

Diseño y Desarrollo de Sistema de Monitoreo y Registro de Eventos en Generadores de Energía Eléctrica

Kleinubing Fernando ^{a,*}, Vera Okulczyk Fabricio Hernán ^{a,*}

Facultad de Ingeniería, Universidad Nacional de Misiones (UNaM), Oberá, Misiones, Argentina.

e-mails: Kleinubing84@gmail.com, fabriciovera7@gmail.com

Resumen

Este trabajo trata sobre el diseño y construcción de un sistema de monitoreo de generadores de energía eléctrica. El proyecto es un trabajo en colaboración entre la Facultad de Ingeniería y el Hospital Samic. El proyecto busca monitorear el suministro eléctrico continuo en el entorno hospitalario, crucial para el funcionamiento de equipos médicos esenciales. El sistema propuesto almacena datos de medición de voltaje, corriente y potencia de las líneas eléctricas y monitorea el nivel de combustible y la operación de los generadores. Utiliza tecnología como Docker, modbus TCP/IP, MariaDB y NodeRed para la adquisición y gestión de datos. Un bot de Telegram permite la visualización de estos datos y envía notificaciones en caso de eventos. Además, el sistema cuenta con medidas de seguridad para asegurar el acceso solo a personal autorizado. La implementación del sistema ha facilitado la supervisión constante y el análisis de tendencias de uso, optimizando el mantenimiento preventivo y el consumo de combustible.

Palabras Clave – Automatización, Docker, Generadores, ModBus, MySQL, Telegram.

1. Introducción

En el entorno hospitalario, la continuidad del suministro eléctrico es vital. Los generadores de energía garantizan que equipos médicos esenciales, sistemas de soporte vital y otros servicios críticos continúen operando sin interrupción durante cortes de energía. La fiabilidad de estos dispositivos puede ser una cuestión de vida o muerte. El Hospital Samic cuenta con 4 generadores y 6 medidores de línea.

Dada la importancia crítica de los generadores en hospitales, es esencial contar con un sistema de monitoreo que pueda abordar los siguientes desafíos:

- **Mantenimiento Preventivo:** Identificación temprana de fallos potenciales para evitar interrupciones en el servicio, asegurando que los generadores estén siempre en condiciones óptimas.
- **Monitoreo de Rendimiento:** Seguimiento constante de parámetros operativos como voltaje, corriente y temperatura.
- **Gestión de Combustible:** Control del nivel de combustible para asegurar la disponibilidad continua del generador. Además, se puede hacer un control del consumo de los generadores.

El sistema actual posee algunos inconvenientes, entre estos, el que no se almacenan los datos obtenidos con los medidores, es decir, solo se conocen los valores de tensión, corriente y potencia

de cada línea en el momento en el que alguien lo revisa. Además, obtener el estado actual de los generadores requiere ir al lugar donde está instalado el generador para acceder al HMI (*Human Machine Interface*) del mismo, o utilizar el software SCADA (*Supervisory Control And Data Acquisition*), pero este está presente en una sola computadora en todo el Hospital, hay que tener en cuenta que el Hospital es bastante grande.

Como alternativa al sistema de monitoreo realizado existe:

- Software SCADA configurado para trabajar con los generadores.
- HMI instalados en cada generador.

Tomando todo esto en cuenta se propone realizar un sistema que almacene los datos, los presente de forma gráfica a través de un bot de *Telegram* y además, que provea alertas ante eventos relacionados en forma de notificaciones (mensajes enviados por el bot). Este software debe contar con un sistema de seguridad para que solo las personas autorizadas puedan ver los datos y gestionar el software.

2. Metodología

2.1. Puntos de medición de las líneas de corriente eléctrica y generadores de energía eléctrica

Se realizó una visita técnica al Hospital Samic de Oberá con el objetivo de obtener los requisitos para el sistema a desarrollar. En esta visita se aprendió que varios de los sistemas del Hospital (incluyendo los que fueron automatizados en este trabajo) cuentan con una interfaz modbus TCP/IP [1]. Las líneas para las diferentes cargas del hospital ya tienen instalados un módulo PM5300 que sirve para medir distintos parámetros eléctricos como tensión, corriente y potencia. Por otro lado, los generadores tienen instalados módulos IntelliCompact NT MINT, se presentan imágenes de cada uno de estos en las Figuras 1a y 1b.

