Universal Library of Medical and Health Sciences Book

ISSN: 3064-6545 | Volume 4, Issue 3

Open Access | PP: 86-95

DOI: https://doi.org/10.70315/uloap.ulmhs.2026.0403007

Architecture of a Cloud Platform for Remote Cardiac Monitoring: A Practical Guide to Scaling and Processing ECG Telemetry

Oleksandr Kydiuk

Abstract

The methodology examines the architecture of a cloud platform for remote cardiac monitoring intended for continuous acquisition, transmission, analysis, and clinical interpretation of ECG telemetry at the scale of a large distributed system. The study's relevance lies in the growing demand for early arrhythmia detection and the transition of cardiology toward a model of prolonged observation outside the hospital. The purpose of the methodology is to formulate a practical guide to the design and scaling of a platform that integrates wearable sensors, secure communication channels, an event-driven backend, storage layers, alerting algorithms, and mechanisms for integration with medical information systems. The scientific novelty of the study lies in consolidating, within a single engineering framework, the requirements for fault tolerance, data security, ECG time-series processing, automatic report generation, and interoperability via HL7 FHIR. It is shown that a multilayer event-driven architecture provides resilient telemetry intake, maintains clinical integrity of the time series, reduces latency in detecting hazardous episodes, and ensures compliance with HIPAA and GDPR requirements. The methodology will be useful for HealthTech engineers, system architects, developers of medical platforms, and specialists in digital healthcare.

Keywords: *Remote Cardiac Monitoring, ECG Telemetry, Cloud Architecture, Microservices, Event-Driven Processing.*

INTRODUCTION

Cardiology is shifting from episodic diagnostics toward a model of continuous remote observation of the patient's condition in everyday life (Hughes et al., 2026). Cardiovascular diseases have remained the leading cause of global mortality for decades, which substantiates the need to implement technologies capable of identifying pathological rhythm changes at early stages, when timely intervention can prevent death or severe complications (Dores et al., 2026). The traditional methodology of ambulatory monitoring, based on portable Holter recorders, presupposes prolonged device wear followed by physical return of the equipment to the clinic for local data download and analysis of the accumulated information (Boriani et al., 2025). This approach, however, is characterized by a temporal gap between the occurrence of a dangerous event and its clinical interpretation, and in some cases, this gap may extend to weeks (Kim et al., 2025). The deployment of cloud platforms and the concept of the Internet of Medical Things enable this transformation by enabling near-real-time data transmission and automating primary filtering and classification of electrocardiographic signals.

The purpose of this guide is to provide engineers, system architects, and HealthTech specialists with a comprehensive methodological framework for the design, deployment, and scaling of high-load systems for the collection and analysis of ECG telemetry. Under conditions of increasing data volume, contemporary infrastructure must be capable of supporting the simultaneous operation of hundreds of thousands of active devices while ensuring compliance with regulatory requirements in personal data protection, such as HIPAA and GDPR (Nandini, 2025). The architectural solution examined in this study is based on a multilayer model in which each layer performs specialized functions, from primary signal processing at the edge to deep retrospective analysis and long-term storage in distributed cloud repositories.

The transition to a real-time remote monitoring model requires resolving a range of technical challenges, including the energy efficiency of wearable sensors, the reliability of wireless data transmission under unstable network conditions, and the creation of fault-tolerant backend systems capable of processing terabytes of high-frequency time-series data. A key aspect concerns both the technical delivery of the signal and its intelligent interpretation. Contemporary

Citation: Oleksandr Kydiuk, "Architecture of a Cloud Platform for Remote Cardiac Monitoring: A Practical Guide to Scaling and Processing ECG Telemetry", Universal Library of Medical and Health Sciences, 2026; 4(3): 86-95. DOI: https://doi.org/10.70315/uloap.ulmhs.2026.0403007.

anomaly detection algorithms based on deep learning make it possible to automate routine ECG analysis operations, thereby releasing medical personnel resources for critical clinical decision-making (Hizem et al., 2025). The following chapters describe each stage in the construction of such a platform in detail, beginning with the organization of secure communication channels and ending with the integration of analytical results into existing hospital information systems.

The framework presented in this study should be regarded as an industry-significant development because it establishes an integrated model for the construction of a cloud platform for remote cardiac monitoring and unifies within a single engineering contour ECG telemetry acquisition, secure data transmission, event-driven processing, rhythm disturbance analysis, generation of clinical reports, and integration with medical systems. Its value lies in describing the entire signal pathway from the wearable sensor to the physician's decision, thereby converting remote monitoring from a set of disjointed solutions into a reproducible architectural foundation for scalable, resilient, and clinically applicable platforms.

CHAPTER 1. ORGANIZATION OF SECURE DEVICE-TO-CLOUD COMMUNICATION

The first architectural layer establishes a reliable, secure data transmission channel from the wearable ECG sensor to the cloud environment. Under ambulatory use conditions, the patient should not be involved in the technical aspects of equipment configuration, which requires the implementation of automatic pairing mechanisms and transparent data transmission. An optimal solution here is to use an intermediate Bluetooth gateway or router that serves as a bridge between the low-energy personal sensor and the global Internet.

