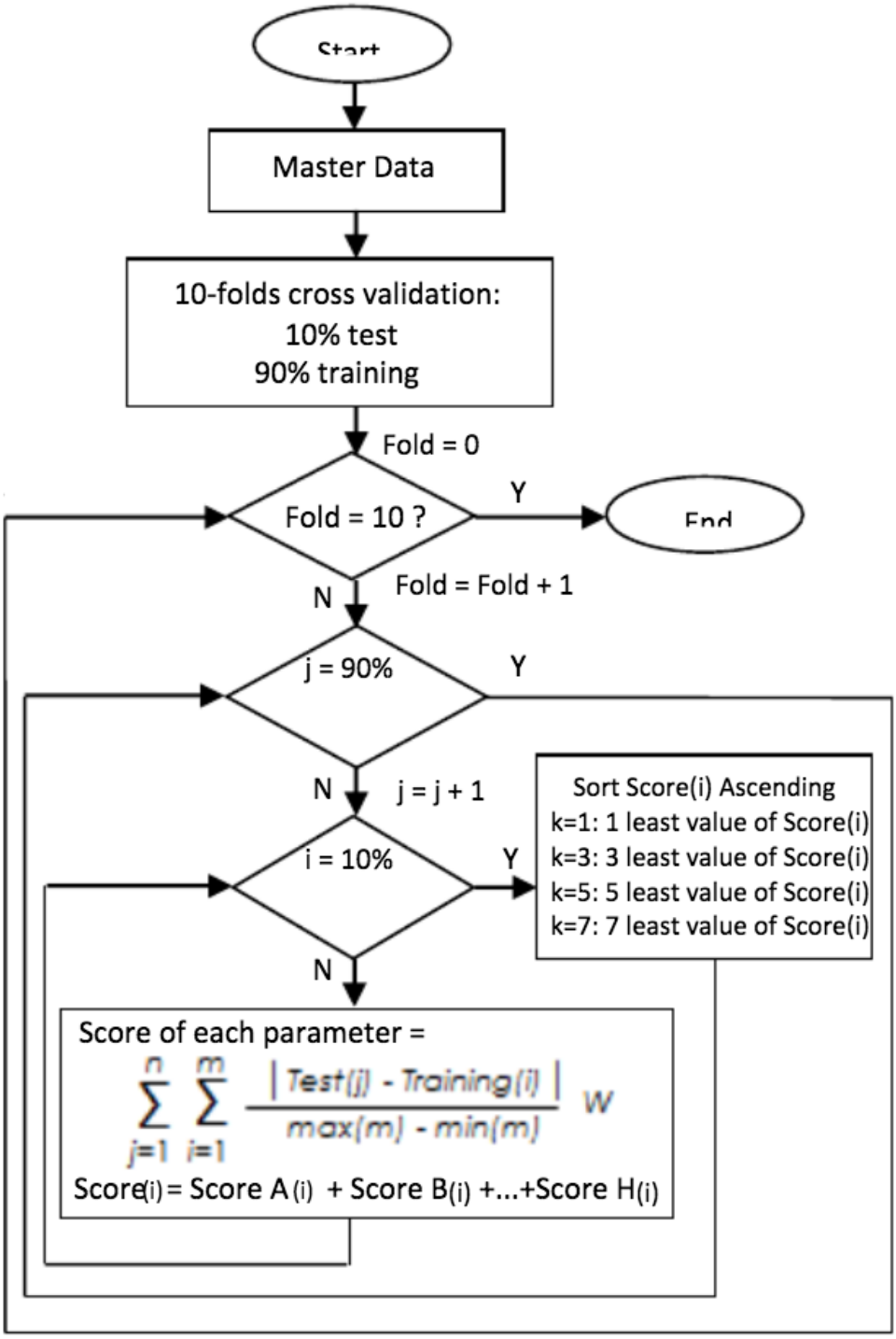


Figure 3.15: FlowChart for KNN

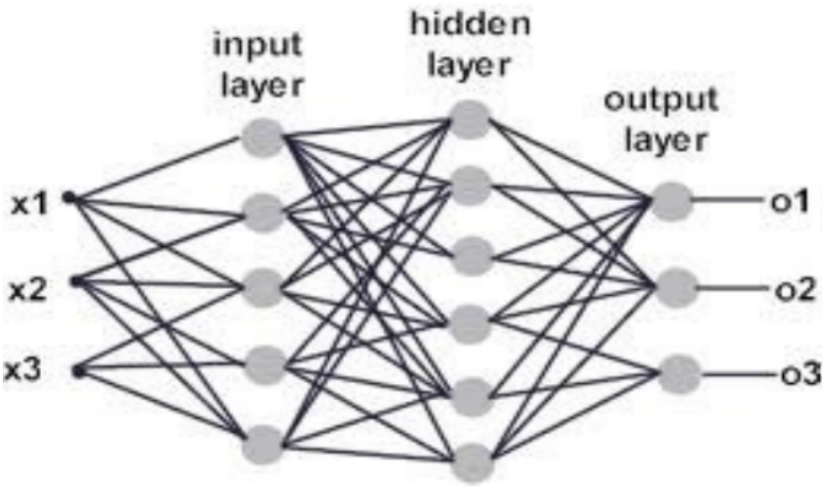


Figure 3.16: Neural Network

The MLPNN comprises of one input layer, one output layer and at least one hidden layers. Each layer comprises of at least one hubs, spoke to by little circles. The lines between hubs demonstrate stream of data starting with one hub then onto the next hub. The input layer gets signals from outside hubs. The output of input layer is given to hidden layer, through weighted association joins. It performs calculations and transmits the outcome to output layer through weighted connections. The output of hidden layer is sent to output layer, it performs calculations and produce final outcome. The working of multilayer perceptron neural system is summed up in ventures as referenced below:

1. Input data is given to the input layer to handling, which creates an anticipated output.

2. The anticipated output is deducted from the real output and blunder esteem is determined.

3. The system at that point utilizes a Backpropagation algorithm which changes the loads.

4. For loads modifying it begins from loads between output layer nodes, what's more, last hidden layer hubs and works in reverse through the system.

