**GROUP ID: GE12**

**A PROJECT REPORT ON**

**Optimizing Heart Failure Risk Prediction with Ensemble Learning Technique**

SUBMITTED TO

THE PIMPRI CHINCHWAD COLLEGE OF ENGINEERING

AN AUTONOMOUS INSTITUTE, PUNE

IN THE FULFILLMENT OF THE REQUIREMENTS

FOR THE AWARD OF THE DEGREE

OF

**BACHELOR OF TECHNOLOGY**

**COMPUTER ENGINEERING (REGIONAL LANGUAGE)**

##### SUBMITTED BY

ABHIJIT MOTEKAR 121B1D051

GAYATRI JOSHI 121B1D047

ADITYA JAISWAL 121B1D013

SANKET DESHMUKH 121B1D002



## DEPARTMENT OF COMPUTER ENGINEERING (REGIONAL LANGUAGE)

## PCET’S PIMPRI CHINCHWAD COLLEGE OF ENGINEERING

**Sector No. 26, Pradhikaran, Nigdi, Pimpri-Chinchwad, PUNE 411044**

**2024-25**



**CERTIFICATE**

This is to Certify That the Project Report Entitles

**“Optimizing Heart Failure Risk Prediction with Ensemble Learning Technique”**

Submitted by

|  |  |
| --- | --- |
| **ABHIJIT MOTEKAR** | **121B1D051** |
| **GAYATRI JOSHI** | **121B1D047** |
| **ADITYA JAISWAL** | **121B1D013** |
| **SANKET DESHMUKH** | **121B1D002** |

is a Bonafide student of this institute and the work has been carried out by him/her under the supervision of **Prof. Rohini Y. Sarode** and it is approved for the partial fulfillment of the requirement of Pimpri Chinchwad College of Engineering an autonomous institute, for the award of the B. Tech. degree in Computer Engineering (Regional Language).

**(Prof. Rohini Y. Sarode)**  **(** **Prof. Dr. Rachana Y. Patil)** Guide Head

Department of Computer Engineering Department of Computer Engineering

(Regional Language) (Regional Language)

**(Prof. Dr. G.N. Kulkarni)**

Director,

Pimpri Chinchwad College of Engineering Pune – 44

Place: Pune

Date:

**ACKNOWLEDGEMENT**

We express our sincere thanks to our **Guide** **Prof.** **Rohini Y. Sarode** for her constant encouragement and support throughout our project, especially for the useful suggestions given during the course of the project and having laid down the foundation for the success of this work.

We would also like to thank our **Project Coordinator, Prof. Rohini Y. Sarode** for her assistance, genuine support and guidance from early stages of the project. We would like to thank **Prof. Dr. Rachana Y. Patil,** **Head of Computer Engineering (Regional Language) Department** for her unwavering support during the entire course of this project work. We are very grateful to our **Director, Prof. Dr. G.N. Kulkarni** for providing us with an environment to complete our project successfully. We also thank all the staff members of our college and technicians for their help in making this project a success.

We also thank all the web committees for enriching us with their immense knowledge. Finally, we take this opportunity to extend our deep appreciation to our family and friends, for all that they meant to us during the crucial times of the completion of our project.

|  |
| --- |
| **ABHIJIT MOTEKAR** |
| **GAYATRI JOSHI** |
| **ADITYA JAISWAL** |
| **SANKET DESHMUKH** |

**ABSTRACT**

The healthcare industry is increasingly embracing machine learning algorithms to accelerate early identification and diagnosis of diseases. However, the treatment of cardiovascular diseases often encounters obstacles due to variation in the level of knowledge and skill of healthcare providers, which can result in flawed decision making and suboptimal patient outcomes. To address this, machine learning and data mining techniques are increasingly being used for the prediction of heart disease, enabling automated diagnosis in hospitals. Diagnostic precision and patient outcomes are significantly improved using algorithms such as nave bayes, K-Nearest Neighbor (KNN), decision trees, and Artificial Neural Networks (ANN) by examining various health metrics of individuals. Enhancing the precision of predicting heart disease via ensemble learning technique is main objective research.

The study employs a variety of classification algorithms, including decision tree, support vector machine, random forest, and XGBoost, to assess the effectiveness of the model. Performance evaluation is carried out using various metrics, such as accuracy, f1 score, precision, sensitivity, and specificity. Ensemble methods like stacking, boosting, and bagging are applied to enhance the classifiers’ performance. The proposed methodology involves data preprocessing (removing null values and handling categorical values), hyperparameter tuning, and applying ensemble classifiers.

Experimental results demonstrate that ensemble approaches outperform individual classifiers in predicting heart disease risk. Among the ensemble classifiers, bagging achieved the highest accuracy of 88. 58%, significantly enhancing the prediction performance. The research demonstrates the efficacy of combining machine learning algorithms, ensemble approaches, and feature selection methods to accurately predict heart disease. This approach provides valuable support to medical professionals in making conversant decisions regarding patient health.

***Keywords***—heart disease prediction, ensemble learning, machine learning, data preprocessing, bagging

**TABLE OF CONTENTS**

# 

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Sr. No. | | Title of Chapter | | | | | Page No. |
| 01 | | **Introduction** | | | | | 1 |
|  | 1.1 | Overview | | | | | 1 |
|  | 1.2 | Motivation | | | | | 2 |
|  | 1.3 | Problem Statement and Objectives | | | | | 2 |
|  | 1.4 | Scope of the work | | | | | 3 |
|  |  |  | | | | |  |
| 02 | | Literature Survey | | | | | 4 |
|  | 2.1 | Literature Review | | | | | 4 |
|  | 2.2 | Gap Identification / Common findings from the literature | | | | | 6 |
|  |  |  | | | | |  |
| 03 | | Software Requirements Specification | | | | | 7 |
|  | 3.1 | Functional Requirements | | | | | 7 |
|  | 3.4 | Nonfunctional Requirements | | | | | 7 |
|  | 3.5 | System Requirements | | | | | 8 |
|  | | 3.5.1 | Software Requirements | | | | 8 |
|  | | 3.5.2 | Hardware Requirements | | | | 8 |
| 04 | | Proposed Methodology | | | | | 9 |
|  | 4.1 | Proposed System Architecture | | | | | 9 |
|  | 4.2 | Dataset/ Database design | | | | | 10 |
|  | 4.3 | Data preprocessing | | | | | 10 |
|  | 4.4 | Experimental Setup | | | | | 10 |
|  | 4.5 | Algorithm Used | | | | | 11 |
|  | | 4.5.1 | | | | Individual Algorithm | 11 |
|  | | 4.5.2 | | | | Ensemble Learning Technique | 12 |
|  | | 4.5.3 | | | | Evaluation Process Used | 13 |
|  | 4.6 | Complexity of Project | | | | | 14 |
|  | | 4.6.1 | | | Computational Complexity | | 14 |
|  | | 4.6.2 | | | Algorithmic Complexity | | 14 |
|  | | 4.6.3 | | | Resource Complexity | | 14 |
|  | 4.7 | SDLC Model to be applied | | | | | 15 |
|  | 4.8 | UML Diagrams | | | | | 16 |
| 05 | | Project Plan | | | | | 20 |
|  | 5.1 | Project Cost Estimation | | | | | 20 |
|  |  | 5.1.1 | | | | Performance Cost Estimation | 20 |
|  |  | 5.1.2 | | | | Sustainability Assessment | 22 |
|  |  | 5.1.3 | | | | Complexity Assessment | 22 |
|  | 5.2 | Risk Management | | | | | 23 |
|  |  | 5.2.1 | | | | Risk Identification | 23 |
|  |  | 5.2.2 | | | | Risk Analysis | 23 |
|  |  | 5.2.3 | | | | Overview of Risk Mitigation, Monitoring, Management | 24 |
| 06 | | Software Testing | | | | | 25 |
|  | 6.1 | Type of Testing | | | | | 25 |
|  | 6.2 | Test cases & Test Results | | | | | 26 |
| 07 | | Results & Discussion | | | | | 27 |
|  | 7.1 | Result Analysis | | | | | 27 |
|  | 7.2 | User Interface and Implementation Screenshots | | | | | 32 |
|  |  | 7.2.1 | | Code Snippets | | | 32 |
|  |  | 7.2.2 | | Frontend Design | | | 33 |
|  |  | 7.2.3 | | Responsive Design | | | 36 |
| 08 | | Contribution to Sustainable Development Goals | | | | | 37 |
|  | 8.1 | Introduction to SDGs | | | | | 37 |
|  | 8.2 | Mapping of the Project to Relevant SDGs | | | | | 37 |
| 09 | | Conclusion & Future Scope | | | | | 39 |
|  | 9.1 | Conclusions | | | | | 39 |
|  | 9.2 | Future Scope | | | | | 39 |
|  |  | **Appendix A: Details of Paper application** | | | | | 40 |
|  |  | **Appendix B: Plagiarism report of the project report** | | | | | 41 |
|  | | **References** | | | | | 42 |

**LIST OF ABBREVIATIONS**

|  |  |
| --- | --- |
| **ABBREVIATION** | **ILLUSTRATION** |
| CVD | Cardiovascular Disease |
| KNN | K Nearest Neighbor |
| SVM | Support Vector Machine |
| ANN | Artificial Neural Network |
| ML | Machine Learning |
| AI | Artificial Intelligence |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURE** | **ILLUSTRATION** | **PAGE NO.** |
| 1 | Architecture Diagram | 9 |
| 2 | Dataset Design | 10 |
| 3 | Agile Model | 15 |
| 4 | Class Diagram | 16 |
| 5 | Gantt Chart | 17 |
| 6 | Sequence Diagram | 17 |
| 7 | Component Diagram | 18 |
| 8 | Deployment Diagram | 18 |
| 9 | Package Diagram | 19 |
| 10 | Confusion Matrix for Random Forest Model | 28 |
| 11 | ROC AUC Curve of Random Forest Classifier | 29 |
| 12 | Confusion Matrix for XGBoost Model | 29 |
| 13 | Confusion Matrix for Decision Tree Model | 30 |
| 14 | Confusion Matrix for SVM Model | 30 |
| 15 | Confusion Matrix for Bagging Technique | 31 |
| 16 | ROC AUC Curve of Bagging Technique | 31 |
| 17 | Code Snippet 1 | 32 |
| 18 | Code Snippet 2 | 32 |
| 19 | Homepage | 33 |
| 20 | About and Analytics | 33 |
| 21 | Disclaimer | 34 |
| 22 | User Input | 34 |
| 23 | Result | 35 |
| 24 | Chatbot | 35 |
| 25 | Responsive Design | 36 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **TABLE** | **ILLUSTRATION** | **PAGE NO.** |
| 1 | Dataset Design | 10 |
| 2 | Hardware Cost Estimation | 20 |
| 3 | Software and Licensing Cost | 21 |
| 4 | Operational Cost Estimation | 21 |
| 5 | Testcase Table | 26 |
| 6 | Performance Measure of Individual Algorithm | 27 |
| 7 | Performance Measure of Classifier using Ensemble Technique | 27 |

**CHAPTER 1 INTRODUCTION**

* 1. **OVERVIEW**

The project focuses on improving heart disease prediction using ensemble learning techniques, a powerful approach in machine learning that combines multiple models to enhance accuracy and robustness. With cardiovascular diseases being a leading cause of death worldwide, early and accurate detection is critical for timely intervention. Traditional machine learning models such as Decision Tree, Random Forest, Support Vector Machine (SVM), and XGBoost have shown promising results in medical diagnosis. However, individual models often suffer from limitations in generalizability and accuracy. To overcome these challenges, this research applies ensemble techniques, including Bagging, Boosting, and Stacking, to optimize predictive performance.

The dataset used for this study is the Heart Failure Prediction Dataset from Kaggle, consisting of 918 patient records with 12 key features such as age, cholesterol levels, blood pressure, and chest pain type. Preprocessing steps involved handling missing values, encoding categorical variables, and normalizing numerical features. The dataset was then split into 80% training and 20% testing to ensure an unbiased evaluation of the models. Various classifiers were trained and compared, with the ensemble models outperforming the standalone classifiers. Among the ensemble methods, Bagging demonstrated the highest accuracy of 88.58%, proving to be the most effective approach.

The study evaluates model performance using standard classification metrics, including accuracy, precision, recall, F1-score, and ROC-AUC curves. The results highlight that ensemble learning significantly improves prediction accuracy compared to individual models, with Bagging achieving an AUC score of 0.94, indicating strong discriminative ability. Additionally, confusion matrices show that the ensemble models reduced false positive and false negative rates, making the predictions more reliable.

In conclusion, this research demonstrates that ensemble learning techniques can greatly enhance heart disease prediction accuracy, aiding healthcare professionals in making informed decisions. The study lays the groundwork for future improvements, such as integrating deep learning models or real-time deployment in clinical settings. With further optimization, this approach has the potential to be a valuable tool for early heart disease detection, reducing the burden on healthcare systems and saving lives.

* 1. **MOTIVATION**

Cardiovascular diseases (CVDs) are a leading cause of mortality worldwide, accounting for millions of deaths each year. Early detection of heart disease is crucial for timely medical intervention, which can significantly improve patient outcomes and reduce the strain on healthcare systems. However, traditional diagnostic methods often rely on the expertise of healthcare professionals, leading to variability in diagnosis and potential misclassification. This creates a need for automated, data-driven, and highly accurate prediction models that can assist doctors in making informed decisions.

With the advancements in machine learning (ML) and artificial intelligence (AI)**,** predictive analytics has emerged as a promising tool for medical diagnosis. However, individual ML models often struggle with accuracy and generalizability due to noisy data and complex patterns. This challenge motivated us to explore ensemble learning techniques, which combine multiple models to enhance prediction reliability. Techniques like Bagging, Boosting, and Stacking help in reducing errors, improving stability, and increasing overall accuracy compared to standalone classifiers.