Integration of Wearable Devices and Bluetooth Gateways

A wearable device for continuous ECG acquisition operates under constraints related to power consumption and computational resources (Verma et al., 2021). Recording of the heart's electrical activity is usually performed at a sampling frequency of 100 to 500 Hz, generating a continuous data stream that requires efficient management (Ansari et al., 2024). Within the proposed architecture, the wearable sensor uses the Bluetooth Low Energy protocol to transmit data to a personal gateway installed at the patient's place of permanent residence. BLE has been selected as the primary interface due to its ability to provide high device autonomy and sufficient throughput for the transmission of segmented ECG recordings.

The technical implementation of automatic pairing relies on unique device identifiers and preconfigured security profiles. The firmware of the Bluetooth router must support background scanning and automatic connection restoration when the sensor comes into range. One of the gateway's important functions is local data buffering. In the event of temporary Internet unavailability, the router stores incoming information in internal memory, preventing the loss of clinically significant ECG fragments. As soon as communication with the cloud is restored, the gateway initiates a data backfill procedure, prioritizing the transmission of newly arriving signals over archived ones to minimize latency within the alerting system. A comparison of wireless communication technologies is shown in Table 1.

Table 1. Wireless Communication Technologies Comparison (Schärer et al., 2025)

Characteristic	Bluetooth Low Energy, BLE	Wi-Fi, 802.11n/ac	Cellular, NB-IoT / LTE-M
Energy Consumption	Very low	High	Low / Medium
Range	Up to 10–50 meters	Up to 100 meters	Global coverage
Throughput	Up to 2 Mbps	Up to 600+ Mbps	Up to 1 Mbps
Deployment Cost	Low	Medium	High, requires SIM
Primary Role	Sensor-to-gateway communication	Gateway-to-cloud communication	Autonomous sensors

For the practical deployment of such a scheme, begin by configuring the wearable sensor to transmit ECG in 2–5-second packets over a low-power wireless channel to a home gateway that remains constantly near the patient and is pre-bound to a specific device via its unique identifier. In the gateway firmware, define three mandatory rules from the outset. Automatic reconnection after a brief communication interruption. Local preservation of the stream in the event of network access failure. Priority transmission of fresh data after channel restoration, so that alarming events reach the system without critical delay.

For example, if the patient is at home and the sensor transmits ECG at 250 measurements per second, the gateway can store the last 24 hours of data in its internal memory and continue data intake without gaps during an Internet outage. After connectivity is restored, it can first transmit the current 30 seconds of recording for arrhythmia assessment and then gradually upload the archive. Such an order is especially useful in the context of nocturnal monitoring, when it is more important for the physician to receive the current episode of rhythm disturbance immediately than to wait for full transmission of the accumulated array.

Secure Transmission Protocols: MQTT and End-to-End Encryption

For data transmission from the gateway to the cloud platform, the lightweight application-layer protocol Message Queuing

Telemetry Transport is the most appropriate choice (Saha et al., 2024). MQTT operates according to the publisher-subscriber model over the TCP/IP stack and is optimized for work in networks with constrained bandwidth and high latency. In the context of medical systems, the use of quality-of-service levels is important. QoS level 2 guarantees that each ECG telemetry packet is delivered to the server exactly once, preventing signal segment duplication or omission during time-series reconstruction on the backend.

The security of medical data in transit is ensured by requiring TLS version 1.2 or 1.3. At the network level, the gateway and the cloud API gateway perform mutual authentication, in which both sides verify each other's X.509 certificates (Toyyib et al., 2026). This prevents Man-in-the-Middle attacks and guarantees that data arrives only from trusted devices. Payload-level encryption may also be applied, in which ECG data are encrypted on the gateway using keys available only within the secure processing environment, thereby implementing end-to-end encryption.

For practical deployment, establish a rule that requires the gateway to send each ECG fragment only after the server acknowledges receipt. Enable mutual certificate verification between the gateway and the cloud ingestion endpoint. Encrypt the medical payload before it leaves the gateway, so that even if traffic is intercepted, the record's contents remain inaccessible. For example, if the patient's home gateway transmits 10-second fragments of the cardiac signal every 10 seconds, a unique number should be assigned to each fragment, a delivery acknowledgment should be awaited, transmission should be repeated if no acknowledgment is received, and the copy should be deleted only after confirmation. The decryption key should be stored inside the protected processing environment. Such a scheme is especially useful under unstable home network conditions because the physician receives the full time series without gaps or duplicates, while the risk of leakage of sensitive data falls sharply even if the intermediate communication channel is compromised.

Tracking Logic and Adaptive Resource Management

Efficient operation of a remote monitoring system is impossible without detailed technical supervision of the device fleet. The platform must implement lifecycle monitoring mechanisms that include tracking battery charge level, device temperature, available memory, and radio signal quality. For this purpose, specialized channels for transmitting status metadata are allocated within the MQTT topic hierarchy, and these are sent by the device with a lower periodicity than the primary ECG signal.

To optimize energy consumption, an adaptive runtime manager such as ADAM may be used, which dynamically adjusts device operation parameters based on the patient's current condition and battery charge level (Antonio et al., 2021). If primary analysis algorithms at the edge fail to identify anomalies, the data transmission rate or signal granularity may be reduced temporarily. When suspicious patterns are detected, however, the device switches to a high-activity mode and ensures transmission of the uncompressed signal for detailed verification in the cloud. This approach enables wearable sensor operation to extend from a single battery charge to several weeks, thereby increasing patient compliance with long-term monitoring. Operational monitoring parameters and event-driven response mechanisms are systematized in Table 2.