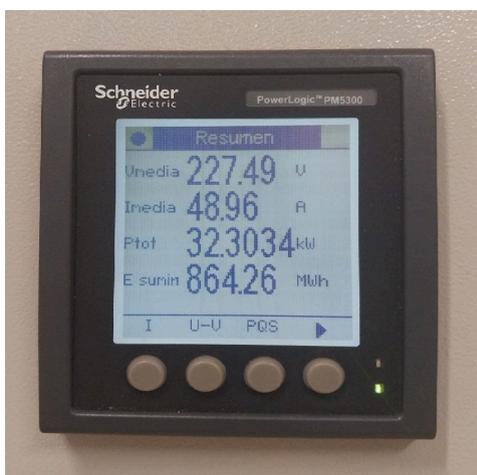


Fig. 1a. Módulo PowerLogic PM5300.



Fig. 1b. Módulo IntelliCompact NT MINT.

Luego en la figura 2a se puede observar las cabinas donde se encuentran las líneas con los módulos antes mencionados y en la figura 2b se tiene las cabinas con los grupos generadores.



Fig. 2a. Cabinas de líneas de corriente eléctrica



Fig. 2b. Cabinas de grupos generadores.

2.2. Estructura de sistema de monitoreo de generadores de electricidad

Se procedió a diseñar un sistema prototipo en base a los requisitos planteados durante una reunión previa con los ingenieros encargados de la administración del hospital. El sistema consiste en un conjunto de programas ejecutando en “contenedores” de Docker separados [2].

Estos programas son:

- *NodeRed*, utilizado para adquisición de datos (utilizando modbus TCP) y registrar notificaciones [3].
- Un programa web de gestión de base de datos (*phpMyAdmin*)
- Base de datos (*MariaDB*).
- Servidor web (*Nginx*), encargado de una página de índice y de ser un proxy reverso (con autenticación) [4].
- Script de *Python*, para guardar inicios de sesión en la base de datos.
- Bot de *Telegram*, encargado de funcionar como interfaz de usuario y enviar las notificaciones.

A continuación en la Figura 3, se presenta un diagrama de contexto que muestra el camino de datos del sistema. Donde los módulos InteliCompact NT MINT obtienen mediciones de los distintos parámetros de los generadores y los guarda en sus registros, posteriormente son adquiridos con protocolo modbus TCP por node-red. Luego node-red guarda los datos procesados en la base de datos y el bot de telegram permite ver estos datos.

Por otro lado, el usuario puede acceder a la base de datos a través de *phpmyadmin* con previa autenticación y puede acceder a node-red para realizar modificaciones en la adquisición y procesamiento de datos.

