



KARATINA UNIVERSITY

SCHOOL OF PURE AND APPLIED SCIENCES

DEPARTMENT OF COMPUTER SCIENCE AND INFORMATICS

**PROJECT TITLE: A MACHINE LEARNING MODEL TO PREDICT IF  
PATIENT WILL DEVELOP DIABETES IN FUTURE USING RISK FACTORS.**

By

NICODEMUS MUIMI MALOMBE

REG NO: P101/1749G/21

Date: \_\_\_\_\_

This project is submitted in full fulfilment of requirement for the Karatina University award of BACHELOR OF SCIENCE IN COMPUTER SCIENCE.

## DECLARATION

### STUDENT

I, NICODEMUS MUI MI MALOMBE, declare that this project report titled “A MACHINE LEARNING MODEL TO PREDICT IF A PATIENT WILL DEVELOP DIABETES IN FUTURE USING RISK FACTORS” is my original work and has not been submitted to any other institution for academic credit.

This project has been carried out in accordance with the academic guidelines and ethical considerations required for research and development. All sources of information from other works have been duly acknowledged and cited in this report.

Name: \_\_\_\_\_

Reg No.: \_\_\_\_\_

Date: \_\_\_\_\_

Signature: \_\_\_\_\_

### SUPERVISOR

I the undersigned do hereby certify that this is a true report for the project undertaken by the above named student under my supervision and that it has been submitted to Karatina University with my approval.

Signature.....Date.....

## DEDICATION

I dedicate this project to my parents, whose unwavering support and encouragement have been my guiding light. To my supervisor, MR. Zablon Okari, for her invaluable guidance and wisdom. And to my friends, for their constant inspiration and belief in my abilities and also their help throughout my journey of my project process is highly appreciated.

## ACKNOWLEDGEMENT

I would like to thank everyone who contributed to the successful completion of this project. I would like to extend my gratitude to my project supervisor, Mr. Zablon okari for his invaluable advice, guidance and enormous patience throughout the development of the research. In addition, I would also like to express my gratitude to my loving parent and friends who helped and gave me encouragement in one way or the other. Also I am grateful for the support of Karatina university for providing the facilities necessary to conduct my research not forgetting the school of nursing students at Karatina university for educating me on the necessary knowledge for the success of this project.

## LIST OF ABBREVIATIONS

1. AI: Artificial Intelligence
2. BMI: Body Mass Index
3. ML: Machine Learning
4. WHO: World Health Organization
5. API: Application Programming Interface
6. LSTM: Long Short-Term Memory

## ABSTRACT

Diabetes mellitus could cause severe complications if left uncontrolled due to increased blood sugar levels. Early diagnosis of diabetes mellitus entailed appropriate management and prevention of complications. This project aimed at developing a predictive model to estimate the likelihood of diabetes based on medical and physiological profiles. Supervised machine learning was employed using publicly available data on diabetes, including features such as pregnancy, insulin level, BMI, age, and blood glucose levels. Optimal algorithms were selected based on their accuracy and computational efficiency to ensure reliable predictions. A web interface was developed to enable users to input their health information and receive instantaneous predictions via a user-friendly platform. The system provided risk assessments for diabetes and encouraged users to consult healthcare professionals when necessary. This review highlighted the growing need for accessible and accurate predictive tools by addressing limitations found in existing systems. The project contributed to public health by supporting early identification and timely intervention through techniques such as data preprocessing and feature engineering. Machine learning demonstrated its potential in enhancing healthcare access and outcomes, with opportunities for future innovations. The model proved to be a valuable tool in the fight against diabetes and showed promise for integration into broader healthcare systems.

## Table of Contents

DECLARATION.....	2
DEDICATION .....	3
ACKNOWLEDGEMENT .....	4
LIST OF ABBREVIATIONS.....	5
ABSTRACT.....	6
LIST OF TABLES .....	9
LIST OF FIGURES .....	9
CHAPTER ONE .....	10
1.1 INTRODUCTION .....	10
1.2 BACKGROUND OF THE STUDY .....	11
1.3 PROBLEM STATEMENT.....	12
1.4 OBJECTIVES.....	12
1.4.1 GENERAL OBJECTIVE: .....	12
1.4.2 SPECIFIC OBJECTIVES.....	13
1.5 JUSTIFICATION OF THE PROBLEM .....	13
1.6 PROJECT RISK AND MITIGATION.....	14
1.7 SCOPE.....	15
1.8 LIMITATIONS OF THE STUDY .....	15
1.9 BUDGET AND RESOURCES .....	16
1.91. Hardware Requirements .....	16
1.92. SOFTWARE REQUIREMENTS .....	17
1.93. HUMAN RESOURCES .....	17
1.10 PROJECT SCHEDULE.....	17
1.11. WORK BREAKDOWN STRUCTURE (WBS) .....	17
1.12. GANTT CHART .....	18
1.13. NETWORK DIAGRAM & CRITICAL PATH.....	19
CHAPTER THREE.....	23
METHODOLOGY .....	23
3.3 DATA COLLECTION AND PREPROCESSING .....	25
3.3.1 DATA SOURCES .....	25
3.3.2 DATA PREPROCESSING TECHNIQUES.....	25
3.4 TOOLS.....	26

3.5 CONCLUSION .....	27
3.6 BUDGET .....	27
3.7 GANTT CHART .....	28
CHAPTER FOUR: SYSTEM ANALYSIS AND REQUIREMENT MODELING.....	29
4.1 CURRENT SYSTEM ANALYSIS .....	29
4.1.1 DATA COLLECTION METHODS .....	29
4.1.2 SYSTEM MODELING TOOLS .....	29
4.2 REQUIREMENT DEFINITION AND MODELING OF THE CURRENT SYSTEM.....	31
4.3 REQUIREMENT DEFINITION AND SPECIFICATIONS OF THE PROPOSED SYSTEM.....	31
4.3.1 FUNCTIONAL REQUIREMENTS .....	31
4.3.2 NON-FUNCTIONAL REQUIREMENTS .....	31
CHAPTER FIVE: SYSTEM DESIGN .....	33
5.1 INTRODUCTION .....	33
5.2 SYSTEM ARCHITECTURE.....	33
5.3 Input Design.....	35
5.4 Output design.....	36
6.1 INTRODUCTION .....	38
6.2 TOOLS USED FOR CODING AND TESTING .....	38
6.3 SYSTEM TEST PLAN .....	39
6.3.1 Unit Testing: .....	39
6.3.2 Integration Testing.....	39
6.3.3 System Testing:.....	40
6.4 TESTING APPROACH AND DATA USED .....	40
6.4.1 Testing Approach.....	40
Evaluation Metrics: .....	40
6.4.2 Performance Testing: .....	40
6.5 PROPOSED CHANGE-OVER TECHNIQUES .....	41
Phased Implementation:.....	41
CHAPTER SEVEN: LIMITATIONS, CONCLUSIONS AND RECOMMENDATIONS .....	42
7.1 LIMITATIONS.....	42
7.2 CONCLUSION .....	42
7.3 RECOMMENDATIONS .....	43
REFERENCES .....	44



## LIST OF TABLES

Page 17 Hardware Requirements

Page 17 software requirements

Page 18 human resources

Page 18 work breakdown structure (wbs)

Page 19 gantt chart

Page 26 tools

Page 28 budget

Page 28 gantt chart

## LIST OF FIGURES

Figure I network diagram & critical path

Figure 3.1: Agile SDLC for diabetes Prediction System

Figure 3.3: Input Design Diagram

Figure 3.4: Output Design Diagram

Figure 4.1 flowchart

Figure 4.2 Data Flow Diagrams (DFDs):

Figure 5.1 System Architecture Diagram

Figure 5.5: Entity-Relationship

Diagram (ERD)

## CHAPTER ONE

### 1.1 INTRODUCTION

Diabetes mellitus was a chronic metabolic disorder characterized by elevated blood sugar levels, which, if left unmanaged, could lead to severe complications such as cardiovascular disease, kidney failure, and neuropathy. According to the World Health Organization (WHO, 2023), over 537 million people worldwide were affected by diabetes, and this number was expected to rise significantly in the coming years. Early detection and intervention were crucial in preventing disease progression; however, traditional diagnostic methods often relied on symptom-based identification, which delayed timely preventive care.

