



Recepción: 30 / 09 / 2017

Aceptación: 28 / 11 / 2017

Publicación: 15/ 12/ 2017



Ciencia de la computación

Artículo de investigación

Gestión de redes de sensores inalámbricos (WSN) para la industria petrolera

Management of wireless sensor networks (WSN) for the oil industry

Gerenciamento de redes de sensores sem fio (WSN) para a indústria de petróleo

Roberto C. Dender-Zurita
rd_yanio@hotmail.com

Pascolin M. Guilcatoma-Moreira
pacoloco08@hotmail.com

José G. Argandoña-Moreira
cm96478@gmail.com

Mike P. Machuca-Avalos
mike.machuca@uleam.edu.ec

Correspondencia: rd_yanio@hotmail.com

- I. Magister en Administración de Empresas Mención Planeación, Magister en Telecomunicaciones, Ingeniero en Electrónica Digital y Telecomunicaciones, Docente Universidad Técnica Luis Vargas Torres de Esmeraldas, Esmeraldas, Ecuador.
- II. Master en Mecatrónica y Robótica, Ingeniero en la Especialidad de Automatización y Control de Procesos Tecnológicos e Industrias, Docente Universidad Técnica Luis Vargas Torres de Esmeraldas, Esmeraldas, Ecuador.
- III. Ingeniero Eléctrico, Docente del Instituto Tecnológico Eloy Alfaro de Manabí, Manta, Ecuador.
- IV. Magister en Comercio y Finanzas Internacionales, Ingeniero Eléctrico, Universidad Laica Eloy Alfaro de Manabí, Manta, Ecuador.

Resumen

Las redes de sensores inalámbricos (WSN) hoy en día son de gran ayuda porque permiten llegar a lugares dificultosos, ya sean estos de gran altura o lugares confinados, lo que facilita su acceso a la obtención de datos de una manera fácil, confiable y en línea sin necesidad de infraestructuras físicas. En este trabajo se expone la implementación de una red de sensores inalámbricos que permite la expansión del sistema de una manera rápida debido a que existen múltiples aplicaciones para la obtención y procesamiento de datos. Después, se muestra la creación de un sistema de gestión de la red inalámbrica de sensores a escala para el modelado de una red de sensores inalámbricos en la industria petrolera. Los resultados obtenidos permitieron obtener mediciones de los eventos ocurridos en cada uno de los sensores que fueron evaluados en la WSN.

Palabras clave: zigbee; sensores; redes; datos; inalámbrico; WSN.

Abstract

The wireless sensor networks (WSN) nowadays are of great help because they allow to reach difficult places, be they high-rise or confined places, which facilitates their access to obtaining data in an easy, reliable way and in line without the need for physical infrastructure. In this paper the implementation of a network of wireless sensors is exposed, which allows the expansion of the system in a fast way because there are multiple applications for obtaining and processing data. Afterwards, the creation of a wireless sensor network management system is shown for the modeling of a network of wireless sensors in the oil industry. The results obtained allowed us to obtain measurements of the events that occurred in each of the sensors that were evaluated in the WSN.

Keywords: zigbee; sensors; networks; data; wireless; WSN.

Resumo

As redes de sensores sem fio (WSN) hoje em dia são de grande ajuda porque permitem alcançar lugares difíceis, sejam eles lugares de alto ou limitado, o que facilita seu acesso à obtenção de dados de forma fácil e confiável e em linha sem a necessidade de infra-estrutura física. Neste artigo, a implementação de uma rede de sensores sem fio é exposta, o que permite a expansão do sistema de forma rápida, porque existem múltiplas aplicações para obtenção e processamento de

dados. Posteriormente, a criação de um sistema de gerenciamento de rede de sensores sem fio é mostrada para a modelagem de uma rede de sensores sem fio na indústria do petróleo. Os resultados obtidos nos permitiram obter medições dos eventos que ocorreram em cada um dos sensores que foram avaliados na WSN.

Palavras chave: ziguezague; sensores; redes; dados; sem fio; WSN.

Introducción

La industria de petróleo y gas incluye procesos de exploración, extracción, refinación, transporte y comercialización de productos petrolíferos. El mayor volumen de productos de la industria son el combustible, el petróleo y la gasolina (nafta). El petróleo es también la materia prima para muchos productos químicos, incluyendo productos farmacéuticos, solventes, fertilizantes, plaguicidas y plásticos. A medida que la demanda de combustibles fósiles continúa creciendo, las compañías de petróleo y gas tendrán que desarrollar nuevas tecnologías y mejorar las operaciones con el fin de aumentar la productividad y ampliar sus capacidades actuales.

