A
PROJECT REPORT ON

Diabetes Prediction System

By

Chudasama Akshayrajsinh - CE007
Darji Mit - CE066

B. Tech. CE Semester – VI
Subject: System Design Practice

Guided by:
Prof. Shaifali P. Malukani
Assistant Professor
Dept. of Comp. Engg.

Faculty of Technology
Department of Computer Engineering
Dharmsinh Desai University

Faculty of Technology
Department of Computer Engineering
Dharmsinh Desai University

CERTIFICATE

This is to certify that the practical / term work carried out in the subject of **System Design Practice** and recorded in this journal is the bonafide work of

Chudasama Akshayrajsinh (CE007) (21CEU0G108)
Darji Mit (CE066) (21CEUON151)

of B.Tech semester **VI** in the branch of **Computer Engineering** during the academic year **2023-2024**

Prof. Shaifali P. Malukani
Assistant Professor,
Dept. of Computer Engg.
Faculty of Technology
Dharmsinh Desai University, Nadiad

Dr. C. K. Bhensdadia
Head of the Department,
Dept. of Computer Engg.
Faculty of Technology
Dharmsinh Desai University,Nadiad

Acknowledgement

In reflecting upon the completion of this project, we are filled with profound gratitude towards those whose unwavering support and guidance have illuminated our path. Without their generous assistance, our journey would have been fraught with challenges.

We extend our sincerest appreciation to our educational institution for providing us with the opportunity to embark on this endeavor, allowing us to apply our knowledge in a practical setting. The experience gained through this project has been invaluable to our growth and development.

A heartfelt thank you is owed to our mentor **Prof. Shaifali P. Malukani**, whose wisdom and encouragement have been the cornerstone of our progress. Their dedication to our success and their expertise have propelled us forward, ensuring that each task was completed with precision and excellence. Without their steadfast guidance, we would not have achieved our goals.

Furthermore, we wish to express our gratitude to all those who offered their support and assistance along the way, whether through insightful advice or timely intervention. Your contributions have been instrumental in overcoming obstacles and achieving our objectives.

In closing, we are deeply grateful to everyone who has played a part in this project, and we cherish the collaborative spirit that has made this endeavor possible.

Abstract

Our project aims to develop a user-friendly website for predicting diabetes risk using machine learning techniques. By analyzing key factors such as pregnancies, glucose levels, blood pressure, BMI, and age, users can easily input their data for personalized risk assessment. Traditional methods of diabetes risk assessment are often complex and prone to errors, hence our website seeks to simplify this process, providing accurate predictions in a straightforward manner. If the prediction indicates a higher likelihood of diabetes, users will receive alerts prompting them to seek medical advice and adopt preventive measures, thus facilitating early detection and effective management of the condition.This proactive stance not only fosters early detection but also ensures effective management of the condition, thereby significantly enhancing health outcomes for individuals susceptible to diabetes.

Table of Contents

1. Introduction

In our effort to improve health awareness, we're introducing a simple-to-use website focused on early detection of diabetes. This website features a special tool designed to help users assess their risk of developing diabetes. By entering basic information like pregnancies, glucose levels, blood pressure, BMI, and age, individuals can receive personalized predictions about their likelihood of developing diabetes. Our goal is to provide an accessible and intuitive platform that empowers users to take proactive steps towards better health.

With traditional methods of diabetes risk assessment often being complicated and prone to mistakes, our website aims to simplify this process. By offering accurate predictions in a clear and straightforward manner, we hope to bridge the gap between complex medical information and everyday understanding. Moreover, if the prediction indicates a higher risk of diabetes, users will be alerted to seek medical advice and adopt preventive measures promptly. By facilitating early detection and intervention, we aspire to contribute to better health outcomes and improved quality of life for individuals at risk of diabetes.

Tools, Technology and Platform used :

1) **Programming Languages:** Python

2) **Web Framework :** Django

3) **IDE:** Visual Studio Code and Google Colab

4) **Python Libraries used :** sci-kit learn,pandas,numpy,matplotlib,seaborn

2. Software Requirement Specifications (SRS)

2.1 Introduction

2.1.1 Purpose

The purpose of this document is to define the software requirements for a Diabetes Prediction System using Python and the Django framework.By leveraging machine learning algorithms and data visualization techniques, the system assists in early detection and intervention, ultimately improving patient outcomes and promoting better healthcare management. Additionally, the system offers functionality to generate detailed reports based on input data, facilitating informed decision-making and enhancing the overall user experience. Our system uses **Random Forest Classifier** which gives the best accuracy among others.

2.1.2 Scope

The Diabetes Prediction System encompasses features such as user input, machine learning model training, prediction, data visualization, and report generation. It aims to handle input parameters including pregnancies, glucose levels, blood pressure, BMI, age, skin thickness, and insulin levels, utilizing multiple machine learning algorithms for prediction. The system prioritizes accuracy, performance, and user-friendliness, ensuring seamless integration into healthcare workflows and adherence to privacy and security standards.

2.2 System Functional Requirements

2.2.1 User Input

Input: User provides parameters including:
Pregnancies,Glucose levels,Blood pressure,BMI (Body Mass Index),Age
Skin thickness,Insulin levels
Output: Input data is validated for accuracy and completeness.
If any invalid or missing data is detected, appropriate error messages are displayed to the user.

2.2.2 Machine Learning Model Training

Input: Historical labeled data for training.
Output: Trained models using multiple machine learning algorithms:
Logistic regression,SVM (Support Vector Machine),Random forest,
Decision tree,XGBoost,Linear regression,KNN (k-Nearest Neighbors)
Naive Bayes,Gradient boosting

2.2. 3 Model Evaluation

Input: Trained models and evaluation dataset.
Output: Performance metrics for each model including:
Accuracy,Precision,Recall
The model with the highest accuracy is selected for deployment.

2.2.4 Prediction

Input: User input parameters for prediction.
Output: Prediction result indicating the likelihood of diabetes:
Positive (high risk of diabetes),Negative (low risk of diabetes)

2.2.5 Data Visualization

Input: Input parameters, prediction results.
Output: Visualizations to aid interpretation, including:
Charts/graphs depicting input parameters and prediction results.

2.2.6 Report Generation

Input: User input parameters, prediction results.
Output: Generated report containing:

Input parameter,Prediction results,Visualizations (if applicable)
These functional requirements describe the expected input and output for each system feature or functionality, ensuring clarity and alignment with user expectations.

