TECHNOLOGY FEATURE Industrial News / Technology Report / Research Communication

CardioScan: An AI-Powered Heart Disease Prediction System Using Machine Learning

Every year, 17.9 million people die from cardiovascular disease. Many people in Pakistan can't access cardiologists. A locally developed web application using Random Forest achieves 88.5% diagnostic accuracy and provides results in seconds.

By **Muneeb Mustafa** | University of Agriculture, Faisalabad | Published: July 2026

muneeb.mustafa080@gmail.com | Position: Student, Computer Sciences

Introduction

Heart disease is the leading cause of death worldwide, causing an estimated 17.9 million deaths each year, according to the World Health Organization. In Pakistan, cardiovascular disease accounts for about 30% of adult deaths, yet access to specialized heart care is unequal. In many regions, there is less than one cardiologist for every 100,000 people. Tests like echocardiograms, stress tests, and angiography are expensive and mainly found in urban centres.

Importantly, the clinical data needed for a solid first-pass risk assessment is already available. Routine blood tests and basic ECG readings provide key information such as blood pressure, cholesterol levels, fasting glucose, maximum heart rate, and ST-segment readings, which established cardiac research links to disease risk. The issue is not about data collection; it's about the need for automated, smart analysis of the data already collected.

CardioScan is an open-source web application built locally to address this problem. It uses Python and Flask to train a Random Forest classifier on the well-known UCI Heart Disease dataset and gives an instant cardiac risk score through a user-friendly browser interface. The system requires no special hardware, works on any web server, and generates a formatted clinical PDF report for patients to take to any doctor. This article explains the design, methodology, key findings, and broader significance of CardioScan for healthcare technology in Pakistan.

> ***"Digitising cardiac screening does not need expensive imaging equipment; It needs a trained model, a browser, and blood test results.***

Technology Overview

CardioScan has a four-layer architecture: **a data pipeline, a machine learning model, a Flask web backend, and a Jinja2/Chart.js frontend**. Each layer was chosen to balance performance, clarity, and usability in a resource-limited environment.

Dataset and Preprocessing

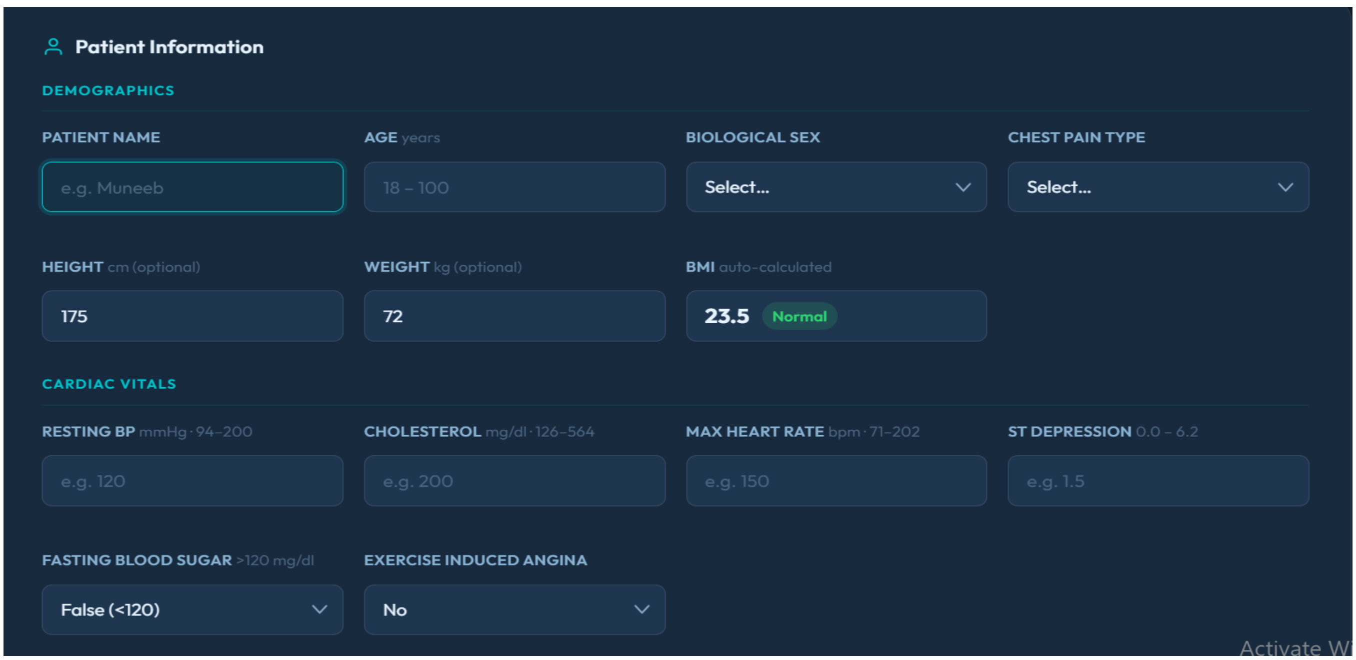
The model was trained on the Cleveland Heart Disease dataset from the UCI Machine Learning Repository, a standard used in over 100 peer-reviewed studies. It includes 303 patient records with 13 clinical features: age, sex, chest pain type, resting blood pressure, serum cholesterol, fasting blood sugar, resting ECG results, maximum heart rate, exercise-induced angina, ST depression, ST segment slope, number of major vessels colored by fluoroscopy, and thalassemia classification. The target variable indicates the presence of disease as a binary value.