Además, las regulaciones medioambientales están cambiando constantemente y cada vez son más estrictas día a día. En 2008, la Comisión de Conservación de Petróleo Oil Conservation Commission, (OCC) aprobó un conjunto de normas sobre la gestión de los residuos superficiales procedentes de las operaciones de petróleo y gas que obligan a las empresas a transportar el suelo y el agua altamente contaminados a sitios de eliminación permanente. La OCC también aprobó sanciones más estrictas para hacer cumplir la industria con las normas ambientales.

Las compañías de petróleo y gas deben desarrollar nuevos métodos para cumplir con las nuevas regulaciones y reducir los accidentes y las emisiones sin afectar la producción. Debido a la evolución de las tecnologías digitales y las comunicaciones inalámbricas, las redes de sensores inalámbricos pueden organizarse rápidamente y adaptarse continuamente para monitorear y controlar las condiciones ambientales y maquinaria en respuesta a los requerimientos del negocio.

Las tecnologías inalámbricas han evolucionado rápidamente, lo que ha permitido obtener datos y facilitar las comunicaciones. En el año 2003 según el Instituto Tecnológico de Massachusetts (MIT), identificó 10 tecnologías salientes, entre las cuales surge Wireless Sensor Networks (WSN). A partir de este suceso el avance de esta tecnología se ha incrementado y se están

implementando millones de redes de este tipo a lo largo del mundo. Inicialmente su principal aplicación fue en el campo militar, ahora se han desarrollado a muchas aplicaciones entre ellas las áreas industriales y de interés social, así como en el control de procesos de producción, control de tráfico, monitoreo de la salud y automatización de hogares.

Los recientes avances en la tecnología de detección inalámbrica fomentan la optimización y mejora de los procesos de desarrollo de productos y prestación de servicios. Las Redes de Sensores Inalámbricos Industriales (IWSN) son una clase emergente de WSN que enfrenta restricciones específicas relacionadas con las particularidades de la producción industrial.

Durante la búsqueda de información de trabajos relacionados con aplicaciones de redes de sensores inalámbricos en el sector industrial y en beneficio de la sociedad. Las WSN permiten conectar múltiples sensores para alguna aplicación específica. Por ejemplo, Ecuador no es la excepción y comienza a incursionar en este campo, gracias al proyecto realizado por investigadores de los siguientes centros de educación superior: Universidad de las Fuerzas Armadas “ESPE”, Universidad de Harvard y la Universidad de Carolina del Norte que recientemente implementaron una WSN, la que permite monitorear las erupciones en el Volcán Tungurahua. (Azuero, 2016)

Otra referencia acerca de las redes de sensores inalámbricos es que se brinda la posibilidad de utilizar una nueva tecnología en el área agrícola para optimizar el uso de un recurso no renovable, como el agua, mediante una interfaz amigable con el usuario, de fácil manejo, logrando así fusionar dos áreas que aparentemente no tienen relación. (Maya, 2014)

Los autores Rao, Santhi, Professor, & Lavanya (2012), en su investigación se centran en el monitoreo y protección de operaciones de petróleo y gas utilizando redes de sensores inalámbricos que están optimizadas para disminuir la instalación, y el costo de mantenimiento, requerimientos de energía, aumentar la confiabilidad y mejorar la eficiencia de la comunicación. Además, de que presentan experimentos de simulación utilizando el modelo propuesto, que permiten proporcionar nuevas herramientas para la investigación en mantenimiento predictivo y monitoreo basado en condiciones de maquinaria de fábrica en general y para sistemas de mecanizado de arquitectura abierta en particular.

La mayor parte de entornos petroleros en nuestro país no tienen un sistema de comunicación inalámbrico para la obtención de los datos y por esto siempre se dependerá del factor humano o de sistemas antiguos para el traslado de la información hacia las centrales, es notorio que las oportunidades que hoy en día brindan las TICs, para la monitorización y control temprana de procesos industriales mediante el uso de WSN, resultaría beneficioso para la sociedad y el país.

La detección inalámbrica mediante sensores ya no necesita ser relegada a lugares donde el acceso es difícil o donde el cableado no es práctico. Los sistemas de monitoreo de condiciones inalámbricas pueden implementarse de manera rentable en aplicaciones extensas que históricamente se manejaban al ejecutar rutas con recolectores de datos.

