Type of Contribution:
Research Paper
Review Paper
Case Study

ENERGY: JURNAL ILMIAH ILMU-ILMU TEKNIK
Vol. 16, No. 1 (2026) pp 159-168
DOI: 10.51747/energy.v16i1.16110

E-ISSN: 2962-2565

This article contributes to:

Development and Functional Evaluation of a Web-Based Medical Equipment Maintenance Reminder System Integrated with Telegram Notifications

Mukhamad Firdaus Ferdiansyah[1]*, Rahma Ajeng Puspitasari[1]

[1] Diploma 3 of Electromedical Technology Study Program, Kadiri University, 64115, **Indonesia**

*firdausfrdnsh@gmail.com

Abstract

Proper maintenance of medical equipment is essential for ensuring patient safety and clinical operational continuity, yet traditional manual documentation often leads to missed maintenance schedules. This research aims to develop and functionally evaluate a web-based Computerized Maintenance Management System (CMMS) integrated with Telegram notifications. The system was developed using the Waterfall software development model, utilizing XAMPP (PHP and MySQL) for backend management and the Telegram Bot API via webhook for automated alert delivery. The functional evaluation was conducted in a prototype environment using black-box testing and API response time analysis. Results demonstrated that the system successfully automates maintenance scheduling and delivers real-time notifications with a 100% success rate across all functional test cases. While this study confirms the technical reliability of the prototype in a controlled environment, further clinical deployment is required to measure its impact on operational efficiency. It is concluded that this digital solution provides a functional and reliable baseline for improving medical equipment management in healthcare facilities.

Keywords: Medical Equipment Management, Preventive Maintenance, Computerized Maintenance Management System, Healthcare Technology Management, Telegram Bot, Web-Based Application

Article Info
Submitted: 2026-05-1
Revised: 2026-06-15
Accepted: 2026-06-25
Published: 2026-06-27

Publisher
Universitas Panca Marga

1. Introduction

The management of medical equipment maintenance is a critical pillar in modern healthcare ecosystems, essential for ensuring patient safety and clinical operational continuity. In accordance with international standards, such as IEC 62353 [1], medical devices require periodic preventive maintenance repairs, and rigorous testing to maintain accurate performance and prevent unexpected device failure. Failure to comply with maintenance schedules not only risks the accuracy of diagnoses but also jeopardizes the safety of both patients and medical staff.

In practice, many healthcare facilities still rely on manual recording techniques, such as paper forms and separate Excel files. This strategy produces various challenges, including data inconsistency, difficulty in tracking equipment history, and inefficiencies in reporting processes. A web-based system offers a centralized solution that permits uniform data recording, faster data retrieval, and automatic documentation of maintenance history [2], [3].

However, practical implementation also often faces significant operational challenges. Many healthcare facilities continue to rely on manual documentation systems or conventional spreadsheets [4]. This approach is highly susceptible to human error, the loss of historical maintenance data, and a lack of real-time monitoring of equipment status. The limitations of these conventional systems often lead to missed maintenance schedules and hinder managerial decision-making regarding medical asset management.

With the rapid digital transformation in the health sector, the development of a Computerized Maintenance Management System (CMMS) has become an urgent necessity [5]. A web-based system offers significant advantages in terms of centralized data accessibility, scalability, and streamlined reporting. Furthermore, integration with instant messaging applications, such as Telegram, enables automated push notifications that can be directly received by electromedical technicians, facilitating faster and more efficient responses to maintenance schedules.

This research aims to design and implement a web-based medical equipment maintenance scheduling system using the XAMPP technology stack (Apache, PHP, and MySQL) [6]. By integrating the Telegram Bot API [7], this system is designed to automate maintenance scheduling reminders, provide ease of data access for technicians, and improve the overall efficiency of medical equipment maintenance workflows in clinical environments [9].

Furthermore, while several notification platforms exist, Telegram was specifically selected for this system due to its open and highly responsive Bot API, lightweight data consumption, and widespread adoption among technicians [7].

Compared to traditional email alerts, which are often delayed or ignored, or SMS messages that incur recurring costs, Telegram provides instantaneous, cost-effective push notifications directly to the technicians' mobile devices, making it highly suitable for fast-paced clinical environments.

