Tribhuvan University

Institute of Science and Technology

A Project Report

On

"Diabetes Prediction System Using Logistic Regression and Linear Regression"

Submitted To:

Office of the Dean

Institute of Science and Technology,

Tribhuvan University

Kritipur, Nepal

Under the Supervision of

Mr. Hiranya Prasad Bastakoti

A Project report submitted for the Partial Fulfillment of the Requirements of Bachelor of Science in Computer Science and Information Technology (BSc. CSIT) Of 7th Semester of Tribhuvan University, Nepal

Submitted By
Bimala Gadtaula (T.U. Roll No. 20987/075)
Dinesh Lohani (T.U. Roll No. 20989/075)
Soniya Paudel (T.U. Roll No. 21014/075)

May 2023

Tribhuvan University

Institute of Science and Technology

DECLARATION

I hereby declare that a project work entitled “Diabetes Prediction System Using Logistic Regression” submitted to the Institute of Science and Technology, Tribhuvan University, is an original piece of work done in the form of college project for Bachelor in Computer Science and Information Technology Program under the supervision of Mr. Hiranya Prasad Bastakoti.

Signatures of candidate:

Bimala Gadtaula

Dinesh Lohani

Soniya Paudel

Date: May 2023

Tribhuvan University

Institute of Science and Technology

Asian School of Management and Technology

LETTER OF RECOMMENDATION

It is my pleasure to recommend that a report on "**Diabetes Prediction System Using Logistic and Linear Regression**" has been prepared under my supervision by **Bimala Gadtaula, Dinesh Lohani**, and **Soniya Paudel** as a partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Information Technology. This project represents an original contribution to the field and utilizes Logistic and Linear Regression techniques to predict diabetes, demonstrating promising potential for future development in this area.

……..

Mr. Hiranya Prasad Bastakoti

Supervisor

Date: May 2023

Tribhuvan University

Institute of Science and Technology

Asian School of Management and Technology

LETTER OF APPROVAL

This is to certify that the project work entitled "Diabetes Prediction System Using Logistic and Linear Regression" submitted by **Bimala Gadtaula, Dinesh Lohani, Soniya Paudel** is satisfactory in the scope and generality as a project in the partial fulfillment of the requirement for Computer Science and Information Technology.

Evaluation Committee

……	…….
ER. Anil Lal Amatya	**Program Co-Ordinator**
Principal	
……	……...
Mr. Hiranya Pd. Bastakoti	**External Examiner**
Supervisor	

ACKNOWLEDGEMENT

We would like to express our gratitude and sincere appreciation to our Highly respected and esteemed guide **Mr. Hiranya Pd. Bastakoti**, for his valuable guidance, encouragement, support throughout the development of this project. His expertise constructive feedback and co-operative behavior were instrumental in shaping the project's direction and ensuring its success.

We would also like to express our deep gratitude to **Mr. Surya Bam**, Coordinator of Asian School of Management and Technology and **Mr. Chakra Narayan Rawal**, faculty member for their inspiration and guidance, unwavering support and motivation. Their mentorship has helped us to navigate the challenges faced during the project development.

Once again, we would like to thank our principal **Er. Anil Lal Amatya** and all faculty members and all our colleagues and other members who helped us directly and indirectly to pursue this project work. Your contributions have been essential to completing this project and we are truly grateful for all of that you have done.

With respect,

Bimala Gadtaula

Dinesh Lohani

Soniya Paudel

May 2023

ABSTRACT

Millions of people worldwide are impacted by diabetes, which is a common chronic disease. Early detection and intervention can significantly improve the quality of life for those at risk. The goal is to identify individuals at high risk of developing diabetes in order to prevent or delay its onset and improve health outcomes.

Major causes of diabetes include age, obesity, lack of physical activity, hereditary factors, unhealthy lifestyle choices, poor diet, and high blood pressure. One can leverage big data analytics to study massive datasets, uncover obscured information and patterns, and gain knowledge from the data, leading to outcome predictions. In existing method potential for bias and their reliance on a limited set of risk factors may not fully capture the complex interplay of genetic, environmental, and lifestyle factors that contribute to the development of diabetes. In this paper of work, we have proposed a diabetes prediction model for better classification, personalized care, public health impact which includes the external factors responsible for diabetes along with regular factors like Glucose, BMI, Age, Insulin, Pregnancy, Diabetes predict function, Skin thickness etc. Additionally, a "Logistic Regression" and “Linear Regression” model was implemented for predicting diabetes, aimed at enhancing the precision of classification.

Diabetes prediction is an integral part of diabetes prevention and management. The prediction system presented in this study utilizes logistic regression for binary classification and linear regression for continuous numerical estimation. By leveraging these two algorithms, the system provides valuable insights into the likelihood of developing diabetes and aids in monitoring different input features. The integration of this system into clinical practice has the potential to enhance early diagnosis, facilitate personalized treatment plans, and ultimately improve patient outcomes in the management of diabetes.

Table of Contents

LIST OF ABBREVIATIONS

BMI	Body Mass Index
CSS	Cascading Style Sheet
CSV	Comma Separated Value
HTML	Hyper Text Markup Language
JS	Java Script
ML	Machine Learning
RAM	Random Access Memory
UML	Unified Modeling Language

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

1.1 Introduction

Diabetes can be classified into two categories such as type 1 diabetes and type 2 diabetes. Type 1 diabetes is an autoimmune disease. In this case, the body destroys the cells that are essential to produce insulin to absorb the sugar to produce energy. Type 2 diabetes usually affects adults who are obese. In this type, the body resists observing insulin or fails to produce insulin. Type 2 generally occurs in the middle or aged groups [1].

When the amount of glucose in the blood increases, e.g., after a meal, it triggers the release of the hormone insulin from the pancreas. Insulin stimulates muscle and fat cells to remove glucose from the blood and stimulates the liver to metabolize glucose, causing the blood sugar level to decrease to a normal level [2]. Diabetes is the fast-growing disease among the people even among the youngsters. Diabetes is caused by the increasing level of sugar (glucose) in the blood. The prediction of the diabetes system means that it recognizes diabetes in a particular person without the help of a doctor. This system helps us to predict diabetes if we have the test result of several parameters. In this system, we take various parameters for a prediction like Pregnancy, Glucose level, Blood Pressure, Skin Thickness, Insulin, BMI, Diabetes Pedigree Function, Age. We used Logistic Regression to predict diabetes and Linear Regression to models the relationship between independent variables and assumes linearity and minimize the difference between observed and predicted values with a best-fit line. According to these parameters, the Logistic Regression gives results 0 or 1. Here O means have no diabetes, and 1 means have diabetes. While Linear regression predict certain numerical outcomes associated with diabetes This project works on the principle that no one should be deprived of the health service because they do not have sufficient information about their illness.

In the context of Nepal, a lot of people are deprived of the health services because of the limited doctors and hospitals in proportion to the population, low economic standard making it infeasible for people to visit the doctor every time. So, this project envisions Nepal to receive diabetes prediction service without visiting a doctor without money.

1.2 Problem Statement

Diabetes refers to a group of metabolic disorders that are amongst the most prevalent diseases worldwide. Despite the availability of various risk assessment tools and guidelines, diabetes continues to be a major public health concern with significant economic and health burdens. Nowadays, Healthcare industries generate large volumes of data. Current approaches for diabetes prediction have limitations in accurately identifying high-risk individuals and tailoring prevention strategies to individual patient characteristics. In the context of Nepal, for some regions there is a lack of good hospitals and equipment and availability of doctors who find it harder to visit health care centers for all health problems.

There is a need for a more accurate and efficient diabetes prediction system that can leverage advanced analytics and machine learning techniques, such as logistic regression, to improve prediction accuracy and inform targeted interventions for diabetes prevention and management.

1.3 Objectives

Diabetes is the fast-growing disease among all the people, so it is important to give all facilities to the patient. So, the central aim of this project is to make health-related knowledge and information reach every individual in a real and smartest way. The major objectives of the project are listed as follows:

- To compare the performance of Logistics Regression and Linear Regression algorithms for diabetes prediction system.
- To enhance diabetes management with real-time feedback and reminders.

1.4 Scope

The primary objective of this system is to aid healthcare professionals in recognizing individuals who are at risk of developing diabetes, enabling them to take proactive measures and provide the necessary treatment. Additionally, the system can also raise awareness among individuals about their risk of developing diabetes and motivate them to take preventive measures and make positive lifestyle changes. The system can be passed from one user to another making more people accessible and aware about their health condition. With the ability to be easily shared between users, the diabetes prediction system has the potential to make a significant impact on public health and improve the lives of people living with diabetes.

1.5 Limitation

There is certain limitation on working of the system which are listed below:

- Not all individuals in need of this service are aware of the user of computer systems or mobile and web applications.
- The project requires a computer machine with internet to operate on.
- Before conducting tests such as Glucose, Insulin, and BMI, this system must be utilized.

1.6 Development Methodology

Agile is a term used to describe software development approaches that employ continual planning, learning, improvement, team collaboration, evolutionary development, and early delivery. It encourages flexible responses to change. Agile is a term used to describe software development approaches that employ continual planning, learning, improvement, team collaboration, evolutionary development, and early delivery. It encourages flexible responses to change.

Agile methodology can be adapted to develop a diabetes prediction system by dividing the development process into several phases, as described below:

- Planning: In this phase, the team defines the requirements of the diabetes prediction system and identifies the data sources, algorithms, and output format that will be used.
- Design: Once the requirements are defined, the team moves on to the design phase. In this phase, the team designs the overall architecture of the system, including the data flow and algorithmic components.
- Development: The team works on developing the software incrementally, focusing on specific features and user stories in each iteration.
- Testing: In the testing phase, the team tests the implementation to ensure that it meets the requirements of the system. The testing should be done at every iteration to identify and fix any issues early on.
- Deployment: Once the testing is complete, the team deploys the system in a controlled environment. This phase involves integration, testing, and deployment of the system into a production environment.

- Maintenance: The final phase of the agile development process is maintenance. In this phase, the team continues to monitor and maintain the system, making any necessary adjustments to ensure it continues to function effectively.

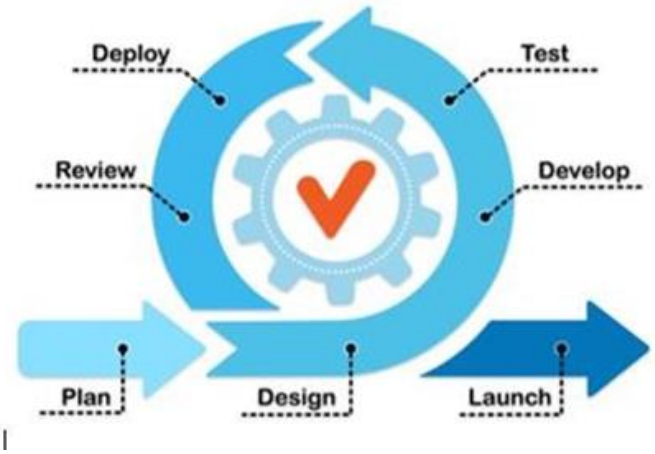

Fig 1.6: Agile Methodology [src: media.istockphoto.com]

1.7 Report Organization

The Report Organization includes the contents about how the study is being organized and carried out. The report is being divided into six chapters and the headings of the chapter includes:

- **Chapter 1 Introduction**: In this section, the main points discussed are about the overview, the background of the project, includes problems on the existing system, objective of the project and scopes and limitations of the project.
- **Chapter 2 Literature Review:** This chapter includes Literature Review related to the project collected from different research papers and data collection information.
- **Chapter 3 Requirement Analysis and Feasibility Study:** Study of functional, nonfunctional requirements and feasibility analysis are used to study the requirement needed for the system and the feasibility of the system to work properly.
- **Chapter 4 System Design:** includes design of the system including system architecture, database design, interface design, Algorithm used for designing system.