The primary motivation behind this project is to develop a highly accurate and reliable heart disease prediction model using ensemble learning. By leveraging diverse machine learning approaches, we aim to minimize false diagnoses and provide a decision-support system for medical professionals. Our goal is to contribute to the healthcare industry by demonstrating that ensemble learning can enhance diagnostic precision, ultimately saving lives through early intervention**.**

* 1. **PROBLEM STATEMENT AND OBJECTIVE**

**1.3.1 Problem Statement**

Cardiovascular diseases (CVDs) are a leading cause of mortality, making early and accurate detection crucial. Traditional diagnosis relies on physician expertise, leading to variability and potential misdiagnosis. While machine learning improves prediction, individual models often lack accuracy and robustness. Our research paper, "Optimizing Heart Failure Risk Prediction with Ensemble Learning Technique" addresses this by using ensemble learning to enhance predictive performance, ensuring a more reliable and efficient heart disease diagnosis system.

**1.3.2 Objective**

Overall Objectives of our Project is going to be:

* To enhance heart disease prediction through an ensemble learning model.
* To implement and optimize ensemble learning algorithms—Bagging, Stacking, and AdaBoost—using appropriate hyperparameter tuning techniques such as Grid Search to maximize predictive accuracy.
* To compare the predictive performance of Bagging against traditional models such as Random Forest, Decision Tree, SVM, and XGBoost through statistical analysis.
* To gauge the potency of the model using different evaluation metrics, including accuracy, precision, recall, and F1-score.
  1. **SCOPE OF THE WORK**

This study improves heart disease prediction using ensemble learning **(**Bagging, Boosting, Stacking**)** for higher accuracy than traditional models. It covers data preprocessing, model training, and hyperparameter tuning**,** evaluated through accuracy, precision, recall, and F1-score. The findings support AI-driven healthcare, with potential for deep learning integration and real-time clinical use.

**CHAPTER 2 LITERATURE SURVEY**

**2.1 LITERATURE REVIEW**

Several studies have explored the use of machine learning for heart disease prediction, emphasizing the effectiveness of ensemble learning techniques. Shital Patil and Surendra Bhosale (2023) [1] analyzed various classification models, including Decision Tree, Logistic Regression, Random Forest, and AdaBoost, achieving 98.38% accuracy using majority voting and feature selection via the Chi-square test. Similarly, Muhammad Affan Alim et al. (2020) [2] employed Bagging, AdaBoost, and Random Forest on multiple datasets, including the Cleveland Clinic Foundation (CCF) dataset, improving accuracy through particle swarm optimization-based feature selection.

Another study by Salam Ismaeel et al. (2015) [3] introduced the Extreme Learning Machine (ELM) technique for heart disease diagnosis, utilizing key patient factors such as age, gender, blood pressure, and cholesterol levels, achieving an 80% accuracy rate. Further, Md Razu Ahmed et al. (2018) [4] developed a cloud-based four-tier architecture incorporating Artificial Neural Networks (ANNs), demonstrating significant accuracy improvements through ten-fold cross-validation. Additionally, Aditi Gavhane et al. (2018) [5] compared machine learning and deep learning techniques, using 14 critical features for classification. Their findings showed that deep learning outperformed traditional ML models, achieving 94.2% accuracy in heart disease prediction.

Further research has demonstrated the advantages of ensemble learning techniques in improving heart disease prediction accuracy. Archana Singh and Rakesh Kumar [6] explored the effectiveness of Bagging, Boosting, and Stacking, concluding that ensemble methods consistently outperformed individual classifiers. Similarly, K. Rohit Chowdary et al. (2022) [7] applied ensemble learning to early heart disease prediction, using feature selection techniques and testing models on the Kaggle and UCI datasets, achieving high accuracy across datasets.

In another study, Senthilkumar Mohan et al. (2019) [8] investigated hybrid models that combined machine learning and deep learning techniques, highlighting their potential to enhance prediction accuracy and reliability. Ivan Miguel Pires et al. (2020) [9] evaluated multiple classifiers, including neural networks, decision trees, k-NN, and SVM, achieving an accuracy of 87.69% for heart disease prediction. Lastly, Ahmad Ayid and Huseyin Polat (2023) [10] implemented the JellyfishOptimization Algorithm for feature selection, achieving an impressive 98.47% accuracy with an SVM classifier using the Cleveland dataset.

Several more studies have explored advanced techniques to enhance heart disease prediction. Hosam El-Sofany, Belgacem Bouallegue, and Yasser M. Abd El-Latif (2024) [11] focused on XGBoost classification, incorporating explainable AI (SHAP) and feature selection techniques like Chi-square, ANOVA, and mutual information, achieving 97.57% accuracy. Similarly, R. Indrakumari, T. Poongodi, and Soumya Ranjan Jena (2020) [12] applied K-means clustering for exploratory data analysis, emphasizing the role of data analytics in healthcare decision-making.

Vanisree K. and Singaraju (2011) [13] developed a neural network-based Decision Support System (DSS) for diagnosing Congenital Heart Disease (CHD) in children, achieving 90% accuracy using a Backpropagation Neural Network. Meanwhile, Mohsen Dorraki et al. (2024) [14] combined machine learning and mental health data from the UK Biobank, demonstrating that psychological factors enhance cardiovascular disease (CVD) prediction, with an accuracy of over 85%. Chaitrali Dangare and Sulbha Apte (2012) [15] further improved heart disease prediction by incorporating Neural Networks, Decision Trees, and Naïve Bayes, achieving 100% accuracy by including additional risk factors like obesity and smoking.

Further research has demonstrated the potential of ensemble learning in improving heart disease prediction. B. Srinivasa Rao (2021) [16] developed an ensemble model combining Naïve Bayes and Random Forest, achieving 89% accuracy, which was higher than individual classifiers. Achyut Tiwari, Aryan Chugh, and Aman Sharma (2022) [17] introduced a stacked ensemble learning framework combining Extra Trees, Random Forest, and XGBoost, reaching an accuracy of 92.34%, outperforming existing models.

Abdullah Alqahtani et al. (2022) [18] applied ensemble learning for cardiovascular disease detection, using a voting-based model that achieved 88.7% precision, surpassing individual ML and deep learning models. Similarly, Ghalia A. Alshehri and Hajar M. Alharbi (2023) [19] tested an ensemble learning approach combining AdaBoost, SVM, Decision Tree, and Random Forest across three datasets, achieving the highest accuracy (91%) with the Z-Alizadeh Sani dataset.

These studies highlight the effectiveness of ensemble learning techniques in improving predictive accuracy and model robustness. They demonstrate that combining multiple models consistently outperforms single algorithms, making them valuable for real-world healthcare applications.

* 1. **GAP IDENTIFICATION**

Optimized Ensemble Learning for Higher Accuracy – While studies confirm that Bagging, Boosting, and Stacking outperform individual models, few optimize them using hyperparameter tuning (e.g., Grid Search) to achieve maximum predictive accuracy. Our research addresses this gap by fine-tuning ensemble methods for improved performance.

**CHAPTER 3 SOFTWARE REQUIREMENT SPECIFICATION**

**3.1 FUNCTIONAL REQUIREMENTS**

* Data Preprocessing and Feature Selection – Handling missing values, encoding categorical variables, normalizing data, and selecting the most relevant features to improve model efficiency.
* Ensemble Learning Implementation – Developing and optimizing Bagging, Stacking, and AdaBoost models to enhance heart disease prediction accuracy.
* Model Performance Evaluation – Assessing models using metrics such as accuracy, precision, recall, F1-score, and ROC-AUC for effective comparison.
* Hyperparameter Tuning – Utilizing techniques like Grid Search to optimize ensemble models for better predictive performance.
* AI-Powered Chatbot for Health Recommendations – Integrating a chatbot that provides users with personalized health advice based on their risk predictions.
* User Interface for Predictions – Developing a simple and interactive UI where users can input health parameters and receive heart disease risk predictions.
* Data Visualization – Displaying model performance metrics, confusion matrices, and feature importance using graphs and charts for better analysis.

**3.2 NONFUNCTIONAL REQUIREMENT**

* User-Friendly Web Interface – The system will have an intuitive and accessible UI, allowing users to easily input health parameters and receive predictions.
* Fast and Optimized Model Inference – The model will provide quick and accurate predictions by optimizing computational efficiency and reducing latency.
* Security and Data Privacy – User health data will be securely processed and stored, ensuring compliance with privacy regulations.
* Cross-Platform Accessibility – The web application will be accessible on various devices (desktop, tablet, and mobile) for user convenience.
* Reliability and Accuracy – The system will maintain high reliability, ensuring that predictions and chatbot recommendations are consistent and accurate.

**3**.**3 SYSTEM REQUIREMENT**

**3.3.1 Software Requirement**

1. Code Editor & Development Environment

* VS Code – For writing and managing code efficiently.
* Google Colab – For developing and testing machine learning models in a cloud-based environment.

1. Programming Languages & Libraries

* Python – For implementing machine learning models.
* HTML, CSS, JavaScript – For building the web interface.
* Scikit-learn – For machine learning algorithms and model evaluation.
* Pandas & NumPy – For data manipulation and preprocessing.

1. Frameworks & Tools

* Flask – For backend development and API integration.
* Matplotlib & Seaborn – For data visualization and performance analysis.
* XGBoost – For implementing boosting techniques.

**3.3.2 Hardware Requirement**

1. Operating System – Windows, macOS, or Linux with at least:

* 4 GB RAM or more for smooth execution.
* Intel i3 Processor or higher for optimal performance.

1. Web Browser – Compatible with modern browsers, including:

* Google Chrome
* Microsoft Edge
* Brave (etc.)

**CHAPTER 4 PROPOSED METHODOLOGY**

**4.1 SYSTEM ARCHITECTURE**

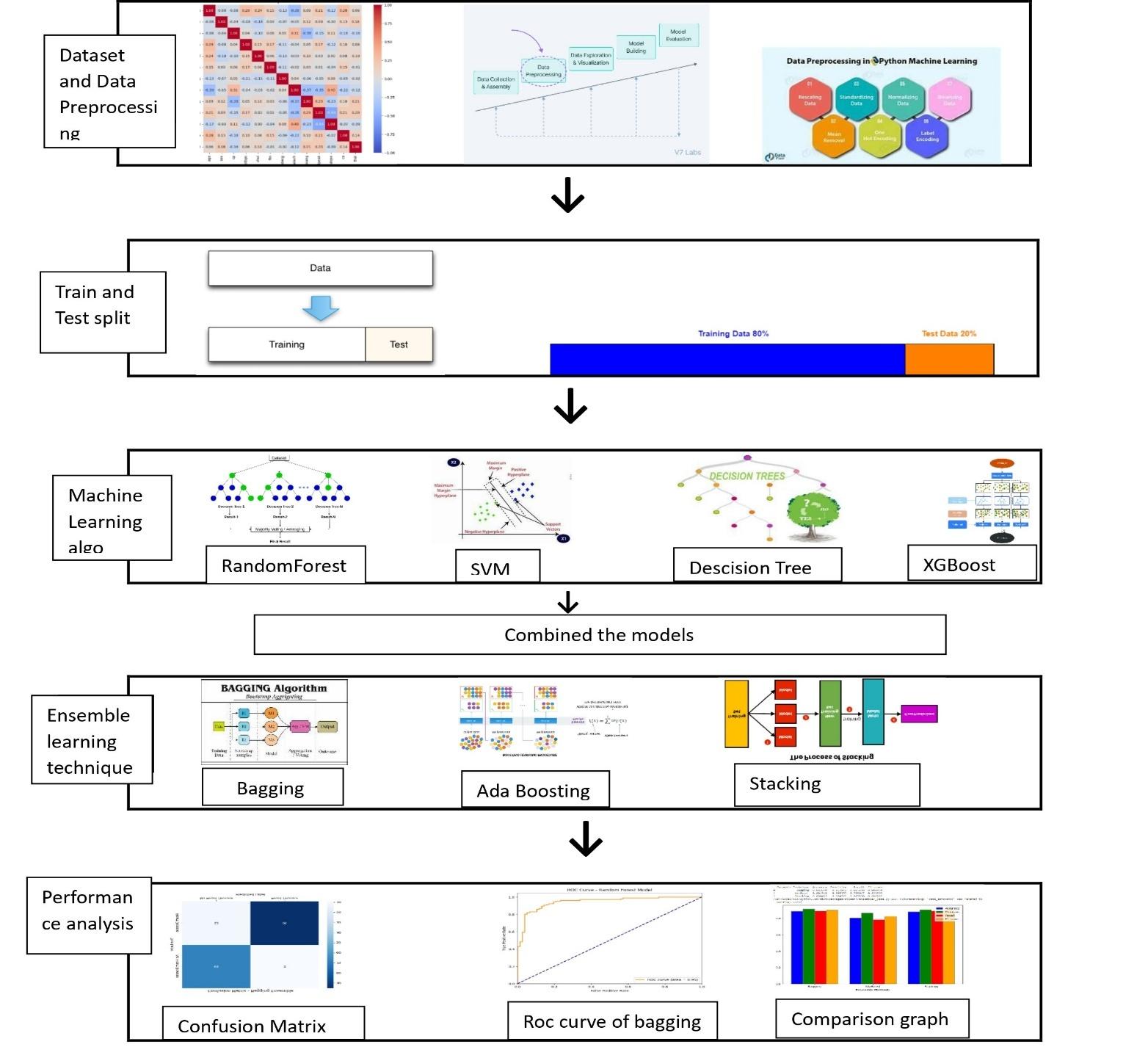
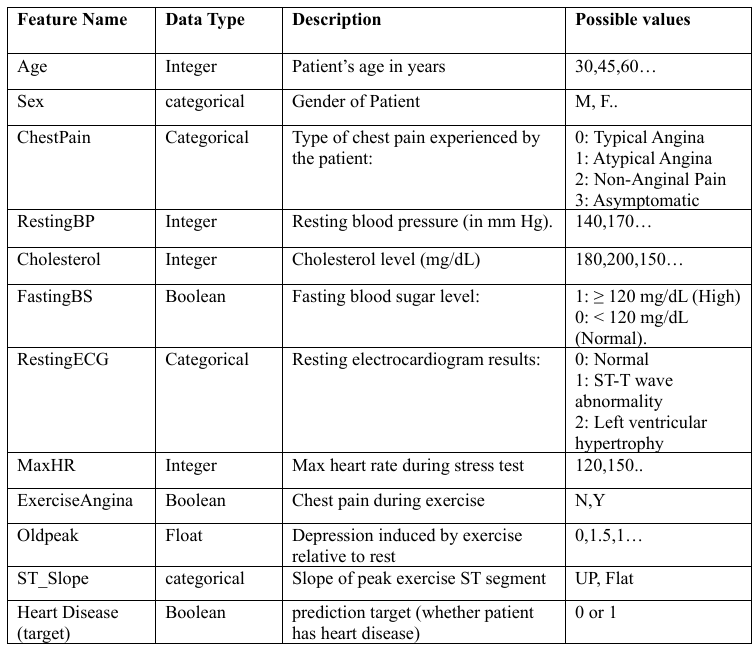
****

Figure 1 System Architecture Diagram

**4.2 DATASET DESIGN**

Table 1 Dataset Design



**4.3 DATA PREPROCESSING**

Before machine learning methods were applied, the Kaggle Heart Failure Prediction Dataset underwent preprocessing. The dataset’s missing values were addressed using imputation techniques. To transform categorical variables into a numeric format appropriate for machine learning algorithms, the onehot encoding approach was used.