2. Methods

This applied research employed the Waterfall Software Development Life Cycle (SDLC) model consisting of requirement analysis, system design, implementation, testing, and evaluation stages. The study focused on the development and functional validation of a web-based Computerized Maintenance Management System (CMMS) integrated with Telegram notifications for medical equipment maintenance management.

2.1 Requirement Analysis and User Roles

The system requirements were identified through observation of conventional maintenance documentation practices and analysis of electromedical technicians' workflow. The proposed system was designed to support three main maintenance activities, namely preventive maintenance, corrective maintenance, and overhaul maintenance.

Two user roles were defined within the system:

a. **Administrator**, responsible for managing user accounts, equipment inventory records, maintenance schedules, and report generation.
b. **Electromedical Technician**, responsible for recording maintenance activities, updating maintenance status, uploading maintenance documentation, and responding to Telegram notifications.

The functional requirements included user authentication, maintenance data entry, maintenance schedule calculation, Telegram notification delivery, maintenance status updates, maintenance history storage, and Excel report generation.

In addition to functional requirements, several non-functional requirements were considered, including data integrity, user accessibility through web browsers, response time performance, notification reliability, and compatibility with standard local-area network environments.

2.2 System Architecture and Database Design

The system was developed using XAMPP as the local server environment, consisting of Apache Web Server, PHP, and MySQL database management system. The application architecture follows a client–server model in which users access the web interface through standard browsers, while maintenance records and scheduling information are processed and stored within the database server.

The database design was modeled using an Entity Relationship Diagram (ERD) to represent the relationships among users, medical equipment, maintenance activities, maintenance history, and component estimation records. The primary entities include Users, Equipment, Maintenance Records, Maintenance History, and Component Estimation tables. These entities collectively support maintenance scheduling, maintenance status tracking, notification triggering, and historical record management. **Figure 1** presents the Entity Relationship Diagram of the developed system.

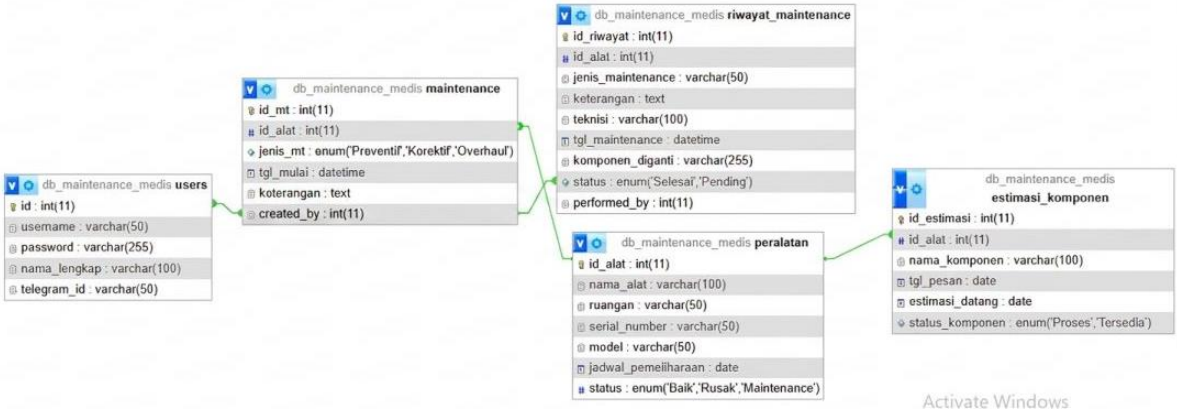

Figure 1. Entity-Relationship Diagram (ERD) of the System (Authors' own work)

2.3 Workflow and Telegram Integration

The system workflow begins with user authentication using a username and password. After successful login, users select the maintenance category (preventive, corrective, or overhaul) and complete a standardized maintenance form containing equipment identification, serial number, location, maintenance description, technician information, maintenance status, and supporting documentation.

The maintenance scheduling algorithm automatically calculates the next maintenance due date based on predefined maintenance frequencies. A scheduled background process (cron job) continuously compares the current server date with stored maintenance schedules.

