Development and Evaluation of An IoT-Based Real-Time Heart Rate Monitoring System Using Android Application

Achmad Dimyati [1], Inayatul Inayah [*2], Lis Sa'odah [3] M. Yusuf [4]

[1, 2, 3, 4] Institut Teknologi dan Sains Nahdlatul Ulama Pekalongan, Pekalongan, Indonesia
[1] achmad1dimyati@gmail.com; [2] inayatul.inayah94@gmail.com*; [3] salissa35ll@gmail.com; [4] uucuf18@gmail.com
* corresponding author

Article Info

Article history:
Received September 12, 2025
Revised November 12, 2025
Accepted December 1, 2025
Available December, 2025

Keywords:
IoT-based heart monitoring; real-time data acquisition; cloud-integrated health system, Android application; pulse sensor accuracy; Firebase database

Abstract

Real-time heart rate monitoring plays an important role in early detection of cardiovascular anomalies. However, many existing IoT-based solutions still lack seamless integration of real-time sensing, cloud synchronization, and mobile visualization. This study develops and evaluates an IoT-based heart rate monitoring system using a pulse sensor, Arduino Uno, NodeMCU ESP8266, Firebase real-time database, and an Android application. Experimental testing involving six participants and 60 measurement samples demonstrates that the system achieves a high accuracy of 98.63% compared to a standard fingertip oximeter, with an average error rate of 1.37%, indicating reliable and stable performance. The system supports continuous data acquisition, real-time cloud storage, and mobile access, enabling users to view live and historical heart rate trends through the Android interface. The main contributions of this research include an end-to-end integrated IoT architecture for continuous monitoring, empirical validation of accuracy using a medical-grade comparator, and the provision of time-stamped cloud-based heart rate data for remote health monitoring. These findings confirm the feasibility and significance of IoT–mobile integration as an accessible solution for continuous cardiovascular monitoring.

INTRODUCTION

Continuous heart rate monitoring is crucial for maintaining cardiovascular health and the early detection of anomalies [1]–[3]. The normal human heart rate ranges between 60 and 100 beats per minute. Generally, a low resting heart rate indicates more efficient cardiac function, and this is why there is an ever-increasing demand for automated devices to measure heart rate and display health-related information [4]. Traditional methods of heart rate monitoring have typically focused on periodic manual measurement or clinical visits, which may be somewhat inconvenient to handle and lack the real-time aspect critical to real health management [5], [6]. The recent development of IoT technology provides a potential solution to these issues by offering the possibility of real-time, integrated, and user-friendly health monitoring systems [7]–[9]. Indeed, through IoT technology, there can be continuous and remotely accessible physiological data collection and visualization, significantly improving personal health management and remote patient care [10]–[12].

Various studies have been conducted related to IoT applications for heart rate monitoring, and the progress is encouraging though still hampered by a few drawbacks. For instance, the IoT based system in [4] sensed and broadcast heart rate data via Bluetooth; however, it did not record data history or log time-stamped measures, which may inhibit it from analyzing long-term trends. Similarly, Putra &

Fahruzi [13] demonstrated that pulse sensors can obtain acceptable accuracy relative to medical-grade devices; however, this system was based on GSM communication only, and further work on calibration would be required to increase measurement precision. The study presented in [2] suggested an IoT-based health monitoring application through cloud storage, but there were also some issues raised, such as concerns about data security, interoperability between platforms, and analytics in real-time. Other works have pinpointed other issues: device dependency, incompatibility of platforms, limited responsiveness in real-time, and a continuous need for internet access [8], [12], [14].

These studies collectively confirm the potential of IoT technologies for real-time heart monitoring, while issues of accuracy, continuous data acquisition, and compatibility with mobile and other platforms, along with secure cloud-based data management, remain unresolved. Many existing systems have failed to provide the integration of continuous real-time monitoring, cloud-based historical data storage, and smooth visualization through a mobile application. Most importantly, there is empirical evidence showing the validation of IoT-based measurements against standard medical devices for ensuring reliability, which a few of the previous studies have only partially addressed.

To address these gaps, this research proposes an IoT-based heart rate monitoring system using a pulse sensor, Arduino Uno, NodeMCU ESP8266, Firebase real-time database, and a customized Android application. It merges continuous heart rate acquisition, real-time Wi-Fi transmission, cloud storage, local OLED display feedback, and mobile visualization in one architecture, which was not fully developed in previous research. Furthermore, the accuracy of the system was empirically evaluated by comparing its measurements with a standard fingertip oximeter, therefore providing quantitative evidence of reliability. This approach not only allows continuous monitoring but also provides times tamped historical data analysis, remote accessibility, and real-time anomaly observation through the Android interface.

The contributions of this research are: (1) A fully integrated IoT architecture that combines sensor acquisition, microcontroller processing, Wi-Fi transmission, cloud synchronization, local display, and Android based visualization. This addresses the fragmentation that can be seen in earlier systems. (2) An accuracy validation using a medical grade oximeter, showing that the system yields high measurement accuracy of 98.63%, improving prior studies that reported larger error margins. (3) A practical solution for personal and remote cardiac monitoring to enable tracking of continuous data, reducing reliance on clinical visits to support early detection of heart anomalies. With these contributions, this study will demonstrate the feasibility and advantages of the end-to-end IoT based real-time heart monitoring system and support wide diffusion of IoT technologies in personal and remote healthcare [15]–[18].