- **Chapter 5 Implementation and Testing:** contains implementation of the system using different tools and finally testing of the system as a unit and as a whole.
- **Chapter 6 Discussion and Conclusion:** The project is concluded with the result of the proposed method that has been analyzed and future enhancements are made according to the results obtained from the analysis.

CHAPTER 2

Background Study and Literature Review

2.1 Background Study

Type 1 diabetes is a condition where the pancreas is damaged and cannot produce insulin, typically affecting individuals under 20 years old [2]. Patients with this type of diabetes experience weight loss. However, it can be difficult to distinguish between Type 1 and Type 2 diabetes [3].

As well as Type 1, Type 2 Diabetes patients also face challenges in producing adequate insulin for their body. Although their pancreas produces insulin, it is insufficient as their body develops a resistance towards it. This type of diabetes is prevalent among the majority of diabetes patients [4]. In normal individuals, the sugar level in the blood should not fall below or rise above the normal level, which is between 4.4 to 6.1 mmol/L [1].

Diabetes is considered one of the most prevalent diseases worldwide, and it cannot be cured entirely, only managed through medication. Controlling its impact becomes more challenging once the effects have become severe. Unfortunately, most people only become aware of their diabetes after their symptoms have worsened. Therefore, early detection is crucial to raise awareness among the public. Three types of diabetes have been identified: Type 1, Type 2, and Gestational Diabetes, each with its unique characteristics. Also, the majority patient who has Diabetes is female rather than male [2].

Pregnant women are highly victims of Gestational Diabetes. During pregnancy, the pregnant women are advised to do a few tests to check if they have this kind of diabetes or not. If the person has this diabetes, the production of insulin cannot be produced as usual as before pregnancy and risk for the baby to suffer from diabetes also becomes higher. For information, usually the high weight baby may be delivered by the Gestational Diabetes mother. Next, for the next pregnancy, the patient has a high risk of getting the same problem. The bad effect on pregnant women who have this diabetes is bleeding during birth or miscarriage may occur [2].

2.2 Literature Review

2.2.1 Diabetes prediction system using Decision Tree

Diabetes prediction systems are very useful in the healthcare field. Classification approach based on Decision Tree (DT) helps to assign each data sample to its appropriate classes. By experiment, the proposed system achieved high classification results i.e. 98.7% compared to the existing system using Pima Indian Diabetes Dataset [4].

2.2.2 Early diabetes prediction system using ANN

Diabetes is a common, chronic disease. Prediction of diabetes at an early stage can lead to improved treatment. Data mining algorithms are widely used for prediction of disease at an early stage. Various tools are used to determine significant attribute selection, and for clustering, prediction, and association rule mining for diabetes. The ANN technique provided a best accuracy of 75.7% which may be useful to assist medical professionals with treatment decisions [5].

2.2.3 Diabetes prediction system using DNN

The deaths by diabetes are increasing each year so the need of developing a system that can effectively diagnose the diabetes patient becomes inevitable. Deep Neural Networks (DNN) are state-of-the-art in computer vision, language processing and image analysis and when applied in healthcare for prediction and diagnosis purposes these algorithms can produce highly accurate results. The results obtained from the algorithm used to build the system based on DNN technique show promising performance, with an accuracy rate of 99.75% and an F1 score of 99.6%. These results suggest that the system is capable of accurately classifying data and could have significant value in its intended application. This improvement can reduce time, efforts and labor in healthcare services as well as increasing the final decision accuracy [6].

2.2.4 Study of Existing System

There are several systems available that offer various compiled data prediction capabilities. These systems are generally more generic and not specific to any particular structure, which means they may not provide detailed information on specific systems or diseases. In contrast, our system is specifically designed for diabetes and offers comprehensive information. Therefore, our system is

both generic and specific, with a targeted focus on collecting all possible parameters related to diabetes.

2.3 Logistic Regression

Logistic regression is a classification model in machine learning, extensively used in clinical analysis. It uses probabilistic estimates which helps in understanding the relationship between the dependent variable and one or more independent variables. We opted to use this algorithm for our project because it is simple to implement and offers superior execution speed and performance.

2.4 Linear Regression

Linear regression is a statistical method used for predicting diabetes in a diabetes prediction system. It models the relationship between independent variables (such as age, BMI, blood pressure) and the dependent variable (diabetes status) using a linear equation. By analyzing the coefficients of the regression equation, the system can estimate the likelihood of an individual having diabetes based on their input data.

CHAPTER 3

System Analysis

3.1 System Analysis

System analysis includes evaluation of a system to achieve certain objectives during the development phase. Including analyzing its functional requirement which is an overview of how a system operates or should behave in any particular situation along with non-functional requirements regarding its usability, availability as well reliability. The feasibility study of any system is mainly intended to study and analyze the proposed system and to decide whether the system under consideration will be viable or not after implementation. That is, determines the usability of the project after deployment. To come to the result a set of queries is answered keeping the efficiency of the app and its impact on the domain for which it was developed.

The system should be able to predict diabetes based on the diagnostic measurements. This requires the use of machine learning algorithms. The system should provide output that is easy to predict diabetes. The system should be affordable and cost-effective to implement and maintain. The system should be able to handle a large amount of data.

3.1.1 Requirement Analysis

3.1.1.1 Functional Requirement

This section gives functional requirements that are applicable to our project Diabetes Prediction System. The input that are required during the construction of the system are:

Diabetes Prediction: Diabetes prediction is the process of using data and machine learning algorithms to predict the likelihood of a person developing diabetes.

Data Set: Common datasets used for diabetes prediction systems include medical records, demographic data, lifestyle data, and genetic data.

Trained Model: Commonly used trained models for diabetes prediction systems include logistic regression, decision trees, Naive Bayes', and artificial neural networks.

Testing Model: Commonly used testing models for diabetes prediction systems include accuracy, precision, recall, and F1 score.

Input Data: Common input data for diabetes prediction systems include age, gender, body mass index, family history, lifestyle habits, and blood test results.

3.1.2 Use Case Diagram

The use case diagram of our system shows the user and model admin as an actor according to the system. The user can input the required data and view the result (check diabetes) as model admin can data preprocessing and use the ML algorithm.

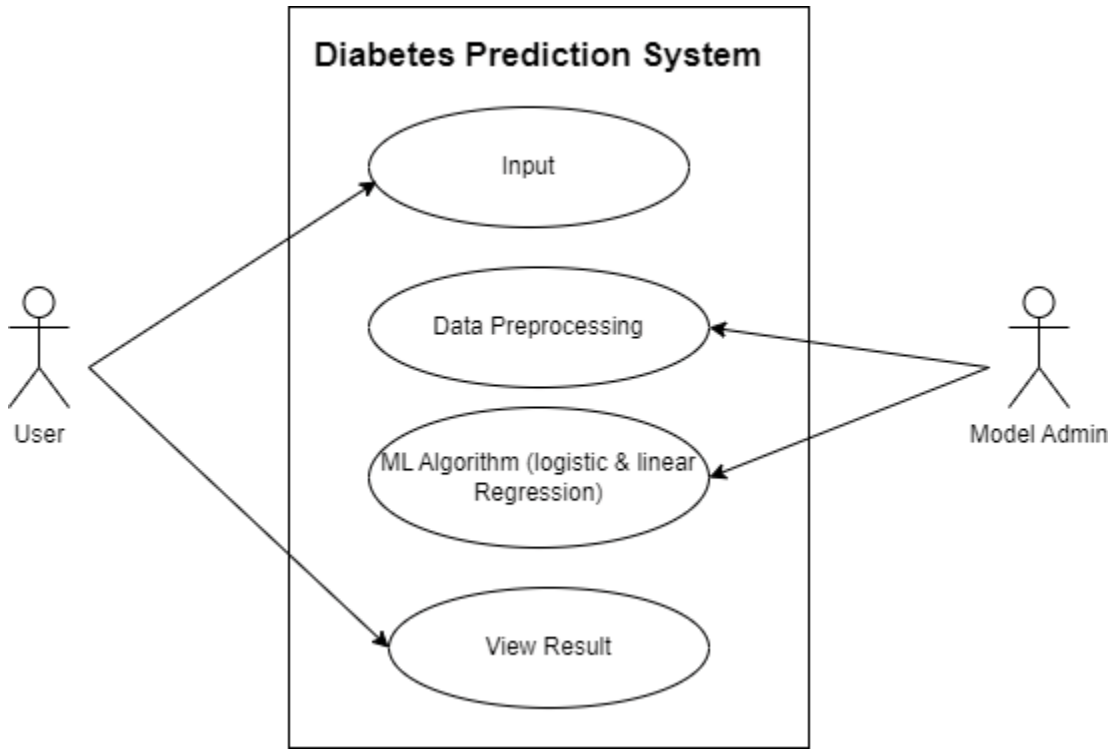

Fig 3.1.2: Use case Diagram

3.1.1.2 Non-Functional Requirement

Non-functional requirements are criteria used to evaluate a system's performance, rather than specific behaviors. They include any requirements beyond the system's core functionality that define how it should operate. These requirements are essential for ensuring the system's overall performance, usability, security, and maintainability. Examples of non-functional requirements for a diabetes prediction system include reliability, performance, security, usability, scalability, maintainability, compatibility, and availability.

Non-functional requirements of the project are as follows:

Performance: The system should be able to predict diabetes with a high level of accuracy, in a reasonable amount of time.

Scalability: The system should be able to handle a large number of users and data points without significant degradation in performance.

Usability: The system should be easy to use and understand for people with a range of technical expertise.

Reliability: The system should be able to function consistently and accurately over time, with minimal downtime or errors.

Maintainability: The system should be easy to maintain and update, with clear documentation and processes for making changes.

3.1.2. Feasibility Analysis

Before initiating a project, research must be conducted to determine the feasibility of the proposed system. It is important to thoroughly evaluate the project's technical, economical, operational, social, and legal aspects prior to beginning the project. This ensures that the project is viable and provides detailed information to determine whether or not to proceed with the project.

3.1.2.1 Technical feasibility

The technical issue usually raised during the feasibility stage of the investigation includes the following.

The current application developed is technically feasible. It is a web-based application. Thus, it provides an easy implementation. It provides the technical guarantee of accuracy, reliability. The work for the project is done with the current equipment and existing software technology. The project will be developed in python implementing the machine learning algorithm. In order to complete the system further study of the python programming language and the machine learning algorithm is required. The system is relatively small and will not require highly skilled technical personnel. It is feasible to develop the system.

Hardware Requirements:

The minimum hardware requirement for developed system and its operations include the following:

- RAM: 4GB
- Processor: intel Pentium 4
- Hard disk: 500GB

Software Requirements:

The minimum Software requirement for developed system and its operations include the following:

- Operating System: Microsoft Windows 7, 8, 8.1, 10, 11
- Software: Django Framework, Jupyter Notebook or Spyder, Command line, PyCharm etc.
- Language: HTML, CSS, Python, Machine Learning etc.

The above hardware and software are easily available and can be used by anyone. Hardware used is very minimal and hence computers can run our system. Also, software is also easily available and easy to use. So, analyzing these hardware and software used it is sure that this system is technically feasible.