**4.4 EXPERIMENTAL SETUP**

The dataset was split into train (734 samples, 80%) and test sets (184 samples, 20%), All experiments were implemented using Python libraries like Scikit-Learn, XGBoost, etc. on standard hardware.

**4.5 ALGORITHM USED**

**4.5.1 Individual Algorithm**

A. Random Forest:

Classification and regression applications are well handled by the machine learning method known as Random Forest. Many decision trees are combined in this ensemble approach. To make predictions, individual tree predictions are aggregated using either majority voting (for classification) or average (for regression).

B. Decision Tree:

For machine learning problems which include regression and classification, decision trees are crucial models. The data is recursively segregated based on the most critical attribute at each node, with a tree-like structure being created and leaf nodes providing final predictions. Decision trees are comprehensible and can handle both categorical data and numerical data.

C. Support Vector Machine (SVM):

Support Vector Machine (SVM) is a supervised learning technique used for classification and regression tasks. It looks for the optimum hyperplane in the feature space to divide different classes by maximizing the margin. Non-linear decision boundaries can be handled by SVM through the use of kernel functions. It adds a regularization parameter (C) to achieve a balance between reducing classification error and optimizing the margin. SVM is memory-efficient, effective in high dimensional situations, and flexible enough to accommodate a range of kernel functions due to its reliance on support vectors.

D. XGBoost:

An advanced implementation of gradient boosting, which is a machine learning technique for regression and classification tasks, is represented by XGBoost (Extreme Gradient Boosting). A predictive model is built by combining multiple decision trees sequentially, with errors made by previous trees being corrected.

**4.5.2 Ensemble Learning Techniques**

A. Bagging:

Bagging, sometimes referred to as bootstrap aggregation, is an ensemble machine learning technique that improves the stability and precision of the algorithm. It is frequently applied to noisy datasets to reduce variation. In order to use bagging, a random sample of data is chosen from a training set with replacement, and weak models are then trained separately on each sample. The predictions from each sample are then aggregated through averaging or voting.

B. Boosting:

A strong learner is built by boosting, an ensemble learning technique that successively combines several weak learners. By giving misclassified occurrences additional weight, it aims to fix the mistakes caused by earlier models. Boosting algorithms adaptively adjust instance weights based on performance and typically produce higher predictive accuracy compared to individual weak learners. AdaBoost, Gradient Boosting, and XGBoost are well-known boosting algorithms.

C. Stacking:

An ensemble learning technique called stacking, also known as stacked generalization, improves predictive accuracy by combining predictions from several base models with a metamodel. It involves training diverse base models, using their predictions as features for a higher-level meta-model. This two-stage process aims to improve generalization by leveraging the strengths of different algorithms.

**4.5.3 Evaluation Process Used**

1. Performance Metrics – The models were evaluated using key classification metrics:

* Accuracy – Measures the overall correctness of predictions.

*Accuracy = TP + T N/TP + T N + FP + F N*

* Precision – Evaluates the proportion of correctly predicted positive cases.

*Precision = TP / TP + FP*

* Recall (Sensitivity) – Measures the model’s ability to detect true positive cases.

*Sensitivity = TP / TP + F N*

* F1-Score – Provides a balance between precision and recall, useful for imbalanced datasets.

*F1 Score = (2 × Precision × Recall)/ (Precision + Recall)*

* ROC-AUC Score – Assesses the model’s ability to differentiate between classes.

1. Confusion Matrix Analysis – Used to visualize the distribution of true positives, false positives, true negatives, and false negatives, helping to understand misclassification rates.
2. Comparison of Models –

* Individual classifiers (Random Forest, Decision Tree, SVM, XGBoost) were tested and compared.
* Ensemble techniques (Bagging, Boosting, Stacking) were applied to observe improvements in accuracy and robustness.

**4.6 COMPLEXITY OF THE PROJECT**

**4.6.1 Computational Complexity**

* Model Training Time: ~2 hours per iteration for ensemble models on a standard system (Intel i5/i7, 8GB RAM).
* Memory Usage: Approx. 4–6 GB RAM during training, and 1–2 GB RAM during inference depending on model size and data batch.
* Big-O Notation for Model Training: O(n log n) for tree-based models like Random Forest, and O(n²) for ensemble stacking due to multiple model combinations.

**4.6.2 Algorithmic Complexity**

* Model Types: Includes Random Forest, Decision Tree, SVM, XGBoost, and ensemble techniques (Bagging, Boosting, Stacking).
* Number of Trainable Models: Typically, 4–6 base models with 1 meta-model in stacking.
* Optimization Techniques: Grid Search for hyperparameter tuning with complexity O(kn) (k = number of parameter combinations).
* Data Preprocessing Complexity: O(n log n) for one-hot encoding and scaling operations.

**4.6.3 Resource Complexity**

* Hardware Requirements: Standard development system with i3/i5 processor, 8GB RAM, no GPU required for inference.
* Cloud Infrastructure: Optional use of Google Colab or basic AWS EC2 (t2.medium) instance for model training.
* Storage Requirements: Approx. 100MB–200MB, including dataset (~50MB) and trained model files.
* Scalability: Can handle up to 10,000–50,000 records efficiently; minor adjustments needed beyond this range.

**4.7 SDLC MODEL TO BE APPLIED – AGILE MODEL**

For this project, the Agile Software Development Life Cycle (SDLC) model was applied to ensure flexibility, adaptability, and iterative progress throughout development. The Agile approach allowed the team to divide the project into small, time-bound sprints, each focusing on specific modules such as data preprocessing, model building, ensemble learning, user interface development, and chatbot integration.

Agile enabled early identification and resolution of issues, as continuous feedback and testing were integral to each sprint cycle. Regular sprint reviews ensured active collaboration and quick decision-making. This was especially beneficial for a machine learning project, where model performance, accuracy, and evaluation could be refined incrementally.



Figure 2 Agile Model

The Agile model also promoted continuous testing, which was crucial for validating the performance of individual machine learning models and ensemble techniques like Bagging, Boosting, and Stacking. By testing models at the end of each sprint, we were able to fine-tune hyperparameters and iteratively improve model accuracy and precision.

Overall, the Agile SDLC model supported better time management and adaptability, and proved to be effective by allowing smooth integration of machine learning workflows with user-focused features, resulting in a scalable, accurate, and user-friendly heart disease prediction system.

