

Siddaganga Institute of Technology, Tumakuru

(An Autonomous institution affiliated to Visvesvaraya Technological University, Belagavi,
Approved by AICTE, New Delhi, Accredited by NAAC and ISO 9001:2015 certified)



Heart Disease Prediction using Machine Learning Algorithms

Predicting type of heart disease

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

Haarika Reddy K R 1SI17CS041

Mohit Sah 1SI17CS060

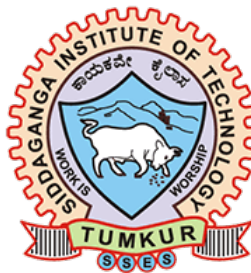
Shrinidhi Shastry 1SI17CS105

Veeksha V Murthy 1SI17CS128

under the guidance of

Dr. Nirmala M B

Associate Professor



Department of Computer Science & Engineering

(Program Accredited by NBA)

Siddaganga Institute of Technology

B.H Road, Tumakuru-572 103, Karnataka, India.

Web : www.sit.ac.in

August, 2021

Department of Computer Science and Engineering
Siddaganga Institute of Technology, Tumakuru
(An Autonomous institution affiliated to Visvesvaraya Technological University, Belagavi,
Approved by AICTE, New Delhi, Accredited by NAAC and ISO 9001:2015 certified)



Certificate

This is to certify that the Project Report entitled "**Heart Disease Prediction using Machine Learning Algorithms**" is a bonafide work carried out by **Haarika Reddy K R (1SI17CS041)**, **Mohit Sah (1SI17CS060)**, **Shrinidhi Shastry (1SI17CS105)** and **Veeksha V Murthy (1SI17CS128)** 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 2020-21. 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
Dr. Nirmala M B
Associate Professor
Dept of CSE, SIT

.....
Group Convener
Dr. Nirmala M B
Associate Professor
Dept of CSE, SIT

.....
Dr. Poornima A S
Professor and Head
Dept of CSE, SIT

.....
Dr. S V Dinesh
Principal
SIT, Tumakuru

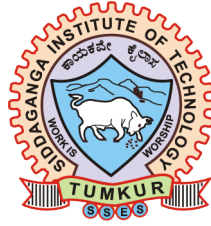
Name of the Examiners

Signature with Date

1. Prof.

2. Prof.

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



DECLARATION

We hereby declare that the entire work embodied in this dissertation has been carried out by us at **Siddaganga Institute of Technology** under the supervision of **Dr. Nirmala M B**. This dissertation has not been submitted in part or full for the award of any diploma or degree of this or any other University.

Haarika Reddy K R (1SI17CS041),
Mohit Sah (1SI17CS060),
Shrinidhi Shastry (1SI17CS105),
Veeksha V Murthy (1SI17CS128).
Department of Computer Science and Engineering
Siddaganga Institute of Technology
Tumakuru - 572103

Acknowledgements

We offer our humble pranams at the lotus feet of his holiness, **Dr. Sree Sree Shivakumara swamigalu**, founder President, **Sree Sree Siddalinga Swamigalu**, president, Sree Siddaganga Education Society, for best owing upon their blessings.

We deem it as a privilege to thank **Dr. M N Channabasappa**, Director, SIT, Tumakuru and **Dr. S V Dinesh**, Principal, SIT, Tumakuru for an excellent academic environment in this institute, which made this endeavor fruitful.

We would like to express our sincere gratitude to **Dr.A S Poornima**, Professor and Head, Department of CSE, SIT, Tumakuru for her encouragement and valuable suggestions.

We thank our guide **Dr. Nirmala M B**, Associate Professor and group convener for valuable guidance, advice and encouragement.

We would like to thank our Family and friends for being a constant source of moral support and encouragement throughout the span of the project.

Haarika Reddy K R (1SI17CS041)

Mohit Sah (1SI17CS060)

Shrinidhi Shastry (1SI17CS105)

Veeksha V Murthy (1SI17CS128)

Abstract

The world which we see today is getting advanced every year. This is because some of the most promising elements, such as field development industrialisation, globalisation, and several other aspects of science and technology additional considerations. Considering the remarkable development in surroundings we can also see a drastic change in the human's health. Humans are facing a lot of health issues because of the modern food and the lifestyle they have adopted. Many diseases are caused by junk food and a sedentary working culture.

Heart disease can also be called cardiovascular disease which has become one of the most hazardous diseases. Its death rate is increasing for decades. This is due to the way the people lead their life in a sedentary style or by their food habits. From the WHO (World Health Organization) statistics around 17.9 million people die globally every year due to cardiovascular diseases which contributes to 31% of the deaths worldwide. Cardiovascular diseases are related to both heart and blood vessels disorder. These also include rheumatic, cerebrovascular, and coronary heart diseases. Heart attacks and strokes make up to 80% of cardiovascular diseases. The factors affecting cardiovascular disease can be classified as habitual and physiological risk factors. Smoking and drinking are two of the most common risk factors. Over-consumption of alcohol and caffeine, as well as stress. Tobacco use raises the risk of cancer. By a factor of two or three, the chances of dying are increased. Physical inactivity, as well as other physiological factors, variables such as high blood pressure, obesity, lipids, obesity, diabetes, and glucose. The heart condition is affected by hypertension, high blood cholesterol, and pre-existing heart abnormalities.

Predicting the cardiovascular disease of a patient at an early stage is a difficult task for doctors, hence an automated system is in need to predict

and to save from major risks. With the increase in heart diseases, there are several pieces of equipment and sophisticated machines to predict the heart condition.

Data mining is the process of extracting essential data from large datasets. Manufacturing Engineering, Financial Banking, and a variety of other fields Corporate surveillance, criminal investigation, and a variety of other services are available. Machine Learning is a burgeoning subject in the study of AI (artificial intelligence), in which discrete data from multiple sources is combined. The discrete information from various datasets can be handled and analyzed to predict the diseases so that the patient can be treated well in advance to avoid menacing circumstances. Some of the most well-known algorithms in machine learning are Nave Bayes, K-nearest Neighbors (KNN), Decision tree, Random Forest, and Support Vector Machine (SVM), which are used to classify and analyse variables in order to forecast heart problems. Above algorithms predicting efficiency is also based on biasness and variance of dataset.

In the proposed system, a patient can easily check the condition of their heart. The proposed system considers some of the famous algorithms of Machine Learning which includes Naïve Bayes, Logistic regression, Support Vector Machine, Decision tree, and Random forest. The System also makes use of Deep learning techniques to analyze and understand the dataset. Cleveland dataset from UCI repository with 14 attributes and 303 instances is used for analysis. The patient just needs to provide some of the information as input and the system will perform predictions on the condition of the heart and other factors.

The Random forest algorithm has provided a comparatively precise result for the study with an accuracy of 95.08%. An android application is implemented using the random forest to predict the type of disease. The various types of disease it predicts are coronary artery, heart failure, stroke, and no heart disease.

Contents

Acknowledgements	iv
Abstract	v
List of Figures	xi
List of Tables	xiii
1 Introduction	1
1.1 Background Study	1
1.2 Related Works	2
1.2.1 Heart Disease Prediction Using Machine Learning Algorithms	2
1.2.2 Heart Disease Prediction Using Machine Learning Algorithms	3
1.2.3 Heart Disease Prediction Using Machine Learning Techniques	3
1.2.4 Analysis of Heart Disease using in Data Mining Tools Orange and Weka	4
1.2.5 Improving the accuracy of prediction of heart disease risk based on ensemble classification techniques	4
1.2.6 Heart Disease Prediction Using Machine Learning Algorithms	5
1.2.7 Neural Network-Based Intelligent System for Predicting Heart Disease	5
1.2.8 Effective Heart Disease Prediction Using Hybrid Machine Learn- ing Techniques	6
1.2.9 Improved Study of Heart Disease Prediction System using Data Mining Classification Techniques	6

1.2.10	A Review on Heart Disease Prediction Using Machine Learning Techniques	7
1.3	Project Problem Statement and Objectives (in detail)	8
1.4	Organization of the Report	8
2	High-level Design	10
2.1	Software development methodology	10
2.2	Architecture	12
2.2.1	System Architecture	12
2.3	Modules Description	15
2.3.1	Analysis	15
2.3.2	Doctors	16
2.3.3	Profile	16
2.3.4	Feedback	16
2.4	Functional Requirements	16
2.4.1	Read_csv	16
2.4.2	Train_test_split	16
2.4.3	Fit	17
2.4.4	Predict	17
2.4.5	Accuracy_score	17
3	Detailed Design	18
3.1	Interface design	18
3.2	Data Structures and Algorithms	34
3.2.1	Extraction of data	34
3.2.2	Data Processing and Classifying	34
3.3	Algorithms	34
3.3.1	Logistic Regression	34
3.3.2	Naive Bayes	36
3.3.3	Support Vector Machine	37
3.3.4	Decision Tree	37
3.3.5	Random Forest	37
3.4	UML diagrams	38
3.4.1	Use case diagram	38
3.4.2	Sequence diagram	40

3.4.3	Activity diagram	42
3.4.4	Data flow diagram	43
3.5	Data Source/Database used and Formats	45
4	Implementation	47
4.1	Tools and Technologies	47
4.1.1	Machine learning	47
4.1.2	Jupyter Notebook	48
4.1.3	Python	48
4.1.4	Android Studio	48
4.1.5	Azure data studio	48
4.2	Experimental Setup	49
4.3	Coding Standards followed	56
4.4	Code Integration details	56
4.5	Implementation work flow	56
4.6	Execution Results and Discussions	57
4.7	Non-functional requirements results	60
5	Testing	62
5.1	Test workflow	62
5.1.1	Testing Methodologies	63
5.1.2	Objectives of testing	63
5.1.3	Test Cases	63
5.1.4	Black Box Testing	64
5.1.5	White Box Testing	64
5.1.6	Unit Testing	65
5.1.7	Integration Testing	65
5.1.8	Functional Testing	65
5.1.9	Output Testing	65
5.2	Test case details	66
5.2.1	Test case id: TC01	66
5.2.2	Test case id: TC02	66
5.2.3	Test case id: TC03	66

6	Conclusions and Future Scope	68
6.1	Conclusion	68
6.2	Future Scope	68
A	Abbreviations	69
	Bibliography	70

List of Figures

2.1	Incremental model	10
2.2	System Architecture	13
2.3	System Architecture of Android Application	14
3.1	Registration Section	19
3.2	Login Section	20
3.3	Analysis Section	21
3.4	Navigation Drawer	22
3.5	Profile Section	23
3.6	Doctors Section	24
3.7	Feedback Section	25
3.8	New Feedback Page	26
3.9	Change Password	27
3.10	Logout	28
3.11	Home page of Admin	29
3.12	Update doctors	30
3.13	Training data	31
3.14	View users	32
3.15	View feedback	33
3.16	ROC-Receiver Operating Characteristic Curve	35
3.17	Regplot characteristic	35
3.18	Confusion Matrix	36
3.19	Use case diagram for user	39
3.20	Use case diagram for admin	40
3.21	Sequence diagram for user	41

3.22	Sequence diagram for admin	42
3.23	Activity diagram	43
3.24	Level-0 Data Flow Diagram	44
3.25	Level-1 Data Flow Diagram	45
3.26	Dataset	46
4.1	Splitting dataset	50
4.2	Logistic Regression	51
4.3	Naive Bayes	52
4.4	Support Vector Machine	53
4.5	Decision Tree	54
4.6	Random Forest	55
4.7	Output: No disease	58
4.8	Output: Coronary Artery Disease	59
4.9	Output: Congestive Heart Failure	60

List of Tables

4.1	Accuracy obtained from various algorithms	55
-----	---	----

Chapter 1

Introduction

This chapter contains a detailed description of the project's introduction. It highlights the reason for the project concept, as well as the project's importance to the industry, its relevance and societal impact. The objectives are explained in conjunction with the problem statement.

1.1 Background Study

In recent times, majorly clinical test outcomes are made based on doctor's instinct, knowledge, and experience as opposed to the huge amounts of data available. Although the requirement of the all the latter is very crucial. Modern medicine can have a greater success rate if the judgments are also influenced by the data and the abundantly available statistics. These methods adopted frequently can help in predicting the disease at an earlier stage and thus reducing the death rates caused. The provision of high-quality services at reasonable prices is a major concern for health-care institutions (hospitals, medical facilities). Quality service entails appropriately diagnosing patients and providing effective therapies. Poor clinical decisions can have disastrous effects, which is unacceptably dangerous. Clinical tests must also be kept to a minimum in hospitals.

The main motivation to do this problem comes from the estimation made by World Health Organization. Based on its prediction, by 2030, about 23.6 million individuals will die because of heart trouble. So to diminish the danger, there must be a solution

to be found that can aid these motives. Analysis of coronary illness is usually made based on the depiction of its signs, manifestations, and physical examination of the individual. The most bothersome and compound assignment in medical science is finding of right illness.

Machine Learning is one of the booming domains in the field of AI (Artificial Intelligence) by which the discrete information from various datasets can be handled and analyzed to predict the desired output. Prediction of patient's disease in early-stage and treated well in advance can avoid menacing circumstances. Machine learning algorithms such as Naïve Bayes, K-nearest Neighbors (KNN), Decision Tree, Random Forest, and Support Vector Machine (SVM) are used to classify the attributes and analyze them to predict heart diseases. The factors affecting efficiency are biasness and variance of the dataset. KNN has a problem of overfitting caused due to high variance and low biasness. High biasness and low variance are always advantageous for small datasets as it takes low computing time for training and testing, considering Naïve Bayes for small datasets, it works efficiently with high biasness and low variance.

1.2 Related Works

1.2.1 Heart Disease Prediction Using Machine Learning Algorithms

Title of the work: Heart Disease Prediction Using Machine Learning Algorithms.

Authors: Archana Singh, Rakesh Kumar. [6]

Publication details: International Conference on Electrical and Electronics Engineering, 2020.

The authors of this paper have implemented algorithms such as KNN, SVM, Decision Tree, and Linear Regression. The Decision tree is a nonparametric algorithm technique that has an overfitting problem but can be resolved using removable methods. Support vector machine is used for classifying the datasets as it has a statistical and algebraic background. Authors have used the Cleveland dataset and have considered

73% for training and 37% for testing. The accuracy of K-Nearest Neighbor is 87%, Support Vector Machine is 83%, and Decision Tree is 79%.