3.1.2.2 Operational feasibility

Operational feasibility aspects of the project are to be taken as an important part of the project implementation. Proposed projects are beneficial only if they can be turned into real world implemented System. That will meet the user's requirements. Some of the important issues raised are to test the operational feasibility of a project includes the following: -

This system is targeted to be in accordance with the above-mentioned issues. Beforehand, the user requirements have been taken into consideration. So, there is no question of resistance from the users that can undermine the possible application benefits. The well-planned design would ensure the optimal utilization of the computer resources and would help in the improvement of performance status.

3.1.2.3 Economic feasibility

Economic feasibility determines if a system is worth the cost and time we spent on it: Our system is economically feasible because it is easy to use with minimum cost. The only cost related to this system will be for hosting space. It doesn't require further hardware and other resources. Science the project is developing using Windows 10 and Windows 11, Atom text editor, Jupyter which are

easily available on the Internet that doesn't require the cost for installation and use. Hence the project is economically feasible.

3.1.2.4 Schedule Feasibility

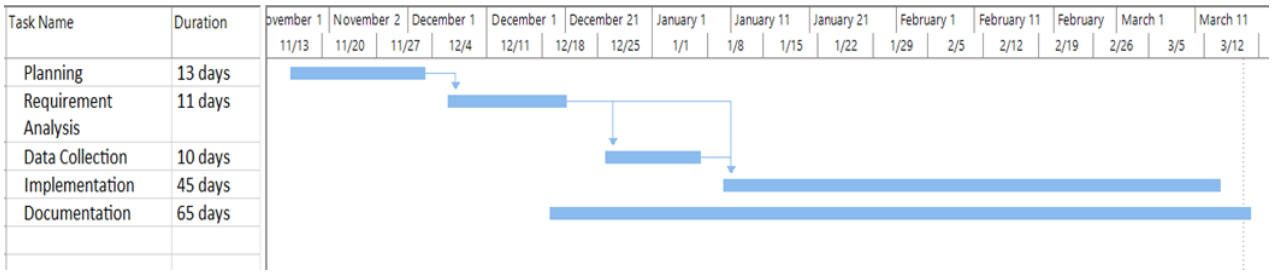

Figure 3.1.2.4: Gantt chart

The above Gantt chart displays the overall timeline of the project. It presents the sequential breakdown of the task involved in the project with the time taken for each task. The Research and Analysis Phase was carried out in parallel with Design. After that coding was done followed by testing and deployment in their mentioned time. And finally, documentation was done and the final report was prepared.

3.3 Object Oriented Analysis

System is based on an object-oriented approach which focuses on modeling data using class and object diagrams, state and sequence and process modDeling using activity diagram.

3.3.1 Object modeling using class and object diagram

A class diagram represents the classes, interfaces, attributes, and methods of a system and their relationships. It is a static representation of the system's structure. Classes represent the blueprint of objects, and they define the properties and behaviors that objects of that class can have.

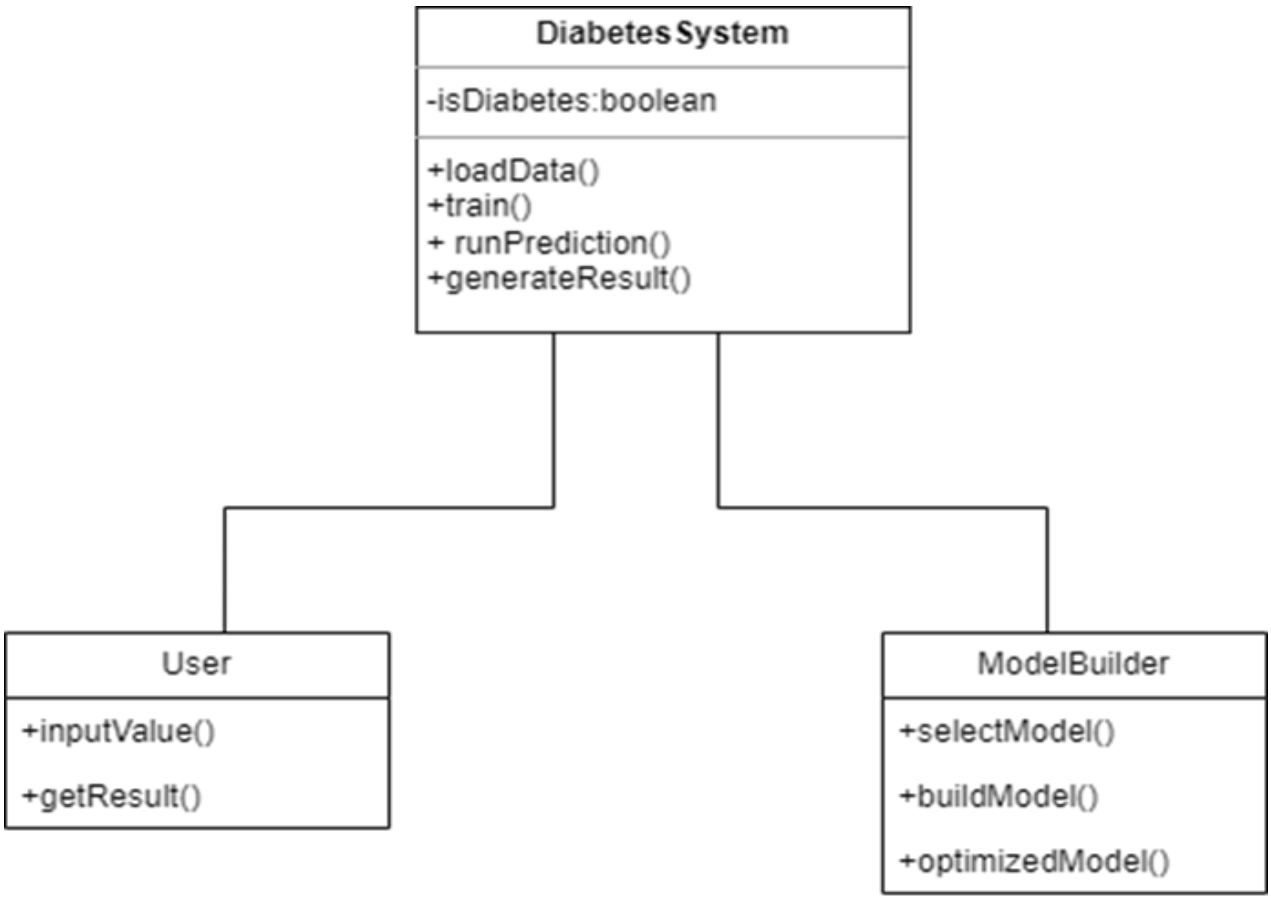

Fig 3.3.1: Class diagram

3.3.2 Dynamic modeling using state and Sequence Diagram

A sequence diagram is a type of UML diagram that illustrates the interactions between objects or components in a system over time. It shows the sequence of events or messages passed between different objects or components, and can be used to model complex systems and processes. The basic elements of a sequence diagram include Lifelines, Messages, Activation Bars and Objects.

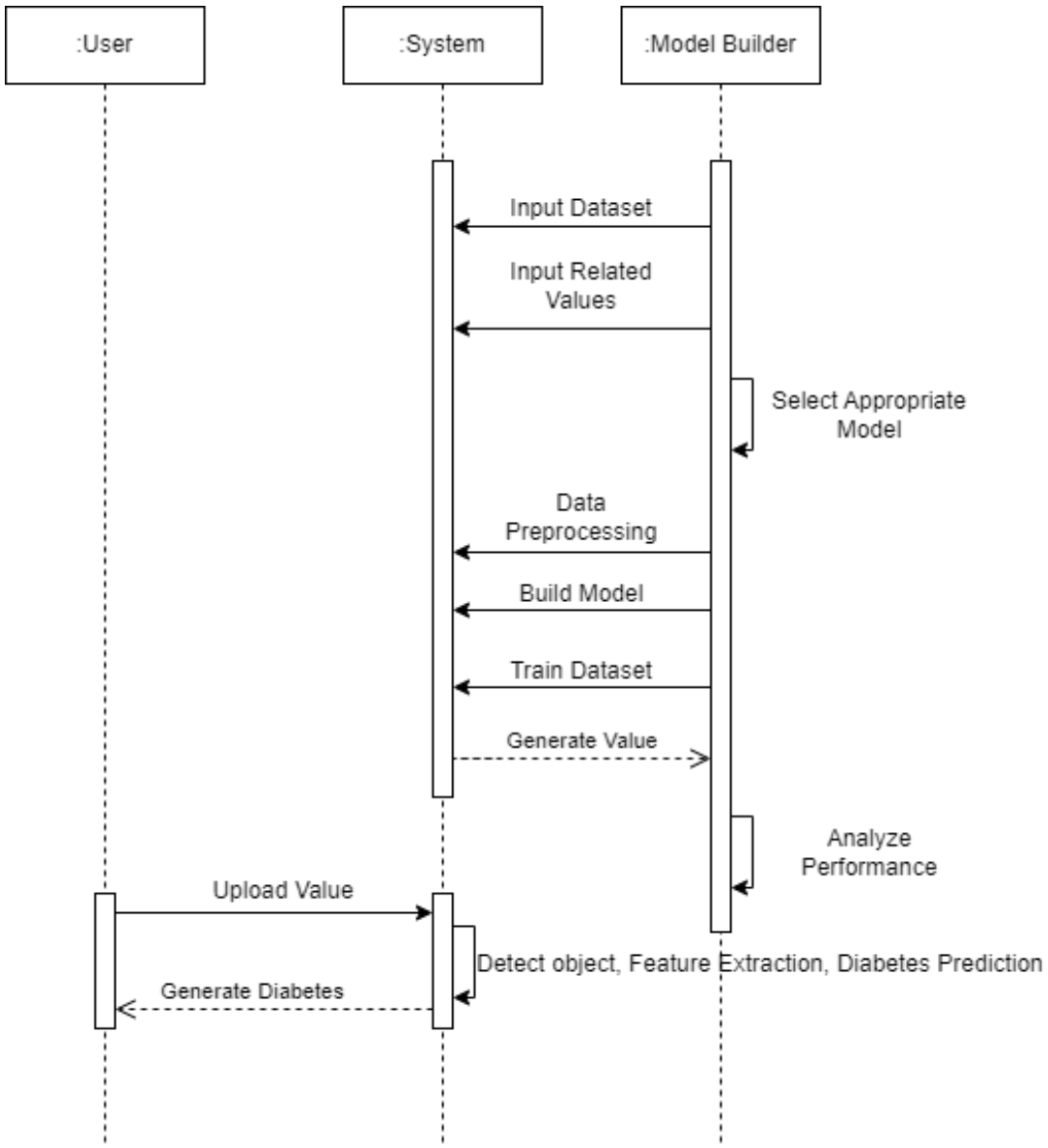

Fig 3.3.2: Sequence diagram

3.3.3 Process modeling using Activity Diagram

An activity diagram is a type of UML diagram used to model workflows or processes. It is a visual representation of a sequence of actions or activities performed by a system.