5. When back engendering is done, the sending procedure begins once more.

6. The procedure is rehashed until the blunder among the anticipated and real output is limited.

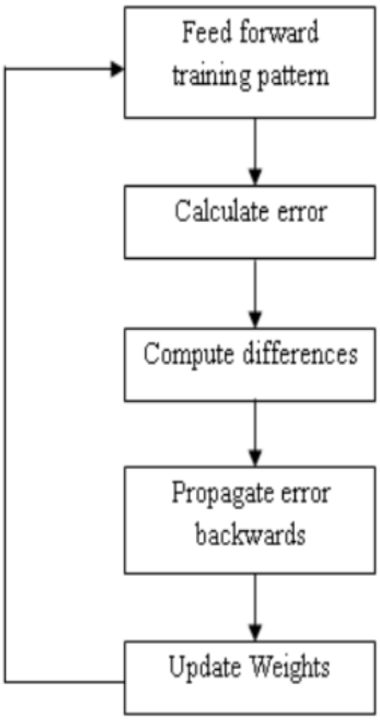


Figure 3.17: BackPropagation

3.2.6.2 Backpropagation network

The most broadly utilized preparing algorithm for multilayer and feed-forward system is Backpropagation. The name given is backpropagation since it computes the distinction among real and anticipated values is proliferated from output hubs in reverse to hubs in the past layer. This is done to improve loads during handling. The working of the Backpropagation algorithm is summed up in ventures as follows:

1. Provide preparing data to the network.
2. Compare the genuine and wanted output.
3. Calculate the mistake in every neuron.
4. Calculate what output ought to be for every neuron and how much lower or higher output must be balanced for wanted output.
5. Then alter the loads

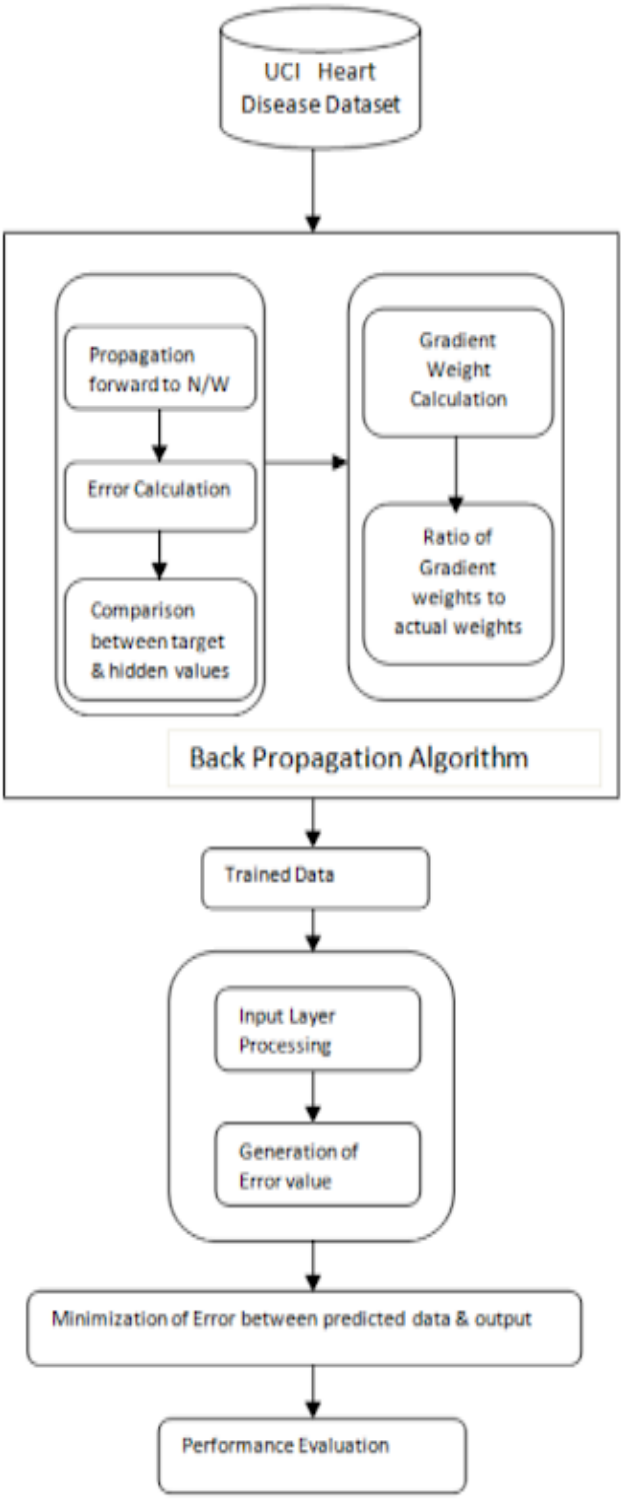


Figure 3.18: Flowchart for Neural Network

3.3 UML diagrams with discussions

UML is notable for its diagrammatic documentation(Refer Fig 3.10). All in all understand that UML is for imagining, demonstrating, building and recording the portions of programming and non-programming structures. In this way, recognition is the most basic part which ought to be grasped and remembered. UML documentation is the most basic components in illustrating. Viable and reasonable use of documentations is basic for making aggregate and huge model. The model is trivial, with the exception of if its inspiration is depicted really. Subsequently, taking in documentations should be underlined from the most punctual beginning stage. Various documentation is available to things and associations. UML charts are made using the documentations of things and associations. Extensibility is another fundamental part which makes UML even more prevailing and versatile. The model's UML additionally is very instinctive and plain as day, this UML have clear and particular classes which makes it effectively justifiable

