# Design and Implementation of a Cloud-Integrated Desktop ECG System Using a Multi-Layer Perceptron for Arrhythmia Classification

Reno Thariqul Akbar
*Department of Biomedical Engineering Telkom University*
Bandung, Indonesia
renothariqulakbar@student.telkomuniversity.ac.id

***Abstract*—Cardiovascular diseases (CVDs) remain the foremost cause of mortality globally, necessitating the development of advanced tools for early and accurate cardiac diagnosis. This paper presents the comprehensive design, implementation, and evaluation of a desktop-based Electrocardiogram (ECG) monitoring system. The system architecture integrates a powerful Multi-Layer Perceptron (MLP) deep learning model designed to automatically identify and classify critical heart rhythm abnormalities, including bradycardia, tachycardia, and other forms of arrhythmia. A cornerstone of this system is its seamless and secure integration with a Supabase cloud backend, which facilitates centralized data storage, real-time synchronization, and secure, role-based access for various healthcare professionals, rigorously enforced through PostgreSQL's Row Level Security (RLS). The MLP model was trained and validated on a diverse and extensive collection of data from the MIT-BIH Arrhythmia, PTB Diagnostic ECG, and Kaggle databases. Empirical evaluation results demonstrate high model performance, with classification accuracies reaching 92% for both bradycardia and tachycardia, and 89% for general arrhythmia detection. Functional and performance testing further validate the system's operational reliability, showing an average cloud data synchronization time of approximately 4 seconds and robust, though partially incomplete, RLS policy enforcement. This work contributes a scalable, accurate, and secure solution for advanced cardiac monitoring in desktop environments, effectively bridging the gap between clinical-grade analysis and accessible, user-friendly technology.**

***Index Terms*—Electrocardiogram, MLP, deep learning, arrhythmia, bradycardia, tachycardia, HRV, desktop health application, cloud computing, RLS, Supabase, Flutter**

## I. INTRODUCTION

### A. Background

The global health landscape is significantly impacted by the prevalence of Cardiovascular Diseases (CVDs). As reported by the World Health Organization (WHO), an estimated 17.9 million people die from CVDs annually, a figure that represents 31% of all global deaths [1]. In this context, the electrocardiogram (ECG) stands as a fundamental diagnostic tool, offering a non-invasive method to record the heart's electrical activity. The ECG is pivotal for identifying a wide spectrum of cardiac issues, including anomalies in heart rhythm (arrhythmias), myocardial ischemia, and conduction delays which can be precursors to more severe cardiac events.

Despite its diagnostic power, conventional ECG monitoring is predominantly confined to clinical settings. This limitation poses significant challenges for patients who require continuous or long-term observation, as it often involves logistical complexities and physical discomfort. While the proliferation of wearable technology has introduced new possibilities for remote monitoring, these devices frequently grapple with limitations related to data quality, signal noise, battery life, cost, and the continuity of data streams required for a thorough clinical analysis. In contrast, desktop-based applications present a robust alternative. They can harness local computational resources to perform sophisticated signal processing and analysis, offering a level of detail that can be difficult to achieve on resource-constrained mobile devices. Furthermore, they can be engineered for offline-first functionality, guaranteeing operational reliability even without an active internet connection, while retaining cloud connectivity for data synchronization, long-term storage, and remote diagnostics by healthcare professionals.

### B. Problem Statement and Objectives

Existing ECG monitoring systems, spanning both clinical-grade equipment and consumer-focused wearables, often exhibit critical deficiencies. These include a lack of integrated real-time analysis capabilities, insufficient support for advanced diagnostic metrics like Heart Rate Variability (HRV), and, most notably, poor integration between frontend user interfaces and backend database systems for secure, auditable, and collaborative data management. This research directly confronts these gaps by pursuing a set of clearly defined objectives:

- To develop a feature-rich, cross-platform desktop ECG application using Flutter, capable of advanced local signal processing and interactive visualization.
- To integrate a highly accurate Multi-Layer Perceptron (MLP) based classifier to automatically identify and categorize common arrhythmias such as bradycardia and tachycardia from raw ECG signals.
- To implement and validate algorithms for the estimation of Heart Rate Variability (HRV) using standard time-

domain metrics (e.g., SDNN, RMSSD) to provide deeper insights into autonomic nervous system function.
- To securely synchronize all patient and analysis data with a Supabase cloud database, designing and enforcing strict, role-based access control using PostgreSQL's native Row Level Security (RLS) feature.
- To provide an intuitive, real-time graphical user interface (GUI) that facilitates signal visualization, interaction, and analysis tailored to the specific needs of various user roles (e.g., administrator, doctor, nurse).