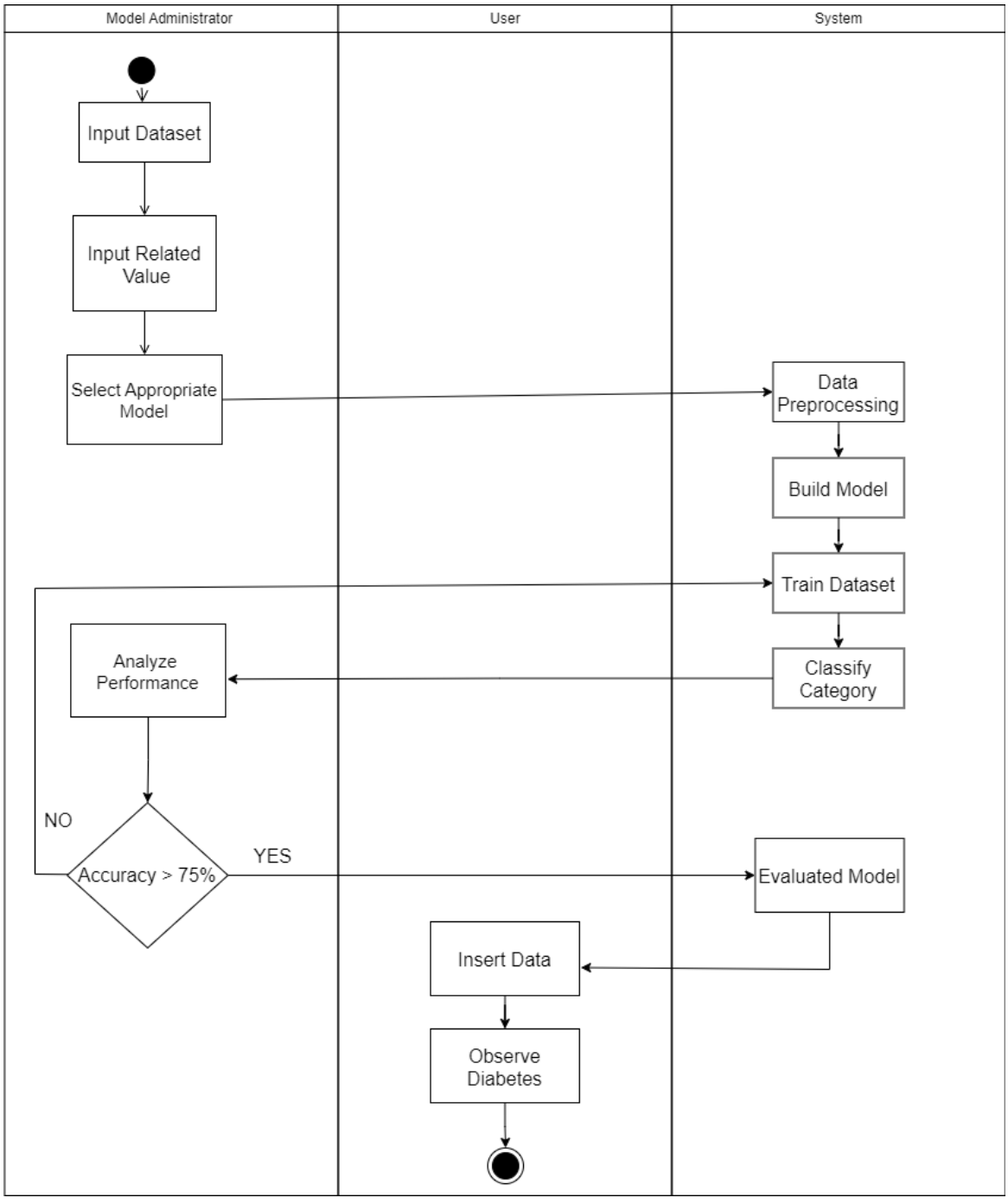

Fig 3.3.3: Activity diagram

CHAPTER 4

SYSTEM DESIGN

4.1 System Design

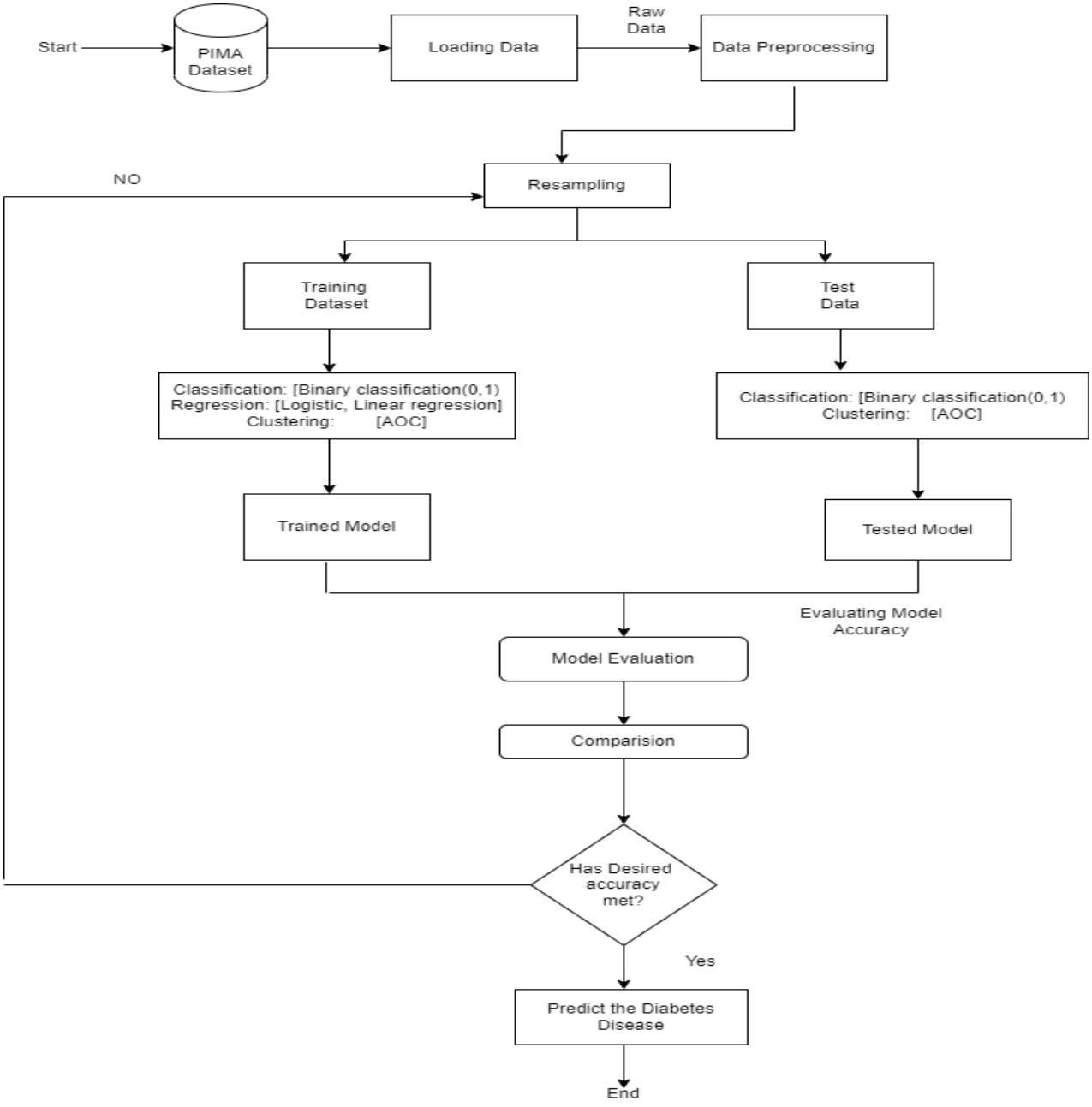

Fig 4.1: Flowchart

4.1.1 Dataset Design

Dataset plays an important role in making sure the data and information in the system display proper information as intended. Dataset contain the information of the past inserted data. These data are used for testing and training and finally these data are used in prediction.

The dataset used in the system is saved as .CSV file. The data used in the system is originally from the National Institute of Diabetes and Kidney Disease. The PIMA Indian diabetes dataset donated by Vincent Sigil to, is a collection of medical diagnostic reports of women from age 21 onwards living in Arizona, USA. In the dataset O means negative of diabetes and 1 means positive of diabetes. Out of 768 instances there are 500 cases of no diabetes and 268 cases of diabetes. The dataset is stored as a CSV file so it can be used by using the import function. There are no missing values in the data, but it needs to be cleaned for duplicate and missing data.

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1
...	...	...	...	...	...	...	...	...	...
763	10	101	76	48	180	32.9	0.171	63	0
764	2	122	70	27	0	36.8	0.340	27	0
765	5	121	72	23	112	26.2	0.245	30	0
766	1	126	60	0	0	30.1	0.349	47	1
767	1	93	70	31	0	30.4	0.315	23	0

Table 4.1.1: Dataset Used

4.1.2 Interface and Dialogue Design

The web application features a user-friendly interface that comprises a homepage and a prediction page. On the prediction page, users are prompted to input various parameters requested by the system. When the user provides correct information, the system generates accurate predictions.