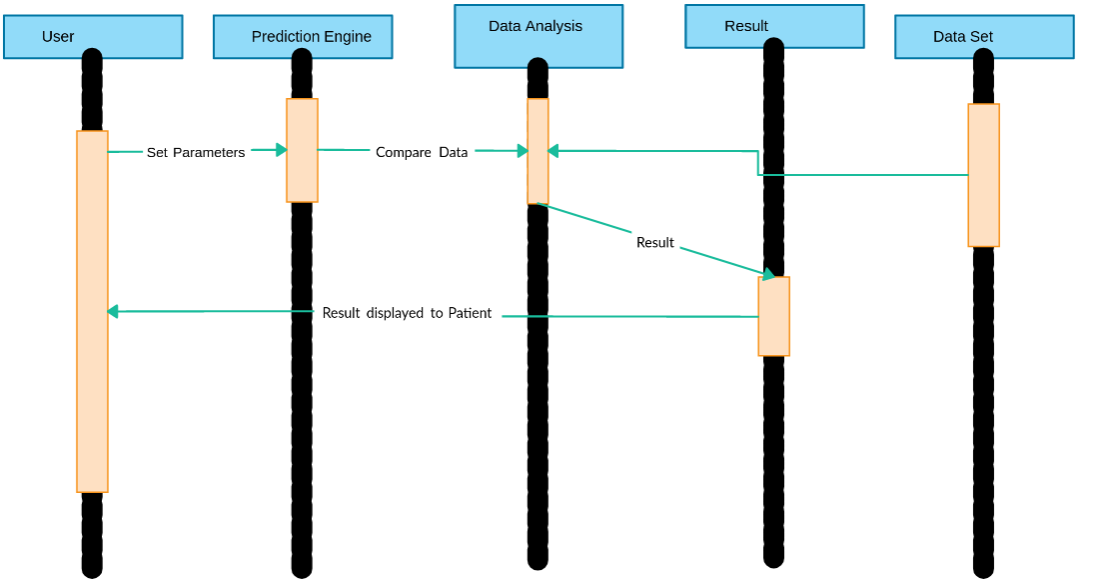
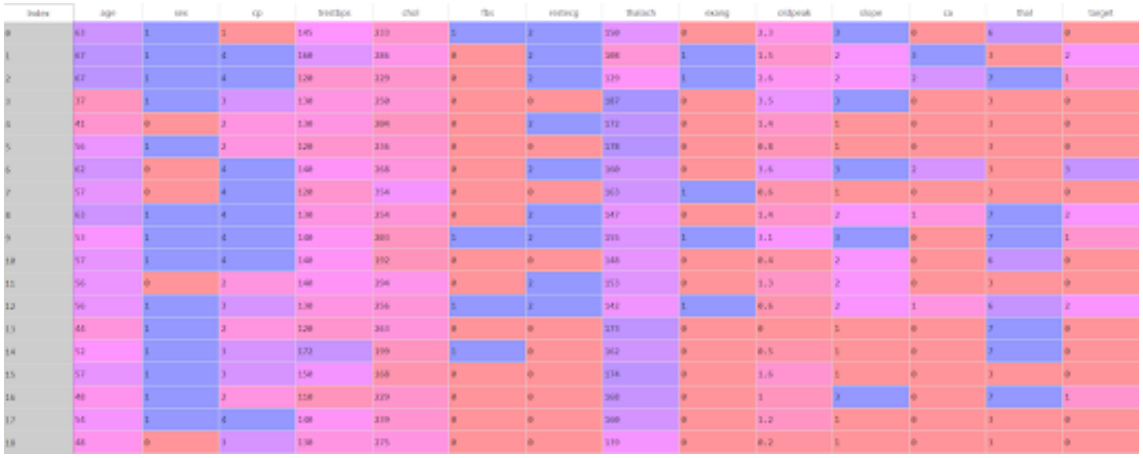


Figure 3.19: UML Sequence Diagram

3.4 Data Source and Formats

3.4.1 Heart Disease DataSet

The dataset used in this project is the Cleveland Heart Disease dataset taken from the UCI repository.



The dataset consists of 303 data points. There are 14 features in the dataset, which are described below.

1. **Age:** It consists of the age of the individual.
2. **Sex:** Shows the gender of the individual in the following format:
   - 1 = Male
   - 0 = Female
3. **Chest-pain type:** displays the type of chest-pain experienced by the individual using the following format :
   - 1 = typical angina
   - 2 = atypical angina
   - 3 = non — anginal pain
   - 4 = asymptotic
4. **Resting Blood Pressure**: displays the blood pressure value of an individual in mmHg (unit)
5. **Serum** Cholestrol: displays the serum cholesterol in mg/dl (unit)
6. **Fasting Blood Sugar**: compares the fasting blood sugar value of an individual with 120mg/dl.