METHODS

Hardware and System Design

The IoT-based heart rate monitoring system proposed in this paper integrates different hardware components towards the acquisition, processing, transmission, and presentation of data. In developing the proposed IoT-based heart rate monitoring system, a systematic approach was employed. To begin with, the design rationale for the development of a continuous health monitoring system was clearly stated as: high-accuracy heart rate detection, robustness against noise and motion artifacts, seamless wireless transmission, cloud-based synchronization, and user-friendly mobile access. These capabilities provided a basis upon which each system component was selected. The pulse sensor is used since it is appropriate in PPG-based heart rate acquisition, the Arduino Uno for its stable analog-to-digital processing, the ESP8266 NodeMCU for cost-efficient Wi-Fi communication, Firebase for real-time and scalable cloud storage, and an Android application for intuitive end-user interaction. This justified framework ensures that each component directly supports the functional requirements of an end-to-end real-time monitoring system, providing coherence before elaborating on the system construction. Figure 1 presents the architecture system.

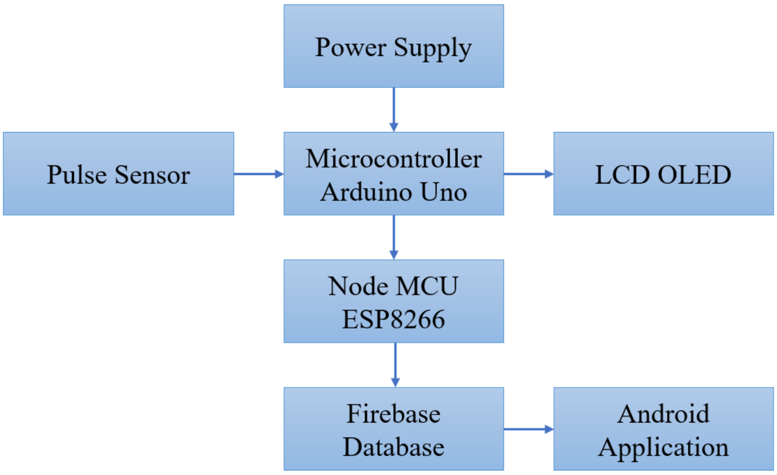


Figure 1. The Architecture System

The system is powered by a steady 5V power supply; hence, the microcontroller and sensor will operate continuously [19]. A pulse sensor measures heart rate through changes in blood flow via a photoplethysmography technique [20]–[22]. Corresponding to the heartbeats, the sensor outputs an analog signal that the microcontroller Arduino Uno will process [23]–[25]. Arduino Uno is utilized as the main microcontroller of the system. It reads the analog signals from the pulse sensor and processes the data by computing the heart rate in beats per minute (bpm) [2], [22], [23]. Moreover, Arduino also interfaces with the Node-MCU ESP8266 for data transmission. This module enables a Wi-Fi capability to transmit the processed heart rate data received from Arduino Uno to Firebase Real-time Database. Therefore, it enables the system to write data to the cloud and make data accessible anywhere in the world through internet access. Firebase acts as the backend for data storage purposes. The heartbeat data transmitted by the Node-MCU will be stored on Firebase Real-time Database so that real-time data synchronization and access are feasible. An Android application offers a user-friendly access and visualization interface for the heart rate data hosted in Firebase. This fetches data from the cloud and displays the same in a readable format [2], [4], [26]. Along with cloud visualization, the OLED display visualizes the data locally. It displays the heart rate in bpm directly on the device as real-time feedback to the user.

This hardware and system design setup enables a robust and efficient heart rate monitoring system that leverages IoT technologies for real-time health monitoring and data accessibility. The hardware configuration and connection are shown in Figure 2.

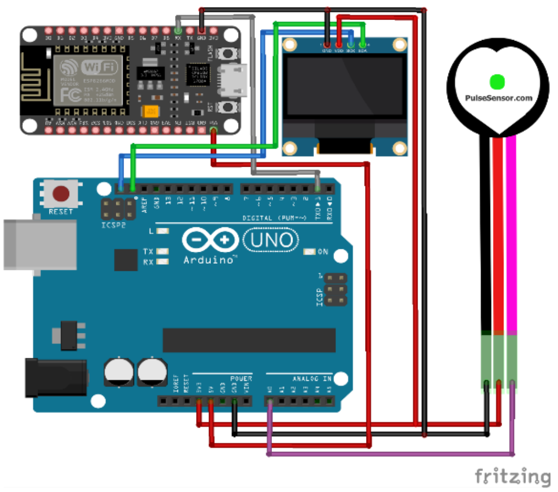


Figure 2. Hardware Configuration and Connection

The pulse sensor is connected to one of the analog input pins on the Arduino Uno (e.g., A0). The power and ground pins of the sensor are connected to the 3.3V and GND pins of the Arduino, respectively. Arduino communicates with the Node-MCU via serial communication. The TX and RX pins on the Arduino are connected to the RX and TX pins on the Node-MCU, respectively, with a voltage divider, if necessary, to protect the Node-MCU from the 5V signal of the Arduino. The OLED display 0.96 is connected to the Arduino using the I2C interface, with the SDA and SCL lines connected to the corresponding pins on the Arduino. A 5V power supply is used to power the Arduino Uno and the Node-MCU ESP8266. A voltage regulator or separate power supply might be used to ensure stable power delivery.

