ESP8266-based generator notification system using the RnD method case study of PT. Telkom Witel Central Sulawesi

Umira Az-Zahra[*], **Danur Wijayanto**

Information Technology, Faculty of Science and Technology, Universitas 'Aisyiyah Yogyakarta, Indonesia
Email:umiraazzahra2002@gmail.com, danurwijayanto@unisayogya.ac.id

Abstract

This research aims to design and implement an automated Telegram notification system to support the preventive maintenance of ESP8266-based generators. The research background is based on the needs of PT. Telkom Witel Central Sulawesi to receive real-time generator operational alerts without continuous manual monitoring.

The research method uses Research and Development (R&D) with the stages of needs analysis, system design, prototype implementation, and testing. The system is designed with ESP8266 as a notification controller, an ACS715 sensor for operation status detection, a ADS1115 module for reading accuracy, and the integration of Google Sheets and Telegram Bot API as an automated messaging medium.

The test results showed that the system managed to send 5 Telegram notifications in a row in a 5.4-minute test session, including "Running Reminder" and "Oil Change" alerts according to operational thresholds. The WiFi connection is stable with a status of "Connected (Reconnects: 1)" and the data is stored in Google Sheets with an HTTP 302 response. The system calculates the generator runtime automatically using the ESP8266's internal timer and sends timely notifications to the operator. The tests were conducted under limited conditions so long-term validation was required to measure the reliability of the system.

Keywords: automatic notifications; genset; IoT; ESP8266; preventive maintenance; telegram bot

1. Introduction

In the current era of digitalization, the reliability of telecommunication infrastructure is a crucial factor to maintain the quality of service for customers. PT. Telkom Witel Central Sulawesi, as the largest service provider in Central Sulawesi, operates various vital devices that require a stable electricity supply. Due to the geographical conditions of the area that are vulnerable to PLN power grid disruptions, the company relies heavily on generator sets (generators) as a backup source(Ilmi et al., 2020)

The main problem faced is the limitation of real-time monitoring and warning systems. Currently, generator monitoring is still carried out manually by field personnel, which can cause several operational risks such as delays in fault detection, suboptimal rotation of generators, delays in routine maintenance schedules, and lack of historical data for performance analysis.

The development of Internet of Things (IoT) technology offers innovative solutions to system monitoring and control problems. With IoT devices projected to exceed 50 billion globally by 2025, the IoT market is showing significant growth (Ramadhoni et al., 2023) ESP8266 has been widely used as a popular WiFi microcontroller in IoT system implementation, including for monitoring generator conditions using various sensors, such as current, temperature, and vibration sensors.

ESP8266 has been widely used as a popular WiFi microcontroller in IoT system implementations, including for monitoring the condition of generators using various sensors, such as current, temperature, and vibration sensors (Zaenal et al., 2024). These microcontrollers are capable of reading the operational data of the generator set and sending it to a cloud-based server for further recording and analysis. In addition, integration with messaging apps like Telegram has proven to be effective for sending real-time notifications, allowing operators to receive important information anytime and from anywhere, as long as they are connected to the internet (Hady Musyafa et al., n.d.)

The importance of preventive maintenance in generator operations cannot be ignored. IoT-based monitoring systems enable the implementation of predictive maintenance to prevent serious damage and reduce downtime. Periodic maintenance, such as scheduled oil changes, is an industry standard to maintain performance and extend the life of a generator (Deka et al., n.d.)(São Paulo et al., 2023).

A similar study has been conducted by the one presenting a comprehensive generator set monitoring system with 5 parameters using Arduino Nano and the Blynk application, achieving success in the test. A different study was conducted by one that emphasizes the importance of monitoring parameters such as vibration, temperature, and working hours to support preventive maintenance using ESP8266 with the integration of AWS EC2, Node-RED, and MQTT Panel IoT applications. (AL ARIFIN, 2020) (Mahfud et al., 2023)

The aspect of electrical energy monitoring is a significant research area developed through a household electrical energy consumption monitoring system using ACS712 current sensors and NodeMCU ESP8266 success rate. Similar research was conducted by developing a real-time electrical current and voltage monitoring application using the MQTT protocol with a C++, PHP, and cloud-based MySQL database backend. (Asrofil Al Banna et al., 2023)

Remote control capabilities are becoming a cr(Yusuf et al., 2025) ucial application of IoT developed in electronic device control systems using the Telegram Bot API with NodeMCU ESP8266.(Permana Putra et al., 2025)

2. Method

This research uses the Research and Development (R&D) method with an experimental approach. The R&D method was chosen because the research aims to develop a technological product in the form of a ESP8266-based generator notification system that can be implemented practically at PT. Stuttgart Stuttgart (Saputra et al., n.d.)

