

# Monitoreo de seguridad y mantenimiento de datacenters y *shelters*

## Security and Maintenance Monitoring of Data Centers and Shelters

Presentación: 26 y 27 de octubre de 2022

### **Leonardo Depetris**

Universidad Tecnológica Nacional Facultad Regional San Francisco. San Francisco, Córdoba. Argentina.  
leodepetris@gmail.com

### **Damian Romani**

Cooperativa de Servicios Públicos de Morteros Ltda. Morteros, Córdoba. Argentina  
dromani@coopmorteros.coop

### **Joaquín Gonzalez**

Cooperativa de Servicios Públicos de Morteros Ltda. Morteros, Córdoba. Argentina  
jgonzalezi@coopmorteros.coop

### **Claudio Bottero**

Cooperativa de Servicios Públicos de Morteros Ltda. Morteros, Córdoba. Argentina  
cbottero@coopmorteros.coop

### **Lorenzo Depetris**

Cooperativa de Servicios Públicos de Morteros Ltda. Morteros, Córdoba. Argentina  
ljdepetris@coopmorteros.coop

### **Diego Vaira**

Cooperativa de Servicios Públicos de Morteros Ltda. Morteros, Córdoba. Argentina  
dvaira@coopmorteros.coop

### **Juan Gonzalez**

Cooperativa de Servicios Públicos de Morteros Ltda. Morteros, Córdoba. Argentina  
fgonzalez@coopmorteros.coop

### **Resumen**

Este proyecto consiste en el desarrollo de un prototipo que permita asegurar condiciones físicas óptimas de operación y mejore la seguridad mediante el control de acceso a *datacenters*, *shelters* o salas con equipos electrónicos en general

El prototipo se centra en obtener información del ambiente interior del recinto, específicamente, la temperatura y humedad en uno o más puntos; para garantizar el normal funcionamiento también se realiza la medición de la corriente consumida por los equipos de aire acondicionado; mientras que, con respecto a la seguridad, se optó por detectar apertura de puertas. Tras su puesta en marcha, se concluye que se trata de un prototipo útil, logrando

mejorar la respuesta ante fallos e incidentes; no obstante, para un monitoreo completo, resta incluir la tele lectura de los equipos UPS y generadores, los cuales se encuentran aún en etapa de desarrollo.

**Palabras clave:** Internet de las Cosas, Protocolo MQTT, Data Center, Shelter

## Abstract

This project involves the development of a prototype that ensures optimal physical operating conditions and guarantees security by controlling access to datacenters, shelters, or rooms with electronic equipment in general.

The prototype focuses on obtaining information about the enclosure internal environment, specifically, the temperature and humidity at one or more points; To guarantee normal operation, the measurement of the current consumed by the air conditioning equipment is also carried out; while, about security, it was decided to detect the opening of doors. After its implementation, it is concluded that it is a useful prototype, managing to improve the response to failures and incidents; however, for a complete monitoring, it remains to include the remote reading of the UPS and generator equipment, which are still in the development stage.

**Keywords:** Internet Of Things, MQTT Protocol, Data Center, Shelter

## Introducción

Producto del aumento en la necesidad de información, los equipos han incrementado la exigencia de energía para procesar los datos, lo que da como resultado un incremento importante en la densidad de los procesadores junto con la escalada de calor correspondiente. Para alcanzar la capacidad máxima de la red de sistemas, deben mantenerse por debajo de cierto rango de temperatura (un requerimiento cada vez más difícil de lograr).

Las altas temperaturas provocan fallas en los equipos impactando con altos costos derivados de las caídas del sistema, lo que hace más importante que nunca contar con equipos de climatización eficientes. Como prácticamente toda la electricidad que utilizan los procesadores se convierte en calor, el centro de datos en donde se aloja el equipo requiere proveer aire frío al sistema y reciclar las emanaciones producidas por el calor, intercambio que de realizarse en forma adecuada proporciona seguridad a la operación de los sistemas. Para prevenir fallas en los sistemas, la temperatura a la que operan debe ser menor a los 35°C / 38°C.