Table 2. Operational Monitoring Parameters and Event-Driven Response Mechanisms

Tracking Component	Monitoring Parameter	Critical Event	System Action
Battery	Voltage, charge level, %	Level < 15%	Notify patient / doctor
Memory	Free space, MB	Usage > 90%	Forced data upload
Network Channel	Latency, packet loss	Connection loss > 1 hour	Generate technical alert
Firmware	Version, checksum	OTA update failure	Roll back to stable version
Sensor	Contact resistance	Electrode disconnection	Notify about poor contact

For practical application, define two operating modes for the device from the outset and tie them to status telemetry so the platform can adjust the load based on the current situation without physician involvement. Under stable rhythm, satisfactory charge, and reliable electrode contact, transmit enlarged signal fragments and service information every few minutes. For example, if a patient wears the sensor at home for two consecutive weeks, the system can send short status summaries every five minutes during ordinary periods. If signs of arrhythmia appear at night, it can immediately transmit a detailed recording segment, notify the patient of the need to check electrode contact, and simultaneously notify the physician that the device has entered an intensified observation mode.

CHAPTER 2. DESIGN OF A HORIZONTALLY SCALABLE BACKEND

The reception and processing of telemetry from hundreds of thousands of patients requires a server infrastructure capable of elastic scaling under varying load. Traditional architectures with synchronous component interaction often become bottlenecks when processing high-frequency ECG streams, underscoring the need to transition to asynchronous, event-oriented models (Pallaprolu, 2025).

Microservice Architecture and Layer Isolation

Backend design should be grounded in the principle of separation of responsibilities, in which the functions of data intake, analysis, storage, and visualization are distributed

across independent microservices. The intake layer consists of a group of lightweight services operating behind a load balancer, whose task is to accept data packets from the MQTT broker or API gateway as quickly as possible, verify their integrity, and place them in a distributed message queue. Isolation of this layer is critical because even if the deep analytics or report generation modules are temporarily unavailable due to high load or failures, data intake from patient devices must continue without interruption.

For containerized microservice management, it is expedient to use an orchestrator such as Kubernetes, which provides automatic horizontal scaling based on CPU load metrics or message queue depth. The deployment of a service mesh, such as Istio, enables advanced monitoring of inter-service interactions, automatic request retries, and traffic management, thereby increasing the overall resilience of the system to partial failures.

For practical deployment, divide the platform into at least four independent processing domains. Telemetry intake. Primary verification and routing. Analytical processing. Long-term storage and data delivery to the physician. That overload in one segment does not stop the entire cardiac-signal flow. The highest priority should be assigned to the intake domain, where each incoming packet is immediately checked, timestamped, and placed in a queue for further processing. The analytical and reporting modules then operate at their own pace.

For example, if 100 patients are simultaneously connected in a clinic and after a mass device update the ECG stream temporarily doubles, the system should automatically add new instances of intake and routing services, preserve continuous loading of telemetry into the queue, and temporarily slow the construction of secondary reports so that the physician continues to receive fresh signals without interruptions even during a short-term load spike.

Asynchronous Processing and Data Buses

The heart of a scalable platform is a distributed event bus such as Apache Kafka or Amazon Kinesis. ECG telemetry arrives as continuous time-series streams distributed across bus partitions based on patient identifiers. This ensures preservation of message order for each patient and allows many consumers to process data in parallel.

An event-driven model enables the implementation of the fan-out pattern. Within this approach, one incoming ECG data stream is consumed simultaneously by several independent subsystems. Such an organization of processing enables the parallel execution of several tasks related to analysis, interpretation, storage, and subsequent data preparation.

One consumer of this stream is an instant alerting service that identifies critical, life-threatening arrhythmias. At the same time, data enters the streaming analytics layer, where average heart rate and rhythm variability indicators are computed in real time. In parallel, a persistent storage service records raw signals in a long-term repository. In addition, an anonymization module is engaged to prepare data for research and the training of new machine-learning models. Data bus requirements are systematized in Table 3.

Table 3. Data Bus Requirements

Data Bus Parameter	Requirement for ECG Telemetry	Rationale
Throughput	> 100 GB/hour	Processing signals from 100k+ devices
Latency	< 100 ms	Required for real-time alerting
Delivery Guarantee	At-least-once / Exactly-once	No loss of clinical data is acceptable
Replication	Minimum across 3 availability zones	Fault tolerance in case of data center failure
Retention	From 24 hours to 7 days	Enables reprocessing in case of failures

For practical deployment, distribute the cardiac-signal stream by patient identifier so that all fragments of one observation always proceed in the same sequence and can be processed in parallel by different modules without being mixed with other patients' data. This allows launching several independent actions immediately upon the recording's arrival. One module tracks life-threatening rhythm disturbances. A second continuously calculates heart rate and rhythm variability. A third preserves the original signal in long-term storage. A fourth removes personal attributes from a copy of the recording for scientific work and training of computational models.

For this scheme to function resiliently, define strict operating rules for the data bus in advance. Preserve message order for each patient. For critical events, establish minimal processing latency. For all telemetry, enable confirmed delivery and store copies across multiple independent availability zones. In addition, specify a stream retention period of at least several days so the system can reprocess the archive in the event of an analytics module failure, an error in arrhythmia-detection rules, or the release of a new algorithm version.