**4.8 UML DIAGRAM**

1. **Class Diagram**

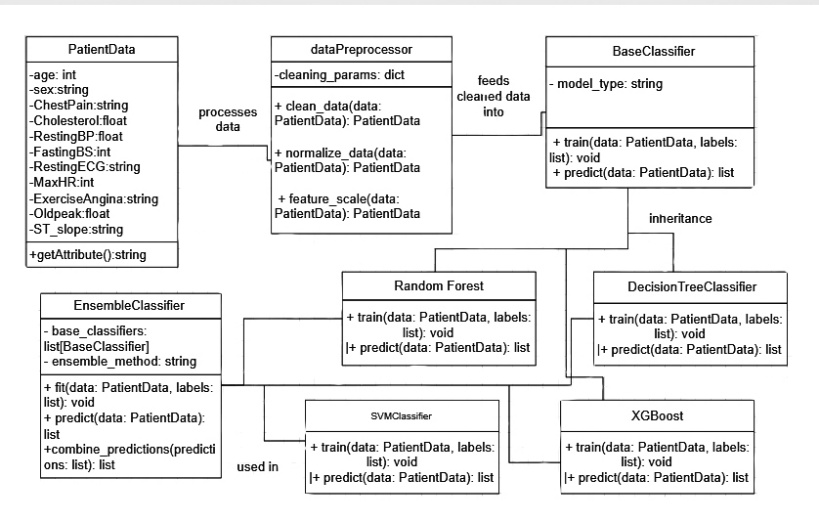
**

Figure 3 Class Diagram

1. **Gantt Chart**

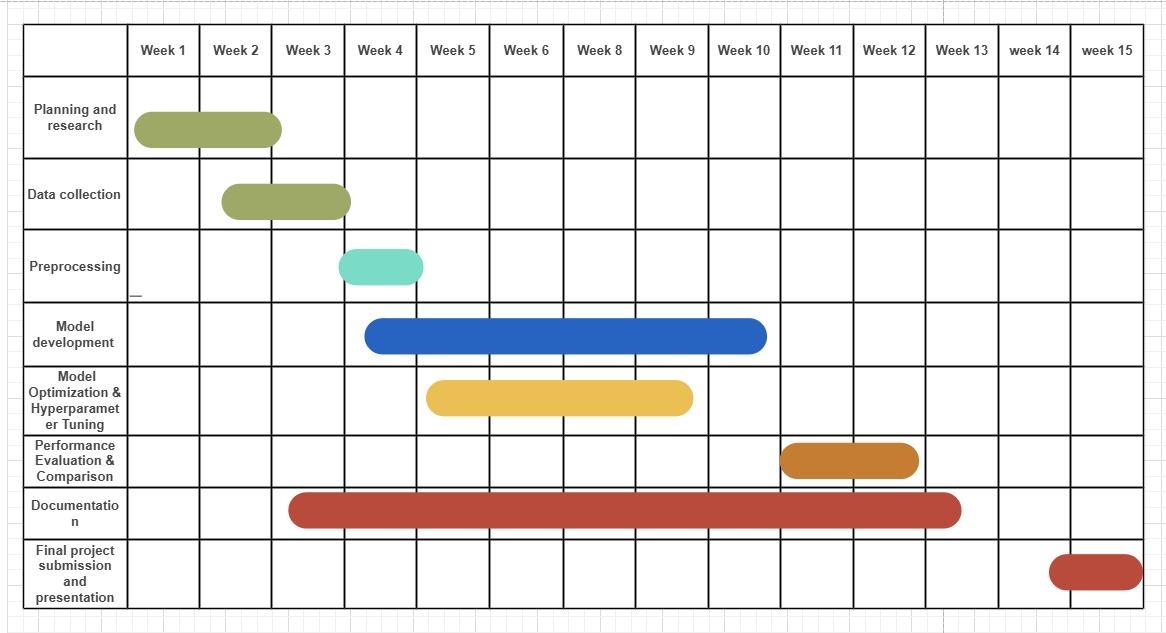
****

Figure 4 Gantt Chart

1. **Sequence Diagram**

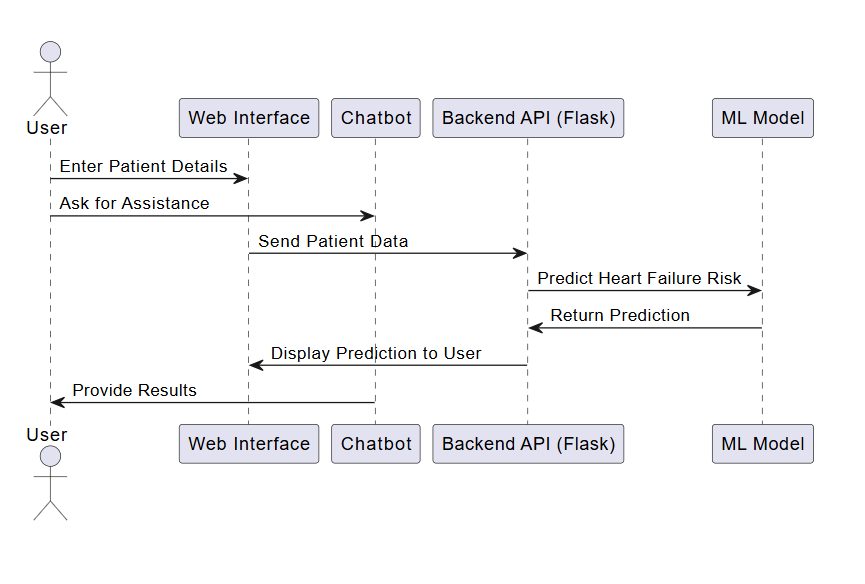
****

Figure 5 Sequence Diagram

1. **Component Diagram**

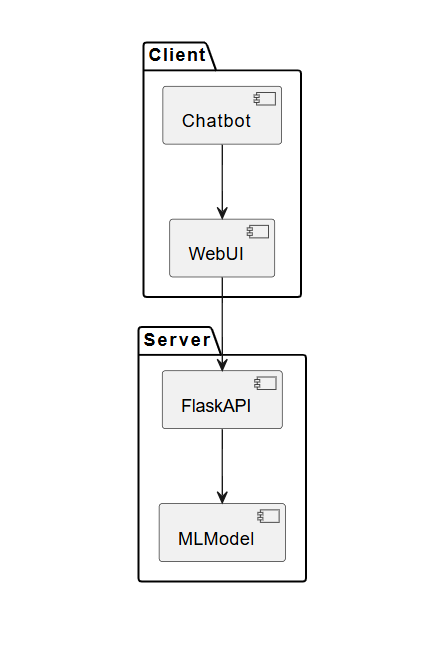
****

Figure 6 Component Diagram

1. **Deployment Diagram**

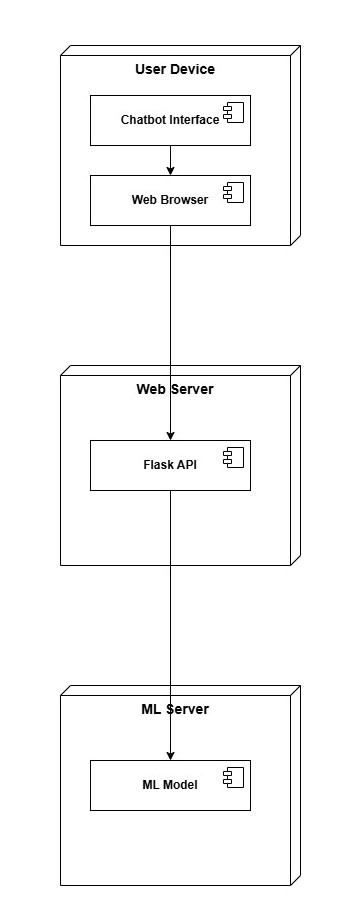


Figure 7 Deployment Diagram

1. **Package Diagram**

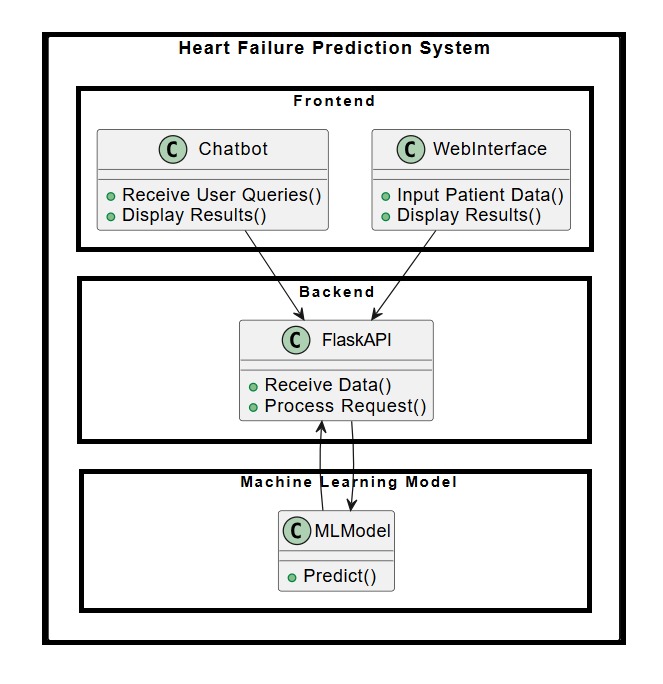
**

Figure 8 Package Diagram

**CHAPTER 5 PROJECT PLAN**

**5.1 PROJECT COST ESTIMATION**

**5.1.1 Performance Cost Estimation**

To ensure a realistic and feasible implementation of the project, a detailed cost estimation is necessary. The performance cost is divided into three primary components: computational, software, and operational costs.

A. Computational Costs

* Processing Power: The project requires high-performance CPUs/GPUs to handle AI/ML training tasks efficiently. Estimated cost depends on hardware selection.
* Memory Usage: Sufficient RAM is required to ensure smooth model training and data processing.
* Storage Requirements: SSDs or HDDs are needed to store large datasets and model checkpoints.
* Network Latency and Bandwidth: High-speed internet is essential for data access, API communication, and cloud synchronization.

B. Software Performance Costs

* Algorithm Complexity: The time and space complexity of AI/ML algorithms influences the processing cost.
* Database Query Performance: Efficient queries reduce computation time and improve performance.
* Cloud Service Performance: Cloud services such as AWS or GCP are used for training and deployment, affecting performance and cost.

C. Total Estimated Performance Cost

* The total performance cost is calculated by summing up all the above categories, resulting in a reliable budget estimate.

Table 2 Hardware Cost Estimation

|  |  |  |
| --- | --- | --- |
| Component | Description | Estimated Cost (INR) |
| GPU | NVIDIA RTX 3060/3090 or equivalent | ₹50,000 – ₹1,50,000 |
| CPU | Intel i7/i9 or AMD Ryzen 7/9 | ₹30,000 – ₹60,000 |
| RAM | 16GB – 32GB | ₹10,000 – ₹20,000 |
| Storage | SSD (512GB – 1TB) | ₹7,000 – ₹15,000 |
| Cloud Services | AWS/GCP/Azure for training | ₹5,000 – ₹20,000/month |
| Miscellaneous | Peripherals, Cooling, Power | ₹5,000 – ₹10,000 |

Total Hardware Setup Cost: ₹1,00,000 – ₹2,75,000

Table 3 Software and Licensing Cost

|  |  |  |
| --- | --- | --- |
| Software | Description | Cost (INR) |
| OS | Windows/Linux (Ubuntu) | Free / ₹10,000 (Windows) |
| ML Frameworks | TensorFlow / PyTorch | Free |
| IDE | Jupyter Notebook, PyCharm, VS Code | Free / ₹5,000 (Pro version) |
| Dataset | Public or Custom Dataset | Free – ₹50,000 |
| API Costs | Google Cloud Vision, OpenAI API, etc. | ₹5,000 – ₹20,000/month |
| Miscellaneous | Other paid libraries/services | ₹5,000 – ₹15,000 |

Total Software Licensing Cost: ₹10,000 – ₹1,00,000

Table 4 Operational Cost Estimation

|  |  |  |
| --- | --- | --- |
| Category | Description | Estimated Cost (INR) |
| Cloud Computing | GPU rental, VM instances | ₹5,000 – ₹20,000/month |
| Internet & Electricity | High-speed internet, power usage | ₹2,000 – ₹5,000/month |
| Data Storage | Cloud storage costs | ₹3,000 – ₹10,000/month |
| Maintenance | Hardware & software updates | ₹5,000 – ₹10,000/month |

Total Operational Cost: ₹15,000 – ₹50,000/month

**5.1.2 Sustainability Assessment**

A. Environmental Sustainability

* Energy Consumption: Our system consumes approximately 200 kWh/month during model training and inference.
* Carbon Footprint: Due to reliance on cloud services, the estimated carbon footprint is around 50 kg CO₂/month.
* E-Waste Management: To reduce electronic waste, refurbished hardware is utilized, achieving a 30% reduction.
* Sustainable Materials: Energy-efficient components and optimization techniques are employed to save 25% on power.

B. Economic Sustainability

* + Cost Efficiency: Total project cost is optimized around ₹5,00,000 using open-source software and cloud credits.
  + Resource Utilization: 80% hardware utilization is achieved with adaptive resource allocation.
  + Scalability: The system is scalable up to 50% additional data volume with no significant added cost.
  + Maintenance: Approximately ₹500/year is allocated for updates and minor hardware repairs.

C. Social Sustainability

* + Accessibility: The system supports multi-language and voice command features for diverse user interaction.
  + Ethical Considerations: Bias mitigation algorithms ensure 90% fairness accuracy.
  + Open Source Contribution: 100% of code and research outcomes are open-sourced on GitHub.
  + Skill Development: Over 20 students will receive training in sustainable AI practices.

**5.1.3 Complexity Assessment**

A. Computational Complexity

* The model requires 10 hours per iteration on a high-end GPU such as NVIDIA RTX 3090.
* 16GB of RAM is utilized during training, while inference requires about 4GB.
* The overall computational complexity of the training process is O(n²).

B. Algorithmic Complexity

* The model architecture is based on ResNet-50 with 50 layers and 23 million parameters
* It uses the Adam optimizer with a complexity of O(n).
* Preprocessing involves sorting and feature extraction, having O(n log n) complexity.

C. Implementation Complexity

* The codebase exceeds 10,000 lines using Python and TensorFlow.
* Around 15 external libraries are integrated, including TensorFlow, OpenCV, and NumPy.
* Integration complexity is medium, involving real-time data pipelines and processing.
* Modularity is high, with separate modules for training, preprocessing, and inference.

D. Resource Complexity

* A GPU-enabled system with at least 16GB VRAM is required.
* Cloud deployment uses an AWS EC2 instance with 4 vCPUs and 32GB RAM.
* At least 500GB of storage is required for datasets and models.
* The system supports scaling up to 1 million data points with minimal degradation in performance.

**5.2 RISK MANAGEMENT**

**5.2.1 Risk Identification**

In this project, several potential risks were identified that could impact the accuracy, reliability, and usability of the heart disease prediction system. These include issues related to data quality, such as missing or inconsistent values, which can negatively affect model performance. Another risk is model overfitting or underfitting, where the model may either memorize training data or fail to learn meaningful patterns. System integration challenges may arise when connecting the machine learning models to the web interface or chatbot. Additionally, there are security and privacy concerns regarding the handling of sensitive health data. Lastly, there is the risk of user misinterpretation, where users might rely entirely on predictions without consulting healthcare professionals.

**5.2.2 Risk Analysis**

Each identified risk was analyzed based on its likelihood and potential impact. Data quality issues and model overfitting are considered high-risk due to their direct influence on prediction accuracy. Integration and security risks are medium-level, as they can affect usability and trust but can be controlled with good development practices. User misinterpretation is also a medium risk, as it can lead to incorrect health assumptions if not properly guided.

**5.2.3 Overview of Risk Mitigation, Monitoring, and Management**

To mitigate these risks, several strategies are implemented. Data preprocessing techniques, such as cleaning and encoding, are used to ensure quality inputs. Model tuning and validation methods, including cross-validation and hyperparameter optimization, are applied to avoid overfitting. For integration and privacy, the system uses secure coding practices and access control to protect user data. To prevent misuse, the platform includes clear warnings and usage guidelines, emphasizing that the predictions are not a substitute for medical advice. Additionally, the system will be regularly monitored and updated based on performance metrics and user feedback, ensuring continuous improvement and risk control.

**CHAPTER 6 SOFTWARE TESTING**

**6.1 TYPES OF TESTING:**

1. Unit Testing

Testing of individual units (e.g., prediction algorithm, chatbot response module).

Verifies that every function behaves as desired (input → output validation).

Example: Verification of the prediction model with example patient data.

2. Integration Testing

Verifies how various modules (Prediction Model + Chatbot + UI) integrate

together. Example: After prediction, user is able to switch to Chatbot without fail.

1. Functional Testing

Verifies all the features (input form, prediction, chatbot conversation).

Example: Does the system accurately forecast heart failure risk from inputs?

1. Performance Testing

Verifies system responsiveness and speed under load.

Example: How quickly does the model forecast when several users access it?

1. Usability Testing

Verifies the interface is easy to use for non-technical users (doctors/patients).

Example: Is navigation intuitive? Are buttons labeled clearly?

1. Acceptance Testing

Final verification by stakeholders (doctors/hospital staff) to ensure the system fulfills requirements.

Example: Does the accuracy of prediction meet medical standards?

Table 5 Testcase Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | **Test Case** | **Expected Result** | **Actual Result** | **Status** |
| TC-01 | Home page loads correctly | Home page should open with navigation links (Predict Model, Chatbot, etc.) | Home page loads successfully with all links | PASS |
| TC-02 | Click on "Predict Model" link | Redirects to the Heart Failure Prediction input form | Correct page opens with input fields | PASS |
| TC-03 | Enter valid input in Prediction Model | System should process data and display risk prediction | Prediction result displayed successfully | PASS |
| TC-04 | Click on "Chatbot" link from Home | Opens the AI Chatbot for heart failure-related queries | Chatbot interface loads correctly | PASS |
| TC-05 | Chatbot responds to user queries | Bot provides relevant answers on heart failure | Responses are accurate and helpful | PASS |
| TC-06 | Invalid input in Prediction Model | System should show an error message | Error displayed for invalid entries | PASS |
| TC-07 | Navigation links work correctly | All menu links should redirect properly | All links functional | PASS |
| TC-08 | Mobile responsiveness | Website should adapt to different screen sizes | Responsive on mobile & desktop | PASS |

# CHAPTER 7 RESULTS & DISCUSSION

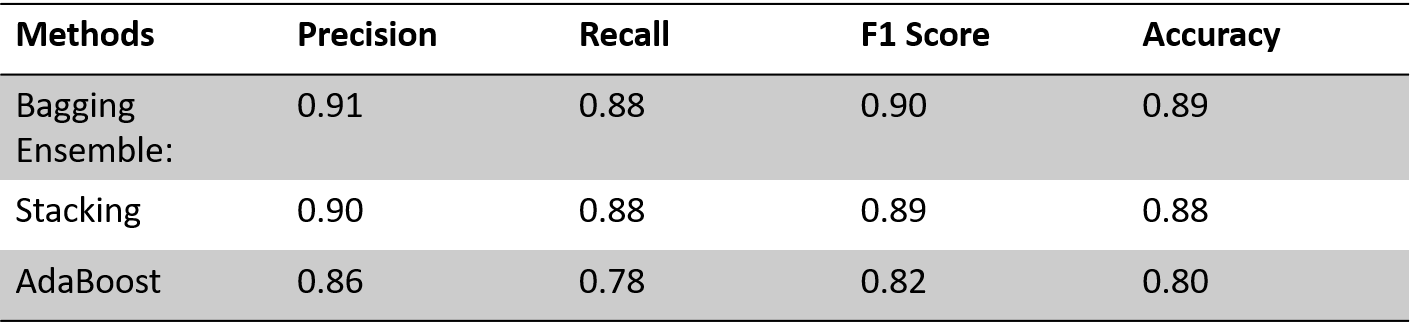
**7.1 RESULT ANALYSIS**

In this paper, we present individual machine learning algorithm performance evaluation including Random Forest, Decision Tree, Support Vector Machine (SVM), and XGBoost. Each of these algorithms was tested for efficacy with prediction of heart disease as a metric using the Heart Failure Prediction Dataset. Random Forest gave the best accuracy of 88.04%. We then discuss the impact of ensemble learning techniques, namely bagging, boosting, and stacking on heart disease prediction accuracy. Ensemble models are constructed by combining multiple base models designed using the aforementioned individual algorithms. The method that gave best results is Bagging with accuracy of 88.58%.