1.2.2 Heart Disease Prediction Using Machine Learning Algorithms

Title of the work: Heart Disease Prediction Using Machine Learning Algorithms

Authors: ApurbRajdhan, Milan Sai, Avi Agarwal, Dundigalla Ravi, Dr. PoonamGhuli. [7]

Publication details: International Journal of Engineering Research & Technology, Vol.9 Issue 4th, April-2020.

The authors aim to provide the doctors with tools and techniques to diagnose heart disease at an early stage. Machine learning plays a vital role in identifying discretely hidden patterns and analyzing the available data. The Authors concentrated on providing an efficient algorithm with high accuracy for predicting whether the patient is suffering or not from heart disease. The input to the system is the patient's report of health which contains different attributes. The accuracy of Random Forest is 90.16%, Logistic Regression is 85.25%, and Naïve Bayes is 85.25%.

1.2.3 Heart Disease Prediction Using Machine Learning Techniques

Title of the work: Heart Disease Prediction Using Machine Learning Techniques

Authors: Devansh Shah, Samir Patel, Santosh Kumar Bharti. [3]

Publication details: 16th, October-2020 in Springer Nature journal.

Authors uses the computer to understand non-linear and complex interactions among various attributes or factors for prediction. Thus reducing the errors in prediction and producing truthful outcomes. Data mining explores large datasets to conclude past collected or repository data for future prediction. WEKA tool was used for pre-processing the data, which was in ARFF (Attribute Relation File Format) format. Out of 76 various attributes, only 14 different attributes were considered for better

accuracy. The accuracy of Naïve Bayes is 88.15%, K-Nearest Neighbor is 90.78%, Decision Tree is 80.26% and Random Forest is 84.21%.

1.2.4 Analysis of Heart Disease using in Data Mining Tools Orange and Weka

Title of the work: Analysis of Heart Disease using in Data Mining Tools Orange and Weka

Authors: Sarangam Kodati, Dr. R Vivekanandam. [10]

Publication details: Global Journal of Computer Science and Technology, Vol18. 2018.

A large amount of hidden and raw data are collected from the health industry which can be used to make efficient decisions by applying data mining techniques. Also, there is a need for an effective analyzing tool to analyze the hidden and raw data, and data mining algorithms can be used to develop such systems for analyzing and classifying the data and to detect, analyze and predict the heart diseases from which the patient is suffering. To find the heart disease there is a need for some tests to be done on the patient. However, by using the data mining technique we can reduce the number of tests, doing this significantly adds to the performance and time. The dataset considered by the authors is from the UCI repository which contains the Cleveland heart disease dataset. 14 out of 76 attributes with 303 instances were used.

1.2.5 Improving the accuracy of prediction of heart disease risk based on ensemble classification techniques

Title of the work: Improving the accuracy of prediction of heart disease risk based on ensemble classification techniques

Authors: C. Beulah ChristalinLatha, S. CarolinJeeva. [4]

Publication details: Informatics in Medicine Unlocked, 2019.

The patient will be in a critical condition because of no early diagnosis of the disease, but this can be solved using Machine Learning. Machine learning algorithms used by

authors are Support Vector Machine, Decision Tree, Naïve Bayes, and Neural Networks. The accuracy of Support Vector Machine (SVM) is 92.1%, Decision Tree is 89.9% and Naïve Bayes is 84.4%.

1.2.6 Heart Disease Prediction Using Machine Learning Algorithms

Title of the work: Heart Disease Prediction Using Machine Learning Algorithms

Authors: RishabhMagar, Rohan Memane, SurajRaut, Prof. V.S. Rupnar. [8]

Publication details: Journal of Emerging Technologies and Innovative Research, Vol7, June 2020.

Prediction of the heart condition is one of the challenging tasks for Doctors. Authors mainly concentrate on designing an automated system that can accurately predict the heart condition by using some attributes as input and also considering some of the famous Machine Learning algorithms. The accuracy of Support Vector Machine is 81.54%, Naïve Bayes is 80.43% and Logistic Regression is 82.89%.

1.2.7 Neural Network-Based Intelligent System for Predicting Heart Disease

Title of the work: Neural Network-Based Intelligent System for Predicting Heart Disease

Authors: K Subhadra, Vikas B. [2]

Publication details: International Journal of Innovative Technology and Exploring Engineering, Vol8. Issue 5th, March-2019.

In the field of medicine, the diagnosis of heart disease has been a crucial task for doctors and patients. The diagnosis of an individual is based on the patient's report of all previous health checkups and tests. The emerging improvement in the machine learning field has made the developing automated intelligent systems that improve the accuracy of prediction in the field of medicine which helps in making better decisions. The authors used a system to save individual life by consuming less time

for prediction and treatment. Various classification techniques are used to achieve an accurate and better diagnosis. One of the classification methods is Neural Networks that can operate as the brain of humans and draw relationships among huge amounts of data in a dataset. The attributes were from ECG and symptoms used to perform prediction on the presence of heart disease. The accuracy obtained using the Cleveland dataset is as follows The back-propagation algorithm has 97.5%, the hidden layer with 20 neurons has 98.58% and the hidden layer with 5 neurons has 93.39%.

1.2.8 Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques

Title of the work: Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques

Authors: Senthilkumar Mohan, ChandrashekarThirumalai, Gautam Srivastava. [9]

Publication details: IEEE Access, 3rd July-2019.

Data mining is one of the computer-based data processing in the field of medicine. It helps in finding the hidden different patterns of various related datasets. This can be used to diagnose heart disease. The data available will always be in heterogeneous form. It is very difficult to draw a conclusion using such a huge volume of data. Hence organizing data is necessary. And an automated system from Authors with different algorithms to diagnose effectively with more accuracy for individual clinical tests and reports. The authors proposed an algorithm with an efficiency of 88.4% whereas Naïve Bayes has 75.8%, Support Vector Machine has 86.1%, Random Forest has 86.1%, and Decision Tree has 85.0%.

1.2.9 Improved Study of Heart Disease Prediction System using Data Mining Classification Techniques

Title of the work: Improved Study of Heart Disease Prediction System using Data Mining Classification Techniques

Authors: Chaitrali S. Dangare, Dr. Sulabha S. Apte. [1]

Publication details: International Journal of Computer Applications, Vol47. 10th June 2012.

The diagnosis of heart disease can be done by using different machine learning and data mining techniques. The authors used the feature selection method to improve the accuracy of the performance of the algorithm. Based on the diagnosis of the type of heart disease, many curable methods can be used for that patient. Using predictive model techniques can only give marginal success. Hence more complex and combination models are needed for better accuracy in prediction at an early stage of heart disease. The system gets more intelligent as the amount of data fed into the system increases. There is a need for future work in improving accuracy and scalability. This can be done by using various discretization methods. The accuracy of Naïve Bayes is 82.4%, Support Vector Machine is 81.7%, and Random Forest is 77.9%.

1.2.10 A Review on Heart Disease Prediction Using Machine Learning Techniques

Title of the work: A Review on Heart Disease Prediction Using Machine Learning Techniques

Authors: Adil Hussain She, Dr. Pawan Kumar Chaurasia. [5]

Publication details: Research Gate, April-2019.

Heart disease prediction is one of the quintessential fields in the medical field. Heart disease is also one of the most fatal in the world if not treated well. It describes the changes that occur in the heart that leads to this fatality. Another major challenge is Quality of Service (QoS). Data mining is the process of determining unknown hidden patterns of the data set. Authors have obtained an accuracy of Naive Bayes is 52.33%, Decision Tree is 52% and K-Nearest Neighbor is 45.67%.

1.3 Project Problem Statement and Objectives (in detail)

This project is aiming to minimize the percentage of risks that are faced by individuals due to the lack of awareness about heart diseases and their impending fatal hazards. The project proposes to make use of the abundant data that is produced daily in the medical industry. The computer-based solutions can help in predicting the disease at a faster pace increasing the chance of treating the patient at an early and salvageable state.

The main objectives of the project are as mentioned below:

1. To develop a machine learning model that can accurately predict heart disease based on the data entered by the user.
2. To incorporate majority attributes required for the prediction.
3. To develop an android application that can provide a better user experience.
4. To make the android application easily accessible and user-friendly.

1.4 Organization of the Report

The report consists of a total of six chapters, an appendix, and a bibliography. These chapters are organized in the following manner.

Chapter 1 gives an introduction to the project. It includes motivation behind the objective to take this topic as the final year major project, the background study, problem statement, objectives, social impact, and industrial impacts of this project.

Chapter 2 interprets the high-level design of the system. This section also contains the architecture of the project, functional requirements, and software development methodology that has been followed.

Chapter 3 illustrates the detailed design of the system. UML diagrams such as use case diagrams, sequence diagrams, activity diagrams, and data flow diagrams are used to represent the developed system's user interface, data structures, and algorithms. It also goes through the project's data source or database.

Chapter 4 emphasizes the implementation phase of the project work. It covers the topics related to the tools and technologies related to the project and implementation workflow of this project.

Chapter 5 provides details of testing. It contains the test workflow, test objectives, and testing methodology. It also briefs on the test case details.

Chapter 6 provides a conclusion of the project. It also contains future work that can be done related to this project.

Appendix A contains abbreviations that are being used in this project.

Bibliography contains the different technical papers that are referred to complete the project.

Chapter 2

High-level Design

The chapter 2 includes Software Development Methodology, System Architecture and Functional requirements of the system.

2.1 Software development methodology

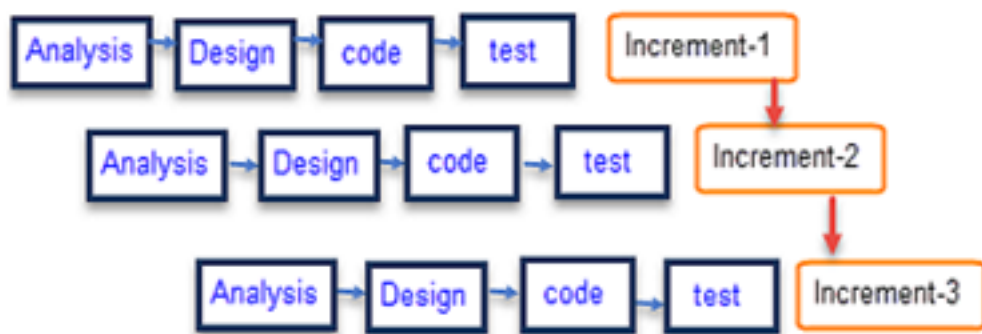


Figure 2.1: Incremental model

The software development model used in this project is incremental. The incremental model is a software development process in which requirements are broken down into many independent modules during the software development cycle. Analysis, design,

implementation, testing, and maintenance are all steps in the process. Every iteration goes through the processes of requirements, design, coding, and testing. And until all of the system's specified functionality has been realised, each succeeding release adds functionality to the prior release. When the first increment is provided, the system is put into production. Following the client's analysis of the core product, the strategy for the next increment is developed. Figure 2.1 depicts the incremental model.

The characteristics of this model includes:

- Highest priority requirement is tackled first.
- The system development is split down into several smaller tasks.
- To create a final total system, partial systems are developed one by one.
- The requirement for that increment is frozen once it has been developed.

This model is preferable in the following situations:

- The system's needs are well-defined.
- There is a desire for an early product release.
- When there are large stakes and a lot of risk involved.
- For product-based businesses.

The following are some of the benefits of this model:

- The software will be created in a timely manner.
- It's adaptable and simple to use.
- Changes can be done throughout the development stages.
- It is economical.
- Errors can be easily identified.

Disadvantages of this model are:

- Good planning and designing is required.
- Each iteration phase is rigid and does not overlap each other.
- Rectifying a problem in one unit requires correction in all the units and consumes a lot of time.

2.2 Architecture

The structure and behaviour of a system are described by its architecture. It also includes a graphical representation of the system's ideas, concepts, elements, and components.

2.2.1 System Architecture

Figure 2.2 describes the prediction model which follows a specific way of implementation starting from loading the dataset followed by pre-processing the data which includes importing libraries, finding missing data and encoding categorical data are the first steps followed by splitting the dataset and applying algorithms such as Logistic Regression, Nave Bayes, Decision Tree, Support Vector Machine, and Random Forest, then analysing the results and modifying the model to achieve the best possible accuracy.

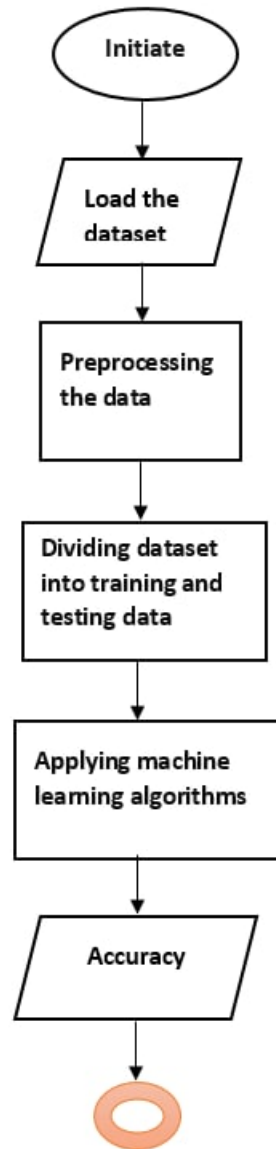


Figure 2.2: System Architecture

From Figure 2.2, pre-processing the data considers importing the required libraries and processing the data, as few records of data might be missing. The missing data can be filled either by mean value or by ignoring it. The categorical data is converted into numerical data for prediction. From the dataset considered above, Gender is categorical data which is transformed into numerical by the following format: 0 for female and 1 for male and few other attributes such as defining the type of chest

pain, slope.

The data is split as training, to feed to the algorithm and testing, to calculate the precision for the algorithms in the ratio of 80:20.

Heart Disease Prediction App

Figure 2.3 depicts the flow of the application process in which the user has to register prior to login. Upon successful registration, the user can access different available pages. They are the analysis page, my profile page, doctor's page. Users also have a feature to change their passwords.