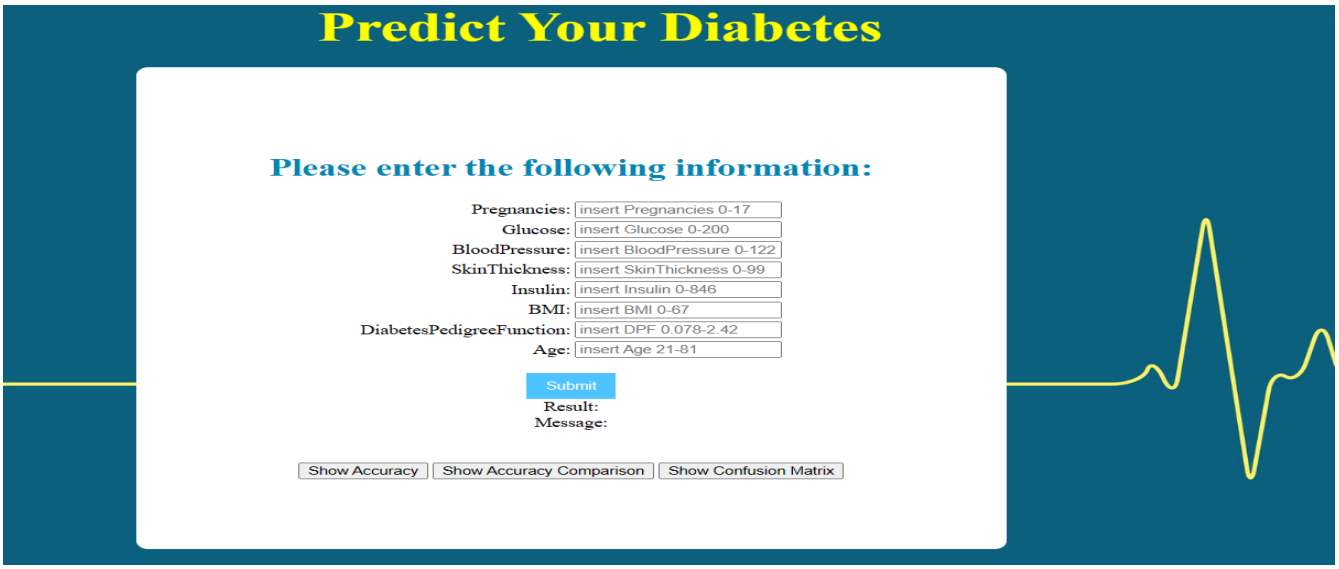

Figure 4.1.2: Interface User Input Page

4.1.3 Component Diagram

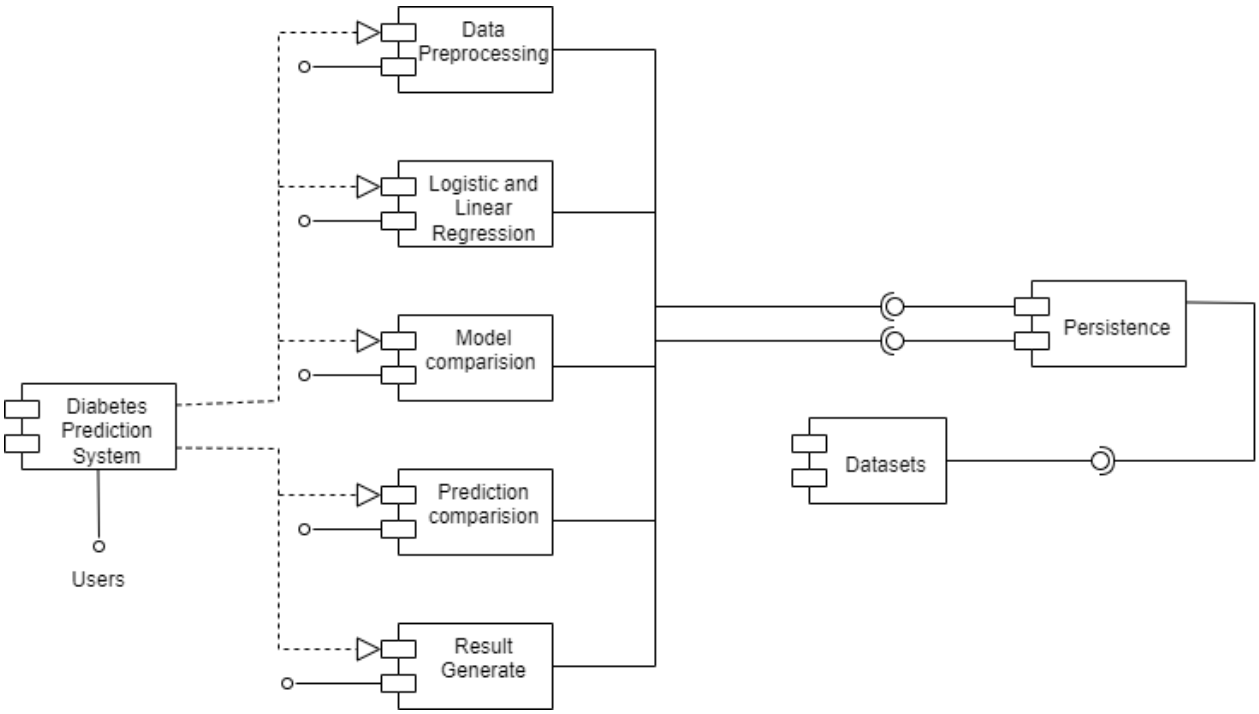

Fig:4.1.3: Component Diagram

The above diagram provides a visual representation of the major building blocks of the system and how they interact with each other to fulfil the system's functionalities.

4.1.4 Deployment Diagram

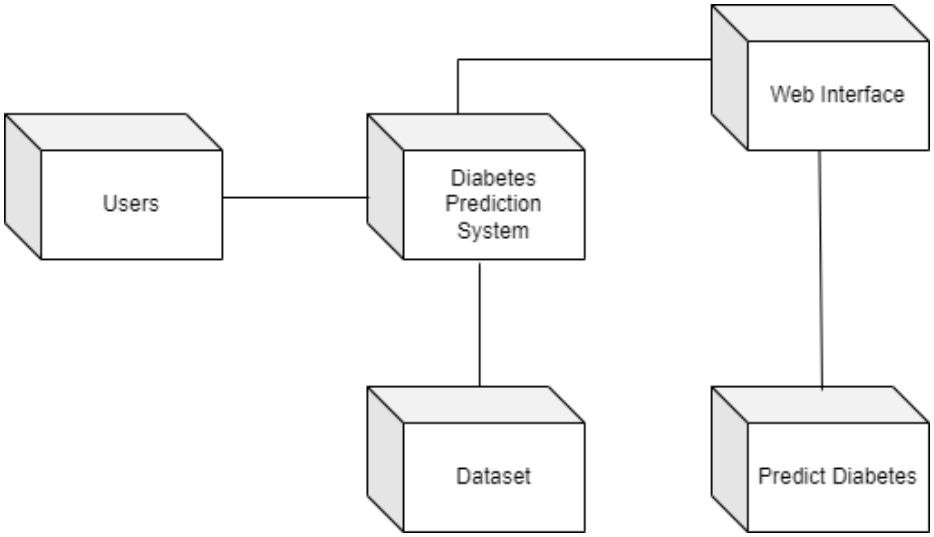

Fig 4.1.4: Deployment Diagram

The above diagram provides a visual representation of how software components are deployed and distributed. They help understand the overall system architecture, including the relationship between software components and the hardware infrastructure.

4.2 Algorithms Details

4.2.1. Logistic Regression Algorithm

Logistic regression is a popular classification algorithm in Machine Learning that uses dependent variables to predict the probability of certain classes. The algorithm calculates a sum of the input features, including a bias term, and applies a logistic function to obtain the output. The result of logistic regression is always between 0 and 1, making it suitable for binary classification tasks. A higher output value indicates a higher probability that the sample belongs to class 1.

As per Logistic Regression is a classification method based on Linear Regression. The classification goal is to predict whether or not the patients in the dataset have diabetes or not. While using machine learning, we usually have two sets of data. (i) training data and (ii) testing data. The training dataset contain of the input data together with correct expected output. The testing dataset is a subset to test the trained model.

The model learns on this data in order to be generalized to other data later on. First, we will train our dataset in logistic regression model on (X_train, y_train) and we will use (X_test, y_test)to

evaluate the model generated. We will build the Logistic Regression model and predict for X_test and compare prediction to the y_test.The formula for logistic regression algorithm for diabetes prediction is as follows:

First, we calculate the weighted sum of the input variables (also known as the logit or linear predictor):

z = b0 + b1x1 + b2x2 + ... + bnxn

where:

z is the weighted sum of the input variables

b0 is the intercept (also known as the bias)

b1, b2, ..., bn are the coefficients (also known as the weights)

x1, x2, ..., xn are the input variables

Next, we apply the logistic function (also known as the sigmoid function) to the weighted sum to obtain the predicted probability of the patient being diabetic:

p = 1 / (1 + e^(-z))

where:

p is the predicted probability of the patient being diabetic

e is the mathematical constant e (approximately 2.71828)

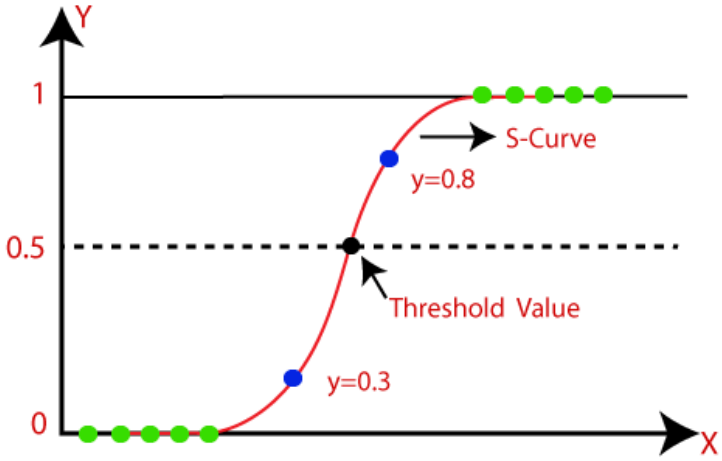

Fig 4.2.1: Logistic Regression in Machine Learning

(Src: Javatpoint)

A hyperplane is used as a decision line to separate two categories (as far as possible) after data points have been assigned to a class using the Sigmoid function. The class of future data points can then be predicted using the decision boundary.

4.2.2. Linear Regression Algorithm

Linear regression is a statistical method used for predicting diabetes in a diabetes prediction system. It models the relationship between independent variables (such as age, BMI, blood pressure) and the dependent variable (diabetes status) using a linear equation. By analyzing the coefficients of the regression equation, the system can estimate the likelihood of an individual having diabetes based on their input data.

$$y = mx + c$$

where,

y : Is a variable to be predicted (ie. Dependent variable) . It is of numerical continuous data-type.

m : here the coefficient ‘m’ is nothing but the slope of the line.

x : Is the variable which is called the independent variable.

c : We know this as a constant value , ie. y-intercept.(The value of ‘y’ when ‘x’ is zero. Basically means it is the point at which crosses the vertical axis.

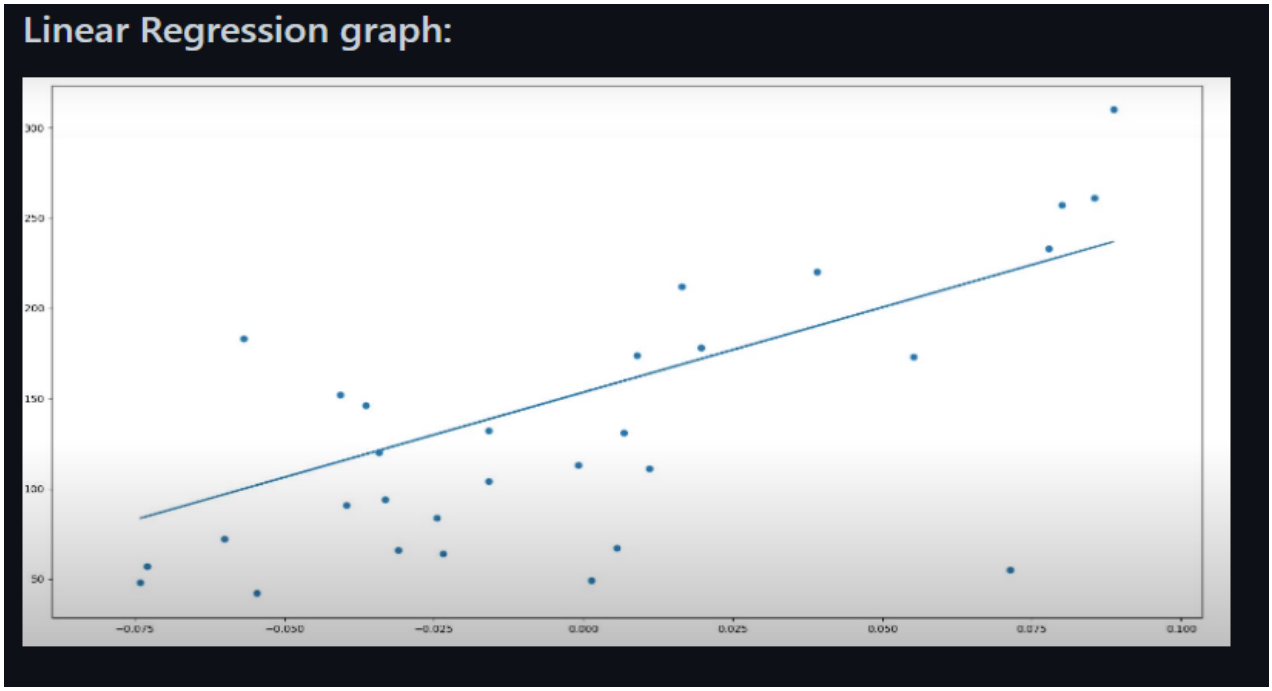

Fig 4.2.2 Linear Regression graph

CHAPTER 5

IMPLEMENTATION

5.1 Implementation

For the implementation of system different frontend, backend technologies and databases were used. The user is asked for input in through a form and behind the scenes the dataset from database is loaded, it is then preprocessed, then it is splinted into testing and training data and then the machine learning algorithm predicts whether the patient is diabetic or not.