2.1. Needs Analysis

The functional needs of the generator notification monitoring system are based on the operational conditions of PT. Telkom Witel Central Sulawesi includes identifying the main problems in generator operations, namely the limitations of manual monitoring, delays in damage detection, and the absence of a real-time warning system. The system must also be able to monitor the run hours of the generator, provide automatic notifications via Telegram, and record operational data to support preventive maintenance.

2.2. Hardware Used

ESP8266 acts as a control center that regulates the entire process from reading sensor data to sending notifications over WiFi networks. The ACS715 sensor is used to detect the electric current from the generator set so that the conditions of its operational duration can be known. For more accurate reading results, ADS1115 is used as an analog to digital converter module with 16-bit resolution. The entire circuit is powered by a 5V adapter that provides a stable power source, while a breadboard and jumper cable are used to facilitate assembly and testing of the system.

2.3. Software Used

In software, the system uses the Arduino IDE to program ESP8266 to read sensor data, then send the data to the server. The data received by Google Apps Script will be processed and stored in Google Sheets as a generator run hour database. In addition, Apps Script is also used to send notifications automatically through Telegram Bots, so users can get real-time information on generator status and maintenance schedules.

2.4. System Planning

The system is designed with a client-server architecture where ESP8266 act as a client that sends data to Google Apps Script as a server. The system data flow follows the order: Sensor Reading → Data Processing → Data Delivery → Cloud Storage → Notification Services. This architecture was chosen because it provides good scalability and allows generator set monitoring from a single centralized dashboard. (Mphale et al., 2024)

The workflow of the ESP8266-based generator notification system is initiated when the system is run. First, the ACS715 current sensor reads the electric current flowing from the generator set. The

results of this reading are then checked to determine the operational condition of the generator. If the detected current is greater than 0.8A, then the generator is declared to be in a running state. On the other hand, if the detected current is less than 0.8A, then the generator set is considered inoperative and the system returns to sensor reading.

When the generator is on, the system will calculate the operating time and check the reminder interval. If the reminder time is reached, then the system sends a "Running Reminder" notification through the Telegram application. In addition, the system also checks operational hours related to maintenance needs. If the operating hours have reached the specified limit, the system will send maintenance notifications to the operator via Telegram.

3. Results and Discussion

In this section, the results of research/community service are explained. to the community while providing a comprehensive discussion. Results can be presented in figures, graphs, tables, and other forms to facilitate reader understanding. Discussions can be conducted in several sub-chapters.

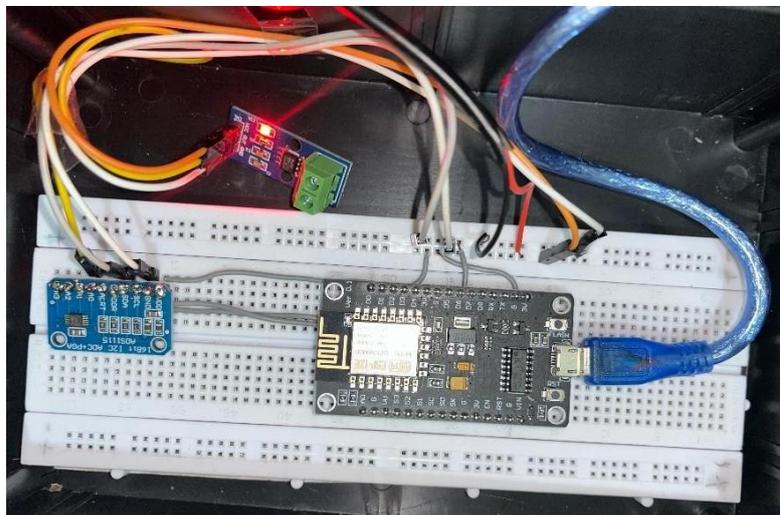### 3.1. Prototype Implementation

Figure 1. Prototype Implementation

Figure 3.1 The implementation of the prototype is carried out by arranging the hardware according to the predetermined design and programming ESP8266 using the Arduino IDE. At this stage, components such as the current sensor, ADC module, power adapter, and breadboard are put together into a single unit. The circuit details with the 5V Adapter serves as the main power source with the positive (+) pole connected to the red rail and the negative (-) pole connected to the blue rail on the breadboard.