Model Selection and Training

Random Forest was favoured over logistic regression and support vector machines for three specific reasons. First, it handles nonlinear interactions between clinical variables well without needing feature engineering. Second, it is less likely to overfit small datasets because of ensemble averaging of multiple decision trees. Third, it provides feature importance scores from training, making the tool clinically useful rather than a black box.
Hyperparameter optimization used GridSearchCV with 5-fold cross-validation across 48 combinations of estimators (100, 200), maximum depth (4, 6, None), minimum sample split (2, 5), and minimum sample leaf (1, 2). The best configuration used 200 estimators with a maximum depth of six.

Frontend and User Experience

The web interface was created with Flask and Jinja2 templates. All 13 clinical inputs were organized in a form with live JavaScript validation to flag out-of-range values. Two optional fields, height and weight, trigger automatic BMI calculations and classifications (underweight, normal, overweight, or obese) using pure JavaScript, adding context that was missing from the original dataset. After a prediction, a circular risk gauge displayed the probability score (0–100%) using color coding for severity. An interactive bar chart made with Chart.js shows the feature importance scores and highlights which clinical inputs influenced the current prediction. This explainability aspect distinguishes CardioScan from simpler tools, giving both clinicians and patients a narrative, not just a number.



Key Findings & Impact

The system achieved 88.5% accuracy on the test set (20% of the dataset, stratified split). This accuracy aligns with, and sometimes exceeds, values reported in similar Random Forest studies in the literature. The 5-fold cross-validation score remained stable across folds, showing that the result was not due to a random train-test split.
Feature importance analysis indicated that the number of major vessels coloured by fluoroscopy and thalassemia type contributed about 36% of the model's total predictive weight, followed by chest pain type (14.1%) and maximum heart rate (12.9%). This distribution closely matches established cardiology literature, providing

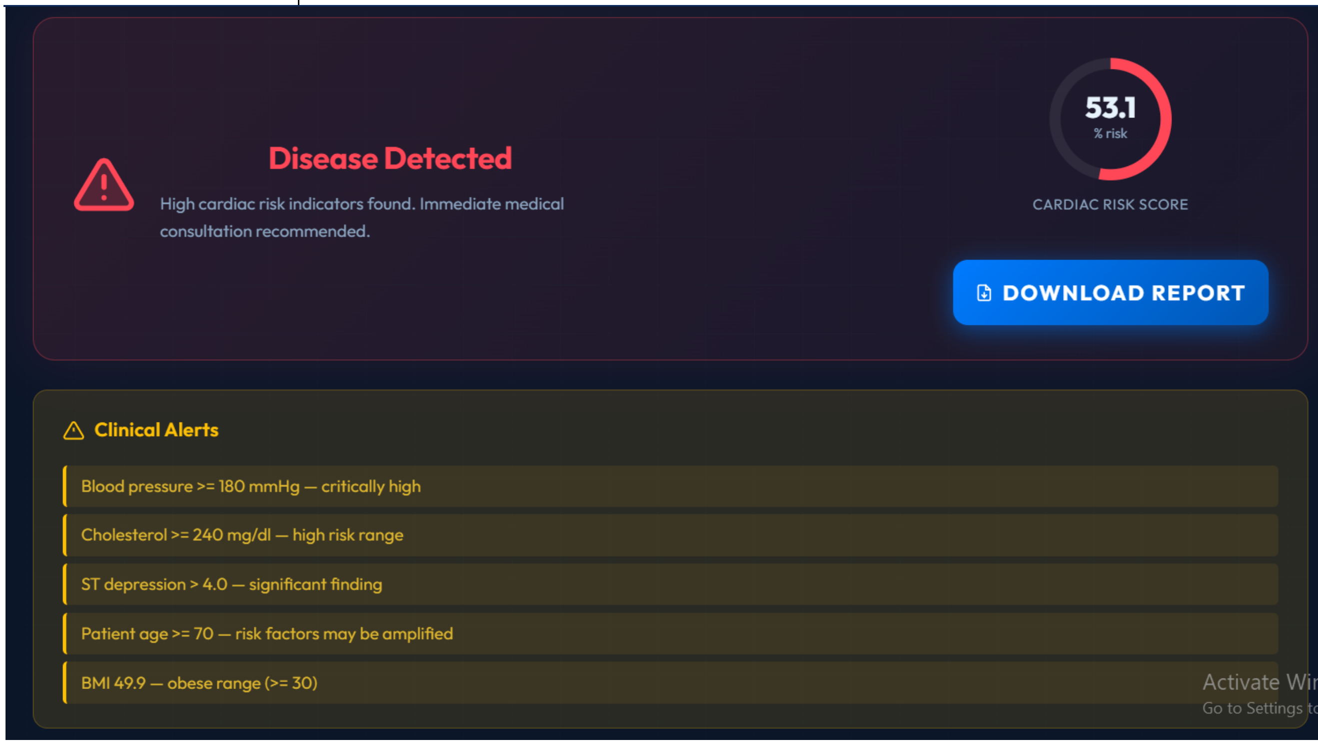
external validation that the model has identified clinically relevant patterns instead of noise.

