Cloud-Native Real-Time Health Monitoring with Bio-Robotic Sensors and MERN-Based Edge Analytics

Uma Ajay Kumar Reddy PS

M. S. Ramaiah University of Applied Sciences, Bangalore, India
Corresponding Author Email: *umaveena.alvas[at]gmail.com*

Abstract: ***Continuous home health monitoring is increasingly feasible, yet most bio-robotic devices remain hardware prototypes with weak longitudinal data infrastructure. This paper proposes a cloud-native architecture that integrates bio-robotic sanitary sensors, edge gateways, and a MERN-based multi-tenant backend for real-time biochemical screening. Non-invasive sensors embedded in household fixtures capture urine, stool, and vital-sign signals under robotic alignment and cleaning control. MQTT and WebSocket transport deliver telemetry to Node.js and Express microservices, while MongoDB stores schema-flexible biometric events and Flask-based analytics services execute Random Forest and long short-term memory models for anomaly detection and trend forecasting. Unlike device-silo designs, the proposed stack treats robotic sensing, data engineering, and clinician-facing visualization as one end-to-end system. The paper discusses the design through a smart toilet case study and defines a literature-grounded operating target for early chronic-risk screening in home-centric care.***

Keywords: Bio-robotic sensors; edge computing; MERN architecture; smart toilet; preventive health monitoring

1. Introduction

Home-centric medicine is moving from episodic measurement to continuous observation. Reviews of wearable and diagnostic-support devices show that modern sensors can capture physiological signals in real time and extend telemedicine beyond the hospital boundary [1]. However, the engineering challenge is no longer only the sensor. For senior system designers, the deeper bottleneck is the data pipeline: how to ingest, organize, analyze, and surface longitudinal bio-signals without turning every device into an isolated prototype.

This problem is especially visible in bio-robotic fixtures. A smart toilet, for example, can automatically analyze urine using colorimetric strips, estimate flow characteristics, classify stool, and securely store outputs in the cloud [2]. That work demonstrates that automated excreta-based monitoring is technically credible. Yet the underlying question remains open for deployment at scale: how should thousands of such devices stream multi-modal telemetry into a production-grade software platform that supports clinician dashboards, historical trending, and model-driven alerts?

Edge-enabled healthcare frameworks already suggest why cloud-only thinking is insufficient. Secure edge architectures have been shown to reduce response time and strengthen controlled access to sensitive health data [3]. At the analytics layer, recent studies report strong performance from Random Forest and LSTM models across wearable and health-monitoring tasks, with published results frequently reaching the high-80% to high-90% range depending on the dataset and target variable [4]-[6]. These results indicate that predictive screening is feasible, but only when sensor capture, event transport, storage, and inference are engineered as one coherent stack.

Accordingly, this paper proposes a full-stack bio-robotic architecture that merges household sampling hardware with SaaS-style cloud software. The core idea is simple: treat robotic sensing as a multi-tenant, event-driven platform problem. The contribution is not merely a device concept, but an implementation-oriented reference design that links edge robotics, Node.js microservices, MongoDB persistence, Python-based machine learning, and a doctor-facing web dashboard into a single real-time monitoring workflow.

2. Literature Survey

The recent literature converges on three important trends. First, the monitoring interface is moving closer to the user. Wearable-device reviews show that non-invasive sensing, wireless communication, and remote supervision have become central to modern diagnostic support [1]. This shift encourages systems that collect data continuously rather than during occasional clinic visits.

Second, sanitary and urine-based platforms are emerging as serious diagnostic endpoints. Park et al. presented an autonomous smart toilet capable of urinalysis, uroflowmetry, stool analysis, and encrypted cloud storage [2]. Bhatia et al. later proposed a layered IoT framework for home-based urine infection prediction [7], and Alqahtani et al. extended this direction using IoT-fog architecture and machine-learning models for smart toilets [8]. These studies show that sample acquisition inside everyday fixtures is practical and clinically relevant.

Third, the intelligence layer is maturing. Edge-health research highlights the value of near-source processing for time-sensitive decisions [3]. In parallel, machine-learning studies on wearable and health data report that Random Forest remains a strong baseline for structured physiological data, while LSTM variants improve temporal anomaly detection and sequential forecasting [4]-[6]. The published evidence supports using a hybrid model stack rather than a single algorithm for all tasks.

Despite this progress, an integration gap remains. Existing studies typically emphasize either the sensor apparatus, the prediction model, or the fog-computing layer. Fewer works describe a production-style software architecture with tenant isolation, schema-flexible storage, model versioning, clinician-facing observability, and a web application layer suitable for deployment by hospitals or remote-care providers. This paper addresses that integration gap.