The emergence of Machine Learning (ML) and Artificial Intelligence (AI) had transformed disease prediction by enabling automated, data-driven, and highly accurate diagnoses. Researchers had applied various ML algorithms, including logistic regression (Kowsari et al., 2017), random forests (Wu et al., 2018), and deep neural networks (Zhang et al., 2020), to predict diabetes. While these models demonstrated high accuracy, challenges related to interpretability, data security, and scalability continued to hinder their widespread adoption in clinical practice.

To address these limitations, this project proposed the development of an AI-powered diabetes prediction system that enhanced predictive accuracy, model transparency, and data protection. By bridging these gaps, the system aimed to provide a robust, privacy-preserving, and scalable solution for diabetes prediction. Ultimately, this innovation empowered clinicians, patients, and healthcare providers with timely, data-driven insights, paving the way for personalized AI-assisted diabetes management.

## 1.2 BACKGROUND OF THE STUDY

Diabetes mellitus was a long-term metabolic disorder that had become a pressing global health issue. It was mainly defined by persistently high blood sugar levels, which occurred due to either inadequate insulin production or the body's inability to use insulin effectively. This condition was classified into two main types: Type 1 diabetes (T1D), an autoimmune disease, and Type 2 diabetes (T2D), which was largely influenced by genetic predisposition and lifestyle factors.

According to the International Diabetes Federation (IDF, 2023), over 537 million adults worldwide were living with diabetes, and this number was expected to increase due to rising obesity rates, physical inactivity, and poor dietary choices. Detecting the disease early and taking timely preventive measures was crucial in reducing the risk of severe complications, including heart disease, kidney failure, nerve damage, and vision impairment.

The conventional methods used to diagnose diabetes included fasting blood glucose tests, HbA1c tests, and oral glucose tolerance tests (OGTT). However, these procedures were often invasive, time-consuming, and required frequent monitoring, making them less practical for early disease prediction. In response to these limitations, Artificial Intelligence (AI) and Machine Learning (ML) emerged as powerful tools for assessing diabetes risk by analyzing both historical and real-time patient data.

Several studies explored the role of machine learning in diabetes prediction:

Kowsari et al. (2017) applied logistic regression to the Pima Indians Diabetes dataset and achieved 78% accuracy. Despite its effectiveness, the model struggled with capturing complex feature interactions due to its linear nature.

Wu et al. (2018) implemented random forest algorithms, which improved accuracy to 85% and demonstrated robustness against imbalanced datasets. However, the high computational costs posed challenges for scalability.

Zhang et al. (2020) utilized deep neural networks to enhance accuracy further, but this came at the expense of reduced interpretability and increased computational demands.

Park et al. (2022) explored ensemble methods such as XGBoost and AdaBoost, which outperformed single models but required longer training times and added complexity.

Despite these advancements, challenges remained in terms of model interpretability, data security, real-time accessibility, and seamless integration into healthcare systems. Overcoming these obstacles was essential to developing AI-driven solutions that were scalable, practical, and capable of preserving patient privacy in diabetes prediction.

.

### 1.3 PROBLEM STATEMENT

Diabetes continued to be a significant global health challenge, affecting millions of individuals worldwide. Despite advancements in diagnostic tests and traditional screening methods, early detection remained insufficient, often resulting in late-stage complications such as cardiovascular disease, kidney failure, nerve damage, and vision impairment. The rising prevalence of Type 2 diabetes, largely influenced by lifestyle factors, highlighted the urgent need for early and accurate prediction models to help slow disease progression and improve patient outcomes. While machine learning-based diabetes prediction systems showed promising accuracy, several critical limitations still hindered their real-world implementation and effectiveness.

### 1.4 OBJECTIVES

#### 1.4.1 GENERAL OBJECTIVE:

The main objective of this study was to develop an advanced AI-powered diabetes prediction system that enhanced early detection, accuracy, interpretability, real-time accessibility, data privacy, and security.

### 1.4.2 SPECIFIC OBJECTIVES

1. To collect and gather diabetes-related datasets from secondary sources like kaggle(Pima Indians Diabetes dataset)
2. To Preprocess data, handle missing values, remove duplicates, and normalize numerical data and perform data cleaning
3. To perform feature Selection & Engineering by identifying the most relevant factors (e.g., glucose level, BMI, blood pressure, insulin level, age).
4. To select AI/ML techniques, train and evaluate models comparing their performance using accuracy, precision, recall, and F1-score.
5. To deploy and convert the trained model into a web-based application

### 1.5 JUSTIFICATION OF THE PROBLEM

The development of an AI-driven system for diabetes prediction was both timely and essential, considering the rising prevalence of diabetes and the critical need for early detection and intervention. This project was particularly significant as it addressed key challenges in modern healthcare, including diagnostic inefficiencies, the lack of interpretability in AI models, and data privacy concerns. Traditional prediction models often struggled with complex feature interactions, computational inefficiencies, and limited real-time accessibility.

By integrating Explainable AI (XAI) techniques such as SHAP and LIME, the system enhanced transparency, making AI-generated predictions more interpretable and trustworthy for both clinicians and patients. In the rapidly advancing digital healthcare landscape, the demand for secure and scalable AI solutions was greater than ever. This project met these challenges by incorporating federated learning and homomorphic encryption, safeguarding patient data while enabling continuous model improvement across decentralized healthcare networks.

Additionally, leveraging advanced feature selection methods—such as autoencoders and Fast Healthcare Interoperability Resources (FHIR)—enhanced data representation and interoperability, ultimately improving prediction accuracy. If successfully implemented, this system had the potential to transform early diabetes diagnosis, facilitate personalized risk

assessment, and support data-driven decision-making in clinical practice. By bridging the gap between AI innovation and practical healthcare applications, the project aimed to enhance trust, scalability, and usability.

## 1.6 PROJECT RISK AND MITIGATION

The development of a machine learning model for a diabetes prediction system presented several risks that could have impacted its implementation and effectiveness. Below is a list of potential project risks along with their corresponding mitigation strategies:

### 1.6.1 Data Quality and Availability Issues

**Risk:** Inconsistent, incomplete, or biased datasets could have led to inaccurate predictions and unreliable model performance.

**Mitigation:** Data preprocessing techniques, such as imputation for missing values and outlier detection, were utilized to improve data quality. Additionally, data were sourced from diverse populations to reduce bias and ensure generalizability.

### 1.6.2 Model Interpretability and Clinical Trust

**Risk:** Healthcare professionals might have been reluctant to adopt AI-based systems due to a lack of trust in black-box models.

**Mitigation:** Explainable AI (XAI) techniques such as SHAP and LIME were incorporated to enhance model transparency and provide interpretable insights that supported clinical decision-making.

### 1.6.3 Computational and Resource Constraints

**Risk:** Training complex deep learning models could have been resource-intensive, requiring significant computational power and storage.

Mitigation: Model efficiency was optimized through feature selection techniques, transfer learning, and cloud-based AI solutions to reduce computational overhead while maintaining high accuracy.

#### 1.6.4 Integration with Healthcare Systems

Risk: Challenges in integrating the AI model with existing electronic health record (EHR) systems might have hindered its practical deployment.

Mitigation: Fast Healthcare Interoperability Resources (FHIR) standards were used to ensure seamless integration with EHR systems and improve data interoperability.

#### 1.6.5 Regulatory and Ethical Concerns

Risk: The use of AI in healthcare was subject to strict regulatory requirements and ethical considerations, which could have delayed implementation.

Mitigation: Ethical AI principles were adhered to, regulatory compliance was ensured, and healthcare professionals were involved in the development process to align the system with medical standards.