La Humedad Relativa (HR) y su velocidad de variación es un factor sumamente importante en el proceso de falla por corrosión dado que promueve el crecimiento de esta. Los equipos de comunicaciones están diseñados para operar en el rango 30-60% de HR. Sin embargo, en ocasiones los sistemas de refrigeración y distribución de aire elevan la HR hasta el 85%, durante el control de la temperatura. Un incremento de la temperatura ambiental puede acelerar la corrosión. Por ejemplo, se ha demostrado que el deterioro de placas de circuitos electrónicos expuestas a agentes corrosivos se acelera sensiblemente a temperaturas superiores a 27°C (Rossini, s.f.).

Algunas magnitudes físicas como la temperatura y humedad se controlan dentro de un rango preestablecido, sin embargo, éstas no se encuentran monitoreadas de forma remota por lo que habitualmente no existe forma de verificar su normal funcionamiento. Esto es de mayor interés en *Datacenter* móviles (*Shelters*) debido a su ubicación remota y sin personal de mantenimiento continuamente. A su vez, el hecho de la ubicación remota y sin personal, plantea la necesidad de garantizar la seguridad de los equipos ante robo y/o vandalismo, cuestión que también se abordará en el desarrollo del prototipo en cuestión.

## Desarrollo

Durante la planificación y desarrollo del proyecto se siguió el concepto de Internet de las cosas (IoT), es decir, integrar a todos los sistemas de dispositivos físicos que reciben y transfieren datos a través de redes inalámbricas con poca intervención humana, independientemente de su naturaleza. Para ello, se debió utilizar un protocolo de comunicación de datos que soporte estos requisitos. Se eligió el Protocolo MQTT, debido a que es muy útil para conexiones con clientes remotos demandando muy bajo ancho de banda de internet.

MQTT utiliza el modelo cliente / servidor. Su arquitectura contiene tres componentes, estos son: *Publishers*, un *Broker* y *Subscribers*. Cada dispositivo está conectado a un servidor (*Broker*). La dirección en la que se publicó el mensaje se llama “tópico” y cada dispositivo puede suscribirse a más de un tópico y recibir todos los mensajes que se publican en estos. Las principales responsabilidades del *Broker* MQTT son procesar la comunicación entre los clientes MQTT y distribuir los mensajes entre ellos en función de sus tópicos de interés, pudiendo trabajar con miles de dispositivos conectados al mismo tiempo. Al recibir el mensaje, el *Broker* debe buscar y encontrar todos los dispositivos que poseen una suscripción a este tópico. El *Broker* fue implementado utilizando Eclipse Mosquitto, siendo este un agente de mensajes de código abierto que implementa las versiones 5.0, 3.1.1 y 3.1 del protocolo MQTT. Es liviano y adecuado para su uso en todos los dispositivos, desde computadoras de placa única de baja potencia hasta servidores completos.

Resulta de gran utilidad mencionar la herramienta “MQTT Explorer”, la cual (una vez configurado con los datos de nuestro servidor MQTT) nos permite visualizar el árbol de tópicos de dispositivos conectados, incluso obtener gráficas rápidas, siempre y cuando tengamos configurados los campos en JSON.

En cuanto a la base de datos, se eligió InfluxDB, no solo por ser de código abierto, sino también debido a la simplicidad de su configuración, su enfoque optimizado para el monitoreo de operaciones, métricas de aplicaciones, datos de sensores de Internet de las cosas y el análisis en tiempo real.

Teniéndose como punto de partida el desarrollo previo de una oficina virtual de usuarios internos de la empresa en etapa de producción, se decidió utilizar ésta como soporte para cálculos, visualización de magnitudes instantáneas y gráficos históricos.

Como hardware se desarrollaron circuitos de medición acoplados a módulos ESP32 o Arduino DUE (con *shield* ethernet). La distinción se realizó considerando como factor predominante la disponibilidad de conexión de red o de WiFi según convenga. Se programó en lenguaje Arduino como lenguaje estándar para los scripts de cada dispositivo desarrollado, principalmente por la disponibilidad de librerías utilizadas y las que puedan necesitarse en ampliaciones futuras.

### Salas en las que se aplicó el prototipo

*Datacenter*: esta sala requiere únicamente monitoreo de humedad y temperatura. Debido a la gran cantidad de servidores y equipos electrónicos, se monitorea temperatura en 2 puntos de la sala. A su vez, para garantizar el correcto funcionamiento de los equipos de aire acondicionado se colocaron sensores de corriente en su cable de alimentación.