3. Problem Definition

Most bio-robotic health devices are still engineered as vertical silos. The sensor firmware, local processing logic, storage format, and user interface are tightly coupled, which creates four major limitations. First, longitudinal analysis becomes difficult because data schemas change as new biomarkers are introduced. Second, clinician access is weak because outputs stay on the device or inside ad hoc mobile applications. Third, scaling to many households becomes operationally complex when there is no multi-tenant identity, alerting, and audit model. Fourth, algorithm deployment is fragile when inference code is embedded directly into device logic rather than exposed as a managed service.

The system problem can therefore be stated as follows: design a cloud-native architecture for bio-robotic health fixtures that supports low-latency telemetry ingestion, flexible storage of heterogeneous biomarker events, remotely maintainable analytics, and real-time clinical visibility without compromising household usability. In practical terms, the architecture must satisfy five objectives: reliable sample capture, lightweight data transport, multi-tenant backend services, hybrid edge-cloud inference, and actionable visualization for medical personnel.

4. Methodology / Approach

A. Architecture Overview

The proposed system is organized as a layered pipeline from robotic sample acquisition to clinician review. Figure 1 summarizes the design. A household bio-robotic fixture collects signals, an edge controller packages events, a publish-subscribe transport channel carries them to cloud services, and a web dashboard exposes both real-time and historical outputs. This layering keeps the device lightweight while preserving central observability and update control.

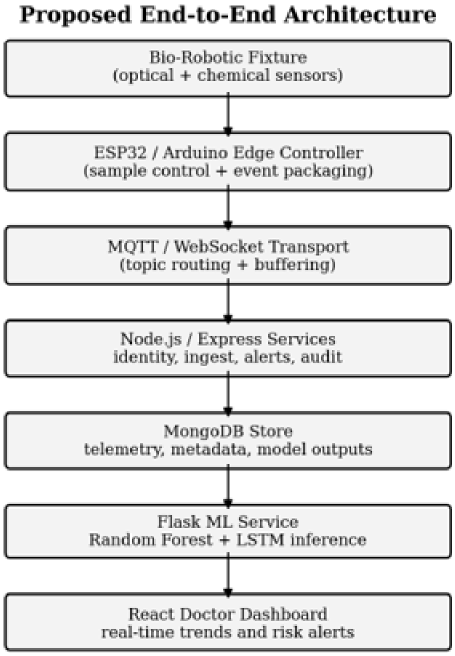

Figure 1: Proposed cloud-native bio-robotic health monitoring stack

B. Edge Layer and Robotic Sampling

At the fixture level, an ESP32 or Arduino-class controller coordinates optical, conductivity, pH, and auxiliary biosensors. A compact robotic assembly aligns the sensing cartridge with the biological sample, triggers illumination or strip reading, and executes a cleaning cycle after measurement. Edge preprocessing performs signal normalization, quality checks, and packaging of a JSON event containing deviceId, tenantId, sampleType, analyte vector, timestamp, confidence score, and firmware version. This structure supports device evolution without forcing rigid relational schemas.

C. Transport Layer

MQTT is used for lightweight event publication from the device, while WebSockets support bidirectional communication for downstream dashboards and control acknowledgments. Topic names are partitioned by tenant and device to simplify routing and isolation. An edge gateway can buffer bursts or temporary connectivity loss so that household devices are not blocked by brief cloud interruptions. The transport design favors low overhead, event ordering, and near-real-time delivery.

D. Cloud-Native Backend

The backend follows a MERN-oriented architecture. Node.js and Express microservices handle device registration, authentication, ingestion, alert rules, clinician access, and audit logging. MongoDB stores event documents, device metadata, model outputs, and consent records. A document database is a strong fit because biomarker payloads may vary across toilet, sink, mirror, or wearable fixtures. Services are containerized and orchestrated so they can scale horizontally as the number of active devices grows. The dashboard layer, built in React, subscribes to live updates, renders patient trends, and supports doctor-level filtering by analyte, patient, or risk status.

E. Analytics Layer

Predictive analysis is separated from the ingestion path through a Flask-based microservice. Structured tabular features are routed to a Random Forest classifier for event-level screening such as dehydration, abnormal urine chemistry, or infection-risk flags. Sequential biometric windows are routed to an LSTM model for trend forecasting and early chronic-risk scoring. Separating inference into an independent service allows model versioning, rollback, and regulated validation without rewriting device firmware. The architecture therefore combines low-latency event handling with maintainable AI operations.

5. Results and Discussion

A. Smart Health Interface Case Study

To illustrate the workflow, consider a smart toilet designed for routine urine and vital-sign monitoring. When the user initiates a flush or sits on the fixture, the system activates presence sensing, aligns the strip reader, collects optical and chemical signals, and transmits an event bundle to the cloud. The clinician dashboard immediately updates the patient timeline with the latest urinalysis indicators, a short-term risk score, and a trend view of deviations from baseline. The user does not need to visit a clinic for routine observation, while the physician receives structured, time-stamped evidence rather than isolated readings.

