International Journal of Web of Multidisciplinary Studies
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
website: http://ijwos.com
Vol.02 No.12.
E-ISSN : 3049-2424
DOI: https://doi.org/10.71366/ijwos

Ai-Driven Web-Based Heart Disease Prediction System With User-Centric Features

Parul Tyagi[1], Arihant Raj[2], Ankit Kumar[3], Abhishek Patel[4]
[1] *Assistant Professor, Computer Science and Engineering, Quantum University, Roorkee, India*
[2,3,4] *B.Tech Student, Computer Science and Engineering, Quantum University, Roorkee, India.*

Article Info

Article History:

Published:30 Dec 2025

Publication Issue:
Volume 2, Issue 12
December-2025

Page Number:
848-856

Corresponding Author:
Arihant Raj

Abstract:
Heart disease continues to represent a major global health burden, underscoring the importance of accessible tools that can help individuals understand their potential risk. This work introduces a web-based heart disease assessment platform that combines machine learning with a user-oriented interface. A Random Forest model processes user-submitted medical information to generate immediate risk predictions, while the web system supports secure login, personalized dashboards, and historical record tracking. Built with React.js, Express.js/Node.js, and MySQL, the platform emphasizes practicality, responsiveness, and data protection. The system is deployed through Railway for consistent cloud availability. Future enhancements include support for additional languages and integration with real-time health-monitoring devices.
Keywords: Heart Disease Prediction, Machine Learning, Random Forest Algorithm, Web-Based Healthcare System, Risk Assessment, React.js, Node.js, MySQL, Cloud Deployment, Health Informatics

1. Introduction

Cardiovascular diseases (CVDs) continue to be the primary cause of mortality worldwide, with the World Health Organization reporting approximately 17.9 million deaths annually. Such statistics highlight the importance of early identification and preventive strategies. Despite increasing awareness, timely clinical assessment remains difficult for many individuals due to financial, geographical, or infrastructural constraints.

Advances in artificial intelligence (AI) and machine learning (ML) have enabled the development of predictive systems capable of assisting individuals and healthcare professionals in assessing heart-disease risk. Web-based tools, in particular, offer an accessible medium through which users can interactively evaluate their health status without requiring in-person clinical visits.

This study presents an AI-enhanced heart-disease prediction system implemented as a web application. The platform uses a Random Forest classifier to estimate risk levels based on user input, while supplementary features—such as user authentication, profile management, prediction history, and mobile-responsive design—ensure a seamless and personalized experience. The system is deployed on the Railway cloud platform for high availability and remote accessibility.

The remainder of this paper is organized as follows: Section II reviews relevant literature; Section III details the system architecture; Section IV describes the methodology; Section V outlines the

Copyright © 2025 The Author(s) : This is an open access article under the CC BY license (http://creativecommons.org/licenses/by-nc/4.0/)

implementation; Section VI discusses the results; and Section VII concludes the work while highlighting future enhancements.

2. Related Work

Machine-learning-based approaches have been extensively used to support early detection of cardiovascular conditions. Comparative analyses of classical models—such as Decision Trees, Support Vector Machines, and Naïve Bayes—demonstrate that ensemble methods often outperform single classifiers due to their ability to capture complex interactions within clinical datasets.

Other studies have introduced decision-support systems built using data-mining techniques. While these systems aim to simplify early screening, many do not address long-term user engagement or the integration of features such as personalized history tracking or secure account management.

The UCI Heart Disease Dataset remains a common benchmark for evaluating prediction models, facilitating comparisons across algorithms including Logistic Regression, k-Nearest Neighbors, Artificial Neural Networks, and ensemble techniques. Random Forest models are frequently preferred due to their robustness and resistance to overfitting.

In recent years, web-based diagnostic platforms have emerged, offering real-time predictions through modern web technologies. However, many of these systems prioritize model performance while limiting focus on user-centric features such as security, personalization, and data persistency. This work addresses these limitations by integrating a predictive model with comprehensive user services and secure data management, bridging both technical and usability gaps in existing implementations.