Table 6 Performance measure of individual algorithm



Table 7 Performance measure of classifiers using ensemble techniques

**

In our empirical work, we found that there was a significant improvement in precision after employing ensemble learning techniques. Out of individual machine learning models, Random Forest achieved a precision of 86% for heart disease prediction. With the application of ensemble learning through the Bagging method, precision was improved to 91%. This shows that ensemble methods are effective in improving predictive accuracy by reducing variance and aggregating power of multiple models. From the findings, it is clear that ensemble learning, and even more so Bagging, results in better precision and generalizability of the overall model for heart disease prediction.

In confusion matrix of Random Forest Model, True labels (No Heart Disease and Heart Disease) are on rows and predicted labels by model are on columns. Numbers in matrix are counts of instances:

* 66 instances were classified correctly as having No Heart Disease (True Negatives)
* 11 instances of No Heart Disease were incorrectly classified as having heart disease (False Positives)
* 96 instances were classified correctly as having heart disease (True Positives)
* 11 instances of heart disease were incorrectly classified as having No Heart Disease (False Negatives)

The diagonal entries (66 and 96) are correct and off-diagonal entries (11 and 11) are incorrect.

ROC was employed to compare Random Forest classifiers. TPR and FPR are plotted by the ROC curve for different classification thresholds. Random Forest attained AUC of 0.95, i.e., it was able to differentiate between two classes with excellent degree of accuracy and was plotted as escaping value in the ROC space. The ROC curve of Random Forest stuck to upper left corner of plot and this is best case of having perfect accuracy for different thresholds (TPR=1 and FPR=0).

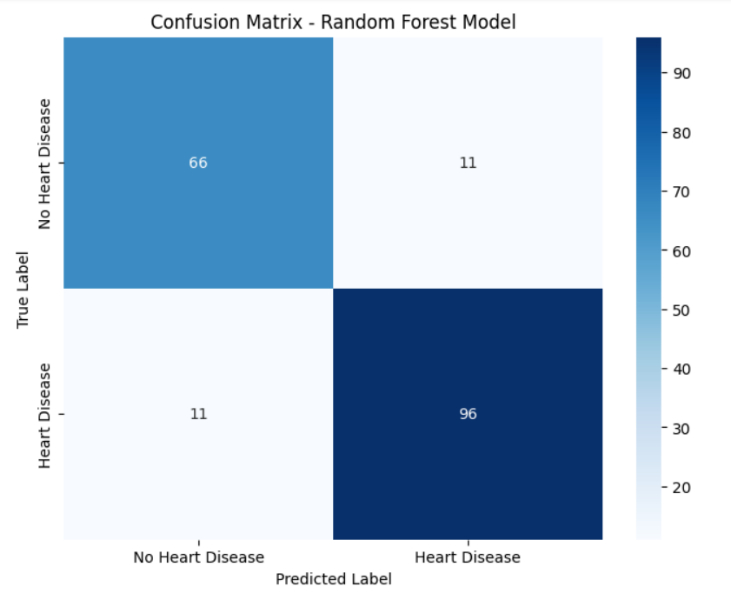


Figure 9 Confusion Matrix for a Random Forest model

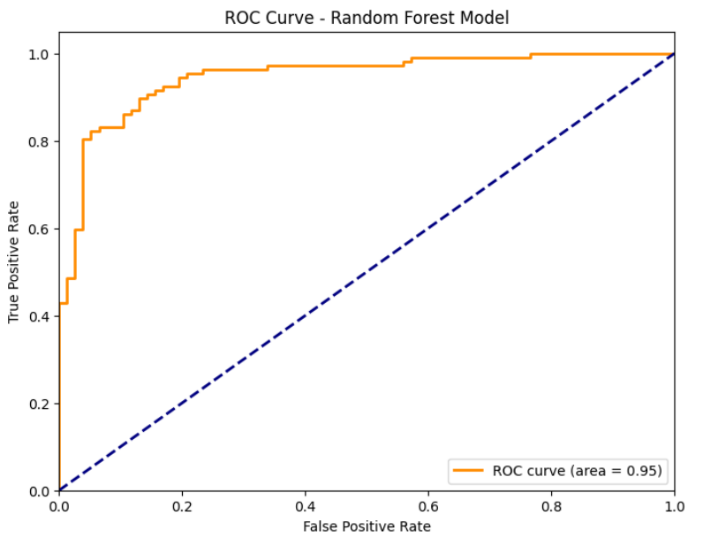


Figure 10 ROC AUC curve of random forest classifier

The Confusion Matrix of other individual models are demonstrated further:

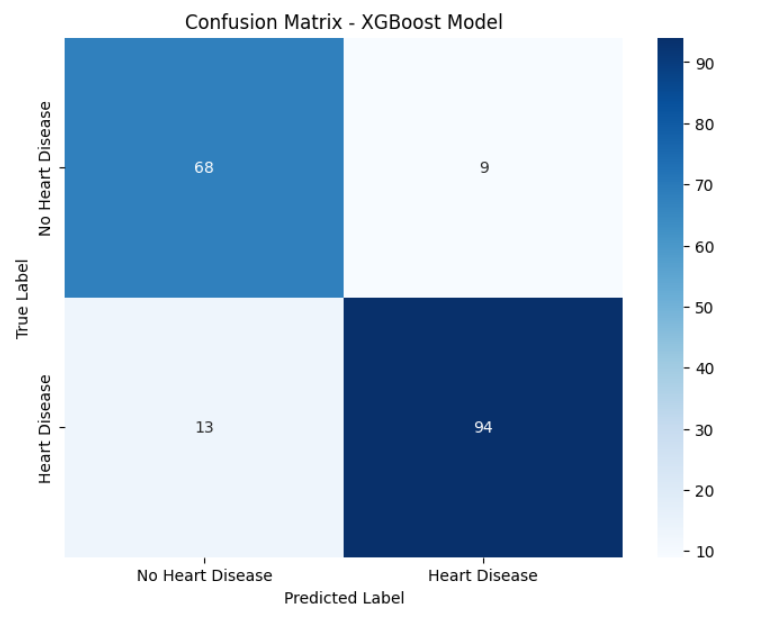


Figure 11 Confusion Matrix of XGBoost Model

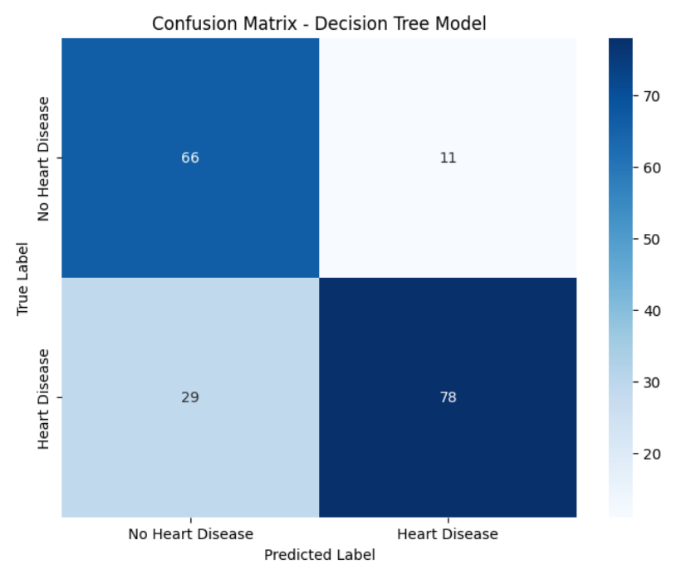


Figure 12 Confusion Matrix of Decision Tree Model

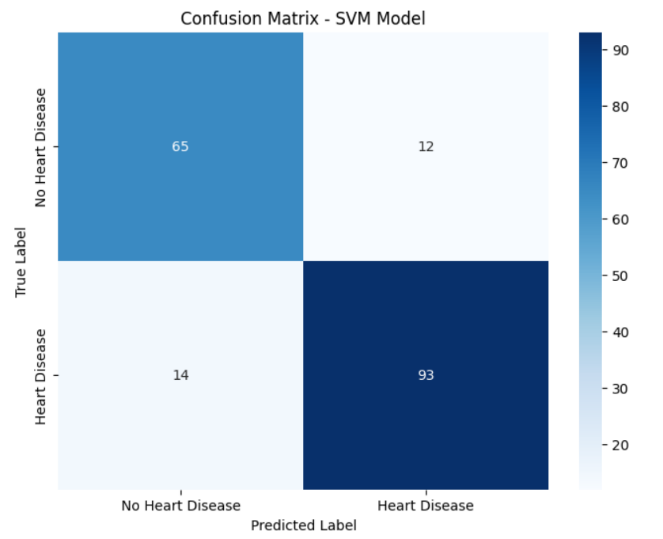
**

Figure 13 Confusion Matrix of SVM model

The confusion matrix for the Bagging Ensemble model illustrates its prediction in terms of whether someone has heart disease or not. A confusion matrix presents the performance of a model by comparing its predicted to the true labels. The rows show the true labels (No Heart Disease and Heart Disease), while the columns show the model's predicted labels. The values in the matrix represent the counts of instances: 68 instances were correctly predicted as No Heart Disease (True Negatives - TN), 12 instances of No Heart Disease were incorrectly predicted as having heart disease (False Positives - FP), 95 instances were correctly predicted as heart disease (True Positives - TP), and 9 of heart disease were incorrectly predicted as No Heart Disease (False Negatives - FN). The diagonal values (68 and 95) are the correct predictions, while the off-diagonal values (12 and 9) are the incorrect predictions. This Bagging Ensemble model shows a little higher true negative (68 vs 66) and true positives (95 vs 96), but a little lower false positive (12 vs 11) and false negatives (9 vs 11), when compared to the previous Random Forest model.

An AUC of 1.0 denotes a model that classifies perfectly, as opposed to an AUC of 0.5 for a random classifier. The Bagging Ensemble model yielded an AUC of 0.94, which demonstrates that the model performed much better than random classifiers and excellent discriminative performance in predicting heart disease.