For example, if the platform serves 100 wearable sensors, then when a nocturnal episode of ventricular tachycardia occurs in a particular patient, the same ECG fragment should simultaneously be entered into the urgent physician notification module, the contour for calculation of derived indicators, the archive of original signals, and the de-identified research repository. If at that moment the analytics module fails, the intake stream continues to move forward, the alert message is generated without pause, and the missed

calculations are then raised again from the preserved bus over the last 24 hours. Such an order provides the clinic with controllable scalability, predictable fault tolerance, and the ability to increase patient capacity without rebuilding the entire platform.

Data Storage and HIPAA/GDPR Compliance

The storage of protected medical information requires a multilayered security approach (Chahal & Epstein, 2025). In accordance with HIPAA technical requirements, ECG data must be encrypted both in transit and at rest. For this purpose, cloud environments use key management services that enforce strict key rotation and access controls.

The storage architecture should be hybrid. Such an approach is required for separate work with data that differ in structure, access modes, and storage reliability requirements.

Metadata and clinical records should be stored in relational databases that support replication and strong consistency. This is required for storing patient profiles, monitoring plans, and final physician conclusions. Time series represented by ECG signals should be stored in specialized time-series databases or in object storage, such as S3. This solution provides efficient storage and rapid access to billions of data points. Audit logs should be stored in a separate repository to record all ePHI access events. This repository must be immutable, in accordance with the WORM principle, that is, Write Once Read Many. This requirement ensures transparency during compliance audits.

For de-identifying data for research purposes, a hashing process using a salt is applied, allowing correlation of records for the same anonymous subject without disclosing identity. Data storage hierarchy is illustrated in Figure 1.

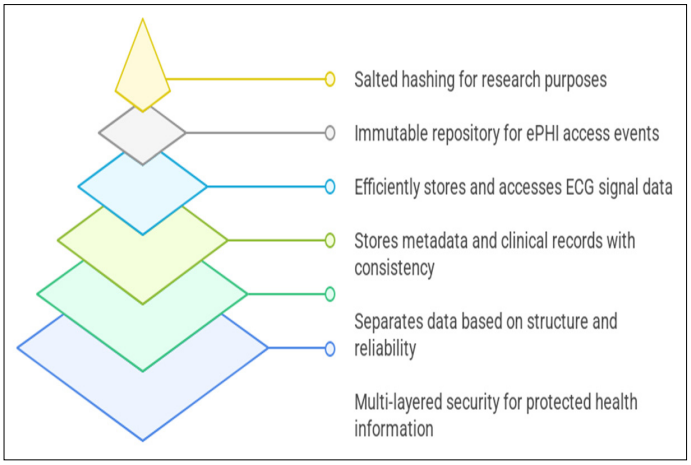

Fig. 1. Data Storage Hierarchy

For practical deployment, separate the processing stream and storage contours for different tasks from the outset, so that the incoming cardiac signal is not delayed by database writes, the preparation of conclusions, or the generation of research cohorts. After intake, route each ECG fragment to the event bus with a link to the patient identifier, then launch parallel routing in three independent directions. Send the stream to the first contour for urgent analysis and identification of dangerous rhythm disturbances. Place the original time series in the second contour for long-term storage. Transfer clinically significant metadata, patient information, observation parameters, and physician conclusions to the third contour. Such an order allows preserving the continuity of telemetry intake even when one of the downstream modules is overloaded, undergoing an update, or temporarily unavailable. At the operational level, this means a simple rule. Once the recording has been accepted and placed into the bus, the platform continues to function resiliently and does not lose clinically significant material.

Build storage with the principle of separating sensitivity and access modes, because different data require different read speeds, protection levels, and lifetimes. Place patient profiles, monitoring prescriptions, examination statuses, and final conclusions in a consistent transactional database with replication and strict role-based access controls. Preserve ECG signals themselves in a scalable time-series repository or in an object-oriented database designed for billions of samples and for fast retrieval of the required interval by time. Move access logs into a separate immutable repository in which each operation of reading, exporting, rights modification, and access to the patient chart is recorded without the possibility of concealed editing. At the same time, enable mandatory encryption during transmission and storage, configure regular key rotation, and link access issuance to the staff member's role, division, and treatment task. For the research contour, create in advance a separate de-identification route that replaces the patient identifier with a stable, salted hash so that records of the same person can be matched without disclosing identity.

For example, if a clinic launches remote monitoring for 50 patients at risk of arrhythmias, then for each incoming ECG fragment the system must, within fractions of a second, place the record into the bus, send a copy to the emergency ventricular tachycardia detection module, preserve the original signal in the time-series archive, update the observation card in the clinical database, and record in the log who obtained access to these data and when. If, one month later, the research department requests an array of nocturnal tachycardia episodes in patients older than 60 years, the platform can provide a de-identified cohort from the prepared contour without direct access to personal information. If a supervisory authority or internal control service requires verification of who opened the record of a particular patient, the organization will raise the immutable access log and demonstrate the complete chain of actions by time. This method of deployment converts the architectural scheme into an operational regulation that embeds scaling, data protection, and compliance with regulatory requirements in the system in advance, before load growth and the first incident.

CHAPTER 3. ALERTING ALGORITHMS AND CLINICAL INTEGRATION

The value of the platform lies in its intelligent interpretation, which enables it to inform the physician promptly of critical changes in the patient's condition. This requires the construction of a complex analytical pipeline that combines classical digital signal processing with advanced deep learning methods.