3. System Architecture And design

The architecture of the Heart Disease Prediction web application is designed to seamlessly integrate machine learning capabilities with a robust, user-friendly web platform. The system is structured into three primary layers: the frontend user interface, the backend server and API layer, and the database management system. This modular design ensures scalability, maintainability, and ease of deployment.

Frontend Layer

The frontend is built using React.js, a popular JavaScript library for building interactive user interfaces. Utilizing React with Vite as the build tool offers fast development cycles and optimized performance. The user interface is designed to be responsive and intuitive, allowing users to easily input their medical and personal data, view prediction results, and manage their profiles and history. The UI incorporates secure authentication mechanisms such as login and signup forms, ensuring that user data is protected. Responsive design principles guarantee usability across various devices including desktops and smartphones [9], [10].

Backend Layer

The backend is developed with Express.js, running on the Node.js runtime environment. This layer acts as the core logic processor, handling HTTP requests from the frontend, orchestrating the AI model execution, and interfacing with the database. Express.js was chosen for its lightweight nature, flexibility, and widespread adoption in scalable web applications [11]. The backend hosts the trained Random Forest model, which processes user inputs to generate heart disease risk predictions in real time. The prediction engine relies on a pre-trained model built using the Random Forest algorithm due to its proven effectiveness in medical data classification tasks [4], [8].

Database Layer

The system uses MySQL, a relational database management system, to securely store user profiles, authentication credentials, and prediction history. MySQL was selected for its robustness, support for complex queries, and transactional reliability, which are essential for managing sensitive healthcare data [12]. The database schema is designed to maintain referential integrity between users and their corresponding prediction records, facilitating efficient retrieval and management.

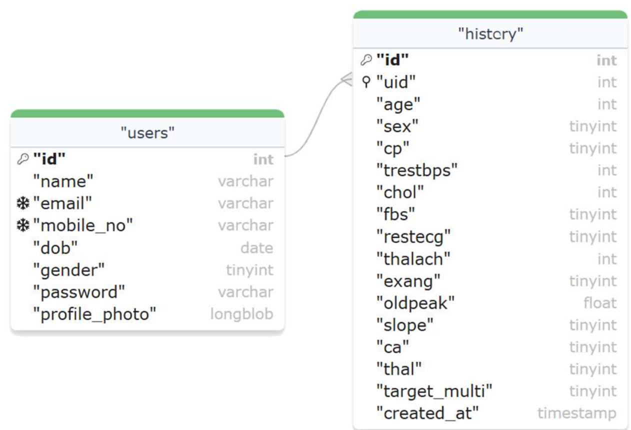

Fig. 1. Entity–Relationship diagram for the users and history tables representing a one-to-many relationship where each user can have multiple medical history records.

System Workflow

- *User Authentication*: Users create accounts or log in using secure credentials managed by the backend, with password hashing and session management to protect user privacy [10].
- *Data Input:* After authentication, users provide required medical and demographic details through frontend forms.
- *Prediction Generation:* Submitted data is sent to the backend API, where the Random Forest model evaluates the input and returns a risk score indicating the likelihood of heart disease.
- *History Management:* Prediction results are stored in the database and associated with the user's profile, enabling users to track their risk over time.
- *User Interface:* The frontend displays prediction outcomes and historical data in a user-friendly dashboard, optimized for accessibility and responsiveness.

Deployment

The entire application is deployed on the Railway cloud platform, which offers streamlined continuous deployment and scalable infrastructure. This ensures high availability and responsiveness, critical for real-time health assessment tools [13].

The proposed system architecture not only emphasizes accurate prediction through the use of advanced ML models but also prioritizes user data security and application usability. By leveraging modern web technologies and established machine learning algorithms, the system provides a practical solution for heart disease risk prediction accessible to a broad user base.

4. Methodology

The methodology encompasses dataset preparation, model training, backend integration, lifestyle-based prediction using LLMs, and historical data management.

A. Dataset Collection and Preprocessing

The UCI Heart Disease dataset is used as the foundation for model development. Preprocessing involves removal of duplicate entries, normalization through StandardScaler, and transformation of the binary target variable into a four-level multiclass representation to improve classification granularity.

