

IoT-Based Real-Time Patient Health Monitoring System Using Raspberry Pi and Cloud Integration

Dhanya sri S¹, Barath kanna G², Dr. R. Karthik³

^{1,2}Department of Electronics and Communication Engineering

³Assistant Professor, Department of Electronics and Communication Engineering

Abstract- Continuous monitoring of patient health parameters is a critical requirement in modern healthcare systems, particularly for elderly patients, individuals with chronic conditions, and post-operative care scenarios. Traditional hospital-based monitoring systems are expensive, stationary, and impractical for home-based care environments. This project proposes an IoT-Based Real-Time Patient Health Monitoring System that uses a Raspberry Pi microcomputer interfaced with multiple biometric sensors to continuously collect and transmit vital health parameters including body temperature, heart rate, blood oxygen saturation level, and blood pressure readings. The collected data is transmitted wirelessly to a cloud platform where it is stored, processed, and made accessible through a web-based dashboard. Automated alert mechanisms notify healthcare providers and designated caregivers when recorded values fall outside predefined safe thresholds. The system is designed to be compact, affordable, and easy to operate without requiring technical expertise from patients or caregivers. Testing results confirm that the system achieves high sensor accuracy and reliable data transmission under real-world operating conditions. This research demonstrates how Internet of Things technology can be integrated with cloud computing to deliver an accessible and effective remote patient monitoring solution that reduces healthcare costs and improves patient outcomes.

Keywords: Internet of Things, Patient Health Monitoring, Raspberry Pi, Cloud Computing, Biometric Sensors, Real-Time Data Transmission, Remote Healthcare, Automated Alerts, Wearable Technology.

I. INTRODUCTION

The global healthcare system is facing increasing pressure due to a growing aging population, a rising prevalence of chronic diseases, and limited availability of hospital infrastructure in rural and underserved regions. Cardiovascular diseases, diabetes, respiratory disorders, and hypertension are among the leading causes of mortality worldwide, and many of these conditions require continuous monitoring of physiological parameters to enable timely medical intervention.

Conventional patient monitoring is predominantly conducted within clinical environments where dedicated medical equipment is used to measure and record vital signs. While this approach provides reliable readings under controlled conditions, it places a significant burden on both patients and healthcare facilities. Patients with chronic conditions must make frequent hospital visits, and medical staff must dedicate considerable time to manual observation and documentation of health data.

The emergence of the Internet of Things has opened new possibilities for delivering healthcare services beyond the physical boundaries of hospitals and clinics. IoT refers to a network of interconnected physical devices embedded with sensors, communication modules, and computing capabilities that allow them to collect and exchange data with minimal human intervention. When applied to healthcare, IoT-enabled devices can continuously measure patient vitals and transmit the data to remote monitoring platforms accessible by medical professionals.

This project proposes an IoT-Based Real-Time Patient Health Monitoring System that integrates multiple biometric sensors with a Raspberry Pi microcomputer to create a compact and portable health monitoring device. The system measures body temperature using a digital temperature sensor, heart rate and blood oxygen saturation using a pulse oximetry sensor, and blood pressure using an integrated pressure sensor module. Collected data is transmitted to a cloud platform over a WiFi connection, where it is stored in a structured

database and displayed on a real-time web dashboard.

The system includes an automated alerting mechanism that monitors sensor readings against configurable threshold values. When any parameter exceeds a safe range, the system immediately sends notifications to registered healthcare providers and caregivers through email or SMS alerts, enabling prompt medical response.

The primary goal of this research is to develop a low-cost, reliable, and user-friendly remote health monitoring system that can be deployed in home environments to improve patient safety and reduce dependence on in-person clinical visits.

II. LITERATURE REVIEW

Remote patient monitoring has been a significant area of research in medical informatics and embedded systems engineering for several decades. Early remote monitoring systems were largely proprietary, expensive, and required dedicated communication infrastructure, limiting their accessibility to large hospital networks and specialized medical facilities.

The widespread adoption of low-cost microcontrollers, wireless communication protocols, and affordable biometric sensors has enabled researchers to develop more accessible monitoring solutions. Platforms such as Arduino and Raspberry Pi have been widely used in academic and prototyping contexts to build health monitoring systems capable of measuring a range of physiological parameters including temperature, heart rate, electrocardiogram signals, and glucose levels.

Cloud computing has become an essential component of modern IoT health monitoring architectures. Cloud platforms such as Amazon Web Services, Google Cloud, and ThingSpeak provide scalable infrastructure for storing large volumes of sensor data and delivering it to web and mobile interfaces with low latency. Several studies have demonstrated the feasibility of integrating

Raspberry Pi systems with cloud platforms for real-time health data visualization and storage.