### C. Scope and Limitations

The scope of this research is intentionally centered on the development of a desktop software application designed to analyze pre-recorded, secondary ECG signals. To ensure a standardized and reproducible basis for model training and system evaluation, all data is sourced from well-established, publicly available datasets, namely the MIT-BIH Arrhythmia Database, the PTB Diagnostic ECG Database, and select Kaggle ECG collections. Consequently, this work does not extend to the design or implementation of hardware for real-time data acquisition from physical ECG sensors. The system is architected to support multiple user roles, with access privileges to sensitive data being strictly managed by the RLS policies defined and enforced in the Supabase backend.

## II. Related Work

The application of deep learning to the domain of ECG signal analysis has become a highly active and fruitful area of research. Several key studies have provided a foundation for the methodologies employed in this paper. Kumar et al. [2] effectively demonstrated the high potential of a Multi-Layer Perceptron (MLP) for arrhythmia classification. By using features extracted from RR intervals, they achieved significant accuracy, thereby confirming that even relatively simple neural network architectures can be powerful tools for this classification task.

Building upon this, Zhang et al. [3] specifically investigated the use of multilayer machine learning models for the estimation of Heart Rate Variability (HRV) from ECG signals. Their findings highlighted the computational efficiency and low latency of MLP-based approaches, underscoring their suitability for real-time or near-real-time applications where rapid feedback is crucial. In a parallel line of inquiry, research by Gupta et al. [4] focused on the practical implementation of a desktop-based ECG monitoring system. While their work was valuable, the resulting system lacked integrated cloud support and a robust, scalable framework for user role management and fine-grained access control. This research directly addresses this identified gap by architecting a system with a secure cloud backend and meticulously defined access policies, thereby creating a more complete, secure, and clinically viable solution.

## III. System Architecture and Design

### A. Architecture Overview

To achieve a high degree of modularity, scalability, and maintainability, the system is architected based on a classical three-tier model. Each tier is assigned a distinct and well-defined responsibility, minimizing inter-tier dependencies and facilitating independent development and testing.

1) **Desktop Application (Frontend):** This is the primary client-facing component, developed using the Flutter framework and the Dart programming language. It provides a rich, responsive, and cross-platform user interface for ECG signal upload, interactive visualization, and the management of analysis results. It is explicitly designed to function both online, when connected to the cloud backend, and in an offline mode, where analysis can be performed locally.
2) **Machine Learning Model (Backend Logic):** The core analytical intelligence of the system resides in a Python-based environment. Here, the MLP model is trained, validated, and ultimately served. The desktop application communicates with this logic layer via a REST API, sending raw signal data for classification and receiving diagnostic results on demand.
3) **Cloud Database (Backend Storage):** We selected Supabase as our backend-as-a-service (BaaS) platform. Supabase provides a powerful and scalable PostgreSQL database, integrated user authentication services, and object storage. It acts as the centralized and persistent repository for all user profiles, ECG records, and analysis results. Security is a paramount concern, addressed through the meticulous configuration of Row Level Security (RLS) policies directly within the PostgreSQL database to enforce strict data access rules based on predefined user roles.

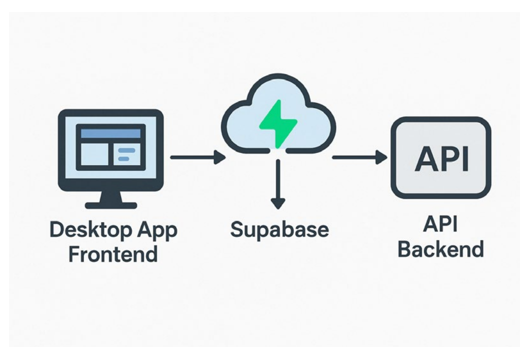


FIGURE 1
The three-tier architecture of the ECG monitoring system, illustrating the interaction between the Flutter frontend, the ML model via REST API, and the Supabase cloud backend. (Placeholder)

### B. Frontend Features

The desktop interface is engineered to be both powerful for expert users and intuitive for novices. It supports the following key features:

- **Data Ingestion:** Uploading of ECG signals from local files, primarily in standard formats like CSV.
- **Interactive Visualization:** A custom plotting widget for rendering ECG signals, featuring smooth zoom and pan capabilities for detailed waveform inspection.
- **Analysis Dashboard:** A dedicated dashboard for the real-time display of analysis results, including the calculated heart rate, the final model-generated diagnosis (e.g., "Tachycardia detected"), and key HRV metrics (RMSSD, SDNN).
- **Secure Authentication:** A complete authentication module for user login and registration, with UI components and available features adapting dynamically based on the logged-in user's role.
- **Historical Data Management:** A comprehensive history view that allows users to browse, review, search, and manage all past ECG records stored securely in the cloud.