Flowchart System

A flowchart can visually represent the sequence of actions performed by the components of an IoT-based heart rate monitoring system, effectively illustrating the flow of data and operations. Figure 3 displays a comprehensive flowchart of the system.

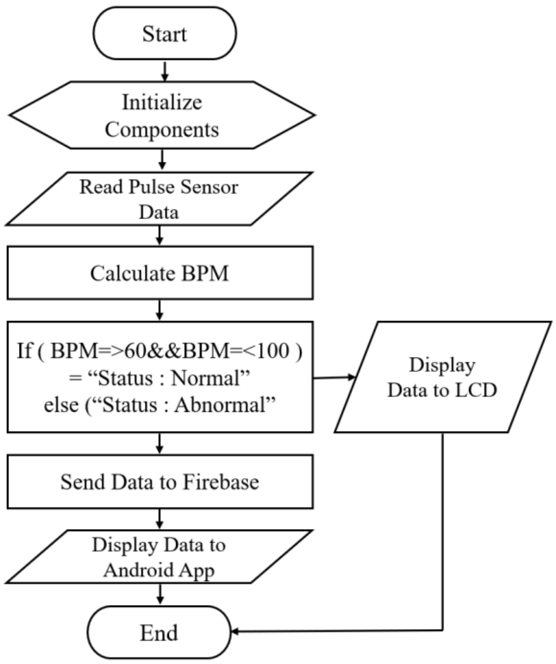


Figure 3. Flowchart System

ISSN 2580-6424 (printed) | ISSN 2477-2399 (online)

It initializes the pulse sensor, OLED display, and Wi-Fi module (Node-MCU ESP8266) upon power-up of the system. It establishes their communication protocols and makes sure all parts are ready to acquire and process data. The pulse sensor continuously detects the heart rate of the user. An Arduino Uno reads the analog signal output from the sensor. The Arduino processes the analog signal to detect heartbeats and calculates the BPM according to the detected pulse intervals. The system checks whether the BPM falls in the normal range of 60-100 BPM [25]. In that case, it displays "Status: Normal" on the OLED; otherwise, it shows "Status: Abnormal." Such calculated BPM and status are transmitted to the Firebase database via Node-MCU ESP8266. The step helps in storing data safely in the cloud for real-time access and analysis. The Android application fetches the latest data from Firebase and updates the user interface with the current heart rate and status. This allows users to monitor their heart rate in real time from their mobile devices.

Data Analysis

The data analysis procedure was designed to rigorously evaluate the accuracy and reliability of the proposed IoT-based heart rate monitoring system. For accuracy assessment, heart rate measurements obtained from the IoT system were compared with those recorded by the Fingertip Pulse Oximeter LK88 medical device. A total of six adult participants were recruited for this study, with each participant completing ten testing sessions in order to produce diverse data points representing different scenarios under controlled conditions. Prior to the collection of data, the IoT-based system comprising the pulse sensor, Arduino Uno, and Wi-Fi module was calibrated to ensure stable signal acquisition. In this setup, the pulse sensor was placed on the participant's fingertip, ensuring that it did not move or exert variable pressure to obtain the best PPG signal.

Two quantitative indicators were used to systematically determine measurement accuracy: percentage error and percentage accuracy. Percentage error reflects the degree of deviation between the IoT-based measurement (X_n) and the actual value obtained from the oximeter (Y_n). Percentage accuracy indicates how close the IoT system's measurement is to the reference value, expressed as the complement of the percentage error [27]–[29]. Both metrics were calculated using Equations (1) and (2):

$$Percentage\ Error = \left|\frac{Y_n - X_n}{Y_n}\right| x100\% \quad \text{.........................} (1)$$

$$Percentage\ Accuracy = \left[1 - \left|\frac{Y_n - X_n}{Y_n}\right|\right] x100\% \quad \text{..........} (2)$$

Where Y_n donates the heart rate measured by the oximeter, and X_n denotes the heart rate obtained from the IoT-based system. By applying these formulas to all 60 paired measurements, the study derived a comprehensive accuracy profile of the system, enabling an objective evaluation of its performance against that of a medical-grade device. This sort of analysis offers a transparent and reproducible means of validating the system for real-time heart rate monitoring.

RESULT AND DISCUSSION

Pulse Sensor Calibration

Sensor calibration is applied to differentiate on which signals are heartbeats by thresholding the analog values read with respect to the sensor. These values are in the range of 0-1024. In Arduino-based approaches, the calibration involves some adjustments of the sensitivity of the sensor within its program code. One option is to monitor the signals in real time using the Arduino serial plotter [4] and adjust the detection threshold. If the signal exceeds the threshold, the attached LED will light up; otherwise, it remains off. The calibration process involves gently placing a finger on the front of the pulse sensor, while observing the pulse waveform on the serial plotter, synchronized with the blinking LED, as shown in Figure 4.

