Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Issue 1, Januari 2026
DOI : https://doi.org/10.33395/sinkron.v10i1.15817

e-ISSN : 2541-2019
p-ISSN : 2541-044X

Experimental Characterization of ESP-Mesh Performance for Resilient Medical IoT Monitoring Systems

Zulfikar Achyar1), **Indrawati**2), **Ilham Safar**3) *, **Dewi Wahyuni**4)
1,2) Computer and Network Engineering Technology, Faculty of Information and Computer Technology, Politeknik Negeri Lhokseumawe, Aceh, Indonesia
3) Multimedia Engineering Technology, Faculty of Information and Computer Technology, Politeknik Negeri Lhokseumawe, Aceh, Indonesia
4) Information Systems, Faculty of Technology, Universitas Battuta, Medan, Indonesia
1) zulfikaerachyar@gmail.com, 2) indrawati@pnl.ac.id, 3) ilham_safar@pnl.ac.id, 4) dewiwahyuni@battuta.ac.id

Submitted : Jan 1, 2026 | **Accepted :** Jan 17, 2026 | **Published :** Jan 20, 2026

Abstract: The reliability of medical Internet of Things (IoT) systems is critically dependent on network resilience, particularly in indoor hospital environments where conventional Wi-Fi infrastructures are vulnerable to single points of failure. Although ESP-Mesh has emerged as a promising self-healing communication protocol, its performance characteristics in medical IoT monitoring contexts remain insufficiently explored. This study aims to experimentally characterize the performance of ESP-Mesh networks for resilient medical IoT monitoring systems by analyzing multi-hop latency behavior, signal degradation, and communication stability under indoor medical-like conditions. A multi-parameter monitoring prototype integrating infusion volume, drip rate, and heart rate sensors was deployed as an experimental platform. Network performance was evaluated through controlled measurements of RSSI, end-to-end latency, and self-healing behavior, while MQTT was employed to assess cloud-based transmission efficiency. The results demonstrate that ESP-Mesh maintains stable self-healing communication with an average multi-hop latency of 0.714 s across distances up to 5 m, with latency increasing consistently as RSSI decreases. MQTT cloud transmission achieved a lower average latency of 0.247 s with zero packet loss, confirming its suitability for lightweight medical data delivery. Sensor evaluation revealed high accuracy for infusion volume monitoring (95.42%), while heart rate and drip rate measurements exhibited lower reliability due to signal interference and environmental sensitivity. These findings provide empirical insights into the performance limits and trade-offs of ESP-Mesh networks in medical IoT environments. The study confirms the feasibility of ESP-Mesh as a resilient communication backbone for medical monitoring, while highlighting the necessity of advanced signal processing to achieve clinical-grade sensing reliability..

Keywords: ESP-Mesh; Medical Internet of Things; Self-Healing Networks; Performance Characterization; Network Resilience; MQTT

INTRODUCTION

Recent studies on medical Internet of Things (IoT) networking can generally be grouped into three main directions. The first focuses on the development of sensor-based monitoring systems for physiological parameters such as heart rate, infusion flow, or oxygen saturation, emphasizing hardware integration and basic data transmission reliability (Syauqy & Primananda, 2019; Maesyarani et al., 2024). The second direction concentrates on the use of lightweight communication protocols, particularly MQTT, to improve real-time data delivery and cloud connectivity efficiency (Abilovani et al., 2018; Imansyah et al., 2023). The third direction explores alternative wireless architectures, including wireless sensor networks and mesh-based topologies, to enhance coverage and fault tolerance in indoor environments (Khan et al., 2022; Irawan et al., 2024). Although these studies demonstrate the feasibility of IoT for healthcare applications, most of them treat the communication network merely as a supporting component rather than as an object of systematic experimental investigation.

Consequently, there remains a significant research gap in the empirical understanding of how self-healing mesh networks behave under medical indoor monitoring constraints. Existing studies have not sufficiently characterized

*name of corresponding author

multi-hop latency behavior, RSSI degradation patterns, and operational stability of ESP-Mesh networks when deployed in real-time medical IoT contexts (Khan et al., 2022; Imansyah et al., 2023). In particular, experimental evidence regarding the performance limits and resilience–latency trade-offs of ESP-Mesh as a communication backbone for medical monitoring systems remains limited, even though such characteristics are critical for healthcare environments where reliability and timeliness are essential (Irawan et al., 2024; Maesyarani et al., 2024).

Therefore, this study aims to experimentally characterize the performance of ESP-Mesh networks in resilient medical IoT monitoring systems. This research investigates multi-hop communication latency, signal strength degradation, and network stability under controlled indoor medical-like environments, using a multi-parameter monitoring platform as an experimental testbed (Syauqy & Primananda, 2019; Khan et al., 2022). Rather than merely developing a monitoring system, this work positions ESP-Mesh as the primary object of investigation in order to reveal its empirical performance characteristics and feasibility for medical IoT applications.