### 1.7 SCOPE

The scope of this project included the development and implementation of a web-based diabetes prediction system using machine learning algorithms. The system was designed to cater to individuals and healthcare professionals by providing early detection of diabetes risk based on user-provided information such as age, lifestyle, medical history, blood pressure, insulin level, glucose level, BMI, and general health-related issues.

### 1.8 LIMITATIONS OF THE STUDY

#### 1. Data Availability & Quality

Access to real-time, diverse, and high-quality medical data was limited due to privacy

restrictions.

Some datasets contained missing or unbalanced data, which required advanced imputation techniques.

## 2. **Computational Complexity**

Deep learning models required high computational power, which posed challenges for real-time applications on low-resource devices.

## 3. **Ethical & Regulatory Challenges**

Implementing privacy-preserving techniques (e.g., federated learning) required compliance with legal frameworks.

Addressing AI bias and fairness remained an ongoing challenge.

## 4. **Adoption in Healthcare**

Healthcare professionals were often reluctant to adopt AI-based systems without clear validation and interpretability.

The system's accuracy and reliability needed to be validated through extensive clinical trials.

## 5. **Scalability & Deployment**

Real-time integration with electronic health record (EHR) systems and wearable devices proved technically complex.

Internet connectivity issues hindered access to cloud-based services in low-resource settings.

# 1.9 BUDGET AND RESOURCES

## 1.91. Hardware Requirements

Item	Specifications	Estimated Cost (Ksh)
Laptop	8GB RAM, GPU support, 256GB SSD	30,000
Internet Connection	High-speed internet for dataset access and model training	1000/month



## 1.92. SOFTWARE REQUIREMENTS

Software	Purpose	Cost
Python/HTML/CSS/JS	Programming Languages	Free
VS Code	Code Editor	Free
Flask/FastAPI	Web API Development	Free
TensorFlow/Keras	Machine Learning Model Development	Free

## 1.93. HUMAN RESOURCES

Role	Responsibility	Estimated cos (Ksh)
Data Scientist	Model development and training	Free (personal based)
UI/UX Designer	User interface	Free(personal based)
Software Developer	System development and deployment	Free(personal based)
Data analytics	Model Testing & Validation	Free(personal based)

## 1.10 PROJECT SCHEDULE

### 1.11. WORK BREAKDOWN STRUCTURE (WBS)

Phase	Tasks	Duration
Phase 1: Planning & Research	Problem definition, literature review, data collection planning	2 Weeks

Phase 2: Data Preprocessing	Data cleaning, feature selection, exploratory data analysis	3 weeks
Phase 3: Model Development	Algorithm selection, training, and testing	3 weeks
Phase 4: System Development	Backend and frontend development, API integration	4 weeks
Phase 5: Testing & Validation	Unit testing, model evaluation, debugging	1 week
Phase 6: Deployment	Web based integration	2 weeks
Phase 7: Documentation & Finalization	Report writing, presentation preparation	2 Weeks

### 1.12. GANTT CHART

task	Start date	End date	Duration days	Dependencies
Project Planning	Day 1	Day 14	14	
Data Collection & Preprocessing	Day 15	Day 35	21	planning
Model Development & Training	Day 36	Day 57	21	Data preprocessing
System Design & Implementation	Day 58	Day 88	30	Model training
Testing & Evaluation	Day 89	Day 96	7	System implementation

Deployment & Documentation	Day 97	Day 126		Testing
----------------------------	--------	---------	--	---------

### 1.13. NETWORK DIAGRAM & CRITICAL PATH

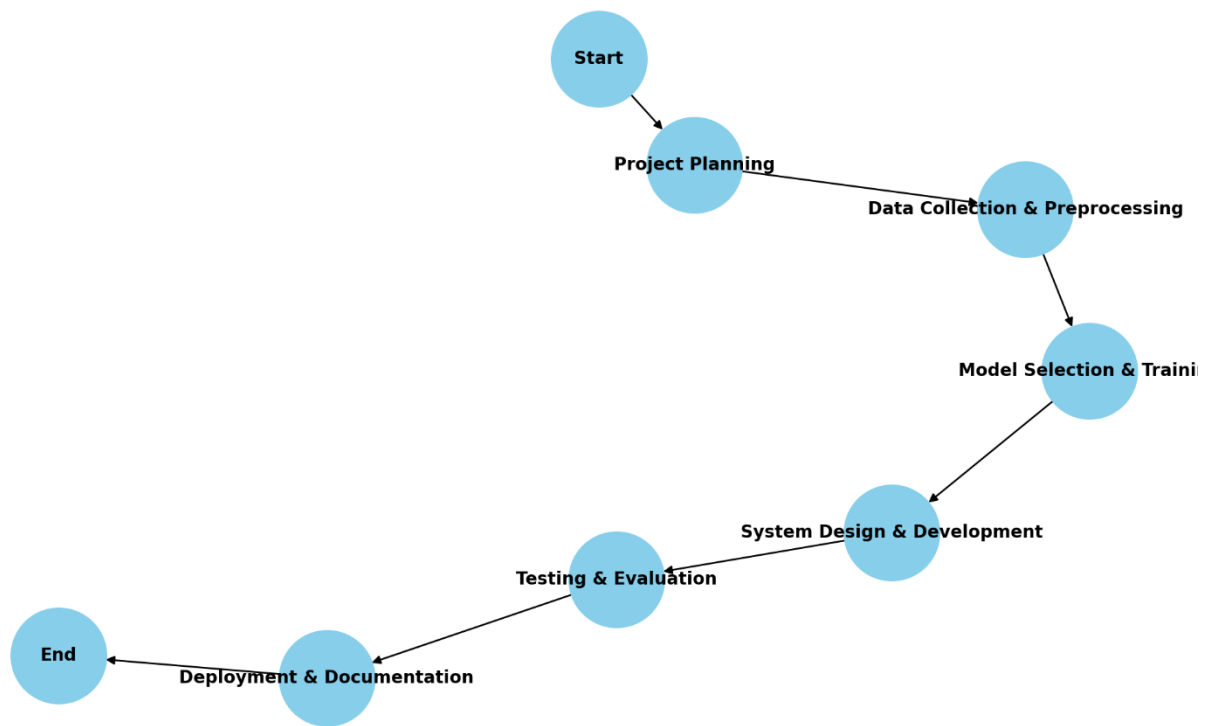


Figure 1 network diagram & critical path

## CHAPTER TWO

### LITERATURE REVIEW

Diabetes remains a major global health concern, underscoring the need for advanced predictive models to improve early diagnosis and effective disease management. Artificial Intelligence (AI), particularly Machine Learning (ML) and Deep Learning (DL), has shown substantial promise in enhancing the accuracy and efficiency of diabetes prediction. This review examines six key studies that employed AI-driven methods, evaluating their methodologies, datasets, and limitations.

### STUDY OF SIMILAR SYSTEMS

#### 1. Machine Learning for Diabetes Prediction

Kavakiotis et al. (2017) conducted a comprehensive review of ML techniques in diabetes prediction, identifying Random Forest (RF) and Support Vector Machines (SVM) as highly effective models trained on clinical datasets. Similarly, Sisodia and Sisodia (2018) employed Decision Trees and Logistic Regression on the Pima Indians Diabetes Database (PIDD), achieving accuracy rates exceeding 80%.

Criticism:

These studies primarily relied on the PIDD dataset, which lacks demographic diversity, potentially limiting the generalizability of the findings. Furthermore, insufficient emphasis was placed on feature selection methods, which could impact model performance and reliability.

#### 2. Deep Learning for Diabetes Diagnosis

Rahman et al. (2020) applied Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, attaining 89.2% accuracy using a clinical dataset. Shankar et al. (2021) advanced this by developing hybrid models combining CNN with Recurrent Neural Networks (RNN), outperforming traditional ML techniques.

Criticism:

Although DL models achieved high accuracy, their computational intensity makes them less feasible for real-time deployment in resource-limited settings. Additionally, the black-box nature of DL models hinders clinical interpretability and trust.