Real-Time Anomaly Detection Pipeline

The ECG analysis process begins at the primary processing level, where the signal passes through a cascade of filters to remove baseline wander, high-frequency noise, and power-line interference. A key stage is R-peak detection, for which an optimized Pan-Tompkins algorithm is often used, enabling precise determination of the QRS complex boundaries even in noisy signals. The calculated RR intervals serve as the basis for assessing heart rhythm variability and primary detection of tachyarrhythmias and bradyarrhythmias.

For more complex pattern classification, such as atrial fibrillation, extrasystoles, and conduction blocks, deep neural networks are used. Contemporary systems employ a two-stage architecture. Such an approach enables data processing to be distributed between user-side devices and cloud infrastructure. This provides a combination of prompt primary analysis and high-accuracy subsequent interpretation.

On the wearable device or gateway side, lightweight TinyML models are used. They perform preliminary real-time screening and identify coarse deviations from the norm. Suspicious fragments are then transmitted to the cloud, where they are analyzed by more powerful models, including CNNs, LSTMs, and Transformers. These models provide high analysis accuracy, reaching 98%, through the use of the full computational capacity of servers.

When an anomaly is confirmed, the system generates an alert that is delivered to the clinical interface immediately. In this case, the physician receives a contextual ECG fragment on which the algorithm has highlighted the detected pathology, reducing the time required for situation assessment to 15–20 seconds.

For practical deployment, arrange cardiac-signal processing as a continuous chain of rapid decisions in which each new fragment is first cleaned of typical interference, then automatically decomposed into cardiac cycles, and immediately checked for coarse deviations in interval duration and complex morphology. Such an order is required so that the system can react at the moment of the event, when the physician still has time for clinical action.

In the operational regulation, define from the outset the duration of the input window, the alert thresholds, and the rule of forwarding upward only those segments in which signs of tachycardia, bradycardia, irregular rhythm, or unusual signal morphology have been detected. This reduces computational load while preserving the high clinical value of the stream, as the records sent to the cloud contour are precisely those that require deeper interpretation. For routine operations, it is especially useful to determine in advance the volume of context associated with a suspicious episode. The physician usually requires the anomalous segment itself, along with several seconds of recording before its onset and after its termination, to quickly understand the clinical picture without repeated archive requests.

On the user side, it is expedient to run only primary screening, which can detect a suspicious rhythm change in time and immediately mark such a fragment as a priority for transmission. Transfer more complex arrhythmia recognition to the cloud computational layer, where more resource-intensive models can be applied, and the signal can be verified with accumulated observation context. In practical terms, this means a simple configuration rule. The local module is responsible for the reaction rate. The cloud module is responsible for improving classification accuracy and reducing false alerts. For the scheme to operate resiliently, add a mechanism of repeated verification. First, the local contour isolates a suspicious episode. Then the cloud contour confirms or rejects it, assigns an event class, and generates a brief explanation for the physician. Together with the notification, transmit the episode duration, average contraction rate, timestamp, degree of algorithmic confidence, and a signal fragment with the pathological section visually highlighted. This format reduces the time of initial assessment and helps the physician distinguish a truly dangerous condition from technical interference or a borderline deviation more rapidly.

For example, if a patient wears a sensor after a prior stroke and the system maintains round-the-clock observation, the local module may analyze a new segment of the recording every 10 seconds, search for irregular intervals between contractions and abrupt changes in complex shape, and then immediately send to the cloud only suspicious episodes with duration from 20 seconds together with five seconds of preceding and following signal. The cloud contour then clarifies within a few seconds whether the recording corresponds to atrial fibrillation, a single extrasystole, or a motion artifact, and then prepares a message to the physician with the context already in place. If atrial fibrillation is confirmed, the physician receives a concrete ECG segment marked with the zone of deviation, the episode onset time, the calculated heart rate, and a brief reliability assessment. As a result, the clinician spends 15–20 seconds reviewing, makes rapid decisions about subsequent actions, and works with the observation stream without being overloaded by unnecessary notifications.

Automatic Generation of Clinical Reports

Aggregation of data over a prolonged period, from 24 hours to several weeks, requires the creation of concise

and informative reports. The report generation process is automated using machine-learning models that translate quantitative indicators into qualitative descriptions. In the wave-to-report architecture, an encoder based on a one-dimensional convolutional network extracts structural features from the ECG, while a decoder based on a Transformer or BERT generates a textual conclusion that mimics the style of a practicing cardiologist.

The visual component of the report is constructed according to the principle of progressive disclosure. The upper level of the report contains summary indicators, total recording duration, mean, minimum, and maximum heart rate, and burden of various arrhythmia types. When necessary, the physician can delve into details by examining trends and specific arrhythmia episodes with the possibility of interval measurement on a digital ECG grid. Such automation enables reducing the time required to prepare the final conclusion several-fold while preserving a high level of diagnostic accuracy. The report generation workflow is shown in Figure 2.