The novelty of this study lies in providing an experimental characterization of ESP-Mesh behavior specifically within a medical IoT monitoring context. The contributions of this research are threefold: (1) experimental characterization of multi-hop ESP-Mesh latency and RSSI behavior under indoor medical environments; (2) empirical identification of performance trade-offs between network resilience and transmission delay in self-healing medical IoT systems (Khan et al., 2022; Irawan et al., 2024); and (3) formulation of a resilient medical IoT communication architecture integrating ESP-Mesh and MQTT as a lightweight cloud interface (Abilovani et al., 2018; Imansyah et al., 2023). These contributions aim to advance the scientific understanding of self-healing mesh networks for healthcare applications and support the development of more reliable medical IoT infrastructures.

LITERATURE REVIEW

Research into IoT-based health monitoring systems has progressed rapidly, particularly in efforts to automate vital parameter monitoring in hospitals (Aprilia, A., & Sollu, T.S., 2021). Several previous studies have served as the foundation for the development of this system.

IoT Communication Protocols in Healthcare (Irawan et al., 2024) highlights the importance of reliable data transmission in medical IoT devices using the ESP-Mesh protocol (Khan, A.U., Khan, M.E., Hasan, M., Zakri, W., Alhazmi, W., & Islam, T., 2022). The results of this study indicate that mesh networks are able to overcome the limitations of signal range in hospital infrastructure that has many physical obstacles. The main differences with this study lie in the more complex sensor integration and the use of the MQTT protocol as a message broker for more efficient data management.

Infusion Fluid and Heart Rate Monitoring Previous infusion monitoring systems often focused on only one parameter (Ananta, A., Nasir, M., & Erdiansyah, U., 2024), such as residual fluid weight or drip rate (Lani, MFN, & Da Ate, YF, 2025). The use of data analytics technology for monitoring health trends, however, has not implemented a distributed network that is resilient to node failures (Imansyah, F., Ratiandi, R., Marpaung, J., Suryadi, D., & Hizballah, F., 2023). This study addresses these shortcomings by combining three parameters at once (infusion volume, drip rate, and BPM) in one integrated monitoring system (Maesyarani, AA, Samjantungsumar, LD, Zaenudin, Z., Akbar, A., & Suryadi, E., 2024).

MQTT Implementation (Syauqy, D., & Primananda, R., 2019) and Data Security The use of the MQTT protocol (Abilovani, ZB, Yahya, W., & Bakhtiar, FA, 2018) in medical data transmission was chosen because of its lightweight nature and low power consumption. Traffic analysis using Wireshark (Ubaedila, I., Nurdiawan, O., Wijaya, YA, & Sidik, J., 2021) is very effective for monitoring the quality of service (Quality of Service) on IoT networks (Rizki, A., Muhammad, M., & Nasri, N., 2021). In this study, the sniffing method was applied to ensure data latency remains within safe limits for real-time medical monitoring.

Visualization and user interface play a crucial role in the successful implementation of systems in medical settings. The use of the Flask framework is highly effective in building responsive helpdesk information systems (Wijayanto & Susetyo, 2022). This study adapts the use of Flask (Azhari, RY, 2022) to build a local website dashboard combined with a Flutter-based mobile application (Taufiq, A., Pratama, M., & Pratama, AR, 2021) to provide flexible data access for medical personnel (Jacobus, A., Lendo, R., & Mapaly, HA, 2023).

Gap Analysis and Novelty Based on the literature review above, there is a research gap where conventional medical IoT systems still rely heavily on a single point of access (Single Point of Failure) (Pratama, RNW, Widagda, MEP, & Hadiyanto, H., 2024). If the central router goes down, the entire monitoring system is paralyzed. The novelty of this research is the implementation of the self-healing feature in the ESP-Mesh protocol which allows the network to recover routes independently without manual intervention, thus ensuring higher data availability compared to standard IoT systems.

METHOD

This research methodology is designed to develop solutions to the limitations of medical monitoring. conventional through an experimental approach. This research includes the design of an integrated system that combines hardware (sensors) (Ramadhan Pradana, D., Sari, MI, & Rimasa, D., 2023), network (ESP-Mesh & MQTT) (Maulana Putra, J.,

Misbahuddin, & Al Sasongko, SM 2022), and software (Web & Mobile) (Raven, Y., & Mulyana, TMS, 2022).
This research was conducted through four main stages: system design, hardware development, ESP-Mesh network configuration, and performance testing.

