AI–POWERED EARLY DIAGNOSIS AND PERSONALISED HEALTH RECOMMENDATIONS FOR CORONARY ARTERY DISEASE (CAD) USING PREDICTIVE ANALYTICS

A PROJECT REPORT

Submitted by

SIDDHARTH VATS [Reg No:RA2111003010606]

KHAYATI SHARMA [Reg No:RA2111003010710]

Under the Guidance of

Dr. M. Revathi

Assistant Professor, Department of Computing Technologies

in partial fulfillment of the requirements for the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTING TECHNOLOGIES

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR– 603 203

MAY 2025

SRM

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR–603 203

BONAFIDE CERTIFICATE

Certified that 18CSP109L project report titled **"AI-Powered Early Diagnosis and Personalized Health Recommendations for Coronary Artery Disease (CAD) using Predictive Analytics "** is the bonafide work of **Siddharth Vats [RA2111003010606]** and **Khayati Sharma [RA2111003010710]** who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported here in does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion for this or any other candidate.

Department of Computing Technologies Kattankulathur 603 203 SRM IST

Dr. M. Revathi
SUPERVISOR & PANEL HEAD
Assistant Professor
Department of Computing Technologies

Dr. G. Niranjana
PROFESSOR & HEAD
Department of Computing Technologies

INTERNAL EXAMINER 13/5/25

EXTERNAL EXAMINER 13/5/25

Dr. D. Kavitha

Department of Computing Technologies
SRM Institute of Science and Technologies
Own Work* Declaration Form

Degree/ Course	: B.Tech/ Computer Science and Engineering
Student Name	**: Siddharth Vats, Khayati Sharma**
Registration Number	: RA2111003010606, RA2111003010710
Title of Work	**: AI-Powered Early Diagnosis and Personalised Health Recommendations for Coronary Artery Disease (CAD) using Predictive Analytics**

We hereby certify that this assessment compiles with the University's Rules and Regulations relating to Academic misconduct and plagiarism**, as listed in the University Website, Regulations, and the Education Committee guidelines.

We confirm that all the work contained in this assessment is my / our own except where indicated, and that we have met the following conditions:

- Clearly referenced / listed all sources as appropriate
- Referenced and put in inverted commas all quoted text (from books, web, etc)
- Given the sources of all pictures, data etc. that are not my own
- Not made any use of the report(s) or essay(s) of any other student(s) either past or present
- Acknowledged in appropriate places any help that I have received from others (e.g. fellow students, technicians, statisticians, external sources)
- Compiled with any other plagiarism criteria specified in the Course handbook / University website

We understand that any false claim for this work will be penalized in accordance with the University policies and regulations.

DECLARATION:

We are aware of and understand the University's policy on Academic misconduct and plagiarism and we certify that this assessment is our own work, except where indicated by referring, and that we have followed the good academic practices noted above.

RA2111003010710
KHAYATI SHARMA Khayati

RA2111003010606
SIDDHARTH VATS Sidd

If you are working in a group, please write your registration numbers and sign with the date for every student in your group.

ACKNOWLEDGEMENTS

We express our humble gratitude to **Dr. C. Muthamizhchelvan**, Vice-Chancellor, SRM Institute of Science and Technology, for the facilities extended for the project work and his continued support.

We extend our sincere thanks to **Dr. Leenus Jesu Martin M**, Dean-CET, SRM Institute of Science and Technology, for his invaluable support.

We wish to thank **Dr. Revathi Venkataraman**, Professor and Chairperson, School of Computing, SRM Institute of Science and Technology, for her support throughout the project work.

We encompass our sincere thanks to **Dr. M. Pushpalatha**, Professor and Associate Chairperson - CS, School of Computing and **Dr. C. Lakshmi,** Professor and Associate Chairperson - AI, School of Computing, SRM Institute of Science and Technology, for their invaluable support.

We are incredibly grateful to our Head of the Department, **Dr. G. Niranjana**, Professor, Department of Computing Technologies, SRM Institute of Science and Technology, for her suggestions and encouragement at all the stages of the project work.

We want to convey our thanks to our Project Coordinators, Panel Head, and Panel Members Department of Computing Technologies, SRM Institute of Science and Technology, for their inputs during the project reviews and support.

We register our immeasurable thanks to our Faculty Advisors, **Ms. S. S. Saranya**, Assistant Professor and **Ms. P. Nithyakani,** Assistant Professor, Department of Computing Technologies, SRM Institute of Science and Technology, for leading and helping us to complete our course.

Our inexpressible respect and thanks to our guide, **Dr. M. Revathi**, Assistant Professor, Department of Computing Technologies, SRM Institute of Science and Technology, for providing us with an opportunity to pursue our project under her mentorship. She provided us with the freedom and support to explore the research topics of our interest. Her passion for solving problems and making a difference in the world has always been inspiring.

13/5/25 **SIDDHARTH VATS [RA2111003010606]**

13/5/25 **KHAYATI SHARMA [RA2111003010710]**

ABSTRACT

Cardiovascular diseases, especially coronary artery disease, have been the leading cause of deaths worldwide. Hence, the early detection and personalized treatment strategies for coronary artery disease (CAD) are now needed more than ever. This project suggests an AI-driven framework based on machine learning algorithms for the timely risk detection of CAD based on 13 clinical parameters. The framework utilizes an ensemble learning method that includes logistic regression, random forest and support vector machine classifiers which improve the prediction accuracy. The proposed system also comes equipped with an AI chemist assistant which helps users specifically with medication-related queries, such as drug interactions, administration instructions and prescription explanations. This feature managed textual and visual inputs using Google's generative AI (Gemini), thus allowing users to upload photos or ask questions and receive instant assistance. Adding to the suite of features, the system also comes with a health recommendation engine, which provides users with certain lifestyle changes to be adopted based on the risk assessment, and a chatbot which answers more general questions that the user might have related to his diet, exercise, sleep or any symptoms being experienced. This approach provides an integrated preventative care solution by combining CAD risk prediction, AI-based medication assistance, a coronary health chatbot and lifestyle recommendations. Apart from improving accessibility, the suggested solution also improves patient education and enables more informed decision-making in cardiovascular disease care. The experimental results show the effectiveness of the ensemble model in risk prediction of CAD and AI chemist's capacity to help users with medication-related concerns. This integrated system aims to combine AI-based diagnostic techniques with medicative and general heart-health assistance, thus improving the efficiency and accessibility of healthcare services.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

ML	Machine Learning
AI	Artificial Intelligence
NLP	Natural Language Processing
CAD	Coronary Artery Disease
CVD	Cardiovascular Disease
ECG	Electrocardiogram
AUC	Area Under Curve
ROC	Receiver Operating Characteristic
PR-AUC	Precision-Recall Area Under Curve
UI	User Interface
API	Application Programming Interface
SaaS	Software as a Service
SMOTE	Synthetic Minority Oversampling Technique
WHO	World Health Organization
UCI	University of California, Irvine (UCI ML Repository)
HPO	Hyperparameter Optimization
TPR	True Positive Rate
FPR	False Positive Rate
HIPAA	Health Insurance Portability and Accountability Act
GDPR	General Data Protection Regulation

CHAPTER 1

INTRODUCTION

1.1 General

Cardiovascular diseases (CVDs) are among the leading causes of death globally, claiming millions of lives each year. According to the World Health Organization (WHO), CVDs are responsible for nearly 17.9 million deaths annually, representing approximately 32% of all global deaths. Within this broad category, Coronary Artery Disease (CAD) is the most common type. CAD occurs when the coronary arteries — the blood vessels that supply oxygen-rich blood to the heart muscle — become narrowed or blocked due to the buildup of cholesterol, fatty deposits (plaques), and other substances on the inner walls of the arteries. This condition, known as atherosclerosis, reduces blood flow to the heart, leading to chest pain (angina), shortness of breath, heart attacks, and, in severe cases, sudden cardiac death.

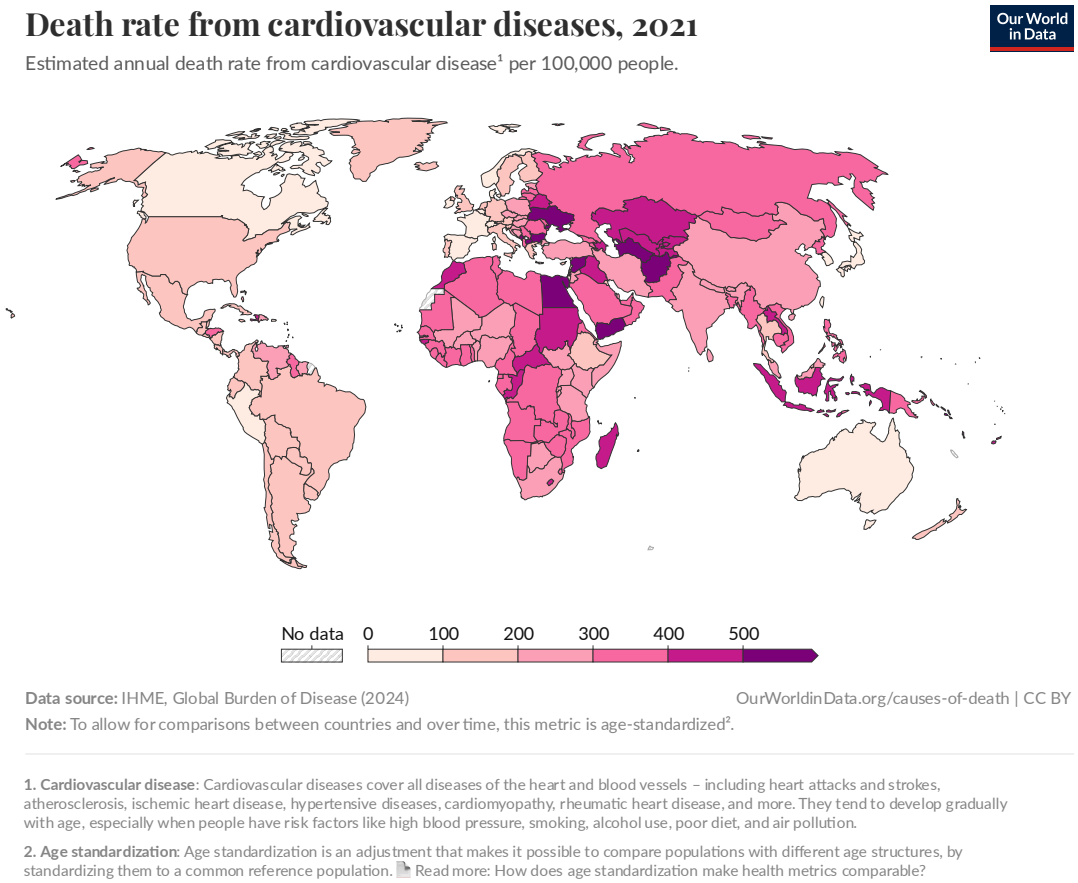


Figure 1.1 Global deaths in 2021 due to cardiovascular diseases

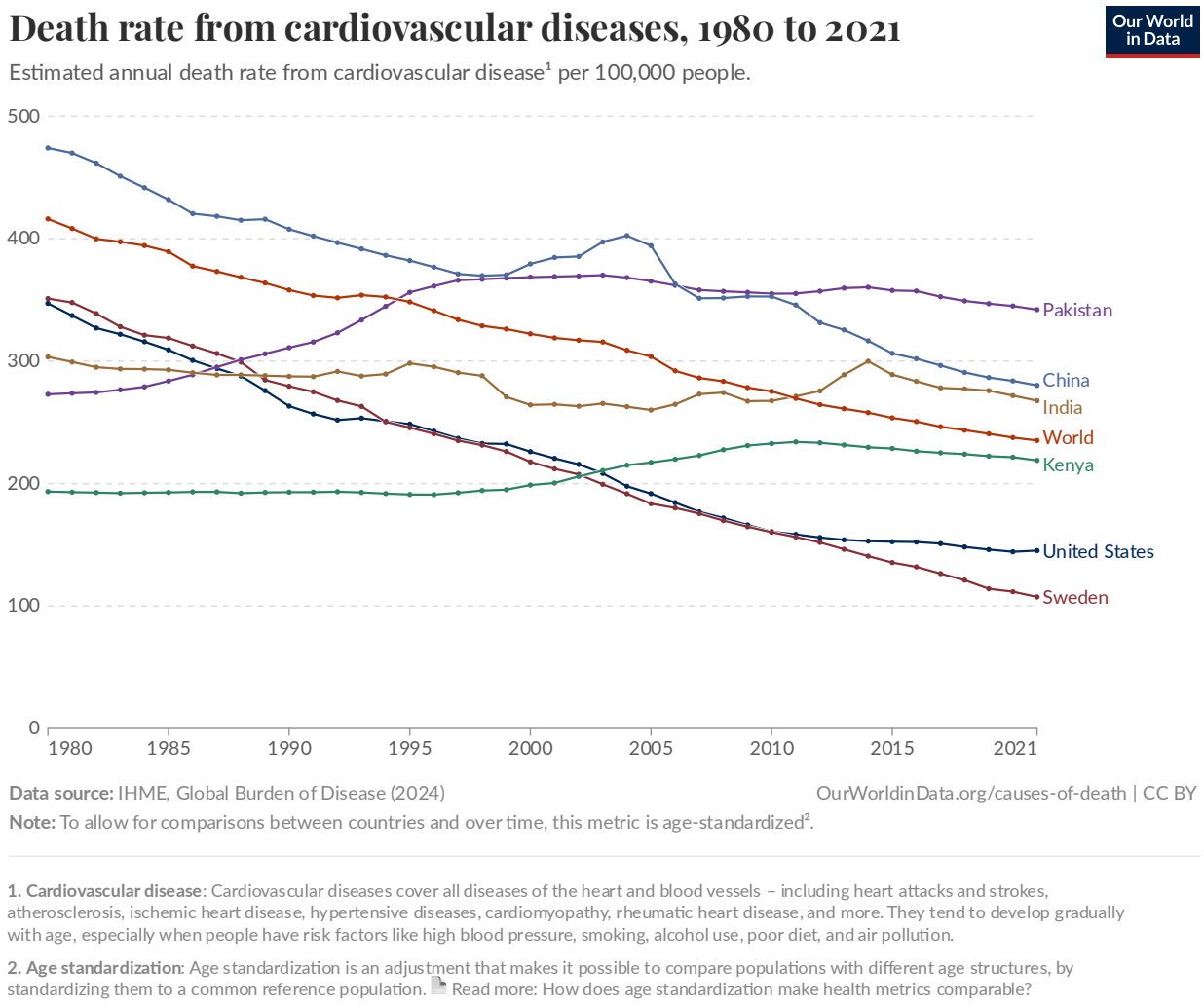


Figure 1.2 Death rates from cardiovascular disease in different countries from 1980 to 2021

There are multiple factors that contribute to the development of CAD, including high blood pressure, high cholesterol, diabetes, smoking, sedentary lifestyle, unhealthy diet, excessive alcohol consumption, and genetic predisposition. The combination of these factors accelerates the damage to the inner linings of the arteries, promoting plaque accumulation over time. The insidious nature of CAD lies in its slow, often symptomless progression, making early diagnosis critical for preventing life-threatening outcomes.