When a maintenance schedule reaches its due date, the system automatically generates a notification through the Telegram Bot API and sends the notification to the assigned technician or maintenance group. The notification includes equipment information, maintenance category, scheduled date, and maintenance status.

For maintenance activities marked as "Pending" or not completed before the scheduled deadline, reminder notifications are repeatedly generated during subsequent execution cycles until the maintenance status is updated. In the event of temporary network interruption or Telegram API failure, the notification request is recorded and reattempted during the next scheduled process execution. **Figure 2** illustrates the complete workflow of the maintenance scheduling and Telegram notification mechanism.

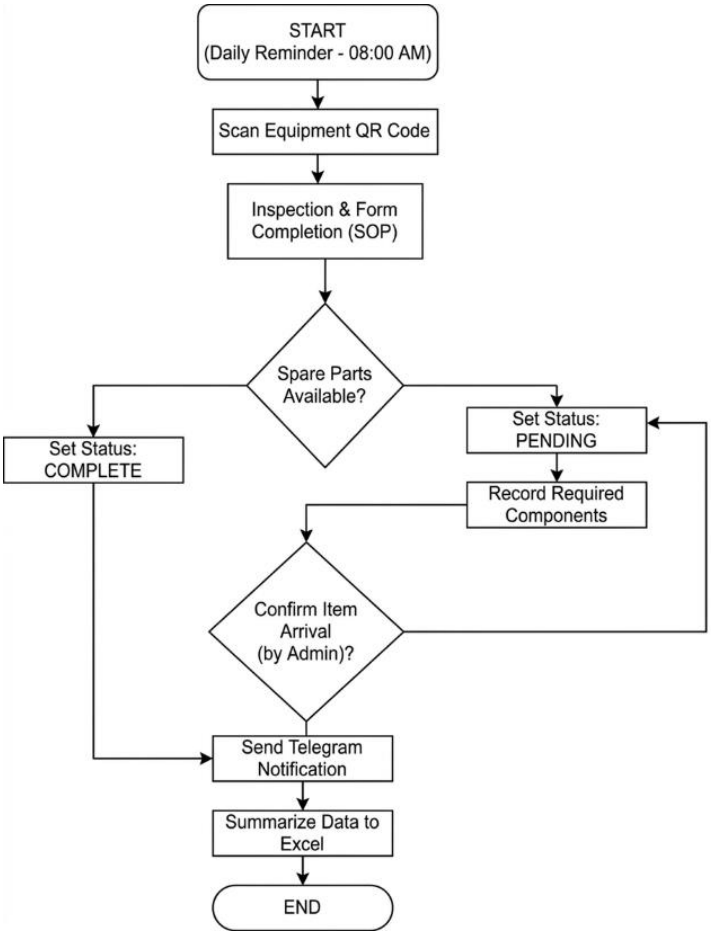

Figure 2. The Complete Workflow of The Maintenance Scheduling and Telegram Notification Mechanism (Authors' Own Work)

2.4 Testing Environment and Evaluation Procedure

System evaluation was conducted in a controlled prototype environment using black-box testing. The testing environment consisted of a laptop running Microsoft Windows operating system with XAMPP, PHP, MySQL, Google Chrome browser, Telegram application, and a stable internet connection.

The evaluation focused on verifying:

a. User authentication functionality.
b. Medical equipment data management (Create, Read, Update, Delete).
c. Maintenance schedule calculation.
d. Telegram notification delivery.
e. Data synchronization and report export functionality.

Each test scenario was executed five times under identical operating conditions to evaluate consistency and reliability. Functional validation was considered successful when the actual system output matched the expected output without generating operational errors.

In addition to functional validation, quantitative performance measurements were collected by recording processing times for several critical operations,

including login authentication, maintenance data entry, image upload, Telegram notification delivery, and report export generation. Response time measurements were performed using a digital stopwatch and repeated trials to obtain representative values.

The testing was conducted under stable network conditions using a local server environment connected to the internet to ensure successful communication with the Telegram Bot API.