a. Device Specifications and Topology: This system uses an ESP8266 microcontroller as the main processing unit on each node. The network topology implemented is a Mesh Network, where data is sent hop-by-hop to a single root node connected to the internet. Data is then sent to the cloud broker using the MQTT protocol with a Quality of Service (QoS) level of 0 to ensure fast transmission.
b. Instrument and Sensor Calibration: To ensure data validity, each sensor is calibrated using a ground truth instrument:
    1. Infusion Volume (Load Cell): Calibrated using a precision digital scale using a linear regression method to obtain an accurate calibration factor.
    2. Drip Rate (Infrared): Uses an infrared sensor to detect fluid drop interruptions in the infusion tube.
    3. Heart Rate (MAX30102): Heart rate parameters are compared with a standard clinical pulse oximeter (Beurer brand or equivalent) to calculate the error rate.
c. Network Testing Scenarios Latency testing was conducted in two scenarios:
    1. ESP-Mesh Test: Measures data travel time from the leaf node to the root node over distances ranging from 1 to 5 meters in indoor conditions with numerous obstructions (walls and furniture).
    2. MQTT Test: Measures end-to-end latency from the root node to the monitoring dashboard on the user's device over the internet, by calculating the average travel time from 50 transmission attempts.

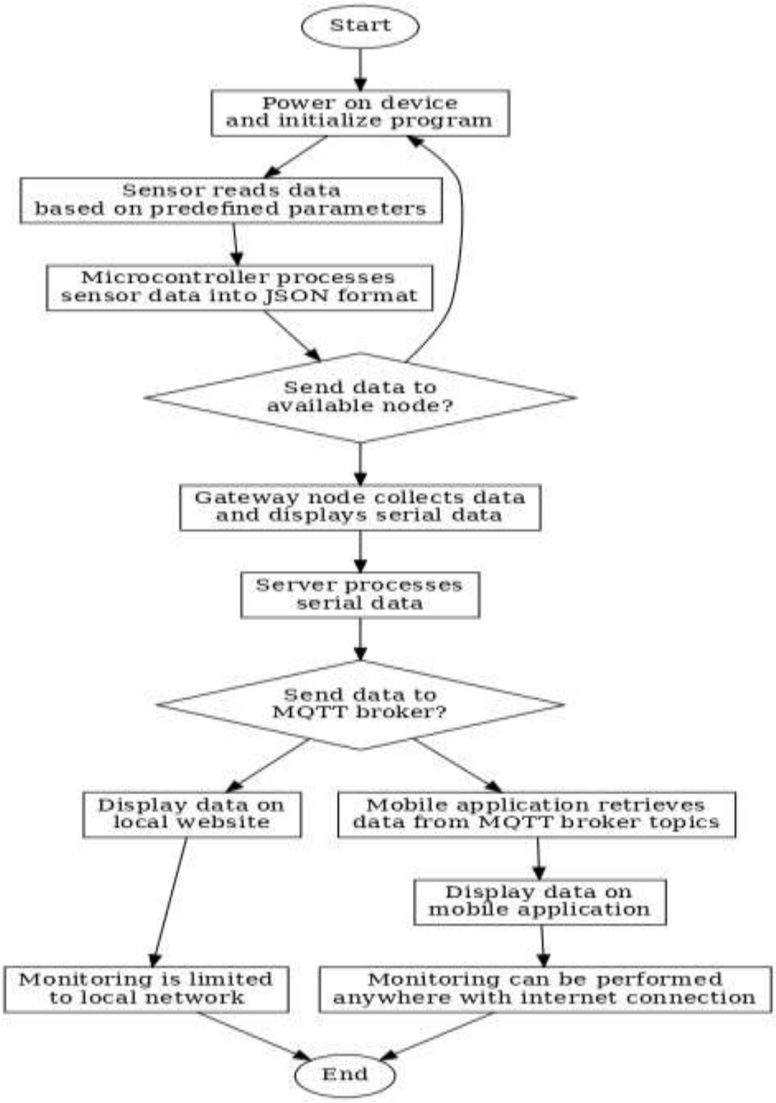

Fig 1. System Flowchart

*name of corresponding author

Data Analysis and Testing Techniques

To ensure the validity of the system developed, a series of quantitative tests were carried out on sensor performance and network reliability.

Sensor Accuracy Testing

Sensor accuracy is a crucial parameter to ensure the reliability of the transmitted medical data.
Testing is carried out by comparing the values read on the IoT system to the actual values from a standard medical measuring instrument (calibrator). The reading error rate is calculated using a percentage formula. *error* as follows:

$$Error(\%) = \frac{|V_{actual} - V_{sensor}|}{V_{Actual}} \ x\ 100\%$$

Where V_{actual} is a value measured on a standard instrument and V_{sensor} is the value that is read by the monitoring system. The smaller the percentage *error* generated, the higher the level of system accuracy.

Load Cell Calibration

When measuring the remaining weight of the infusion fluid, the sensor *Load Cell* requires a calibration process

$$y = mx + c$$

to convert the analog signal from the HX711 module into units of weight (grams/milliliters). This process uses a linear regression approach to determine the appropriate scale factor, with the basic equation:

The variable y represents the final weight output, x is the raw value input (*raw data*) of the sensor, while m and c are calibration constants obtained through static load mapping during the initial testing phase.