Mientras que para Obodoeze, Inyiyama, & Idigo (2012) en su investigación, introducen arquitecturas prácticas de implementación y de mecanismos que pueden asegurar que las instalaciones de petróleo y los sensores inalámbricos sean atacados físicamente para que puedan supervisar con éxito y reportar con éxito las incidencias de vandalización de tuberías y equipos de manera fácil y puntual, manteniendo al mismo tiempo la seguridad de los datos de la red de sensores inalámbricos.

Las aplicaciones de WSN en el dominio de detección de eventos se pueden clasificar en cinco categorías principales:

- a) monitorización de la calidad del agua mediante STORM 3 Data Loggers [investigación publicada por los autores Sun, Ahmed, Sun, Qian, & Xiao, (2016)]
- b) medición de la temperatura de fibra óptica distribuida en áreas de minas de carbón [investigación publicada por los autores Zhao, Li, Zhang, & Zheng, (2010)]
- c) localización y detección de emisiones acústicas del oleoducto [investigación publicada por los autores Wang, Jiao, Yang, & Niu, (2016)]
- d) monitorización de pulso y temperatura de pacientes basados en ZigBee [investigación publicada por Agajo, (2016)]

- e) monitorización del tránsito vehicular y de contaminación CO₂ [trabajo realizado por los autores Granda Bravo & Belduma Belduma, (2017)]
- f) monitoreo en aplicaciones domóticas [trabajo realizado por los autores Cedeño Villarroel & Eras Pérez, (2010)]
- g) control de los signos vitales en pacientes adultos mayores [trabajo realizado por Ruiz Sánchez, (2016)]
- h) monitoreo continuo de la humedad del suelo [investigación desarrollada por los autores Flores-Medina, Flores-García, Velasco-Martínez, González-Cervantes, & Jurado-Zamarripa, (2015)]

En estos términos, IWSN se enfrenta a varios desafíos como la fiabilidad y robustez en entornos agresivos, así como la capacidad de ejecutar correctamente y alcanzar la meta en paralelo con todos los otros procesos industriales. Además, las soluciones de IWSN son versátiles, fáciles de usar e instalar, de larga vida útil y dispositivos de bajo costo - de hecho, la combinación de requisitos difíciles de cumplir.

Este trabajo presenta como objetivo: Diseñar un sistema de gestión de una red de sensores inalámbricos para la industria petrolera.

Metodología

Esta investigación tiene un corte cuantitativo, y a veces mixto (cuali-cuantitativo). La metodología de investigación utilizada en el presente trabajo es del tipo descriptivo y explicativo con enfoque empírico-analítico. Se puede considerar que la simulación y programación es otro método de investigación, pero propio. Esto permite el desarrollo del sistema de gestión de redes de sensores inalámbricos (WSN) para la industria petrolera.

Desarrollo

En el análisis de campo mediante medición se obtuvieron las distancias requeridas para cubrir la conectividad de los sensores. Para la conectividad de ciertos sensores se utilizaron los módulos inalámbricos ESP8266, por su facilidad de adquisición en el mercado nacional y por el costo fueron ideales para realizar la implementación a pequeña escala.

El módulo ESP8266 wifi es capaz de alojar una aplicación y distribuir cualquier resultado dentro de una red wifi ya que utiliza el protocolo TCP/IP, este se encuentra integrada al dispositivo en mención.

Cada módulo ESP8266 viene preprogramado con un firmware de conjunto de comandos AT, lo que significa que puede conectarse a una placa embebida y poder transmitir información a un servidor externo.

Además, cuenta con una gran variedad de pines o puertos digitales y analógicos que pueden ser utilizados en lo que se requiera. En este módulo se cargará un código fuente que responderá a la variación de voltaje por medio de un potenciómetro para simular la temperatura o presión como resultado de un sensor, por ejemplo, que se encuentre ubicado en un tanque de almacenamiento.

Para la red inalámbrica del presente trabajo se necesita un punto de acceso (Access Point, AP), el mismo que nos va permitir conectar varios puntos de red. Para esto se utiliza el router inalámbrico modelo DIR-600 de la marca D-link ya que por su bajo costo fue escogido para la implementación.

El router utiliza la tecnología Wireless 150 que ofrece velocidades superiores a los estándares 802.112b/g, además incluye un firewall de inspección de estado de paquete (SPI) que analiza el tráfico de red. Además, soporta cifrado WEP, WPA y WPA2 para mantener la red segura ante intrusos.

Para la implementación a pequeña escala del proyecto se utilizó una pequeña computadora con arquitectura i686 en el que se instaló el sistema operativo Debian 8.