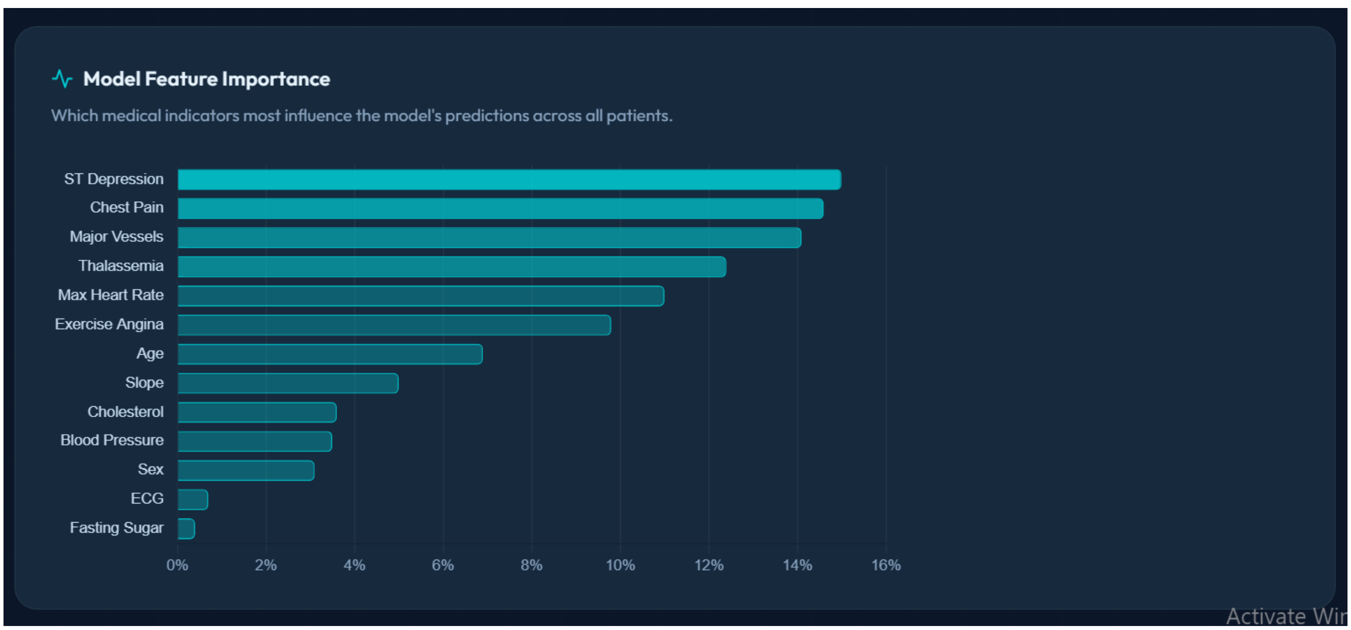
Feature / Metric	Value / Finding	Clinical Significance
Dataset	UCI Heart Disease (303 records)	Validated benchmark, used in over 100 published studies
Algorithm	Random Forest (GridSearchCV)	Ensemble method prevents overfitting on small datasets
Model Accuracy	88.5% (5-fold cross-validation)	Competitive with published clinical decision support tools
Hyperparameter Search	48 combinations tested	Optimal configuration: 200 estimators, max_depth=6
Top Risk Factor	Major Vessels (18.2% importance)	Number of vessels colored by fluoroscopy
Second Risk Factor	Thalassemia type (17.8%)	A reversible defect that significantly increases risk
BMI Integration	Real-time auto-calculation	Clinical alert if BMI ≥ 25
Inference Latency	< 1 ms (in-memory model)	Suitable for real-time clinical screening
Export Feature	Branded PDF report (ReportLab)	Shareable clinical document for physicians