2.3 Non-Functional Requirements

2.3.1 Performance

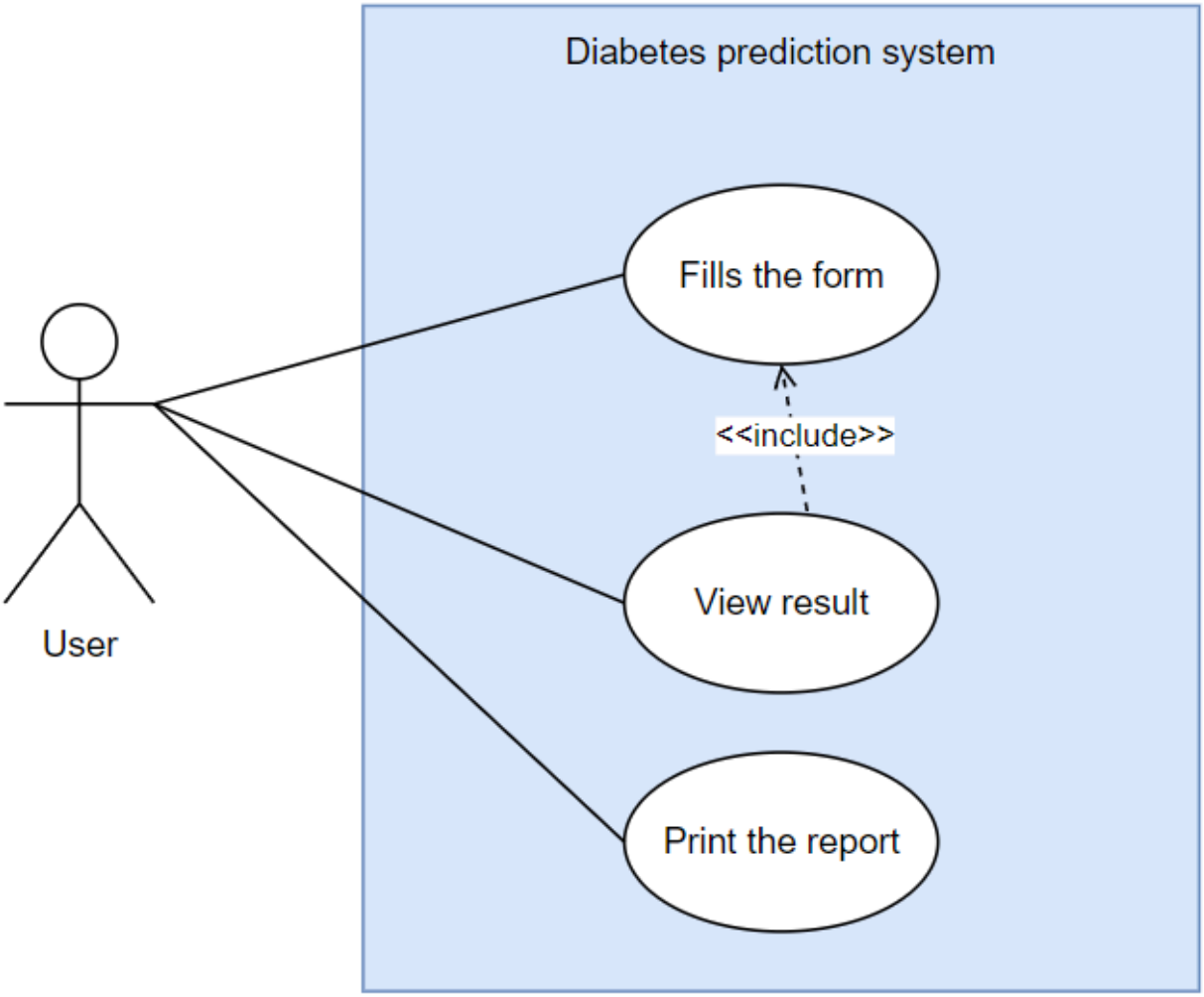
- The system should provide fast and accurate predictions, with minimal latency.
- Ensure efficient memory utilization and processing to handle large datasets.

2.3.3 Usability

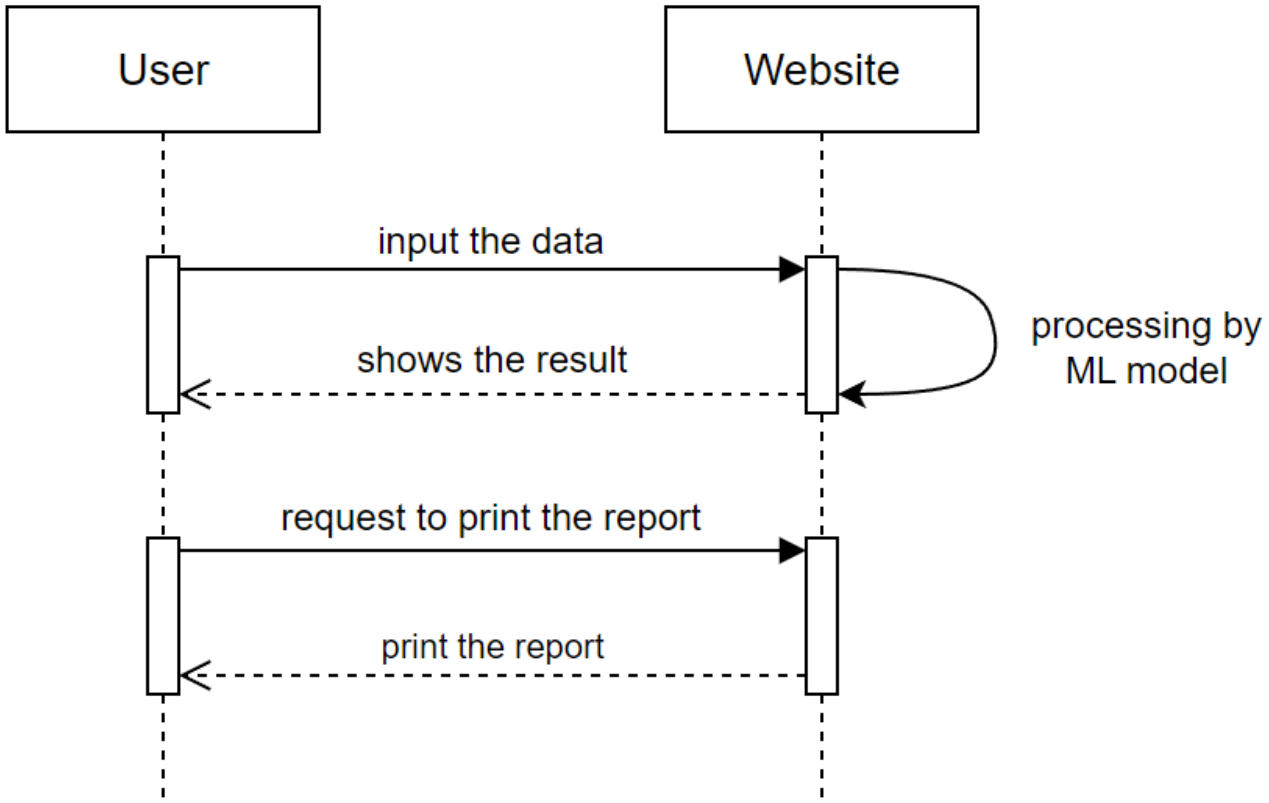
- Design a user-friendly interface with intuitive input fields and clear output.
- Provide informative error messages for input validation.