The ESP8266 module is connected to the ADS1115 module, where the D1 pin ESP8266 connected to the SCL pin on the ADS1115, while the D2 pin ESP8266 connected to the ADS1115 SDA pin. The G (ground) pin ESP8266 connected to the negative rail, and the Vin pin ESP8266 connected to the positive rail. In the ADS1115 module, the VDD pin is connected to the positive rail, the GND pin to the negative rail, and the ADDR pin is also to the negative rail. Meanwhile, the A0 pin ADS1115 connected to the ACS715 sensor output pin.

3.2. System Testing
3.2.1. WiFi Connectivity Test

Figure 2. Output Source Code Wifi connected

Testing is carried out to ensure that all functions of the ESP8266-based generator notification system run according to design. The module was successfully stably connected to the network in the tests performed, with the status 'Connected (Reconnects: 1)' as seen in the test results of Figure 3.2. This shows that ESP8266 is capable of functioning well as an IoT-based system control center.

Figure 3. Source Code Wifi

Figure 3.3 The program has a retry system of up to 20 attempts if the connection fails, equipped with visual indicators in the form of dots when trying to connect. The code also has good error handling, giving a success message with an IP address if successful, or a failed message if the connection is unsuccessful. The reconnection tracking system allows network stability monitoring for performance evaluation.

3.2.2. Test Sending Data to Google Sheets

Figure 4. Submit data to Google Sheets

Figure 3.4 The system successfully sends 8 data parameters (date, timestamp, location, generator, runhour, running, current, power) in a URL-encoded format that complies with the HTTP POST standard to Google Sheets. The HTTP 302 response received indicates that the Google Apps Script server successfully received and processed the data to be stored to Google Sheets. The data sent will be automatically saved in Google Sheets and also enable the sending of notifications via Telegram Bot.

3.2.3. Telegram Notification Test

Figure 5. Telegram Notifications

Figure 3.5 of the system successfully sends a "Running Reminder" notification during the generator is operating. Maintenance alert notifications in the form of "Oil Change" are also successfully sent automatically when the run hour reaches a predetermined threshold. The system successfully sends 5 notifications in a row in one test session without experiencing any failures. The test was carried out in stable WiFi network conditions during the duration of the generator operation of 5.4 minutes.

The notification time configuration is set as follows: REMINDER_1: 0.5 minutes (30 seconds), REMINDER_2: 1.5 minutes, REMINDER_3: 3.0 minutes, MAINTENANCE: 4.0 minutes for oil change warnings, and MAX_RUNTIME: 6.0 minutes as the maximum operating time limit. Integration with the Telegram app via the Telegram Bot API proved consistent in initial testing, where the system successfully sent notifications in sequence:

a. Running Reminder (1/3): Delivered at 0.6 minutes of operation
b. Running Reminder (2/3): Delivered in 1.8 minutes of operation
c. Running Reminder (3/3): Delivered at 3.0 minutes of operation
d. Oil Change Maintenance: Delivered at 4.2 min runtime
e. Generator Stopped: Delivered after 5.4 minutes of total operation

The runtime value displayed on the message can be slightly different from the configuration value, for example 0.6 minutes or 1.8 minutes. This is due to the real-time reading of sensor data so that it produces an actual value that is more precise than the predetermined threshold value.

3.3. Evaluation and Revision

The test results show that the ESP8266-based generator notification system has been running as designed, but there are some findings that need to be evaluated and improved. In the connectivity test, the system successfully connected to WiFi and sent notifications on schedule under stable network conditions. However, testing has not been carried out on varied network conditions. To improve system reliability, it is recommended to add an automatic retry mechanism and testing on various network conditions to validate system performance in diverse operational situations.

In the test data submission to Google Sheets, all data was recorded well, but there were storage capacity limitations. Therefore, database-based storage alternatives such as Firebase are recommended for further system development. In the Telegram notification test, messages were successfully sent at intervals, but the format was still simple. Improvements can be made by adding detailed information, such as the start time of operation and total hours worked.