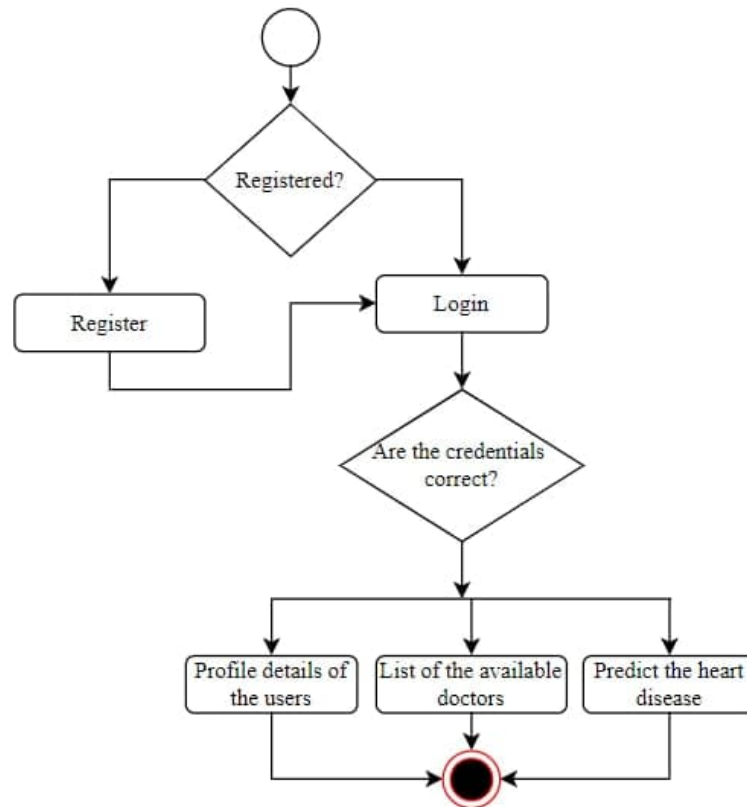


Figure 2.3: System Architecture of Android Application

My profile/account contains the user's information as well as the ability to change and update the information. The details of the available doctor are listed on the doctor's page. Based on the values entered by the user for the attributes under consideration, the analysis page predicts the kind of heart disease. It predicts coronary artery disease as the type of heart disease. No heart disease, coronary artery disease, and congestive heart failure Damage to the heart's primary blood vessels is known as coronary artery disease. This causes artery constriction and restricts blood flow to the heart. The heart does not pump blood as well as it should in congestive heart failure.

2.3 Modules Description

2.3.1 Analysis

The analysis page will ask the user for the following attributes to predict the type of heart disease.

- Age
- Gender
- Chest pain
- Blood Sugar
- Resting ECG
- Exercise induced Angina
- Slope
- Number of Major Values (CA)
- Thalach
- Rest Blood Pressure

- Serum Cholesterol
- Maximum heart-rate achieved
- ST depression

2.3.2 Doctors

The user can contact their appropriate doctor by search operation with the list of detailed doctors provided. Details of doctors includes doctor name, specialization type and address. This helps them to discuss the doctor regarding their disease.

2.3.3 Profile

The user can view and update his details such as name, email address, contact number, age, gender, and address.

2.3.4 Feedback

The user can provide the feedback to the system.

2.4 Functional Requirements

2.4.1 Read_csv

Name of the module: read_csv()

Parameters: Pandas.read_csv("file.csv")

Purpose: Reads a comma-separated values (csv) file into DataFrame.

2.4.2 Train_test_split

Name of the module: train_test_split()

Parameters: sklearn.model_selection. train_test_split(*arrays, test_size=None, train_size=None, random_state=None, shuffle=True, stratify=None)

Purpose: Splits the data arrays into two subsets: for training data and for testing

data.

2.4.3 Fit

Name of the module: `fit(X,y)`

Parameters: `fit(X,y)`

Purpose: `fit` is an estimator which will be able to predict the classes to which unseen samples belong. The `fit()` method takes the training data as arguments, which can be one array in the case of unsupervised learning, or two arrays in the case of supervised learning. Note that the model is fitted using `X` and `y` , but the object holds no reference to `X` and `y` .

2.4.4 Predict

Name of the module: `predict(X)`

Parameters: `predict(X)`; `X`:array like or sparse matrix, shape (n-samples, n-features)

Purpose: Given a trained model, predict the label of a new set of data. This method accepts one argument, the new data `X-new` (e.g. `model.predict(X-new)`), and returns the learned label for each object in the array.

2.4.5 Accuracy_score

Name of the module: `accuracy_score()`

Parameters: `sklearn.metrics.accuracy_score(y_true, y_pred, *, normalize=True, sample_weight=None)`

Purpose: In multilabel classification, this function computes subset accuracy: the set of labels predicted for a sample must exactly match the corresponding set of labels in `y_true`. Parameters `y_true`1d array-like, or label indicator array / sparse matrix. Ground truth (correct) labels.

Chapter 3

Detailed Design

The Detailed Design chapter briefs about system design in detail. It contains interface design, data structures and algorithms, data source, databases and discussions on UML diagrams.

3.1 Interface design

Before login to the system, user has to register by providing the information such as name, gender, email, phone number, age, address and password. In which he/she will have to remember password for further login details.

The figure 3.1 shows the details of registration page provided for the user where they have to provide the required details like name, gender, email, phone number, password and age.



Figure 3.1: Registration Section

To login to the system, user has to provide their email and password. Database connected to system validates the credentials. And provides further features to user.

The figure 3.2 shows login section where in user has to enter credentials like email and password.

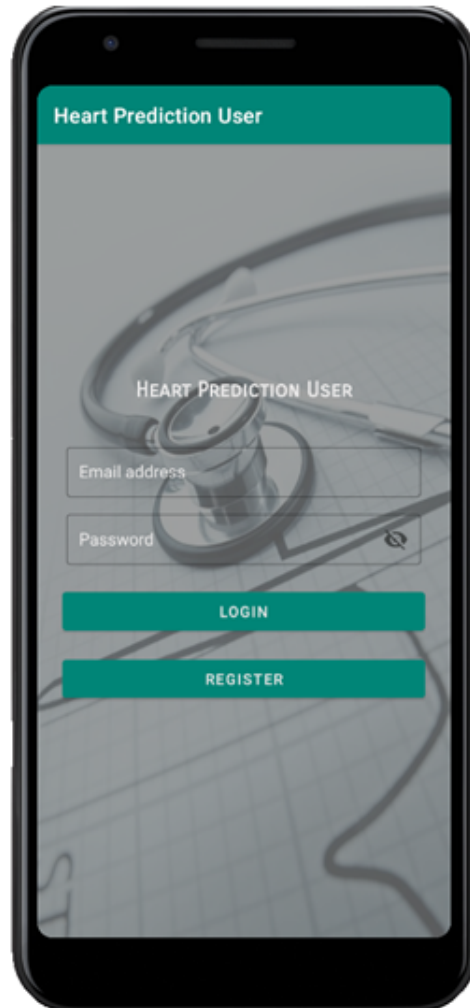


Figure 3.2: Login Section

The analysis page takes the input from the user for the attributes as follows: Age in number/ integer of the user, Gender of the user as radio buttons, Chest Pain type as 1 for typical angina, 2 for atypical angina, 3 for non- angina pain, 4 for asymptotic. Blood Sugar as number/ integer, Resting Electrographic (ECG) as 0 for normal, 1 for abnormality of ST-T wave, 2 for hypertrophy of left ventricle. Exercise-Induced Angina as 0 for no and 1 for yes. Slope as 1 for up sloping, 2 for flat, 3 for downsloping. The number of Major vessels indicates the value of colored vessels by fluoroscopy. Thal is Thalassemia as 3 for normal, 6 for fixed defect, 7 for a reversible defect. Rest Blood Pressure in number/ integer. Serum Cholesterol in number/ integer. Thalach

is the Maximum Heart Rate Achieved in number/ integer. Old Peak is ST Depression induced by exercise in number/ integer.

The figure 3.3 shows the analysis page where user can provide the values for the attributes listed. And gives the analysis based on the values provided by user.

The image shows a mobile application interface for an 'Analysis' section. The screen has a teal header with a menu icon, the title 'Analysis', and a vertical ellipsis. Below the header, there are several input fields and buttons. The fields are: 'Age', 'Choose Gender' (with radio buttons for 'Male' and 'Female'), 'Check Pain', 'Blood Sugar', 'Resting Electrographic', 'Exercise Induced Ang...', 'Slope', 'Number of Major Ves...', 'Thal', 'Rest Blood Pressure', 'Serum Cholesterol', 'Maximum Heart Rate Achieved (Thalach)', and 'ST Depression Induced by Exercise (Old Peak)'. At the bottom, there is a teal 'SUBMIT' button.

Figure 3.3: Analysis Section

Navigation Drawer is a button link structure which acts like slide, it will have the options like

- Analysis

- Profile
- Doctors
- Feedback

The figure 3.4 shows the navigation drawer section for user to select the required option.



Figure 3.4: Navigation Drawer

In the profile page, the system provides the user with their own details and are given an option to modify it if required. The details provided are name, gender,

email, contact number, age, address.

The figure 3.5 shows the personal details of the user.

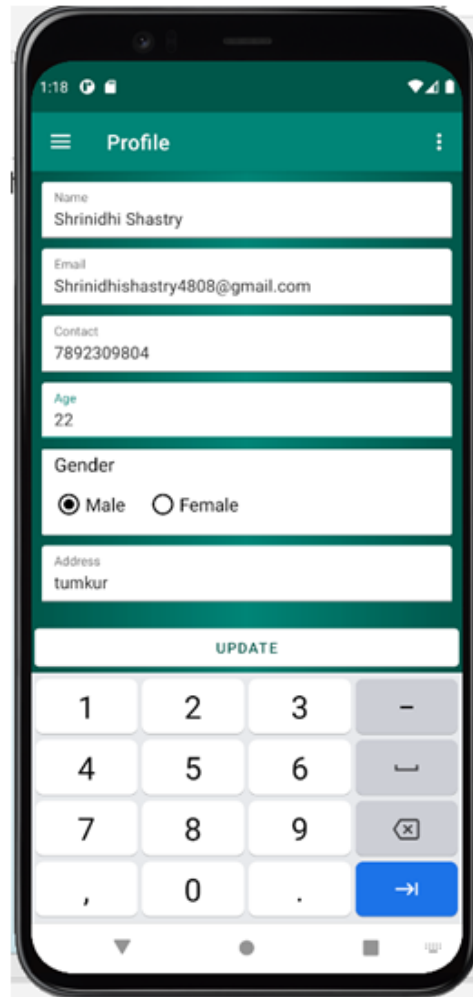


Figure 3.5: Profile Section

In the doctors section, the system provides users with a list of doctors in which they can consult using name, specialization, and address of doctor.

The figure 3.6 shows the detailed list of doctors. The user can use the details and can consult the doctor.

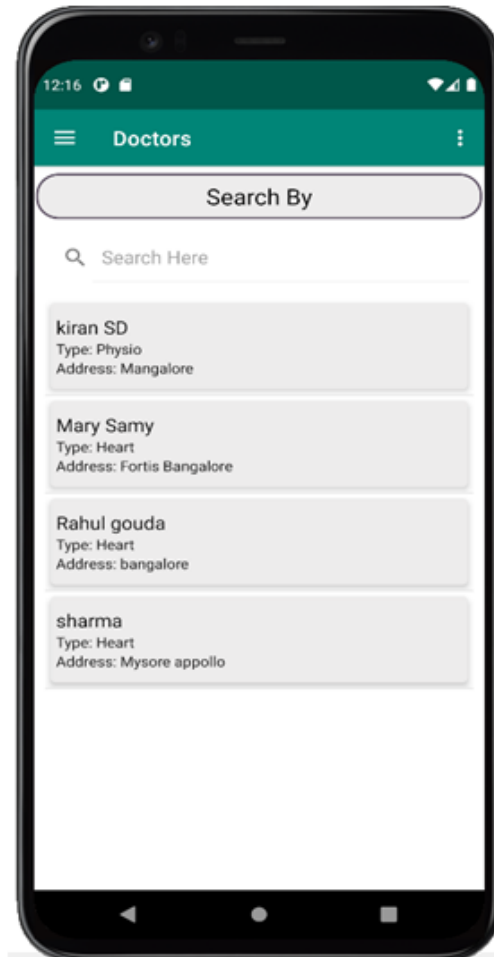


Figure 3.6: Doctors Section

The user can give their valuable feedback in the feedback section. For new feedback, they have to click on plus symbol in the bottom right most corner. And they can send their feedback by clicking send feedback. The feedback along with date and time will be recorded.

The figure 3.7 shows the list of feedback given by the users along with the time and date.



Figure 3.7: Feedback Section

The figure 3.8 shows feedback page provided for user to give their valuable feedback.

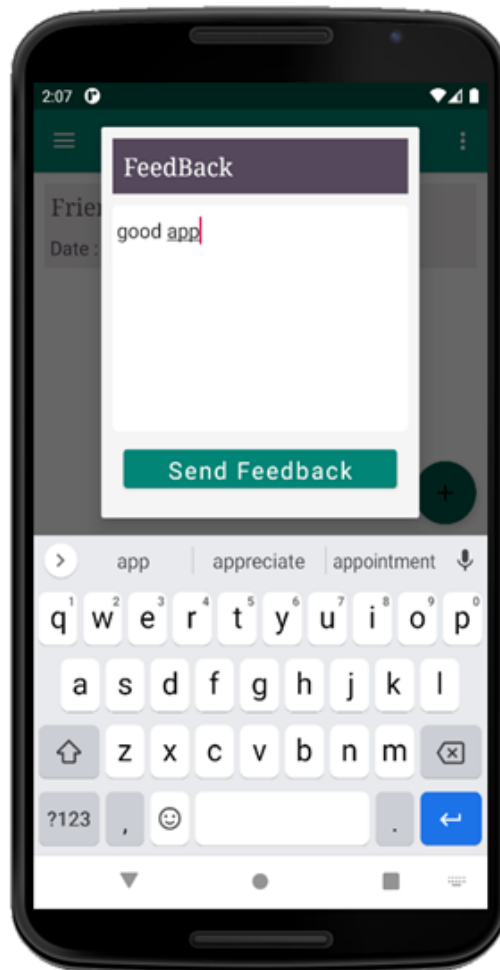


Figure 3.8: New Feedback Page

The user can update their password whenever required. By just clicking change password option in top right corner.