Entre las características del CPU están: Procesador AMD turbo 3600+x 1.7GHz, Memoria RAM de 2 GBytes, Disco duro de 160 GBytes, Tarjeta de red VIA Technologies, Inc. VT6102 [Rhine-II], Unidad de CD incluida LG

En la figura 1 se muestra la topología utilizada para la conectividad de los sensores y el servidor en nuestro router. Esta topología se aplica después de realizar un análisis de propagación en el sitio que se desea implementar, ya que puede existir interferencias el cual perjudicarían a la transmisión de resultados. Se puede ver las asignaciones de direcciones IP: (a) para el servidor Debian 192.168.100.16, y (b) de los cuatro nodos sensores ESP8266, que son 192.168.100.20 /21 /22 /23.

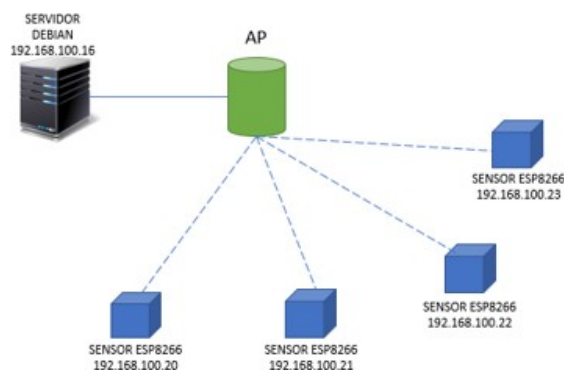


Figura 1: Topología utilizada en la implementación de WSN. Elaboración propia

Se utilizan los módulos ESP8266 para simular el funcionamiento de sensores de temperatura, presión, flujo y nivel, ya que son los datos más requeridos en la industria petrolera. Para ello es necesario cargar un software en código de programación que nos ayude junto con un potenciómetro a realizar una variación de una señal analógica y convertirla en digital con un máximo de 12 bits.

En la figura 2 se presenta el diseño esquemático de la tarjeta Arduino nano (para ISIS Proteus está librería la llaman Simulino Nano) para simular por código la conversión de una señal analógica a digital. Se utiliza esta tarjeta Arduino nano para facilitar la simulación, debido a que los módulos inalámbricos ESP8266 no se encuentran en la librería del software ISIS Proteus 8.

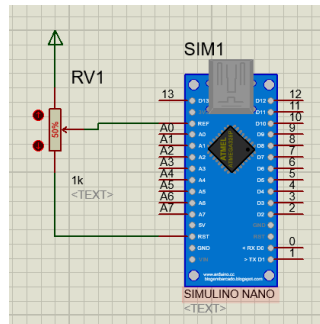


Figura 2: Esquema para la simulación de sensor de temperatura. Elaboración propia

Adicionalmente, vale hacer énfasis que el mismo código de programación que se utiliza para estas tarjetas embebidas serán cargadas en cada uno de los módulos ESP8266 activando unas librerías del IDE de Arduino. A continuación, se presentan por segmentos el código de programación en C++ aplicado en el módulo ESP8266 desde el IDE de Arduino. En la primera parte se llama a las librerías ESP8266Wifi y pubsubclient que son responsables para especificar la tarjeta con la que se trabaja en la implementación y de realizar el intercambio de datos y negociación entre el módulo y el router wifi para permitir la conectividad a la red.

Una vez llamadas las librerías mencionadas anteriormente, se procederá a crear las variables char con el contenido de los datos para la red wifi del presente trabajo, que para nuestro caso el SSID. SSID es el nombre de la red wifi, <<password>> es la clave para acceder y por último el <<mqtt_server>> que es la dirección IP asignada a nuestro servidor.

El <<mqtt>> es un protocolo de comunicación utilizado en el internet de las cosas (IoT) que nos ayuda a comunicar diferentes dispositivos en una sola red y enviar datos leídos por sensores digitales o analógicos a un servidor para almacenarlos en una base de datos. Este protocolo se utiliza para la simulación e implementación del sistema de gestión, además puede ser sustituido por cualquier otro que se requiera.

Posteriormente se especifican las variables <<float val>>, este será el contenedor del valor analógico que se ingresa por el pin 0. Las siguientes líneas que indican <<Wificlient>> y <<PubSubClient>> son modos en los que trabajan las librerías, en este caso especificamos que el módulo será configurado como cliente. El código que permite crear una función que realiza la

negociación y anclaje a la red wifi con el router o AP. Con esto se logra la seguridad de la placa que ha sido anclada a una red wifi, el cual fue especificado en el código de programación.