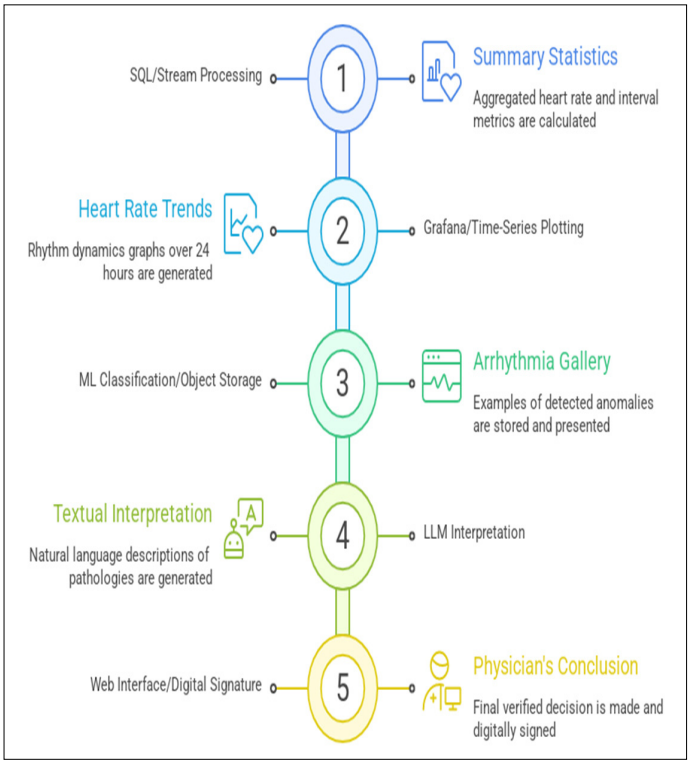

Fig. 2. Report Generation Workflow

For practical deployment, define a rigid structure for the automated report from the outset so that the physician receives the same order of information each time and can spend time on clinical assessment. At the beginning of the document, display only those parameters on the basis of which the primary decision on observation significance is made. Total recording duration. Mean, minimum, and maximum heart rate. Aggregate proportion of arrhythmia episodes. Number of clinically significant events. Signal quality over the entire monitoring period. Then add a brief textual conclusion composed according to previously approved medical templates and interpretation rules. After that, place a list of the most important episodes with exact onset time, duration, rhythm disturbance type, and a link to the corresponding segment of the cardiac signal. In high-volume settings, the physician needs to determine within several tens of seconds whether the recording requires urgent attention, a scheduled revision, or routine archiving.

For the report to be suitable for real clinical work, immediately limit the volume of automatically generated text and prohibit the system from including vague formulations in the final conclusion. Each textual inference must rest on a specific measurable attribute that can be verified in the digital recording. For this purpose, it is useful to connect each sentence of the conclusion with a numerical basis and a concrete signal fragment. If the system flags probable atrial fibrillation, the report must automatically include the episode duration, the degree of rhythm irregularity, the mean contraction rate in that window, and a link to the recording section that opens the required interval. If signal quality was reduced, this must also be brought into the upper part of the document so that the physician understands the limits of reliability of the automatic interpretation. For prolonged observations, it is useful to introduce a prioritization rule. Life-threatening and clinically significant episodes are reported first. Then stable recurring disturbances. After that, rare or doubtful events require additional verification.

For example, if a patient underwent continuous monitoring for 14 days after catheter intervention, the system must automatically assemble a unified report in which on the first page the physician immediately sees that the recording lasted 13 days 18 hours, the mean heart rate was 72 beats per minute, the minimum was 46, the maximum was 148, the aggregate burden of atrial fibrillation episodes reached 3.2 percent, and the longest episode lasted 18 minutes 40 seconds and began on April 12 at 03:14. Below, the system must present a brief conclusion on the presence of recurrent paroxysms with indication of their frequency and temporal distribution, and then attach the three most representative fragments of the cardiac signal for rapid verification. As a result, the physician receives material that is already assembled and ordered, confirms or refines the conclusions in 2–3 minutes, and then signs the final report without manually collecting indicators from a multi-day data array.

Ensuring Interoperability through HL7 FHIR

Integration of the cloud platform into the broader healthcare ecosystem is impossible without support for international data exchange standards. The HL7 FHIR standard is a generally recognized mechanism for ensuring semantic interoperability (Toi & Adochiei, 2025). Within the platform, monitoring results are represented as a set of FHIR resources. The Patient resource contains demographic data. The Observation resource serves as the primary mechanism for transmitting ECG data. For waveform representation, it uses the valueSampledData component, which efficiently

packages the measurement array, sampling frequency, and calibration coefficients.

The DiagnosticReport resource is intended for grouping monitoring results. It also contains a link to the final textual conclusion and a PDF version of the report. For seamless integration with electronic medical records, the SMART on FHIR framework is used, enabling the cloud platform interface to be embedded directly into the physician's work window within the hospital system. This ensures automatic data synchronization and eliminates the need for medical personnel to manually transfer analytical results between systems.

For practical deployment, configure the platform from the outset so that each cardiac monitoring result is automatically translated into a unified medical format and can be entered into the hospital information system without manual revision. For this purpose, determine in advance which pieces of information and in what form must be transmitted outward. Patient demographic data, observation identifier, recording start and end times, sampling parameters, the measurement array itself, the physician's clinical conclusions, and a link to the final report must be assembled into a formalized package using a single template. In a multidisciplinary clinic the same result is used by the cardiologist, the duty physician, the functional diagnostics department, and the medical documentation archive. If the exchange structure is defined in advance, data from the cloud platform is immediately incorporated into the broader clinical picture and can be used within the treatment pathway without additional copying or rewriting.

In practice, it is useful to implement a simple rule. Each completed monitoring episode must automatically generate three linked entities. The first stores patient information. The second contains the cardiac signal itself with time linkage, recording frequency, and calibration parameters. The third combines the results into a unified diagnostic document with a brief conclusion, a list of detected rhythm disturbances, and an attached final report. These data must then be transmitted to the medical record immediately after the physician confirms the result or after a predefined event, for example, upon completion of a 24-hour observation period or identification of a clinically significant arrhythmia. To avoid integration errors, the regulation should separately require verification of the match between the patient identifier, recording time, and study number before the result is sent to the hospital system. This sharply reduces the risk of erroneously linking the cardiogram to another patient's chart and simplifies subsequent audit.