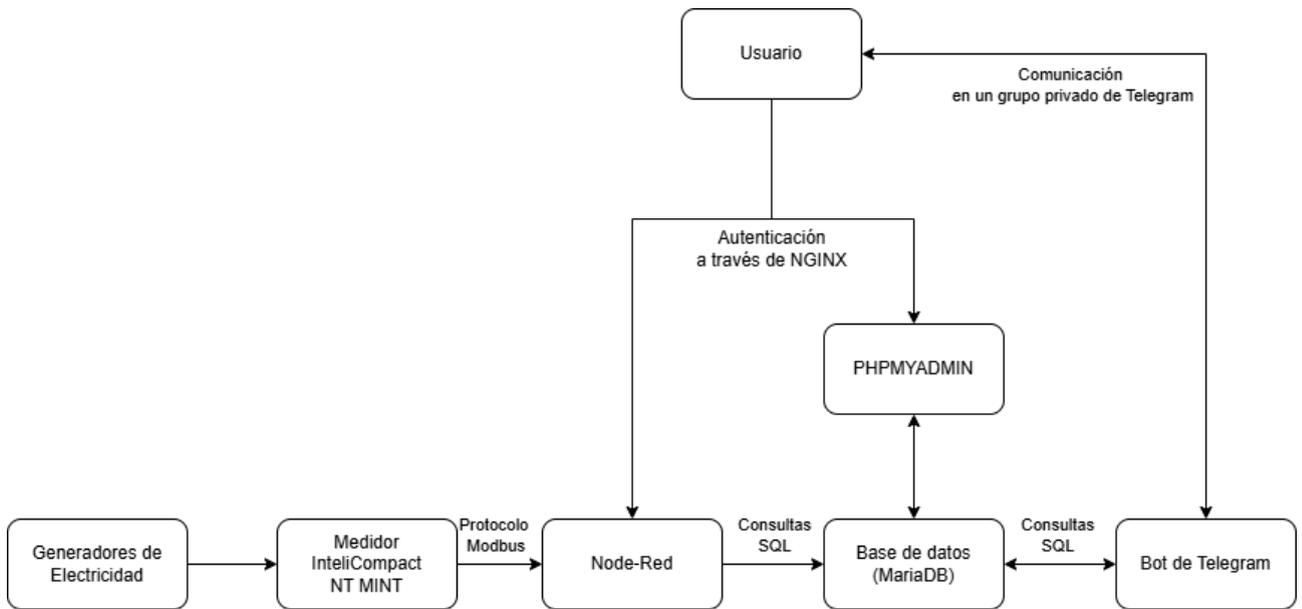


Fig. 3. Diagrama de contexto del sistema de monitoreo de generadores de electricidad.

2.3. Parámetros de interés

Para la construcción de la base de datos se consultó a los Ingenieros que trabajan en el hospital, se llegó a la conclusión que los parámetros a guardar para las líneas son:

- Un Identificador de Medición Único.
- El Identificador de Línea (Nombre Clave de la Línea).
- La marca de tiempo del momento en el que se hizo la medición.
- Los 3 voltajes de cada fase de la línea trifásica en triángulo (VL1L2, VL2L3, VL3L1).
- Las 3 corrientes pertenecientes a cada una de las fases (I1, I2, I3).
- Los 3 tipos de potencia (aparente, reactiva y activa).

Por otro lado para los generadores se guardan los siguientes datos:

- Un Identificador de Medición Único.
- El Identificador de Generador (Número de Grupo de los Generadores).
- La marca de tiempo del momento en el que se hizo la medición.
- Nivel de Combustible.
- Modo de Operación.
- Prioridad (orden en el que se encienden los generadores).

Para tomar las mediciones se programó en node-red scripts que toman mediciones periódicamente por modbus de cada uno de los parámetros que se encuentran guardados en registros del medidor InteliCompact NT MINT y los guarda en la base de datos. Los números de los registros a consultar para obtener los parámetros se obtuvieron de la lista de registros del producto encontrada en la página web “Schneider Electric” [5].

2.4. Interfaz de datos del bot de telegram

Se programó un bot de telegram que, haciendo uso de las bases de datos diseñadas, permite mostrar los datos de las últimas mediciones tomadas de cada uno de los generadores o líneas, y además, graficar con la librería de python Matplotlib las mediciones del último día [6].

En las Figuras 4 y 5 podemos ver una interacción entre el usuario y el bot de Telegram, en la Figura 4 se solicita una gráfica con los últimos niveles de tensión en una línea específica, por otro lado, en la Figura 5 el bot informa los niveles de combustible, la prioridad (orden de encendido) y la fecha y hora en la que se obtuvo dicha medición.