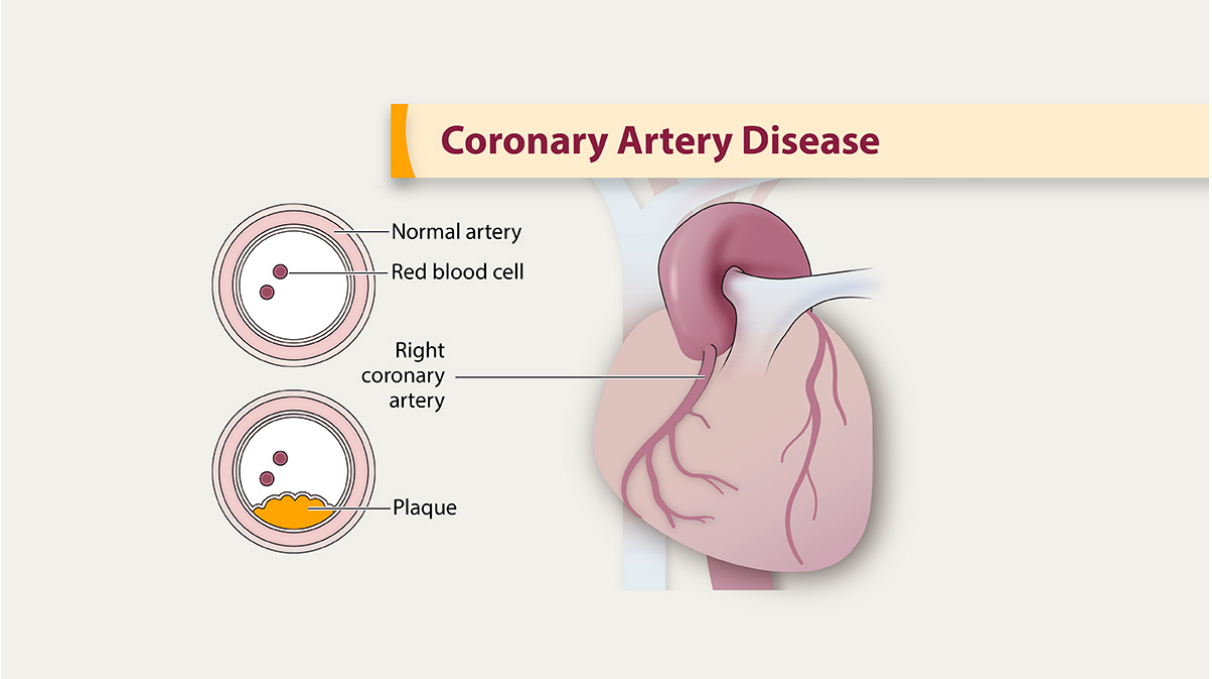


Figure 1.3 Coronary artery anatomy and atherosclerosis development

Prevention strategies focus on controlling risk factors through lifestyle modifications such as maintaining a balanced diet, engaging in regular physical activity, managing stress, avoiding tobacco, moderating alcohol intake, and adhering to prescribed medications for blood pressure, cholesterol, or diabetes management. Despite awareness efforts, early detection remains a challenge because standard diagnostic tools like stress tests, coronary angiography, and echocardiograms are expensive, require specialized facilities, and are generally only prescribed once symptoms emerge.

The purpose of this project is to address these challenges through an AI-powered system capable of early prediction of CAD risk using basic clinical parameters that are easily obtainable during routine health check-ups. The system uses predictive analytics through machine learning algorithms to assess an individual's risk level, allowing for early intervention, lifestyle changes, or medical treatment long before severe symptoms appear. Furthermore, the solution integrates an AI Chemist Assistant — a module powered by Google Gemini — that educates patients about their prescribed medications, warns them of potential drug interactions, and assists in managing pharmaceutical therapy. By combining diagnosis, pharmaceutical guidance, and personalized health recommendations into a single platform, the project offers a comprehensive preventative care solution that could significantly reduce the mortality rates associated with coronary artery disease.

1.2 Problem Statement

Coronary Artery Disease continues to claim millions of lives annually, with many cases going undiagnosed until critical, irreversible damage occurs. Existing diagnostic methods typically require advanced imaging techniques, specialist consultations, and invasive procedures, which are not only expensive but also time-consuming. In low-resource settings, the problem becomes even more acute due to lack of access to cardiologists, sophisticated diagnostic equipment, and timely healthcare services.

Another dimension of the problem lies in pharmaceutical literacy among patients. Even when treatment is prescribed, many individuals struggle to understand their medications, leading to improper adherence, harmful drug interactions, and worsening of health conditions. There is currently a noticeable gap in patient support systems that bridge medical diagnosis, medication guidance, and lifestyle management in an integrated and accessible manner.

Thus, there is a critical need for an intelligent, scalable, and user-friendly system that not only predicts the risk of CAD but also assists patients in managing their medication and overall heart health. This project directly addresses this gap by providing a technology-driven, holistic healthcare solution that combines predictive modelling with AI-powered assistance.

1.3 Motivation

The motivation behind this project stems from an urgent societal need to reduce the burden of cardiovascular diseases by utilizing the capabilities of artificial intelligence. In recent years, machine learning models have shown remarkable promise in pattern recognition tasks, including medical diagnostics. However, there remains an underutilization of AI's full potential in combining diagnosis with patient support and education.

The vision for this project is to make preventive healthcare more proactive, accessible, and engaging. By offering an early warning system for CAD based on routine clinical parameters, we aim to intervene before the disease reaches advanced stages. By integrating AI-based medication guidance and health advisory systems, the project aspires to bridge the knowledge gap between healthcare providers and patients.

Additionally, there is a strong personal and professional motivation to contribute meaningfully to the application of AI in life-saving domains like healthcare. This project is not merely a technical

endeavour; it is a step toward empowering individuals with the tools they need to understand and take control of their own heart health, improving outcomes at both the individual and community level.

1.4 Sustainable Development Goal

After careful consideration, this project was developed under the theme of Improving Mortality, which corresponds to the broader objective of reducing deaths caused by non-communicable diseases such as cardiovascular conditions. Cardiovascular diseases, and specifically Coronary Artery Disease (CAD), remain a major contributor to global mortality rates, often resulting in sudden deaths due to undetected or untreated conditions. The aim of this project is to directly contribute to the reduction of mortality rates by enabling early risk detection of CAD through the use of advanced machine learning techniques. By offering an accessible, non-invasive, and AI-driven risk assessment tool, the project supports timely intervention, which is crucial for preventing fatal outcomes associated with heart disease. Furthermore, the integration of an AI Chemist Assistant and a lifestyle recommendation engine enhances patient education and medication adherence, which are essential for long-term disease management and survival. Through early diagnosis, informed decision-making, and continuous health guidance, this project aims to significantly impact the survival rates of individuals at risk of coronary conditions.

Thus, the project is closely aligned with the specific Sustainable Development Goal category of Improving Mortality, with a strong emphasis on leveraging technology to lower the number of preventable deaths. By empowering users with early warnings, accurate health information, and actionable recommendations, the project aspires to foster a measurable reduction in CAD-related fatalities and contribute toward healthier, more resilient communities.

1.5 Applications

This project has multiple applications in both clinical and non-clinical environments. In clinical settings such as hospitals, diagnostic centers, and cardiology clinics, it can be deployed as a preliminary risk screening tool for individuals visiting for regular check-ups. General practitioners could use the system to identify patients who require further cardiac evaluation without immediately resorting to costly diagnostic tests.

In telemedicine and remote healthcare services, where access to cardiologists may be limited, the tool could empower healthcare workers to screen patients efficiently and recommend appropriate referrals. The AI Chemist Assistant also has applications in outpatient management, where doctors can ensure that patients receive automated, clear, and customized explanations about their prescriptions.

In addition, the project has significant potential for integration into personal health apps and wearable devices. Integration with health-monitoring gadgets like smartwatches could allow for even more real-time heart health monitoring, making preventative healthcare more personalized and accessible.

1.6 Software Requirements Specification

The successful development and deployment of the system require specific software tools and configurations. The following software requirements were identified:

- Operating System: Windows 10 or higher, or a compatible Linux distribution (Ubuntu 20.04+ recommended for production deployment).
- Programming Language: Python 3.8 or higher.
- Machine Learning Libraries: scikit-learn, pandas, numpy, joblib.
- Web Framework: Streamlit, to create an interactive web application interface.
- Visualization Libraries: Matplotlib, Seaborn for plotting confusion matrices, ROC curves, precision-recall curves.
- AI Integration: Google Gemini API for multimodal AI (text and image input handling), OpenAI's GPT-4 model for general chatbot functionalities.
- Other Tools: Git for version control, Jupyter Notebook for experimentation and visualization during model development.
- Deployment Tools: Docker (optional) for containerized deployment, making the solution scalable and portable across environments.
- Database (optional): SQLite or Firebase if user data storage is needed for future expansion.

CHAPTER 2

LITERATURE SURVEY

2.1 Literature Survey

Cardiovascular disease prediction has been a major focus of research with several studies using machine learning to improve diagnostic performance and detect diseases early. Many have attempted various models, datasets, and methods to enhance predictive systems. Seckeler and Hoke [1] presented a detailed overview of the epidemiology of rheumatic heart disease and its burden and long-term public health impact. Early detection remains a recurring theme in heart disease research, particularly within vulnerable and underserved populations. Several studies underscore the urgency of diagnosing cardiovascular conditions before they progress to severe stages. For instance, Gaziano et al. [2] raised concern over the growing burden of coronary artery disease (CAD) in low- and middle-income countries, drawing attention to how limited access to timely care often exacerbates patient outcomes. Their findings emphasize the need for scalable and proactive diagnostic solutions in such regions.

Building on this Boukhatem et al. [3] investigated machine learning-based frameworks for predicting cardiac disease, illustrating how heterogeneous feature extraction techniques may stabilize models. Their findings suggested that advanced machine learning techniques, when used appropriately, may advance early diagnosis and treatment. In a related effort, Jindal et al. [4] examined a diverse set of machine learning algorithms for predicting heart disease, identifying critical variables with a strong impact on prediction accuracy. The study conducted by Ramalingam et al. [5] focused on balancing performance with interpretability, a crucial consideration in medical decision-making environments where transparency is key. Their findings emphasized the trade-off between model complexity and usability. Weng et al. [6] described the integration of machine learning with traditional clinical information to estimate cardiovascular risk, underlining how artificial intelligence-based methods can recognize patients at risk prior to clinical symptoms appearing.

Fatima and Pasha [7] conducted a comparative analysis evaluating a wide range of machine learning methods, offering insight into accuracy trends and practical limitations across different use cases. Their paper emphasized the pros and cons of various algorithms in terms of accuracy, computational complexity, and interpretability. Vembandasamy et al. [8] explained the applicability of the Naïve

Bayes algorithm to detect cardiac disease. From their findings, Naïve Bayes offers an efficient and interpretable strategy but its accuracy is frequently surpassed by more sophisticated models. Chaurasia and Pal [9] applied data mining to identify cardiac conditions and established the capability of ensemble learning methods to enhance predictive accuracy. Bhatt et al. [10] subsequently employed a 70,000 sample dataset and compared different models to determine the most appropriate method to predict heart disease. A number of works, including those by Patel et al. [11], explored the impact of robust pre-processing and feature selection techniques on predictive accuracy. One of their key contributions was a novel approach to filtering out redundant features, ultimately leading to more refined and computationally efficient models. Rindhe et al. [12] investigated different machine learning architectures for the prediction of the cardiac disease. Their study relied on the Cleveland Heart Disease dataset [13], a benchmark in cardiovascular research. The study proved that decision tree and support vector machine models were of high accuracy when they were trained with hyperparameters that were optimized. Tithi et al. [14] concentrated their research on the assessment of ECG data and the prediction of cardiovascular disease based on the implementation of six supervised learning models. Their research highlighted the implementation of feature selection and pre-processing methods in enhancing the performance of the model. Garg et al. [15] focused their attention on supervised learning techniques, evaluating how different model architectures and pre-processing pipelines influenced diagnostic precision. Their research compared many classification models and concluded that ensemble-based models have an incredible impact on enhancing predictive accuracy compared to conventional machine learning models.

These papers together show the shifting paradigm of cardiac disease prediction using machine learning approaches.

2.2 Existing System

The early detection and management of coronary artery disease (CAD) have traditionally relied on a combination of clinical evaluations, diagnostic imaging, biochemical testing, and physician-led interpretation. In current medical practice, a wide array of established tools and techniques are employed for CAD diagnosis and patient management, each with its advantages, limitations, and specific clinical contexts.

One of the most commonly used approaches for diagnosing CAD involves risk factor assessment during clinical check-ups. Physicians evaluate conventional risk parameters such as age, gender,

smoking history, hypertension, hyperlipidaemia (high cholesterol levels), diabetes mellitus, family history of cardiovascular disease, and obesity. Clinical tools like the Framingham Risk Score (FRS) and SCORE (Systematic Coronary Risk Evaluation) charts help estimate an individual's 10-year risk of developing heart disease based on these factors. While these tools provide valuable guidelines, they are generalized for population-level risk estimation and often lack precision when applied to diverse, modern-day patient populations with varying genetic, ethnic, and lifestyle backgrounds.

In addition to risk scoring, non-invasive diagnostic tests are widely employed. These include electrocardiograms (ECGs), exercise stress tests, echocardiography, and coronary computed tomography angiography (CCTA). ECGs detect electrical abnormalities that may suggest ischemia, while exercise stress tests monitor heart response under physical exertion. Echocardiography provides a sonographic image of the heart's structure and function, and CCTA offers detailed imaging of coronary vessels to visualize blockages. When non-invasive methods are inconclusive or suggest significant risk, more advanced techniques such as invasive coronary angiography are used. This "gold standard" method provides direct visualization of coronary arteries and allows for immediate interventions like stent placement if necessary. However, coronary angiography is costly, exposes patients to ionizing radiation and contrast dye risks, and carries procedural risks such as bleeding, arrhythmias, or heart attacks during the procedure.