Research into automated alert systems for patient monitoring has highlighted the importance of configurable threshold-based notification mechanisms. Studies have shown that timely alerts significantly improve outcomes for patients experiencing sudden changes in vital parameters. Machine learning models have also been explored for anomaly detection in health data streams to reduce false alert rates and improve notification reliability.

Wearable health monitoring devices have gained considerable commercial attention, with products such as smartwatches and fitness trackers demonstrating consumer demand for continuous health tracking. Academic research has sought to extend these capabilities to medical-grade monitoring through the integration of clinically validated sensors and secure data transmission protocols.

The proposed system builds upon these research foundations by combining a multi-sensor hardware platform with a cloud-integrated software architecture and an intuitive web dashboard, creating a complete end-to-end health monitoring solution suitable for home-based patient care.

III. SYSTEM ARCHITECTURE

The IoT-Based Patient Health Monitoring System is organized into four primary layers that handle data acquisition, processing, transmission, and visualization respectively. This layered architecture ensures clear separation of responsibilities and facilitates independent development and maintenance of each system component.

Sensor and Hardware Layer

The hardware layer consists of a Raspberry Pi 4 Model B microcomputer connected to three biometric sensors. A DS18B20 digital temperature sensor is used for body temperature measurement. A MAX30102 pulse oximetry module measures heart rate in beats per minute and blood oxygen

saturation percentage. A pressure sensor module captures systolic and diastolic blood pressure readings. All sensors are interfaced with the Raspberry Pi through GPIO pins and I2C communication protocols.

Data Processing Layer

The data processing layer runs on the Raspberry Pi and is implemented using Python scripts. This layer reads raw sensor values at configurable intervals, applies signal filtering algorithms to remove noise and artifacts, and converts raw ADC readings into clinically meaningful units. The processed values are validated against acceptable measurement ranges before being prepared for transmission.

Cloud Communication Layer

Processed health data is transmitted to a cloud platform over WiFi using the MQTT messaging protocol, which is optimized for low-bandwidth IoT communication. The cloud backend, hosted on ThingSpeak, receives incoming data streams, stores readings in time-series format, and exposes data through REST API endpoints for consumption by the web dashboard and alert engine.

Dashboard and Alert Layer

The web-based dashboard displays real-time sensor readings using graphical charts and numerical indicators. Healthcare providers and caregivers can access the dashboard from any internet-connected device. The alert engine continuously evaluates incoming readings against predefined threshold values and triggers email and SMS notifications when anomalous values are detected.

IV. SYSTEM IMPLEMENTATION

The hardware assembly was completed by connecting the biometric sensors to the Raspberry Pi GPIO interface using jumper cables and a breadboard. The DS18B20 temperature sensor operates on the one-wire protocol and requires a single data GPIO pin. The MAX30102 sensor communicates over the I2C bus using the SDA and SCL GPIO lines. Sensor connections were validated using multimeter continuity testing before software development began.

The Python-based data acquisition script initializes each sensor driver library and enters a continuous sampling loop. Temperature readings are collected every five seconds while heart rate and SpO2 values are averaged over a ten-second measurement window to improve accuracy. The blood pressure module is polled every thirty seconds to allow sufficient recovery time between measurements.

Signal processing routines apply a moving average filter to heart rate readings to smooth out transient noise caused by patient movement. Temperature readings are compensated for ambient temperature variation using a calibration offset determined during initial device setup. Measurement outliers beyond clinically plausible ranges are discarded and flagged in the system log.

MQTT communication is implemented using the Paho MQTT Python library. The device authenticates with the cloud broker using unique device credentials and publishes health readings to dedicated topic channels for each sensor parameter. The ThingSpeak cloud platform subscribes to these channels and archives incoming data with automatic timestamping.

The alert engine is implemented as a serverless function on the cloud platform that is triggered on each new data record. The function retrieves the latest readings, compares them against threshold values stored in a configuration database, and dispatches email notifications using the SendGrid API and SMS alerts using the Twilio API when threshold violations are detected.

The web dashboard is developed using HTML, CSS, and JavaScript with Chart.js for real-time data visualization. Dashboard charts refresh automatically every ten seconds by polling the ThingSpeak REST API, ensuring that healthcare providers always view current patient data without manually refreshing the page.

V. RESULTS AND DISCUSSION

The completed system was evaluated through a two-week testing period involving five adult participants

of varying age groups. Sensor readings obtained from the IoT system were compared against measurements taken simultaneously using certified medical instruments to assess accuracy.