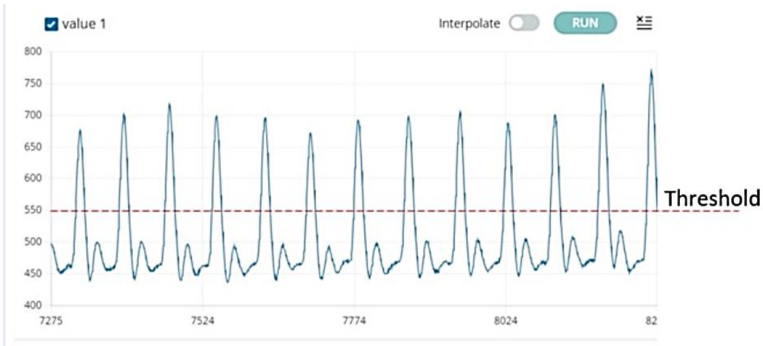


Figure 4. Observing The Pulse Waveform on The Serial Plotter

The real-time pulse waveform in Figure 4 shows the calibration stage pulse waveform on the Arduino Serial Plotter. The waveforms peaks show the signal of one detected heartbeat. The signal overall showing an even signal stream/consistency indicates that the sensor is correctly placed and that there is just the right amount of pressure from the finger. Visualization serves as an effective feedback mechanism in calibrating the sensor and determining the best detection threshold. The user can see the pulses in real time and set levels that correspond to the height of the blood pulse. When the sensor is powered with exactly 5V, the pulse sensor signal value's analog range is 500 to 580 during a heartbeat cycle. A threshold of approximately 550 had to be set in this study, because of the need to make sure all detected peaks are indeed. Sufficient sensor pressure is necessary as well; pressure that is too little or too firm will cause the waveforms to disappear. This visualization helps with proper placement of the sensor for detecting the waveforms.

Pulse Sensor Testing

The pulse sensor testing aims to determine the accuracy of beats per minute (BPM) measurements compared to a standard medical device (oximeter), assess the consistency of the pulse sensor readings over multiple sessions, and identify any potential discrepancies or issues such as noise interference or sensitivity to motion. The heart rate measurement from both the pulse sensor and the oximeter are shown in Figure 5.

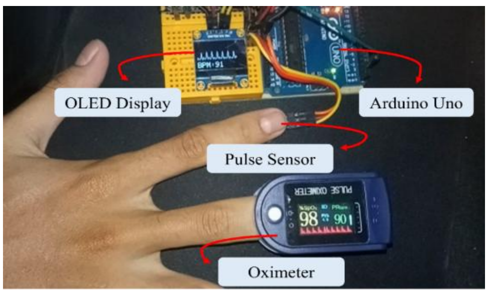
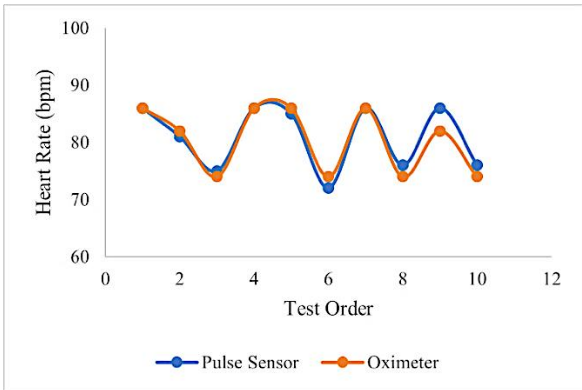
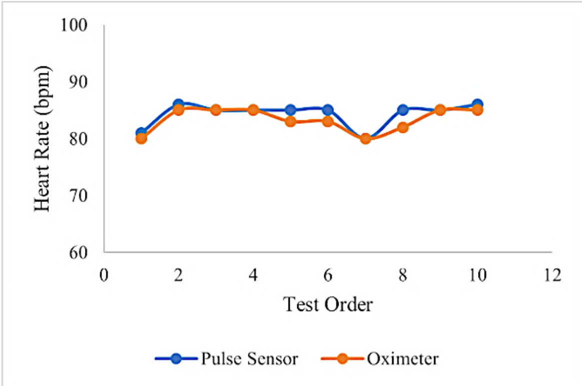


Figure 5. Heart Rate Measurement from Both The Pulse Sensor and The Oximeter

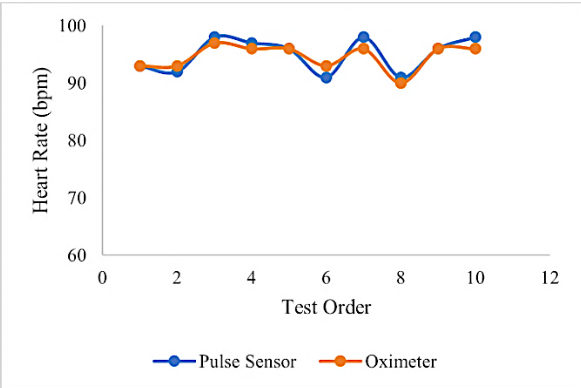
To optimize the pulse sensor's accuracy, the study was conducted in a soundproof room with limited other stimuli that would interfere with the electronics. A total of 6 participants between the ages of 18 to 23 were tested in 10 iterations each. The pulse sensor was positioned on the fingertips, as in Fig. 5. The processor of BPM from the basal Arduino Uno microcontroller received the analog signals. An oximeter was also positioned on the opposite fingertip to serve as a reference for the BPM. The collection of measurements from the pulse sensor and the oximeter was performed in 1-minute intervals for a total of 10 minutes. BPM measurements were taken in minute intervals for 10-minute intervals resulting in 60 BPM measurements for each device. A sample of the heart rate measurement is shown in Figure 6.