For example, if a patient undergoes 72-hour monitoring after complaints of interruptions in cardiac activity, the system must automatically generate, upon completion of observation, a digital package containing the patient's personal data, the continuous cardiac-signal recording, detected episodes of supraventricular tachycardia, and the final physician's conclusion. After that, the result must immediately appear in the electronic medical record as a ready study section, where the treating physician can view the summary, open the required signal fragment, and, if necessary, download the official report for a multidisciplinary consultation or hospitalization. In such a scenario, the medical staff receives a complete and legally significant result within the familiar working window. This accelerates decision-making, reduces manual tasks, and makes the cloud platform a fully integrated component of the treatment process.

To ensure system readiness for industrial operation, architects and engineers must verify that the following requirements are met. For this purpose, a checklist is proposed, shown in Table 4.

Table 4. Architecture Layer Compliance Checklist

Architecture Layer	Validation Criterion	Relevant Standard / Recommendation
Edge, Devices	Is automatic data backfill implemented after connection restoration?	Fault tolerance recommendation
Security, Transit	Is mutual authentication, mTLS, using X.509 certificates implemented?	NIST SP 800-52
Ingestion, Intake	Can the system handle a 10× peak load via message queues?	Scaling pattern
Privacy, PHI	Is ePHI excluded from monitoring logs and crash reports?	HIPAA Technical Safeguard
Analytics, AI	Has the model been validated on an independent dataset with Sensitivity/Specificity?	Clinical validation
Integrations, EHR	Is mapping of ECG signals to FHIR resource valueSampledData supported?	HL7 FHIR Implementation Guide
Ops, Monitoring	Are alerts configured for anomalous device behavior, e.g., rapid battery drain?	IoMT Device Tracking
Legal, Compliance	Is a Business Associate Agreement, BAA, signed with all cloud providers?	HIPAA Administrative Rule

For transitioning such an architecture to industrial deployment, first convert the verification checklist into a mandatory system acceptance regulation before launch in real patients. For each layer, assign in advance a measurable readiness criterion, a responsible executor, and a verification method under conditions as close as possible to operational ones. At the device level, verify that after communication restoration, the accumulated recordings are automatically uploaded to the cloud without loss of temporal sequence. At the secure transmission level, confirm mutual authentication between the gateway and server using digital certificates. At the data-intake level, artificially create a load that exceeds the normal level by a factor of 10 and verify that the message queue preserves the stream without loss. At the confidentiality level, separately prohibit medical information from entering monitoring logs and crash reports. At the analytics level, establish the rule that any model is admitted to operation only after verification on an independent dataset using previously established thresholds of sensitivity and specificity.

For example, if a clinic is preparing the launch of remote cardiac monitoring for hundreds of patients, then before the start, it should conduct a unified acceptance week during which engineers simulate mass evening device connection, temporary home-Internet disconnection, abrupt growth in the number of alarming episodes, and attempts at erroneous transmission of data into diagnostic logs. Only such a result should be regarded as successful in which, after a failure, the system automatically uploads the entire missed signal, withstands a tenfold peak of incoming flow, generates an alert in the event of anomalous battery consumption, preserves medical information outside technical logs, and transfers the final recording into the patient's electronic record without distortion of the time series. Such an order is useful because it reveals vulnerabilities before the first clinical incident and transfers architectural requirements from general intentions to verifiable operational actions.

CONCLUSION

The presented framework forms an integrated engineering model of a cloud platform for remote cardiac monitoring, in which continuous patient observation is given architectural embodiment at all system levels. Its significance lies in its role in binding together a unified production logic for wearable ECG sensors, home Bluetooth gateways, secure transmission channels, an event-driven backend, distributed storage, analytical modules, and clinical interfaces. Such a composition establishes for the HealthTech industry a reproducible scaffold for building a platform for prolonged observation, early identification of rhythm disturbances, and work with high-frequency ECG telemetry arrays at scale, with hundreds of thousands of devices.

The distinctiveness of the development is revealed through its multilayered signal-processing architecture and the precise alignment of requirements for reliability, security, and scalability. At the edge, the framework establishes the principles of energy-efficient acquisition, automatic pairing, local buffering, and adaptive device resource management. At the transport level, it relies on MQTT, end-to-end encryption, mutual authentication, and guaranteed delivery of ECG fragments. Within the server contour, it constructs a microservice and event-driven architecture in which data intake, analytics, archiving, anonymization, and report generation are independent streams, preserving the clinical integrity of the time series. Through such a structure, the framework establishes an industrial standard for systems in which growth in the number of patients and devices does not undermine platform resilience or diminish the value of the incoming signal to the physician.

A substantial contribution of the study is that the architectural scaffold has been brought to the level of a clinically significant interpretation pipeline. Within it, primary filtering, R-peak detection, RR interval calculation, two-stage anomaly analysis, and automatic report generation form a continuous chain from raw signal to medical conclusion. The framework shows how the ECG stream is converted into a contextual alert, a verifiable arrhythmia episode, a structured report, and a set of standardized entities for integration into hospital systems through HL7 FHIR and SMART on FHIR. For the industry, this means the emergence of an applied model in which computational architecture ceases to be a background infrastructure layer and becomes part of the treatment process, influencing the speed of data routing, the quality of physician assessment, and the completeness of the digital trace of observation.