5.1.1 Tools Used

1. Data and Information

Dataset plays an important role in making sure the data and information in the system display proper information as intended. Dataset contains the information of the past inserted data. These data are used for testing and training and finally these data are used in prediction. The dataset used in the system is saved as a .CSV file.

Data is collected from the National Institute of Diabetes and Digestive and Kidney Disease. It is also known as PIMA Indian dataset which was donated by Vincent Sigilito, a collection of medical diagnostics of women from age 21 onwards living in Arizona, US. The Pima population has been under study since 1965, with data collected at regular intervals of 2 years. Since epidemiological evidence indicates that Type 2 Diabetes Mellitus (T2DM) results from a combination of genetic and environmental factors, the dataset contains information about attributes that could be associated with the onset of diabetes and its future complications. The dataset comprises eight features and a class variable that serves as the label for each training instance. The features include the number of times pregnant, plasma glucose concentration, diabetes pedigree function, triceps skin fold thickness (mm), diastolic blood pressure (mmHg), 2-hour serum insulin (mU/mL), body mass index (kg/m^2), and age in years. The class variable assumes values of 0 and 1, where 0 indicates a healthy person and 1 indicates a person with diabetes.

2. Programming Language

Table 5.1.1: Development Tools used

Frontend Tools Used	HTML, CSS, Bootstrap, JS
Backend Tools Used	Python
IDE	

5.1.2 Implementation detail of Modules

Generating the final prediction is a multi-step process, which involves the system undergoing several stages before arriving at the desired output.

5.1.2.1 Preparing Training Data

Vincent Sigilito donated the PIMA Indian diabetes dataset, which comprises medical diagnostic reports of women aged 21 and above residing in Arizona, USA. The dataset classifies diabetes as either positive (1) or negative (0), and it contains 500 instances of no diabetes and 268 instances of diabetes out of a total of 768 instances. The dataset is stored in a CSV file format and can be imported for use. Although there are no missing values in the data, it requires cleaning to remove any duplicate or missing data.

5.1.2.2 Training Data

In the Training phase, The Logistic regression first draws an N-dimensional hypercube by representing each feature as a separate dimension. It then uses the numerical value of those features to plot points on the N-dimensional hypercube. Then it attempts to find boundary that separates the two classes of data. Points where outcome is 0 (no diabetes) and points where outcome is 1 (diabetes), for example. The boundary is a (N-1) dimension hyperplane.

5.1.2.3 Train/Test Split

To input data into the machine learning model, it is essential to partition the dataset into distinct training and testing subsets. The typical split is done using an 80-20 ratio, where 80% of the data is allocated to the training set and 20% is reserved for testing.

#Train Test Split

X_train, X_test, y_train, y_test = train_test_split (X, y, train_size=0.8, random_state=2)

5.1.2.4. Testing Data

During the testing phase, real-time patient data, including age, number of pregnancies, insulin level, BMI, and other relevant factors, can be inputted. The Logistic Regression algorithm evaluates this data and produces a 1 or 0 outcome indicating the presence or absence of diabetes, respectively, based on which side of the boundary the data falls on.

5.1.2.5 Applying Logistic Regression Algorithm

In the subsequent stage, the logistic regression model was constructed using the provided code. To build the model, specific features were selected for inclusion and passed alongside the target variable to the fit method of the logistic regression model. This allowed the model to learn the underlying patterns and relationships between the selected features and the outcomes. By fitting the logistic regression model to the data, it was trained to make accurate predictions and estimate the coefficients associated with each feature. The resulting model could then be utilized to predict the likelihood of diabetes based on the chosen features.

Code:

```
import numpy as np
class LogisticRegression:

def __init__(self, lr=0.01, num_iter=100000, fit_intercept=True, verbose=False):

self.lr = lr

self.num_iter = num_iter

self.fit_intercept = fit_intercept

self.verbose = verbose
```

```
def __add_intercept(self, X):

X = np.array(X)

intercept = np.ones((X.shape[0], 1))

return np.concatenate((intercept, X), axis=1)

def __sigmoid(self, z):

return 1 / (1 + np.exp(-z))

def __loss(self, h, y):

return (-y * np.log(h) - (1 - y) * np.log(1 - h)).mean()

def fit(self, X, y):

if self.fit_intercept:

X = self.__add_intercept(X)

X = np.array(X)

self.theta = np.zeros(X.shape[1])

for i in range(self.num_iter):

z = np.dot(X, self.theta)

h = self.__sigmoid(z)

gradient = np.dot(X.T, (h - y)) / y.size

self.theta -= self.lr * gradient

if self.verbose and i % 10000 == 0:

z = np.dot(X, self.theta)

h = self.__sigmoid(z)

print(f'loss: {self.__loss(h, y)} \t')

def predict_prob(self, X):
```

```
if self.fit_intercept:

X = self.__add_intercept(X)

return self.__sigmoid(np.dot(X, self.theta))

def predict(self, X, threshold=0.5):

return self.predict_prob(X) >= threshold
```

5.1.2.6 Applying Linear Regression Algorithm

Next, a linear regression model was built. The code involved selecting the desired features and providing them along with the target variable to the fit method of the linear regression model. This allowed the model to learn the relationships between the features and the target variable. By estimating the coefficients, the model could make predictions for continuous measures such as blood sugar levels. The trained linear regression model could then be applied to new data to predict the target variable based on the feature values.

Code:

```
import numpy as np
class LinearRegression:
def __init__(self, lr=0.01, num_iter=1000, fit_intercept=True, verbose=False):

self.lr = lr

self.num_iter = num_iter

self.fit_intercept = fit_intercept

self.verbose = verbose

def __add_intercept(self, X):

X = np.array(X)

intercept = np.ones((X.shape[0], 1))

return np.concatenate((intercept, X), axis=1)
```

```
def __cost(self, X, y, theta):

m = y.size

h = X.dot(theta)

J = (1 / (2 * m)) * np.sum(np.square(h - y))

return J

def fit(self, X, y):

if self.fit_intercept:

X = self.__add_intercept(X)

self.theta = np.zeros(X.shape[1])

for i in range(self.num_iter):

h = X.dot(self.theta)

gradient = X.T.dot(h - y) / y.size

self.theta -= self.lr * gradient

if self.verbose and i % 100 == 0:

print(f"Cost: {self.__cost(X, y, self.theta)}")

def predict(self, X):

if self.fit_intercept:

X = self.__add_intercept(X)

return X.dot(self.theta)
```

5.2 Result Analysis

A Diabetes prediction system is designed to identify individuals at risk of developing diabetes by analyzing various factors such as age, family history, lifestyle choices, and medical history. By

accessing data like BMI, blood pressure, cholesterol levels, and glucose levels, the system can estimate an individual's likelihood of developing diabetes. The system aims to achieve early identification, provide personalized interventions, improve health outcomes, and optimize resource allocation. It helps healthcare professionals intervene early, implement preventive measures, and allocate resources efficiently. However, it is important to note that it is a risk assessment tool, not a diagnostic tool, providing valuable information for informed decision-making and proactive healthcare management.

Evaluation Metrics:

In our project, we used a correlation matrix from the dataset to evaluate our diabetes prediction system. The correlation matrix helped us analyze the relationships between variables like age, BMI, glucose levels, and family history. By studying the correlation coefficients, we gained insights into the strength and direction of these relationships. This information allowed us to identify potential risk factors and predictors for diabetes. While the correlation matrix has limitations and doesn't establish causation, it serves as a valuable tool in understanding the interconnections within our prediction system.

TP	**TN**
74	**25**
15	**40**
FP	**FN**

Table 5.2.2: Confusion matrix

Four cases are considered as the result of the classifier:

True Positives (TP): This indicates the number of instances correctly classified as positive. In this case, there are 74 instances classified as positive that are actually positive.

True Negatives (TN): This shows the number of instances correctly classified as negative. Here, there are 25 instances classified as negative that are actually negative.

False Positives (FP): These are instances incorrectly classified as positive when they are actually negative. In this case, there are 15 instances that were mistakenly labeled as positive.

False Negatives (FN): This represents instances incorrectly classified as negative when they are actually positive. Here, there are 40 instances that were mistakenly labeled as negative.

Performance summary:

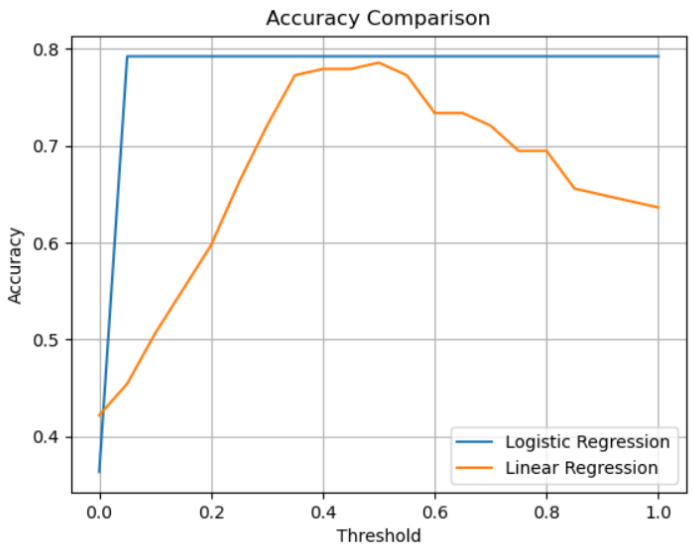

Fig 5.2.2**:** Accuracy Comparison Graph

To analyze the performance of the classification, the accuracy and AUC measures are adopted.

The accuracy graph illustrates the performance of logistic regression and linear regression in our diabetes prediction system, showing their respective accuracy across iterations or variations.

Logistic regression consistently outperformed linear regression in terms of accuracy in our experiments. It demonstrated a higher overall accuracy rate, effectively classifying individuals as diabetic or non-diabetic based on the provided features. In contrast, linear regression, although reasonably accurate, showed slightly lower performance compared to logistic regression.

The accuracy graph serves as a valuable visual representation of the performance of logistic regression and linear regression in our diabetes prediction system. It clearly showcases the superior accuracy achieved by logistic regression, emphasizing its effectiveness in accurately predicting diabetes.

Performance testing in assessing a logistic regression and linear Regression model's ability to predict diabetes involves metrics such as accuracy, sensitivity, specificity, precision, and F1 score. Sensitivity captures the percentage of true positives, accuracy measures overall correct predictions, specificity represents true negatives, precision focuses on true positives among positive predictions, and the F1 score balances precision and sensitivity. These metrics provide valuable insights into the model's effectiveness and enable comparisons with other models or benchmarks.

	precision	recall	f1-score	support
0	0.83	0.75	0.79	99
1	0.62	0.73	0.67	55
accuracy			0.74	154
macro avg	0.72	0.74	0.73	154
weighted avg	0.75	0.74	0.74	154

Table 5.2.3: Precision Recall Table

Result Interpretation:

Logistic Regression is a statistical modeling technique used to predict the probability of a binary outcome based on one or more independent variables. In our project to categorize the input features and outcome for the classification purpose we have the input features labeled as Pregnancies, Glucose, BMI, Blood Pressure, Skin Thickness, Age, Insulin, Diabetes Pedigree Function and Outcome in the logistic regression model that have statistically significant coefficients. Using logistic regression while interpreting the coefficients by considering the positive and negative magnitude indicates the direction of relationship between the input features and probability of diabetes. The magnitude of the coefficients provides information about the strength of relationship between the input features and the probability of diabetes. Input features that have statistically significant coefficient impact on the prediction of diabetes.