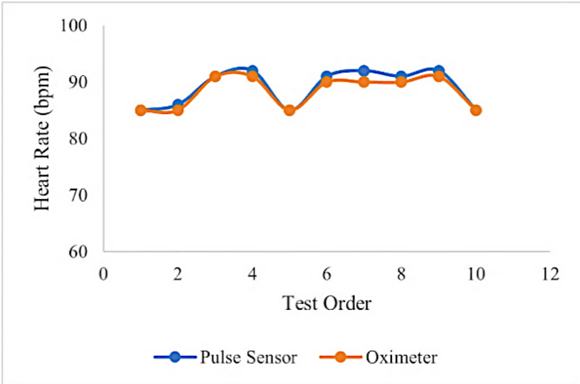
(a)



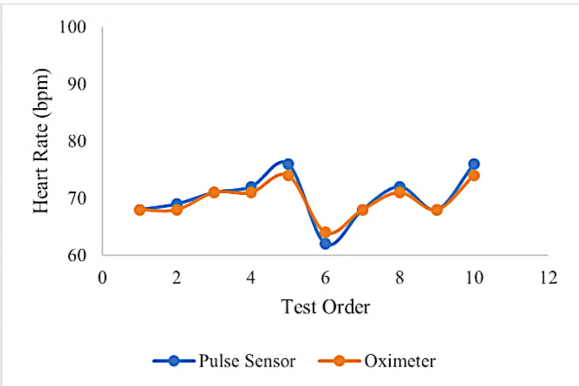
(b)



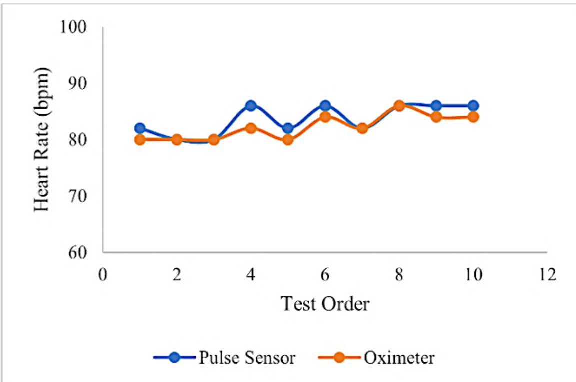
(c)



(d)



(e)



(f)

Figure 6. Heart Rate Measurement for (a) First, (b) Second, (c) Third, (d) Fourth, (e) Fifth, and (f) Sixth Participant

Heart rate values for each participant after the completion of the measurement cycles are illustrated in each of the subfigures (a–f) in figure 6. These plots display the steady state values from the measurements as well as physiological fluctuations from the measurements. The pulse sensor outcomes for every participant correspond with the values from the oximeter, which indicates the IoT-based system can be used reliably.

The pulse sensor BPM readings were compared to the oximeter readings to obtain error and accuracy. The error is a difference between the pulse sensor and the standard medical device (oximeter) heart rate reading. It shows the extent to which the measurements are not accurate, or the actual/reference measurements offered by the oximeters. Accuracy, on the other hand, implies how close the

ISSN 2580-6424 (printed) | ISSN 2477-2399 (online)

measurements of the heart rate of the IoT-based system are to the reference values given by the oximeter [4], [25]. Table 1 indicates the percentage of error and accuracy of measurement.

Table 1. The Error and Accuracy of Measurement

Participant	Error (%)	Accuracy (%)
1	1.67	98.33
2	1.21	98.79
3	1.06	98.94
4	1.28	98.72
5	1.70	98.30
6	1.31	98.69

The mean error between the pulse sensor and the oximeter readings had a low error average, which implied that the readings of the pulse sensor were reasonably near to the oximeter. Average accuracy of the system was estimated at 98.63. This system can enhance precision in past studies [13]. This accuracy rate is high, showing that pulse sensors are a reliable measuring instrument of heart rate. The pulse sensor testing indicated that the sensor has the capability of measurement by giving correct and stable heart rate readings that were like reading off a conventional medical oximeter. The outcomes suggest that pulse sensors can be a feasible element of the IoT-based heart rate monitoring system, which provides credible information to monitor health in real-time. It was also noted that, when the participants shifted their fingers slightly, small differences were recorded in the readings of BPM, which led to the development of temporary inaccuracies [21], [22]. This indicates that the pulse sensor can be affected by motion artifact, which is a characteristic of PPG-based sensors [22]. The pulse sensor should be placed properly to get accurate reading. To read the sensor, inconsistent placement or differentiated pressure put on the sensor can have an impact on the signal quality [20]–[22].

Physical Device and Android Application

The hardware device consists of four main hardware parts, among them pulse sensors, Arduino Uno, Node-MCU ESP8266, an OLED Display, and a power supply. All the components were mounted on a small, handheld prototype board and thus participants were allowed to manipulate and utilize them conveniently. The pulse sensor was connected to the Arduino Uno which interprets the signal and transmits it to the Node-MCU to upload it to Firebase. The OLED display displayed BPM and status instantly and the people could check their heart rate in real-time. The device was also checked to be functional in different conditions, that is, different positions of the user and movements. It aimed at ensuring constant readings of the pulse sensor, consistent data processing by the Arduino, and stability in the data uploaded to Firebase. The physical device worked as intended as it would show relevant data on the OLED screen and send data to Firebase regularly and with the correct recording of the heart rate. The display on the screen was satisfactory with the OLED screen being visible and with information being easily readable. Figure 7 displays the physical heart rate measuring device.