Network Performance Analysis (QoS)

The reliability of data transmission via ESP-Mesh and MQTT protocols is measured based on the latency parameter (*delay*). Latency is calculated to determine the data travel time from when the sensor takes a sample until the data is successfully displayed on the screen *Dashboard* user. The latency calculation is formulated as follows:

$$Latency = T_{recieve} - T_{send}$$

In that equation, the broker or application, and $T_{recieve}$ is a timestamp (timestamp) when the data packet is received by T_{send} is the time when the data packet starts to be published by node sensor.

The results of these measurements are then categorized based on network service quality standards to determine the suitability of the system for monitoring *real-time*.

RESULT

This section presents the experimental results of sensor performance and network behavior characterization under the proposed medical IoT monitoring architecture.

Sensor Performance Evaluation

Testing is performed by comparing the device's readings to standard measuring instruments. The sensor calibration test results are summarized in the following table:

Table 1. HX711 Load Cell Sensor Accuracy Test Results

Test	Value Listed	Sensor Results	Difference	Accuracy
1.	100 ml	109 ml	+ 9 ml	91.00%
2.	200 ml	212 ml	+ 12 ml	94.00%
3.	300 ml	308 ml	+ 8 ml	97.33%
4.	400 ml	413 ml	+ 13 ml	96.75%
5.	500 ml	510 ml	+ 10 ml	98.00%
Average			+ 10.4 ml	95.42%

Table 2. Infrared Sensor Accuracy Test Results

Test	Original Value	Sensor Results	Difference	Accuracy

*name of corresponding author

1.	23 drops	26 drops	6 drops	70.00%
2.	27 drops	32 drops	7 drops	72.00%
3.	30 drops	38 drops	8 drops	73.33%
4.	35 drops	45 drops	10 drops	71.43%
5.	40 drops	52 drops	12 drops	70.00%
6.	45 drops	59 drops	14 drops	68.89%
7.	50 drops	66 drops	16 drops	68.00%
8.	55 drops	70 drops	15 drops	72.73%
9.	60 drops	76 drops	16 drops	73.33%
10.	65 drops	81 drops	16 drops	75.38%
Average				71.41%

There is a significant difference in accuracy between the load cell and the infrared sensor. Meanwhile, the infrared sensor's accuracy for detecting infusion drops, which reached 71.41%, is affected by light refraction on the infusion tube wall and the inconsistent droplet velocity. The sensor's position, which is not perpendicular to the droplet's fall, causes signal interruptions to be inconsistently read by the microcontroller.

Table 3. Pulse Oximeter Sensor Accuracy Test Results

Test	Original Value (BPM)	Sensor Results (BPM)	Difference (BPM)	Accuracy (%)
1.	84	0	84	0.00
2.	75	108	33	56.00
3.	80	79	1	98.75
4.	72	71	1	98.61
5.	78	76	2	97.44
6.	85	83	2	97.65
7.	69	68	1	98.55
8.	70	40	30	57.14
9.	88	0	88	0.00
10.	90	93	3	96.67
11.	74	75	1	98.65
12.	77	77	0	100.00
13.	82	82	0	100.00
14.	85	127	42	50.59
15.	73	72	1	98.63
16.	76	75	1	98.68
17.	70	68	2	97.14
18.	83	50	33	60.24
19.	91	89	2	97.80
20.	68	70	2	97.06

21.	77	77	0	100.00
22.	84	83	1	98.81
23.	80	79	1	98.75
24.	90	60	30	66.67
25.	75	74	1	98.67
Average				80.64%

Based on each of the tables above, the average percentage of accuracy level on the HX711 Load Cell sensor (infusion fluid weight) is 95.42%, while for the infrared sensor (infusion drops per minute) it is 71.41%, and the Pulse Oximeter sensor (heart rate) is 80.64%.

Further testing was performed on the sensor *Pulse Oximeter* to determine the level of accuracy of reading heart rate data via the sensor by conducting tests on 10 different respondents with the test results included in Table 4 below.

Table 4. Success Rate of Sensor Detection in Several Respondents

Patient Initials	Sensor Value (BPM)	Success
SF	79	Succeed
ETM	0	Fail
MAF	86	Succeed
ZA	92	Succeed
MM	0	Fail
RM	82	Succeed
AAW	0	Fail
MR	0	Fail
RA	0	Fail
MF	76	Succeed

Based on Table 4, the heart rate monitoring system achieved a 50% success rate, where only 5 out of 10 respondents produced stable heart rate readings. Several trials resulted in zero BPM values or highly fluctuating measurements, indicating inconsistent signal acquisition. These results show that although the MAX30102 sensor can provide accurate readings under certain conditions, its operational reliability remains unstable.