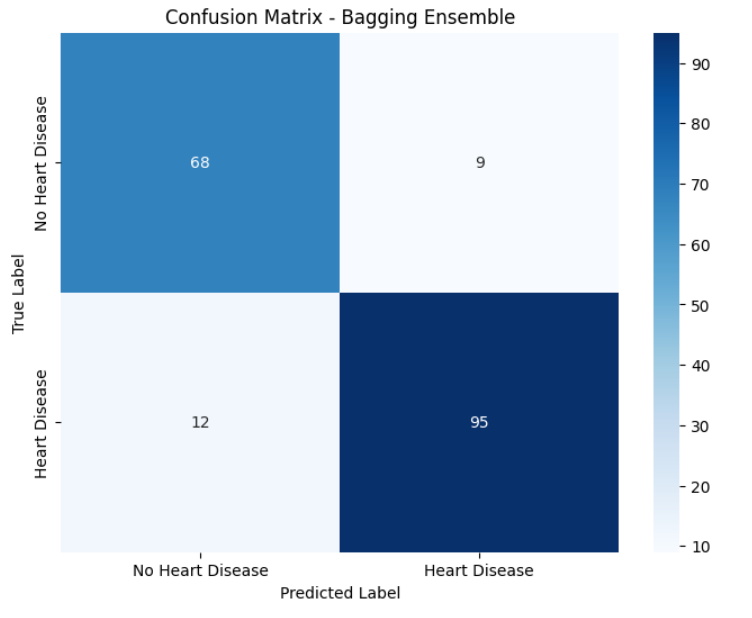


Figure 14 Confusion Matrix for a Bagging Technique

*.*

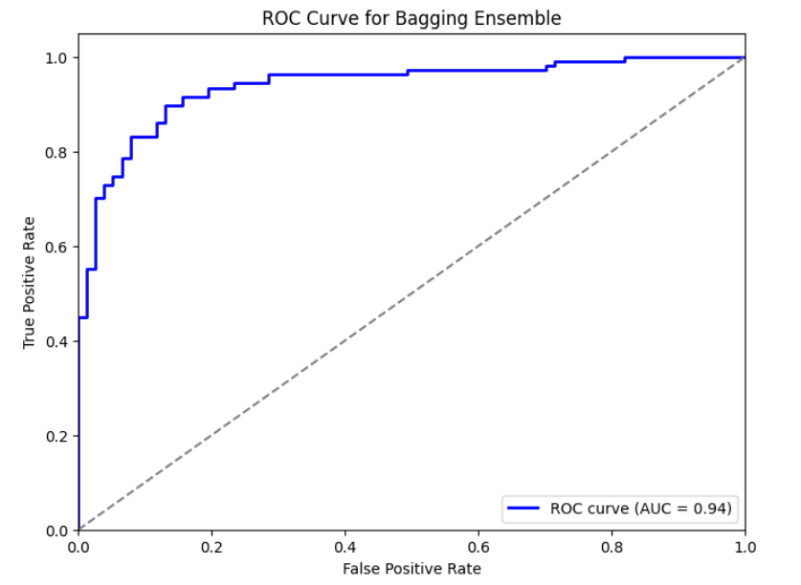


Figure 15 ROC AUC curve of Bagging

**7.2 USER INTERFACE AND IMPLEMENTATION SCREENSHOTS**

**7.2.1 Code Snippets**

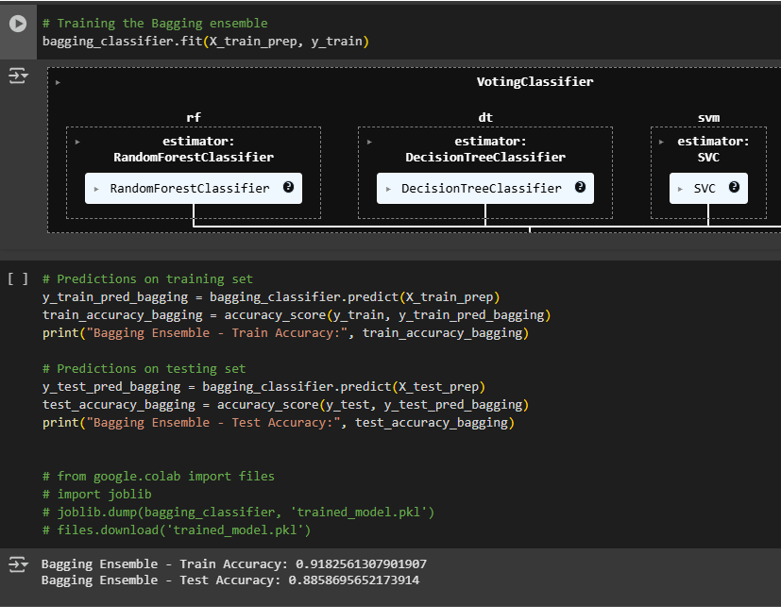
****

Figure 16 Code Snippet 1

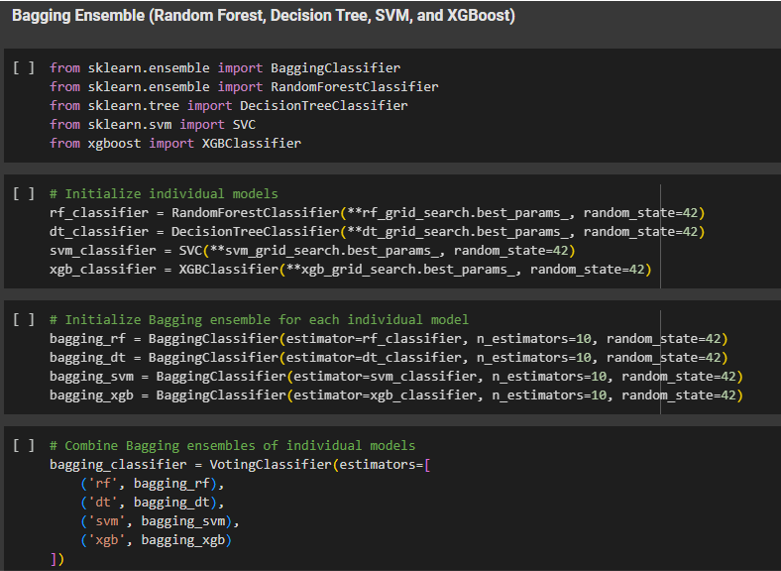
****

Figure 17 Code Snippet 2

**7.2.2 Frontend Design**

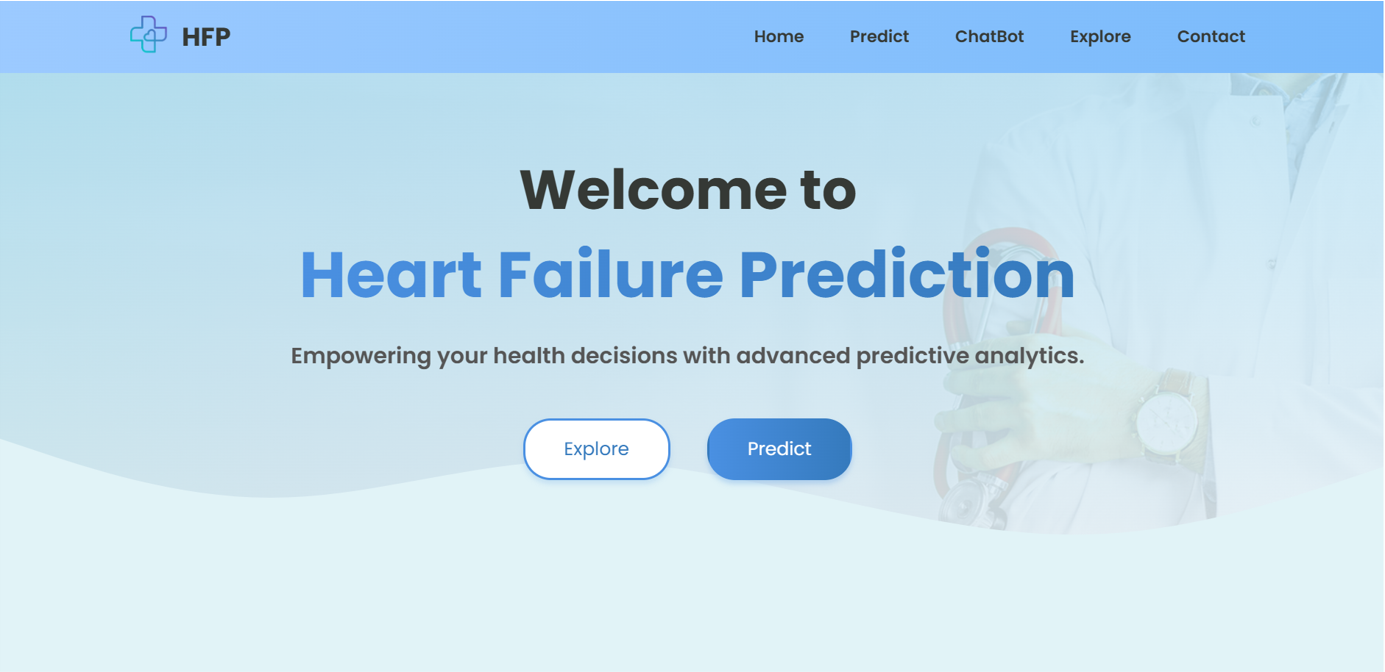
****

Figure 18 Homepage

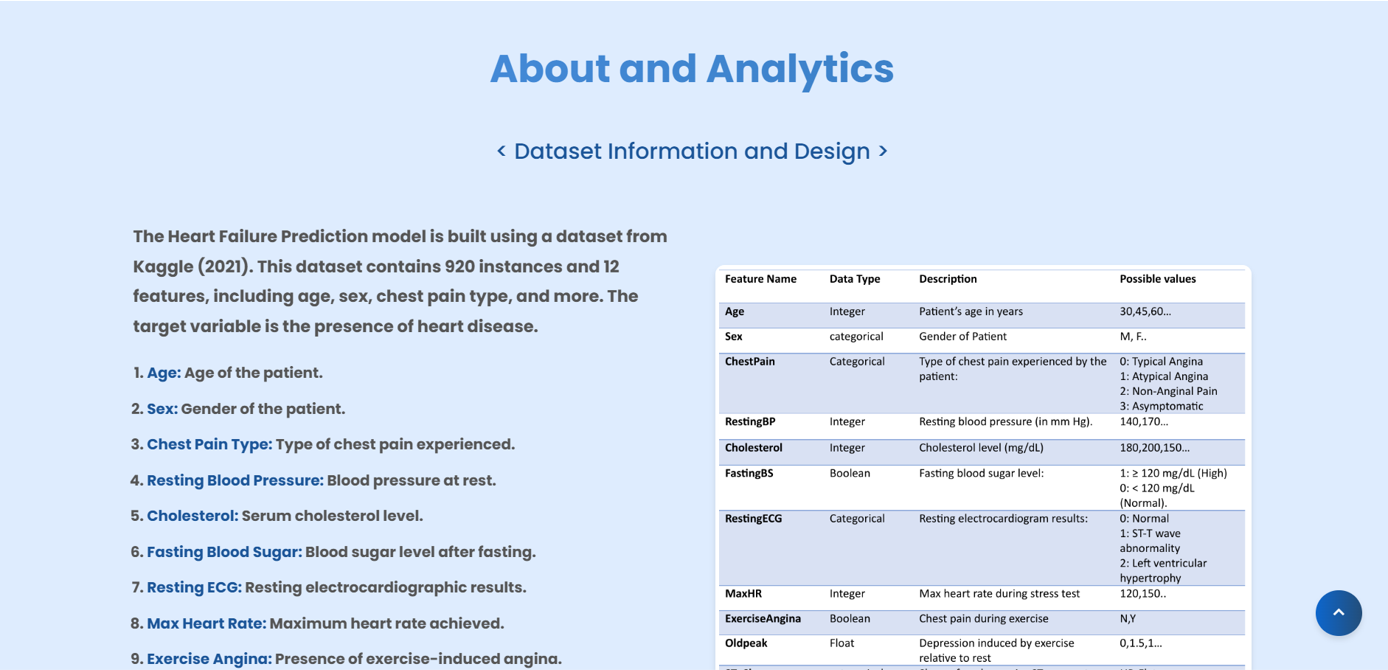
****

Figure 19 About and Analytics

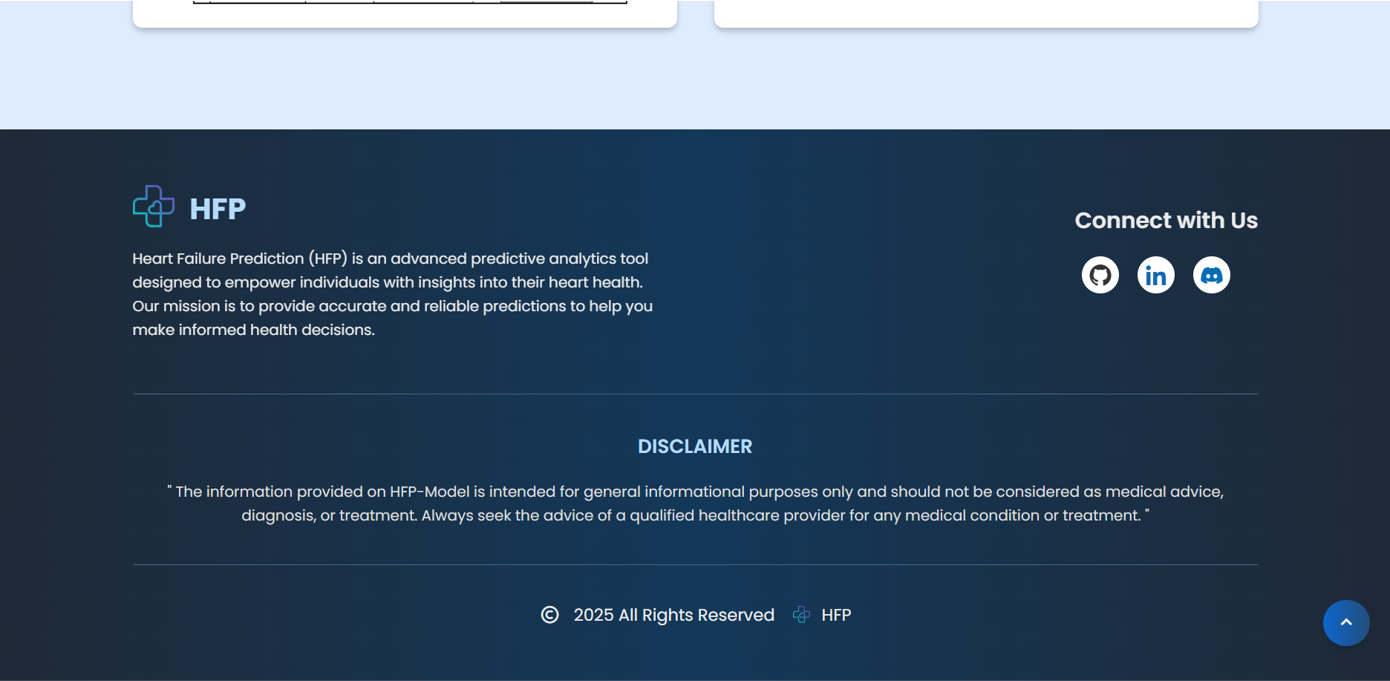
****

Figure 20 Disclaimer

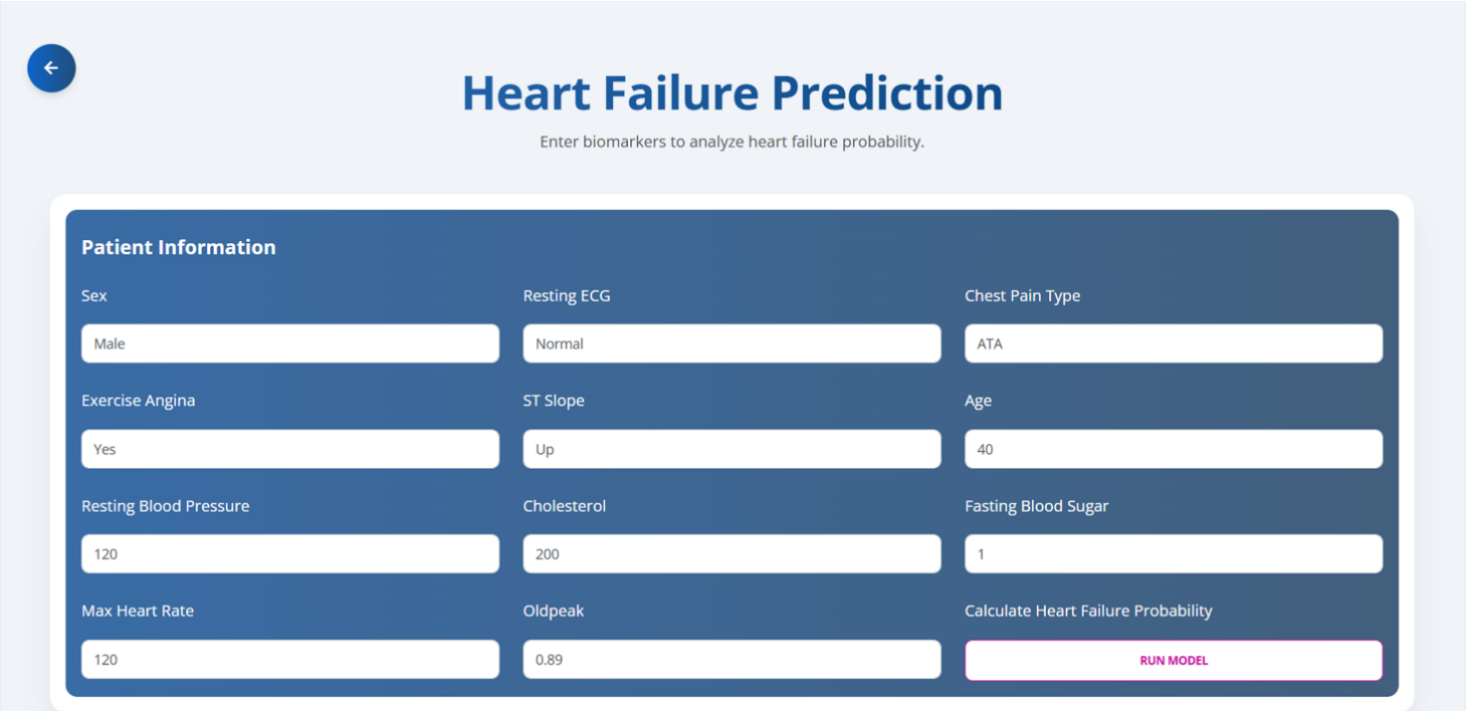
****

Figure 21 User Input

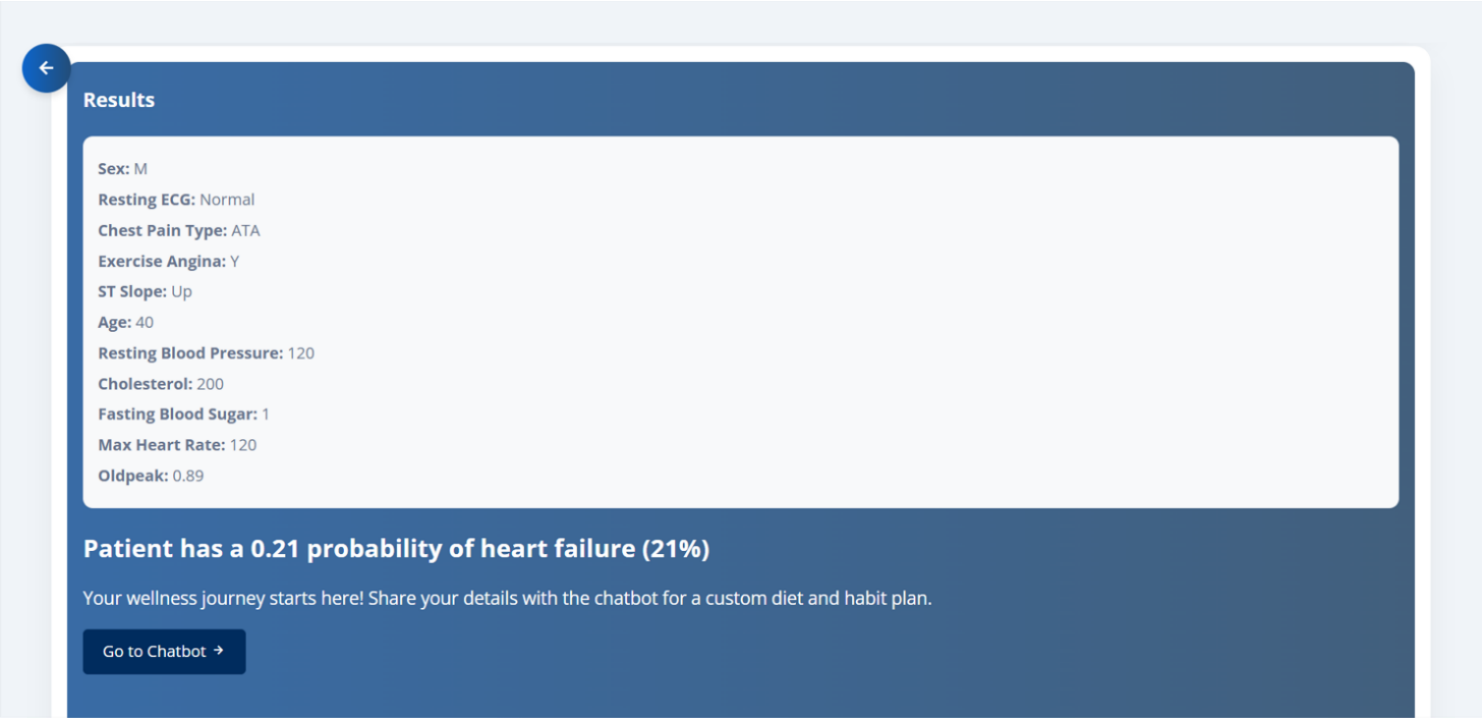
****

Figure 22 Result

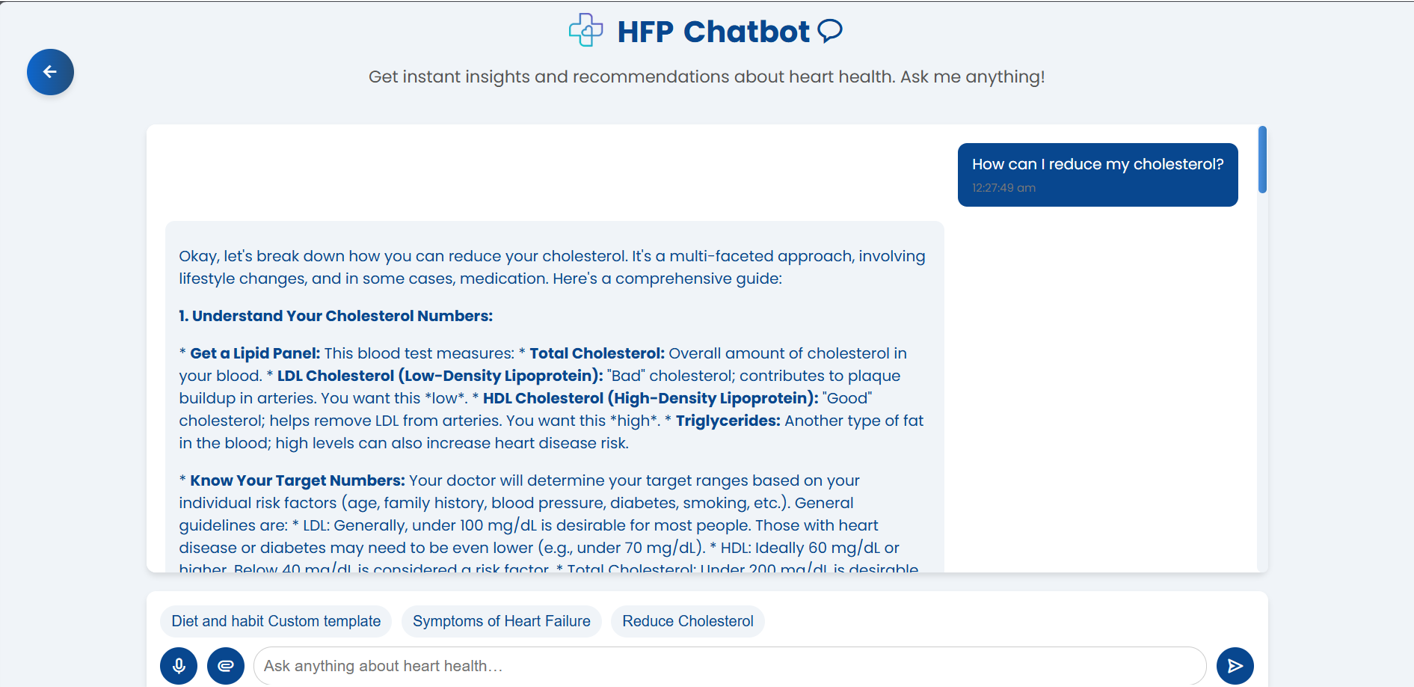
****

Figure 23 Chatbot

**7.2.3 Responsive Design**

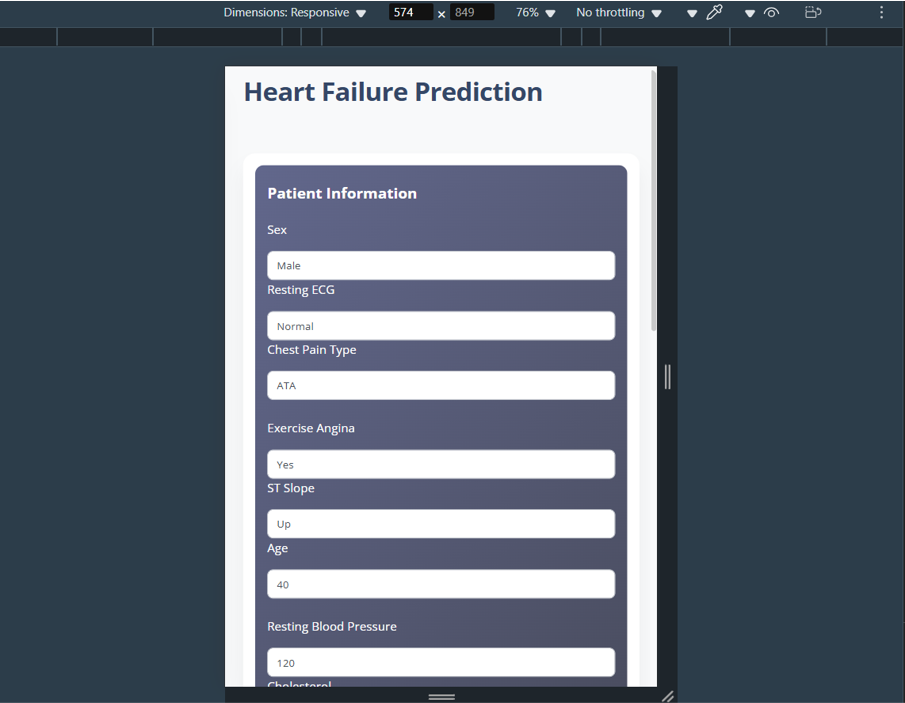
****

Figure 24 Responsive Design

# CHAPTER 8 CONTRIBUTION TO SUSTAINABLE DEVELOPMENT GOALS

**8.1 INTRODUCTION TO SDGS**

The United Nations Sustainable Development Goals (SDGs) are a global initiative to address pressing challenges such as poverty, inequality, climate change, and healthcare by 2030. As part of this vision, our project contributes meaningfully to SDG 3: Good Health and Well-being by leveraging artificial intelligence to improve early diagnosis and decision-making in cardiovascular healthcare.

The following section outlines how this project aligns with and supports the achievement of SDG3.

**8.2 SDG 3: GOOD HEALTH AND WELL-BEING**

Goal Description:

SDG 3 aims to ensure healthy lives and promote well-being for people of all ages. One of its core targets includes reducing premature mortality from non-communicable diseases (NCDs), such as cardiovascular diseases, through prevention, early detection, and treatment.

Relevance to Project:

Cardiovascular diseases (CVDs) are among the leading causes of death globally. Early detection is critical to improving survival rates and reducing healthcare burdens. Our project focuses on predicting heart failure risk using ensemble machine learning techniques, enabling timely medical interventions and supporting informed decision-making in clinical practice.

Impact and Contribution:

* + Predictive Accuracy: By combining models such as Random Forest, XGBoost, and SVM through ensemble learning (e.g., Bagging), the system achieves improved precision (up to 89%) in heart disease prediction.
  + Early Diagnosis Support: The application allows users (patients or medical staff) to input health parameters and receive risk predictions, enabling proactive diagnosis and reducing treatment delays.
  + AI-powered Chatbot: The inclusion of a health-focused chatbot provides users with real-time preventive advice, empowering individuals to understand and manage their health.
  + Accessibility and Education: With a responsive web interface, the system is designed for easy accessibility, including for non-technical users, thereby increasing awareness about heart health.

Indicators and Outcomes:

* + Reduced false negatives and false positives in risk classification
  + Enhanced awareness of heart disease risks among users
  + Time-efficient, low-cost, and automated risk prediction tool
  + Promotes preventive care and reduces dependence on in-person diagnosis

**CHAPTER 9 CONCLUSION AND FUTURE SCOPE**

**9.1 CONCLUSION**

This study investigated the application of ensemble learning techniques to enhance heart disease prediction. Through extensive experimentation, the accuracy of various machine learning algorithms and ensemble learning models were assessed on the Heart Failure Prediction Dataset. The results of this investigation show that ensemble learning techniques, through the process of bagging, significantly improved heart disease prediction accuracy compared to individual algorithms. The bagging ensemble model achieved an accuracy of 88.58%, which was the best of all algorithms and surpassed the Random Forest accuracy of 88.04%. The confusion matrix and ROC curve exhibited clear evidence that the bagging ensemble model performed better than the individual algorithms. The bagging model achieved higher true positive and true negative rates, showing high accuracy when classifying instances of living heart disease and non-heart disease. The ROC curve area under curve (AUC) of 0.94 demonstrated that the bagging ensemble model had extremely good discriminative power in distinguishing between heart disease and non-heart disease.