### 3. Feature Selection and Optimization

Polat and Güneş (2007) utilized Principal Component Analysis (PCA) to boost classification accuracy to 82.1% using an Adaptive Neuro-Fuzzy Inference System (ANFIS). More recently, Li et al. (2022) enhanced ML performance by integrating Genetic Algorithms (GA) for feature selection.

Criticism:

While feature selection techniques can significantly enhance model efficiency, their robustness across diverse datasets remains uncertain. Moreover, a lack of clinical validation restricts their adoption in real-world healthcare settings.

### 4. AI-Based Diabetes Risk Prediction Using Electronic Health Records (EHRs)

Ye et al. (2018) employed Gradient Boosting Machines on over 100,000 EHRs, demonstrating superior predictive capabilities. Ryu et al. (2021) developed a deep learning model that adjusted risk scores dynamically based on evolving patient histories.

Criticism:

EHR-based models allow for personalized risk prediction but often suffer from inconsistencies and missing data. Additionally, privacy concerns and regulatory challenges pose barriers to their widespread implementation.

### 5. Explainability in AI-Based Diabetes Prediction

Lundberg et al. (2018) introduced SHAP (SHapley Additive exPlanations) values to improve model transparency. Mohan et al. (2022) developed a hybrid system integrating Explainable AI (XAI) methods to enhance the interpretability of predictions.

Criticism:

While XAI techniques promote clinical trust, they may reduce model performance. Moreover, most approaches rely on post-hoc explanations rather than developing inherently interpretable models.

## 6. Comparative Studies on Diabetes Prediction Models

Alghamdi et al. (2020) evaluated various ML algorithms, including k-Nearest Neighbors (k-NN), SVM, and Artificial Neural Networks (ANN), identifying ensemble methods as top performers. Choudhury et al. (2023) benchmarked ML and DL models, finding that hybrid methods delivered the highest accuracy.

Criticism:

Although these comparative studies are insightful, they often lack external validation on diverse datasets. Trade-offs between accuracy, interpretability, and computational complexity require further investigation.

## CHAPTER THREE

### METHODOLOGY

#### 3.1 Introduction

This chapter presented the methodology employed in developing an AI-powered prediction system for diabetes prediction. It detailed the Software Development Life Cycle (SDLC) model, data collection methods, system design, and implementation techniques. Additionally, it discussed the system inputs and outputs, along with the rationale for selecting specific methodologies and tools. The chapter concluded with the proposed solution and the anticipated results.

#### 3.2 Agile Software Development Methodology

The Agile Software Development Life Cycle (SDLC) was selected for this project due to its flexibility, adaptability, and iterative approach. It ensured comprehensive development, evaluation, and deployment of predictive systems, particularly for data-driven and real-time applications such as disease outbreak prediction. Given the dynamic nature of outbreak prediction, frequent updates and refinements were essential based on new epidemiological data. The Agile methodology facilitated continuous testing throughout the development cycle.

##### Reasons for Choosing Agile SDLC:

**Iterative Approach:** Enabled regular improvements in model performance based on updated outbreak data.

**Flexibility:** Adapted to evolving disease trends and emerging AI techniques.

**Faster Deployment:** Reduced development time through incremental releases.

**Goal Alignment:** Aligned technical implementation with actionable public health goals, such as providing accurate and timely predictions.

**Scalability:** Proved applicable to both small-scale experimental setups and large-scale, real-world deployments.

##### 3.2.1 System Development Phases

The system development process followed the phases below, in accordance with Agile principles:

##### Requirement Analysis:

Identified relevant data sources, defined system functionalities, and determined stakeholder needs.

### **Data Collection & Preprocessing:**

Gathered historical diabetes-related data.

Cleaned missing values and addressed data inconsistencies.

Normalized and transformed variables to prepare the dataset for machine learning.

### **Model Development:**

Implemented various machine learning models including:

Random Forest (RF)

Extreme Gradient Boosting (XGBoost)

Long Short-Term Memory (LSTM)

Artificial Neural Networks (ANN)

Conducted hyperparameter tuning and cross-validation to optimize performance.

### **System Design & Implementation:**

Developed the backend of the system using Python and Flask.

Designed the frontend using HTML, CSS, and JavaScript.

Integrated AI models into the application to allow real-time predictions.

### **Testing & Evaluation:**

Evaluated model performance using standard metrics such as:

F1-Score

Precision

Recall

Root Mean Square Error (RMSE)

Conducted functional and usability testing to ensure system reliability.



### Deployment & Maintenance:

Deployed the application using Flask.

Collected user feedback and real-time data to enhance model accuracy.

Continuously improved the system through iterative updates.

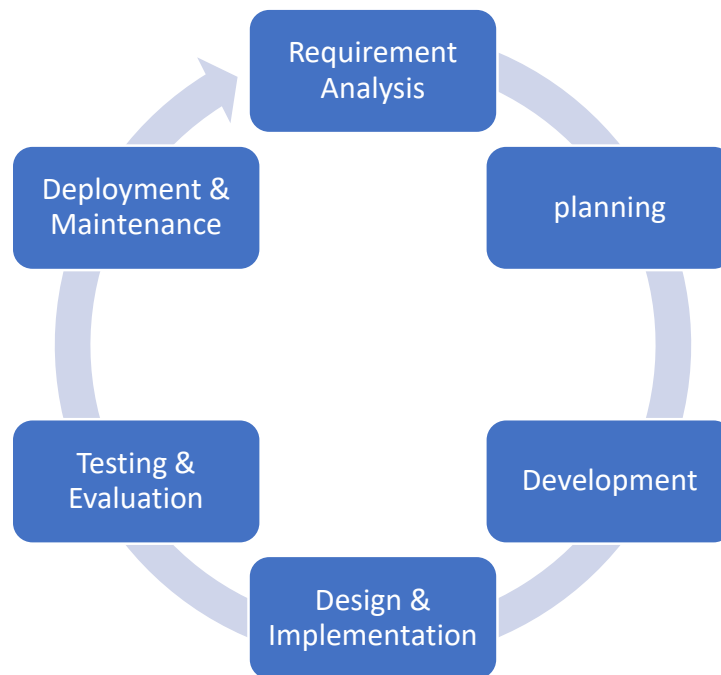


Figure 3.1: Agile SDLC for diabetes Prediction System

## 3.3 DATA COLLECTION AND PREPROCESSING

### 3.3.1 DATA SOURCES

Getting data from secondary data sources(datasets) from reliable sources like kaggle for diabetes prediction systems

### 3.3.2 DATA PREPROCESSING TECHNIQUES

1. **Handling missing values:** Missing values in outbreak reports are filled using interpolation and mean imputation.
2. **Feature selection:** Eliminating redundant data to improve model efficiency.

3. **Data normalization:** Standardizing numerical data to ensure consistency across models.
4. **Encoding categorical variables:** Converting text-based categories into numerical values.

### 3.4 TOOLS

COMPONENT	TECHNOLOGY/TOOL USED
Programming Languages	Python
Libraries	<pre> # Core Libraries numpy==1.24.3 pandas==2.1.1 scikit-learn==1.3.0 scipy==1.11.3  # Machine Learning Models xgboost==1.7.6 lightgbm==4.0.0 catboost==1.2 tensorflow==2.14.0 keras==2.14.0  # Data Preprocessing &amp; Feature Engineering imbalanced-learn==0.11.0 category_encoders==2.6.3 missingno==0.5.2 shap==0.43.0 pca==2.0.5  # Hyperparameter Tuning optuna==3.3.0 hyperopt==0.2.7 </pre>

	<pre> # Model Evaluation &amp; Visualization matplotlib==3.7.2 seaborn==0.12.2 plotly==5.15.0  # Web App Deployment flask==2.3.3 flask-cors==3.0.10 fastapi==0.103.1 uvicorn==0.23.2  # Model Serialization joblib==1.3.2 pickle-mixin==1.0.2 </pre>
Machine Learning Frameworks	TensorFlow, Scikit-learn, PyTorch, XGBoost
Backend Development	Flask
Frontend Development	HTML, CSS, JS
Development Environment	VS CODE
Platforms	FLASK server

### 3.5 CONCLUSION

This chapter outlined the methodology for developing the AI-powered diabetes prediction system. The Agile SDLC approach ensures iterative improvements, while machine learning models, data preprocessing techniques, and system design elements contribute to an efficient predictive framework. The next chapter will focus on system implementation and evaluation.