ESP-Mesh Performance Characterization

This test measures the network's ability to perform route recovery (*self-healing*) and signal strength (RSSI) according to the provisions in equation (2).

Table 5. Results of Network Self-Healing Testing

No	Node Distance (Meters)	RSSI (dBm)	Time Sent	Time Accepted	Latency (s)
1	1 Meter	- 50	10:12:08.417	10:12:08.927	0.51
2		- 52	10:12:22.677	10:12:23.132	0.455
3		- 55	10:12:36.409	10:12:36.869	0.46
4		- 51	10:12:50.959	10:12:51.547	0.588
5		- 53	10:13:03.734	10:13:04.210	0.476
6	2 Meter	- 56	10:13:16.555	10:13:17.207	0.652
7		- 59	10:13:28.285	10:13:28.960	0.675
8		- 57	10:13:40.959	10:13:41.510	0.551
9		- 61	10:13:53.189	10:13:53.818	0.629
10		- 60	10:14:04.645	10:14:05.217	0.572

11	3 Meter	- 63	10:14:16.173	10:14:16.794	0.621
12		- 66	10:14:27.604	10:14:28.311	0.707
13		- 64	10:14:39.386	10:14:40.036	0.65
14		- 62	10:14:52.890	10:14:53.552	0.662
15		- 65	10:15:04.567	10:15:05.398	0.831
16	4 Meter	- 69	10:15:16.313	10:15:17.126	0.813
17		- 70	10:15:29.834	10:15:30.743	0.909
18		- 68	10:15:43.492	10:15:44.282	0.79
19		- 72	10:15:57.136	10:15:57.925	0.789
20		- 71	10:16:10.757	10:16:11.527	0.77
21	5 Meter	- 74	10:16:23.242	10:16:24.248	1,006
22		- 77	10:16:38.084	10:16:39.010	0.926
23		- 76	10:16:51.832	10:16:52.717	0.885
24		- 78	10:17:05.256	10:17:06.272	1,016
25		- 75	10:17:18.779	10:17:19.679	0.9
Average					0.714 s

This section experimentally characterizes the empirical behavior of ESP-Mesh communication under indoor medical-like conditions, focusing on RSSI degradation, multi-hop latency, and transmission stability.

RSSI degradation profile.
Based on Table 5, RSSI consistently decreases as transmission distance increases. At 1 m, RSSI remains in the strong-signal range (approximately −50 to −55 dBm), while at 5 m it degrades to weak-signal conditions (approximately −74 to −78 dBm). This pattern confirms the influence of indoor attenuation and obstacles on ESP-Mesh link quality.

Latency distribution across distance.
Across all measurements, ESP-Mesh achieves an average end-to-end multi-hop latency of 0.714 s. Latency increases monotonically with distance, rising from approximately 0.45–0.59 s at 1 m to 0.88–1.02 s at 5 m. Latency variability becomes more pronounced under weaker RSSI conditions, indicating less stable transmission timing.

Empirical relationship between RSSI and latency.
A clear empirical relationship is observed between RSSI and latency. Under strong signal conditions (−50 to −55 dBm), latency remains below 0.60 s. Under moderate conditions (−56 to −65 dBm), latency increases to 0.55–0.83 s. When RSSI degrades to weak levels (≤ −66 dBm), latency rises significantly to 0.70–1.01 s. This trend suggests that RSSI degradation is a practical indicator of ESP-Mesh timeliness in real-time monitoring scenarios.

Summary of ESP-Mesh characterization.
Overall, ESP-Mesh maintains self-healing connectivity across all tested indoor distances. However, a measurable resilience–timeliness trade-off is observed, where increased network robustness through multi-hop routing is accompanied by higher transmission latency as signal quality decreases.

Performance Characterization of MQTT Transmission

Latency data was captured using Wireshark during 50 data transmissions from the node to the dashboard according to the provisions in equation (3).

Table 6. MQTT QoS Parameters

No	Sent ***Client*** (s)	Response ***Server*** (s)	Latency (s)
1	55.435006	55.715292	0.280286

2	65.431495	65.744807	0.313312
3	75.431892	75.650944	0.219052
4	85.432376	85.651589	0.219213
5	95.43205	95.652774	0.220724
6	105.432331	105.683516	0.251185
7	115.433138	115.639631	0.206493
8	125.433343	125.648569	0.215226
9	135.434116	135.645386	0.21127
10	145.434152	145.639805	0.205653
11	155.43492	155.651386	0.216466
12	165.434823	165.789557	0.354734
13	175.435151	175.72172	0.286569
14	185.435617	185.656496	0.220879
15	195.435948	195.690865	0.254917
16	205.436251	205.642327	0.206076
17	215.436541	215.679212	0.242671
18	225.437357	225.693422	0.256065
19	235.438029	235.732995	0.294966
20	245.438006	245.661977	0.223971
21	255.43794	255.643846	0.205906
22	265.438608	265.645884	0.207276
23	275.438766	275.648967	0.210201
24	285.439094	285.700608	0.261514
25	295.439355	295.736206	0.296851
26	305.439804	305.678496	0.238692
27	315.440325	315.647295	0.20697
28	325.440577	325.682015	0.241438
29	335.440959	335.650778	0.209819
30	345.441739	345.688399	0.24666
31	355.441585	355.748175	0.30659
32	365.442338	365.67333	0.230992
33	375.442303	375.657369	0.215066
34	385.442864	385.654252	0.211388
35	395.443147	395.649739	0.206592
36	405.443395	405.729065	0.28567
37	415.444788	415.781437	0.336649
38	425.444924	425.652276	0.207352
39	435.445043	435.653857	0.208814