3. System Design

3.1 Use Case Diagram



3.2 Sequence Diagram

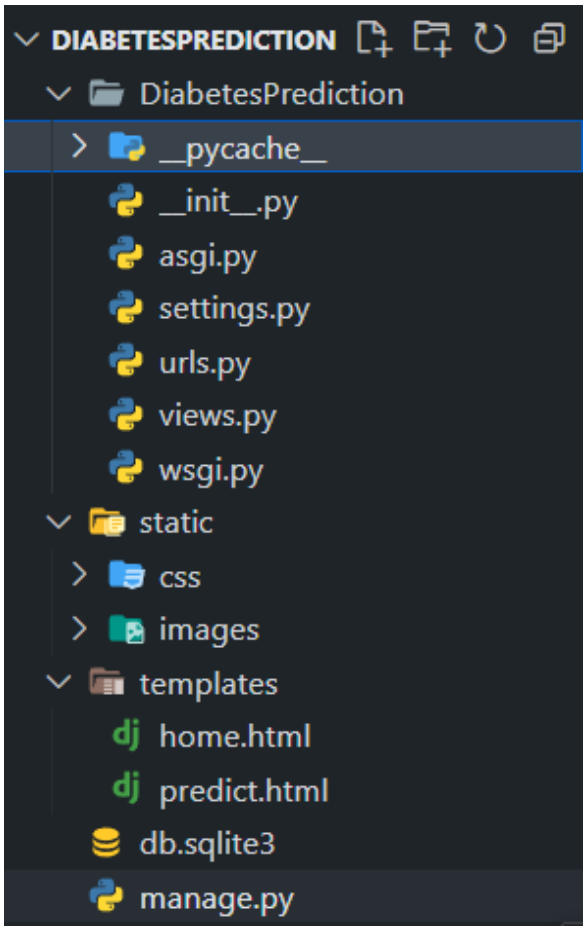


4. Implementation Details

4.1 Front-end Implementation Details

Front end is designed using the django framework. Develop a user interface using Django templates or frontend frameworks like Bootstrap.Ensure compatibility with different device and browsers.

Django App's folder structure:



Folder Structure :-

1) Urls.py :- Defines URL patterns

2) View.py :- Handles request logic

3) Manage.py :- A management tool

4) Templates :- Contains HTML files for presentation

5) Static :- Stores static files such as css, javascript,images

In the templates folder,

- home.html file represents the home page.
- predict.html contains a form for user input, providing diabetes prediction results based on the entered data

4.2 Back-End Design Implementation Details

We trained our model on the diabetes.csv database, implementing and comparing various machine learning algorithms, including:

- Logistic Regression
- Support Vector Machines (SVM)
- Random Forest
- Decision Tree
- XGBoost
- Linear Regression
- K-Nearest Neighbors (KNN)
- Naive Bayes
- Gradient Boosting

After comparing the accuracy of these models, we found that the Random Forest algorithm provided the highest accuracy. As a result, we chose to perform our diabetes prediction using the **Random Forest model**.

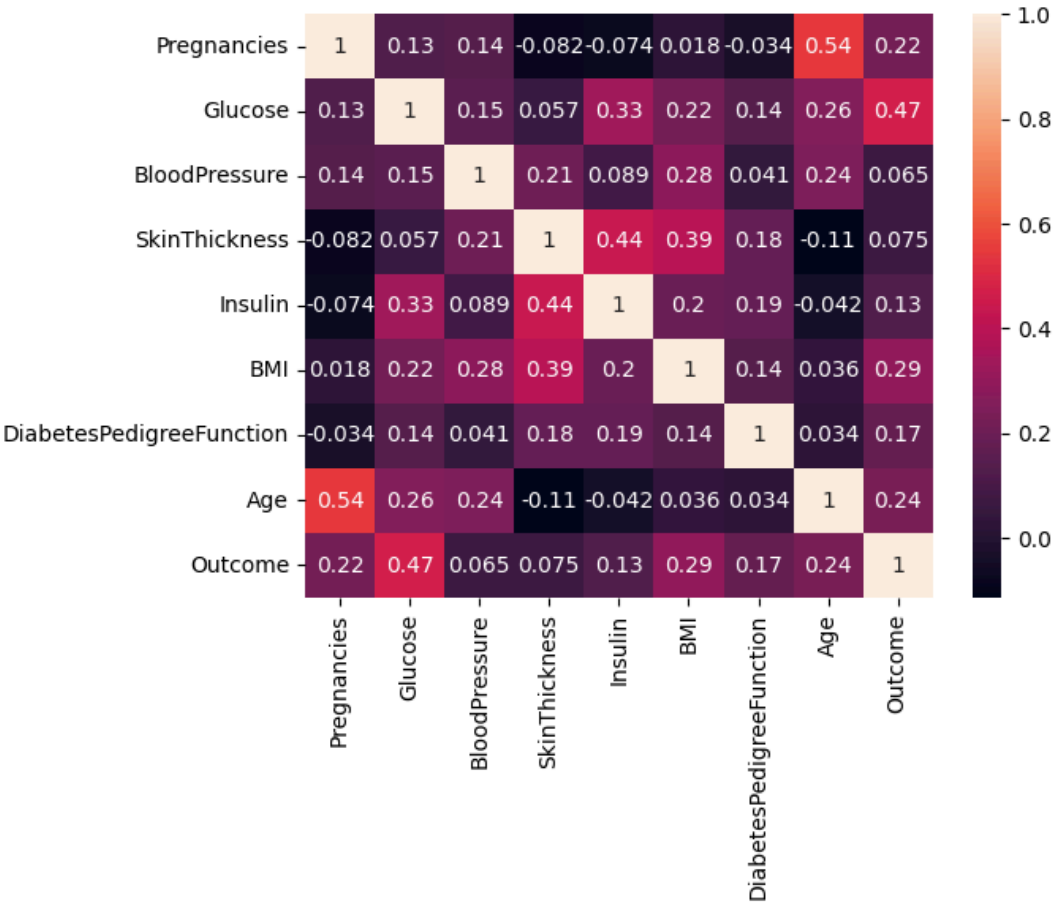
We also calculated the precision and recall values for our model to evaluate its performance further.