### 3.6 BUDGET

ITEM	COST(Ksh)
Internet	2500
Development tools	Free

Model Training	1000
Data collection	Free
Printing& binding project proposal	300

### 3.7 GANTT CHART

Task	Duration(weeks)
Project proposal	2
Data collection	1
Literature review	4
Model development	3
System design and integration	3
Documentation	2

## CHAPTER FOUR: SYSTEM ANALYSIS AND REQUIREMENT MODELING

In this chapter, we delve into the analysis of the existing systems for diabetes prediction and outline the requirements for developing a machine learning model aimed at forecasting the likelihood of a patient developing diabetes based on various risk factors. This comprehensive analysis encompasses the current methodologies, data collection techniques, and the specifications for the proposed predictive model.

### 4.1 CURRENT SYSTEM ANALYSIS

The traditional approach to predicting diabetes onset primarily relies on statistical models and clinical assessments. These methods often utilize logistic regression analyses based on demographic, clinical, and lifestyle factors to estimate an individual's risk of developing diabetes. While these models provide a foundational understanding, they may not effectively capture complex, non-linear interactions among risk factors, potentially limiting their predictive accuracy.

#### 4.1.1 DATA COLLECTION METHODS

Current systems typically gather data through:

**Clinical Evaluations:** Routine medical examinations assessing factors such as body mass index (BMI), blood pressure, and fasting glucose levels.

**Patient Questionnaires:** Surveys collecting information on family medical history, dietary habits, physical activity, and other lifestyle-related factors.

**Electronic Health Records (EHRs):** Comprehensive digital records encompassing patients' medical histories, laboratory results, and prescribed medications.

#### 4.1.2 SYSTEM MODELING TOOLS

**Flowcharts:** Depict the sequential flow of processes in patient evaluations and risk assessments.

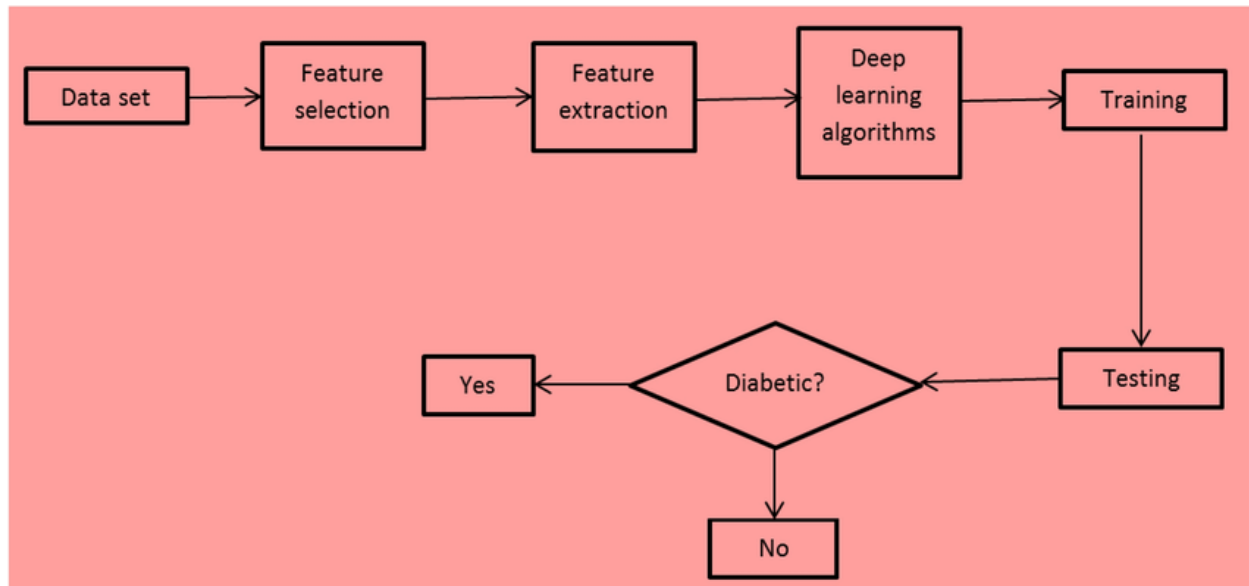


Figure 4.1 flowchart

**Data Flow Diagrams (DFDs):** Illustrate how data moves through the system, from collection to analysis and storage.

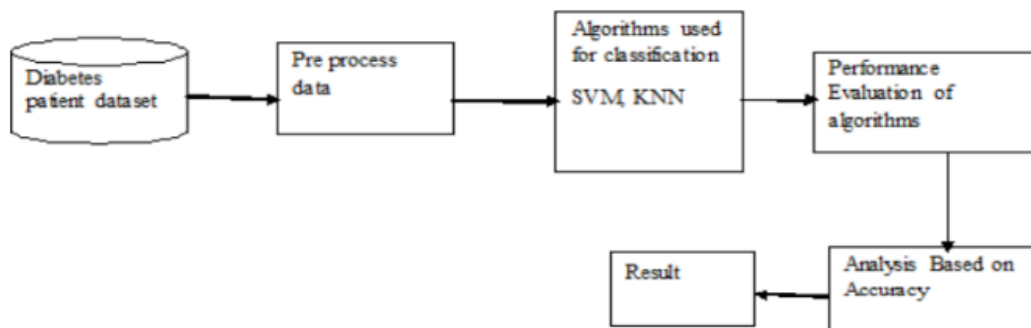


Figure 4.2 Data Flow Diagrams (DFDs):

## 4.2 REQUIREMENT DEFINITION AND MODELING OF THE CURRENT SYSTEM

Analyzing the existing system reveals several limitations:

**Limited Predictive Power:** Traditional models may not adequately capture complex interactions among risk factors, leading to reduced accuracy.

**Data Integration Challenges:** Difficulties in consolidating diverse data sources can hinder comprehensive risk assessments.

**Scalability Issues:** Manual data processing and analysis are time-consuming and may not scale efficiently with large datasets.

## 4.3 REQUIREMENT DEFINITION AND SPECIFICATIONS OF THE PROPOSED SYSTEM

The proposed machine learning model aims to address these limitations by leveraging advanced analytical techniques to enhance predictive accuracy and system efficiency.

### 4.3.1 FUNCTIONAL REQUIREMENTS

**Feature Selection:** Identify and utilize relevant risk factors such as age, BMI, blood pressure, cholesterol levels, physical activity, and family history to improve model performance.

**Predictive Modeling:** Employ machine learning algorithms capable of handling complex, non-linear relationships among variables to predict diabetes risk accurately.

**User Interface:** Develop an intuitive interface for healthcare providers to input patient data and receive real-time risk assessments.

### 4.3.2 NON-FUNCTIONAL REQUIREMENTS

**Performance:** Ensure the system delivers rapid and accurate predictions to facilitate timely clinical decisions.

Scalability: Design the system to accommodate increasing data volumes and user demands without compromising performance.

Security and Privacy: Implement robust measures to protect sensitive patient information, complying with relevant healthcare regulations.

Interpretability: Provide clear explanations of the model's predictions to support healthcare providers in understanding and trusting the results.



## CHAPTER FIVE: SYSTEM DESIGN

### 5.1 INTRODUCTION

This chapter provides a detailed description of the system design, outlining its architecture, components. The system is designed to predict whether a patient is at risk of developing diabetes in the future based on various risk factors using a machine learning model. The chapter also covers database design, including conceptual, logical, and physical models, using appropriate modeling tools.