3.Results and Discussion

3.1 Functional Evaluation

The functional evaluation was conducted using the black-box testing method to verify whether each system module operated according to the predefined functional requirements. The evaluation focused on validating user authentication, maintenance data management, maintenance scheduling, Telegram notification delivery, and report generation functionalities. **Tabel 1** presents the results of the functional testing performed on the developed web-based maintenance management system.

Tabel 1. Black-Box Testing Results

No.	Test Case	Expected Outcome	Actual Result	Status
1	Admin Login with valid credentials	Access granted to Dashboard	As expected	Success
2	Admin Login with invalid credentials	Access denied, error message displayed	As expected	Success
3	Add new medical equipment data	Data saved to MySQL database	As expected	Success
4	Calculate maintenance due date	System accurately sets the next date	As expected	Success
5	Trigger Telegram Webhook	Notification received on Telegram app	As expected	Success

The results indicate that all functional modules operated correctly and achieved a 100% success rate. Authentication mechanisms successfully distinguished valid and invalid login attempts. The database management module stored maintenance records without data loss, while the maintenance scheduling module correctly generated future maintenance dates according to predefined maintenance intervals. Furthermore, Telegram notifications were delivered successfully in all testing scenarios, confirming the reliability of the communication subsystem.

3.2 System Performance Evaluation

In addition to functional validation, system performance was evaluated by measuring the response time of critical operations. The objective was to assess whether the developed system could provide adequate responsiveness for routine

maintenance management activities. Each testing scenario was executed five times under identical operating conditions, and the average response time was calculated. The system performance evaluation shown in **Tabel 2**.

Tabel 2. Functional and Performance Testing Results

Module	Test Scenario	Trial (n)	Avg. Response Time (s)	Expected Result	Actual Result	Status
Login	Valid credentials	5	0.022	Dashboard displayed	Successful	Pass
Data Entry	Save maintenance record	5	0.093	Data stored	Successful	Pass
Upload Photo	Upload image	5	0.030	Image stored	Successful	Pass
Telegram Notification	Send reminder	5	0.068	Notification received	Successful	Pass
Status Synchronization	Update status	5	0.041	Database updated	Successful	Pass
Excel Export	Generate report	5	0.054	Report generated	Successful	Pass

The results demonstrate that all system operations were completed in less than 0.1 seconds on average. Login authentication exhibited the fastest response time at 0.022 seconds, while maintenance data entry required 0.093 seconds due to database processing and validation procedures. Telegram notification delivery was completed within 0.068 seconds, indicating that the notification subsystem is capable of supporting near real-time maintenance reminders.

The relatively low response times observed across all tested operations suggest that the proposed system can support routine maintenance documentation activities without causing significant delays to electromedical technicians.

3.3 Discussion

The results demonstrate that the developed web-based Computerized Maintenance Management System (CMMS) successfully fulfilled all predefined functional requirements. The integration of maintenance scheduling, centralized database storage, and Telegram-based notifications enabled the automation of several maintenance management activities that are traditionally performed manually [5], [10].

One of the main contributions of the proposed system is the implementation of an automated notification mechanism through the Telegram Bot API. Unlike conventional paper-based maintenance documentation systems, which require technicians to manually review maintenance schedules, the developed system proactively sends maintenance reminders directly to technicians. This approach may

reduce the risk of missed preventive maintenance activities and improve maintenance traceability [5].

The findings are consistent with previous studies that reported the benefits of web-based maintenance information systems in improving maintenance documentation and record accessibility [2], [10]. Fuaddi and Sabarguna reported that centralized maintenance databases facilitate maintenance tracking and reporting processes [10]. Similarly, Firdaus et al. demonstrated that digital maintenance management systems improve data organization and reduce administrative burdens associated with paper-based documentation [2].

Compared with previous web-based maintenance systems, the present study extends functionality by integrating Telegram-based notifications and maintenance status synchronization [2], [10]. This integration provides a lightweight and cost-effective communication channel that can support faster technician responses while minimizing operational complexity.

The implementation of Telegram notifications also offers practical advantages over conventional communication approaches such as email reminders [11]. Telegram notifications are delivered directly to mobile devices and support interactive communication features, allowing technicians to update maintenance statuses more efficiently [11]. This capability aligns with current Healthcare Technology Management (HTM) recommendations emphasizing timely preventive maintenance and effective maintenance tracking [5]. Furthermore, the developed system supports the documentation requirements recommended by IEC 62353 through structured maintenance records, maintenance histories, and traceable technician activities [1].