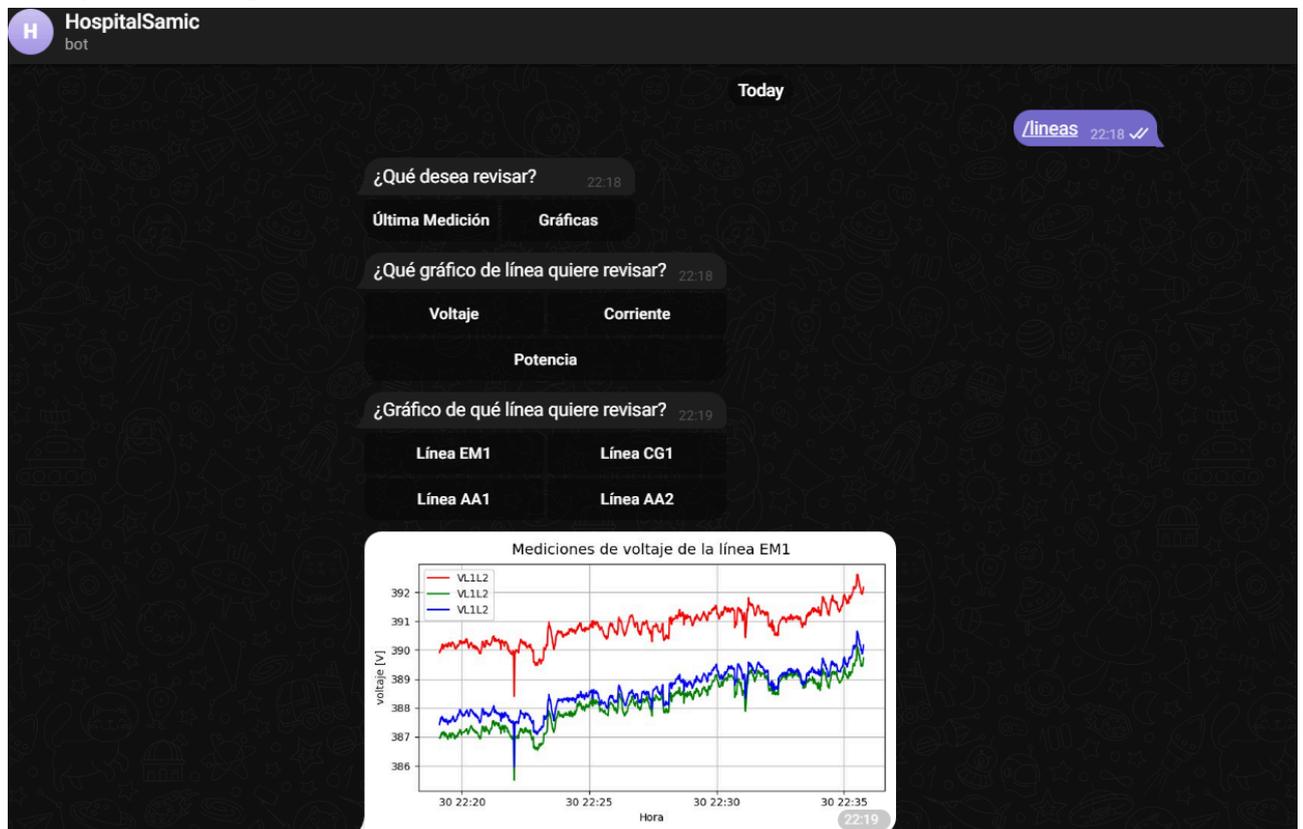


Fig. 4. Petición de gráfico de voltaje de línea para electromedicina.