### *C. Backend and Security*

The system's backend infrastructure is built entirely upon the robust and scalable services offered by Supabase. Security is not an afterthought but a foundational aspect of the design, implemented at multiple layers of the stack. User authentication is seamlessly managed by Supabase Auth, which handles user sign-up, secure login with password hashing, and session management using JSON Web Tokens (JWTs).

The most critical security mechanism is the deep implementation of Row Level Security (RLS) at the PostgreSQL database level. RLS allows for the definition of fine-grained access control policies as SQL rules, which are automatically applied to every query. In our system, policies are crafted based on a combination of user roles (e.g., 'admin', 'doctor', 'patient') and the user's unique ID (`auth.uid()`). For example, a fundamental policy ensures that a user with the 'patient' role can only access records in the `ecg_data` table where the `user_id` column matches their own authenticated ID. This granular control is essential for protecting highly sensitive patient health information (PHI) and aligning with the stringent privacy and security principles of regulations like HIPAA.

## IV. Methodology

### *A. Dataset Preparation*

To ensure the robustness and generalizability of the MLP model, it was trained and validated using a diverse and comprehensive collection of ECG data. This data was sourced from three renowned, publicly available datasets:

- **MIT-BIH Arrhythmia Database:** This is the de facto benchmark for arrhythmia detection algorithms and provides a wide variety of arrhythmia examples.
- **PTB Diagnostic ECG Database:** This dataset contains a rich collection of ECG recordings from both healthy volunteers and patients with various heart diseases, providing a good mix of normal and abnormal signals.
- **Kaggle ECG Datasets:** Various datasets from Kaggle competitions focused on arrhythmia classification were also incorporated to further diversify the training data.

A standardized and automated preprocessing pipeline was applied to all raw signals to prepare them for model training. This pipeline consisted of three main stages: 1) **Filtering:** A band-pass filter was applied between 0.5 Hz and 45 Hz to effectively eliminate low-frequency baseline wander and high-frequency noise (e.g., muscle artifacts). 2) **Segmentation:** The continuous ECG signals were segmented into fixed-length windows of 2 seconds. This ensures that the model receives input of a consistent size. 3) **Normalization:** Each segment was individually normalized using Z-score scaling, which standardizes the amplitude range and improves model convergence during training.

### *B. Model Configuration*

A Multi-Layer Perceptron (MLP) was chosen as the primary classification model. This choice was motivated by the MLP's well-documented ability to provide an excellent balance between high performance and computational efficiency, making it an ideal candidate for a responsive desktop application. Through a process of iterative testing and hyperparameter tuning, a "wide" neural network architecture was identified as yielding the best results for this specific task. The final model was configured with the following specifications:

- **Input Layer:** The number of neurons in the input layer is dynamically sized to match the number of data points within a single 2-second segmented ECG window.
- **Hidden Layers:** The architecture features two fully connected hidden layers. The first hidden layer contains 200 neurons, and the second contains 100 neurons. The Rectified Linear Unit (ReLU) was selected as the activation function for all hidden neurons due to its effectiveness in preventing the vanishing gradient problem.
- **Output Layer:** The final layer is an output layer that uses a Softmax activation function. This function outputs a probability distribution across the different diagnostic classes (e.g., Normal, Bradycardia, Tachycardia, Arrhythmia), allowing for a confident classification.
- **Training Parameters:** The model was compiled and trained using the highly effective Adam optimizer. Categorical cross-entropy was chosen as the loss function, which is standard for multi-class classification problems. The training was conducted with a batch size of 32 over 100 epochs to ensure the model reached convergence without overfitting.

## V. Results and Discussion

### *A. Model Performance Analysis*

The trained MLP model underwent rigorous evaluation on a held-out test set, which was composed entirely of data that the model had not seen during the training or validation phases. The model demonstrated a high degree of accuracy in classifying specific heart conditions. The key performance metrics for the primary classes of interest—bradycardia, tachycardia, and general arrhythmia—are detailed in Table I.