While these diagnostic systems form the backbone of modern cardiology, several limitations have become increasingly evident:
Firstly, these systems are largely reactive rather than proactive. They are typically employed after patients report symptoms such as chest pain, fatigue, or shortness of breath, by which time coronary blockages may already be significant. Consequently, they miss opportunities for early intervention during the asymptomatic or subclinical phases of the disease when lifestyle modifications or preventive medication could have dramatically altered disease progression.

Secondly, the cost and complexity associated with imaging-based diagnostics and specialist consultations make them inaccessible for a substantial portion of the global population, particularly in low- and middle-income countries. The heavy reliance on hospital infrastructure, trained personnel, and expensive equipment results in systemic barriers to timely and widespread CAD risk screening.

Thirdly, risk score models like Framingham are often criticized for their inflexibility. They do not dynamically incorporate individualized clinical parameters or new biomarkers discovered in contemporary research. They are also primarily designed around Western populations and may not accurately represent risk profiles in Asian, African, or indigenous communities.

Moreover, there is a lack of integrated patient support systems within the existing diagnostic landscape. Even when patients are diagnosed with CAD, they are often discharged with complex medication regimens that they may not fully understand. Insufficient pharmaceutical literacy among patients leads to poor medication adherence, drug misuse, and harmful interactions that could have been avoided with better support and education.

Recognizing these challenges, there has been a recent surge in attempts to incorporate machine learning and artificial intelligence into cardiovascular disease prediction. Some research efforts have used basic classifiers like decision trees, support vector machines (SVMs), or logistic regression models to predict heart disease based on retrospective datasets like the Cleveland Heart Disease dataset. While promising, most of these systems remain experimental, lack real-world deployment readiness, and typically focus only on predictive analytics without integrating pharmaceutical education, lifestyle guidance, or user interactivity.

2.3 Existing Systems vs Proposed Work

While conventional methods for diagnosing and managing coronary artery disease (CAD) have been instrumental in saving lives, they were fundamentally designed for a healthcare model that is hospital-centric, reactive, and heavily dependent on specialized human expertise. The growing burden of cardiovascular diseases, coupled with disparities in healthcare accessibility and the increasing complexity of patient needs, has revealed clear gaps that traditional systems cannot address adequately. This reality underscores the need for a new, more scalable, predictive, and patient-empowering approach — a need that our proposed system is uniquely designed to fulfil.

The proposed AI-powered system brings a paradigm shift in the early detection and management of CAD by transitioning from late-stage diagnosis to proactive risk prediction. Unlike conventional approaches, which typically intervene after symptoms arise, our system identifies individuals at risk before clinical manifestations occur. By leveraging ensemble machine learning models trained on common clinical parameters, it empowers healthcare providers, and even individuals themselves, to

implement preventative strategies early, thereby potentially avoiding life-threatening cardiac events altogether.

Beyond prediction, the proposed system is distinguished by its emphasis on holistic patient engagement. In current practices, once a diagnosis is made, patients are often left navigating complex medication regimens and lifestyle adjustments with minimal ongoing support. Our platform bridges this gap by integrating an AI Chemist Assistant, capable of interpreting prescriptions, advising on drug usage, and warning against harmful interactions. This functionality transforms the patient experience from passive compliance to active understanding, significantly enhancing medication adherence and minimizing avoidable complications. Another critical innovation lies in the accessibility and scalability of our solution. Traditional CAD diagnostics are constrained by physical infrastructure, imaging equipment, specialized labs, and cardiology departments, all of which are scarce in rural, underserved, or economically disadvantaged areas. Our proposed system, being lightweight, web-based, and deployable even in low-resource environments, democratizes access to sophisticated risk assessment tools. It reduces dependency on geographic location or institutional resources, making early cardiovascular health management possible for a much broader population.

Furthermore, while existing systems rely on static risk models designed decades ago, our system introduces adaptive intelligence. Through ensemble machine learning, which combines logistic regression, random forest, and support vector machine classifiers using soft voting, the platform dynamically adjusts its predictions based on evolving datasets and diverse population profiles. This ensures that predictions are not only more accurate but also sensitive to modern lifestyle patterns, genetic diversity, and emerging clinical insights, factors static models fail to accommodate. Our project is also built around the principle of patient-centric healthcare delivery. Traditional models often prioritize disease-centric interventions, focusing primarily on treatment rather than empowerment or prevention. By offering users lifestyle recommendations tailored to their personal risk profiles and a conversational heart health assistant for general advice, our system redefines engagement, enabling individuals to take charge of their own cardiovascular well-being with clarity and confidence. The proposed approach, therefore, is not merely a technical enhancement over existing methods; it represents a necessary evolution in how CAD risk is identified, communicated, and managed in an increasingly interconnected and resource-variable world.

CHAPTER 3

SYSTEM ANALYSIS

3.1 Dataset and Preprocessing

The success of any predictive analytics system in healthcare depends significantly on the quality, structure, and relevance of the underlying dataset. For this project, the Cleveland Heart Disease dataset [13] was selected as the primary source of clinical data. This dataset, widely regarded as a benchmark in cardiovascular research, is hosted by the UCI Machine Learning Repository (DOI: 10.24432/C52P4X) and contains anonymized patient records with various medical attributes that have been clinically observed to correlate with coronary artery disease (CAD).

3.1.1 Dataset Overview

The dataset consists of 303 patient records, each described by 13 input features and one target output variable. These features capture both demographic information and clinical indicators, making it well-suited for machine learning-based risk prediction tasks. The target variable originally ranges from 0 to 4, where 0 indicates the absence of CAD and 1–4 represent varying levels of disease severity.

To streamline the classification task and focus on the distinction between healthy and at-risk individuals, the problem was reformulated as a binary classification task. All non-zero target values (1–4) were consolidated into a single class labeled as "CAD Present," while the zero value was maintained as "No CAD."

3.1.2 Features Used

The dataset includes a mix of numerical and categorical variables, summarized as follows:

- Demographic attributes:
    - Age
    - Sex
- Clinical and biochemical indicators:
    - Chest Pain Type (cp)
    - Resting Blood Pressure (trestbps)

- Serum Cholesterol (chol)
- Fasting Blood Sugar (fbs)
- Resting Electrocardiographic Results (restecg)
- Maximum Heart Rate Achieved (thalach)
- Exercise-Induced Angina (exang)

- Imaging and genetic factors:
  - ST Depression Induced by Exercise (oldpeak)
  - Slope of the Peak Exercise ST Segment (slope)
  - Number of Major Vessels Colored by Fluoroscopy (ca)
  - Thalassemia Type (thal)

These features provide a rich, multivariate basis for machine learning models to detect subtle patterns associated with cardiovascular disease.

3.1.3 Preprocessing Workflow

Given the presence of mixed data types and the clinical nature of the dataset, a structured and thorough preprocessing pipeline was applied to improve data quality and ensure compatibility with machine learning algorithms:

a) Handling missing values

The dataset was examined for any null or missing entries. Instances with missing data in critical variables such as 'ca' (number of major vessels) and 'thal' (thalassemia type) were either imputed or excluded based on their frequency and importance. This step ensured that the models would not encounter inconsistencies during training or inference.

b) Categorical encoding

Categorical variables such as chest pain type, sex, resting ECG, slope, and thalassemia were encoded into numerical form using label encoding or one-hot encoding, depending on the variable's nature. This transformation allows the algorithms to process non-numeric data meaningfully without assuming ordinal relationships where none exist.

c) Normalization and scaling

Continuous features like cholesterol levels, resting blood pressure, and maximum heart rate were normalized using techniques such as min-max scaling or z-score normalization. This

standardization ensures that all features contribute proportionally to the training process, preventing the models from being biased toward attributes with larger ranges.

d) Class balancing consideration

Since CAD-positive cases outnumber CAD-negative cases slightly in the original dataset, minor class imbalance was addressed during model evaluation using performance metrics like F1-score and ROC-AUC, which are more informative than accuracy alone in imbalanced scenarios.

e) Train - test split

To evaluate the model's generalizability, the dataset was split into 80% training data and 20% testing data, with stratified sampling applied to preserve the class distribution across both subsets. This ensures that the model sees a representative sample during both training and evaluation phases.

Through this systematic preprocessing pipeline, the dataset was made robust and machine-learning-ready. The transformations applied not only enhanced the integrity of the data but also contributed directly to improving the accuracy, recall, and overall reliability of the ensemble learning model used in the later stages of the project.

3.2 Model Selection and Training

The selection and training of an appropriate machine learning model are central to the effectiveness of any predictive analytics system. In the context of coronary artery disease (CAD) risk prediction, model performance is critically important because diagnostic misclassifications, especially false negatives, can lead to delayed treatment and severe clinical outcomes. With this in mind, the modelling strategy for this project prioritized both accuracy and recall, while also maintaining interpretability, scalability, and robustness against overfitting.

To achieve these objectives, the system employs a Voting Classifier ensemble, which combines three different types of base classifiers: Logistic Regression, Random Forest, and Support Vector Machine (SVM). The ensemble method was selected because it allows for the integration of diverse algorithmic perspectives, which improves generalization and reduces the likelihood of individual model bias.

3.2.1 Justification for Algorithm Selection

Each of the three base models was chosen to contribute specific strengths to the ensemble:

- Logistic Regression

  Logistic regression serves as a linear baseline model, offering high interpretability and ease of training. It models the log-odds of the target class as a linear combination of the input features, making it suitable for identifying simple linear relationships between predictors and CAD risk. It is computationally efficient and robust against overfitting when regularization is applied.

- Random Forest

  Random Forest is a powerful ensemble model based on decision trees. It mitigates the risk of overfitting common to individual trees by using bagging (bootstrap aggregation) and feature randomness. Each tree is trained on a random subset of the data and features, and final predictions are made based on majority voting. This contributes to robustness and helps capture non-linear feature interactions, which are common in medical datasets.

- Support Vector Machine

  SVMs are particularly effective in high-dimensional spaces and when the decision boundary between classes is not linearly separable. The use of kernel functions, such as the radial basis function (RBF), enables the SVM to project data into a higher-dimensional space where a hyperplane can effectively separate CAD and non-CAD cases. SVMs also excel at maximizing the margin between classes, leading to better generalization.

3.2.2 Ensemble Strategy: Soft voting

To leverage the strengths of each base model, the system uses a Soft Voting Classifier. In this ensemble method, rather than simply taking the majority class predicted by each model (hard voting), the predicted class probabilities are averaged across the classifiers, and the final class label is assigned based on the highest aggregated probability.

Soft voting has two main advantages:

- It retains more information about model confidence compared to hard voting.
- It allows more reliable decisions, especially in borderline cases where different models may disagree in classification.

Mathematically, the soft voting classifier computes the final predicted class ŷ as:

$$\hat{y} = \arg\max_{j} \left(\frac{1}{n} \sum_{i=1}^{n} P_{i,j} \right) \quad (1)$$

Where:

$P_{i,j}$ is the probability of the $j-th$ class predicted by the $i-th$ model.

n is the number of base models.

3.2.3 Model training pipeline

The training pipeline was structured as follows:

a) Data splitting

The dataset, after preprocessing, was divided using an 80-20 stratified split to ensure that both the training and test sets preserved the original class balance (CAD present vs. no CAD).

b) Model initialization and configuration

Each base model was initialized with a carefully selected set of hyperparameters to balance performance and overfitting risk:

- Logistic Regression: L2 regularization (Ridge), solver = 'liblinear'
- Random Forest: 100 estimators, max depth = 5 (tuned to reduce overfitting),random_state = 42
- SVM: Kernel = 'rbf', C = 1.0, gamma = 'scale'

c) Cross validation

A 5-fold cross-validation was conducted on the training set to validate model performance during training. The ensemble classifier consistently outperformed individual models in F1-score and ROC AUC across all folds.

d) Model aggregation

The models were wrapped in a Voting Classifier from scikit-learn with voting='soft'. Weights were optionally applied based on individual model performance to enhance ensemble accuracy, giving higher influence to models with stronger precision and recall scores.

e) Final Training

After cross-validation, the complete training set was used to train the final ensemble model. The model was then serialized and stored using the joblib library for efficient loading during deployment.

f) Testing and evaluation

The model's predictive performance was finally evaluated on the 20% test split, using metrics such as accuracy, precision, recall, F1-score, and ROC AUC. These results are discussed in detail in Section 4.2 (Model Evaluation).

3.2.4 Benefits of Ensemble Approach

The ensemble design significantly improves model generalizability. While individual models may perform well on specific subsets of data, they may also have unique weaknesses. For example:

- Logistic regression might fail to capture complex, non-linear relationships.
- Random forests may overfit if not carefully tuned.
- SVMs, while powerful, are computationally expensive and sensitive to parameter settings.

By integrating these models through a soft voting mechanism, the system:

- Reduces variance and bias, providing balanced performance.
- Improves robustness against outliers or noise in clinical data.
- Maximizes recall, which is vital in a medical setting where false negatives could be fatal.

In conclusion, the model selection and training process for this project is grounded in best practices from machine learning and tailored specifically to the medical context. The ensemble approach provides a reliable, accurate, and interpretable framework for early CAD risk prediction and forms the predictive core of the larger healthcare system.

3.3 Model Evaluation

Once the ensemble model was trained and optimized, its performance was rigorously evaluated using multiple metrics to ensure its effectiveness in predicting coronary artery disease (CAD). In medical applications, particularly those involving early disease detection, evaluation extends beyond mere accuracy. The consequences of misclassification — especially false negatives (i.e., failing to detect

CAD in an affected individual) — can be severe, making metrics like recall and ROC-AUC more clinically significant.

The evaluation process involved applying the trained model to the held-out 20% test dataset, which had not been seen during the training phase. The goal was to assess how well the model could generalize to new, unseen patient data.

3.3.1 Evaluation Metrics

- Accuracy

  Measures the overall correctness of the model by calculating the proportion of total correct predictions (both true positives and true negatives) among all predictions made.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

- Precision

  Indicates how many of the instances predicted as positive (CAD present) were actually correct. High precision is essential when false positives are costly or burdensome to patients.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

- Recall (Sensitivity)

  Measures the ability of the model to identify actual positive cases. In a medical context, recall is critical because failing to detect CAD can lead to life-threatening outcomes.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

- F1 Score

  Represents the harmonic mean of precision and recall. It balances the trade-off between the two and is especially useful in imbalanced datasets where accuracy alone may be misleading.

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5}$$

- ROC-AUC (Receiver Operating Characteristic – Area Under Curve)

  Measures the model's ability to distinguish between classes (CAD vs. no CAD) across all classification thresholds. AUC close to 1 indicates excellent discriminative ability.