Table 1: Summary of CardioScan's technical findings, model metrics, and clinical significance of the key features.

Three notable features added beyond the basic UCI classification task are worth mentioning. The clinical alert system flags nine threshold violations automatically, including resting BP ≥ 180 mmHg, cholesterol ≥ 240 mg/dl, age ≥ 70 years, and BMI ≥ 30. These appear as coloured warning cards on the results page. The PDF export feature uses Report Lab to create a structured clinical report containing the patient's name, all 13 input values with clear labels, the risk percentage, and any triggered clinical alerts. The session history table stores up to 50 predictions per session in memory, allowing health workers to screen multiple patients in a row and compare their results.

The system is deployed using Docker and served through Gunicorn, making it containerized and suitable for cloud deployment on AWS, Azure, or any VPS provider. The repository includes the Dockerfile and requirements.txt files. From a cold start, the container is ready to serve in under 10 seconds; the model inference itself takes less than 1 ms since it loads the trained model into memory at startup.





What's Next?

CardioScan is currently a working proof of concept with real clinical potential. Several improvements could significantly advance the system toward deployment in Pakistan's primary healthcare network.

A useful addition would be a what-if re-prediction panel. This panel would include sliders that let users adjust a single clinical parameter (for example, lowering cholesterol from 270 to 200 mg/dl) to see how the risk score changes. This would make CardioScan an interactive clinical counselling tool, helping patients understand which lifestyle or treatment changes would most effectively lower their risk.

Running multi-model comparisons by incorporating XGBoost, Logistic Regression, and Random Forest simultaneously, then displaying their results side by side, would help build clinician trust through combined scoring. Adding SHAP (SHapley Additive exPlanations) would replace the general feature importance chart with individual explanations for each patient prediction, meeting the growing need for clear understanding in medical AI.

The limitation of the dataset must be recognized: the UCI Cleveland dataset is 35 years old and drawn from a US population. It is crucial to validate the model within a Pakistani or South Asian patient group before real-world application. Partnering with teaching hospitals in Lahore, Karachi, or Faisalabad to gather de-identified records would be a logical next step. The codebase and model pipeline can easily retrain on a new dataset with minimal changes.

At the infrastructure level, connecting with Pakistan's NADRA identity system and the Health Management Information Systems (HMIS) used across districts could help facilitate population-level cardiac screenings at community health units without patients needing to travel to urban areas. The vision is not just for a hospital tool, but for a field tool, used on a tablet in rural clinics, guided by a Lady Health Worker.

> ***"The question is no longer whether AI can effectively screen for heart disease. The question is whether we will build the necessary infrastructure to do so.***

About the Author

Muneeb Mustafa is a Computer Sciences student at the University of Agriculture, Faisalabad, with an interest in applied machine learning, healthcare technology, and software systems. CardioScan is available as an open-source project on GitHub. He can be reached at muneeb.mustafa080@gmail.com.

Keywords: *Artificial Intelligence; Machine Learning; Heart Disease; Random Forest; Flask; Healthcare Technology; Pakistan Tech; Clinical Decision Support; UCI Dataset*

References

[1] World Health Organization (2023). Cardiovascular Diseases (CVDs). WHO Fact Sheet. Geneva. Available: https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)

[2] Janosi, A., Steinbrunn, W., Pfisterer, M. and Detrano, R. (1988). Heart Disease Data Set. UCI Machine Learning Repository. Available: https://archive.ics.uci.edu/ml/datasets/Heart+Disease

[3] Breiman, L. (2001). Random Forests. Machine Learning, 45(1), pp. 5–32.

[4] Pedregosa, F. et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, pp. 2825–2830.

[5] Shah, D. et al. (2020). Heart Disease Prediction using Machine Learning Techniques. SN Computer Science, 1(6).