The figure 3.9 provides the user to change and update their password.

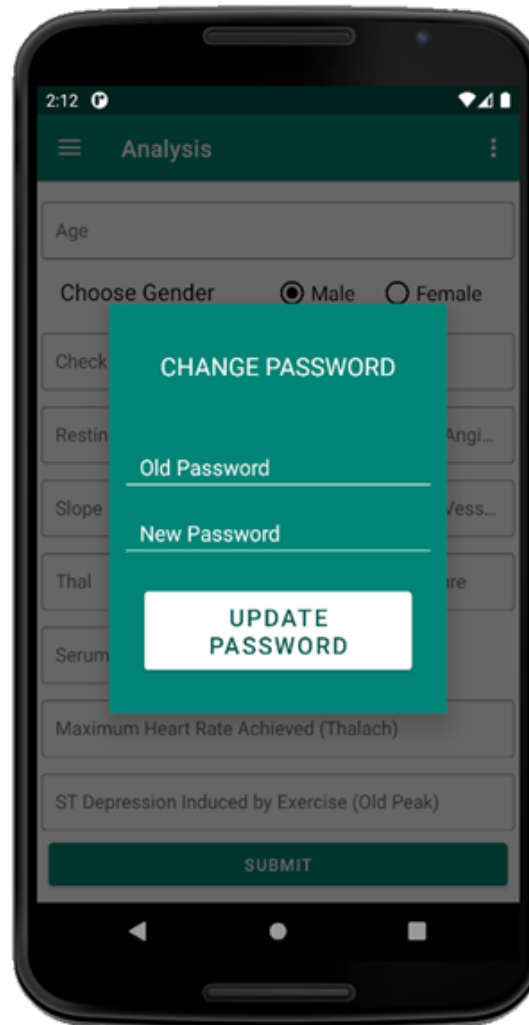


Figure 3.9: Change Password

The user can exit by selecting logout option.

The figure 3.10 shows the option for logout to user.

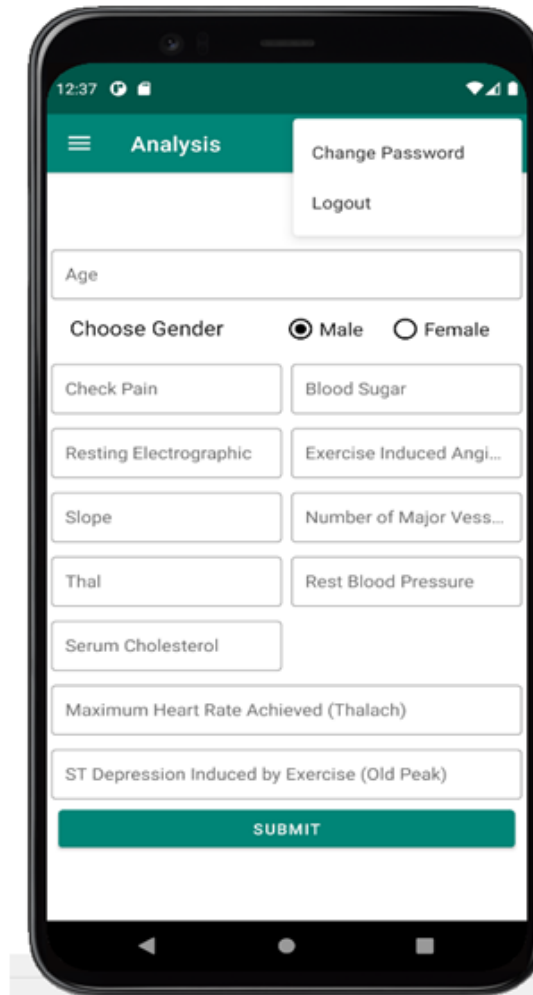


Figure 3.10: Logout

Upon successful validation of the admin, he/she is directed to the home page, where in they can select the required functionalities.

The available functionalities are :

- Doctors
- Training data
- User
- Feedback

- Logout

The figure 3.11 shows the home page of Admin consisting of doctor, training data, users, feedback and logout.

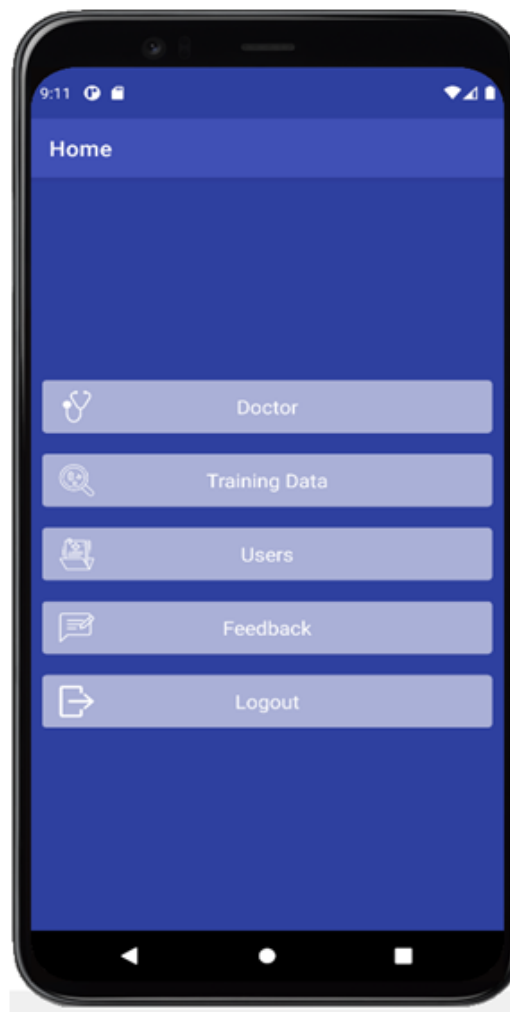


Figure 3.11: Home page of Admin

In the doctors section, the details like name, specialization and the address of the doctor is provided. These details can be further used by the user to contact the suitable doctor.

The figure 3.12 gives the list of doctors.

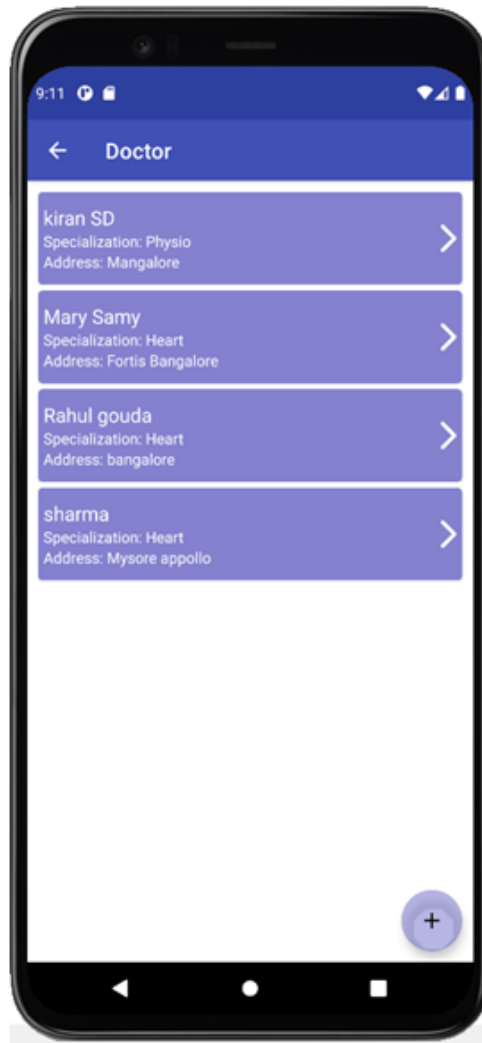


Figure 3.12: Update doctors

In the training data section, the data used to train the model is listed for a quick reference.

The figure 3.13 shows the list of data collected by users.

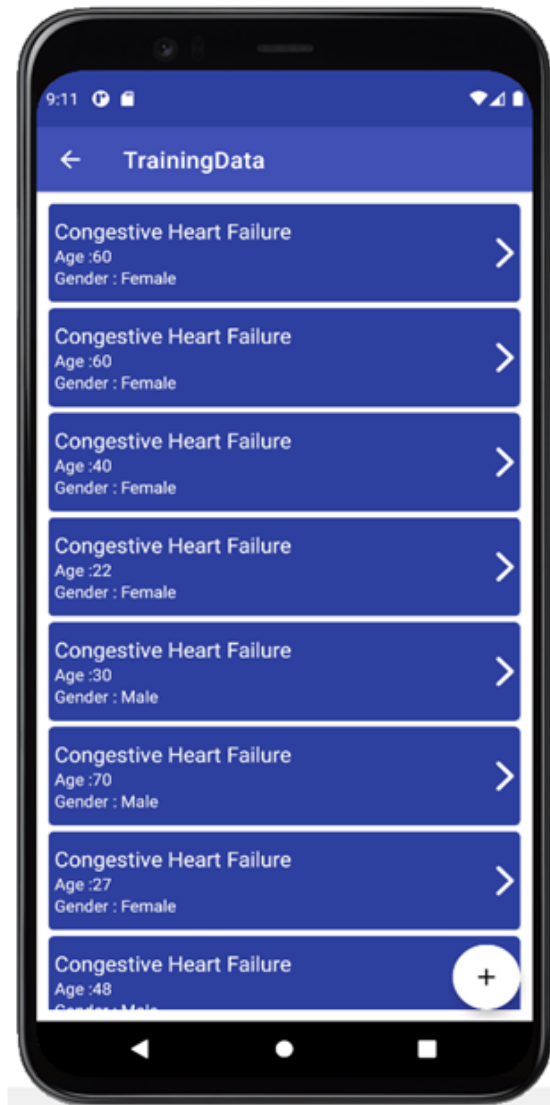


Figure 3.13: Training data

In the Users section, the list of the users are provided and upon clicking on a particular user more details about them is provided.

The figure 3.14 shows the list of users data.



Figure 3.14: View users

In the feedback section, the feedbacks provided by several users are listed along with name and date.

The figure 3.15 displays all the feedback given by users.



Figure 3.15: View feedback

3.2 Data Structures and Algorithms

3.2.1 Extraction of data

Purpose: Collection of data

Data Structures used: CSV file, DataFrame

Use Cases: Get data from the local disk and pre-process it and store it into dataframe.

Algorithm: Not using

Error handling: Removing data which is having null value.

3.2.2 Data Processing and Classifying

Purpose: To categorize the data

Data Structures used: DataFrame

Use Cases: To process and classify the data

Algorithm: Random Forest, Logistic Regression

Error handling: Splitting of data into training and testing data for higher accuracy.

3.3 Algorithms

3.3.1 Logistic Regression

Under the Supervised Learning approach, one of the most well-known machine Learning algorithms is logistic regression. It focuses on predicting dependent variables using independent factors, with the output being a probabilistic value, such as 0 or 1. Logistic Regression is much similar to Linear Regression but Regression problems are solved using Linear Regression whereas classification problems are solved using Logistic regression.

Figure 3.16 depicts a ROC (Receiver Operating Characteristic) curve which offers the overall performance of a class version for any class thresholds.

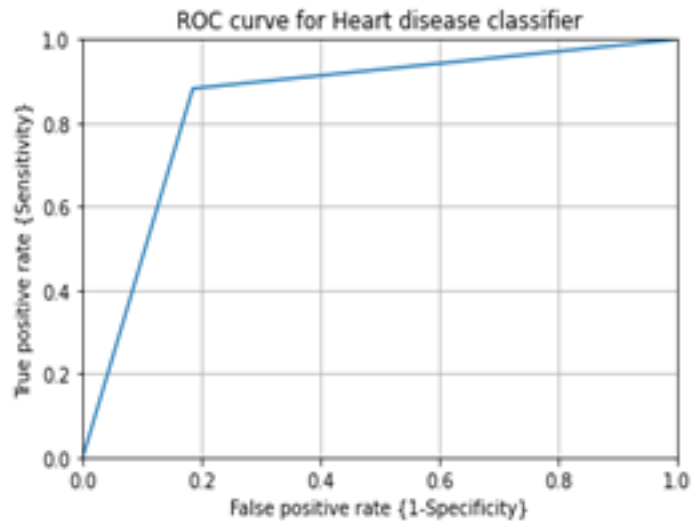


Figure 3.16: ROC-Receiver Operating Characteristic Curve

Figure 3.17 is the `regplot()` characteristic gives the plot of fit by using true or false from the data provided and also permits to specify whether or not to estimate the logistic regression model.

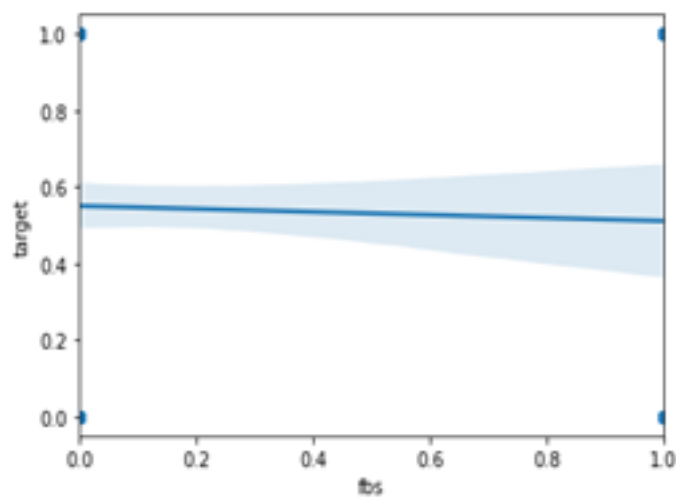


Figure 3.17: Regplot characteristic

Figure 3.18 depicts the mxm (m-represents range of classes) confusion matrix is

used for the assessment of the class model based on the known true values known.