Figure 7. Physical Device for Heart Rate Measurement

The prototype of the hardware is assembled and presented in Figure 7: pulse sensor, Arduino Uno, NodeMCU ESP8266, and OLED. Physical design illustrates the integration of the elements that serve to receive, process and transmit heart rate information in real time. The OLED display gives instant BPM feedback to the user.

Android Studio was used to create the Android application that serves as the heart rate monitoring system which uses the Android system as the IoT system, and Java as the programming language. The user interface that will be used to access real-time heart rate data in Firebase database is the application. The app has a well-organized and user-friendly interface, showing real-time heart rate measurements and activity (Normal/Abnormal). It also has the option of notifying about abnormal readings and exporting data to perform additional analysis. The application is connected to the Firebase database so that it can retrieve the most recent heart rate information. It also allows the user to see historical data in graphic formats so that users can see the changes over time. Heart rate monitoring user interface under Android application is illustrated in Figure 8.

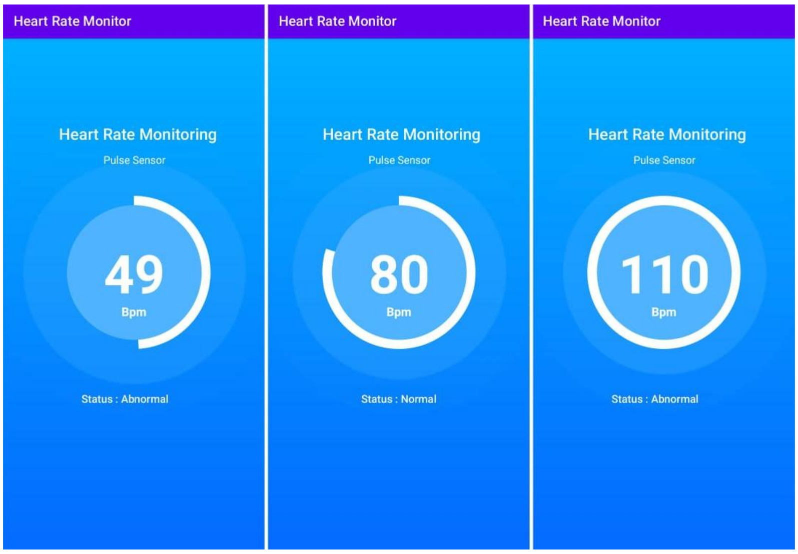


Figure 8. User Interface for Heart Rate Monitoring in Android Application

The interface of the Android application that is used to visualize the real-time heart rate data retrieved through Firebase is depicted in figure 8. The interface shows the present BPM, status (Normal/Abnormal), and helps it to access time-stamped historical data. This visualization shows that it is possible to combine a cloud-synchronized heart rate monitoring system with a mobile platform.

The app showed the real-time heart rate information correctly and with a low delay. The visual representations of data were understandable and educative, and the users were able to analyze their heart rate patterns against the past studies [30]–[33]. The app was also compatible with various devices, its data loaded quickly and its user interface was user-friendly. The layout and design elements were recommended to be slightly improved to make the visual appeal and readability.

Accuracy testing and system performance obtained results show that the proposed architecture is viable in the case of continuous real-time heart rate monitoring. This accuracy of 98.63% is an indication that pulse sensor and processing algorithm could effectively record physiological signals and therefore makes it possible to use low-cost IoT elements to monitor medical related situations. The fact that the real-time flow of the information to Firebase and instant visualization on the Android application have been made successful is the testament that the IoT-mobile integration facilitates the unhindered transfer of the information between sensing and end-user access. This feature is necessary in maintaining continuous monitoring because users or healthcare providers can watch changes in heart rate in real-time, without having to interact with the device directly. The fact that historic data in the cloud can be timed also enhances the superiority of the system compared with the Bluetooth-only system or non-logging system [4] in that it allows one to analyze long-term trends and monitor the system remotely. Taken together, the results indicate that the suggested IoT-based system is both functional and reliable and offers feasible advantages to both personal and remote heart health management.

The combination of local feeds of OLED display, Wi-Fi-based data transmission, and mobile accessibility emphasizes that the system was specifically beneficial in multi-layered monitoring. The user can get immediate insight on the device itself and with the help of the Android app, the same can be viewed by clinicians or caregivers remotely. This dual-access scheme directly solves the shortcomings in the literature including the limited monitoring range [2], the absence of historical data [4] and the reliance on individual communication schemes [13]. These benefits justify the possible use of the system in the early detection of anomalies and remote patient monitoring, which is the principal aim of the research.