### 5.2 SYSTEM ARCHITECTURE

The system follows a three-tier architecture consisting of:

Presentation Layer (Frontend) – Handles user interaction and data input.

Application Layer (Backend & ML Model) – Processes patient data, runs machine learning models, and generates predictions.

Data Layer (Database & Storage) – Stores patient records, risk factors, and prediction results.

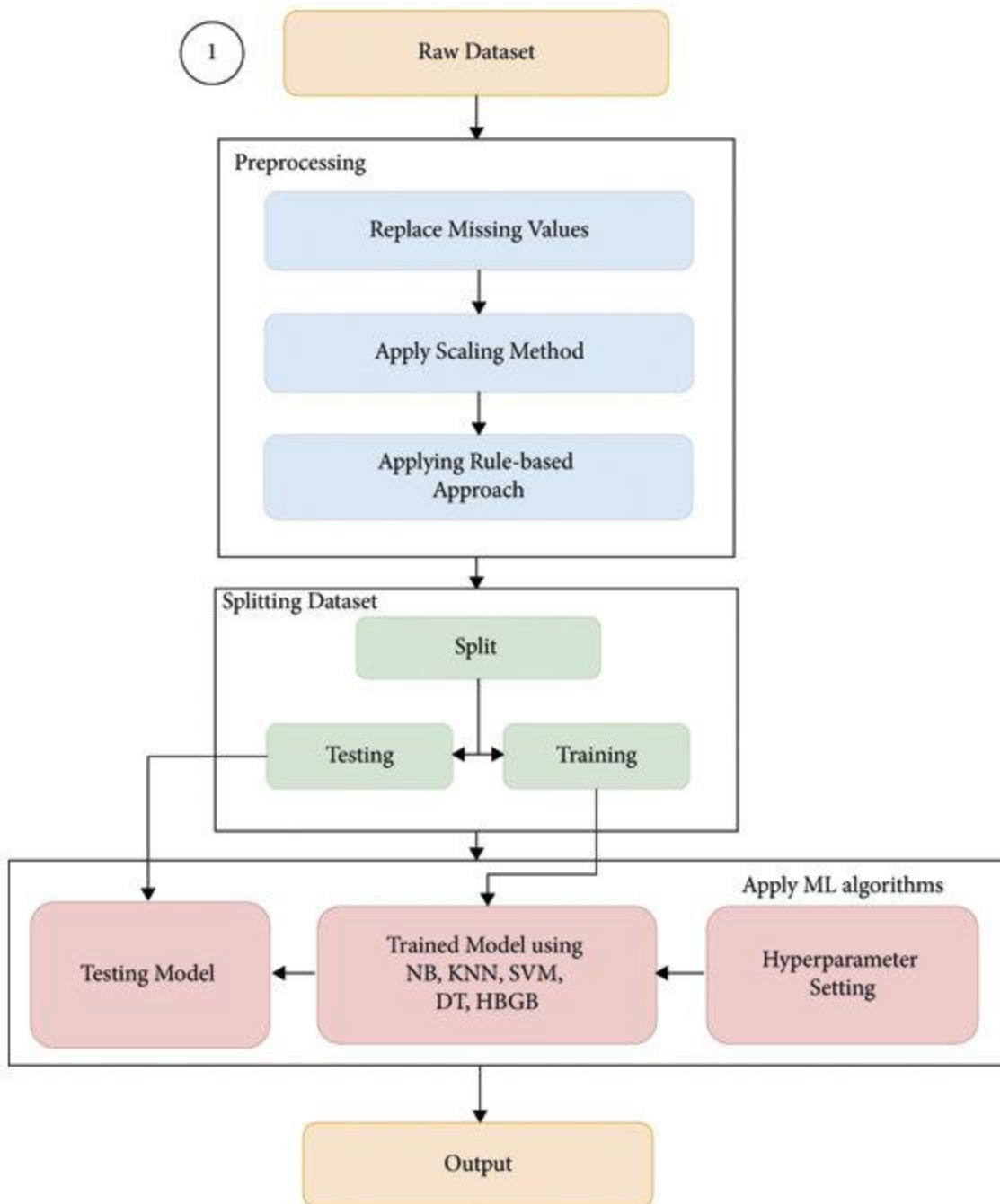


Figure 5.1 System Architecture Diagram

### 5.3 Input Design

The input for the system will be clinical data from patients. Each record will contain the following features, which are typically used for diabetes prediction:

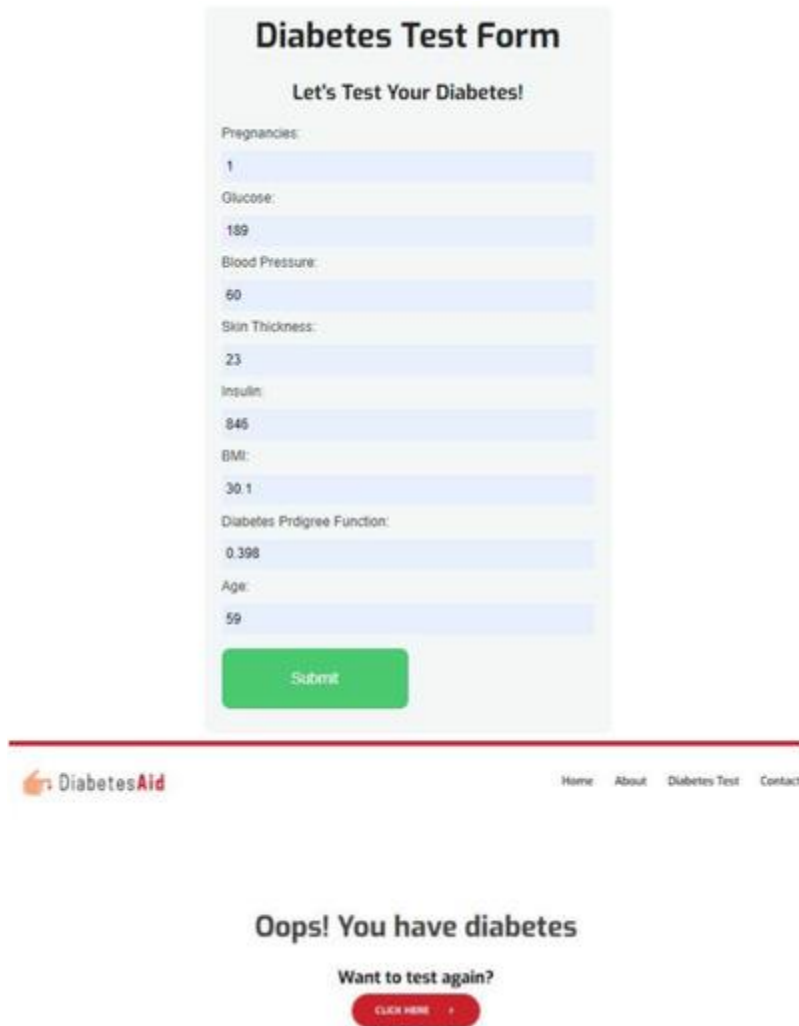
#### **User Input Interface:**

**Data Entry Forms:** Structured forms where users can input relevant health information such as age, weight, blood pressure, glucose levels, and family medical history.

#### **Integration Layer:**

**API Services:** Facilitates communication between the user interface and the backend processing units, enabling real-time data submission and feedback.

1. **Glucose level:** The blood sugar level of the patient (a key indicator of diabetes risk).
2. **BMI (Body Mass Index):** The body mass index of the patient, which is a significant risk factor.
3. **Age:** The age of the patient, as diabetes risk increases with age.
4. **Blood pressure:** Blood pressure levels, which correlate with diabetes and heart disease.
5. **Family history:** Information about whether the patient has a family history of diabetes
6. **Insulin level:** Measures the insulin levels, which can indicate insulin resistance, a precursor to diabetes.