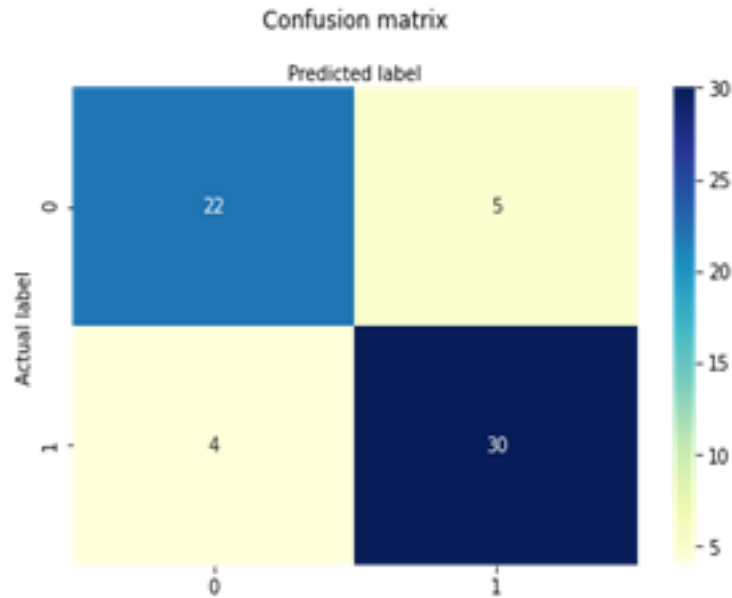


Figure 3.18: Confusion Matrix

3.3.2 Naive Bayes

The presence of one feature in a class is unrelated to the presence of any other feature, according to the Nave Bayes classification algorithm. Naive Bayes is a basic algorithm that may be quickly created to analyse enormous datasets. It predicts quickly and performs admirably across a wide range of classes.

The Nave Bayes is represented by the following equation.

$$P(x | y) = (P(y | x).P(x))/P(y)$$

- $P(x|y)$ – posterior probability of a class.
- $P(x)$ – probability of a class.
- $P(y|x)$ – probability of predictor class.

- $P(y)$ – probability of the prediction.

3.3.3 Support Vector Machine

Support Vector machine is another Machine Learning technique that may be used for both classification and regression, however it is more commonly used for classification. SVM (Support Vector Machine) mainly concentrates on creating the best decision boundary that can separate n-dimensional spaces into classes. Support Vector Machine considers support vectors to point to the best decision boundary.

3.3.4 Decision Tree

The decision tree is a supervised machine learning technique that can be used to do both regression and classification. A decision tree is a tree with internal nodes that represent datasets. The branches reflect the decision rules, while each leaf node represents the conclusion. There are two sorts of nodes in a decision tree: decision nodes and leaf nodes. The leaf nodes represent the output of decision trees, which are used to make decisions. The tree is constructed using the CART method. The CART algorithm stands for Classification and Regression Tree.

3.3.5 Random Forest

Random Forest uses the approach of supervised learning. It can be utilised for both classification and regression issues. Random Forest is based on ensemble learning, which is the act of mixing several classifiers to solve a complex problem and improve the model's performance. The random forest requires the least amount of training and predicts the outcome with great accuracy and efficiency.

3.4 UML diagrams

UML stands for Unified Modeling Language. It is an approach which models the software and documents it. It represents software components using diagrams.

3.4.1 Use case diagram

Use case diagram is a type of UML. It uses actors and usecases to model the functionality of a system. The figure 3.19 depicts user interaction and 3.20 depicts system interaction.

Figure 3.19 gives the brief introduction of system behaviour in the perspective of the user. The user is provided with following functionalities:

- Input the data - To provide details for registration.
- Change password - To update with new password.
- Check for doctors - To consult a doctor, with the available list of doctors.
- Check the type of heart disease - Upon the successful data given, system predicts the type of heart disease, if any.
- Add feedback - User has given facility to give feedback for the system.
- Logout - To logout from the system.

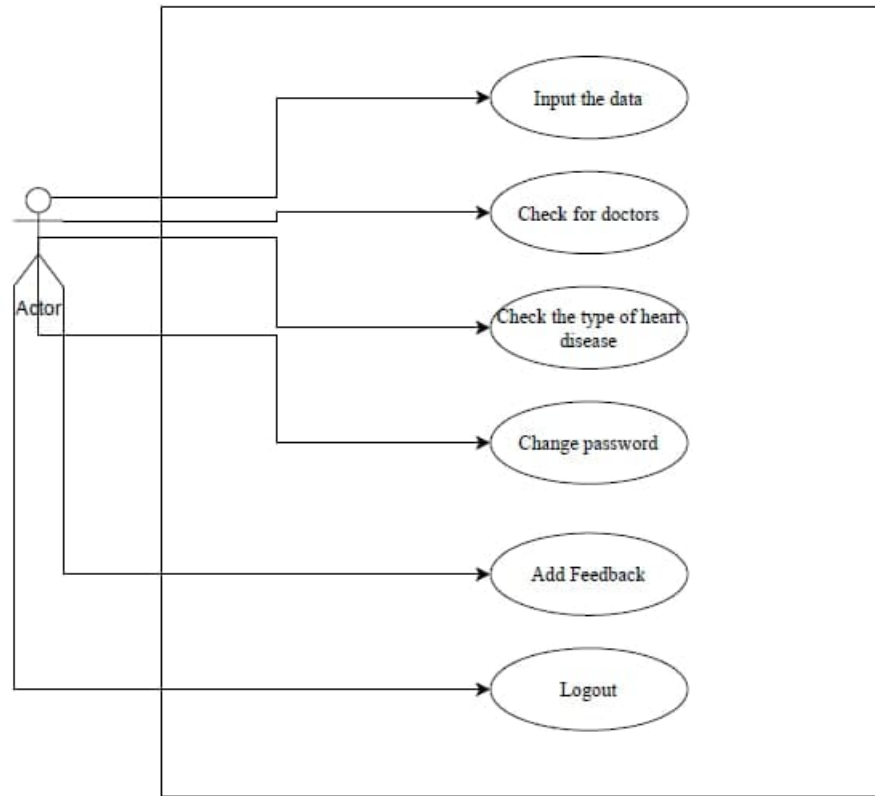


Figure 3.19: Use case diagram for user

Figure 3.20 gives the brief introduction of system behaviour in the perspective of the admin. Admin has following functionalities:

- User data - Admin has a provision to add or delete i.e, control over the user.
- Doctor information - Admin can add or delete or update with the doctors list.
- View feedback - Admin can view the feedbacks given by user.
- Training data - Admin provide training data to system to analyze the type of heart diseases.
- Logout - To logout of system.

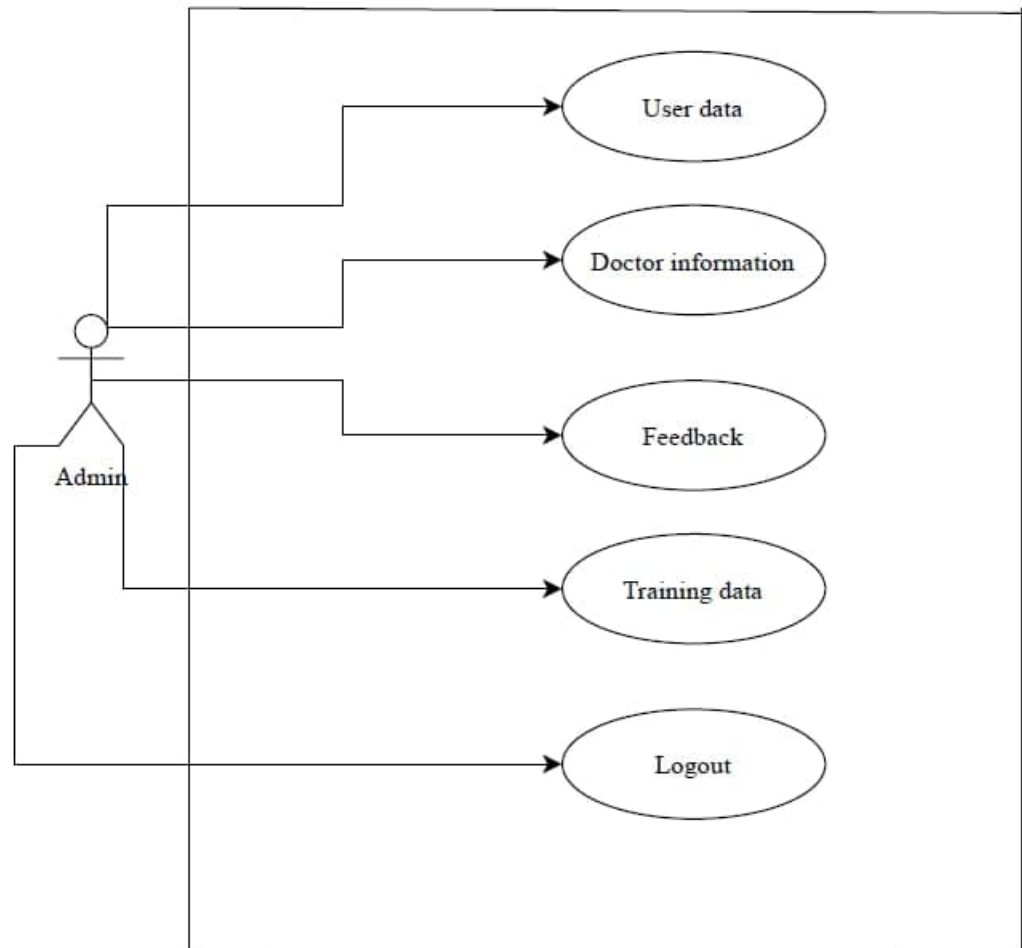


Figure 3.20: Use case diagram for admin

3.4.2 Sequence diagram

Sequence diagrams presents all the communication in a chronological manner. The figure 3.21 depicts the sequence diagram of the system with respect to the user. This figure depicts the system behaviour in the user perspective.

When once the user login with the credentials the database validates it before they

can access any other part of application. After successful login of user, they are provided with the analysis page where they enter the symptoms and system uses the random forest to predict the type of disease and based on analysis they prints the output.

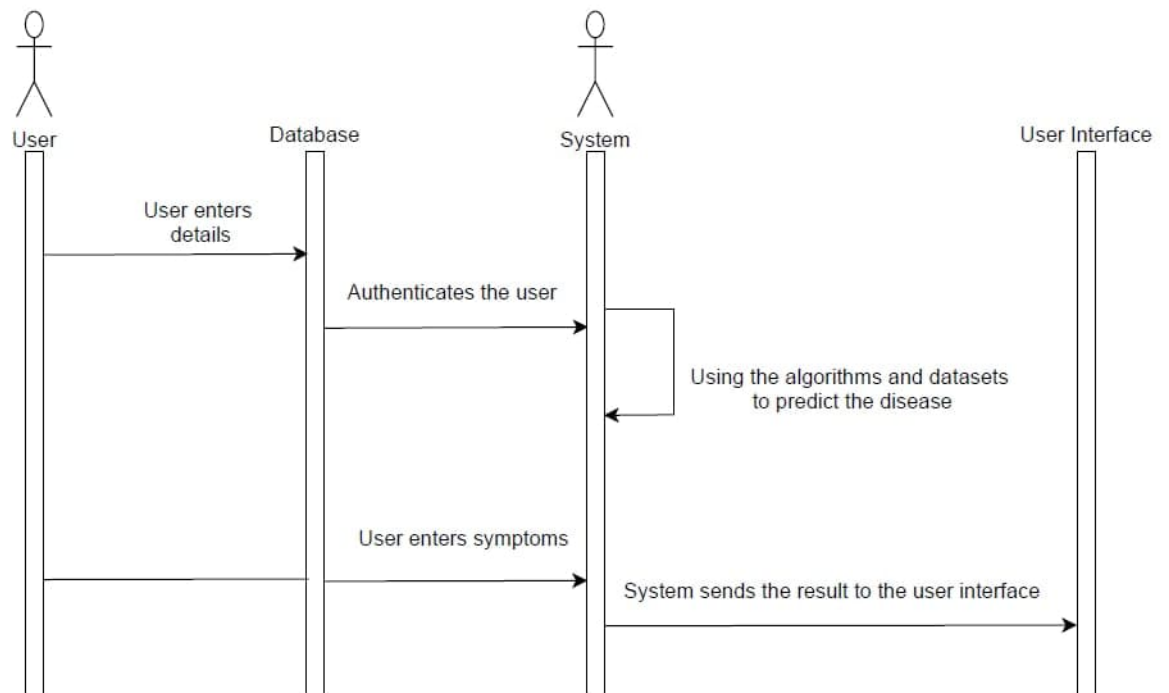


Figure 3.21: Sequence diagram for user

When Admin enters the details, Database connected authenticates the details. And provides with functionalities like viewing the details of user, the data provided for training, viewing the feedback given by the user and updating the details of doctors.

The figure 3.22 shows the sequence diagram for admin.

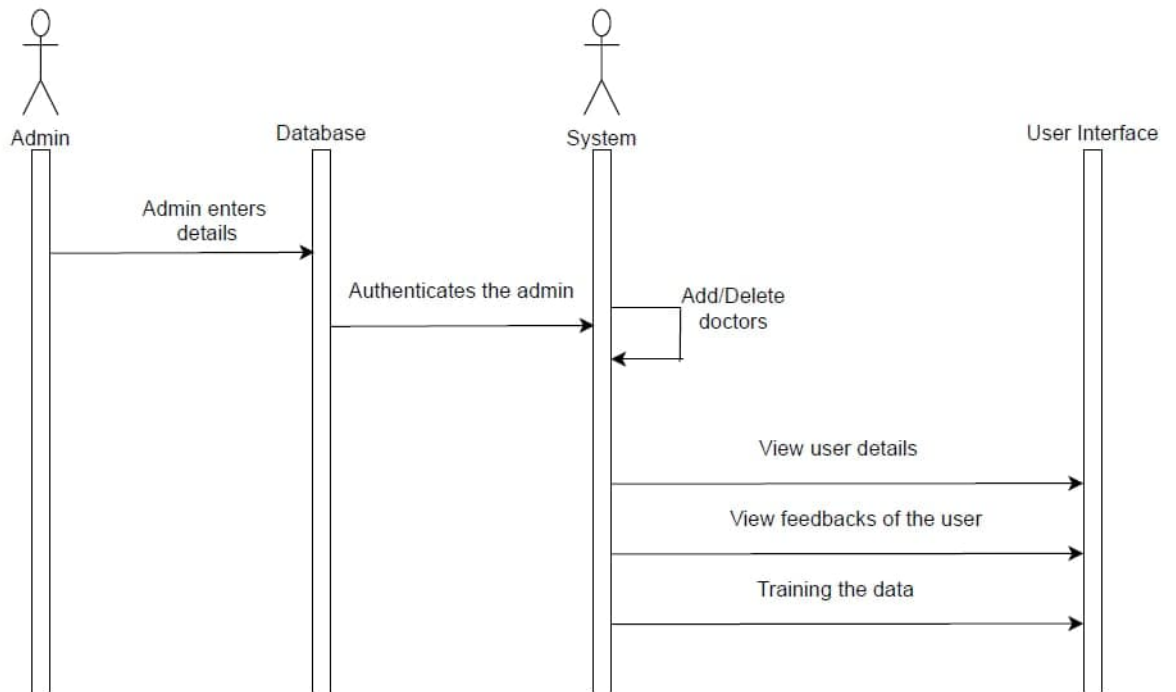


Figure 3.22: Sequence diagram for admin

3.4.3 Activity diagram

As part of UML, an activity diagram is used to depict the system's dynamic aspects. An activity diagram is a flowchart that depicts the movement of information from one action to the next.