- If fasting blood sugar > 120mg/dl then, : 1 (true)
- else : 0 (false)

7. **Resting ECG:** displays resting electrocardiographic results
   - 0 = normal
   - 1 = having ST-T wave abnormality
   - 2 = left ventricular hyperthrophy
8. **Max heart rate achieved:** displays the max heart rate achieved by an individual.
9. **Exercise induced angina** :
   - 1 = yes
   - 0 = no
10. **ST depression induced by exercise relative to rest:** displays the value which is an integer or float.
11. **Peak exercise ST segment:**
    - 1 = upsloping
    - 2 = flat
    - 3 = downsloping
12. **Number of major vessels (0–3) colored by flourosopy :** displays the value as integer or float.
13. **Thal :** displays the thalassemia :
    - 3 = normal
    - 6 = fixed defect
    - 7 = reversible defect
14. **Diagnosis of heart disease :** Displays whether the individual is suffering from heart disease or not :
    - 0 = absent
    - 1, 2, 3, 4 = present.

Why these parameters: In the actual dataset, we had 76 features but for our study, we chose only the above 14 because :

1. **Age:** Age is the most significant hazard factor in creating cardiovascular or heart ailments, with roughly a significantly increasing of hazard with every time of life. Coronary greasy streaks can start to shape in immaturity. It is assessed that 82 per cent of individuals who pass on of coronary illness are 65 and more established. At the same time, the danger of stroke pairs each decade after age 55.

2. **Sex:** Men are at more serious danger of coronary illness than pre-menopausal ladies. Once past menopause, it has been contended that a lady's hazard is like a man's albeit later data from the WHO and UN questions this. On the off chance that a female has diabetes, she is bound to create coronary illness than a male with diabetes. Angina (Chest Pain): Angina is chest torment or uneasiness caused when your heart muscle doesn't get enough oxygen-rich blood. It might feel like a weight or pressing in your chest. The distress likewise can happen in your shoulders, arms, neck, jaw, or back. Angina torment may even feel like heartburn.

3. **Resting Blood Pressure:** After some time, high blood pressure can damage supply routes that feed your heart. Hypertension that happens with different conditions, for example, stoutness, elevated cholesterol or diabetes, increases your risk significantly more.

4. **Serum Cholesterol:** An elevated level of low-thickness lipoprotein (LDL) cholesterol (the "bad" cholesterol) is well on the way to limit conduits. A significant level of triglycerides, a sort of blood fat identified with your eating routine, likewise ups your danger of coronary failure. In any case, a significant level of high-thickness lipoprotein (HDL) cholesterol (the "great" cholesterol) brings down your danger of cardiovascular failure.

5. **Fasting Blood Sugar:**Not delivering a sufficient hormone emitted by your pancreas (insulin) or not reacting to insulin appropriately causes your body's glucose levels to rise, expanding your danger of cardiovascular failure.

6. **Resting ECG:** For individuals at generally safe of cardiovascular illness, the USPSTF finishes up with moderate conviction that the potential damages of screening with resting or exercise ECG approach or surpass the potential advantages. For individuals at middle of the road to high hazard, current proof is deficient to evaluate the parity of advantages and damages of screening.

7. **Max heart rate achieved:** The expansion in cardiovascular hazard, related to the quickening of the pulse, was similar to the increment in chance saw with

hypertension. It has been demonstrated that an expansion in pulse by 10 beats for each moment was related with an increment in the danger of cardiovascular passing by in any event 20%, and this increment in the hazard is like the one saw with an increment in systolic blood pressure by 10 mm Hg.

8. **Exercise induced angina:** The pain or uneasiness related to angina, for the most part, feels tight, grasping or pressing, and can fluctuate from gentle to extreme. Angina is normally felt in the focal point of your chest however may spread to either or both of your shoulders, or your back, neck, jaw or arm. It can even be felt in your grasp. o Types of Angina a. Stable Angina/Angina Pectoris b. Flimsy Angina c. Variation (Prinzmetal) Angina d. Microvascular Angina.

9. **Peak exercise ST segment:** A treadmill ECG stress test is viewed as anomalous when there is an even or down-slanting ST-section melancholy ≥ 1 mm at 60–80 ms after the J point. Exercise ECGs with up-slanting ST-section miseries are commonly revealed as a 'dubious' test. All in all, the event of flat or down-inclining ST-portion sadness at a lower outstanding task at hand (determined in METs) or pulse shows a more awful guess and a higher probability of multi-vessel malady. The length of ST-portion wretchedness is likewise significant, as drawn-out recuperation after pinnacle pressure is steady with a positive treadmill ECG stress test. Another finding that is profoundly demonstrative of huge CAD is the event of ST-fragment rise > 1 mm (regularly proposing transmural ischemia); these patients often allude desperately for coronary angiography.

3.4.2 Diabetes DataSet

The dataset used in this project is the dataset taken from Kaggle which consist of 768 individual data with 8 column variables.
Description of variables in the dataset:

1. **Pregnancies:** The number of times the given individual has been pregnant.

2. **Glucose:** Plasma glucose concentration over 2 hours in an oral glucose tolerance test.

3. **BloodPressure:** Diastolic blood pressure (mm Hg).

4. **SkinThickness:** Triceps skin fold thickness (mm).

5. **Insulin:** 2-Hour serum insulin (mu U/ml).