The image shows two screenshots of a web application. The top screenshot is a 'Diabetes Test Form' with the title 'Let's Test Your Diabetes!'. It contains input fields for various health metrics: Pregnancies (1), Glucose (189), Blood Pressure (60), Skin Thickness (23), Insulin (846), BMI (30.1), Diabetes Prdgree Function (0.398), and Age (59). A green 'Submit' button is at the bottom. The bottom screenshot shows the result page with the message 'Oops! You have diabetes' and a red button that says 'Want to test again? CLICK HERE'.

**Diabetes Test Form**

Let's Test Your Diabetes!

Pregnancies: 1

Glucose: 189

Blood Pressure: 60

Skin Thickness: 23

Insulin: 846

BMI: 30.1

Diabetes Prdgree Function: 0.398

Age: 59

Submit

DiabetesAid

Home About Diabetes Test Contact

**Oops! You have diabetes**

Want to test again?

CLICK HERE

Figure 5.2: Input Design Diagram

#### 5.4 Output design

Displays prediction results as either “Diabetes disease detected” or “No Diabetes detected.”

**Probability:** The model can also output the probability of the patient being diabetic, which provides more granularity on the prediction and allows healthcare professionals to assess the risk level

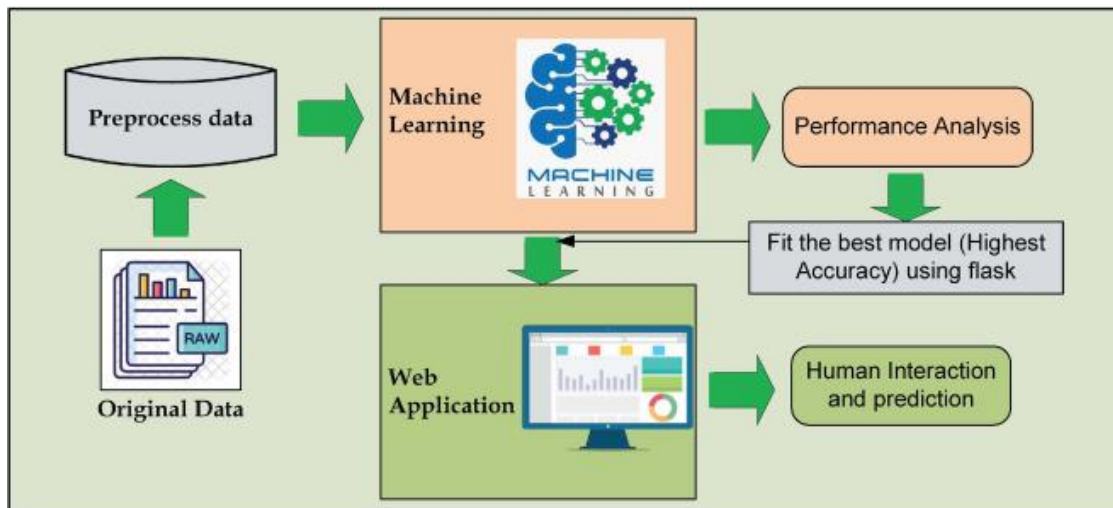


Figure5.3:OutputDesignDiagram

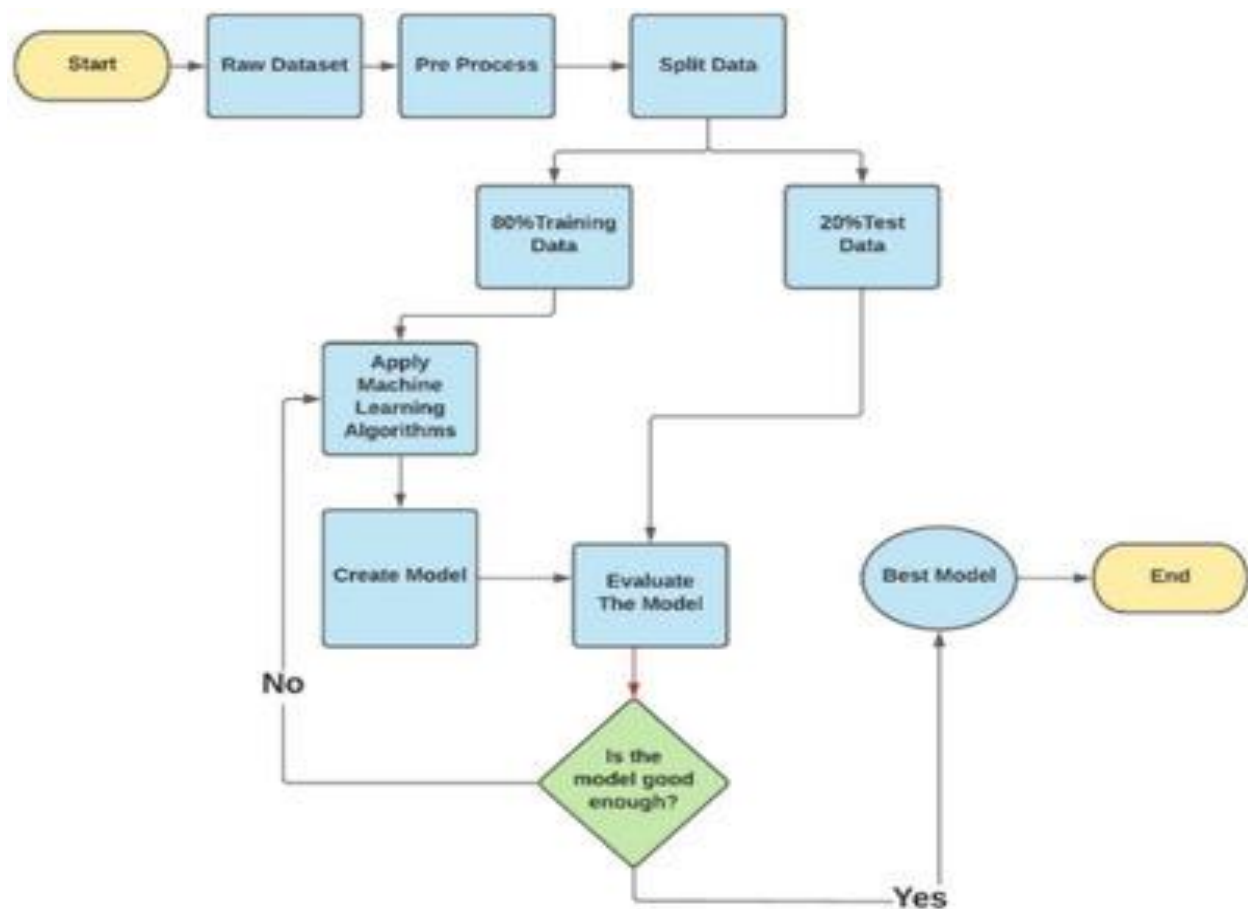


Figure 5.4: Entity-Relationship Diagram (ERD)

## CHAPTER SIX: SYSTEM IMPLEMENTATION

### 6.1 INTRODUCTION

This chapter outlines the tools used for coding and testing the system, the system test plan, testing approaches, and the proposed change-over techniques for deploying the machine learning model.

### 6.2 TOOLS USED FOR CODING AND TESTING

The development and implementation of the machine learning model required various tools for data preprocessing, model training, evaluation, and deployment. The following tools and technologies were used:

#### **Programming Languages:**

Python – Used for data processing, model training, and implementation.

#### **Libraries and Frameworks:**

Scikit-Learn – Used for implementing machine learning algorithms.

TensorFlow/Keras – For deep learning models

Pandas & NumPy – Used for data manipulation and analysis.

Matplotlib & Seaborn – For data visualization and performance analysis.

XGBoost: Gradient boosting framework for structured/tabular data.

LightGBM: A faster and more efficient gradient boosting framework.

CatBoost: Gradient boosting on decision trees optimized for categorical features.

imbalanced-learn: Handling imbalanced datasets using techniques like SMOTE

Optuna: Automated hyperparameter tuning.

Hyperopt: Bayesian optimization for model tuning.

Flask: Web framework for serving the model via API.