Si se conecta el módulo a un puerto del PC este procede a enviar los datos del status por el puerto serial, con lo cual se puede visualizar los mensajes en el código de la función <<setup_wifi>>. Una vez conectado a la red wifi, es necesario buscar el servidor <<mqtt>> para poder comenzar a enviar los datos de los resultados a este, en el caso de que no sea posible conectar este, debe quedar en un ciclo intentándolo reiteradas veces.

Es necesario inicializar el puerto serial para monitorear el estado del módulo en el terminal del puerto serial, así también inicia la función <<setup_wifi>> para que inicialmente en el código se establezca una conexión entre el módulo y la red. Además, con la línea <<client-setServer>> especificamos la IP del servidor <<mqtt>> y del puerto que en este caso será 1883.

En la figura 3 se muestra el código principal que se encarga de tomar los datos del puerto ADC (convertidor A/D) pin 0 y transformarlos en digital, enviando al servidor como una variable string.

```
void loop()
{
  if (!client.connected()) {
    reconnect();
  }
  client.loop();

  val = analogRead(analogPin);
  val = map(val, 0, 1023, -50, 200);
  Serial.println(val);
  temp = val;
  delay(500);
  if((temp > -50) && (temp < 200))
  {
    client.publish("ha/_temperature1", String(temp).c_str());
  }
}
```

Figura 3. Función principal del código. Elaboración propia

En la primera parte de la función principal condicionamos el inicio, si el módulo se encuentra correctamente anclado a la red y al servidor, caso contrario deberá intentarlo nuevamente. Después, se asigna la lectura del puerto analógico a la variable <<val>> y con la función <<map>> se va a transformar el resultado a los valores entre -50 a 200 que corresponden a la

temperatura. Posteriormente, se asigna el valor <<val>> a <<temp>> para enviarlo al servidor mediante la conectividad de cliente del módulo con la correcta sintaxis.

Para que el sistema de gestión sea funcional es necesario implementar un servidor que nos permita obtener y administrar los datos recibidos por cualquier sensor o periférico. Para esto fue necesario instalar en el servidor el sistema operativo Debian 8, ya que se tiene gran cantidad de información sobre su instalación y repositorios disponibles. No fue necesario instalar una interfaz gráfica, ya que una vez instalado el sistema operativo se puede administrar por <<ssh>> conectado remotamente con el programa llamado <<Putty>> el cual nos permite acceder desde cualquier punto de la red al terminal de este y poder realizar las instalaciones y configuraciones necesarias.

Una vez especificada la IP del servidor con el respectivo puerto podemos tener acceso al servidor e iniciar con las credenciales de usuario y contraseña.

Cuando se inicializa el servidor se procede con la instalación y configuración del sistema de gestión web. Es recomendable, que antes de iniciar cualquier tarea se tenga que actualizar el sistema, debido a que pueden existir repositorios descontinuados. La actualización se la puede conseguir escribiendo el siguiente comando <<apt-get update>>. Además, es necesario tener actualizado e instalado una lista de programas en el servidor como Python 2.7, Python-pip, Python-setuptools, MariaDB y librerías SSL.

Cuando se tiene listo el sistema operativo y los programas necesarios, se debe crear dos carpetas donde se va instalar el núcleo del sistema de gestión. Asignamos un usuario para administrar las carpetas, para que esto suceda se debe escribir las líneas de código que se muestran en la figura 4.

Las líneas permiten crear una carpeta en el directorio <</opt>> con el nombre <<dmg>>, donde se descargan e instalan los dos paquetes necesarios para el desarrollo del sistema de gestión. En la figura 4 se presenta el procedimiento para descargar ambos paquetes de internet. Se descargan los paquetes <<domogik>> y <<domogik-mq>> que se guardan en la carpeta creada para descomprimir e instalar. En general, estos paquetes serán el núcleo de todo el sistema, es decir, que se administra y configura en un entorno web los datos recibidos de los sensores instalados.

```

root@Server:/opt/dmg/domogik-mq# cd /opt/dmg/
root@Server:/opt/dmg# wget --content-disposition https://github.com/domogik/domogik/archive/0.5.2.tar.gz
--2017-06-22 20:59:04-- https://github.com/domogik/domogik/archive/0.5.2.tar.gz
Resolviendo github.com (github.com)... 192.30.253.112, 192.30.253.113
Conectando con github.com (github.com)[192.30.253.112]:443... conectado.
Petición HTTP enviada, esperando respuesta... 302 Found
Localización: https://codeload.github.com/domogik/domogik/tar.gz/0.5.2 [siguiendo]
--2017-06-22 20:59:04-- https://codeload.github.com/domogik/domogik/tar.gz/0.5.2
Resolviendo codeload.github.com (codeload.github.com)... 192.30.253.120, 192.30.253.121
Conectando con codeload.github.com (codeload.github.com)[192.30.253.120]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: no especificado [application/x-gzip]
Grabando a: "domogik-0.5.2.tar.gz"

domogik-0.5.2.tar.gz      [          ] 26,57M  2,22MB/s  en 12s
2017-06-22 20:59:17 (2,13 MB/s) - "domogik-0.5.2.tar.gz" guardado [27862532]
root@Server:/opt/dmg#