40	445.445373	445.659679	0.214306
41	455.445171	455.689086	0.243915
42	465.446247	465.720911	0.274664
43	475.446064	475.757305	0.311241
44	485.446293	485.68956	0.243267
45	495.446829	495.724872	0.278043
46	505.447735	505.793124	0.345389
47	515.448199	515.700711	0.252512
48	525.448056	525.65458	0.206524
49	535.44847	535.763154	0.314684
50	545.448732	545.696036	0.247304
Average			0.24732034 s

Table 6 presents 50 latency data samples measured from the data communication process between *client* And *server* using the MQTT protocol *over* TLS on *Hive MQ Cloud platform*. Each entry records the time when the data was sent by *client*, time when *response* received back, as well as the latency value which is the difference between the two in seconds (s). The latency data is displayed in the graph in Figure 2.

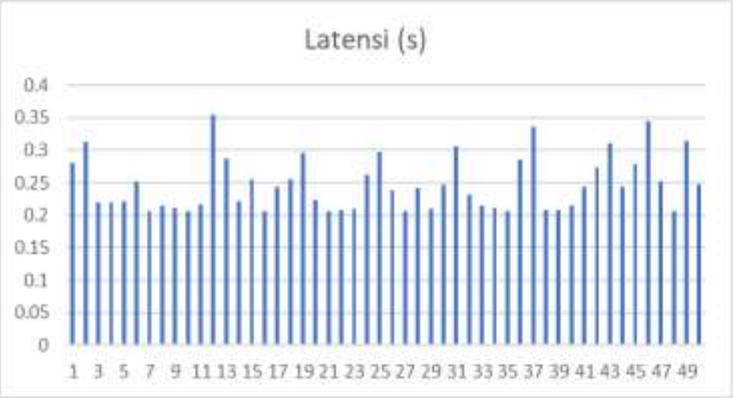

Fig 2. Graph of data delivery latency to *server*

Figure 2 shows a graph of data transmission latency from *client* to *server* in units of seconds (s) for 50 measurements. Latency data is calculated based on the time difference between when data is sent by *client* and when *response* received from *server*. Based on Table 5, the latency values varied between approximately 0.20 s and 0.35 s, with an average of 0.247 s. This variation in latency values is still within a relatively small and stable range, indicating that the data communication process takes place consistently without significant disruption. The highest latency peaks occurred in the 12th and 46th measurements, at 0.354 s and 0.345 s, respectively, which were likely caused by momentary network fluctuations or processing load. *server*.

Overall System Performance Summary

To provide an overall evaluation of the proposed medical IoT monitoring system, Table 7 summarizes the comparison between the expected performance targets and the experimentally obtained results.

Tabel 7. System Performance Comparison

Parameters	**Initial Target (Expected)**	**Test Results (Actual)**	**Status**
Load Cell Accuracy	> 99%	95.42%	Approaching
Approaching Pulse Oximeter Accuracy	> 95%	80.64%	Below Target

HR Success Rate	100%	50%	Needs Improvement
MQTT Latency	< 200 ms	247 ms (0.247 s)	Approaching
ESP-Mesh Latency	< 500 ms	714 ms (0.714 s)	Beyond Target
Data Loss	0%	0%	**Achieved**

Based on the comparison between the initial performance targets and the experimental results (Table 7), gaps are observed in heart rate accuracy and ESP-Mesh latency. The system achieved an average ESP-Mesh latency of 0.714 s, exceeding the initial target of 0.5 s. In addition, heart rate monitoring achieved an average accuracy of 80.64%, which did not meet the expected reliability level. Despite these gaps, the system demonstrated excellent network resilience, achieving a packet loss rate of 0%

DISCUSSIONS

The experimental results provide empirical insights into the behavior of ESP-Mesh networks when deployed in indoor medical-like environments. The observed RSSI degradation across increasing distances confirms that ESP-Mesh communication is strongly influenced by indoor attenuation and obstacles, which is typical in hospital-like settings. More importantly, the results reveal a clear and systematic relationship between signal strength and multi-hop latency.