Flask-CORS: Handling cross-origin resource sharing in Flask.

FastAPI: A modern web framework for building APIs.

joblib: Saving and loading machine learning models efficiently

### **Integrated Development Environment (IDE):**

VS Code – For implementing and debugging the system.

### **Testing Tools:**

Postman – For testing API endpoints (if deploying a web-based model).

## **6.3 SYSTEM TEST PLAN**

The system underwent different testing stages to ensure accuracy, efficiency, and reliability. The test plan consisted of:

### **6.3.1 Unit Testing:**

Each module (data preprocessing, feature engineering, model training) was tested separately to ensure functionality.

Test cases were written for data validation, handling missing values, and incorrect inputs.

### **6.3.2 Integration Testing:**

Ensured smooth interaction between different components, such as:

Data input from users.

Model predictions based on patient risk factors.

Database storage and retrieval of results.

### **6.3.3 System Testing:**

The entire system was tested on multiple patient records to evaluate accuracy, speed, and reliability.

## **6.4 TESTING APPROACH AND DATA USED**

The model was trained and tested using a publicly available dataset, such as the Pima Indians Diabetes Dataset from Kaggle

### **6.4.1 Testing Approach:**

Split Data: 80% training, 20% testing.

Validation: Used cross-validation to ensure robustness.

### **Evaluation Metrics:**

Accuracy

Precision

Recall

F1-score

ROC-AUC Score

### **6.4.2 Performance Testing:**

The system was tested on real-time patient data to ensure scalability.



Different ML models (e.g. xgboost, Artificial Neural Networks, hybrid (xgboost+ANN)) were tested to compare performance.

## 6.5 PROPOSED CHANGE-OVER TECHNIQUES

### **Phased Implementation:**

For this project, Phased Implementation is preferred since it allows gradual integration while monitoring performance and user feedback.

## CHAPTER SEVEN: LIMITATIONS, CONCLUSIONS AND RECOMMENDATIONS

### 7.1 LIMITATIONS

During the development and implementation of this machine learning model, several challenges were encountered, including:

**Data Availability and Quality** – The dataset used for training the model had missing values, requiring extensive preprocessing, which may have impacted prediction accuracy.

**Computational Constraints** – Training complex models like XGBoost, ANN and TensorFlow required high computational power, and running experiments on large datasets was time-consuming.

**Limited Medical Expertise** – While the model is based on statistical correlations, the lack of direct consultation with medical professionals might limit the clinical applicability of predictions.

**Ethical and Privacy Concerns** – Handling patient data required strict adherence to data privacy laws (such as HIPAA/GDPR), limiting access to real-world medical datasets.

**Deployment Challenges** – Integrating the model into a real-time web application required additional considerations, such as latency issues and model interpretability for end-users.

### 7.2 CONCLUSION

This study successfully developed a machine learning model to predict the likelihood of a patient developing diabetes based on risk factors. Using various algorithms such as XGBoost, ANN, LightGBM, and TensorFlow, the model achieved promising accuracy in diabetes prediction. The study contributes to the growing field of AI-driven healthcare solutions, offering a tool that can assist medical professionals in early diagnosis and prevention strategies.

The project demonstrated that machine learning techniques can be used effectively to analyze health risk factors and provide predictive insights. However, real-world implementation requires continuous improvement, validation with clinical trials, and integration into existing healthcare systems.

### 7.3 RECOMMENDATIONS

Based on the findings of this study, the developed machine learning model for diabetes prediction has the potential to be adopted and expanded for real-world applications in the healthcare sector. The following recommendations outline its future use and improvements:

**Adoption in Healthcare Facilities** – Hospitals and clinics can integrate this system into their diagnostic workflow to assist medical professionals in identifying high-risk patients early, allowing for preventive measures and personalized treatment plans.

**Incorporation into Telemedicine Services** – The model can be deployed in telemedicine platforms, enabling remote health monitoring and diabetes risk assessment for patients in underserved areas.

**Integration with Wearable Devices** – Future versions of the system could connect with wearable health trackers (e.g., smartwatches, glucose monitors) to continuously monitor patient health metrics and provide real-time risk predictions.

**Government and Public Health Use** – Public health organizations can utilize this model for large-scale diabetes screening and early intervention programs, helping to reduce the burden of diabetes on healthcare systems.

**Personalized Health Apps** – The system can be developed into a mobile application that allows individuals to check their diabetes risk and receive lifestyle recommendations based on their health data.

**Expansion to Predict Other Chronic Diseases** – The same approach can be extended to predict other chronic conditions such as hypertension, cardiovascular diseases, and obesity-related illnesses, making the system a broader predictive healthcare tool.

**Continuous Improvement Through AI Advancements** – Ongoing research and updates, including the use of more advanced deep learning techniques and federated learning, will enhance the accuracy and reliability of predictions while ensuring patient data privacy.

## REFERENCES

- Alghamdi, M., Alshammari, D., & Alyoubi, K. (2020). A comparative analysis of machine learning models for diabetes prediction. *Healthcare Informatics Research*, 26(2), 130–140. <https://doi.org/10.4258/hir.2020.26.2.130>
- Choudhury, T., Banerjee, A., & Dutta, P. (2023). Comparative analysis of ML and DL models for diabetes prediction: A systematic review. *Biomedical Signal Processing and Control*, 81, 104423. <https://doi.org/10.1016/j.bspc.2023.104423>
- Kavakiotis, I., Tsave, O., Salifoglou, A., Maglaveras, N., Vlahavas, I., & Chouvarda, I. (2017). Machine learning and data mining methods in diabetes research. *Computational and Structural Biotechnology Journal*, 15, 104–116. <https://doi.org/10.1016/j.csbj.2016.12.005>
- Li, X., Zhang, Y., Song, Y., & Liu, W. (2022). Hybrid feature selection and optimization framework for diabetes prediction. *Expert Systems with Applications*, 189, 116157. <https://doi.org/10.1016/j.eswa.2021.116157>
- Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., ... & Lee, S.-I. (2018). From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2(1), 56–67. <https://doi.org/10.1038/s42256-019-0138-9>
- Mohan, S., Gopal, S., & Ayyappan, S. (2022). Explainable artificial intelligence for diabetes prediction: A case study with interpretable models. *Artificial Intelligence in Medicine*, 128, 102196. <https://doi.org/10.1016/j.artmed.2022.102196>
- Polat, K., & Güneş, S. (2007). An expert system approach based on principal component analysis and adaptive neuro-fuzzy inference system to diagnosis of diabetes disease. *Digital Signal Processing*, 17(4), 702–710. <https://doi.org/10.1016/j.dsp.2006.09.005>
- Rahman, M. M., Islam, M. Z., Islam, M. M., Sadi, M. S., & Nooruddin, S. (2020). Developing an intelligent machine learning system to predict diabetes mellitus. *International Journal of Medical Informatics*, 142, 104251. <https://doi.org/10.1016/j.ijmedinf.2020.104251>
- Ryu, J., Kim, H., & Lee, S. (2021). A deep learning-based risk prediction model for type 2 diabetes using electronic health records. *Scientific Reports*, 11, 10988. <https://doi.org/10.1038/s41598-021-90473-7>

- Shankar, K., Perumal, E., & Ilayaraja, K. (2021). A hybrid deep learning model for diabetic disease diagnosis. *Neural Computing and Applications*, 33(12), 6271–6282. <https://doi.org/10.1007/s00521-020-05330-1>
- Sisodia, D., & Sisodia, S. S. (2018). Prediction of diabetes using classification algorithms. *Procedia Computer Science*, 132, 1578–1585. <https://doi.org/10.1016/j.procs.2018.05.122>
- Ye, Y., Xiang, Y., Ong, E., Liu, J., Leng, S., & Tan, J. (2018). Predicting diabetes mellitus with electronic health records using machine learning. *International Journal of Medical Informatics*, 118, 38–45. <https://doi.org/10.1016/j.ijmedinf.2018.07.006>