B. Feasibility Discussion

The proposed architecture is supported by prior technical evidence. Autonomous toilet-mounted health capture has already been demonstrated with cloud-linked analysis [2]. Edge-enabled healthcare frameworks have shown that a near-source layer is valuable for time-sensitive response and controlled data access [3]. Machine-learning studies further show that Random Forest and LSTM-family methods can deliver strong performance on health-monitoring tasks [4]-[6]. For that reason, this paper treats 92% chronic-risk screening accuracy as a realistic future validation target for the analytics layer rather than an already completed clinical result. That target is deliberately conservative relative to the upper end of published performance but high enough to justify deployment-oriented design choices.

C. Architectural Implications

From a systems perspective, the design offers three advantages. First, it decouples robotic innovation from software evolution; a new sensor cartridge can be added without redesigning the full backend. Second, it creates a unified longitudinal record so that single events become clinically meaningful trends. Third, it enables a SaaS operating model in which one provider can manage many household fixtures, apply model updates centrally, and expose role-based dashboards to clinicians. The main limitations are data governance, household acceptance, sensor calibration drift, and the need for rigorous clinical validation before diagnostic use. Therefore, the architecture is best understood as an implementation-ready research framework for preventive monitoring, not as a substitute for regulated medical diagnosis.

6. Conclusion

This paper argued that the central challenge in bio-robotic health interception is not merely the robotic sampler, but the software system that makes its data clinically useful. A cloud-native, multi-tenant architecture built on edge controllers, MQTT and WebSocket transport, Node.js microservices, MongoDB storage, Flask-based inference, and a React dashboard provides a practical answer to that challenge. By combining household robotics with scalable web technologies, the proposed framework moves preventive medicine toward continuous, home-centric screening. This manuscript presents a systems architecture and literature-grounded case study; it does not report a human- or animal-subject clinical trial.

7. Future Scope

Future work should focus on four directions. First, the sensor portfolio can be expanded to saliva, sweat, and breath biomarkers so that the architecture supports more than excreta-based analysis. Second, federated or split-learning strategies can be added to reduce raw-data exposure while still improving model quality over time. Third, clinical deployment studies should validate user acceptance, calibration stability, and medically meaningful threshold selection across diverse populations. Fourth, interoperability with hospital information systems and standards-based health records would make the platform easier to integrate into real care pathways. In short, the convergence of full-stack web engineering and bio-robotics remains a promising frontier for preventive digital health.

References

[1] E. Escobar-Linero, L. Munoz-Saavedra, F. Luna-Perejon, J. Sevillano, and M. L. Dominguez-Morales, "Wearable Health Devices for Diagnosis Support: Evolution and Future Tendencies," Sensors, vol. 23, no. 3, Art. no. 1678, 2023, doi: 10.3390/s23031678.

[2] S.-m. Park et al., "A mountable toilet system for personalized health monitoring via the analysis of excreta," Nature Biomedical Engineering, vol. 4, pp. 624-635, 2020, doi: 10.1038/s41551-020-0534-9.

[3] A. Singh and K. Chatterjee, "Edge computing based secure health monitoring framework for electronic healthcare system," Cluster Computing, vol. 26, no. 2, pp. 1205-1220, 2023, doi: 10.1007/s10586-022-03717-w.

[4] A. Yusuf, T. Al Jaber, and N. Gordon, "Comprehensive Health Tracking Through Machine Learning and Wearable Technology," Journal of Data Science and Intelligent Systems, vol. 4, no. 1, pp. 64-75, 2026, doi: 10.47852/bonviewJDSIS52023588.

[5] Y. Rimal et al., "Comparative analysis of heart disease prediction using logistic regression, SVM, KNN, and random forest with cross-validation for improved accuracy," Scientific Reports, vol. 15, Art. no. 13444, 2025, doi: 10.1038/s41598-025-93675-1.

[6] D. Pavithra et al., "Application of LSTM Networks for Continuous Patient Monitoring and Anomaly Detection in Wearable Health Devices," Indian Journal of Science

and Technology, vol. 17, no. 37, pp. 3909-3921, 2024, doi: 10.17485/IJST/v17i37.2600.

[7] M. Bhatia, S. Kaur, and S. K. Sood, "IoT-inspired smart home based urine infection prediction," Journal of Ambient Intelligence and Humanized Computing, vol. 14, no. 5, pp. 5249-5263, 2023, doi: 10.1007/s12652-020-01952-w.

[8] A. Alqahtani, S. Alsubai, A. Binbusayyis, M. Sha, A. Gumaei, and Y.-D. Zhang, "Prediction of Urinary Tract Infection in IoT-Fog Environment for Smart Toilets Using Modified Attention-Based ANN and Machine Learning Algorithms," Applied Sciences, vol. 13, no. 10, Art. no. 5860, 2023, doi: 10.3390/app13105860.