For instance, if the coefficient for the "glucose" variable is positive and statistically significant, it suggests the higher glucose levels are associated with an increased probability of diabetes. On the other hand, if the coefficient for "BMI" is negative and statistically significant, it implies that a higher BMI is associated with the decreased likelihood of diabetes.

Similarly. Linear regression is also a statistical modeling technique which assumes a linear relationship between the predictors and response variables. When interpreting the result of a linear regression with input features and an outcome variable we firstly extract the coefficients from the model. Interpret the coefficient by considering their magnitude as positive and negative significant. If the coefficient for the "Glucose" variable is positive and statistically significant, it suggests the higher glucose levels are associated with higher values of the outcome variable.

Comparison with Benchmark

In comparison to the existing benchmark, our project developed a diabetes prediction system using Logistic Regression and Linear Regression algorithms. The existing system utilized the Support Vector Machine (SVM) algorithm, which achieved an accuracy of 76 percent, along with sensitivity and specificity metrics.

In our project, we divided the dataset into input features such as age, glucose, BMI, insulin, and more, along with the target variable representing the diabetes outcome. Both Logistic Regression and Linear Regression models were trained on the data after feature scaling. Logistic Regression

estimated coefficients that indicate the impact of each feature on the log-odds of the target variable, while Linear Regression represented the change in the target variable for a unit change in the corresponding input feature.

Our Logistic Regression model achieved an accuracy of 78 percent, while Linear Regression achieved around 76 percent accuracy. Like the existing system, we also used hyperparameters in our models, and in SVM, they included the choice of kernel function, regularization parameter, and gamma parameter.

The trained Logistic Regression model provided the probability of an individual having diabetes, with a threshold applied to convert probabilities into binary predictions. This threshold can be adjusted based on the desired balance between sensitivity and specificity. The trained Linear Regression model predicted the target variable (e.g., blood sugar levels) for new individuals based on their input features, providing continuous estimated outcomes.

SVM, Logistic Regression, and Linear Regression each have their strengths. SVM handles linearly separable and non-linearly separable data through kernel functions, while Logistic Regression is interpretable and suitable for binary classification tasks. Linear Regression is appropriate for predicting continuous measures. The choice of algorithm depends on the data characteristics, prediction task, and desired interpretability versus predictive performance trade-off. Evaluation using appropriate metrics helps determine the most effective approach for a diabetes prediction system.

Robustness Analysis:

In a diabetes prediction system, it is crucial to conduct a robustness analysis of logistic regression and linear regression to evaluate their performance and reliability. Logistic regression is sensitive to outliers, violations of linearity, overfitting, and multicollinearity. Outliers can distort the relationship between features and the target variable, while violations of linearity assumptions can compromise the model's performance. Overfitting and multicollinearity can lead to less reliable results. Techniques such as outlier handling, regularization, and feature selection can enhance the robustness of logistic regression. Similarly, linear regression is also sensitive to outliers and violations of linearity assumptions. Employing robust regression techniques and addressing

multicollinearity can improve its robustness. Violations of the homoscedasticity assumption can also affect linear regression, which can be mitigated using methods like robust standard errors and weighted least squares regression. Overall, considering these factors and employing appropriate techniques enhances the robustness of both logistic regression and linear regression in a diabetes prediction system.

Performance of System

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Sensitivity = \frac{TP}{TP+FN} \qquad Specificity = \frac{TN}{FP+TN}$$

Dataset	Accuracy	Sensitivity	Specificity
Diabetes	76.67%	50%	88%

Table 5.3: Performance of the system

CHAPTER 6

CONCLUSION AND FUTURE RECOMMENDATION

6.1 Conclusion

The project successfully demonstrates the classification of diabetes cases using Logistic Regression and Linear Regression from a small dataset. Unlike transfer learning, the model was built from scratch, showing computational efficiency. The algorithms are utilized to achieve an accuracy of 78 % and 76% respectively.

Both models were trained on relevant input features to predict the diabetes outcome. Logistic regression provided interpretability, while linear regression focused on continuous measures.

The models were trained on key input features such as age, glucose, BMI, insulin, and more, enabling the prediction of diabetes outcomes. Logistic regression provided valuable interpretability, while linear regression focused on predicting continuous measures. Comparing the results with the existing benchmark, the accuracy levels were found to be similar. Robustness analysis highlighted the importance of addressing outliers and violations of linearity for reliable performance. Overall, the diabetes prediction system demonstrated promising results, offering valuable insights for informed decision-making and proactive healthcare management. This project paves the way for future enhancements, such as incorporating medication recommendations and expanding the system's usability to enable self-diagnosis and support from non-expert personnel.

6.2. Future Recommendation

Presently, this system is limited to identifying whether the patient has diabetes or not. This being our initial phase for the development of machine learning model and implementation in the health sector specially focused in rural areas. In the near future, the system will receive an upgrade with new features, including the ability to classify diabetes into type 1 and type 2 and offer medicine recommendations. Additionally, it could provide the user with information about nearby hospitals and health facilities. A fully functioning real-life system, based on a monthly subscription model, could be developed from these upgrades.

REFERENCE

[1] World Health Organization, 2021

https://www.who.int/news-room/fact-sheets/detail/diabetes

[2] National Institute of Diabetes and Kidney Diseases, 2021

https://www.niddk.nih.gov/health-information/diabetes

[3] Ayush Anand and Divya Shakti," Prediction of Diabetes Based on Personal Lifestyle Indicators", 1st International Conference on Next Generation Computing Technologies, 978-1-4673-6809-4, September 2015.

[4] Diabetes Prediction with Logistic Regression: Kaggle

https://www.kaggle.com/code/tanyildizderya/diabetes-prediction-with-logistic-regression

[5] B. Nithya and Dr. V. Ilango," Predictive Analytics in Healthcare Using Machine Learning Tools and Techniques", International Conference on Intelligent Computing and Control Systems, 978-1-5386-2745-7, 2017.

[6] Pandey, Dr. Subhash. (2016). Data Mining Techniques for Medical Data: A Review. 10.1109/SCOPES.2016.7955586.

[7] S. I. Ayon and M. Islam, "Diabetes prediction: A deep learning approach," International Journal of Information Engineering and Electronic Business, vol. 11, no. 2, pp. 21–27, 2019.

[8] Sisodia, Deepti & Sisodia, Dilip. (2018). Prediction of Diabetes using Classification Algorithms. Procedia Computer Science. 132.1578-1585. 10.1016/j.procs.2018.05.122.

[9] Dutta, D., Paul, D., & Ghosh, P. (2018, November). Analysing Feature Importances for Diabetes Prediction using Machine Learning. In 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON) (pp. 924-928). IEEE.