6. **BMI:** Body mass index $(weightinkg)/(heightinm)^2$

7. **DiabetesPedigreeFunction:** Diabetes pedigree function(a function which scores likelihood of diabetes based on family history).
8. **Age:** Age (years).
9. **Outcome:** Class variable.
    - 0 = absent
    - 1 = present

Chapter 4

Implementation

4.1 Tools and Technologies

4.1.1 Django

Django styles itself as "an elevated level Python WebApplication system that supports fast turn of events and perfect, down to business plan. Worked by experienced creators, it manages an incredible piece of the issue of web improvement, so you can focus on forming your application without hoping to sit around idly." And they truly would not joke about this! This enormous web system accompanies such a significant number of batteries incorporated that as a rule during advancement it very well may be a riddle concerning how everything figures out how to cooperate. Notwithstanding the system itself being enormous, the Django people group is totally gigantic. Actually, it's so enormous and dynamic that there's an entire site dedicated to the outsider bundles individuals have intended to plug into Django to do an entire host of things. This incorporates everything from confirmation and approval, to all out Django-controlled substance the executives frameworks, to web based business additional items, to combinations with Stripe. Discussion about not re-developing the wheel; odds are on the off chance that you need something finished with Django, somebody has just done it and you can simply maneuver it into your venture.

4.1.2 Python

Python is a deciphered, huge stage, comprehensively valuable programming language. Made by Guido van Rossum and first released in 1991, Python's arrangement hypothesis highlights code coherence with its unmistakable usage of colossal whitespace. Its language manufactures and thing arranged procedure expect to help programming engineers with forming clear, genuine code for nearly nothing and huge extension ad-

ventures.

Python is progressively composed and trash gathered. It bolsters various programming standards, including organized (especially, procedural), object-arranged, and hands-on programming. Python is regularly portrayed as a "batteries included" language because of its extensive standardised library.

Python was considered in the late 1980s as a substitution to the ABC language. Python 2.0, released in 2000, introduced features like overview understandings and a refuse combination system fit for social occasion reference cycles. Python 3.0, released in 2008, was a huge revision of the language that isn't absolutely in invert great.

4.1.3 SqlLite

SQLite is a little database application that is utilized in a great many programming and gadgets. SQLite was concocted by D.Richard Hipp in August 2000. SQLite is an elite, lightweight social database. In the event that you are happy to get familiar with the internals of a database at a coding stages, at that point SQLite is the best open-source database accessible out there with profoundly coherent source code with bunches of documentation. SQLite database engineering split into two unique segments named as center and backend. Center segment contains Interface, Tokenizer, Parser, Code generator, and the virtual machine, which make an execution request for database exchanges. Backend contains B-tree, Pager and OS interface to get to the document framework. Tokenizer , Parser and code generator out and out named as the compiler which creates a lot of opcodes that sudden spikes in demand for a virtual machine.

4.1.4 IntelliJ IDEA

IntelliJ IDEA is a integrated development environment (IDE) written in Java for creating PC programming. It is created by JetBrains (once in the past known as IntelliJ), and is accessible as an Apache 2 Licensed people group version, and in an exclusive business release. Both can be used for bussiness advancement. The IDE gives certain highlights like code fruition by investigating the unique situation, code route which permits bouncing to classes or statement in the code straightforwardly, code refactoring, code troubleshooting , linting and choices to fix irregularities through proposals. The IDE furnishes mix with construct/bundling devices like snort, grove, gradle , and SBT. It bolsters adaptation control frameworks like Git, Mercurial, Perforce, and SVN. Databases like Microsoft SQL Server, Oracle, PostgreSQL, SQLite and MySQL can be gotten to straightforwardly from the IDE in the Ultimate release, through an inserted variant of DataGrip. IntelliJ underpins modules through which one can add extra usefulness to the IDE. Modules can be downloaded and introduced either from IntelliJ's

module store site or through the IDE's inbuilt module look and introduce highlight. Every version has separate module storehouses, with both the Community and Ultimate releases totaling more than 3000 modules each starting at 2019..

4.1.5 Machine Learning

A better than average start at a Machine Learning definition is that it is a middle sub-territory of Artificial Intelligence (AI). ML applications gain for a reality (well data) like individuals without direct programming. Exactly when introduced to new data, these applications learn, create, change, and make without any other person. All things considered, with Machine Learning, PCs discover astute information without being exhorted where to look. Or maybe, they do this by using computations that gain from data in an iterative technique.

While the idea of Machine Learning has been around for quite a while (think about the WWII Enigma Machine), the capacity to robotize the use of complex scientific estimations to Big Data has been picking up energy in the course of the most recent quite a long while.

At a significant level, Machine Learning is the capacity to adjust to new information freely and through emphasess. Fundamentally, applications gain from past calculations and exchanges and use "design acknowledgment " to deliver dependable and educated outcomes.

4.1.6 HTML

Hypertext Markup Language (HTML) is the standard markup language for archives intended to be displayed in an internet browser. It very well may be helped by advancements, for example, Cascading Style Sheets (CSS) and scripting languages, for example, JavaScript.

Web programs get HTML chronicles from a web server or from neighborhood amassing and render the reports into intuitive media site pages. HTML depicts the structure of a site page semantically and at first included signs for the presence of the report.

HTML parts are the structure squares of HTML pages. With HTML creates, pictures and various articles, for instance, natural structures may be introduced into the rendered page. HTML gives an approach to make composed documents by significance assistant semantics for content, for instance, headings, sections, records, associations, refers to and various things. HTML parts are depicted by names, made using point segments. Marks, for instance, <img/> and <input/> direct carry content into the page. Various names, for instance, <p> envelop and give information about record

message and may join various names as sub-parts. Projects don't show the HTML marks, yet use them to decipher the substance of the page.