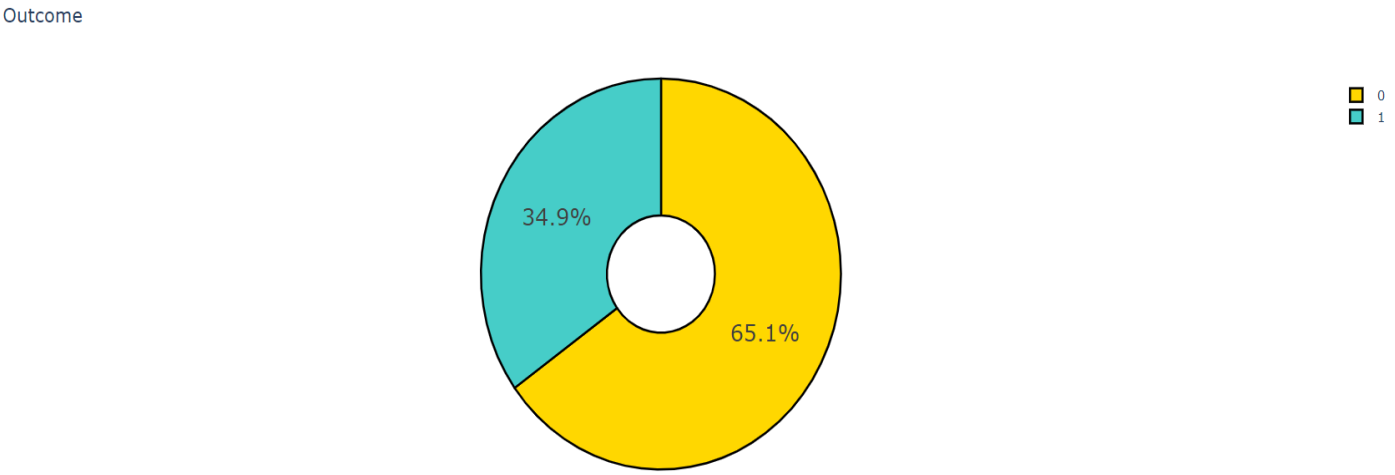
Data Visualization :-

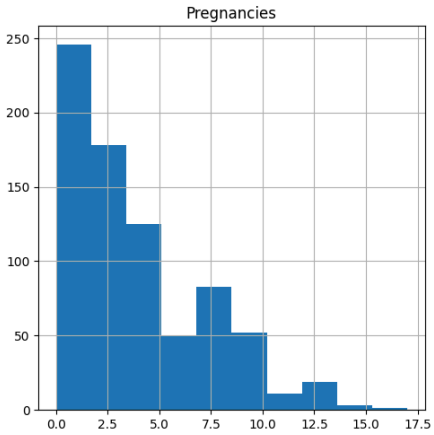
To help us better understand the data and the performance of our model, we used various data visualization libraries, including:

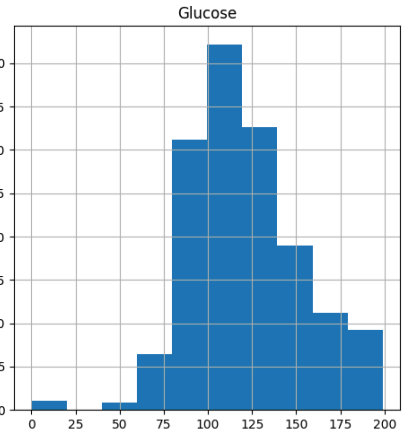
- Scikit Learn
- Pandas
- NumPy
- Matplotlib
- Seaborn

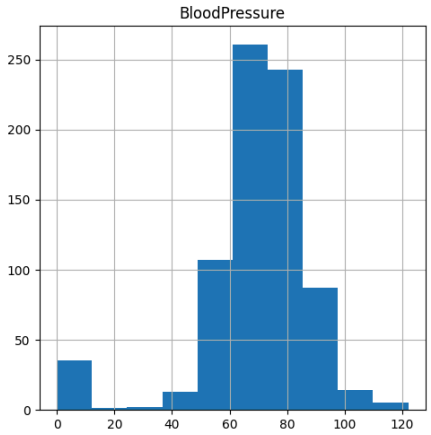
We used these libraries to create visualizations such as scatter plots, histograms, and heatmaps to gain insights into the data and the model's performance.

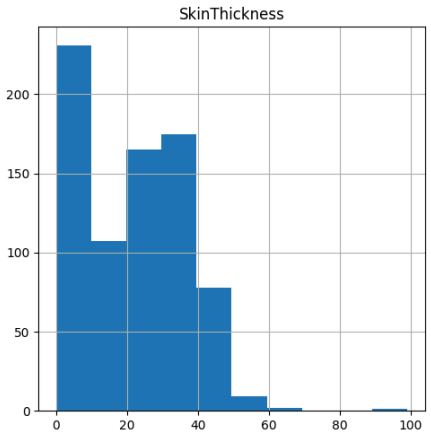

Pregnancies 1 0.13 0.14 -0.082 -0.074 0.018 -0.034 0.54 0.22
Glucose 0.13 1 0.15 0.057 0.33 0.22 0.14 0.26 0.47
BloodPressure 0.14 0.15 1 0.21 0.089 0.28 0.041 0.24 0.065
SkinThickness -0.082 0.057 0.21 1 0.44 0.39 0.18 -0.11 0.075
Insulin -0.074 0.33 0.089 0.44 1 0.2 0.19 -0.042 0.13
BMI 0.018 0.22 0.28 0.39 0.2 1 0.14 0.036 0.29
DiabetesPedigreeFunction -0.034 0.14 0.041 0.18 0.19 0.14 1 0.034 0.17
Age 0.54 0.26 0.24 -0.11 -0.042 0.036 0.034 1 0.24
Outcome 0.22 0.47 0.065 0.075 0.13 0.29 0.17 0.24 1
1.0
0.8
0.6
0.4
0.2
0.0
Pregnancies
Glucose
BloodPressure
SkinThickness
Insulin
BMI
DiabetesPedigreeFunction
Age
Outcome