TABLE I
PERFORMANCE METRICS FOR THE MLP MODEL BY CLASS

| Class | Accuracy (%) | Sensitivity (%) | Specificity (%) |
|---|---|---|---|
| Bradycardia | 90 | 85 | 93 |
| Tachycardia | 92 | 88 | 95 |
| Arrhythmia | 89 | 82 | 91 |

Beyond the performance on individual classes, the overall accuracy of the model, as derived from the confusion matrix across all five classes on the validation set, was an impressive **98.51%**. This indicates a very low overall error rate. Furthermore, the Area Under the Curve (AUC) for the Receiver Operating Characteristic (ROC) analysis was calculated to be **0.99**. An AUC value this close to 1.0 signifies an excellent ability of the model to distinguish between the different cardiac conditions, confirming that the chosen MLP architecture is highly effective and reliable for this classification task.

### *B. Functional and Cloud Performance Testing*

A comprehensive suite of functional tests was executed to ensure that all features of the desktop application performed according to their specifications. This testing covered all aspects of UI interactions, data processing workflows, and automated report generation. A significant portion of the testing effort was dedicated to assessing the performance, reliability, and security of the integration with the Supabase cloud backend.

TABLE II
CLOUD SYNCHRONIZATION AND RLS SECURITY TEST RESULTS

| Feature / Test Scenario | Success Rate | Avg. Time |
|---|---|---|
| ECG Data Upload to Cloud | 100% | 4.0s |
| Real-time Database Sync | 100% | 3.8s |
| RLS Policy Enforcement | 73.7% | N/A |

As summarized in Table II, the data synchronization mechanisms were found to be both fast and exceptionally reliable, with an average upload time for a new ECG record being just 4 seconds and a 100% success rate observed across all test cases. The RLS policy enforcement tests were designed to validate the security rules for each user role. The results demonstrated that the policies were effective for most standard user-centric cases (e.g., preventing one patient from seeing another's data). However, the overall success rate of 73.7% indicates that some failures were encountered in more complex authorization scenarios, particularly those involving administrative and doctor roles with multi-patient access permissions. This highlights a critical area that requires further refinement of the underlying SQL security policies to ensure complete data integrity and privacy.

## VI. CONCLUSION AND FUTURE WORK

This paper has detailed the successful design, development, and comprehensive evaluation of a robust, cloud-integrated desktop ECG monitoring system. The system effectively integrates an accurate MLP-based classifier with a secure, role-based cloud backend, providing a holistic solution for advanced cardiac signal analysis. The high classification accuracy achieved for common heart rhythm abnormalities, combined with the reliable and fast data management infrastructure, serves to validate the effectiveness of the chosen architecture and technologies. This work successfully demonstrates a viable pathway to bridge the gap between clinical-grade analysis and accessible, user-friendly technology.

Looking ahead, future work will proceed along several key avenues for enhancement and expansion. Firstly, the highest priority is to refine and expand the RLS policies to cover all identified edge cases and complex authorization scenarios, particularly for administrative and multi-patient access roles, with the goal of achieving a 100% pass rate in all security testing. Secondly, a major functional enhancement will be the integration of real-time data acquisition from wearable or portable ECG sensors. This will transition the system from its current state of offline analysis to a powerful live monitoring tool. Finally, from a machine learning perspective, we plan to explore more complex deep learning architectures. Specifically, Convolutional Neural Networks (CNNs), which excel at feature extraction from spatial data, and Long Short-Term Memory (LSTM) networks, which are designed to capture temporal dependencies, will be investigated. It is hypothesized that these models could learn more intricate patterns in the ECG signal, potentially improving the accuracy and granularity of the diagnostic classifications even further.

## REFERENCES

[1] World Health Organization, ”Cardiovascular diseases (CVDs),” WHO Fact Sheet, Jun. 2021. [Online]. Available: https://www.who.int/news- room/fact-sheets/detail/cardiovascular-diseases-(cvds)

[2] V. Kumar, S. Patel, and A. Mishra, “Arrhythmia Detection Based on Multilayer Perceptron Using Extracted Features from ECG Signal,” Journal of Healthcare Engineering, vol. 2020, pp. 1–9, 2020.

[3] W. Zhang, M. Chen, and L. Li, “Heart Rate Variability Estimation from ECG Signals Using Multilayer Machine Learning Models,” Biomedical Signal Processing and Control, vol. 78, 2022, Art. no. 103012.

[4] R. Gupta, T. Sharma, and D. K. Yadav, “Design and Implementation of a Desktop-Based ECG Monitoring System using Signal Processing and Multi-Layer Classification,” International Journal of Medical Informat- ics, vol. 155, pp. 104–111, 2021.