4.1.7 Cascading Style Sheets

Cascading Style Sheets (CSS) is a template language used for depicting the presentation of a chronicle written in a markup language like HTML. CSS is an establishment development of the World Wide Web, near to HTML and JavaScript. CSS is proposed to engage the parcel of presentation and substance, including configuration, tones, and literary styles. This parcel can improve content receptiveness, give more prominent versatility and control in the assurance of presentation characteristics , engage different site pages to share masterminding by demonstrating the critical CSS in an alternate .css report, and lessen eccentrics and emphasis in the essential substance .

4.2 Experimental Setup

1. 10th Generation Intel® Core™ i7 Processors 8M Cache , up to 3.90 GHz
2. Disk space 1TB.
3. Operating System: Linux 18.04.

Recommended System Requirements

- Intel® Core™ i5-8257U Processor 6M Cache, up to 3.90 GHz , 8 GB of DRAM.
- Operating systems: Linux 18.04.

Minimum System Requirements

1. Processors: Intel Atom® processor or Intel® Core™ i3 processor.
2. Disk space: 1 GB
3. Operating systems: Windows* 7 or later , macOS, and Linux
4. Python* versions: 2.7.X, 3.6.X
5. Included development tools: conda*, conda-env, Jupyter Notebook* (IPython)
6. Compatible tools: Microsoft Visual Studio*, PyCharm*Included Python packages - NumPy, SciPy, scikit-learn*, pandas, Matplotlib, Numba* , Intel® Threading Building Blocks , pyDAAL, Jupyter, mpi4py, PIP* , and other

4.3 Coding Standards followed

With regards to the coding styles or coding principles, Developers have wide scope of adaptability, based on a few parameters, be it any plan. A decent structured code as that in the present task considers the following :

- prper documentation has been done by comments
- caode is been refactored and extra spaces has been removed
- proper libary has been imported
- proper message hass been added while commiting the code
- proper naming conventions has been followed
- meaningful variable names has been used
- OOP concept has been used
- Code is written in a genral for proper re-usabilty

4.4 Code Integration details

The integration of code has been finished utilizing GIT as the form control. The python code has been well packed in modules which has provided ease of integration . The HTML forms are provided by means of python class by creating object for each input entity and adding constraints to those classes which makes the HTML page dynamic and easy to adapt new changes. The design of our developed project is such that files names and location are specified through regex. The modules are enclosed in packages which increase the readability and simplicity of code. The machine learning algorithms uses a file to read new data provided by user and to predict the result. This prediction result is passed to the prediction page through a data provider class. The prediction data is stored in database for keeping track of predictions.

4.5 Implementation work flow

Implementation considers sort of calculation utilized in the task, its proficiency, with the goal that the utilization of the most conceivable proficient algorithms can be utilized, in view of the utilization case, designers and the developers group. This undertaking has been worked over comparative meaning of usage, wherein every part of execution has been dealt with, directly from the structure of the UI to database,

calculations to programming interface . A specific structure design has been followed to advance adaptability and legitimate support of the code. The work process begins directly from the plan execution and finishes on code usage. Algorithms has been utilized which figures the chance of sicknesses dependent on the input gave by the patient. Authentication and validation of client information has been considered and all the historical backdrop of searches made are put away in the database with timestamp.

4.5.1 Data cleaning

Information cleaning undertakings utilizing Python's Pandas library. In particular, we've center around most likely the greatest information cleaning task, missing attributes values.

4.5.1.1 Sources of Missing Values

- User forgot to fill in a field.
- Data was lost while transferring manually from a legacy database .
- Error in Programming.
- Users decided not to round out a field attached to their convictions about how the outcomes would be utilized or deciphered.

On the off chance that we investigate the coloums of dataset,we can see that Pandas occupied in the blanked space with "NA". Utilizing the isnull() strategy, we can affirm that both the missing worth and "NA" were perceived as missing qualities. Both boolean reactions are True. Pandas will perceive both void cells and "NA" types as absent values.But there are estimations of certain sorts that Pandas won't recognize.

4.5.1.2 Non-Standard Missing Values

On the off chance that there's numerous clients physically entering information, at that point this is a typical issue. Possibly some prefer to utilize "n/a" yet other like to utilize "na". A simple method to recognize these different configurations isto placed them in a list. At that point when we import the data, Pandas will remember them right away . Below is the case of how its done here.

4.5.2 Handling unstructured and structured Data

Python supports good libraries for handling structured and unstructured data

- **Python processing CSV data**

```
# Making a list of missing value types
missing_values = ["n/a", "na", "--"]
df = pd.read_csv("property data.csv", na_values = missing_values)
```

Figure 4.1: making a list of missing values

- The csv document is a text record in which the qualities in the columns are isolated by a comma. How about we consider the accompanying information present in the record named input.csv.
- we can make this record utilizing windows notebook by reordering this information. Spare the document as input.csv
- The read csv capacity of the pandas library is utilized perused the contents of a CSV document into the python condition as a pandas DataFrame. The method can peruse the records from the OS by utilizing appropriate way to the file.
- The read csv method of the pandas library can likewise be utilized to peruse some particular lines for a given section.

- **Python processing JSON data**
  - Make a JSON record by duplicating the beneath information into a content tool like scratch pad. Spare the document with .json expansion
  - The read json function of the pandas library can be used to read the JSON file into a pandas DataFrame.
  - the read json capacity of the pandas library can likewise be utilized to understand a specific columns and specific rows after the JSON file is read to a DataFrame. We use the multi-axes indexing method called .loc ( ) for this purpose.
  - We can also apply the tojson function along with parameters to read the JSON file content into individual records.