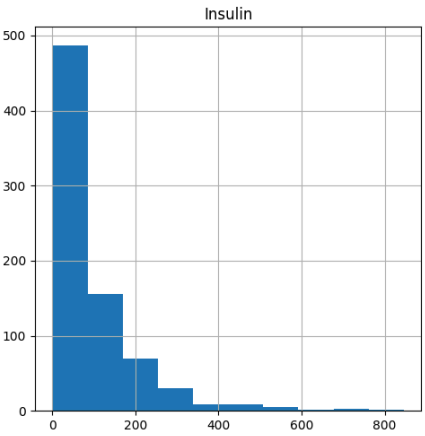

Outcome
34.9%
65.1%
0
1

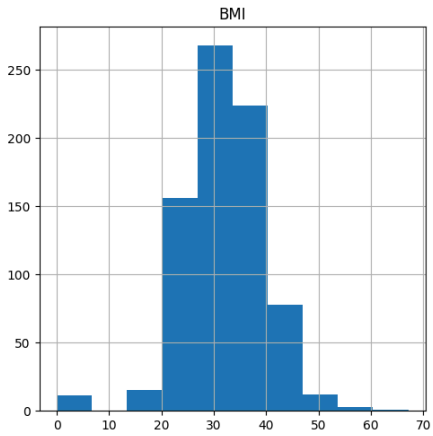

Pregnancies
0
50
100
150
200
250
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5