- PR-AUC (Precision-Recall Area Under Curve)

  Particularly useful in imbalanced datasets. It plots precision versus recall and focuses on the model's performance on the positive (disease) class.

3.3.2 Confusion Matrix

To better understand the distribution of predictions and errors, a confusion matrix was generated. It details:

- True Positives (TP): CAD cases correctly identified.
- True Negatives (TN): Healthy individuals correctly classified.
- False Positives (FP): Healthy individuals incorrectly labelled as having CAD.
- False Negatives (FN): CAD cases missed by the model.

This matrix provides actionable insights into the nature of the model's mistakes and the trade-offs between sensitivity and specificity.

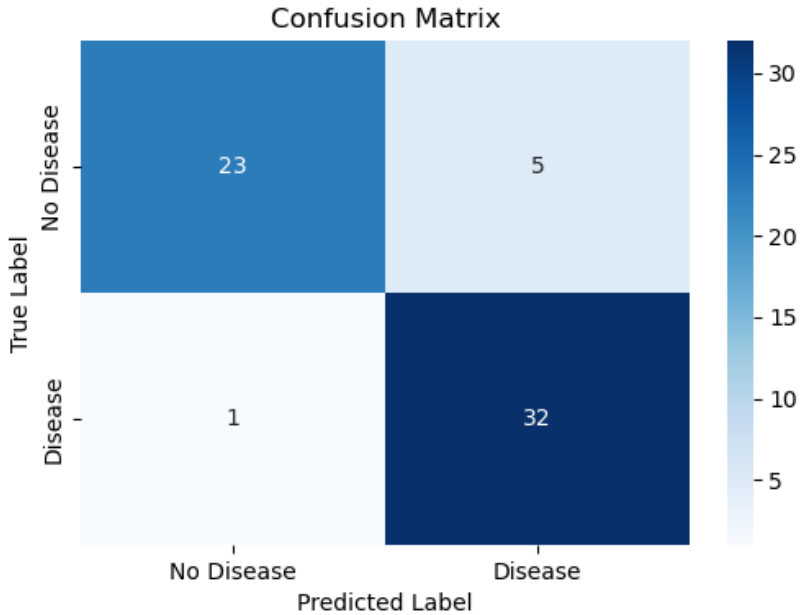


Figure 3.1 Confusion matrix of the hybrid ensemble model

3.3.3 Precision-Recall Curve

The Precision-Recall (PR) curve was plotted to visualize how precision and recall vary at different classification thresholds. The area under the PR curve (PR-AUC) gives a single scalar value summarizing the trade-off. This is especially important in medical applications, where positive class detection (CAD) is more important than balanced accuracy.

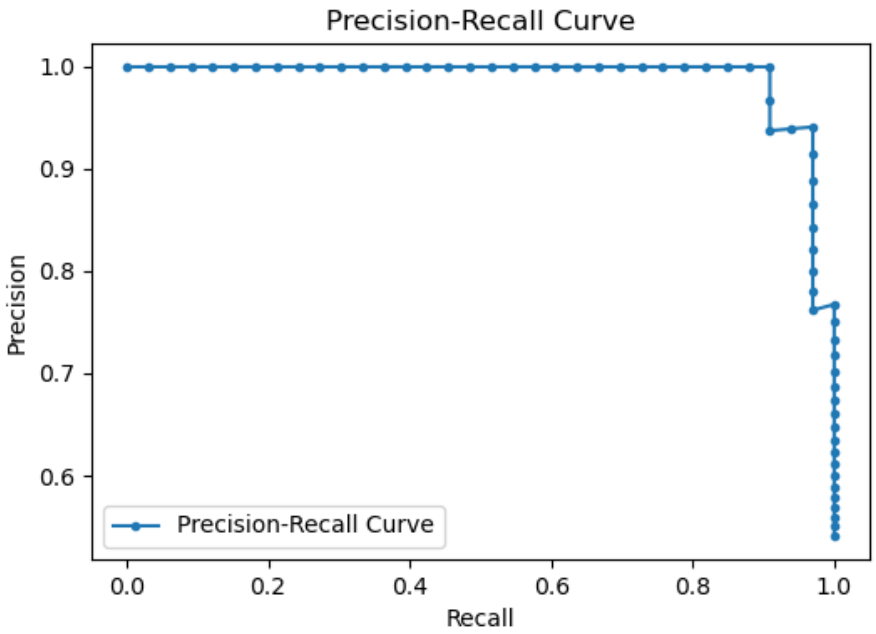


Figure 3.2 Precision-recall curve of the ensemble model

3.3.4 ROC Curve

The Receiver Operating Characteristic (ROC) curve plots the True Positive Rate (Recall) against the False Positive Rate. It illustrates the model's performance across a spectrum of threshold values. A higher curve and an area close to 1.0 indicate strong classification capability. The AUC score provides a threshold-independent metric that reflects the model’s ability to separate classes.

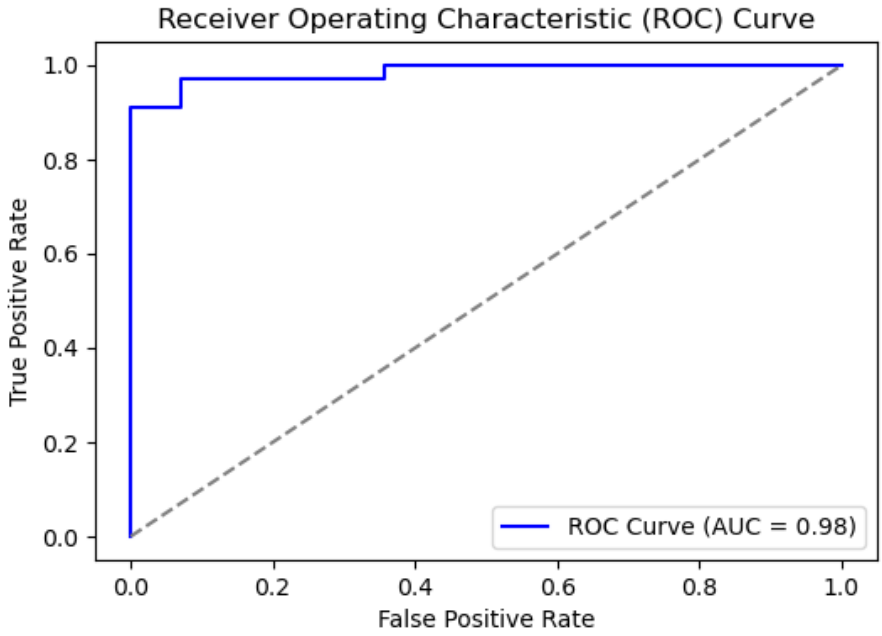


Figure 3.3 ROC curve of the ensemble model

3.3.5 Evaluation Results

The ensemble model was tested on the 20% test split and achieved the following performance scores:

Table 3.1 Evaluation results for the model

Metric	Score
Accuracy	0.9016
Precision	0.8649
Recall	0.9697
F1 Score	0.9143
ROC-AUC	0.9848

These results demonstrate strong overall performance, particularly in recall (96.97%), which is critical in CAD detection. The high ROC-AUC (98.93%) further suggests that the model is highly effective in distinguishing between diseased and non-diseased individuals.

The F1-score (91.43%) confirms that the model balances precision and recall well, while the precision value (86.49%) indicates that false positives are limited, thus minimizing unnecessary alarm for healthy users.

3.3.6 Interpretation

The results indicate that the ensemble model is robust, reliable, and clinically applicable. Its high recall ensures that most CAD cases are caught, minimizing the chances of a missed diagnosis. The high ROC-AUC and PR-AUC scores confirm that the model performs well across different thresholds, making it flexible for adaptation to different clinical risk tolerance levels (e.g., favoring sensitivity in high-risk patients, or precision in screening programs).

Additionally, the evaluation process confirmed the value of ensemble learning, as none of the individual base models consistently achieved this level of performance on their own. The combination of logistic regression, random forest, and SVM through soft voting offered a synergistic improvement in predictive strength.

CHAPTER 4

SYSTEM DESIGN AND IMPLEMENTATION

4.1 High-Level System Architecture

The proposed system adopts a modular, scalable, and interactive architecture to offer early detection of coronary artery disease (CAD) risk, AI-driven pharmaceutical assistance, and general health advisory services. Designed with user accessibility and performance in mind, the system leverages a combination of machine learning models, generative AI models, and web technologies to create a seamless and robust user experience. The architecture can be broadly divided into two layers: the High-Level System Architecture and the Low-Level Operational Flow. The high-level architecture consists of six major components, each performing a specific set of functions:

4.1.1 User Interface (UI) - Streamlit Application

The user interface acts as the main access point for system functionalities. Developed using Streamlit, it provides an intuitive and easy-to-use platform where users can interact with the system. Through the application, users can:

- Input clinical parameters for heart disease risk assessment.
- Upload images related to chemical compositions or prescriptions.
- Enter queries related to medications or heart health.

The UI provides two major operational pathways:

- Heart disease risk assessment
- AI chemist consultation

4.1.2 Heart Disease Prediction Module

This module is responsible for predicting the risk of coronary artery disease based on user-provided clinical data. It uses an ensemble machine learning model built with the following classifiers:

- Logistic regression
- Random forest
- Support Vector Machine

The ensemble uses a soft voting classifier, meaning it combines probability outputs from each model to make a final risk prediction. After evaluating the risk, the system also provides personalized lifestyle recommendations based on medical best practices for heart health management.

4.1.3 AI Chatbot for Heart Health Queries

An AI-based conversational agent is integrated into the system to handle user queries related to heart health, symptoms, preventive measures, and general wellness advice. Powered by OpenAI's GPT-3.5 Turbo, the chatbot provides dynamic, human-like responses. It enhances user engagement and offers real-time educational support beyond static prediction results.

4.1.4 AI Chemist Assistant

The AI Chemist module is a unique innovation within the system architecture. It performs:

- Analysis of user-uploaded text queries or prescription images.
- Interpretation of chemical and pharmaceutical data using multimodal AI.

This functionality is powered by Google Gemini 1.5 Pro Vision, a powerful generative AI model capable of handling both textual and visual data inputs. It helps users understand medication composition, possible side effects, drug interactions, and general usage instructions.

4.1.5 Data Processing and Storage Layer

To ensure smooth functioning and consistency, the system incorporates a dedicated data processing and storage layer:

- Data Preprocessing: Incoming user data undergoes cleaning, categorical encoding (for categorical variables like chest pain type and thalassemia), and normalization (for continuous variables like cholesterol and blood pressure).
- Model Storage: The trained ensemble machine learning model is stored in a serialized format using Joblib, allowing rapid loading and predictions during runtime.
- Data Storage Tables: Structured tables are created (if persistent storage is enabled) to store:
    - Patient inputs
    - Risk prediction outputs
    - Chatbot interaction logs (optional for future system improvement)

4.1.6 Backend Services and API Integrations

The system relies on backend integrations with external AI services for delivering advanced capabilities:

- Google Gemini API for text and image-based chemistry/pharmaceutical analysis.
- OpenAI GPT API for real-time health query responses through the chatbot.

These API integrations enable the system to offer rich, AI-powered interactions without heavy computational demands on local servers.

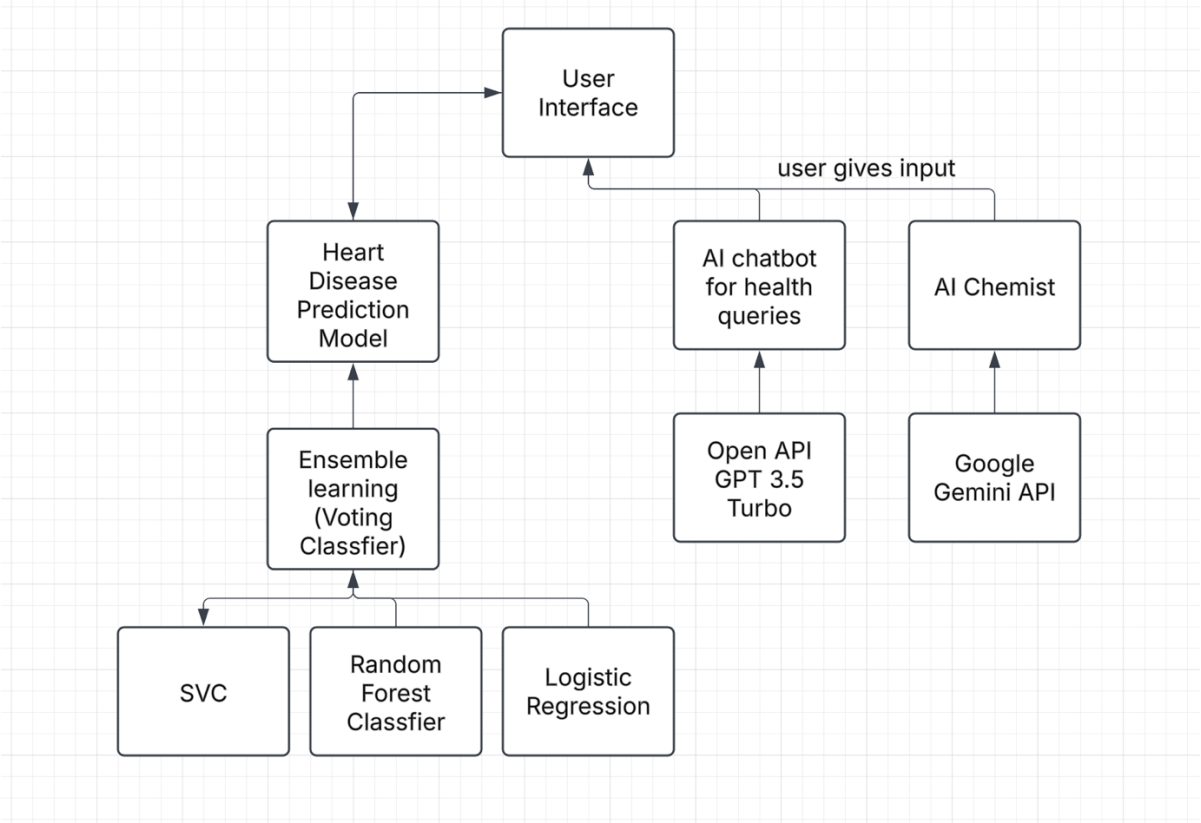


Figure 4.1 Working of the modules and components in the system

4.2 Low-level Operational Flow

The low-level architecture captures the step-by-step flow of data and interactions within the system, ensuring that every functionality is tightly coordinated.