4.6 Execution Results and Discussions

The consequences of execution of the code are very good. A total flow of the application could been watched, with no deadly mistake and code break. A portion of the occasions are as per the following:

- User are able to signup with email and password

- Existing user are able to login with valid email and password of more the length of 6 characters
- User must able to predict heart disease
- User must able to predict diabetes
- User shoulc able to move to the contact page of development team by clicking on contact button
- User should be able to logout from its account succesfully

4.7 Non-functional requirements results

1. **Performance parameters** This incorporates the reaction time of the framework usage level of both static and volumetric sort throughput and so forth these parameters are normalized so a framework needs to tail them
2. **Corrective Maintenance** In case of any bugs left in the system, the bugs and issues will be fixed for smooth running of application. The accuracy of the system can be further improved with other algorithm if needed
3. **Adaptive maintenance** The features in the applications can be added sauch as history of the disease can be kept in the log. the available list of symptoms can also be added for covering more number of diseases.
4. **Security** Amazon web administrations utilizes a few operational security highlights like powerlessness managements, malware anticipation, checking, incident management, server and programming stack security, believed server boot, made sure about help APIs and verified access, information encryption, organize firewall rule upkeep.
5. **Interoperability** This requirement demands the system to be built so that it can work in joining with different operating systems and can be changed as per the user requirements.Since the end product of this project is a web app hosted on AWS cloud, therefore the web app can be accessed by any user on any device through the internet.

Chapter 5

Testing

5.1 Test workflow

5.1.1 Integration Testing

Testing for the integration of components of the application. This kind of testing is done after the application is entirely evolved, so as to check the integration with all the limit cases considered. An application is said to pass the integration testing just when all the components are incorporated to one another with an undeniable stream, and the code doesn't break over any module or segment. This testing is done commonly through mechanization as manual testing concerning integration would not test the equivalent on various parameters dependent on ongoing information.

5.1.2 Unit Testing

This kind of testing manages testing of each class, technique or bussiness rationale as a unit. Unit testing is one of the incredible strategies for testing which advances free coupling as every bit of utilization case as a business rationale is kept up as a nuclear unit . No two techniques have some usefulness. Not withstanding that a strategy is planned uniquely to deal with one use case . The strategies just hold bussines rationale decoupled from any device components, explicitly android for our situation. In the application , MVP configuration design has been utilized which is the most capable plan design with regards to unit testing. This design has been utilized in light of the fact that it makes the business rationale liberated from the android components and makes it simpler to perform unit testing with no coupling. All the system calls have been done in the moderator layer, wherein the view layer is obscure of the system calls and related information. The moderator is put to test as and when required .

5.1.3 Validation Testing

It begins around the finishing of coordination testing when explicit parts have been worked out, the thing is completely assembled as a group, and interfacing messes up have been uncovered and redressed. At the support or framework level, the capacity between normal programming, battle engineered programming and WebApps vanishes. The path toward investigating programming in the midst of the change framework or toward the aggregate of the advancement strategy to pick in the event that it satisfies picked business necessities. It ensures that the module truly addresses the client's issues. It can in like manner be depicted as to show that the thing fulfills its run of the mill use when sent on reasonable condition. It reacts to the request, Are we making the right thing?

5.2 Test case details

5.2.1 Test case 1:

Unit to test : User Authentication
Assumptions :

- Patient is a first time user
- Patient may be an existing user
- He/She enters the correct password which belongs to him/her

Test data: minimum six digit passwors

Steps executed:

- A first time user enters his/her username, email, password, confirm password
- Following the above, he/she clicks on the submit button
- Moves to the home page of the web app

Expected result:

- user should be redirected to the home page of the web app where he/she gets option to predict heart or diabeties disease

Actual result:when user enters first time and signUp with email and password, he shoud be redirected to the home page

Result(Pass or Fail): Passed

Comments: The test passed successfully and everything worked fine

5.2.2 Test case 2:

Unit to test: verification of home page details
Assumptions:

- Patient is a first time user
- Patient may be an existing user
- He/She enters the correct password which belongs to him/her

Test data: user should have login with correct username and password

Steps to be executed:

- Home page should have buttons for heart diseases prediction and diabeties prediction
- It should also contains all the navigable buttons in the top corners of website
- Moves to the corresponding page of the web app when any button is clicked.

Expected result: Moves to to the corresponding page of the application when click on any button.

Actual result: The user was able to navigate throughout the website

Result(Pass or Fail): Passed

5.2.3 Test case 3:

Unit to test: database storage of patients
Assumptions:

- Admin user name and password.

Test data: Admin should have login with correct password and username

Steps to be executed:

- Admin login button should be clicked
- admin should enter correct login details in django administrator
- admin should see all the patients results and the user accounts registered

Expected result: admin should see all the patients results and the user accounts registered

Actual result: Admin should be able to see all the records for patients

Result(Pass or Fail): Passed

Comments: The test passed successfully and everything worked fine

5.2.4 Test case 4:

Unit to test: working of prediction engine for Heart diseases
Assumptions:

- Home page should have buttons for heart diseases prediction and diabeties prediction
- it should also contains all the navigable buttons in th top corners of website
- Moves to to the corresponding page of the application when click on any button.

Test data: patient report details

Steps to be executed:

- user should click on predict heart disease button
- user should enter all the details in the form for heart diseases prediction engine

- user should click on predict button

Expected result: user should be able to see the possibility of heart disease by various algorithms used to predict

Actual result: result should be displayed by different algorithm

Result(Pass or Fail): Passed

Comments: The test passed successfully and everything worked fine

5.2.5 Test case 5:

Unit to test: working of prediction engine for diabeties diseases
Assumptions:

- Home page should have buttons for heart diseases prediction and diabeties prediction
- it should also contains all the navigable buttons in th top corners of website
- Moves to to the corresponding page of the application when click on any button.