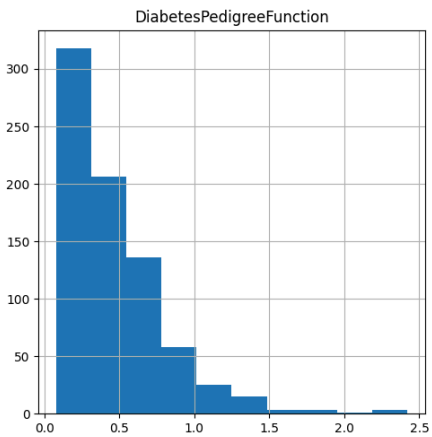

Glucose
0
25
50
75
100
125
150
175
200
0
25
50
75
100
125
150
175
200

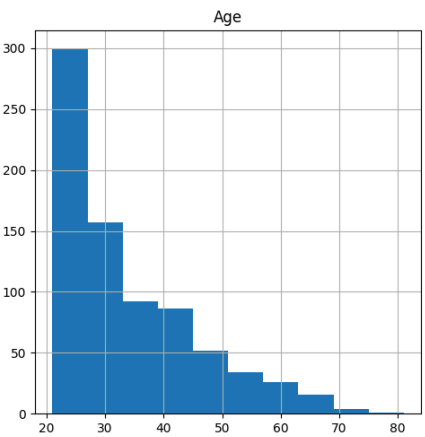

BloodPressure
0
50
100
150
200
250
0
20
40
60
80
100
120

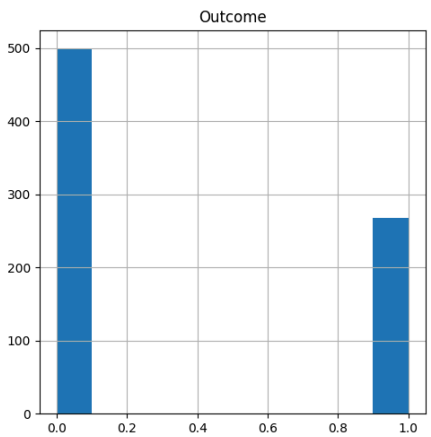

SkinThickness
0
50
100
150
200
0
20
40
60
80
100


Insulin
0
100
200
300
400
500
0
200
400
600
800


BMI
0
50
100
150
200
250
0
10
20
30
40
50
60
70

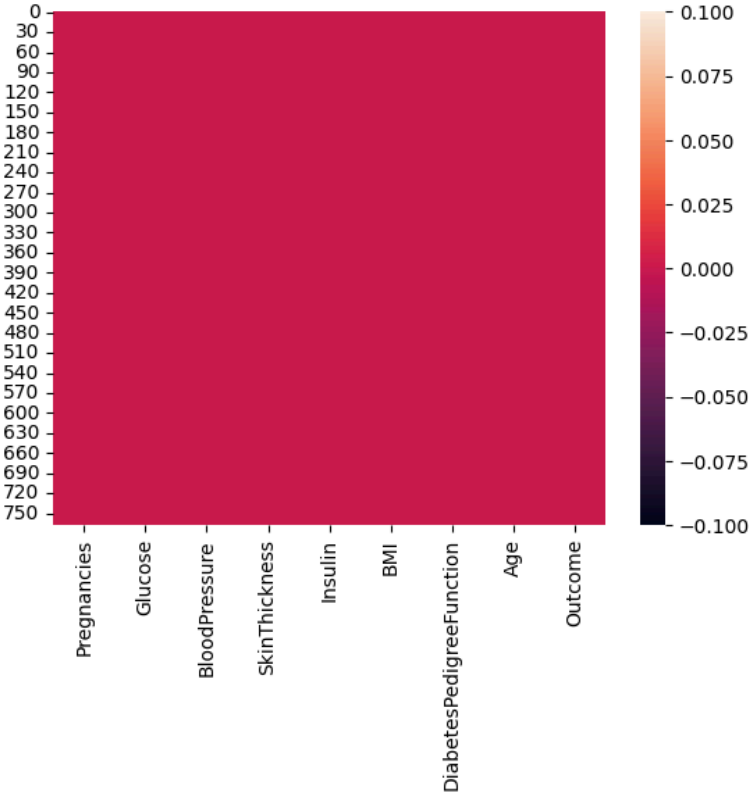

DiabetesPedigreeFunction
0
50
100
150
200
250
300
0.0
0.5
1.0
1.5
2.0
2.5

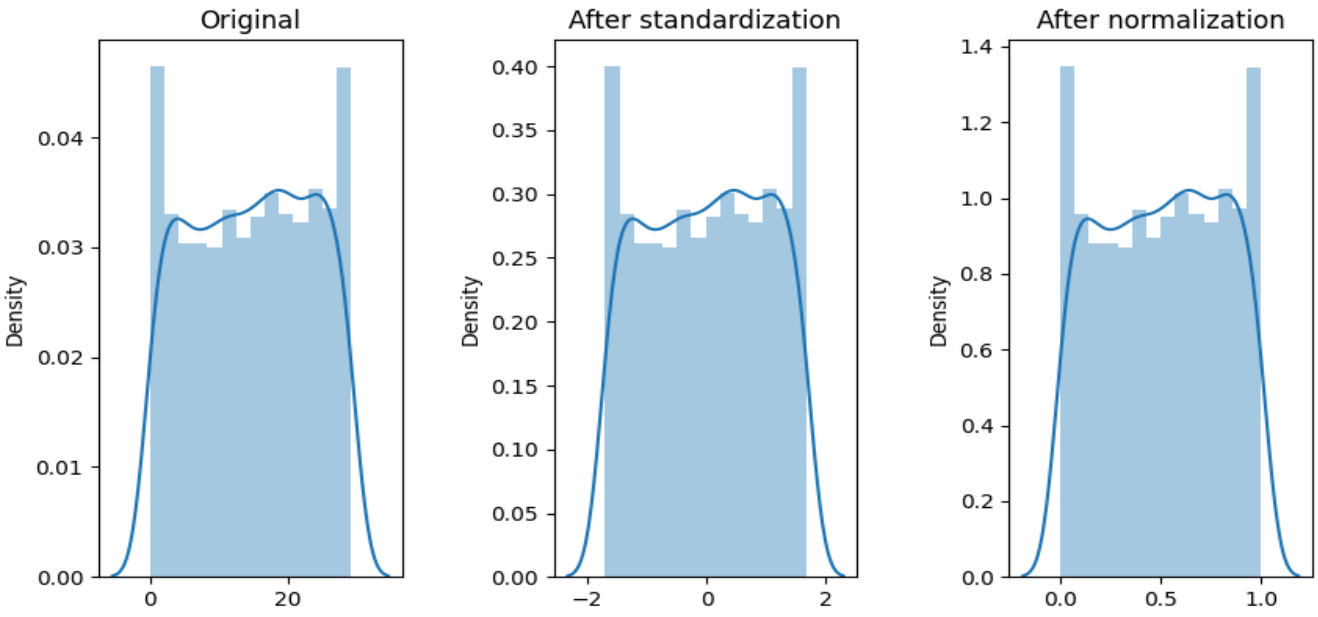

Age
0
50
100
150
200
250
300
20
30
40
50
60
70
80


Outcome
0
100
200
300
400
500
0.0
0.2
0.4
0.6
0.8
1.0

Missing Data and Standardization:-

Before training our model, we checked for missing data and handled it appropriately. We also used standardization techniques such as scaling and normalization to ensure that all features had the same range and distribution.