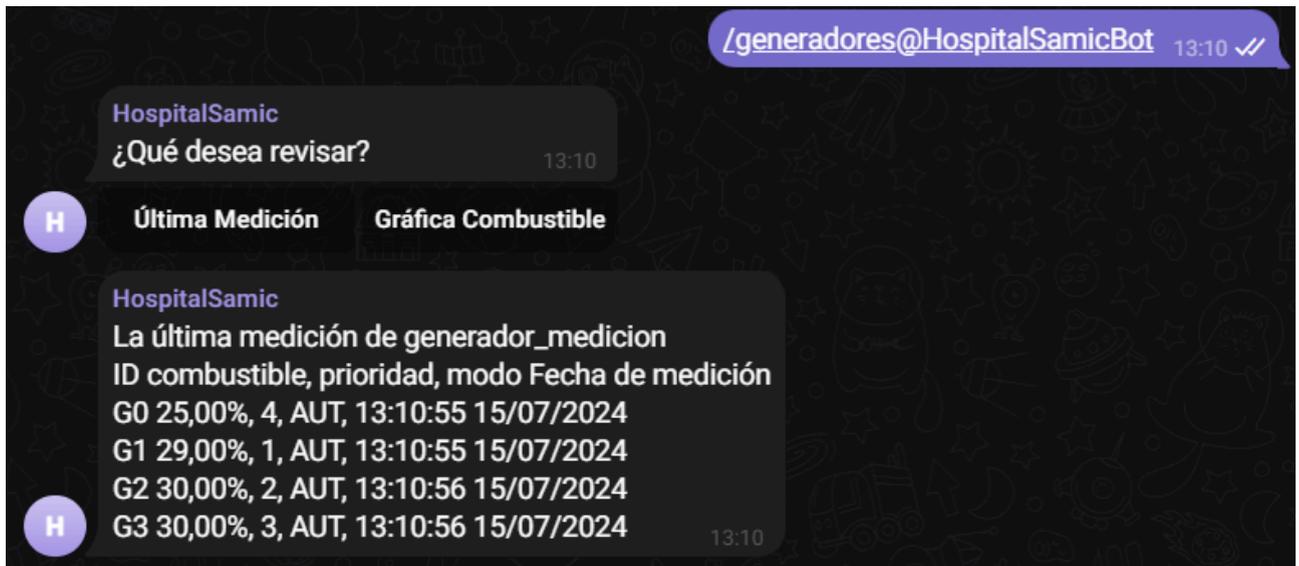


Fig. 5. Petición de última medición de nivel de combustible, prioridad y modo de los generadores.

2.5. Notificaciones del bot de telegram

Por otro lado, se implementó un script que detecta variaciones de combustible mayores a 3%, este rango de variación es debido a que el combustible tiene propiedades de dilatación térmica, lo que produce variaciones aunque no se haya consumido o cargado el tanque. Cuando se produce estas variaciones se guarda en otra tabla llamada “notificaciones” en la base de datos con la siguiente estructura:

- Un Identificador de Notificación Único.
- El mensaje que se notifica.
- Una columna llamada “notificado” que empieza en 0, y cuando el bot de telegram manda la notificación lo cambia a 1.
- Una marca de tiempo.

En la Figura 6 podemos ver una notificación enviada por el bot. Esta notificación informa sobre una variación significativa en el nivel de combustible de uno de los generadores.

Para las notificaciones de eventos como activaciones de los generadores o bajos niveles de refrigerante el módulo que se encuentra instalado en los generadores tiene un registro donde escribe la ocurrencia de estos eventos. En él se encuentra un código hexadecimal que representa al evento correspondiente. En la figura 7 y 8 se puede observar notificaciones enviadas por el bot de telegram cuando suceden estos eventos.

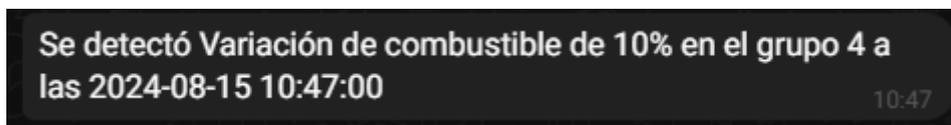


Fig. 6. Notificación sobre variación del nivel de combustible.

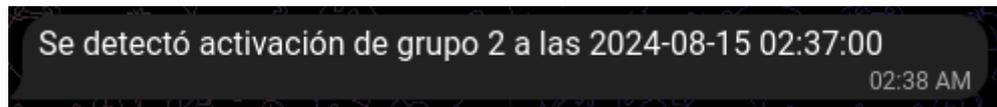


Fig. 7. Notificación sobre activación de un generador.

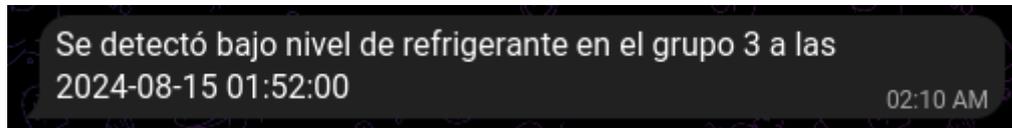


Fig. 8. Notificación sobre falta de refrigerante.