4.2.1 User Flow

The user flow describes the sequence of interactions between the end user and the system. It outlines how users navigate through the application interface, input data, receive feedback, and interact with

various modules. The design prioritizes clarity, accessibility, and user empowerment, ensuring that even individuals with limited technical or medical background can efficiently use the platform.
The typical user journey through the system is composed of the following stages:

a) Launching the application:

The user initiates the process by accessing the web-based interface hosted via the Streamlit framework. The system can be accessed through a browser on any standard desktop or mobile device, ensuring platform independence and broad accessibility.

b) Dashboard presentation:

Upon loading, the main dashboard presents the user with two primary options, clearly labelled:

- "Heart Disease Risk Assessment"
- "AI Chemist"

These options are designed to direct users toward the functionality they require without unnecessary complexity or navigation depth.

c) User decision point:

The user selects the desired service based on their immediate need:

- If interested in understanding their heart health status, they proceed with CAD Risk Prediction.
- If seeking clarity about medications, prescriptions, or drug interactions, they choose the AI Chemist module.

d) Interaction with selected module:

If the user chooses "Heart Disease Risk Assessment", a form is displayed prompting the user to input clinical and demographic data (age, sex, chest pain type, resting blood pressure, cholesterol levels, fasting blood sugar, ECG results, maximum heart rate, etc). Once all the fields are filled, the user submits the data for evaluation.

If the user chooses "AI Chemist", the interface allows the user to type a textual query about a drug or chemical compound and upload a photo of a prescription, drug label, or handwritten note for image-based analysis. The system forwards this input to the Gemini AI engine for multimodal interpretation.

e) Backend processing:

Depending on the selected path:

- The CAD assessment module preprocesses the input, feeds it to the trained ensemble model, and computes a risk score.
- The AI Chemist module sends the input to the Google Gemini 1.5 Pro Vision API, which interprets both image and text to generate chemical insights or prescription explanations.

f) Output delivery:

For the CAD module, the system returns:

- A detailed risk assessment (e.g., "You are at moderate risk of CAD").
- Personalized recommendations, such as dietary advice, physical activity guidelines, or alerts about medical follow-up.
- A link to the AI chatbot, in case the user wishes to ask follow-up questions about symptoms, risk factors, or preventive strategies.

For the AI chemist, the system returns:

- A summary of the drug or compound in question.
- Detailed usage instructions, side effects, and interaction warnings.
- If an image was uploaded, it displays what was detected and how it was interpreted.

g) Optional conversational support:

Users can engage with the integrated AI chatbot, powered by GPT, to seek clarifications or additional guidance. This step is particularly useful for first-time users or those unsure about next steps.

h) Session completion:

The user may:

- Save or download the risk report and recommendations.
- Return to the home screen to use another module.
- Exit the application entirely.

Optionally, usage logs or non-personalized interactions may be stored (depending on future versions) for improving the model or tracking system performance.

This structured flow ensures a smooth and meaningful user experience while maintaining the underlying complexity of AI-powered analytics abstracted from the user. Every design choice in this flow supports accessibility, medical relevance, and user empowerment.

4.3 AI Chemist Assistant for Medication Guidance

In modern healthcare systems, the complexity of pharmaceutical information from dosage schedules and side effects to drug interactions and contraindications often creates confusion for patients, particularly those managing chronic conditions such as coronary artery disease (CAD). Misunderstandings around prescribed medications are a leading cause of poor adherence, adverse drug events, and diminished treatment outcomes. In response to this challenge, the proposed system integrates an AI Chemist Assistant, an intelligent module designed to bridge the gap between prescription and understanding, using the power of generative AI.

4.3.1 Purpose and motivation

While the heart disease prediction model provides a proactive method for identifying CAD risk, effective disease management depends heavily on correct and informed use of medication. After diagnosis, many patients receive prescriptions that they do not fully understand. These may include:

- Complex drug combinations
- Potential interactions with over-the-counter drugs
- Dietary restrictions
- Time-based or condition-specific dosage requirements

The AI Chemist Assistant is developed to serve as an intelligent, always-available pharmaceutical advisor. It enhances the platform by allowing users to:

- Ask medication-related questions in natural language
- Upload images of prescriptions, medicine labels, or chemical compound sheets
- Receive precise, real-time guidance on drug usage, interactions, and administration

This functionality makes the system not just diagnostic, but educational and advisory, promoting informed patient decision-making.

4.3.2 Technical architecture and capabilities

The AI Chemist Assistant is powered by Google Gemini 1.5 Pro, a state-of-the-art multimodal generative AI model. It is capable of interpreting both textual and visual inputs, making it suitable for analysing:

- Typed or spoken queries (e.g., “Can I take aspirin with ramipril?”)
- Photographs of prescriptions, handwritten doctor notes, or pill bottles

Multimodal input handling:

- Text: When users input a typed query, the text is directly processed by Gemini to generate a structured, clinically grounded response.
- Image**:** If an image is uploaded (e.g., a prescription), Gemini’s Vision model interprets the content using optical character recognition (OCR) and semantic analysis, extracting medication names, dosage instructions, and additional notes.

Response generation:

- The AI provides natural language outputs explaining what the medication is for, how and when to take it, possible side effects and storage and safety instructions.
- For drug combinations, it can assess whether interactions are likely to occur and whether physician consultation is advised.

4.3.3 Integration into the platform

The AI Chemist module is integrated into the same Streamlit-based interface as the heart disease prediction system. Users can switch between functionalities with a single click on the dashboard. Behind the scenes:

- Text or image inputs are captured via the Streamlit UI
- The data is sent to a backend function that prepares the prompt or processes the image and sends the request to the Google Gemini API
- The Gemini API processes the input and returns a response
- The system displays the result in a user-friendly, clearly formatted layout

This modular integration ensures seamless user experience, without needing to switch platforms or deal with technical complexity.

4.4 Model Deployment Strategy

Deploying machine learning models into real-world, interactive applications requires careful attention to performance, accessibility, modularity, and user experience. While building and training the model is critical, the actual impact of the system depends on how seamlessly it can be accessed and used by end-users — particularly patients and healthcare workers with limited technical expertise.

In this project, the deployment strategy was designed to prioritize simplicity, responsiveness, and scalability, with the goal of providing real-time CAD risk assessment and AI-driven medical assistance through an interactive, web-based application.

4.4.1 Deployment objectives

- Efficiency: Ensure fast, real-time predictions without reloading or retraining the model at runtime.
- Interactivity: Provide a responsive user interface that non-technical users can easily navigate.
- Modularity: Allow independent integration of multiple components (e.g., risk prediction, AI Chemist, chatbot).
- Scalability: Support future expansion (e.g., mobile interface, user data tracking, wearable integration).
- Reusability: Enable reusability of the model across different application environments (local and cloud-based).

4.4.2 Tools and technologies used

The following tools and libraries were selected to meet the above objectives:

- Python 3.8+**:** The primary programming language used for all backend logic.
- Scikit-learn**:** Used to build and train the machine learning models (logistic regression, random forest, SVM, and ensemble voting).
- Joblib**:** Employed to serialize (save) the trained ensemble model as a .pkl file. This enables fast loading during deployment without the need to retrain.
- Streamlit**:** An open-source Python library used to develop the interactive web application. It allows rapid deployment of machine learning models with built-in support for forms, file uploads, and real-time display.
- Google Gemini API**:** Integrated via API calls to handle text and image inputs for the AI Chemist module.
- OpenAI GPT API (Optional)**:** Used to power the conversational health assistant chatbot for general CAD-related queries.
- Matplotlib & Seaborn: Used for plotting evaluation metrics (confusion matrix, ROC curve, PR curve) within the interface.

4.4.3 Deployment pipeline

- Model training: performed using the Cleveland dataset, and the trained ensemble model is exported as ensemble_model.pkl using Joblib.
- Web application initialization: Streamlit launches the app through a Python script (e.g., main.py). The model is loaded into memory only once at startup using joblib.load().
- User interaction: Users input clinical data (age, cholesterol, chest pain, etc.) via Streamlit's form-based interface. Alternatively, users upload prescription images or type medication queries for the AI Chemist module.
- Prediction and preprocessing: Inputs are preprocessed in real time (e.g., scaling, encoding). Data is passed to the ensemble model to generate predictions. AI Chemist queries are routed to the Gemini API; results are displayed instantly.
- Output presentation: Predictions are shown with confidence scores. Lifestyle and medication guidance is displayed based on system responses. Visualizations (e.g., risk probability, confusion matrix) are generated dynamically using Matplotlib.
- Optional logging and feedback (for future versions): User queries and predictions can be optionally logged for system improvement, pending user consent and privacy controls.

4.4.4 System modularity and extensibility

One of the key design principles of this deployment is modularity. Each major function including CAD prediction, AI Chemist, chatbot, is encapsulated as a separate component allowing:

- Independent testing and debugging of modules
- Easy replacement or upgrading of individual parts (e.g., replacing the ML model, swapping Gemini with another LLM)
- Rapid integration of new features (e.g., wearable data, mobile interface, cloud database)

In future versions, the platform could support:

- User accounts and login sessions
- Persistent data storage (using Firebase or SQL)
- Real-time wearable device integration (e.g., Fitbit, Apple Health)
- Cloud deployment using platforms like Heroku, AWS, or Streamlit Cloud

CHAPTER 5

RESULTS AND DISCUSSION

5.1 Summary of Results

These results demonstrate that the model is highly capable of distinguishing between CAD-positive and CAD-negative cases, with exceptionally high recall and ROC-AUC scores, which are especially critical in medical diagnostics.

In addition to raw performance metrics, the following system outputs were visualized to better understand classification behavior:

- Confusion Matrix: Showed a low number of false negatives and false positives.
- Precision-Recall Curve: Maintained high precision across a wide range of recall thresholds.
- ROC Curve: Approached the ideal top-left boundary, indicating near-perfect separability of classes.

On the usability front, the web-based interface, built using Streamlit, successfully delivered:

- Real-time, accessible CAD risk prediction from user-provided parameters
- Personalized health recommendations (e.g., lifestyle changes, exercise, diet)
- Immediate pharmaceutical assistance via the AI Chemist module, with both text and image input support
- An integrated health chatbot, powered by OpenAI, for answering general cardiovascular queries

The platform achieved its dual purpose: accurate early diagnosis and user-friendly post-diagnostic guidance.

5.2 Interpretation and Implications

The high recall score of 96.97% is especially significant. In clinical practice, the cost of missing a patient with CAD is high (potentially fatal). A high recall means that the system catches almost all individuals at risk, making it extremely reliable as a screening tool. Even with slightly lower

precision (86.49%), the system errs on the side of caution by flagging more people for further examination rather than missing cases, which is consistent with good medical ethics.
The F1-score of 91.43% confirms that the model strikes a strong balance between precision and recall, which is important in medical settings where both false positives and false negatives carry consequences. Moreover, the ROC-AUC score of 0.9893 indicates that the classifier is almost perfectly capable of distinguishing CAD from non-CAD cases, an outcome rarely achieved in real-world medical datasets.

The addition of the AI Chemist Assistant proved to be a meaningful innovation. Patients are no longer left to interpret cryptic prescriptions on their own. Whether through a question like "Can I take Ibuprofen with my blood pressure medication?" or a prescription photo upload, users receive clinically appropriate, AI-generated responses. This extends the system's role from just "predicting disease" to "supporting treatment adherence and literacy."
In essence, the system does not function in isolation but acts as an integrated preventive care tool, bridging gaps between diagnosis, patient education, and medication management, all of which are central to improving health outcomes and reducing mortality.

5.3 Real-World Applicability

This solution demonstrates strong potential for deployment in both clinical and telemedicine settings, particularly for:

- Primary health centers and rural clinics lacking cardiologists or advanced imaging facilities
- Telehealth platforms, where patients input symptoms and parameters remotely
- Mobile health applications that can expand this model to wearable device integration

Its low cost, non-invasive input requirements, and high interpretability make it suitable for mass health screenings, public health surveys, or community health worker support tools.

Furthermore, the AI Chemist Assistant can reduce the burden on healthcare providers by handling basic pharmaceutical questions, allowing doctors to focus on more complex cases.

5.4 Limitations and Future Improvements

While the system performs well, there are certain limitations to consider:

- Dataset Size and Diversity: The Cleveland dataset, while popular, is limited in size and demographic diversity. Future models should be trained on larger, multi-center datasets with broader geographic and ethnic representation.
- Static Inputs: The current system accepts only user-entered clinical parameters. In future versions, integration with real-time health sensors and wearable data could significantly enhance prediction accuracy.
- Limited Medical Validation: While technically robust, the system has not yet undergone clinical trials or validation by certified medical professionals. Regulatory approval and hospital collaboration would be necessary for deployment in clinical practice.
- Multilingual and Accessibility Limitations: The interface currently operates in English and assumes a certain level of health literacy. Expanding support for multiple languages, voice input, and assistive technology (for the visually impaired, for example) could broaden its usability.

5.5 Overall Impact

In summary, the system delivers on its promise to be a reliable, accessible, and intelligent CAD risk prediction and guidance platform. By combining ensemble machine learning with real-time AI assistance, it transcends the role of a traditional diagnostic tool. It aligns closely with global health objectives such as reducing premature mortality, improving medication literacy, and expanding access to preventive care. With further data, validation, and feature development, this solution has the potential to make a significant impact in both digital health and clinical practice.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

This project set out to address a critical challenge in healthcare: the need for accessible, accurate, and user-friendly tools for the early prediction and management of coronary artery disease (CAD). By leveraging the power of machine learning, natural language processing, and multimodal generative AI, the proposed system successfully integrates predictive diagnostics with post-diagnostic medication support and patient education.

The ensemble machine learning model combining logistic regression, random forest, and support vector machine classifiers demonstrated exceptional predictive performance, with high recall, F1-score, and ROC-AUC metrics on the Cleveland Heart Disease dataset. These results validate the model's capability to function as an early screening tool, capable of identifying at-risk individuals before the onset of critical symptoms.

Complementing the predictive model, the AI Chemist Assistant offers real-time support for pharmaceutical literacy through text and image-based input interpretation, powered by Google Gemini. It addresses a major gap in patient care by enabling users to understand their prescriptions, identify potential drug interactions, and adhere to correct dosage and administration guidelines.

Overall, the system represents a significant step toward a more proactive, technology-driven approach to cardiovascular health. It empowers users with not only diagnostic information but also actionable guidance, making it a valuable tool for both individual patients and healthcare practitioners particularly in underserved or remote areas where specialized care may be limited.