The system checks if the user has already been registered or new user. If new user, then they have to register first before logging into the system. Upon login, the system authenticates the details provided. Once successful, they can enter the required details for analysis and get the predicted type of heart disease as output.

The figure 3.23 shows the activity diagram of system.

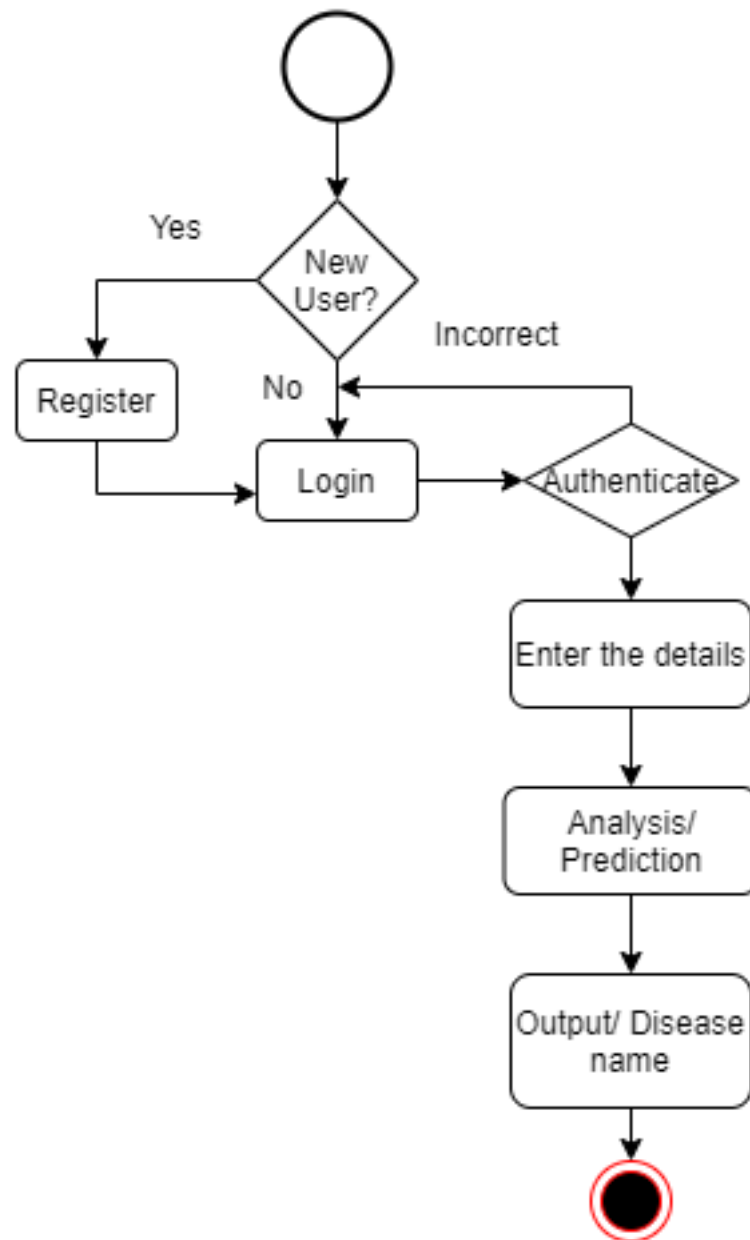


Figure 3.23: Activity diagram

3.4.4 Data flow diagram

Context Diagram is another name for DFD Level 0. It's a high-level overview of the entire system or process that's being studied or modelled. It's intended to be a

quick overview, presenting the system as a single high-level process with its external relationships.

The heart disease prediction system takes the user details and predicts the type of heart disease based on provided values for the required data.

The figure 3.24 gives the data flow diagram of the system at level 0.

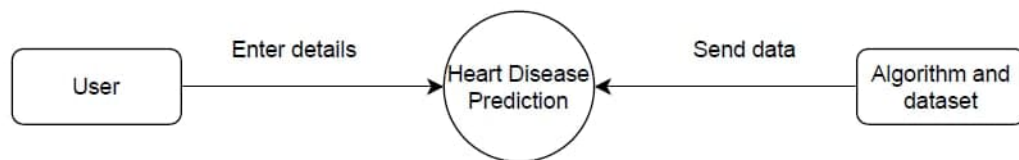


Figure 3.24: Level-0 Data Flow Diagram

A level 1 DFD lists all of the major sub-processes that make up the entire system. A level 1 DFD can be thought of as a "exploded perspective" of the context diagram.

The data provided by user is stored in database and an algorithm is applied to process and check the type of heart disease, if any. The result is displayed to user.

The figure 3.25 gives the data flow diagram of the system at level 1.

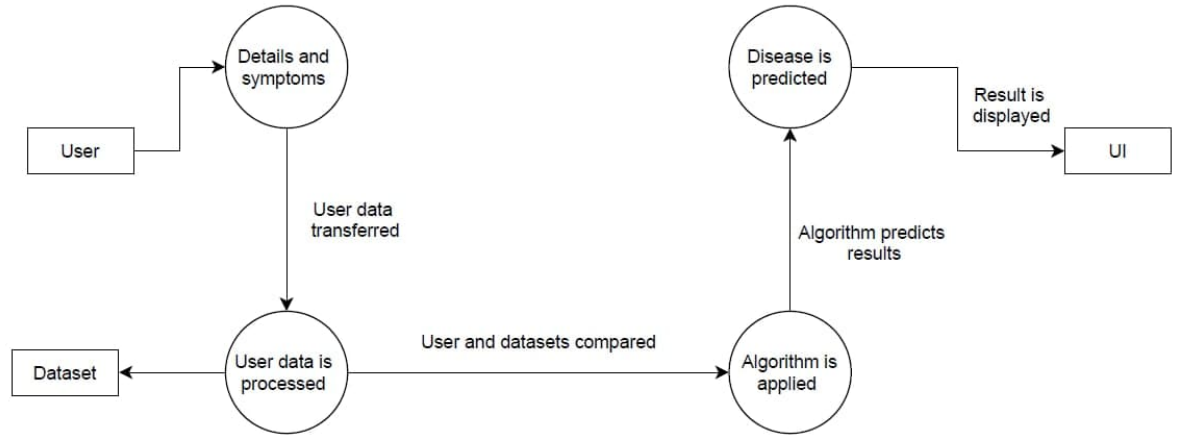


Figure 3.25: Level-1 Data Flow Diagram

3.5 Data Source/Database used and Formats

Dataset used The dataset used for prediction is 'Cleveland' taken from the UCI repository, which has the following attributes:

1. Age: Individual age.
2. Gender: Gender of individuals following the format: 1 = male, 0 = female.
3. Resting Electrographic: It has values of 0: normal, 1: abnormality of ST-T wave, 2: hypertrophy of left ventricle.
4. Serum Cholesterol: Taken in terms of mg/dl (standard unit).
5. Blood Sugar: Compared with 120mg/dl. 1: if it is >120mg/dl, else: 0.
6. Exercise-induced angina: 1: yes, 0: no.
7. Chest pain: Type of chest pain experienced by the individual and defined by 1: typical angina, 2: atypical angina, 3: non - anginal pain, 4: asymptotic.
8. Thalach: Maximum heart rate achieved by an individual.

9. Slope: Peak exercise ST segment, 1: up sloping, 2: flat, 3: downsloping.
10. The number of major vessels: (0-3) colored by fluoroscopy.
11. Thal: Thalassemia with 3: normal, 6: fixed defect, 7: reversible defect.
12. Old Peak: ST depression induced by exercise relative to rest.
13. Rest Blood Pressure: Taken in terms of mmHg.
14. Diagnosis of heart disease: Outputting whether the individual is suffering from heart disease or not: 0 = absence 1,2,3,4 = present.

The figure 3.26 shows the data used for the system.

A	B	C	D	E	F	G	H	I	J	K	L	M	N
age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
57	1	0	140	192	0	1	148	0	0.4	1	0	1	1
56	0	1	140	294	0	0	153	0	1.3	1	0	2	1
44	1	1	120	263	0	1	173	0	0	2	0	3	1
52	1	2	172	199	1	1	162	0	0.5	2	0	3	1
57	1	2	150	168	0	1	174	0	1.6	2	0	2	1
54	1	0	140	239	0	1	160	0	1.2	2	0	2	1
48	0	2	130	275	0	1	139	0	0.2	2	0	2	1
49	1	1	130	266	0	1	171	0	0.6	2	0	2	1
64	1	3	110	211	0	0	144	1	1.8	1	0	2	1
58	0	3	150	283	1	0	162	0	1	2	0	2	1
50	0	2	120	219	0	1	158	0	1.6	1	0	2	1
58	0	2	120	340	0	1	172	0	0	2	0	2	1
66	0	3	150	226	0	1	114	0	2.6	0	0	2	1
43	1	0	150	247	0	1	171	0	1.5	2	0	2	1
69	0	3	140	239	0	1	151	0	1.8	2	2	2	1

Figure 3.26: Dataset

Chapter 4

Implementation

This implementation chapter briefs about tools, technologies, coding standards followed, experimental setup, details of code integration, implementation workflow and non-functional requirements results.

4.1 Tools and Technologies

This project uses many software tools and technologies which are being briefed in this section.

4.1.1 Machine learning

The scientific study of algorithms and statistical models is known as machine learning. Artificial intelligence is a subset of it. Machine learning algorithms are created using mathematical models on training data, which aids in prediction and decision-making. Text analysis, document classification, speech recognition, and other applications also use machine learning methods.

4.1.2 Jupyter Notebook

Jupyter Notebook is a server-client application. The Jupyter Notebook App lets us edit and run notebooks on our browser. The application can be installed on a distant server or run on a PC without internet connectivity. The two primary components are kernels and a dashboard. A kernel is a program that runs and examines the code written by the user. The application's dashboard displays the note book documents and may also be used to control and shut off kernels if necessary.

4.1.3 Python

Python is a high-level, interpreted, object-oriented, general-purpose programming language. It's a programming language with multiple paradigms. It can be used for both object-oriented and structured programming. Functional and aspect-oriented programming are supported by many of its features. Python is a scalability oriented programming language. It also has versatility in problem-solving scenarios. Many firms and industries use it to produce a variety of applications.

4.1.4 Android Studio

Android Studio is Google's official integrated development environment (IDE), based on JetBrains' IntelliJ IDEA software and designed exclusively for Android development. It provides the most reliable tools for developing apps for every Android device. It can be downloaded for Windows, Mac OS X, and Linux operating systems. The parts of the android application are built using the JAVA programming language.

4.1.5 Azure data studio

Azure Data Studio is a cross-platform database tool for data professionals who use on-premises and cloud data platforms on Windows, macOS, and Linux. It has IntelliSense, code snippets, source control integration, and an integrated terminal for

a modern editor experience. With built-in visualisation of query result sets and customised dashboards, it is designed with the data platform user in mind. We can use it to query, construct, and administer our databases and data warehouses on the local computer or in the cloud.

4.2 Experimental Setup

This section includes hardware details, and other infrastructure details for obtaining the required output. The procedure as follows:

- Splitting dataset
- Logistic Regression
- Naive Bayes
- Support Vector Machine
- Decision Tree
- Random Forest

Using the sklearn package, the dataset has been divided into training and testing data in the ratio of 80:20. The training set has 242 observations where as testing dataset has 61 observations.

The figure 4.1 shows the split of dataset.

```
In [13]: from sklearn.model_selection import train_test_split

        pred = dataset.drop("target",axis=1)
        target = dataset["target"]

        X_Train,x_test,Y_Train,y_test = train_test_split(pred,target,test_size=0.20,random_state=0)

In [14]: X_Train.shape
Out[14]: (242, 13)

In [15]: x_test.shape
Out[15]: (61, 13)

In [16]: Y_Train.shape
Out[16]: (242, )

In [17]: y_test.shape
Out[17]: (61, )
```

Figure 4.1: Splitting dataset

Logistic regression is imported from `sklearn.linear_model` and module used is `LogisticRegression`. The training data is fit into model. Using testing data the accuracy is predicted.

The figure 4.2 shows the logistic regression algorithm.

```
In [18]: from sklearn.metrics import accuracy_score

In [19]: from sklearn.linear_model import LogisticRegression

logis = LogisticRegression()

logis.fit(X_Train,Y_Train)

Y_pred_lreg = logis.predict(x_test)

In [20]: Y_pred_lreg.shape
Out[20]: (61,)

In [21]: score_lreg = round(accuracy_score(Y_pred_lreg,y_test)*100,2)

print("The accuracy score achieved using Logistic Regression is: "+str(score_lreg)+" %")

The accuracy score achieved using Logistic Regression is: 85.25 %
```

Figure 4.2: Logistic Regression

Naive Bayes is imported from `sklearn.naive_bayes` and module used is `GaussianNB`. The training data is fit into model. Using testing data the accuracy is predicted.