2.6. Medidas de seguridad

Como medida de seguridad, se agregó un script encargado de procesar el registro (*log*) del proxy reverso (*Nginx*) y cargar la información relevante a la base de datos. A esto se le agregó una verificación para evitar cargar múltiples entradas idénticas y consecutivas a la base de datos. Las entradas que este realiza a la base de datos tienen la siguiente forma:

- Un identificador único auto incrementable.
- Marca de tiempo en el que se cargó la entrada a la base de datos.
- Dirección IP con la que se accede al servidor.
- Nombre de usuario HTTP o “Anonimo” en caso de no haber iniciado sesión.
- Hora en la que el servidor registró la petición (*request*) HTTP.
- La petición HTTP en cuestión.
- Código de estado HTTP retornado por el servidor.

En la Figura 9 se observa la página principal (*index*), la cual es utilizada para listar las demás páginas en uso. Intentar acceder a cualquiera de las páginas listadas en la misma generará un cuadro de diálogo solicitando autenticación de usuario. En la Figura 10 se puede ver un cuadro de diálogo para autenticación en HTTP.



Fig. 9. Pagina principal.

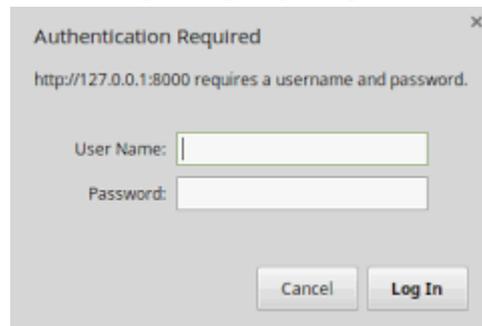


Fig. 10. Diálogo de autenticación HTTP.

3. Conclusión

El sistema proporciona una plataforma centralizada para el monitoreo en tiempo real de parámetros esenciales como voltaje, corriente, potencia de 4 de las líneas existentes. También se logró la visualización del nivel de combustible, las prioridades y los modos de operación de cada uno de los generadores. Al almacenar los datos históricos en una base de datos estructurada, el sistema permite a los ingenieros realizar análisis detallados de tendencias y patrones de uso facilitando así la detección de anomalías. Esta capacidad es crucial para la planificación de mantenimiento preventivo y la optimización del consumo de combustible.

La integración con un bot de Telegram proporciona un canal eficaz para alertar al personal autorizado sobre eventos, como variaciones significativas en el nivel de combustible, activaciones de los generadores ante cortes de energía eléctrica o bajo nivel de refrigerante. Esta característica

mejora la capacidad de respuesta ante emergencias y minimiza el riesgo de interrupciones en el suministro eléctrico.

Se han implementado medidas de seguridad para asegurar que solo el personal autorizado pueda acceder a los datos y funciones del sistema. Esto es fundamental para proteger información sensible y garantizar la integridad operativa del sistema.

Gracias a la forma modular en la cual fue diseñado este sistema de monitoreo, es posible agregar y/o modificar funciones sin mucha dificultad, esto da lugar a que futuros proyectos puedan aprovechar lo realizado en este proyecto.

4. Agradecimientos

Agradecemos al Hospital Samic por proporcionar el espacio y la colaboración para este proyecto.

5. Referencias

- [1] Modbus, Accedido: 20/07/2024, Available: <https://en.wikipedia.org/wiki/Modbus>
- [2] Docker, Accedido: 18/07/2024, Available: <https://www.docker.com>
- [3] NodeRed, Accedido: 18/07/2024, Available: <https://nodered.org>
- [4] Nginx, Accedido: 18/07/2024, Available: <https://docs.nginx.com/nginx/admin-guide>
- [5] Schneider Electric, Accedido: 19/07/2024, Available: <https://www.se.com/ar/es/faqs/FAQ000256516/>
- [6] Matplotlib, Accedido: 22/07/2024, Available: <https://matplotlib.org/stable/>