Body temperature readings from the DS18B20 sensor showed a mean absolute error of 0.2 degrees Celsius compared to a calibrated clinical thermometer, which falls within acceptable clinical tolerance. Heart rate measurements demonstrated a mean absolute error of 2.1 beats per minute when compared against a certified pulse oximeter device, confirming reliable heart rate tracking during rest and mild activity.

Blood oxygen saturation readings achieved a mean absolute error of 1.4 percent compared to the reference device, meeting the accuracy requirements for non-invasive monitoring applications. Blood pressure measurements showed slightly higher variance due to sensitivity to sensor placement and patient movement, indicating that this parameter benefits from standardized measurement protocols. Data transmission latency from the Raspberry Pi to the cloud dashboard was measured at an average of 1.8 seconds under a standard home WiFi connection, confirming that the system delivers near-real-time monitoring capability. Alert notifications were delivered within 4 seconds of a threshold violation being detected, demonstrating the reliability of the automated alert mechanism.

All five participants reported that the device was comfortable to use and that the web dashboard was easy to interpret. Caregivers confirmed that the alert notifications provided timely and actionable information about patient health events, enhancing their confidence in managing home-based care scenarios.

VI. CONCLUSION

The IoT-Based Real-Time Patient Health Monitoring System successfully demonstrates how affordable embedded hardware, cloud computing infrastructure, and wireless communication protocols can be combined to deliver effective remote healthcare monitoring. The system provides

continuous measurement of critical vital signs, secure cloud-based data storage, intuitive real-time visualization, and reliable automated alerting in a compact and cost-effective platform.

The experimental results confirm that the system achieves clinically acceptable sensor accuracy for temperature, heart rate, and blood oxygen saturation measurements. The low data transmission latency and fast alert delivery times demonstrate that the system is capable of supporting time-sensitive medical monitoring scenarios.

Future work will focus on expanding the range of monitored parameters to include electrocardiogram signals and blood glucose levels. The integration of machine learning models for anomaly detection and predictive health analytics represents a promising direction for improving alert precision and enabling early identification of developing health conditions. A dedicated mobile application with push notification support will also be developed to improve accessibility for caregivers and patients. Long-term clinical validation studies with larger participant groups will be conducted to further establish the reliability of the system in real-world healthcare deployment contexts.

REFERENCES

1. L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A Survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
2. S. Amendola, R. Lodato, S. Manzari, C. Occhiuzzi, and G. Marrocco, "RFID Technology for IoT-Based Personal Healthcare in Smart Spaces," *IEEE Internet of Things Journal*, vol. 1, no. 2, pp. 144–152, 2014.
3. A. Dohr, R. Modre-Osprian, M. Drobits, D. Hayn, and G. Schreier, "The Internet of Things for Ambient Assisted Living," *International Conference on Information Technology*, pp. 804–809, 2010.
4. Raspberry Pi Foundation, *Raspberry Pi Documentation*, Available: <https://www.raspberrypi.org/documentation>

5. Maxim Integrated, MAX30102 High-Sensitivity Pulse Oximeter and Heart-Rate Sensor for Wearable Health, Datasheet, 2018.
6. Dallas Semiconductor, DS18B20 Programmable Resolution 1-Wire Digital Thermometer, Datasheet, 2019.
7. MathWorks, ThingSpeak IoT Analytics Platform Documentation, Available: <https://thingspeak.com/docs>
8. Eclipse Foundation, Paho MQTT Python Client Documentation, Available: <https://www.eclipse.org/paho/>
9. Python Software Foundation, Python Documentation, Available: <https://docs.python.org>
10. Twilio Inc., Twilio SMS API Documentation, Available: <https://www.twilio.com/docs/sms>
11. A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," IEEE Communications Surveys and Tutorials, vol. 17, no. 4, pp. 2347–2376, 2015.
12. S. Kakria, N. K. Tripathi, and P. Kitipawang, "A Real-Time Health Monitoring System for Remote Cardiac Patients Using Smartphone and Wearable Sensors," International Journal of Telemedicine and Applications, vol. 2015, pp. 1–11, 2015.
13. D. Malan, T. Fulford-Jones, M. Welsh, and S. Moulton, "CodeBlue: An Ad Hoc Sensor Network Infrastructure for Emergency Medical Care," International Workshop on Wearable and Implantable Body Sensor Networks, 2004.
14. N. Kulkarni and B. U. Salankar, "Heart Rate Monitoring System Using IR Fingertip Sensor," International Journal of Advanced Research in Electronics and Communication Engineering, vol. 3, no. 1, pp. 12–15, 2014.