4. Conclusion

The ESP8266-based generator notification system was successfully designed and implemented success in sending 5 consecutive notifications during a 5.4-minute test. The system is able to detect the operational status of the generator using the ACS715 sensor, automatically calculate the runtime, and send "Running Reminder" and "Oil Change" notifications via the Telegram Bot API. The WiFi connection is stable and the data is stored in Google Sheets with an HTTP 302 response. For further development, long-term testing and the addition of GSM backup communication lines are recommended.

References

AL ARIFIN, M. (2020). IOT-BASED GENERATOR MONITORING SYSTEM AT BTS REMBANGAN. *UNIVERSITY OF MUHAMMADIYAH JEMBER.* Retrieved from http://repository.unmuhjember.ac.id/id/eprint/3985

Asrofil Al Banna, H., Muhammad Gafar, V., Darwis, M., Hendrowati, R., Engineering Sciences, F., Raya Headquarters Hankam NoKav, J., Cipayung, K., & East Jakarta, K. (2023). DESIGNING AN IOT-BASED ELECTRICAL ENERGY MONITORING APPLICATION WITH THE MQTT PROTOCOL. In Journal of Informatics & Electronics Engineering) (Vol. 6, Issue 2). Retrieved from http://e-journal.stmiklombok.ac.id/index.php/jireISSN.2620-6900

Deka, M., Supriyono, A. ;, & Sedyono, J. (n.d.). *IOT-BASED MONITORING SYSTEM FOR PREDICTIVE MAINTENANCE GENERATOR SET USING MACHINE LEARNING.*

Ilmi, M., Junaidi, & Hiendro, A. (2020). Techno-Economic Analysis of Generator Set (GENSET) as a Backup Electrical Energy Source at PT. Telekomunikasi Indonesia TBK. Pontianak. *Journal of Electrical Engineering, Energy, and Information Technology (J3EIT)*, *7*(2). doi: https://doi.org/10.26418/j3eit.v7i2.34818

Mahfud, A., & Ariansyah, M. S. (2023). Prototype of Vibration, Temperature and Working Hours Monitoring on IoT-Based Electromotors. *Journal of Citra Widya Edukasi*, *15*(3).

Mphale, O., Gorejena, K. N., & Nojila, O. (2024). The Future of Things: A Comprehensive Overview of Internet of Things History, Definitions, Technologies, Architectures, Communication and beyond. *Journal of Information Systems and Informatics*, *6*(2), 1263–1286. doi: 10.51519/journalisi.v6i2.738

Permana Putra, F., Sabirin, S., & Soetanto, H. (2025). Prototype of Internet of Things-Based Control System Using Telegram with Bot API Method. *Heading Syntax Transformation*, *6*(2).

Romadhoni, H., Priyandoko, G., & Effendy, D. U. (2023). JASEE Journal of Application and Science on Electrical Engineering DESIGN AND BUILD A GENERATOR MONITORING SYSTEM THROUGH TELEGRAM APPLICATION NOTIFICATIONS. *JASEE Journal of Application and Science on Electrical Engineering*, *4*(2), 33–43. doi: 10.31328/jasee

Saputra, S., Oktania, R., Saputra, Z., & Manufaktur Negeri Bangka Belitung, P. (n.d.). *GENERATOR CONTROL AND MONITORING SYSTEM WITH AUTOMATIC MAIN READINESS SUPPORTS IOT-based failure.*

Yudha, H., Munthe, P., Farizi Hernando, M., Amalia, D., Indra Martadinata, M., Soleh, A. M., & Kurniadi, D. (2023). Optimization of Maintenance Management, IoT, and Design Generator Set Equipment Monitoring. In Page: xx-xxx Journal of Engineering and Transportation (JET) (Vol. 1, Issue 1).

Yusuf, M. B., Rosyidi, L., Saptono, H., Informatics, T., Tinggi, S., Integrated, T., & Fikri, N. (2025). IMPLEMENTATION OF IOT SYSTEMS FOR MONITORING ELECTRICAL ENERGY CONSUMPTION IN SMART HOMES. *Journal of Digital Business and Technology Innovation (DBESTI)*, *2*(1), 28–34. Retrieved from www.pemantauan.com,

Zaenal, E., Abrianto, H., Sidik, D., & Kunci, K. (2024). PROTOTYPE GENERATION CONTROL AND MONITOR GENERATOR USING IOT-BASED ESP 8266. *Journal of Cahaya Mandalika*, *5*(1). Retrieved from http://ojs.cahayamandalika.com/index.php/JCM