CONCLUSION

The results of the proposed study provide a clear response to the research question on the possibility of implementing the IoT technology to a mobile application to constantly monitor the heart rate. The system manages to acquire real-time data, transmit it using a stable Wi-Fi, and auto-synchronize with Firebase, which proves that the general architecture is reasonably reliable in practice. Android application is also helpful in the retrieval and presentation of live heart rate information, which proves that it is a useful mobile monitoring tool. The accuracy test with 98.63% match to a standard oximeter is a strong evidence that the system can be able to generate reliable heart rate readings with the use of low-cost parts. These findings address the research objective of making the system valid against a medical reference device. The advantages of the system that included constant monitoring, remote access to data and recording of historical data were directly illustrated by testing. Nevertheless, there are still certain drawbacks, such as the sensitivity to finger motion, and the necessity to have a stable Wi-Fi connection. The problems offer the possibility to advance them in the workplace in the future.

REFERENCES

[1] T. F. Bernadus, L. B. Subekti, and Y. Bandung, "IoT-based fall detection and heart rate monitoring system for elderly care," *Proceeding - 2019 International Conference on ICT for Smart Society: Innovation and Transformation Toward Smart Region, ICISS 2019*, 2019, doi: 10.1109/ICISS48059.2019.8969845.

[2] M. Alruwaili *et al.*, "A health monitoring system using IoT-based android mobile application," *Computer Systems Science and Engineering*, vol. 47, no. 2, pp. 2293–2311, 2023, doi: 10.32604/CSSE.2023.040312.

[3] S. U. Rehman, I. Sadek, B. Huang, S. Manickam, and L. N. Mahmoud, "IoT-based emergency cardiac death risk rescue alert system," *MethodsX*, vol. 13, no. April, p. 102834, 2024, doi: 10.1016/j.mex.2024.102834.

[4] Sahana S Khamitkar, "IoT based system for heart rate monitoring," *International Journal of Engineering Research and*, vol. V9, no. 07, pp. 1563–1571, 2020, doi: 10.17577/ijertv9is070673.

[5] S. Sivanandan, V. Muthukumar, G. Suresh, and P. Anandan, "Integrated system for real-time health management in rural areas with medicine dispatcher box," *Proceedings of the 2nd International Conference on Edge Computing and Applications (ICECAA 2023)*, no. Icecaa, pp. 704–708, 2023, doi: 10.1109/ICECAA58104.2023.10212228.

[6] R. Latha, U. Arul, S. Dhanalakshmi, R. Ramya, and N. Mishra, "Cloud-based smart sensing system for real-time health management of dementia patients," *2023 2nd International Conference on Smart Technologies for Smart Nation (SmartTechCon 2023)*, pp. 923–927, 2023, doi: 10.1109/SmartTechCon57526.2023.10391592.

[7] L. Catarinucci *et al.*, "An IoT-aware architecture for smart healthcare systems," *IEEE Internet of Things Journal*, vol. 2, no. 6, pp. 515–526, 2015, doi: 10.1109/JIOT.2015.2417684.

[8] G. Singh, N. Chaudhary, D. Ather, R. Kler, and M. Arora, "Revolutionizing remote healthcare: proposing an IoT & arduino-based integrated approach for real-time health monitoring," *2023 4th International Conference on Computation, Automation and Knowledge Management (ICCAKM 2023)*, no. Iccakm, pp.

1–5, 2023, doi: 10.1109/ICCAKM58659.2023.10449530.

[9] N. S. Hammed and M. I. Owis, "An integrated health monitoring system," *2015 International Conference on Advances in Biomedical Engineering (ICABME 2015)*, pp. 197–200, 2015, doi: 10.1109/ICABME.2015.7323286.

[10] G. Neagu, S. Preda, A. Stanciu, and V. Florian, "A cloud-IoT based sensing service for health monitoring," *2017 E-Health and Bioengineering Conference, EHB 2017*, pp. 53–56, 2017, doi: 10.1109/EHB.2017.7995359.

[11] D. Tiwari, D. Prasad, K. Guleria, and P. Ghosh, "IoT based smart healthcare monitoring systems: A review," *Proceedings of IEEE International Conference on Signal Processing,Computing and Control (ISPCC 2k21)*, vol. 2021-Octob, pp. 465–469, 2021, doi: 10.1109/ISPCC53510.2021.9609393.

[12] B. C. Kavitha, K. V. Reddy, U. J. Kumar, K. Sivakrishna, K. S. Sankaran, and G. Vimala Rani, "IoT based remote health monitoring system," *2021 International Conference on Computational Performance Evaluation, (ComPE)*, pp. 209–212, 2021, doi: 10.1109/ComPE53109.2021.9751721.

[13] D. Putra and I. Fahruzi, "Design and implementation internet of medical things for heart rate monitoring system using arduino and GSM network," *Proceedings - 2021 International Seminar on Application for Technology of Information and Communication: IT Opportunities and Creativities for Digital Innovation and Communication within Global Pandemic (iSemantic 2021)*, pp. 48–53, 2021, doi: 10.1109/iSemantic52711.2021.9573244.

[14] T. Jagadesh, S. Rithik, B. Nithish Kumar, R. R. Pv, and R. Rithika, "IoT based health monitoring in hospitals," *Proceedings of 8th IEEE International Conference on Science, Technology, Engineering and Mathematics (ICONSTEM)*, pp. 1–5, 2023, doi: 10.1109/ICONSTEM56934.2023.10142416.

[15] C. Li, J. Wang, S. Wang, and Y. Zhang, "A review of IoT applications in healthcare," *Neurocomputing*, vol. 565, no. May 2023, 2024, doi: 10.1016/j.neucom.2023.127017.