Thus, the proposed framework should be regarded as a significant industry development, shifting remote cardiac monitoring from local pilot solutions to a scalable, regulated cloud ecosystem. Its strength lies in the combination of technical feasibility, operational verifiability, and compliance with HIPAA, GDPR, and contemporary clinical integration requirements. Within the article, it functions as a methodological foundation for the design of next-generation platforms in which fault tolerance, controllable scaling, ePHI protection, algorithmic ECG interpretation, and incorporation of results into the electronic medical record are treated as elements of a unified system. It is this connectivity that makes the framework significant for the industry and sets the direction for implementing cloud solutions in the practice of continuous cardiological observation.

REFERENCES

1. Ansari, M. Y., Qaraqe, M., Righetti, R., Serpedin, E., & Qaraqe, K. (2024). Enhancing ECG-based heart age: impact of acquisition parameters and generalization strategies for varying signal morphologies and corruptions. *Frontiers in Cardiovascular Medicine*, *11*. https://doi.org/10.3389/fcvm.2024.1424585

2. Antonio, S. M., Loi, D., Raffo, L., & Meloni, P. (2021). An adaptive cognitive sensor node for ECG monitoring in the Internet of Medical Things. *ArXiv*. https://doi.org/10.48550/arXiv.2106.06498

3. Boriani, G., Brachmann, J., Lewalter, T., Wright, D. J., Badertscher, P., Gale, C. P., Merino, J. L., Pürerfellner, H., & Lip, G. Y. H. (2025). Access and reimbursement of ambulatory cardiac monitoring across Europe. *European Heart Journal - Digital Health*, *6*, 1282–1292. https://doi.org/10.1093/ehjdh/ztaf102

4. Chahal, S., & Epstein, H. (2025). HIPAA security rule updates & IAM compliance recommendations. *IDPro Body of Knowledge*, *1*(16). https://doi.org/10.55621/idpro.120

5. Dores, H., Santos, J. F., Gil, V., & de Araújo Gonçalves, P. (2026). Cardiovascular Prevention: Current Gaps and Future Directions. *Diagnostics*, *16*(1), 16. https://doi.org/10.3390/diagnostics16010016

6. Hizem, M., Bousbia, L., Ben Dhiab, Y., Aoueileyine, M. O.-E., & Bouallegue, R. (2025). Reliable ECG Anomaly Detection on Edge Devices for Internet of Medical Things Applications. *Sensors*, *25*(8), 2496. https://doi.org/10.3390/s25082496

7. Hughes, A. M., Taylor, D. J., Morris, P. D., & Brittain, E. L. (2026). Wearable devices and cardiovascular health: revolutionizing remote monitoring and disease prevention. *European Heart Journal*, ehag189. https://doi.org/10.1093/eurheartj/ehag189

8. Kim, H. A., Lee, H., Park, H.-S., Ahn, J., Lee, S.-M., Choi, S.-Y., Oh, E. H., Choi, J.-H., Park, J.-Y., & Choi, K.-D. (2025). Wearable ECG patch monitoring for 72 h is comparable to conventional Holter monitoring for 24 h to detect cardiogenic vertigo. *Scientific Reports*, *15*, 7744. https://doi.org/10.1038/s41598-025-92472-0

9. Nandini, A. (2025). Advancements in secure data architectures for remote patient monitoring. *World Journal of Advanced Research and Reviews*, *26*(1), 3262–3274. https://doi.org/10.30574/wjarr.2025.26.1.1392

10. Pallaprolu, S. (2025). Event-driven architecture: A modern paradigm for real-time responsive systems. *World Journal of Advanced Engineering Technology and Sciences*, *15*(2), 2860–2867. https://doi.org/10.30574/wjaets.2025.15.2.0826

11. Saha, N., Paul, P., Ji, K., & Harik, R. (2024). Performance evaluation framework of MQTT client libraries for IoT applications in manufacturing. *Manufacturing Letters*, *41*, 1237–1245. https://doi.org/10.1016/j.mfglet.2024.09.150

12. Schärer, N., Polonelli, T., & Magno, M. (2025). Pushing Wi-Fi HaLow to the Extreme Edge: A Performance Study on a Low-Power IoT Node. *Proceedings of 2025 10th International Workshop on Advances in Sensors and Interfaces (IWASI)*, 1–6. https://doi.org/10.1109/iwasi66786.2025.11121933

13. Toi, F.-A., & Adochiei, I.-R. (2025). HL7 FHIR-Based Open-Source Framework for Real-Time Biomedical Signal Acquisition and IoMT Interoperability. *Applied Sciences*, *15*(23), 12803. https://doi.org/10.3390/app152312803

14. Toyyib, T. O., Olose, S. A., & Kamil, S. K. (2026). Assessing the Significance of Cryptographic Techniques in Enhancing Personal Data Security in Modern Information Systems. *Zenodo*, *2*(1), 152–169. https://doi.org/10.5281/zenodo.18904491

15. Verma, N., Singh, S., & Prasad, D. (2021). A Review on existing IoT Architecture and Communication Protocols used in Healthcare Monitoring System. *Journal of the Institution of Engineers (India): Series B*, *103*(1), 245–257. https://doi.org/10.1007/s40031-021-00632-3