APPENDICES

Appendices A: Wire Frame

1. Main page

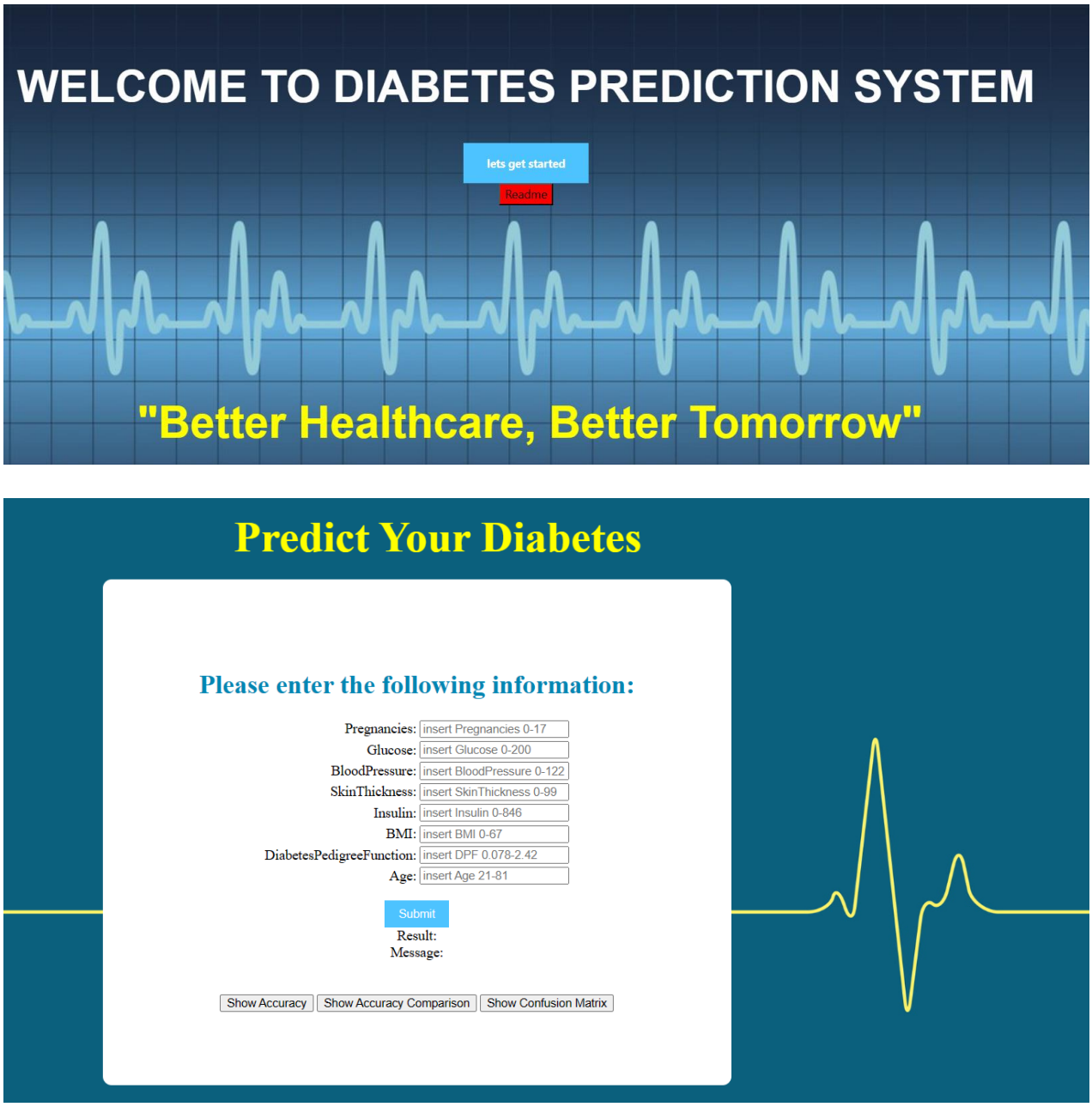

Fig A1: Main Page

2. Predict Page

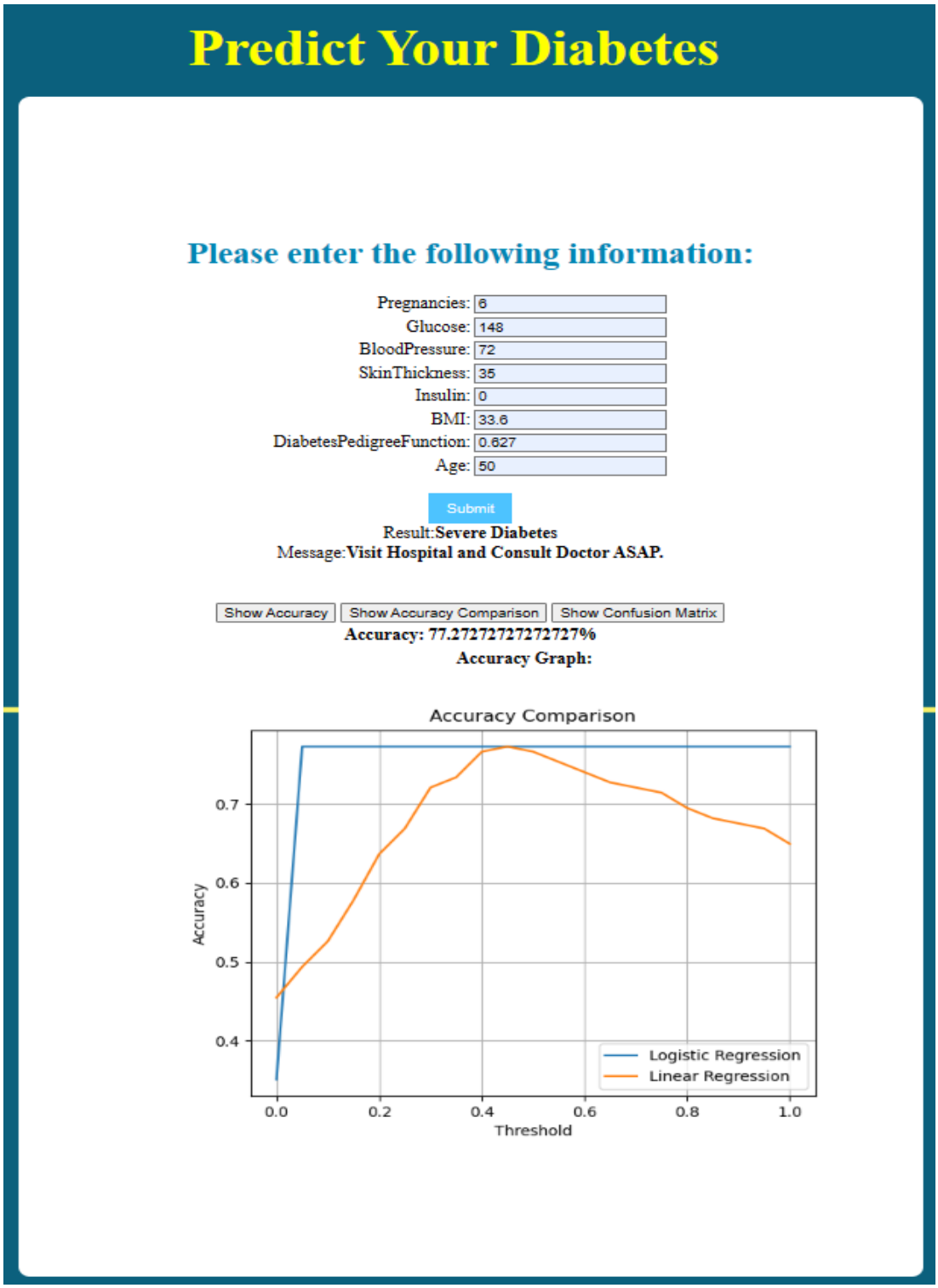

Fig A2: Predict Page

Appendices B: Major source code

1. Code For checking missing data

```
sns.heatmap(data.isnull())
```

```
<AxesSubplot:>
```

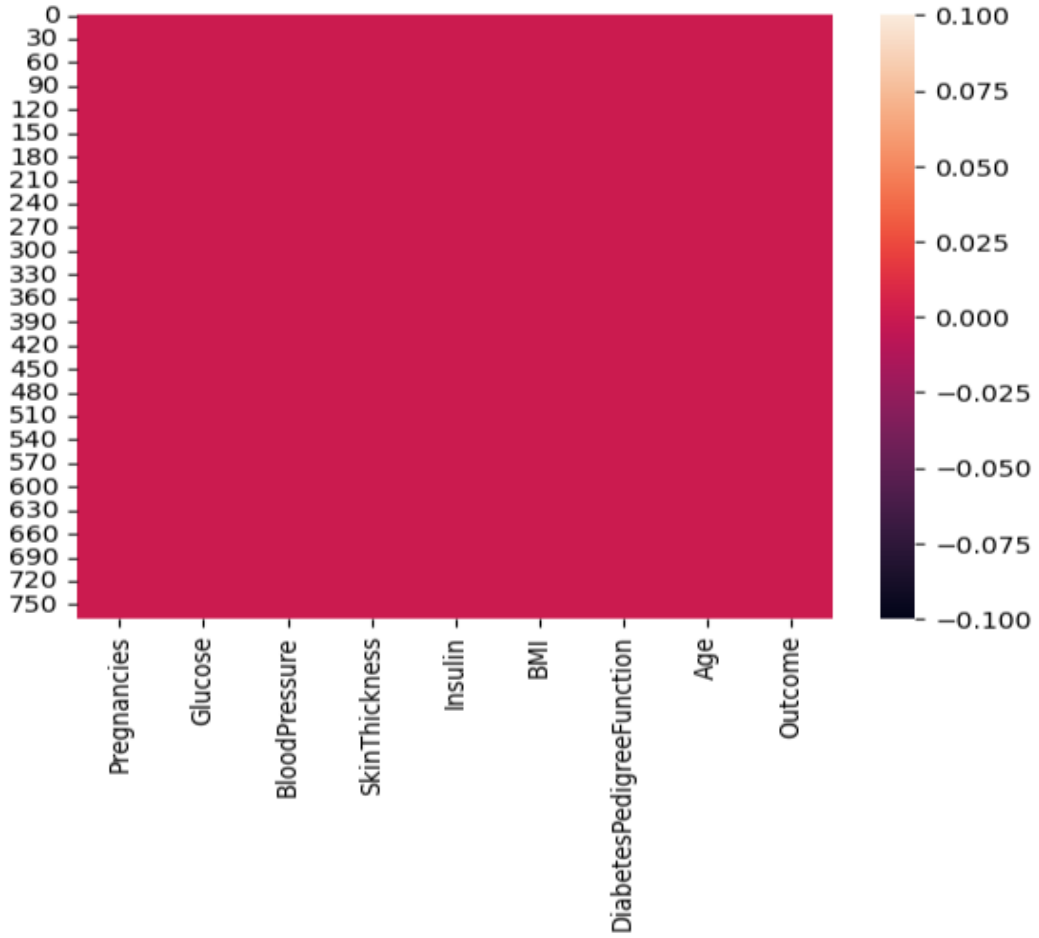

Fig B1: Checking Missing data

2. Code for Correlation Matrix

```
correction = data.corr()
sns.heatmap(correlation)
```

```
<AxesSubplot:>
```

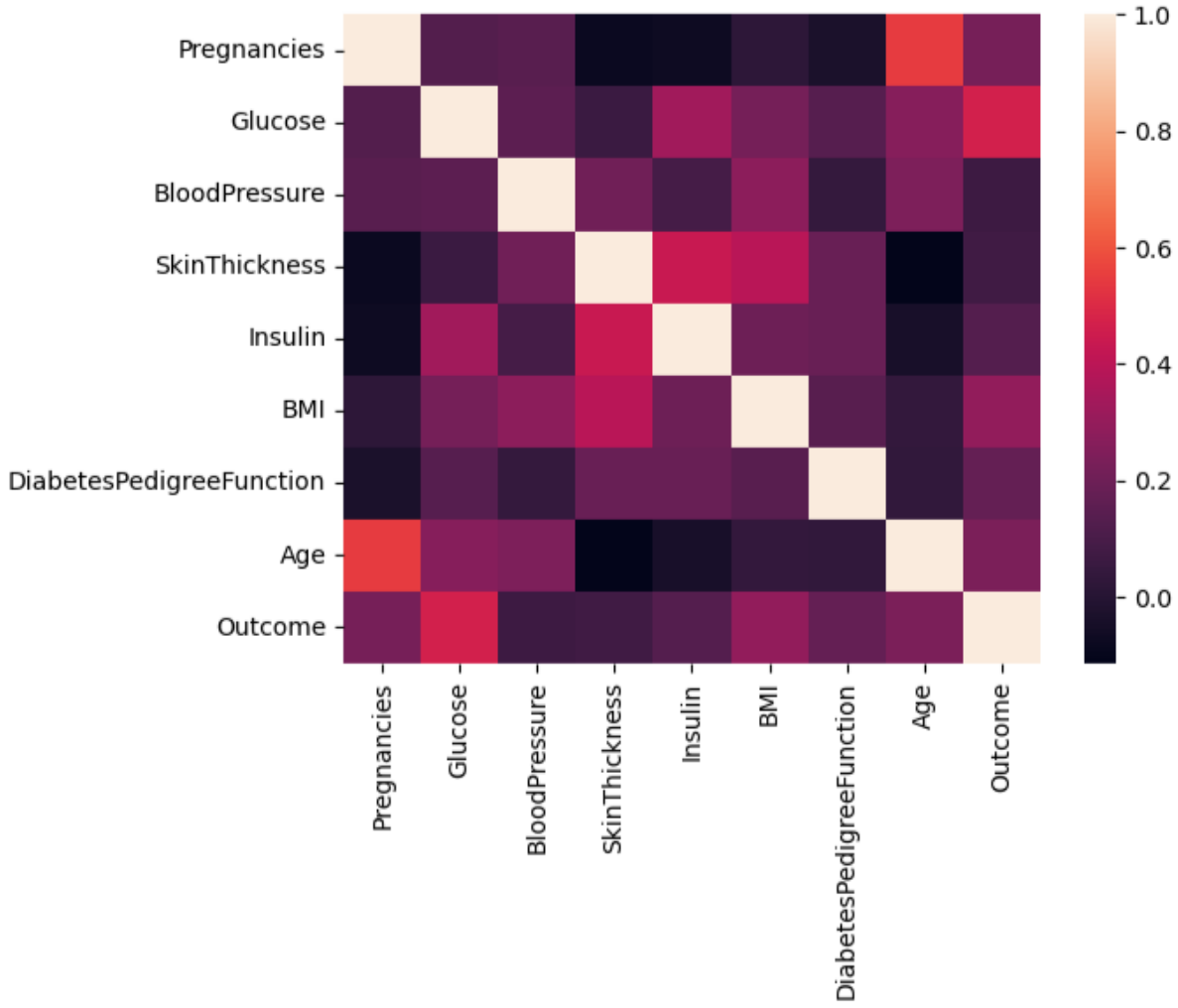

Fig B2: Correlation Matrix

Appendices C: Log of Visit to Supervisor

Date	Topic of discussion	Suggestions	Signature
12/18/2022	Discussion on our project topic, necessary requirements and planning to execute the project successfully.	To examine relevant research papers and identify previous projects that are related to the topic at hand.	
12/28/2022	Discussion on relevant projects to determine the optimal user interface design for the web application.	To create proper dataset and integrate it as some basic web application	
01/05/2023 - 01/31/2023	Made the project more user-friendly to show its usability and development of an ML Model to evaluate the dataset for Diabetes.	Build a basic system authorization system and use the appropriate classifier and preprocessing techniques to improve model accuracy.	
02/01/2023 - 03/13/2023	Increased model accuracy and properly integrated the model into the user interface.	To train the model in Logistic regression for a longer time to improve accuracy.	
4/2/2023 - 4/15/2023	Drafted the documentation regarding the project.	To create the documentation according to the standard provided in the syllabus.	
04/20/2023	Reviewed the final documentation.	To Format the documentation and consult the supervisor for final proofreading.	
5/6/2023	Mid Defense report submission	To rewrite the conclusion and result analysis, compare the logistic and linear regression	

		algorithms, and enhance the dashboard.	