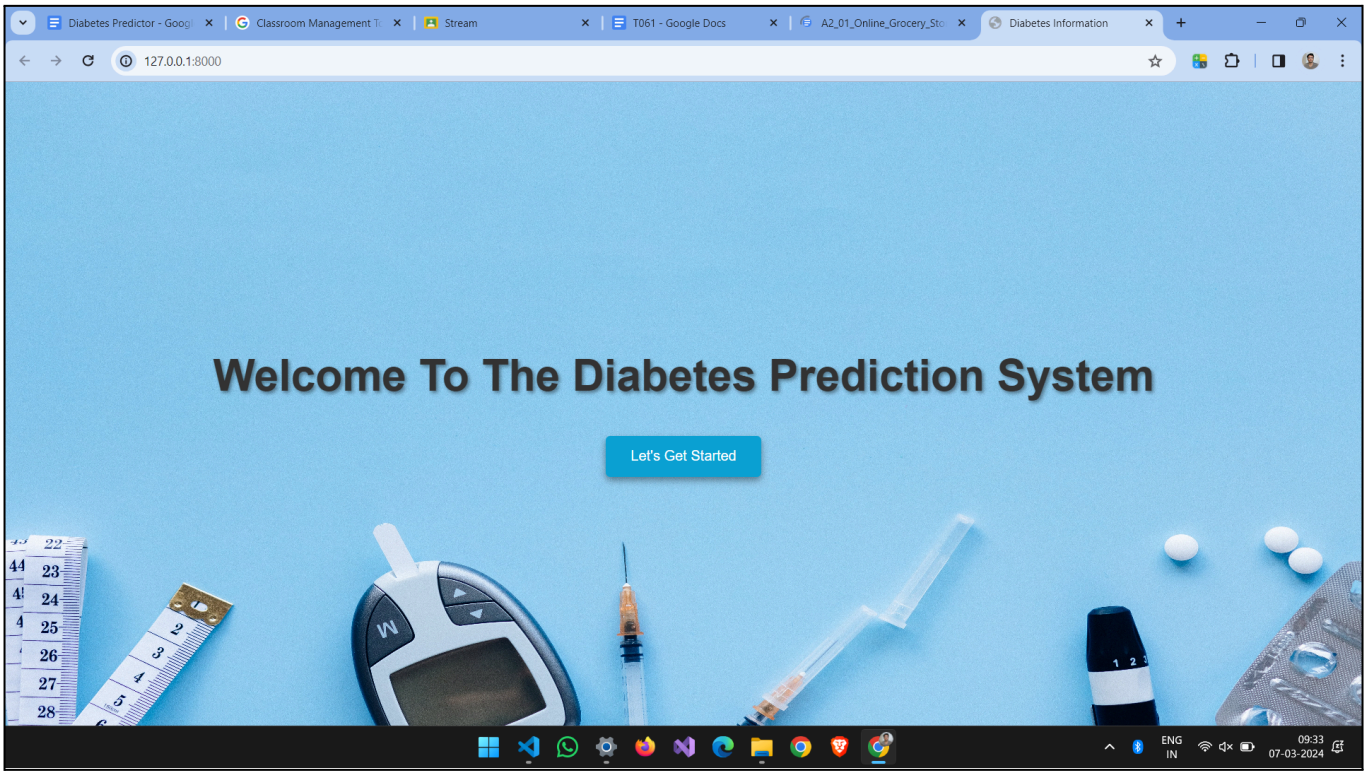




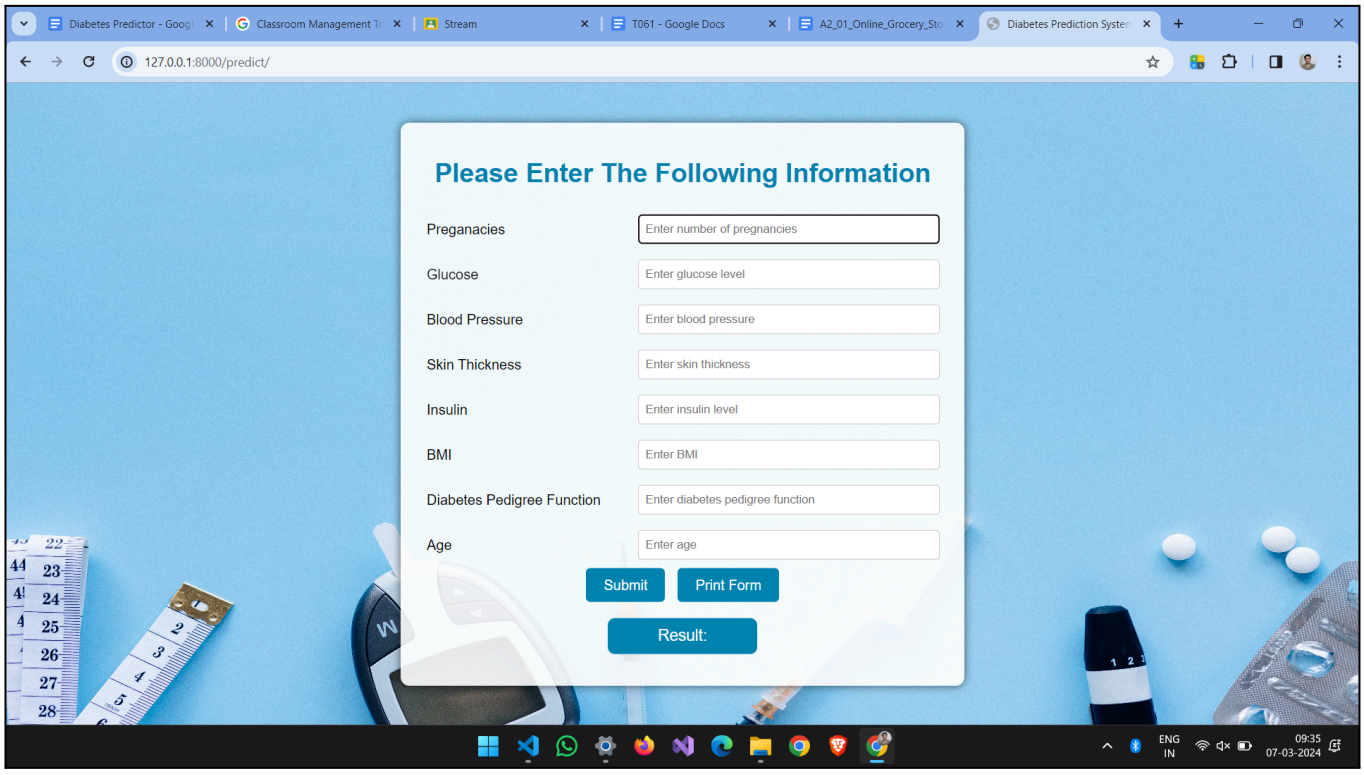
5. Model Accuracy Score

Model	Accuracy
Linear Regression	26.50
Logistic Regression	79.22
Polynomial Regression	29.98
SVM (Support Vector Machine)	76.62
Random Forest Classifier	80.51
Decision Tree Classifier	74.02
XG Boosting	76.62
KNN (K-Nearest Neighbors)	68.83
Naive Bayes Classifier	74.02
Gradient Boosting	72.72