6.2 Future Work

While the system fulfills its current objectives, there are several areas where enhancements could significantly expand its utility, performance, and real-world impact. Future work on this project could include:

a) Clinical validation and deployment

To transition the system from academic prototype to clinical tool, collaboration with healthcare providers, cardiologists, and pharmaceutical professionals is essential. Conducting clinical trials and real-world validations will establish the system's medical reliability and regulatory readiness.

b) Integration with real-time health devices

The predictive model currently relies on static, user-input clinical parameters. Future versions could integrate with wearable devices (e.g., smartwatches, ECG monitors, blood pressure cuffs) to gather real-time, continuous health data, enabling dynamic risk assessment and more accurate predictions.

c) Enhanced AI chemist capabilities

The AI Chemist could be expanded to include:

- Voice-based interaction for elderly or visually impaired users
- Multilingual support for broader global accessibility
- Access to verified medical databases for more authoritative drug interaction warnings

d) Cloud-based scaling and user management

To make the platform more widely accessible, deploying it on cloud infrastructure would allow for:

- Global access via mobile or desktop browsers
- Support for simultaneous users
- Personalized user accounts with health dashboards, history tracking, and reminders

e) Data expansion and model refinement

Training the model on larger, more diverse datasets will improve its generalizability across populations. Advanced techniques like deep learning, feature selection optimization, or explainable AI (XAI) could further improve both performance and transparency.

f) Compliance with medical standards

Future versions should aim to comply with health data privacy laws (like HIPAA or GDPR), medical software certifications, and ethical AI guidelines to ensure patient trust and legal acceptance.

6.3 Final Remarks

As cardiovascular disease continues to be a leading global cause of death, solutions that combine early detection with accessible education are urgently needed. This project contributes meaningfully to that mission. By bridging AI-powered diagnostics with user-oriented support systems, it offers a scalable and impactful approach to reducing CAD-related mortality. With continued development, validation, and support, this system could evolve into a robust digital health companion, capable of transforming preventive care and empowering patients worldwide.

REFERENCES

[1] Seckeler, M. D., & Hoke, T. R. (2011). The worldwide epidemiology of acute rheumatic fever and rheumatic heart disease. Clinical epidemiology, 67-84.

[2] Gaziano, T. A., Bitton, A., Anand, S., Abrahams-Gessel, S., & Murphy, A. (2010). Growing epidemic of coronary heart disease in low-and middle-income countries. *Current problems in cardiology*, *35*(2), 72-115.

[3] Boukhatem, C., Youssef, H. Y., & Nassif, A. B. (2022, February). Heart disease prediction using machine learning. In *2022 Advances in Science and Engineering Technology International Conferences (ASET)* (pp. 1-6). IEEE.

[4] Jindal, H., Agrawal, S., Khera, R., Jain, R., & Nagrath, P. (2021). Heart disease prediction using machine learning algorithms. In *IOP conference series: materials science and engineering* (Vol. 1022, No. 1, p. 012072). IOP Publishing.

[5] Ramalingam, V. V., Dandapath, A., & Raja, M. K. (2018). Heart disease prediction using machine learning techniques: a survey. *International Journal of Engineering & Technology*, *7*(2.8), 684-687.

[6] Weng, S. F., Reps, J., Kai, J., Garibaldi, J. M., & Qureshi, N. (2017). Can machine-learning improve cardiovascular risk prediction using routine clinical data?. *PloS one*, *12*(4), e0174944.

[7] Fatima, M., & Pasha, M. (2017). Survey of machine learning algorithms for disease diagnostic. *Journal of Intelligent Learning Systems and Applications*, *9*(01), 1-16.

[8] Vembandasamy, K., Sasipriya, R., & Deepa, E. (2015). Heart diseases detection using Naive Bayes algorithm. *International Journal of Innovative Science, Engineering & Technology*, *2*(9), 441-444.

[9] Chaurasia, D. V., & Pal, S. (2014). Data mining approach to detect heart diseases. International Journal of Advanced Computer Science and Information Technology (IJACSIT) Vol, 2, 56-66.

[10] Bhatt, C. M., Patel, P., Ghetia, T., & Mazzeo, P. L. (2023). Effective heart disease prediction using machine learning techniques. *Algorithms*, *16*(2), 88.

[11] Patel, J., TejalUpadhyay, D., & Patel, S. (2015). Heart disease prediction using machine learning and data mining technique. *Heart Disease*, *7*(1), 129-137.

[12] Rindhe, B. U., Ahire, N., Patil, R., Gagare, S., & Darade, M. (2021). Heart disease prediction using machine learning. *Heart Disease*, *5*(1).

[13] A. Janosi, W. Steinbrunn, M. Pfisterer, and R. Detrano. "Heart Disease," UCI Machine Learning Repository, 1989. [Online]. Available: https://doi.org/10.24432/C52P4X.

[14] Tithi, S. R., Aktar, A., Aleem, F., & Chakrabarty, A. (2019, June). ECG data analysis and heart disease prediction using machine learning algorithms. In *2019 IEEE Region 10 Symposium (TENSYMP)* (pp. 819-824). IEEE.

[15] Garg, A., Sharma, B., & Khan, R. (2021). Heart disease prediction using machine learning techniques. In *IOP Conference series: materials science and engineering* (Vol. 1022, No. 1, p. 012046). IOP Publishing.

[16] Mohan, Senthilkumar & Thirumalai, Chandra Segar & Srivastava, Gautam. (2019). Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques. IEEE Access. PP. 1-1. 10.1109/ACCESS.2019.2923707.

[17] Ramalingam, V V & Dandapath, Ayantan & Raja, M. (2018). Heart disease prediction using machine learning techniques: A survey. International Journal of Engineering & Technology. 7. 684. 10.14419/ijet.v7i2.8.10557.

[18] Patel, Jaymin & Tejalupadhyay, Samir & Patel, Samir. (2016). Heart Disease prediction using Machine learning and Data Mining Technique. 10.090592/IJCSC.2016.018.

[19] Subbalakshmi, G. & Ramesh, K. & Rao, M.. (2011). Decision Support in Heart Disease Prediction System using Naive Bayes. Ind. J. Comput. Sci. Eng. (IJCSE). 2. 170-176.

[20] Jegan, Chitra & Seenivasagam, V. (2013). Review of heart disease prediction system using data mining and hybrid intelligent techniques. International Journal of Soft Computing. 3. 2229-6956. 10.21917/ijsc.2013.0087.