The increasing latency trend as RSSI decreases indicates that ESP-Mesh exhibits a measurable resilience–timeliness trade-off. While the self-healing mesh architecture successfully maintains connectivity across all tested distances, weaker signal conditions introduce additional routing overhead, retransmissions, and processing delays, resulting in longer end-to-end transmission times. This finding demonstrates that ESP-Mesh is capable of sustaining network resilience, but timeliness becomes increasingly constrained as link quality deteriorates.

The stable MQTT transmission performance, characterized by low average latency and zero packet loss, indicates that cloud-level communication is not a bottleneck in the proposed architecture. Instead, the dominant performance limitations are primarily associated with local mesh network behavior. This highlights that in medical IoT deployments, network design efforts should prioritize robust local connectivity and multi-hop performance optimization rather than cloud communication efficiency alone.

From the sensing perspective, the significant performance gap between infusion volume measurement and heart rate acquisition indicates that system reliability in medical IoT is jointly constrained by both communication networks and biomedical signal quality. While the load cell sensor demonstrates high stability, the inconsistent heart rate detection emphasizes the vulnerability of photoplethysmography-based sensors to environmental interference and motion artifacts. This reinforces the necessity of combining resilient network architectures with advanced signal processing to achieve clinically reliable monitoring systems.

These findings are consistent with Khan et al. (2022), who reported that ESP-Mesh networks provide robust connectivity under indoor and obstructed environments, but experience increased transmission delay as network complexity grows. Similarly, Irawan et al. (2024) highlighted that mesh-based architectures improve coverage and fault tolerance in hospital monitoring systems, although they did not explicitly analyze the latency behavior associated with self-healing operations.

Unlike most previous medical IoT studies that primarily focus on system implementation and data delivery feasibility (Syauqy & Primananda, 2019; Maesyarani et al., 2024), this study experimentally characterizes the relationship between RSSI degradation and multi-hop latency in ESP-Mesh networks. This provides empirical evidence of performance trade-offs that have not been sufficiently addressed in earlier medical monitoring research.

The results highlight that in medical IoT deployments, local network architecture plays a more critical role than cloud transmission efficiency. Although MQTT demonstrated stable low-latency performance, the dominant performance constraints emerged at the mesh-network layer. This implies that the design of resilient medical monitoring systems should prioritize local multi-hop optimization, node placement strategies, and adaptive routing mechanisms.

Furthermore, the substantial performance disparity between infusion volume sensing and heart rate acquisition indicates that reliable medical IoT systems are constrained not only by network robustness but also by biomedical signal quality. Therefore, resilient communication backbones such as ESP-Mesh must be complemented with advanced signal processing techniques to ensure clinically reliable sensing..

Despite the promising results, this study has several limitations. First, the experimental deployment involved a limited number of nodes and relatively short indoor distances, which may not fully represent large-scale hospital environments. Second, the heart rate sensor performance was significantly affected by motion artifacts and environmental interference, indicating that the sensing subsystem is not yet suitable for clinical-grade deployment. Third, the evaluation focused primarily on latency and RSSI, without assessing other critical network metrics such as throughput, jitter, or long-term stability.

Future research should investigate ESP-Mesh behavior under denser network topologies, extended deployment durations, and dynamic mobility conditions. Additionally, integrating advanced signal processing and machine learning-based noise reduction techniques is necessary to enhance the reliability of biomedical sensing and move toward clinically deployable medical IoT systems.

CONCLUSION

Based on the performance analysis results, it can be concluded that this research successfully implemented a multi-parameter monitoring system for patients using the ESP-Mesh network and MQTT protocol. The integration of ESP-Mesh provides a robust communication backbone, achieving a stable average latency of 0.714 s and ensuring data continuity even in complex indoor environments. The MQTT protocol facilitated efficient cloud data distribution with an average latency of 0.247 s and no data loss during testing. In terms of hardware performance, the Load Cell sensor proved highly reliable for infusion volume monitoring with 95.42% accuracy. However, significant limitations were identified in the drip rate (71.41%) and heart rate (80.64%) measurements. The Pulse Oximeter, in particular, faced a 50% failure rate during respondent testing, primarily due to signal interference and sensor placement sensitivity. In conclusion, while the system's network architecture is highly feasible for resilient medical monitoring, the heart rate and drip rate modules require advanced signal processing and better sensor shielding before being considered for clinical implementation.

REFERENCES

Abilovani, Z. B., Yahya, W., & Bakhtiar, F. A. (2018). Implementasi Protokol MQTT Untuk Sistem Monitoring Perangkat IoT. Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya, 2(12), 7521–7527.

Ananta, A., Nasir, M., & Erdiansyah, U. (2024). *Smart Infusion Digitalization Based on IoT, Long-Range Communication, and Cloud. Journal of Artificial Intelligence and Software Engineering*, 4(2), 123–134.