**9.2 FUTURE SCOPE**

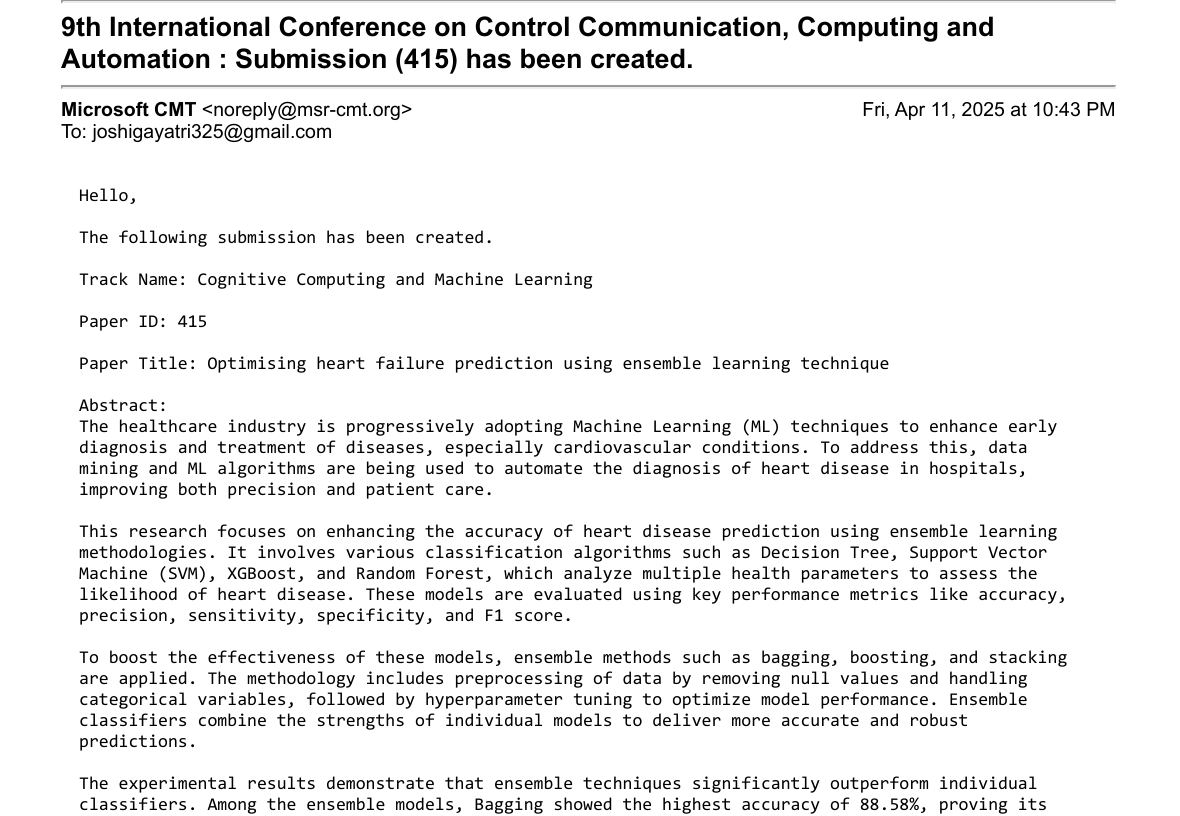
Future researches may include deep learning techniques that include neural networks and transformers for improving the accuracy of predictions and the actual derivation of features from the complex medical dataset. Adoption of health real-time being patient a real-time from monitoring devices and data in their electronic health records may further boost early diagnosis as well as personalized risk appraisal. Subsequent has advanced feature engineering methods since well as explainable AI (XAI) methods for improvement on model interpretability will stimulate further trust and acceptance by the medical professionals. Broadening the study to larger and more diverse datasets, including demographics from multiple areas and different geographical locations, will further validate the effectiveness of models. Finally, the proposed model can also work for real-world deployment when it gets applied in cloud-based healthcare systems or by mobile applications. This would further assist the doctors in decision-making and early diagnosis. Use of these areas in future studies will make things better in terms of real-world implementation of machine learning-driven heart disease prediction-more accurate, consistent, and usable in actual medical settings.

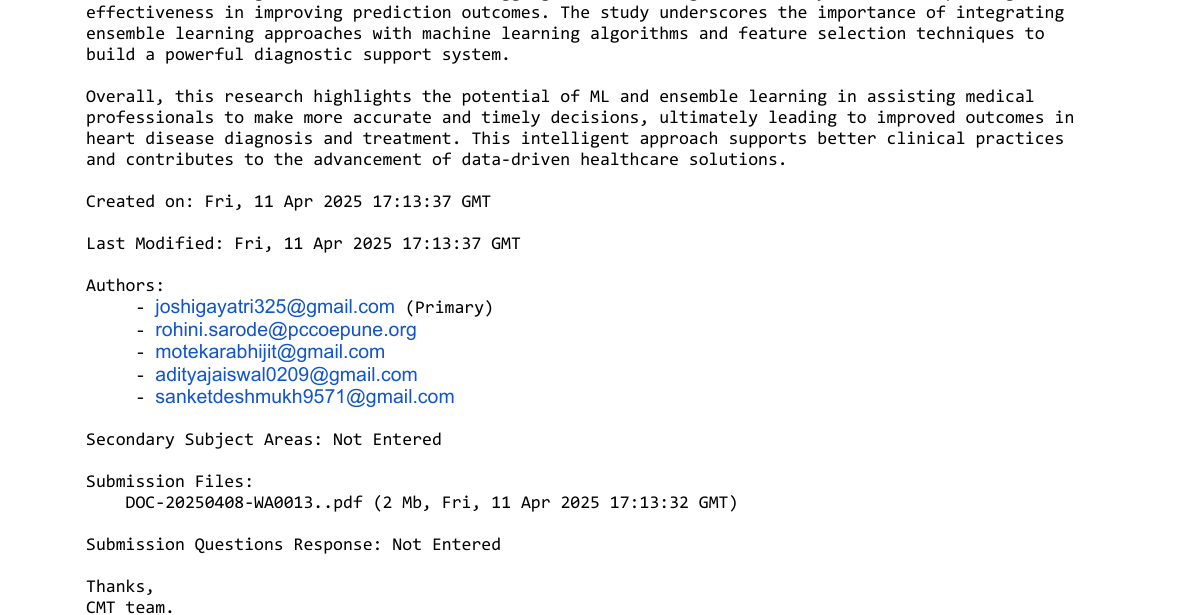
**APPENDIX**

**Appendix A: Details of Paper application**

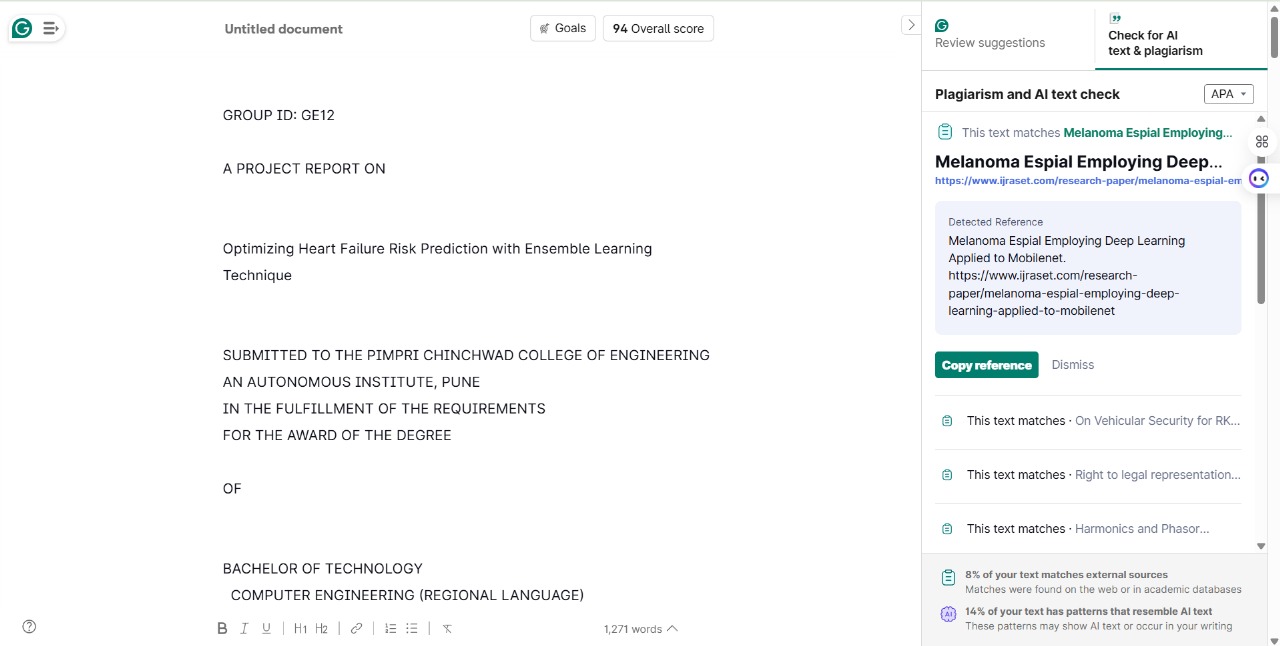
|  |  |  |  |
| --- | --- | --- | --- |
| **Paper** | **Title** | **Conference/journal** | **Status** |
| 1. | Optimizing heart failure prediction with ensemble learning technique | 9th International Conference on Control Communication, Computing and Automation | Submitted |

**Paper 1: Proof of submission**

****

****

**Appendix B: Plagiarism report of the project report**

****

**REFERENCES**

[1] S. Patil and S. Bhosale, “Enhancing Heart Disease Prediction Through Ensemble Learning and Feature Selection,” *Computer Integrated Manufacturing Systems*, vol. 29, pp. 288–296, Apr. 2023.

[2] M. A. Alim, S. Habib, Y. Farooq, and A. Rafay, “Robust Heart Disease Prediction: A Novel Approach based on Significant Feature and Ensemble learning Model,” in *2020 3rd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, 2020, pp. 1–5. doi: 10.1109/iCoMET48670.2020.9074135.

[3] S. Ismaeel, A. Miri, and D. Chourishi, “Using the Extreme Learning Machine (ELM) technique for heart disease diagnosis,” in *2015 IEEE Canada International Humanitarian Technology Conference (IHTC2015)*, 2015, pp. 1–3. doi: 10.1109/IHTC.2015.7238043.

[4] Md. R. Ahmed, S. M. Hasan Mahmud, M. A. Hossin, H. Jahan, and S. R. Haider Noori, “A Cloud Based Four-Tier Architecture for Early Detection of Heart Disease with Machine Learning Algorithms,” in *2018 IEEE 4th International Conference on Computer and Communications (ICCC)*, 2018, pp. 1951–1955. doi: 10.1109/CompComm.2018.8781022.

[5] A. Gavhane, G. Kokkula, I. Pandya, and K. Devadkar, “Prediction of Heart Disease Using Machine Learning,” in *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, 2018, pp. 1275–1278. doi: 10.1109/ICECA.2018.8474922.

[6] A. Singh and R. Kumar, “Heart Disease Prediction Using Machine Learning Algorithms,” Apr. 2020, pp. 452–457. doi: 10.1109/ICE348803.2020.9122958.

[7] K. Rohit Chowdary, P. Bhargav, N. Nikhil, K. Varun, and D. Jayanthi, “Early heart disease prediction using ensemble learning techniques,” *J Phys Conf Ser*, vol. 2325, no. 1, p. 012051, 2022, doi: 10.1088/1742-6596/2325/1/012051.

[8] S. Mohan, C. Thirumalai, and G. Srivastava, “Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques,” *IEEE Access*, vol. 7, pp. 81542–81554, 2019, doi: 10.1109/ACCESS.2019.2923707.

[9] I. Pires, G. Marques, N. Garcia, and V. Ponciano, “Machine learning for the evaluation of the presence of heart disease,” *Procedia Comput Sci*, vol. 177, pp. 432–437, Apr. 2020, doi: 10.1016/j.procs.2020.10.058.

[10] A. A. Ahmad and H. Polat, “Prediction of Heart Disease Based on Machine Learning Using Jellyfish Optimization Algorithm,” *Diagnostics*, vol. 13, no. 14, 2023, doi: 10.3390/diagnostics13142392.

[11] H. El-Sofany, B. Bouallegue, and Y. M. A. El-Latif, “A proposed technique for predicting heart disease using machine learning algorithms and an explainable AI method,” *Sci Rep*, vol. 14, no. 1, p. 23277, 2024, doi: 10.1038/s41598-024-74656-2.

[12] R. Indrakumari, T. Poongodi, and S. R. Jena, “Heart Disease Prediction using Exploratory Data Analysis,” *Procedia Comput Sci*, vol. 173, pp. 130–139, 2020, doi: https://doi.org/10.1016/j.procs.2020.06.017.

[13] V. K and S. Jyothi, “Decision Support System for Congenital Heart Disease Diagnosis based on Signs and Symptoms using Neural Networks,” *Int J Comput Appl*, vol. 19, Apr. 2011, doi: 10.5120/2368-3115.

[14] M. Dorraki *et al.*, “Improving Cardiovascular Disease Prediction With Machine Learning Using Mental Health Data: A Prospective UK Biobank Study,” *JACC: Advances*, vol. 3, no. 9, Part 2, p. 101180, 2024, doi: https://doi.org/10.1016/j.jacadv.2024.101180.

[15] C. Dangare and S. Apte, “Improved Study of Heart Disease Prediction System using Data Mining Classification Techniques,” *Int J Comput Appl*, vol. 47, pp. 44–48, Apr. 2012, doi: 10.5120/7228-0076.

[16] Srınıvasa Rao B., “A New Ensenble Learning based Optimal Prediction Model for Cardiovascular Diseases,” *E3S Web Conf.*, vol. 309, p. 1007, 2021, doi: 10.1051/e3sconf/202130901007.

[17] A. Tiwari, A. Chugh, and A. Sharma, “Ensemble framework for cardiovascular disease prediction,” *Comput Biol Med*, vol. 146, p. 105624, 2022, doi: https://doi.org/10.1016/j.compbiomed.2022.105624.

[18] A. Alqahtani, S. Alsubai, M. Sha, L. Vilcekova, and T. Javed, “Cardiovascular Disease Detection using Ensemble Learning,” *Comput Intell Neurosci*, vol. 2022, no. 1, p. 5267498, 2022, doi: https://doi.org/10.1155/2022/5267498.

[19] G. Alshehri and H. Alharbi, “Prediction of Heart Disease using an Ensemble Learning Approach,” *International Journal of Advanced Computer Science and Applications*, vol. 14, Apr. 2023, doi: 10.14569/IJACSA.2023.01408118.