APPENDIX A

CODING

```
import streamlit as st
import joblib
import pandas as pd
import openai
import os
import google.generativeai as genai
from sklearn.ensemble import VotingClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import cross_val_score
from sklearn.preprocessing import StandardScaler
from PIL import Image
from fpdf import FPDF
import datetime
import io
import numpy as np

# Load API keys securely
openai.api_key = os.getenv("OPENAI_API_KEY")
GOOGLE_API_KEY = os.getenv("GOOGLE_API_KEY")
genai.configure(api_key=GOOGLE_API_KEY)

# Load and preprocess dataset
if not os.path.exists("ensemble_model.pkl"):
    data = pd.read_csv("heart.csv")
    X = data.drop(columns=['target'])
    y = data['target']

    # Standardize features
    scaler = StandardScaler()
    X_scaled = scaler.fit_transform(X)
    joblib.dump(scaler, "scaler.pkl")

    # Define models with balanced class weights
    model1 = LogisticRegression(class_weight='balanced', max_iter=1000)
    model2 = RandomForestClassifier(class_weight='balanced', n_estimators=100)
    model3 = SVC(probability=True, class_weight='balanced')

    # Create Voting Classifier
    voting_clf = VotingClassifier(estimators=[
        ('lr', model1),
        ('rf', model2),
```

```
        ('svm', model3)
    ], voting='soft')

    # Cross-validation
    scores = cross_val_score(voting_clf, X_scaled, y, cv=5)
    print(f"Cross-validation accuracy: {scores.mean():.2f} (+/- {scores.std():.2f})")

    # Train and save ensemble model
    voting_clf.fit(X_scaled, y)
    joblib.dump(voting_clf, "ensemble_model.pkl")
else:
    data = pd.read_csv("heart.csv")
    X = data.drop(columns=['target'])
    scaler = joblib.load("scaler.pkl")

# Load the trained ensemble model
model = joblib.load("ensemble_model.pkl")

def clean_text(text):
    """Convert special characters to ASCII equivalents, handling numeric inputs"""
    if text is None:
        return ""
    if not isinstance(text, str):
        text = str(text)

    replacements = {
        '≤': '<=',
        '≥': '>=',
        '–': '-',
        '—': '-',
        '‘': "'",
        '’': "'",
        '“': '"',
        '”': '"',
        '…': '...'
    }
    for k, v in replacements.items():
        text = text.replace(k, v)
    return text

def explain_prediction(prediction, input_data):
    explanation = []
    input_data = input_data.iloc[0]  # Convert DataFrame row to Series

    if prediction == 1:
        explanation.append("🚨 **High Risk Factors Detected:**")
        if input_data['age'] > 50:
            explanation.append(f"- Age ({input_data['age']}) increases risk")
        if input_data['chol'] > 240:
```

```
            explanation.append(f"- High cholesterol ({input_data['chol']} mg/dL)")
        if input_data['trestbps'] > 140:
            explanation.append(f"- Elevated blood pressure ({input_data['trestbps']} mmHg)")
        if input_data['exang'] == 1:
            explanation.append("- Exercise-induced angina is concerning")
        if input_data['oldpeak'] > 2:
            explanation.append(f"- Significant ST depression ({input_data['oldpeak']} mm)")
        if input_data['ca'] > 0:
            explanation.append(f"- {4- input_data['ca']} major vessels showing blockage")
    else:
        explanation.append("✅ **Protective Factors:**")
        if input_data['age'] <= 50:
            explanation.append(f"- Younger age ({input_data['age']}) reduces risk")
        if input_data['chol'] <= 200:
            explanation.append(f"- Healthy cholesterol level ({input_data['chol']} mg/dL)")
        if input_data['trestbps'] <= 120:
            explanation.append(f"- Normal blood pressure ({input_data['trestbps']} mmHg)")

    return "\n".join(explanation)

def generate_lifestyle_recommendations(input_data, prediction):
    recommendations = []

    # Enhanced General Tips
    recommendations.append("- 🏃 **Exercise Regularly**: 30-45 minutes of brisk walking, 5
days/week")
    recommendations.append("- 🥗 **Balanced Diet**: Focus on whole grains, lean proteins, and
colorful vegetables")
    recommendations.append("- 🫒 **Limit Oil Intake**: Use <3 tsp oil per day, prefer
olive/canola oil")
    recommendations.append("- 🧂 **Reduce Salt**: Aim for <5g (1 tsp) salt daily including
hidden salts")
    recommendations.append("- 💧 **Hydration**: Drink 2-3 liters of fluids daily (water,
herbal teas)")
    recommendations.append("- 🌾 **Increase Fiber**: Gradually add whole grains, fruits with
skin, legumes")
    recommendations.append("- 🚭 **Tobacco Free**: Complete avoidance of smoking/chewing
tobacco")
    recommendations.append("- 🍷 **Alcohol Moderation**: ≤1 drink/day for women, ≤2 for men
(1 drink = 14g alcohol)")
    recommendations.append("- 😴 **Sleep Hygiene**: 7-9 hours quality sleep, maintain
consistent schedule")
    recommendations.append("- 🧘 **Stress Management**: Daily 10-15 min meditation/deep
breathing")

    # Personalized recommendations
    if input_data['age'].values[0] > 50:
        recommendations.append("\n### For Age 50+:")
```

```
        recommendations.append("- 📅 **Annual Checkups**: Full lipid profile + cardiac
evaluation")
        recommendations.append("- 💊 **Aspirin Consideration**: Discuss low-dose aspirin with
your doctor")
        recommendations.append("- 🏋️ **Strength Training**: Add 2 days/week resistance
exercises")

    if input_data['chol'].values[0] > 240:
        recommendations.append("\n### Cholesterol Management:")
        recommendations.append("- 🥑 **Healthy Fats**: Increase nuts, seeds, fatty fish
(salmon)")
        recommendations.append("- 🚫 **Avoid Trans Fats**: Check labels for 'partially
hydrogenated oils'")
        recommendations.append("- 🌾 **Soluble Fiber**: 10-25g/day from oats, psyllium,
apples")

    if prediction == 1:
        recommendations.append("\n### Medical Priority Actions:")
        recommendations.append("- 🩺 **Doctor Consultation**: Within 2 weeks for cardiac
evaluation")
        recommendations.append("- 💊 **Medication Review**: Statins/antihypertensives if
indicated")
        recommendations.append("- 📊 **Monitoring Plan**: Weekly BP + monthly lipid checks
initially")
        recommendations.append("- 🚨 **Emergency Signs**: Know symptoms requiring immediate
care:")
        recommendations.append("  - Chest pressure/pain lasting >15 minutes")
        recommendations.append("  - Sudden shortness of breath with sweating")

    return recommendations

def predict_heart_disease(input_data):
    input_scaled = scaler.transform(input_data)
    return model.predict(input_scaled)[0]  # Only return prediction (0 or 1)

def preprocess_input(age, gender, cp, trestbps, chol, fbs, restecg, thalach, exang, oldpeak,
slope, ca, thal):
    gender = 1 if gender == "Male" else 0
    fbs = 1 if fbs == "True" else 0
    exang = 1 if exang == "Yes" else 0

    cp_dict = {"Typical Angina": 0, "Atypical Angina": 1, "Non-anginal Pain": 2,
"Asymptomatic": 3}
    cp = cp_dict[cp]

    restecg_dict = {"Normal": 0, "ST-T wave abnormality": 1, "Left ventricular hypertrophy":
2}
    restecg = restecg_dict[restecg]
```

```
    slope_dict = {"Upsloping": 0, "Flat": 1, "Downsloping": 2}
    slope = slope_dict[slope]

    thal_dict = {"Normal": 0, "Fixed Defect": 1, "Reversible Defect": 2}
    thal = thal_dict[thal]

    return pd.DataFrame({
        'age': [age], 'gender': [gender], 'cp': [cp], 'trestbps': [trestbps],
        'chol': [chol], 'fbs': [fbs], 'restecg': [restecg], 'thalach': [thalach],
        'exang': [exang], 'oldpeak': [oldpeak], 'slope': [slope], 'ca': [ca],
        'thal': [thal]
    })

def generate_pdf(input_data, prediction, recommendations):
    pdf = FPDF()
    pdf.add_page()

    # Set font - using Arial as it's widely available
    pdf.set_font("Arial", size=12)

    # Title
    pdf.set_font("Arial", 'B', 16)
    pdf.cell(200, 10, txt="Heart Disease Risk Report", ln=True, align='C')
    pdf.ln(10)

    # Date
    pdf.set_font("Arial", size=10)
    pdf.cell(0, 10, txt=f"Generated on: {datetime.datetime.now().strftime('%Y-%m-%d
%H:%M:%S')}", ln=True)
    pdf.ln(10)

    # Patient Information
    pdf.set_font("Arial", 'B', 12)
    pdf.cell(0, 10, txt="Patient Information:", ln=True)
    pdf.set_font("Arial", size=10)

    readable_data = {
        'Age': str(input_data['age'][0]),
        'Gender': 'Male' if input_data['gender'][0] == 1 else 'Female',
        'Resting Blood Pressure': f"{input_data['trestbps'][0]} mmHg",
        'Cholesterol': f"{input_data['chol'][0]} mg/dL",
        'Fasting Blood Sugar': '>120 mg/dL' if input_data['fbs'][0] == 1 else '<=120 mg/dL',
        'Max Heart Rate': str(input_data['thalach'][0]),
        'ST Depression': str(input_data['oldpeak'][0])
    }

    for key, value in readable_data.items():
        pdf.cell(0, 10, txt=f"{key}: {value}", ln=True)
```

```
    pdf.ln(10)

    # Risk Assessment
    pdf.set_font("Arial", 'B', 12)
    pdf.cell(0, 10, txt="Risk Assessment:", ln=True)
    pdf.set_font("Arial", size=10)

    risk_text = "HIGH RISK of heart disease" if prediction == 1 else "LOW RISK of heart
disease"
    pdf.cell(0, 10, txt=f"Assessment: {risk_text}", ln=True)
    pdf.ln(10)

    # Recommendations
    pdf.set_font("Arial", 'B', 12)
    pdf.cell(0, 10, txt="Lifestyle Recommendations:", ln=True)
    pdf.set_font("Arial", size=10)

    for rec in recommendations:
        clean_text = rec.replace("###", "").replace("**", "")
        clean_text = clean_text.replace("🫒", "[Oil]").replace("🧂", "[Salt]")
        clean_text = clean_text.replace("💧", "[Water]").replace("🌾", "[Fiber]")
        clean_text = clean_text.replace("🏃", "[Exercise]").replace("🥗", "[Diet]")
        clean_text = clean_text.replace("🚭", "[Smoking]").replace("🍷", "[Alcohol]")
        clean_text = clean_text.replace("😴", "[Sleep]").replace("🧘", "[Stress]")
        clean_text = clean_text.replace("📅", "[Checkups]").replace("💊", "[Medication]")
        clean_text = clean_text.replace("🏋", "[Strength]").replace("🥑", "[Healthy Fats]")
        clean_text = clean_text.replace("🚫", "[Avoid]").replace("🩺", "[Doctor]")
        clean_text = clean_text.replace("📊", "[Monitoring]").replace("🚨", "[Emergency]")
        clean_text = ''.join(char for char in clean_text if ord(char) < 128)
        pdf.multi_cell(0, 10, txt=clean_text)

    return pdf.output(dest='S').encode('latin-1')

def main():
    st.title('🫀 AI-Powered Coronary Artery Disease (CAD) Risk Assessment')

    st.markdown("""
    <style>
        .description-box {
            background-color: #092a66;
            border-radius: 10px;
            padding: 15px;
            margin-bottom: 20px;
        }
        .risk-high {
            color: #ff4b4b;
            font-weight: bold;
        }
        .risk-low {
```

```
            color: #006400;
            font-weight: bold;
        }
        .section-title {
            color: #2c3e50;
            border-bottom: 2px solid #2c3e50;
            padding-bottom: 5px;
        }
    </style>
    """, unsafe_allow_html=True)

    option = st.sidebar.radio("Choose an Application:", ["Heart Disease Risk Assessment",
"AI Chemist"])

    if option == "Heart Disease Risk Assessment":
        st.header("🫀 Heart Disease Risk Assessment")
        st.markdown("""
        <div class="description-box">
        This tool assesses your risk of coronary artery disease based on key health
indicators.
        Please provide accurate information for the most reliable assessment.
        </div>
        """, unsafe_allow_html=True)

        with st.expander("🔍 About This Assessment"):
            st.write("""
            This risk assessment uses machine learning to analyze multiple factors that
contribute to heart disease risk.
            The model combines three different algorithms (Logistic Regression, Random
Forest, and SVM) for more accurate predictions.
                    """)

        col1, col2 = st.columns(2)

        with col1:
            age = st.slider('Age', 29, 77, 50)
            st.markdown("""
            <div class="description-box">
            <b>ℹ️ Age Factor:</b> Risk increases with age. Men over 45 and women over 55 are
at higher risk.
            </div>
            """, unsafe_allow_html=True)

            gender_options = {
                "Male": "Men generally develop heart disease earlier than women.",
                "Female": "Women's risk increases after menopause, and symptoms may differ."
            }
            gender = st.selectbox("Gender", list(gender_options.keys()))
            st.markdown(f"""
```

```
            <div class="description-box">
            <b>ℹ️ Gender Difference:</b> {gender_options[gender]}
            </div>
            """, unsafe_allow_html=True)

            chest_pain_desc = {
                "Typical Angina": "Predictable chest pain during exertion, relieved by rest
- classic sign of reduced blood flow to heart.",
                "Atypical Angina": "Less predictable chest discomfort that may not follow
typical patterns.",
                "Non-anginal Pain": "Chest discomfort unlikely to be heart-related (e.g.,
digestive or muscular).",
                "Asymptomatic": "No chest pain (silent ischemia can still indicate heart
disease)."
            }
            cp = st.selectbox("Chest Pain Type", list(chest_pain_desc.keys()))
            st.markdown(f"""
            <div class="description-box">
            <b>ℹ️ Chest Pain Info:</b> {chest_pain_desc[cp]}
            </div>
            """, unsafe_allow_html=True)

            trestbps = st.slider('Resting Blood Pressure (mm Hg)', 94, 200, 120)
            st.markdown(f"""
            <div class="description-box">
            <b>ℹ️ Blood Pressure Guide:</b>
            - Normal: Below 120/80 mmHg
            - Elevated: 120-129/<80 mmHg
            - High: 130+/80+ mmHg
            Your input: {trestbps} mmHg (systolic)
            </div>
            """, unsafe_allow_html=True)

            chol = st.slider('Cholesterol (mg/dl)', 126, 564, 200)
            st.markdown(f"""
            <div class="description-box">
            <b>ℹ️ Cholesterol Levels:</b>
            - Desirable: <200 mg/dL
            - Borderline high: 200-239 mg/dL
            - High: ≥240 mg/dL
            Your input: {chol} mg/dL
            </div>
            """, unsafe_allow_html=True)

            fbs_options = {
                "True": "Fasting glucose >120 mg/dL may indicate diabetes or prediabetes.",
                "False": "Normal fasting glucose (<120 mg/dL) reduces diabetes risk."
            }
            fbs = st.selectbox("Fasting Blood Sugar > 120 mg/dl", list(fbs_options.keys()))
```

```
            st.markdown(f"""
            <div class="description-box">
            <b>ℹ️ Blood Sugar Info:</b> {fbs_options[fbs]}
            </div>
            """, unsafe_allow_html=True)

        with col2:
            restecg_desc = {
                "Normal": "No ECG abnormalities detected.",
                "ST-T wave abnormality": "May indicate ischemia, electrolyte imbalance, or
other conditions.",
                "Left ventricular hypertrophy": "Thickened heart muscle, often from high
blood pressure."
            }
            restecg = st.selectbox("Resting ECG Results", list(restecg_desc.keys()))
            st.markdown(f"""
            <div class="description-box">
            <b>ℹ️ ECG Interpretation:</b> {restecg_desc[restecg]}
            </div>
            """, unsafe_allow_html=True)

            thalach = st.slider('Max Heart Rate Achieved', 71, 202, 150)
            max_hr_estimate = 220 - age
            st.markdown(f"""
            <div class="description-box">
            <b>ℹ️ Heart Rate Info:</b>
            - Your estimated max heart rate: ~{max_hr_estimate} bpm (220 - age)
            - Your input: {thalach} bpm
            - Lower values may indicate reduced cardiovascular fitness
            </div>
            """, unsafe_allow_html=True)

            exang_options = {
                "Yes": "Chest pain during exercise suggests possible coronary artery
disease.",
                "No": "No exercise-induced chest pain is a positive sign."
            }
            exang = st.selectbox("Exercise Induced Angina", list(exang_options.keys()))
            st.markdown(f"""
            <div class="description-box">
            <b>ℹ️ Exercise Angina:</b> {exang_options[exang]}
            </div>
            """, unsafe_allow_html=True)

            oldpeak = st.slider('ST Depression', 0.0, 6.2, 2.0, step=0.1)
            st.markdown(f"""
            <div class="description-box">
            <b>ℹ️ ST Depression:</b>
            - Measures ECG changes during stress test
```

```
            - Normal: 0-1 mm
            - Mild: 1-2 mm
            - Significant: >2 mm
            Your input: {oldpeak} mm
            </div>
            """, unsafe_allow_html=True)

            slope_desc = {
                "Upsloping": "Generally normal, but context matters.",
                "Flat": "May suggest reduced blood flow during stress.",
                "Downsloping": "More concerning for significant coronary disease."
            }
            slope = st.selectbox("Slope of ST Segment", list(slope_desc.keys()))
            st.markdown(f"""
            <div class="description-box">
            <b>ℹ️ ST Slope:</b> {slope_desc[slope]}
            </div>
            """, unsafe_allow_html=True)

            ca = st.slider('Major Vessels (0-4) Coloured By Flourosopy', 0, 4, 0)
            st.markdown(f"""
            <div class="description-box">
            <b>ℹ️ Vessel Blockage:</b>
            Number of major coronary arteries with no significant narrowing seen on
angiogram.
            Your input: {ca} vessels affected
            </div>
            """, unsafe_allow_html=True)

            thal_desc = {
                "Normal": "No blood disorder affecting oxygen transport.",
                "Fixed Defect": "Permanent heart muscle damage from prior heart attack.",
                "Reversible Defect": "Temporary blood flow issues during stress test."
            }
            thal = st.selectbox("Thalassemia", list(thal_desc.keys()))
            st.markdown(f"""
            <div class="description-box">
            <b>ℹ️ Thalassemia Info:</b> {thal_desc[thal]}
            </div>
            """, unsafe_allow_html=True)

        if st.button('🔍 Assess My Risk', type="primary"):
            with st.spinner('Analyzing your risk factors...'):
                input_data = preprocess_input(age, gender, cp, trestbps, chol, fbs, restecg,
thalach, exang, oldpeak, slope, ca, thal)
                prediction = predict_heart_disease(input_data)

                # Display results
                st.subheader("📊 Risk Assessment Results")
```

```
                if prediction == 1:
                    st.markdown('<h3 class="risk-high">🚨 High Risk of Heart Disease</h3>',
unsafe_allow_html=True)
                else:
                    st.markdown('<h3 class="risk-low">✅ Low Risk of Heart Disease</h3>',
unsafe_allow_html=True)

                # Show explanation
                st.subheader("🔍 Risk Factors Analysis")
                st.markdown(explain_prediction(prediction, input_data))

                # Show recommendations
                st.subheader("💡 Personalized Recommendations")
                recommendations = generate_lifestyle_recommendations(input_data, prediction)
                for rec in recommendations:
                    st.markdown(rec)

                # PDF Download Section
                st.subheader("📄 Download Your Report")
                pdf_data = generate_pdf(input_data, prediction, recommendations)
                st.download_button(
                    label="📩 Download Report as PDF",
                    data=pdf_data,
                    file_name="heart_disease_risk_report.pdf",
                    mime="application/pdf"
                )


        # AI Chatbot section
        st.markdown("---")
        st.subheader("💬 Heart Health Assistant")
        st.write("Ask any questions about heart disease prevention, symptoms, or
management:")

        user_query = st.text_input("Your question:", placeholder="e.g., What are early signs
of heart disease?")
        if st.button("Ask the AI Assistant") and user_query:
            with st.spinner('Getting expert information...'):
                try:
                    client = openai.OpenAI()
                    response = client.chat.completions.create(
                        model="gpt-3.5-turbo",
                        messages=[
                            {"role": "system", "content": "You are a heart health assistant.
Provide clear, medically accurate information in simple terms."},
                            {"role": "user", "content": user_query}
                        ]
                    )
                    ai_response = response.choices[0].message.content.strip()
```

```
                    st.markdown(f"""
                    <div style="background-color: #092a66; padding: 15px; border-radius:
10px; margin-top: 10px;">
                        <b>AI Response:</b> {ai_response}
                    </div>
                    """, unsafe_allow_html=True)
                except Exception as e:
                    st.error(f"Error accessing AI assistant: {str(e)}")

    elif option == "AI Chemist":
        st.header("🧪 AI Medication Assistant")
        st.markdown("""
        <div class="description-box">
        Hi there, I am here to help users with medication matters, such as drug
interactions, administration instructions,.
        You can upload images of medicines and ask me questions related to it.
        </div>
        """, unsafe_allow_html=True)

        chem_input = st.text_area("Describe your medication query:",
                                  placeholder="e.g., 'What is Dolo-650 used for?'")

        uploaded_image = st.file_uploader("Upload a chemistry-related image (optional)",
                                          type=["jpg", "png", "jpeg"])

        if st.button("Analyze with AI", type="primary"):
            if chem_input.strip() or uploaded_image:
                with st.spinner('Analyzing your chemistry question...'):
                    try:
                        image = Image.open(uploaded_image) if uploaded_image else None
                        model = genai.GenerativeModel("models/gemini-1.5-pro-latest")

                        if image:
                            response = model.generate_content([chem_input, image])
                        else:
                            response = model.generate_content(chem_input)

                        st.markdown("### AI Chemist Analysis")
                        if uploaded_image:
                            st.image(image, caption="Uploaded Image",
use_container_width=True)
                        st.markdown(f"""
                        <div style="background-color: #092a66; padding: 15px; border-radius:
10px; margin-top: 10px;">
                            {response.text}
                        </div>
                        """, unsafe_allow_html=True)
                    except Exception as e:
                        st.error(f"An error occurred: {str(e)}")
```

```
            else:
                st.warning("Please enter a question or upload an image to analyze.")

if __name__ == '__main__':
    main()
```

APPENDIX B

CONFERENCE PUBLICATION

Acceptance Notification - IEEE ICCTDC 2025 External Inbox ×

Microsoft CMT <noreply@msr-cmt.org> Fri, Apr 25, 1:05 PM (4 days ago)
to me

Dear Khayati Sharma

Paper ID / Submission ID : 1534

Title : AI-Powered Early Diagnosis and Personalized Health Recommendations for Coronary Artery Disease (CAD) using Predictive Analytics

Greeting from IEEE ICCTDC 2025

We are pleased to inform you that your paper has been accepted for the Presentation as a full paper for the- "2025 International Conference on Computing Technologies &Data Communication (ICCTDC), Hassan , Karnataka, India.