Aprilia, A., & Sollu, T. S. (2021). SISTEM MONITORING *REALTIME* DETAK JANTUNG DAN KADAR OKSIGEN DALAM DARAH PADA MANUSIA BERBASIS IoT (*INTERNET of THINGS*). Foristek, 10(2), 95–103. https://doi.org/10.54757/fs.v10i2.43

Azhari, R. Y. (2022). *Technology and Informatics Insight Journal Web Service Framework: flask* dan fastAPI. *Technology and Informatics Insight Journal*, 1(1).

Imansyah, F., Ratiandi, R., Marpaung, J., Suryadi, D., & Hizballah, F. (2023). Perancangan Sistem Pengujian *Throughput Publishing* Data pada Modul ESP8266 dengan Protokol MQTT. JUSTIN (Jurnal Sistem dan Teknologi Informasi), 11(3), 419–424.

Irawan, R. F., Hidayah, A. K., & Abdullah, D. (2024). Sistem Perancangan Monitoring Inravena (Infus) Berbasis *Internet Of Things*. Jurnal Media Infotama, 20(1), 304–310.

Jacobus, A., Lendo, R., & Mapaly, H. A. (2023). *Design and Development of Mobile-Based Digital Library Application Using Flutter*. Jurnal Teknik Informatika, 18(1), 353–362. https://doi.org/10.35793/jti.v18i1.50490

Khan, A. U., Khan, M. E., Hasan, M., Zakri, W., Alhazmi, W., & Islam, T. (2022). *An Efficient Wireless Sensor Network Based on the ESP-MESH Protocol for Indoor and Outdoor Air Quality Monitoring. Sustainability*, 14(24), 16630. https://doi.org/10.3390/su142416630

Lani, M. F. N., & Da Ate, Y. F. (2025). Implementasi Standar Prosedur Operasional (SPO) Pemberian Obat Intravena Melalui Selang Infus Di Ruang Dahlia BLUD RSUD dr. T.C. Hillers Maumere. Jurnal Keperawatan Duta Medika, 5(1), 23–32.

Maesyarani, A. A., Samjantungsumar, L. D., Zaenudin, Z., Akbar, A., & Suryadi, E. (2024). SISTEM MONITORING PENGUKUR DETAK JANTUNG DAN OKSIGEN DALAM DARAH BERBASIS INTERNET OF THINGS (IoT). *Journal of Computer Science and Information Technology*, 1(4), 341–350.

Maulana Putra, J., Misbahuddin, & Al Sasongko, S. M. (2022). Perancangan Smart Lamp Berbasis *Mesh Network* Menggunakan Protokol Komunikasi Mqtt Pada Esp8266 [Skripsi]. Universitas Mataram.

Pratama, R. N. W., Widagda, M. E. P., & Hadiyanto, H. (2024). Sistem monitoring cerdas ruang kelas berbasis *internet of thing* (IoT) dengan menggunakan flutter. Jurnal Hasil Inovasi Masyarakat, 1(1), 23–30. https://doi.org/10.70310/axt1q513

Ramadhan Pradana, D., Sari, M. I., & Rimasa, D. (2023). Monitoring Denyut Nadi Dan Suhu Tubuh Menggunakan *Pulse Sensor* Dan Sensor Suhu Pada Atlet Anggar Koni Kota Bandung. e-Proceeding of Applied Science, 9(6), 3042.

Raven, Y., & Mulyana, T. M. S. (2022). *Implementasi TCP/IP untuk Membuat Server Database Access*. Jurnal Teknologi Informasi, Universitas Bunda Mulia.

Rizki, A., Muhammad, M., & Nasri, N. (2021). Analisis *Quality Of Service* (QoS) Layanan Jaringan Data Internet Pada Game Online. JURNAL TEKTRO, 5(1), 43.

Syauqy, D., & Primananda, R. (2019). Implementasi Protokol MQTT (*Message Queuing Telemetry Transport*) Untuk Monitoring Infus Pasien Secara Terpusat. Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer, 3(2), 1366–1371.

Taufiq, A., Pratama, M., & Pratama, A. R. (2021). Rancang Bangun Aplikasi Android "Kuliah Apa?" Berbasis Flutter dan *TensorFlow Lite*. AUTOMATA, 2(1), 1-8.

Ubaedila, I., Nurdiawan, O., Wijaya, Y. A., & Sidik, J. (2021). Layanan Jaringan Menggunakan Metode *Sniffing*

Berbasis *Wireshark*. *Informatics For Educators And Professional,* 6(1), 95–104. https://doi.org/10.51211/itbi.v6i1.1697

Wijayanto, C., & Susetyo, Y. A. (2022). Implementasi *Flask Framework* Pada Pembangunan Aplikasi Sistem Informasi *Helpdesk* (SIH). JIPI (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika), 7(3), 858–868. https://doi.org/10.29100/jipi.v7i3.3161