Despite the positive results obtained during prototype testing, several limitations should be acknowledged. The evaluation was conducted in a controlled environment and did not involve deployment within an operational healthcare facility. Consequently, the direct impact of the system on maintenance compliance rates, equipment downtime, Mean Time to Repair (MTTR), and technician productivity was not quantitatively measured.

Furthermore, the system remains dependent on internet connectivity, Telegram API availability, and local server reliability. Security testing, scalability assessment, and multi-user stress testing were beyond the scope of this study. Future research should therefore include User Acceptance Testing (UAT), cybersecurity evaluation, scalability analysis, and long-term implementation studies in clinical environments to assess the operational effectiveness of the proposed system more comprehensively.

4. Conclusion

This study successfully developed and functionally evaluated a web-based Medical Equipment Maintenance Reminder System. By integrating a local CMMS (PHP/MySQL) with the Telegram Bot API, the system effectively automates maintenance scheduling and delivers real-time notifications. The functional evaluation demonstrated that all system modules operate correctly without errors in a prototype environment. While this system establishes a reliable technical foundation to replace manual documentation, future implementation in actual clinical settings is necessary to fully quantify its impact on hospital operational efficiency and patient safety.

Acknowledgement

The authors would like to acknowledge the support from the Electrical Engineering Laboratory, Universitas Islam Malang, to run the experiments of the research.

Authors' Declaration

Authors' contributions and responsibilities - The authors made substantial contributions to the conception and design of the study. The authors took responsibility for data analysis, interpretation, and discussion of results. The authors read and approved the final manuscript.

Funding - No funding information from the authors.

Availability of data and materials - All data is available from the authors.

Competing interests - The authors declare no competing interest.

Additional information - No additional information from the authors.

References

[1] IEC, IEC 62353:2014, 2014.

[2] A. J. A. Firdaus, D. Pramono, and W. Purnomo, "Pengembangan Sistem Informasi UPT Kalibrasi Dinas Kesehatan Kabupaten Malang Berbasis WEB," JUST-SI, vol. 1, no. 1, Aug. 2020, doi: 10.25126/justsi.v1i1.3.

[3] M. Fuaddi and B. S. Sabarguna, "Web-based maintenance information system in medical equipment management | Request PDF," vol. 2092, no. 1, 2019, doi: 10.1063/1.5096736.

[4] N. Wiyono, Riyanto, and A. S. Rejeki, "Perancangan Sistem Informasi Preventive Maintenance Berbasis Web Pada PT Macroprima Panganutama," 1, vol. 9, no. 2, 2021, doi: 10.58217/ipsikom.v9i2.206.

[5] WHO, Computerized maintenance management system. in WHO Medical device technical series. Geneva: WHO Press, 2011.

[6] A. A. N. Gunawan, I. W. Suharta, and I. B. M. Suryatika, "Metode Penelitian

Terapan di Era Digital: Panduan Praktis untuk Teknologi Informasi," Mutiara Intelektual Indonesia.

[7] W. Adhiwibowo, F. W. Christanto, and A. F. Daru, "Implementasi API Bot Telegram Untuk Sistem Notifikasi Pada The Dude Network Monitoring System," 1, pp. 593–599, Nov. 2021.

[8] Telegram, "Telegram."

[9] D. Dani, "Medical Equipment Inventory Information System in Public Hospitals," bit-Tech, vol. 8, no. 3, pp. 4011–4021, Apr. 2026, doi: 10.32877/bt.v8i3.3771.

[10] M. F. Fuaddi and B. S. Sabarguna, "Web-Based Preventive Maintenance Management for Hospital Medical Devices BT -," in Proceedings of the International Conference on Health Informatics, IEEE, 2019.

[11] R. Andriani and A. Sa'di, "Implementation of Telegram Bot Notification on Network Device Monitoring System," SISTEMASI, vol. 13, no. 1, p. 74, Jan. 2024, doi: 10.32520/stmsi.v13i1.3110.