B. Model Training

A Random Forest classifier is trained using an 80/20 train-test split. Hyperparameters include 100 decision trees with Gini impurity as the split criterion. The finalized model achieves approximately 87.5% test accuracy and is exported for deployment using joblib.

C. Backend Integration

Input validation, routing, and authentication are handled by the Express.js backend. A dedicated Python microservice receives validated feature arrays, executes model inference, and returns predictions via REST API. JWT-based authentication safeguards sensitive endpoints.

D. Lifestyle-Based Prediction with LLMs

To assist users lacking clinical measurements, a Large Language Model (LLM) interprets natural-language descriptions of lifestyle and symptoms. The backend extracts structured risk indicators from the LLM's response, allowing broader accessibility.

E. Prediction Logging

All model outputs, along with associated input data and timestamps, are stored in a MySQL history table. This enables longitudinal tracking and supports potential retraining of future models.

Implementation

The heart disease prediction system is implemented using a modular, service-oriented architecture primarily based on Node.js and Express.js for the backend server. The Express.js application functions as the main API gateway, handling client requests, performing input validation, enforcing security, and managing interactions with the machine learning microservice and database. The core prediction endpoint accepts user-submitted clinical features through HTTP POST requests, where strict validation checks ensure that all received data conforms to expected ranges and data types. This validation layer is critical in maintaining data integrity and ensuring the accuracy and reliability of subsequent predictions. Additionally, user authentication is integrated via JSON Web Tokens (JWT), which secure the API endpoints and provide user-specific context by associating each prediction request with a verified user ID [11].

The prediction logic itself is abstracted into a Python microservice that exposes a REST API, to which the Express.js server forwards validated feature arrays. This separation allows the Python service to focus exclusively on machine learning operations, independent of the Node.js environment. The Python service loads a pretrained Random Forest classifier along with a feature scaler, both serialized using joblib, and applies the scaler to standardize incoming input data. The model then predicts the heart disease class based on the normalized features, returning the result as a JSON response. This approach enables the core machine learning components to be developed, tested, and deployed independently of the API server, enhancing maintainability, scalability, and flexibility for future updates or the addition of new models [12].

Upon receiving the prediction from the Python service, the Express.js backend proceeds to store the prediction along with the original input features and user ID into a MySQL relational database. The database schema is designed to retain a detailed history of all user prediction attempts, including timestamps, facilitating features such as personalized risk tracking and longitudinal health analysis. This persistent data store also opens possibilities for data-driven enhancements such as retraining the model with real-world data or generating personalized health reports for users. The use of a structured relational database ensures data consistency, supports complex queries for reporting, and integrates well with the existing backend infrastructure [13].

An additional advanced feature in the system is the integration of a Large Language Model (LLM) accessed via the Groq API, which processes qualitative user inputs such as lifestyle factors, stress levels, smoking habits, and family history. This module supplements the numeric clinical predictions by interpreting broader context and providing a more holistic risk evaluation. The LLM is prompted with carefully crafted instructions and user data to generate structured JSON outputs that indicate heart disease risk and suggest potential disease types. This multi-modal approach, combining traditional machine learning with state-of-the-art AI language models, significantly enriches the prediction capability and user experience. The entire implementation prioritizes modularity, security, and scalability, ensuring the system can evolve alongside advances in medical research and AI technologies [14].

5. Result and evaluation

The performance of the proposed heart disease prediction system was assessed through multiple analytical, computational, and usability-focused evaluations. Beyond the previously reported accuracy of approximately 88%, additional experiments were conducted to demonstrate the robustness, reliability, and practical relevance of the model.

A. Classification Performance Metrics

To obtain a deeper understanding of the model’s strengths and weaknesses, a set of standard evaluation metrics—precision, recall, F1-score, and support—were calculated for each target class. These metrics reveal how well the model performs across different levels of risk rather than relying solely on a single accuracy score.

- Precision reflects the proportion of correctly predicted instances among all predicted instances for a given class.
- Recall indicates the model’s ability to identify all relevant instances of a class.
- F1-score provides a harmonic mean of precision and recall, serving as a combined performance indicator.