*Sala con equipos electrónicos varios*: de igual manera que la sala anterior, se determinó la necesidad de monitorear la temperatura y humedad, como así también contar con la disponibilidad de adquirir información de consumo, según sea el caso.

**Nodo de TV:** siguiendo la misma lógica que en los casos anteriores, vemos que la necesidad primordial radica en monitorear las condiciones ambientales de su interior y el correcto funcionamiento de los equipos de aire acondicionado. Esta sala incluye como nuevo requerimiento el control y monitoreo de las balizas de seguridad presentes en la antena.

**Shelters remotos:** por lo expuesto inicialmente, estos equipos se encuentran en peores condiciones debido a su ubicación, quedando más expuestos a condiciones climáticas adversas y vandalismo. Es por ello, que además de requerir las anteriormente mencionadas mediciones de temperatura, humedad, consumo de aires acondicionados y control de balizas, se debe garantizar la seguridad de este, cuestión que se monitorea mediante sensores de puertas.

Para desarrollar un único hardware que permita su aplicación indistintamente en cada una de estas salas se consideró que el prototipo sea capaz de tomar datos de dos sensores de temperatura y humedad, capaz de medir dos corrientes, detectar cierre/apertura en dos puertas y controlar una baliza. A su vez se decidió desarrollar un segundo dispositivo capaz de medir corriente (monofásica o trifásica), debido a que pueden existir varios puntos de medición de interés dentro de una misma sala. Cabe aclarar que cada una de estas salas complementa su seguridad con sistemas de cámaras y alarmas, cuestiones que exceden a este prototipo.

#### *Sensores utilizados*

Para la determinación de humedad y temperatura se utilizaron sensores de DHT11, el cual tiene la suficiente exactitud de medición para la aplicación que se busca.

En cuanto a la medición de corriente se decidió utilizar el módulo ACS712, capaz de medir tanto corriente alterna como de corriente continua utilizando la tecnología de efecto Hall. Consta de un circuito de tamaño reducido y a su vez, resulta ser de muy bajo costo. El módulo de sensor ACS712 tiene 3 variantes: ACS712-05B en el rango de -5A a 5A, ACS712-20A en el rango de -20A a 20A, ACS712-30A en el rango de -30A a 30A; permitiendo versatilidad de medición ya que un mismo hardware implementado puede adaptarse según el consumo estimado del equipo a controlar simplemente intercambiando el módulo.

Se utilizó un transistor TIP122 para comandar las luces led propias de las balizas, mientras que la secuencia de encendido/apagado se determina por software.

Por último, para la medición de corriente alterna monofásica o trifásica del segundo prototipo se utilizó el sensor de corriente SCT-013. Esta familia de sensores de corriente es no invasiva para medir la intensidad de corriente alterna que circula por un conductor, es decir, sin cortar el cable o conductor. Los sensores SCT-013 están contruidos como un transformador de corriente, que proporciona una medición proporcional a la corriente que atraviesa el circuito. Este tipo de medición indirecta, se la conoce como inducción electromagnética. Debido a que la señal proveniente del sensor presenta un offset, se utiliza un ADC externo que permita maximizar la resolución de medición.

Luego de la implementación física de los módulos se procedió a realizar los ajustes finos de calibración y de transmisión. Una vez eliminadas estas fallas, se desarrollaron los indicadores y gráficas a usuario dentro de la Oficina Virtual, que se observan en la Figura 1.