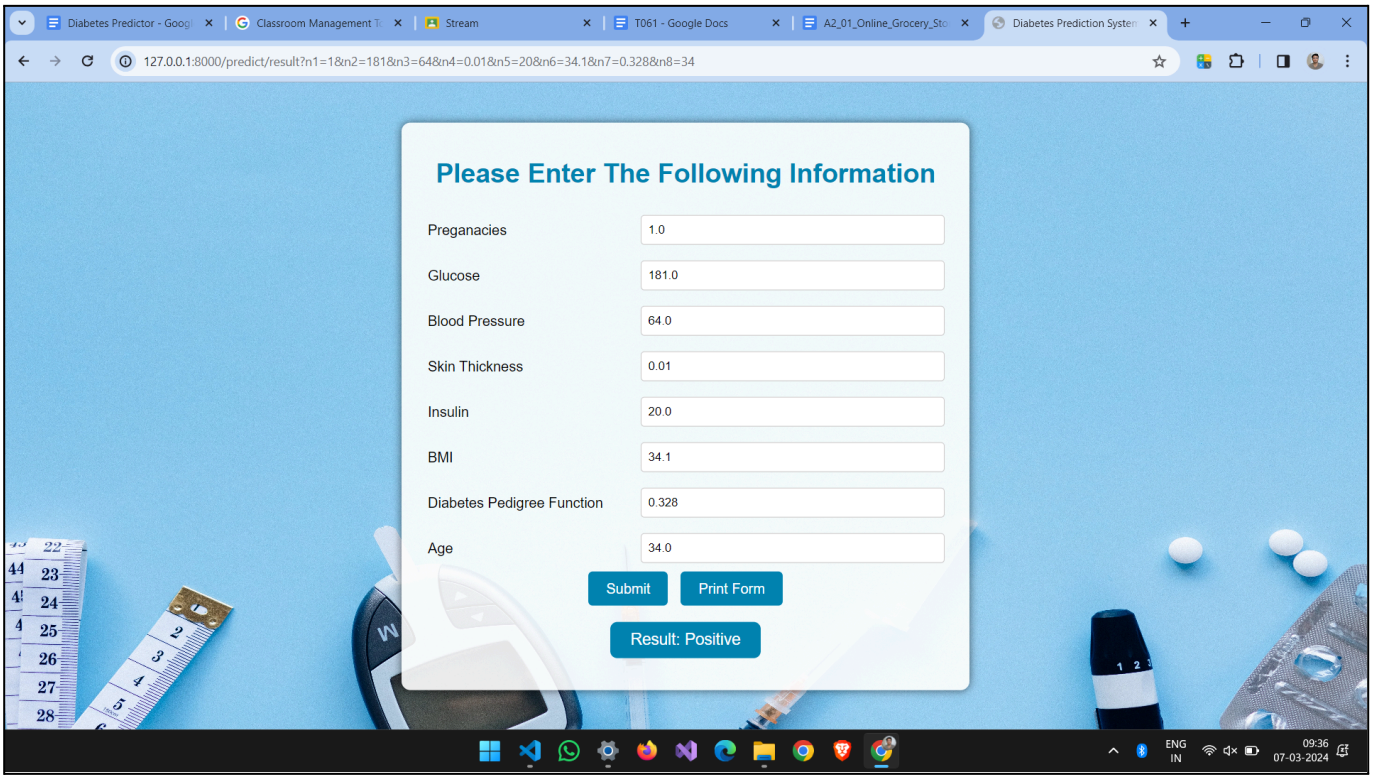
6. Screenshots



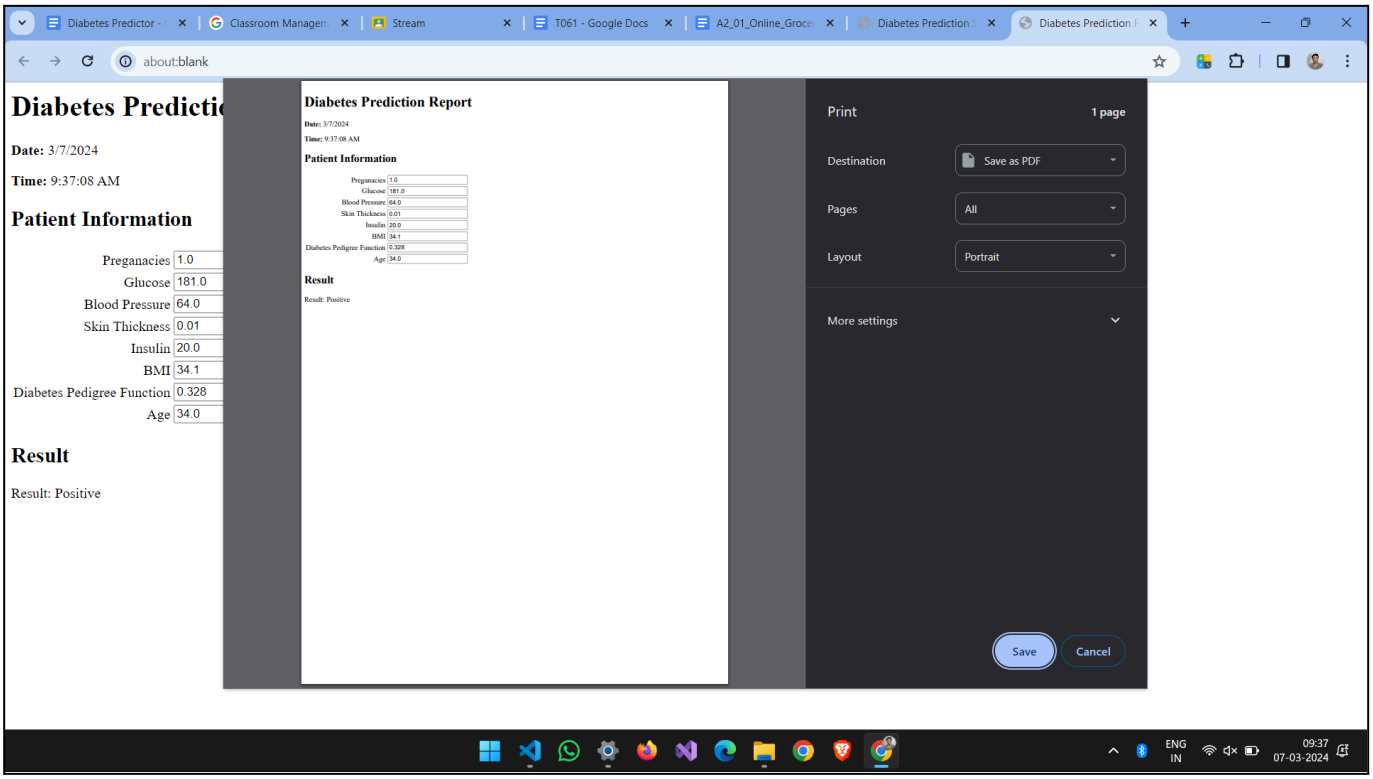
6.1 Home page



6.2 Prediction Page



6.3 Prediction Page



6.4 Report page

7. Conclusion

In conclusion, the creation of our diabetes predictor website using Django has been a significant endeavor in utilizing machine learning for proactive healthcare. Through the implementation of various machine learning algorithms such as linear regression, k-nearest neighbors (KNN), logistic regression, and decision trees, we have successfully developed a robust tool capable of predicting an individual's risk of developing diabetes based on key health metrics.

The integration of our predictive model into a Django-based website ensures accessibility and user-friendliness for individuals seeking to assess their diabetes risk. The backend infrastructure, powered by Django, facilitates seamless interaction between the website and machine learning algorithms, ensuring reliable performance and accurate predictions.

As we conclude this project, we reflect on the achievement of our goals in creating a functional and effective diabetes predictor website. Our endeavor has been driven by the desire to empower individuals with valuable insights into their health, and we are proud to have developed a tool that contributes to proactive healthcare practices.

In summary, this project exemplifies our commitment to leveraging technology for the betterment of healthcare outcomes. By providing a user-friendly platform for diabetes risk assessment, we aim to support individuals in making informed decisions about their health and well-being.

"Machine learning is the new electricity."

-Andrew Ng

8. Limitation and Future Extension

Limitations of the Project:

1. Data Quality: The accuracy of the predictions heavily relies on the quality and quantity of the input data. Inaccurate or insufficient data may lead to unreliable predictions.
2. Generalization: The predictive model may not generalize well to unseen data, particularly if the dataset used for training is not representative of the population.
3. User Interface Complexity: The user interface of the website may not be intuitive for all users, potentially limiting its accessibility and usability.

Future Extensions:

1. Enhanced User Interface: Improve the website's user interface to make it more intuitive and user-friendly, catering to a broader range of users.
2. Feature Engineering: Explore additional features or incorporate domain-specific knowledge to improve the accuracy of the predictive model.
3. Model Deployment: Deploy the predictive model on a scalable platform to handle a larger volume of users and ensure real-time predictions.

9. Bibliography

1. Pima Indians Diabetes Database
2. Diabetes Prediction System - YouTube
3. Diabetes Prediction Using Machine Learning - Analytics Vidhya
4. Diabetes Prediction using Machine Learning Algorithms - ScienceDirect