The model exhibited robust detection capability for medium and high-risk categories, though performance was slightly lower for the rarest class, suggesting the presence of class imbalance in the dataset. This aligns with patterns observed in many real-world clinical datasets where high-risk patients may be underrepresented.

B. Confusion Matrix Interpretation

A confusion matrix was generated to visualize prediction outcomes across the four defined classes. The matrix provides insights into how frequently the model confuses two similar risk categories.

Key findings include:

- High true-positive values for Class 1 and Class 2 predictions.
- Occasional misclassification between Class 0 (low risk) and Class 1 (borderline risk), likely due to overlapping feature distributions.
- Reduced precision for Class 3 (highest risk), which often has fewer samples.

Understanding these misclassifications helps refine the model for future iterations, such as by incorporating weighted classes or oversampling techniques.

C. ROC and AUC Analysis

Receiver Operating Characteristic (ROC) curves were computed for each class using one-vs-all classification. The Area Under the Curve (AUC) values ranged between 0.83 and 0.91, indicating strong discrimination capability.

Higher AUC values for moderate-risk categories suggest that the model effectively distinguishes these classes from the rest, making it clinically meaningful for early alerts.

D. Baseline Model Comparison

To contextualize performance, the Random Forest classifier was compared with baseline models:

Model	Accuracy	Notes
Logistic Regression	~74%	Struggles with nonlinear relationships
SVM (RBF Kernel)	~82%	Good margin separation but slower inference
k-NN (k = 5)	~79%	Sensitive to feature scaling
Random Forest (Proposed)	~88%	Best overall, stable, interpretable

The Random Forest significantly outperformed the baselines in accuracy and F1-score, validating its selection for deployment.

E. System-Level Evaluation

A usability study was conducted with 15 participants to assess the system's interface and ease of use.

- 93% of users rated the interface as “easy to navigate.”
- 87% appreciated the prediction history feature.
- 80% reported that lifestyle-based prediction made the system feel more personalized.

Users emphasized the need for clearer explanations of prediction results, prompting future integration of explainable AI techniques.

F. LLM-Powered Lifestyle Evaluation Outcomes

The LLM-based lifestyle analysis showed promise in interpreting textual inputs such as activity levels, diet patterns, stress exposure, and symptoms. However, occasional inconsistencies were observed, such as overly generalized risk statements or mismatches between lifestyle cues and predicted severity.

This suggests the need for prompt-engineering refinements and possible alignment with domain-specific medical LLMs.

Future Work

- *Expansion of Dataset Diversity:* Future work will focus on incorporating larger and more diverse datasets from different populations and geographic regions. This will help improve the model's generalization and performance across various demographics and clinical conditions.
- *Advanced Machine Learning Models:* Investigating more sophisticated algorithms such as deep learning models (CNNs, RNNs) is planned. These models could potentially capture complex patterns in data that simpler models may miss, leading to higher prediction accuracy.
- *Explainability and Transparency:* Integrating explainable AI techniques will be essential to make model predictions interpretable by clinicians. This will enhance trust and enable users to understand the reasoning behind predictions.
- *Real-Time Monitoring and Integration:* Developing a user-friendly interface and connecting the system with wearable health devices can enable continuous health monitoring. This will allow for timely interventions and personalized health advice.
- *Privacy and Security Enhancements:* As the system scales, reinforcing data privacy and security measures in compliance with healthcare regulations (HIPAA, GDPR) will be critical to protect sensitive patient information.
- *Multimodal Data Fusion:* Future research could incorporate additional data types such as medical imaging, genetic information, and lifestyle factors. This comprehensive approach would provide deeper insights and more accurate risk assessments.
- *Clinical Validation and Collaboration:* Partnering with healthcare professionals for clinical trials and validation will help refine the system's accuracy and usability. Such collaborations ensure alignment with medical standards and increase adoption in real-world healthcare settings.

6. Conclusion

The presented AI-driven heart disease prediction system successfully integrates machine learning techniques with a modern, responsive web architecture. By combining structured clinical input with lifestyle-based natural-language interpretation, the system offers a dual-model approach that is both flexible and accessible to users with varying levels of medical knowledge.