The figure 4.3 shows the naive bayes algorithm.


```
In [22]: from sklearn.naive_bayes import GaussianNB

nv_by = GaussianNB()

nv_by.fit(X_Train,Y_Train)

Y_pred_nv_by = nv_by.predict(x_test)

In [23]: Y_pred_nv_by.shape
Out[23]: (61,)

In [24]: score_nv_by = round(accuracy_score(Y_pred_nv_by,y_test)*100,2)

print("The accuracy score achieved using Naive Bayes is: "+str(score_nv_by)+" %")

The accuracy score achieved using Naive Bayes is: 85.25 %
```

Figure 4.3: Naive Bayes

Support Vector Machine is imported from sklearn and module used is svm. The training data is fit into model. Using testing data the accuracy is predicted.

The figure 4.4 shows the support vector machine algorithm.

```
In [25]: from sklearn import svm

sv = svm.SVC(kernel='linear')

sv.fit(X_Train, Y_Train)

Y_pred_svm = sv.predict(x_test)

In [26]: Y_pred_svm.shape
Out[26]: (61,)

In [27]: score_svm = round(accuracy_score(Y_pred_svm,y_test)*100,2)

print("The accuracy score achieved using Linear SVM is: "+str(score_svm)+" %")

The accuracy score achieved using Linear SVM is: 81.97 %
```

Figure 4.4: Support Vector Machine

Decision Tree is imported from sklearn.tree and module used is DecisionTreeClassifier. The training data is fit into model. Using testing data the accuracy is predicted.

The figure 4.5 shows the decision tree algorithm.

```

In [32]: from sklearn.tree import DecisionTreeClassifier
max_acc = 0

for x in range(200):
    dtc = DecisionTreeClassifier(random_state=x)
    dtc.fit(X_Train,Y_Train)
    Y_pred_dtc = dtc.predict(x_test)
    cur_accuracy = round(accuracy_score(Y_pred_dtc,y_test)*100,2)
    if(cur_accuracy>max_acc):
        max_acc = cur_accuracy
        best_x = x

print(max_acc)
print(best_x)
dtc = DecisionTreeClassifier(random_state=best_x)
dtc.fit(X_Train,Y_Train)
Y_pred_dtc = dtc.predict(x_test)

81.97
11

In [33]: score_dtc = round(accuracy_score(Y_pred_dtc,y_test)*100,2)
print("The accuracy score achieved using Decision Tree is: "+str(score_dtc)+" %")

The accuracy score achieved using Decision Tree is: 81.97 %

```

Figure 4.5: Decision Tree

Random Forest is imported from sklearn.ensemble and module used is RandomForestClassifier. The training data is fit into model. Using testing data the accuracy is predicted.

The figure 4.6 shows the random forest algorithm.

```

In [34]: from sklearn.ensemble import RandomForestClassifier

max_accuracy = 0

for x in range(2000):
    rf = RandomForestClassifier(random_state=x)
    rf.fit(X_train,Y_train)
    Y_pred_rf = rf.predict(X_test)
    current_accuracy = round(accuracy_score(Y_pred_rf,Y_test)*100,2)
    if(current_accuracy>max_accuracy):
        max_accuracy = current_accuracy
        best_x = x

print(max_accuracy)
print(best_x)

rf = RandomForestClassifier(random_state=best_x)
rf.fit(X_train,Y_train)
Y_pred_rf = rf.predict(X_test)

95.08
1818

```

Figure 4.6: Random Forest

The accuracy acquired through the algorithms is depicted in Table 4.1.

Table 4.1: Accuracy obtained from various algorithms

Algorithm	Accuracy score
Logistic Regression	85.25%
Naïve Bayes	85.25%
Support Vector Machine	81.97%
Decision Tree	81.97%
Random Forest	95.08%

From Table 4.1, the accuracy of Logistic Regression is 85.25%, Naïve Bayes is 85.25%, Decision Tree is 81.97%, Support Vector Machine is 81.97% and Random Forest is 95.08%.

4.3 Coding Standards followed

This project uses Android Studio and Java for creating the android application. The application uses significant names for functions. Comments are written to understand the code. Indentation is maintained. The project imports required modules wherever necessary. Some of the programming norms used in this application are discussed in this section.

4.4 Code Integration details

This section presents details on code integration. The main phases are

- Environmental setup: The dependencies to configure and to run this project are fair in number. Installation of python, Android Studio and Machine Learning libraries are essential to run the project.
- Project setup and development: The project creates an android application using android studio and implements classification algorithm using machine learning on Jupyter notebook using python. Integration is done in the android application which considers the data stored in the azure data studio and provides the proper predictions.

4.5 Implementation work flow

1. The pre-processing modules that are required are imported.
2. The data is separated into two categories: training and testing.
3. To discover the best algorithm, different machine learning algorithms such as logistic regression, Naive Bayes, Support Vector Machine, KNN algorithm, Decision tree, and Random forest algorithms are used. Because the random forest method has the best predictive accuracy, it is utilised to determine the kind of heart disease.

4. The user provides the information needed for the prediction.
5. The data in the database is used to train the machine learning model.
6. To forecast the kind of heart disease, the input data is compared to the trained machine learning model that uses the data contained in the database.

4.6 Execution Results and Discussions

After successfully inputting the values to the system, it uses the random forest for the disease type prediction.

Figure 4.7 shows no disease found in the user as it indicates a healthy heart.

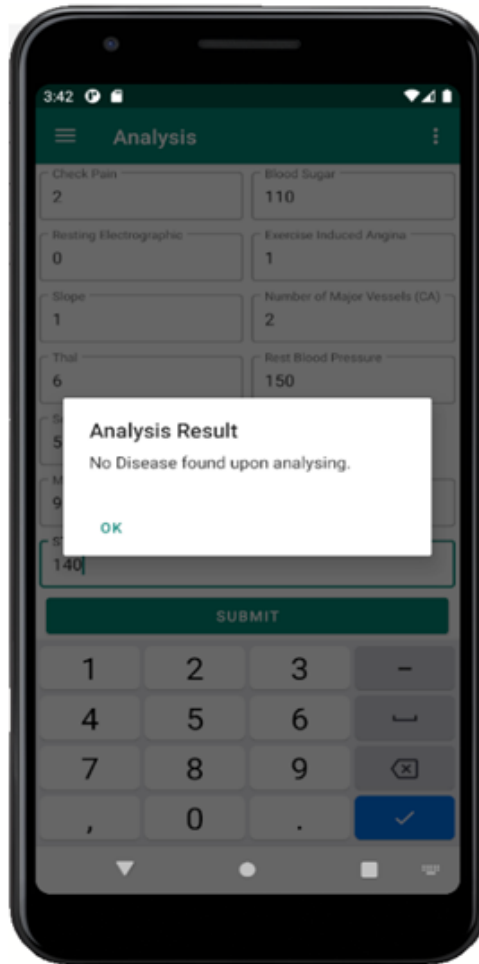


Figure 4.7: Output: No disease

Figure 4.8 predicts that the user has Coronary artery a disease type wherein major blood vessels of the heart are damaged which in turn limits the flow of blood to the heart by narrowing coronary arteries.

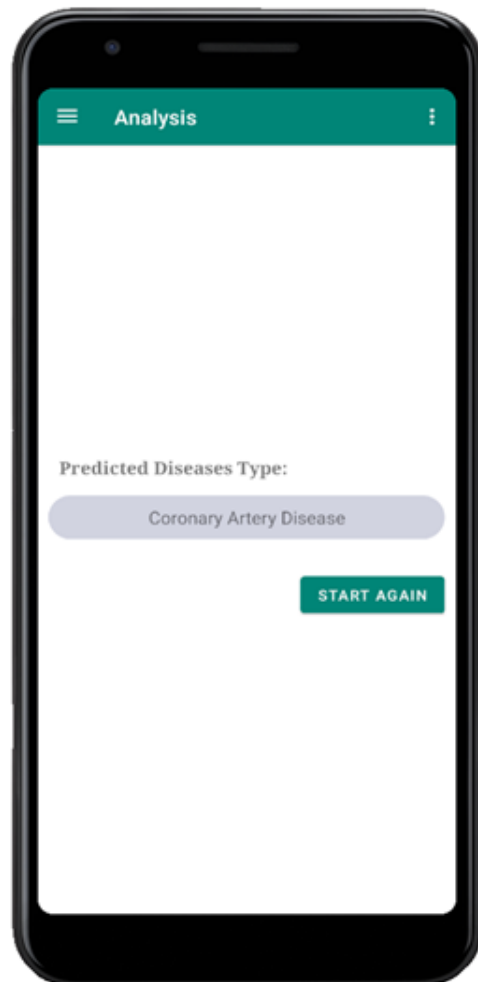


Figure 4.8: Output: Coronary Artery Disease

Figure 4.9 shows that the user is suffering from Congestive heart failure which is caused due to inadequate blood supply to the heart.

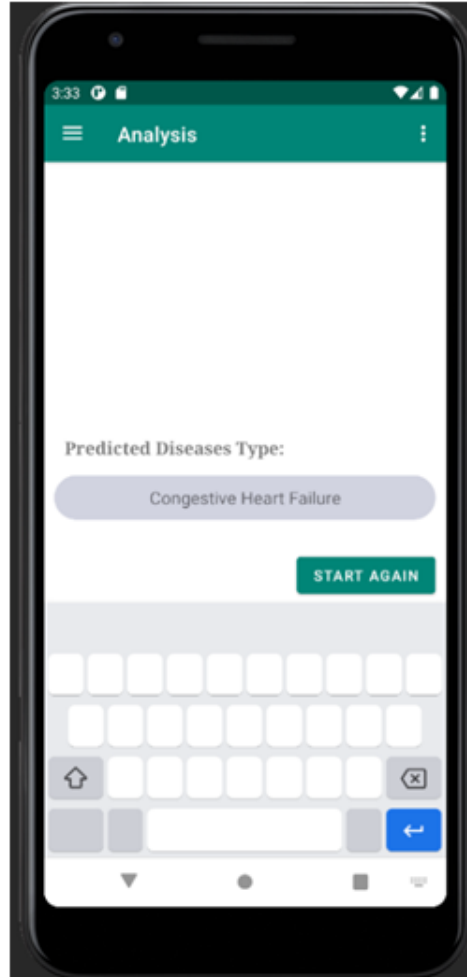


Figure 4.9: Output: Congestive Heart Failure

4.7 Non-functional requirements results

These requirements are the parameters in system engineering and requirements engineering that specifies criteria to judge the operations rather than specific behaviour of the system. They often compare with functional requirements. The project delivers the following non-functional requirements.

- Performance: The project uses Random forest algorithm for the heart disease classification which is suited to find a high accuracy among large amounts of

data.

- Reliability: The project gets data from Microsoft Azure feed which are highly reliable. Libraries used are more efficient and works effectively.
- Usability: This can be easily modified and made changes so that it performs optimally under various situations.
- Maintainability: Easy and less maintenance is required, robust and easy to update.
- Portability: It can be accessed on any android phones and the user friendly interface makes it easy to use.

Chapter 5

Testing

This section briefs on testing, testing workflow and details of test cases. Software testing is a methodology that ensures the proper functionality of a software and also to verify if the application meets the specified requirements or not. They ensure that the application is free from the bugs and defects. This ensures the delivery of a high quality end product to the end user.

5.1 Test workflow

Testing is performed to identify the defects in the product at the early stage before the product actually goes into the production. The testing can be done at various levels such as module, sub-module level. The complete product can be tested for checking its integration among the modules and also to ensure the product is fault free end to end. There are various testing methods. These methods require various strategies and they test the various components forming the application.

Before a product is deployed, number of tests are performed on the developed application. Various test methods are organized into a class which is referred as test. Tests are conducted to verify that the workflow of each of the cases reject the business case of the application. Test suites are the group of tests that have a specific objective.

5.1.1 Testing Methodologies

- Objectives of testing
- Test Cases
- Black Box Testing
- White box Testing
- Unit Testing
- Integration Testing
- Functional Testing
- Output Testing

5.1.2 Objectives of testing

Testing is the procedure for locating flaws in a program. The goal of the testing is to uncover mistakes that aren't immediately identified by the developer or the user but cause issues with the application's workflow. It is a method of systematically discovering the faults or bugs in a code base. The main benefit of testing is that it ensures that the product meets the specifications established by the end users before it is released for usage. It also checks whether it is logically correct. Testing ensures that the application workflow is maintained and it works as intended. The testing is done from the users' perspective, so all possible edge cases will be analysed reducing the risk of breaking of application. It also ensures that the product has higher quality which sustains for more period of time.

5.1.3 Test Cases

Test cases are the basic unit of the testing paradigm. They may be used to test a single entity or they can be used to test the working of an entire module or it can be used to test the integration among the modules. Test case has only two possibilities, either a test case can pass or it can fail. If a test case is passed, it means that it is working exactly as intended. If the test case fails then we can infer that

there is something wrong in the work flow or the code base. The more number of test cases indicate that the product is well tested. That is why it is preferred to write more test cases verifying the end to end working of the application considering the boundary conditions as well helps to build a more reliable product. Large number of test cases helps to measure a project accuracy and effectiveness.

5.1.4 Black Box Testing

Black box testing is another name for behavioural testing. This form of testing is primarily concerned with the application's functionality. In general the person who is testing need not have the idea of the internal working of the application to perform this testing. Following types of errors can be addressed using this type of testing.

- Errors corresponding to missing or incorrect interface.
- Data structure or data storage related errors.
- Errors caused by the initialization and termination of the entities.

5.1.5 White Box Testing

White box testing is another name for glass box testing. It's a test-driven strategy to identify application flaws. To test using this method, the tester must have a thorough understanding of the application's internal workings. This method can be utilised in the following situations.

- A module's standalone modules have all been called at least once.
- The binary nature of the logical judgments should be adhered to.
- Tester needs to check the boundary values to ensure that it is consistent and valid.

5.1.6 Unit Testing

Unit testing is responsible for testing the application's most fundamental unit. A unit is a short amount of code that can be considered its own program. Unit testing is used to test such things. It is used to check whether the module has any errors and its validity. The module is verified to check whether it is being correctly executed. The basic working of the module and its sub-modules is verified. Also the interactions between such modules are also tested to ensure that the communication happens across the modules. Unit testing is done all along the product development phase. Finally once the product development is complete, the units are tested to check if they are still working as intended. If all are working then product is said to have a better quality as the basic units are all correctly working.