[16] E. Mbunge, B. Muchemwa, S. Jiyane, and J. Batani, "Sensors and healthcare 5.0: transformative shift in virtual care through emerging digital health technologies," *Global Health Journal*, vol. 5, no. 4, pp. 169–177, 2021, doi: 10.1016/j.glohj.2021.11.008.

[17] S. Akinola and A. Telukdarie, "Sustainable digital transformation in healthcare: advancing a digital vascular health innovation solution," *Sustainability (Switzerland)*, vol. 15, no. 13, 2023, doi: 10.3390/su151310417.

[18] P. Mishra and G. Singh, "Internet of medical things healthcare for sustainable smart cities: current status and future prospects," *Applied Sciences (Switzerland)*, vol. 13, no. 15, 2023, doi: 10.3390/app13158869.

[19] I. Inayah, H. Kiswanto, A. Dimyati, and M. G. Prasetia, "Design and implementation of non-contact infrared thermometer based MLX90614 and ultrasonic sensors," *Jurnal Fisika dan Aplikasinya*, vol. 18, no. 2, p. 42, 2022, doi: 10.12962/j24604682.v18i2.10746.

[20] J. Park, H. S. Seok, S. S. Kim, and H. Shin, "Photoplethysmogram analysis and applications: An integrative review," *Frontiers in Physiology*, vol. 12, no. March, pp. 1–23, 2022, doi: 10.3389/fphys.2021.808451.

[21] K. B. Kim and H. J. Baek, "Photoplethysmography in wearable devices: A comprehensive review of technological advances, current challenges, and future directions," *Electronics (Switzerland)*, vol. 12, no. 13, 2023, doi: 10.3390/electronics12132923.

[22] E. M. Dewi, D. Rahmawati, and N. W. Kirana, "Wireless photoplethysmography (PPG) measurement with pulse wave velocity (PWV) method for arterial stiffness evaluation," *Jurnal Rekayasa Elektrika*, vol. 20, no. 1, pp. 1–5, 2024, doi: 10.17529/jre.v20i1.33486.

[23] M. Haryo, N. Arvika, and W. Latri, "Implementation of heart rate system using AD8232 and arduino microcontrollers," *Signal and Image Processing Letters (SIMPLE)*, vol. 2, no. 1, pp. 36–44, 2020.

[24] A. M.A Shakarji and A. Golcuk, "Arduino based system design for measuring heart rate and body temperature," *International Journal of Applied Mathematics Electronics and Computers*, vol. 10, no. 1, pp. 15–19, 2022, doi: 10.18100/ijamec.1029843.

[25] Y. Yohana, M. R. Hidayat, Y. Y. Kusmala, and S. Sambasri, "Heart rate measurement system based on pulse sensors with pathophysiological appearance," *Article in Journal of Measurements Electronics Communications and Systems*, vol. 05, no. December, pp. 15–19, 2019, doi: 10.25124/jmecs.v5i1.2049.

[26] M. Hamim, S. Paul, S. I. Hoque, M. N. Rahman, and I. Al Baqee, "IoT based remote health monitoring system for patients and elderly people," *1st International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST)*, pp. 533–538, 2019, doi: 10.1109/ICREST.2019.8644514.

[27] I. Inayah, N. Hayati, A. Nurcholis, A. Dimyati, and M. G. Prasetia, "Realtime monitoring system of solar panel performance based on internet of things using blynk application," *Elinvo (Electronics, Informatics, and Vocational Education)*, vol. 7, no. 2, pp. 135–143, Feb. 2023, doi: 10.21831/elinvo.v7i2.53365.

[28] M. Roza Lia, H. Harmadi, and A. Muttaqin, "Development of measuring instruments for lung vital capacity and human respiratory rate based on fiber optic sensors," *Jurnal Ilmu Fisika | Universitas Andalas*, vol. 15, no. 1, pp. 39–47, 2022, doi: 10.25077/jif.15.1.39-47.2023.

[29] S. F. Nasution, H. Harmadi, S. Suryadi, and B. Widiyatmoko, "Development of river flow and water quality using IoT-based smart buoys environment monitoring system," *Jurnal Ilmu Fisika*, vol. 16, no. 1, pp. 1–12, 2023, doi: 10.25077/jif.16.1.1-12.2024.

[30] N. Dyanningrum, D. Zulherman, and H. Pujiharsono, "Analisis rancangan sistem pengukuran denyut nadi berbasis internet of things," *Proceedings on Conference on Electrical Engineering*, vol. 1, no. 2, pp. 13–18, 2018.

[31] I. Anshari Putra, A. Ali Muayyadi, and D. Perdana, "Implementasi sistem monitoring detak jantung dan suhu tubuh menggunakan sensor pulse dan blynk application berbasis internet of things," *e-Proceeding of Engineering*, vol. 8, no. 6, pp. 3116–3123, 2022.

[32] M. A. Saputro, "Implementation of a wireless heart rate and body temperature monitoring system," *Development of Information Technology and Computer Science*, vol. 1, no. 2, pp. 148–156, 2017.

[33] J. W. Jeong, W. Lee, and Y. J. Kim, "A real-time wearable physiological monitoring system for home-based healthcare applications," *Sensors*, vol. 22, no. 1, 2022, doi: 10.3390/s22010104.