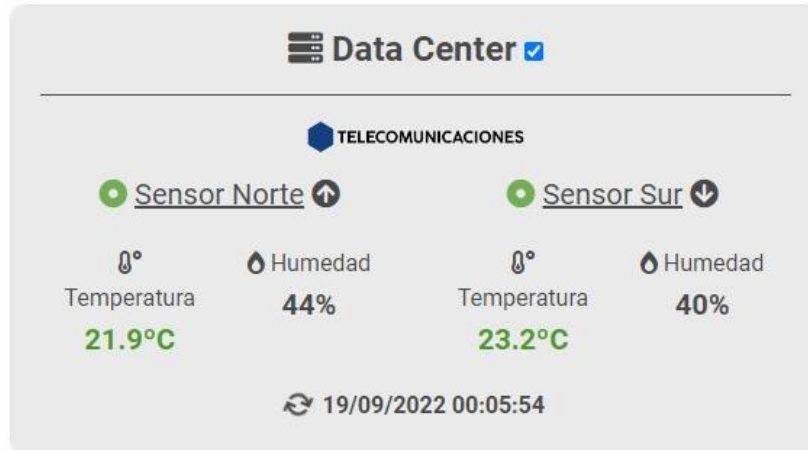


Figura 1: ejemplo de indicador en Oficina Virtual CoopMorteros

## Conclusiones

Tras varios meses desde la implementación física del desarrollo se puede concluir favorablemente, considerándolo un prototipo de gran utilidad en ambientes como los descritos. Puntualmente, existieron eventos que fueron detectados por los diversos sensores y dieron origen a una pronta respuesta correctiva, minimizando los impactos que se podrían haber causado. Por mencionar algunos, mediante las alertas y análisis de gráficas de temperatura en el nodo de TV se detectó el mal funcionamiento de un aire acondicionado, procediendo con su respectiva limpieza y reparación. Otro evento de importancia se ocasionó en uno de los *shelters* remotos, que mediante el análisis de humedad y temperatura se descubrió falta de aislación con el medio exterior, lo que llevó a la reparación del chasis para asegurar su hermetismo y estanqueidad. Como último ejemplo, también se puede mencionar un error humano que fue un cierre incorrecto de una de las puertas de otro de los *shelters* remotos, rápidamente advertido por el sistema al quitar el “modo de mantenimiento” colocado al inicio de una actualización/servicio.

A partir del uso del prototipo, surgen nuevos requerimientos de desarrollos aplicables en estos ambientes. Por un lado, debido a que estas instalaciones funcionan de manera continua y sin interrupciones, es común encontrar generadores eléctricos y fuentes de alimentación ininterrumpidas (UPS) encargados del suministro eléctrico ante un corte; por ello, las nuevas líneas de investigación se centran lograr comunicación con estos equipos para constatar su normal funcionamiento y realizar los mantenimientos oportunamente. Mientras que, por otro lado, se encuentra en fase de desarrollo un prototipo de toma eléctrica múltiple, con control remoto online individual, que permita cortar el suministro de energía; este desarrollo resulta de gran interés en aquellos equipos que deseen apagarse o reiniciarse remotamente y no tengan esta característica.

## Referencias

¿CÓMO AFECTA LA TEMPERATURA A LOS EQUIPOS ELÉCTRICOS Y ELECTRÓNICOS? (s.f.). Obtenido de <https://hoffman-latam.com/blog/como-afecta-la-temperatura-a-los-equipos-electricos-y-electronicos/>

Automatización para todos. (3 de septiembre de 2020). Sensor de Temperatura y Humedad Relativa DHT11 con Arduino. Obtenido de <https://www.automatizacionparatodos.com/sensor-dht11-arduino/>

Eclipse Mosquitto - Un bróker MQTT de código abierto. (s.f.). Obtenido de <https://mosquitto.org/>

Naylamp Mechatronics. (s.f.). TUTORIAL SENSOR DE CORRIENTE ACS712. Obtenido de [https://naylampmechatronics.com/blog/48\\_tutorial-sensor-de-corriente-ac712.html](https://naylampmechatronics.com/blog/48_tutorial-sensor-de-corriente-ac712.html)

Rossini, L. A. (s.f.). Estación de Monitoreo Remoto de Variables Climáticas y Contaminantes Atmosféricos. JUI, Jornadas de Vinculación Universidad-Industriacd.

Shiva Shankar J; Dr.S.Palanivel; Dr.S.China Venkateswarlu; M.Sowmya. (diciembre de 2019). MQTT in Internet of Things. International Research Journal of Engineering and Technology (IRJET), 06(12).

Soloelectronicos. (s.f.). WATIMETRO ARDUINO Aplicaciones del SCT-013. Obtenido de <https://soloelectronicos.com/tag/watimetro-arduino/>