5.1.7 Integration Testing

Integration testing is majorly carried to verify whether the modules that are grouped together are working as intended or not. Grouping of multiple modules may cause some friction among them due to mismatch in the code base. This testing is done to ensure that there is no such problem occurring on association of the modules with each other. The issue can also be caused by the data structures that we are using. This convergent testing was developed to find the faults in the interfaces. Primarily individual modules are tested and then they are integrated and tested.

5.1.8 Functional Testing

Black box testing can be replaced by functional testing. This method of testing focuses on the system's functional components while ignoring the application's logical aspects.

5.1.9 Output Testing

Output testing is carried out post integration testing where the output is examined. Even though there is no problem with the integration of the modules the output may not be as expected. As a result, output testing aids in determining whether or not the required or desired output is produced.

5.2 Test case details

5.2.1 Test case id: TC01

Unit to test: User Validation

Assumptions: To check if the user has provided correct credentials.

Test data: Email id and password

Steps to be executed:

1. User enters the email id and password
2. Database connected to it validates the input
3. If the credentials are valid the user is logged in successfully.

Expected result: The database is well synchronized to validate the user

Actual result: Successfully validates the user

Pass/Fail: Pass

5.2.2 Test case id: TC02

Unit to test: Launching the android application

Assumptions: The gradle connection is successful for launching the emulator.

Steps to be executed: Launch the android application.

Expected result: The android application is launched.

Actual result: Successfully launches the application.

Pass/Fail: Pass

5.2.3 Test case id: TC03

Unit to test: Predicting the type of heart disease

Assumptions: Coronary artery disease , Congestive heart failure and no disease is to be predicted

Steps to be executed:

1. Division of the data set into training and testing data.
2. Enter the required attribute details.
3. Click on submit.

Expected result:Heart disease type prediction

Actual result: Successful in heart disease type prediction.

Pass/Fail: Pass

Chapter 6

Conclusions and Future Scope

6.1 Conclusion

The major goal of this research is to develop a better model for predicting an accurate heart disease in order to reduce the death rate caused by the lack of an adequate automated system. The Cleveland data set from UCI is used to train and evaluate the model in this study. To determine the most accurate prediction, it compares Logistic Regression, Nave Bayes, Support vector machine, Decision tree, and Random forest methods. The Random forest algorithm has provided a comparatively precise result for the study with an accuracy of 95.08%. An android application is developed and random forest algorithm is used to predict the type of heart disease. The various types of disease it predicts are coronary artery, heart failure, stroke, and no heart disease. The android which is being integrated with the machine learning algorithm prepared has been successful in providing the expected results.

6.2 Future Scope

- The image processing algorithms can be used to incorporate a feature of collecting the image format input from the user like the scan and ECG's to probably make enhanced prediction than the one achieved so far in the project.
- A extra feature can be enabled to help the user to have a real time conversation with the available doctors.

Appendix A

Abbreviations

UML - Unified Modeling Language

ML - Machine Learning

MVT - Model View Template

SVM - Support Vector Machine

KNN - K-nearest neighbour

DB - DataBase

Bibliography

- [1] Chaitrali S. Dangare Dr. Sulabha S. Apte. Improved study of heart disease prediction system using data mining classification techniques. *International Journal of Computer Applications*, 47, 2012.
- [2] K Subhadra Vikas B. Neural network-based intelligent system for predicting heart disease. *International Journal of Innovative Technology and Exploring Engineering*, 18, 2019.
- [3] Devansh Shah Samir Pate Santosh Kumar Bharti. Heart disease prediction using machine learning techniques. *Springer Nature journal*, 2018.
- [4] C. Beulah ChristalinLatha S. CarolinJeeva. Improving the accuracy of prediction of heart disease risk based on ensemble classification techniques. *Informatics in Medicine Unlocked*, 2019.
- [5] Adil Hussain She Dr. Pawan Kumar Chaurasia. A review on heart disease prediction using machine learning techniques. *Reasearch Gate*, 2019.
- [6] Archana Singh Rakesh Kumar. Heart disease prediction using machine learning algorithms. *International Conference on Electrical and Electronics Engineering*, 2020.
- [7] ApurbRajdhan Milan Sai Avi Agarwal Dundigalla Ravi Dr. PoonamGhuli. Heart disease prediction using machine learning. *International Journal of Engineering Research & Technology*, 9, 2020.

- [8] RishabhMagar Rohan Memane SurajRaut Prof. V.S. Rupnar. Heart disease prediction using machine learning. *Journal of Emerging Technologies and Innovative Research*, 7, 2020.
- [9] Senthilkumar Mohan Chandrashekar Thirumalai Gautam Srivastava. Effective heart disease prediction using hybrid machine learning techniques. *IEEE Access*, 2019.
- [10] Sarangam Kodati Dr. R Vivekanandam. Analysis of heart disease using in data mining tools orange and weka. *Global Journal of Computer Science and Technology*, 18, 2020.

Project Planning AY-2020-21

Activites	No.Of Week s	Plan/ Actual	Septmeber				October				November				December				January			Feb		March			April				May				June				July			
			1	2	3	4	1	2	3	4	1	2	3	1	2	3	4	1	2	3	1	2	1	2	3	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3		
Problem Identification and Literature Survey	6W	Plan																																								
		Actual																																								
Software Requirments and Specifications	3W	Plan																																								
		Actual																																								
Architecture, Design and Prototype	4W	Plan																																								
		Actual																																								
Implementation	10W	Plan																																								
		Actual																																								
Testing and Validation	3W	Plan																																								
		Actual																																								
Project Closure - Results Observations -Demonstration -Report Writing	3W	Plan																																								
		Actual																																								

 PLAN
 ACTUAL

COST ESTIMATION

A cost estimate is an approximation of the cost of a program, project or operation. In this project, most of the work is done using open source software.

WHY COST ESTIMATION?

- Cost estimations are prepared to different ends throughout the project lifecycle.
- Goal is to provide input for investment decisions.
- The cost estimate is used to determine the size of the required investment to create or modify assets.
- The cost estimate is a deliverable that serves the decision-making process at each phase of the project lifecycle.

Elements of Cost Estimation in Project Management

There are two types of Cost Estimation:

Direct Cost: Direct Cost is associated with single area such as particular project or department, this includes materials, equipment and fixed labour.

Indirect Cost: Indirect Cost is incurred by the organization at large scale such as quality control and utilities. Within these two types, other types of element that Cost Estimation will take into account are:

- **Labour:** The cost of the project team members working on the project, both in terms of wages and time. We being the team of 4 divided the work equally among ourselves, it took total duration of 10 months to complete this project by working approximately 2 to 3 hours per day.
- **Internet Charges:** The internet charges are Rs. 250/- per month per person, which comes around Rs. 10,000/- for 10 months.
- **Other miscellaneous cost:** Other miscellaneous costs are Rs. 800/- per person, which comes around Rs. 3200/-.
- **Materials and Equipment:** The cost of the resources required for the project, from physical tools to software to legal permits. Our project does not involve any hardware components and all the software technologies used are free of cost.
- **Research Paper:** The cost for registration for conference is Rs. 6500/-.
- **Printing charges:** The cost for printing one copy of report is Rs. 400/-. For 6 copies Rs. 2400/-.

Name	Cost
Internet Charges	10,000
Miscellaneous cost	3,200
Research Paper	6,500
Printing charges	2,400
Total	22,100

PO ATTAINMENT

Programme Outcomes (POs):		Task Preformed	Attainment				
			Excellent 5	Very Good 4	Good 3	Fair 2	Poor 1
PO1	Engineering knowledge	1. Applied the knowledge of Machine Learning, and Programming Languages and Software Engineering.	✓				
PO2	Problem analysis	1. Literature Survey done on "Heart Disease Prediction using Machine Learning Algorithms". 2. The objectives of the project were set. 3. Knowledge of Machine Learning, Programming and Software Engineering was found to be useful in implementing the project	✓				
PO3	Design/development of solutions	1. Solutions are developed using incremental model.	✓				
PO4	Conduct investigations of complex problems	1. Requirements for “Heart Disease Prediction using Machine Learning Algorithms” are gathered through Literature Survey. 2. Analyzed the problems. 3. Suitable solutions to meet the requirements are developed.		✓			
PO5	Modern tool usage	1. Jupyter, Python, Android studio, Java, Azure data studio are used.	✓				
PO6	The engineer and society	1. This project helps in detecting the type of heart disease to reduce the death rate.		✓			

Programme Outcomes (POs):		Task Preformed	Attainment				
			Excellent 5	Very Good 4	Good 3	Fair 2	Poor 1
PO7	Environment and sustainability	1. This project is sustainable in every aspect as it uses jupyter and azure data studio.		✓			
PO8	Ethics	1. This project is used to get the type of heart disease. 2. References are quoted. 3. Report is prepared by students and plagiarism check is made with turnitin software.		✓			
PO9	Individual and team work	1. Each student took up the responsibility of executing one module of the project. 2. The report content was contributed by each of the team members. 3. Integration of the modules was done as a team work. 4. Incorporating the suggested changes were done. 5. As a team, presentations and demo of the project was given.	✓				

Programme Outcomes (POs):		Task Preformed	Attainment				
			Excellent 5	Very Good 4	Good 3	Fair 2	Poor 1
PO10	Communication	1. Phase-wise presentation and Demo of progress of project work before the panel. 2. Presentation and Demonstration of project before Industry Experts. 3. Preparation of Report spread across the entire Semester. 4. Regular interaction with Guide and Panel members to incorporate the suggestions given during evaluations. 5. Answering queries during presentations and Demos.	✓				
PO11	Project management and finance	1. Project Scheduling using Gantt Chart. 2. Maintaining Project Diary. 3. Estimating Man Hour Requirement.		✓			
PO12	Life-long learning	1. Working on Machine learning. 2. Reading papers and articles on Heart disease prediction.	✓				



Digital Receipt

This receipt acknowledges that Turnitin received your paper. Below you will find the receipt information regarding your submission.

The first page of your submissions is displayed below.

Submission author: Nirmala M B
Assignment title: UG major paper
Submission title: UG Report
File name: from_abstract_3.pdf
File size: 2.05M
Page count: 72
Word count: 9,437
Character count: 49,099
Submission date: 03-Aug-2021 10:34AM (UTC+0530)
Submission ID: 1627234348

Abstract

The world which we see today is getting advanced every year. This is because some of the most promising elements, such as field development industrialisation, globalisation, and several other aspects of science and technology additional considerations. Considering the remarkable development in surroundings we can also see a drastic change in the human's health. Humans are facing a lot of health issues because of the modern food and the lifestyle they have adopted. Many diseases are caused by junk food and a sedentary working culture.

Heart disease can also be called cardiovascular disease which has become one of the most hazardous diseases. Its death rate is increasing for decades. This is due to the way the people lead their life in a sedentary style or by their food habits. From the WHO (World Health Organization) statistics around 17.9 million people die globally every year due to cardiovascular diseases which contributes to 31% of the deaths worldwide. Cardiovascular diseases are related to both heart and blood vessels disorder. These also include rheumatic, cerebrovascular, and coronary heart diseases. Heart attacks and strokes make up to 80% of cardiovascular diseases. The factors affecting cardiovascular disease can be classified as habitual and physiological risk factors. Smoking and drinking are two of the most common risk factors. Over-consumption of alcohol and caffeine, as well as stress. Tobacco use raises the risk of cancer. By a factor of two or three, the chances of dying are increased. Physical inactivity, as well as other physiological factors, variables such as high blood pressure, obesity, lipids, obesity, diabetes, and glucose. The heart condition is affected by hypertension, high blood cholesterol, and pre-existing heart abnormalities.

UG Report

ORIGINALITY REPORT

20%

SIMILARITY INDEX

14%

INTERNET SOURCES

9%

PUBLICATIONS

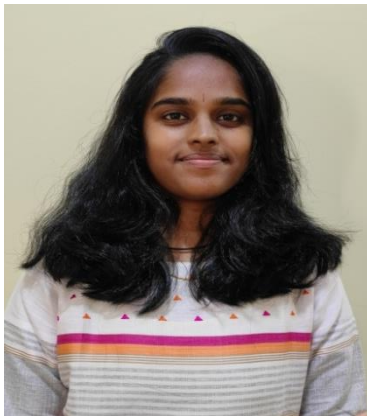


12%

STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Visvesvaraya Technological University, Belagavi Student Paper	1 %
2	Submitted to King's Own Institute Student Paper	1 %
3	stackoverflow.com Internet Source	1 %
4	thecleverprogrammer.com Internet Source	1 %
5	Submitted to Siddaganga Institute of Technology Student Paper	1 %
6	usermanual.wiki Internet Source	1 %
7	article.sciencepublishinggroup.com Internet Source	<1 %
8	sersc.org Internet Source	<1 %
9	www.ijert.org	

VITAE

<p>Name: Haarika Reddy K R USN: 1si17cs041 DOB: 26/11/1999 Permant Address: #193, Jakka Reddy Street, Kadagathuru, Madhugiri taluk, Tumkur dist. Phone No. 9902115620 Email: haarikareddykr26@gmail.com CGPA: 9.03 (Upto 7th sem) Placed: Yes, Target CTC: Rs. 11,73,500/-</p>	
<p>Name: Mohit Sah USN: 1si17cs060 DOB: 22/05/2000 Permant Address: "Radhe mai", Birgunj, Parsa, Nepal Phone No. 7619106154 Email: lalsahmohit@gmail.com CGPA: 7.97(Upto 7th sem) Placed: No CTC: -</p>	
<p>Name: Shrinidhi Shastry USN: 1si17cs105 DOB: 06/07/1999 Permant Address: "Sadhane", opposite to narayanappa garden, devnur church road, behind Nalanda convent, sapthagiri extension, shrinidhi layout. Phone No. 7892309804 Email: shrinidhishastry1999@gmail.com CGPA: 7.25 (Upto 7th sem) Placed: Yes, Cognizant CTC: Rs 4.5lpa</p>	

Name: Veeksha V Murthy
USN: 1si17cs128
DOB: 04/12/1998
Permant Address: "Sri Ranga", 2nd
main, 3rd cross, TPK Road,
Sapthagiri extension, Tumkur
Phone No. 9481177312
Email:
veekshaamurthy@gmail.com
CGPA: 8.39(Upto 7th sem)
Placed: Yes, Cisco
CTC: Rs. 14,33,000/-