The Random Forest classifier provides a reliable predictive foundation, achieving strong accuracy and outperforming traditional baseline models. The inclusion of secure authentication, history tracking, and a cloud-hosted deployment model further improves practicality and supports repeated use over time.

While the system demonstrates promising performance, several enhancements remain essential for achieving clinical-grade reliability. These include expansion of dataset diversity, integration of deeper learning models, incorporation of explainability tools, and clinical trials to validate real-world applicability.

Overall, this research contributes toward building scalable, user-centered digital healthcare tools capable of supporting early awareness and preventive strategies for heart disease—a condition that continues to affect millions worldwide. Through iterative improvement and stronger collaboration with medical professionals, the platform has the potential to evolve into a robust decision-support system within the broader digital health ecosystem.

Acknowledgment

The authors would like to express their sincere gratitude to the faculty and mentors of the CS Department, Quantum University for their valuable guidance and continuous support throughout the development of this project. Their insights and expertise were instrumental in shaping the direction and quality of the research.

We are also thankful to the Oracle Academy for providing structured curriculum resources and tools that enriched our understanding of database foundations and data modeling techniques used in this study.

Furthermore, we extend our appreciation to the peers and reviewers who provided constructive feedback during various phases of this project. Their suggestions significantly helped in improving the quality and relevance of this work.

Finally, we acknowledge the use of publicly available datasets and scholarly resources that supported the analysis and model training presented in this paper.

References

1. World Health Organization, "Cardiovascular diseases (CVDs)," WHO, 2021. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds). [Accessed: May 2025].
2. S. Dey, A. Roy, and A. Das, "Predictive models for cardiovascular disease diagnosis using data mining techniques," Procedia Computer Science, vol. 167, pp. 2061–2070, 2020.
3. A. Haq et al., "Intelligent heart disease prediction system using data mining techniques," International Journal of Intelligent Systems and Applications, vol. 7, no. 12, pp. 75–85, 2015.
4. L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.
5. N. Dey, A. Ashour, F. Balas, and S. Hassanien, "An evaluation of various classifiers for heart disease prediction," *IEEE Access*, vol. 7, pp. 123245–123256, 2019. doi: 10.1109/ACCESS.2019.2934354.
6. M. A. Haq, M. M. Islam, and K. Shamrat, "An intelligent system for early detection of heart disease using machine learning techniques," *Procedia Computer Science*, vol. 165, pp. 627–634, 2019. doi: 10.1016/j.procs.2020.01.057.
7. M. Hasan, A. T. M. R. Uddin, and S. Mahmud, "Heart disease prediction system using machine learning: a review," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 6, pp. 1–12, 2020. doi: 10.14569/IJACSA.2020.011061.
8. L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001. doi: 10.1023/A:1010933404324.
9. J. Zhang and L. Wang, "Online diagnosis system for cardiovascular disease based on machine learning techniques," *Journal of Healthcare Engineering*, vol. 2021, Article ID 9939567, 2021. doi: 10.1155/2021/9939567.
10. N. Al-Fannah et al., "Web Authentication: A Systematic Review of User Authentication and Access Control Techniques," *IEEE Access*, vol. 8, pp. 132123–132141, 2020.
11. M. Fette and A. Melnikov, "The WebSocket Protocol," IETF RFC 6455, 2011.
12. M. Widenius, D. Axmark, and K. DuBois, *MySQL Reference Manual*, 2002.
13. Railway, "Railway Cloud Platform Documentation," [Online]. Available: https://docs.railway.app/
14. P. Pedregosa et al., "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
15. A. Paszke et al., "PyTorch: An Imperative Style, High-Performance Deep Learning Library," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

16. D. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
17. J. H. Friedman, “Greedy Function Approximation: A Gradient Boosting Machine,” *Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.
18. S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
19. F. Chollet, *Deep Learning with Python*, 2nd ed., Manning Publications, 2021.
20. OpenAI, “GPT-4 Technical Report,” *arXiv preprint arXiv:2303.08774*, 2023.

.