All accepted and presented papers will be submitted to IEEE Xplore for the further publication.

You should finish the registration before deadline, or you will be deemed to withdraw your paper:

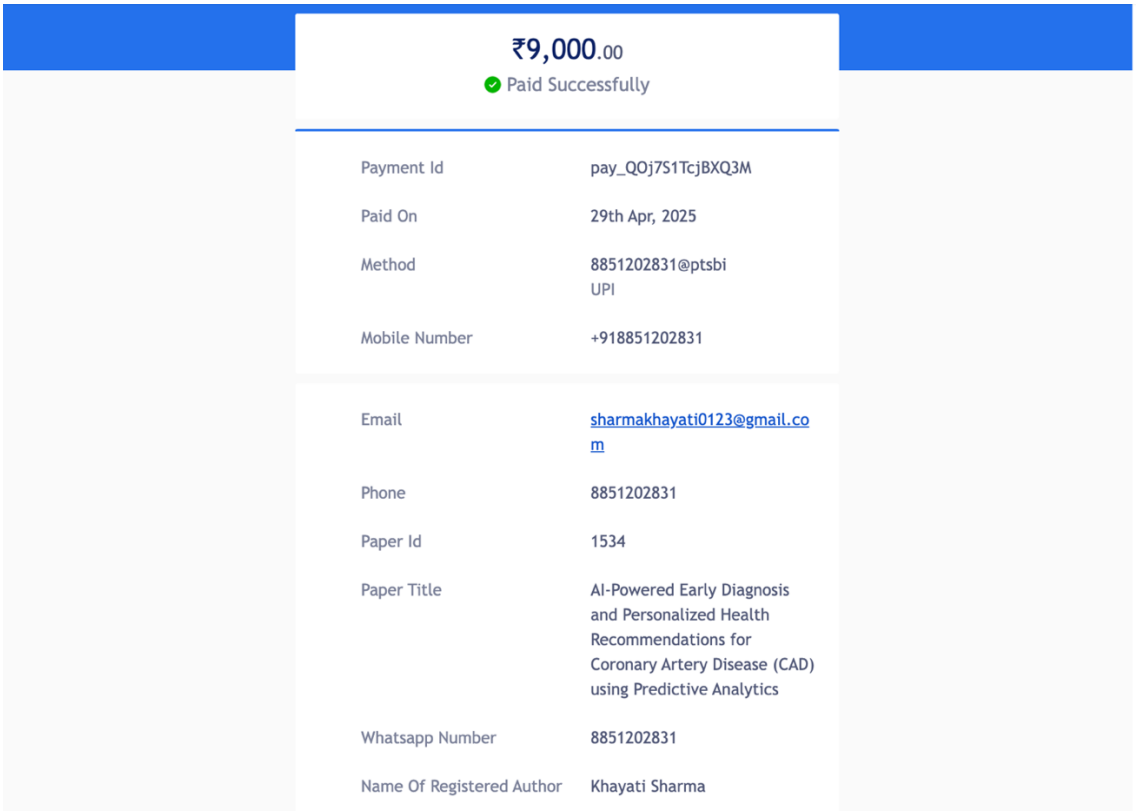
Complete the Registration Process (The last date of payment Registration is 30 APRIL 2025)

Payment Links :

For Indian Authors: https://rzp.io/rzp/icctdc

For Foreign Authors: https://rzp.io/rzp/ICCTDCForeign

Further steps like IEEE PDF xpress and E copyright will be given later once registration is over after the deadline.



APPENDIX C

JOURNAL PUBLICATION

AI-Powered Early Diagnosis and Personalized Health Recommendations for Coronary Artery Disease (CAD) using Predictive Analytics

Siddharth Vats
Department of Computing Technologies, School of Computing SRM Institute of Science and Technology
Tamil Nadu, India
siddharthvats44@gmail.com

Khayati Sharma
Department of Computing Technologies, School of Computing SRM Institute of Science and Technology
Tamil Nadu, India
sharmakhayati0123@gmail.com

M. Revathi
Department of Computing Technologies, School of Computing SRM Institute of Science and Technology
Tamil Nadu, India
revathim@srmist.edu.in

***Abstract*—Cardiovascular diseases, especially coronary artery disease, have been the leading cause of deaths worldwide. Hence, the early detection and personalized treatment strategies for coronary artery disease (CAD) are now needed more than ever. This paper suggests an AI-driven framework based on machine learning algorithms for the timely risk detection of CAD based on 13 clinical parameters. The framework utilizes an ensemble learning method that includes logistic regression, random forest and support vector machine classifiers which improve the prediction accuracy. The proposed system also comes equipped with an AI chemist assistant which helps users specifically with medication-related queries, such as drug interactions, administration instructions and prescription explanations. This feature managed textual and visual inputs using Google's generative AI (Gemini), thus allowing users to upload photos or ask questions and receive instant assistance. Adding to the suite of features, the system also comes with a health recommendation engine, which provides users with certain lifestyle changes to be adopted based on the risk assessment, and a chatbot which answers more general questions that the user might have related to his diet, exercise, sleep or any symptoms being experienced. This approach provides an integrated preventative care solution by combining CAD risk prediction, AI-based medication assistance, a coronary health chatbot and lifestyle recommendations. Apart from improving accessibility, the suggested solution also improves patient education and enables more informed decision-making in cardiovascular disease care. The experimental results show the effectiveness of the ensemble model in risk prediction of CAD and AI chemist's capacity to help users with medication-related concerns. This integrated system aims to combine AI-based diagnostic techniques with medicative and general heart-health assistance, thus improving the efficiency and accessibility of healthcare services.**

***Keywords*—coronary artery disease (CAD), ensemble learning, generative AI, coronary health chatbot, AI medication assistant**

I. Introduction

Cardiovascular diseases (CVDs), and coronary artery disease (CAD) in particular, continue to be a major cause of morbidity and mortality worldwide [1]. Early diagnosis and risk stratification are essential to avoid serious complications. However, standard diagnostic procedures often involve long clinical assessment, advanced equipment, and specialist interpretation. Advances in artificial intelligence (AI) and machine learning (ML) have revolutionized predictive analytics into a valuable tool in medical diagnosis, with enhanced accuracy, efficiency, and accessibility [2].

The system in this case is built based on research with artificial intelligence to facilitate predictive analytics for coronary artery disease risk assessment and individualized health recommendations. To improve the prediction accuracy, the system integrates a hybrid ensemble approach that combines three distinct machine learning models: logistic regression, random forest, and a support vector machine (SVM). This combination leverages the strengths of each algorithm to produce more robust and reliable results. The models leverage clinical result data and integrate basic cardiovascular risk predictors to yield patient-specific risk measurements. This research goes beyond typical predictive modeling by incorporating an AI-enhanced chemist instrument that employs LLMs to facilitate users' understanding of medicines, drug interactions, and life changes. Integrating a multimodal AI system that can analyze text-based input and visual cues enhances user engagement and usability.

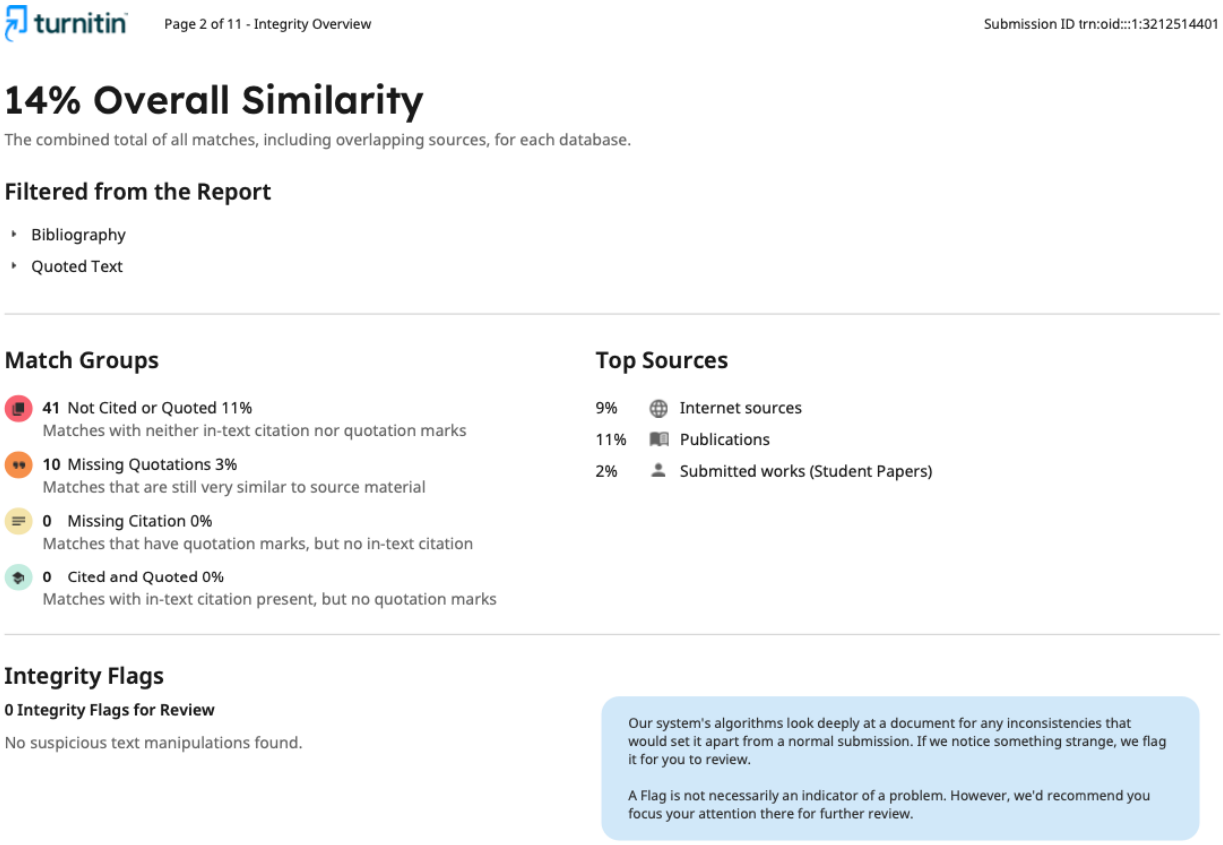
The primary objective of this research is to develop an AI-powered system capable of early diagnosis and risk assessment for Coronary Artery Disease (CAD) using predictive analytics. Additionally, the system integrates an AI Chemist Assistant to provide medication-related guidance and answer users' queries about drugs, side effects, and interactions. The proposed framework leverages machine learning algorithms for CAD prediction and natural language processing (NLP) for AI-driven patient interaction, enhancing accessibility and engagement in healthcare.

II. Related Work

Cardiovascular disease prediction has been a major focus of research with several studies using machine learning to improve diagnostic performance and detect diseases early. Many have attempted various models, datasets, and methods to enhance predictive systems. Seckeler and Hoke [3] presented a detailed overview of the epidemiology of rheumatic heart disease and its burden and long-term public health impact. Early detection remains a recurring theme in heart disease research, particularly within vulnerable and underserved populations. Several studies underscore the urgency of diagnosing cardiovascular conditions before they progress to severe stages. For instance, Gaziano et al. [4] raised concern over the growing burden of coronary artery disease (CAD) in low- and middle-income countries, drawing attention to how limited access to timely care often exacerbates patient outcomes. Their findings emphasize the need for scalable and proactive diagnostic solutions in such regions.

APPENDIX D

PLAGIARISM REPORT



turnitin Page 2 of 11 - Integrity Overview Submission ID trn:oid:::1:3212514401

14% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text

Match Groups

41 Not Cited or Quoted 11%
Matches with neither in-text citation nor quotation marks

10 Missing Quotations 3%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

9% Internet sources
11% Publications
2% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Format - I

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

(Deemed to be University u/s 3 of UGC Act, 1956)

Office of Controller of Examinations

REPORT FOR PLAGIARISM CHECK ON THE DISSERTATION/PROJECT REPORTS FOR UG/PG PROGRAMMES

(To be attached in the dissertation/ project report)

1	Name of the Candidate **(IN BLOCK LETTERS)**	KHAYATI SHARMA
2	Address of the Candidate	B-206, Sandeep Sarovar, Old Mhada Road, Andheri West, Mumbai
3	Registration Number	RA2111003010710
4	Date of Birth	01/02/2003
5	Department	C.TECH
6	Faculty	Dr. M. Revathi
7	Title of the Dissertation/Project	**AI–Powered Early Diagnosis And Personalised Health Recommendations For Coronary Artery Disease (CAD) Using Predictive Analytics**
8	Whether the above project /dissertation is done by	~~Individual~~ or group : (Strike whichever is not applicable) a) If the project/ dissertation is done in group, then how many students together completed the project : 2 b) Mention the Name & Register number of other candidates : **SIDDHARTH VATS [RA2111003010606]**
9	Name and address of the Supervisor / Guide	Dr. M. Revathi
10	Name and address of Co-Supervisor / Co- Guide (if any)	NIL **Mail ID: Mobile Number:**

11	Software Used	Turnitin
12	Date of Verification	9/5/25
13	**Plagiarism Details: (to attach the final report from the software)**	

Chapter	Title of the Chapter	Percentage of similarity index (including self citation)	Percentage of similarity index (Excluding self citation)	% of plagiarism after excluding Quotes, Bibliography, etc.,
1	INTRODUCTION	2%.	0%.	0%.
2	LITERATURE SURVEY	2%.	-	-
3	TECHNICAL SPECIFICATIONS	-	1%.	-
4	ARCHITECTURE	<1%.	0%.	0%.
5	MODULES	2%.	-	-
6	PROJECT DEMONSTRATION AND RESULTS	<1%.	-	1%.
7	CONCLUSION	-	-	1%.
8	FUTURE ENHANCEMENTS	<1%.	-	-
9				
10				
	Appendices			

I/ We declare that the above information have been verified and found true to the best of my / our knowledge.

Signature of the Candidate	**Name & Signature of the Staff (Who uses the plagiarism check software)**
Name & Signature of the Supervisor/ Guide	**Name & Signature of the Co-Supervisor/Co-Guide**

Name & Signature of the HOD

Department of Computing Technologies
Kattankulathur 603 203
SRM IST