Test data: patient report details

Steps to be executed:

- user should click on predict diabeties button
- user should enter all the details in the form for diabeties prediction engine
- user should click on predict button

Expected result: User should be able to see the possibility of diabeties by various algorithms used to predict

Actual result: Result should be displayed by different algorithm

Result(Pass or Fail): Passed

Comments: The test passed successfully and everything worked fine

Chapter 6

Conclusions and Future Scope

This project uses the various machine learning algorithms such as support vector machine, NaïveBayes, decision tree, k-nearest neighbour, neural network, logistic regression which were applied to the database. It utilize the data such as blood pressure , cholesterol , diabetes etc and then tries to predict the possibility of heart disease. Family history of heart disease can also be a reason for developing a heart disease as mentioned earlier.This work will be useful in identifying the possible patient who may suffer from heart disease in the next ten years. The most efficient algorithm was to be selected based on various criteria. The accuracies found by different algorithims are as follow :-

- support vector machine 0.8289
- NaïveBayes 0.8000
- decision tree 0.8043
- k-nearest neighbour 0.8913
- neural network 0.9700
- logistic regression 0.8500

We found out that the neural network algorithm was the most efficient out of the three with an accuracy of 97 percentage. Thus the logistic regression algorithm was further implemented using different applications. For this, jupiter notebook was used. Since heart diseases are major killer in India and throughout the world, the application of a promising technology like machine learning to the initial prediction of heart diseases will have a profound impact on society. There are numerous conceivable enhancements that could be investigated to improve the adaptability and exactness of this predicted system. By training the model with different dataset may lead to best fit model because this heart disease data set may vary with years. It could be more benefited by changing the data set and by implementing different algorithms for the prediction of heart disease may increase the efficiency of prediction.

This may help in taking preventivemeasures and hence try to avoid the possibility ofheart disease for the patient. So when a patient ispredicted as positive for heart disease, then themedical data for the patient can be closely analysedby the doctors. An example would be suppose thepatient has diabetes which may be the cause forheart disease in future and then the patient can begiven treatment to have diabetes in control which inturn may prevent the heart disease.The heart disease prediction can be done usin gother machine learning algorithms. Logistic regression can also perform well in case of binary classification problems such as heart disease prediction. Random forests can perform well than decision trees. Also , the ensemble methods and artificial neural network can be applied to the dataset. The results can be compared and improvised

Appendix-A

Pseudocode

- **Dividing data into train and test data** The data is divided into 10 fold out of which 9 folds are used to train the data and 1 fold is for testing.

- **HTML Forms** To provide signup detils a form.py class is used which consist of objects for each entity

  class UserForm(forms.ModelForm):

  username = forms.CharField (widget = forms.TextInput (

  attrs = ['class': 'form-control', 'placeholder': 'Enter username']

  ), required = True, maxlength = 50)

  email = forms.CharField (widget = forms.EmailInput(

  attrs = ['class': 'form-control', 'placeholder': 'Enter Email Id']

) , required = True, maxlength = 50)

  password = forms.CharField(widget = forms.PasswordInput (

  attrs = ['class': 'form-control', 'placeholder': 'Enter password']

) , required = True, minlength = 6, maxlength = 50)

confirmpassword = forms.CharField (widget = forms.PasswordInput (

attrs = ['class': 'form-control', 'placeholder': 'Confirmpassword']

), required = True, minlength = 6, maxlength = 50)

- **Executing Machine learning models** The prediction results of each machine learning model is stored in a file with the regex expression

  predictions = [

  'SVC': str (SVCClassifier.predict(features) [0]),

  'LogisticRegression': str (LogisticRegressionClassifier.predict(features) [0]),

  'NaiveBayes': str (NaiveBayesClassifier.predict(features) [0]),

  'DecisionTree': str (DecisionTreeClassifier.predict(features) [0]),

  'KNN': str (KNeighborsClassifier.predict(features) [0]),

  'NeuralNetwork': str (NeuralNetworkClassifier.predic(features) [0]),]

  The string NaiveBayesClassifier will look for file naivebaiyes.py and execute it to generate naivebaiyes.pkl (executable file)

- **Dataprovider.py** It consist of class

  GetAllClassifiersForHeart ( ) :

  return (GetSVCClassifierForHeart ( ),

  GetLogisticRegressionClassifierForHeart ( ),

  GetNaiveBayesClassifierForHeart ( ),

  GetDecisionTreeClassifierForHeart ( ),

  GetKNeighborsClassifierForHeart ( ),

  GetNeuralNetworkClassifierForHeart ( ))

Which takes the prediction result from machine learning pkl file and provide it to HTML file to display the result.

Bibliography

- https://www.tutorialspoint.com/pythondatascience/
- https://towardsdatascience.com/data-cleaning-with-python-and-pandas-detecting-missing-values
- Prediction of Heart Disease Using Machine Learning Algorithms by Rajesh N, T Maneesha, Shaik Hafeez, Hari Krishna
- A Review on Heart Disease Prediction using Machine Learning and Data Analytics Approach By M. Marimuthu, M. Abinaya , K. S. Hariesh .
- https://bmcmedinformdecismak.biomedcentral.com/articles/
- https://www.kdnuggets.com/2015/12/machine-learning-data-science-apis.html