```

Figura 4: Descarga de paquete Domogik desde github

Una vez obtenido ambos paquetes se procederá a su descompresión, instalación y configuración de las bases de datos, las mismas son posicionadas en las carpetas correspondientes para ejecutar el archivo install.py.

Posterior a la instalación se desplegará un mensaje de confirmación, indicando que todo se ha procedido correctamente. Finalmente, el paquete domogik debe ser configurada la base de datos donde se van almacenar los resultados de las muestras obtenidas por los sensores.

Finalizado el proceso de instalación de los paquetes, es recomendable su actualización para evitar problemas futuros con la configuración de los plugins requeridos.

Entre los paquetes requeridos para la ejecución del núcleo del sistema de gestión es necesario además instalar un servidor mqtt, más conocido como <<mosquitto>>. En línea se puede encontrar información referente a la instalación. Después de la instalación del paquete, se procede a ejecutarlo.

Este servidor se encarga de recibir las muestras obtenidas por el convertidor ADC del módulo wifi ESP8266 para luego almacenarlos en la base de datos ya creada. Debido a que se trabaja con un sistema base para implementar el portal web de gestión, se debe instalar los plugins. Entre ellos está el client plugin-mqtt-server hasta domogik server. Este paquete es externo y abierto, podemos conseguir los detalles desde el github.

Ahora, el sistema de gestión se encuentra con una base de datos lista para ser configurada, pero como se va trabajar con un portal web se requiere la instalación de un paquete llamado

<<domoweb>>. Este permitirá la administración y configuración del panel en el cual se reflejan de manera gráfica cada uno de los sensores, los cuales fueron previamente configurado. La figura 5 muestra las líneas de código en resumen que va permitir la obtención del paquete en línea y su instalación.

```
$ cd /opt/dmg
$ wget --content-disposition https://github.com/domogik/domoweb/archive/0.5.2.tar.gz
$ tar xvzf domoweb-0.5.2.tar.gz
$ ln -s domoweb-0.5.2 domoweb
$ cd domoweb
$ sudo ./install.py
```

Figura 5: Instalación de paquetes requeridos para interfaz web. Elaboración propia

También, es necesario ejecutar los servicios y proceder a comprobar que todos ellos estén funcionando correctamente. Para lo cual es necesario que desde un PC a cualquier punto de la red se pueda acceder al portal <<backend>> y a <<frontend>> iniciando el navegador y direccionando a la IP del servidor. En la figura 6 se presenta el procedimiento de acceso al backend, lo cual es necesario para identificarse al sistema de administración con las credenciales creadas en la instalación de domogik.

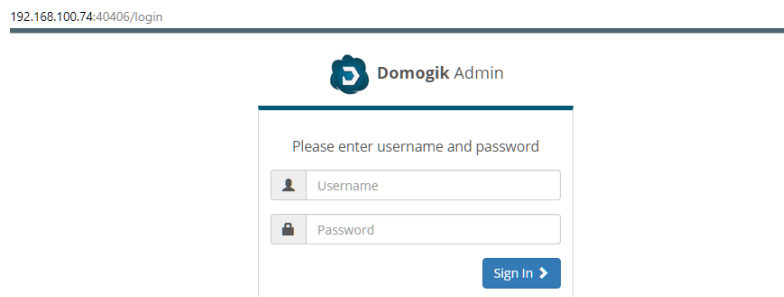


Figura 6: Inicio de sesión para administrar desde el backend.

Posterior al ingreso del backend (parte administrativa del portal web), se procede a comprobar que los plugins son mostrados a nodos sensores. Esta parte es necesaria habilitarla, si es posible el reseteo del servidor mosquito, y agregar los objetos que en este caso serán los sensores.

Como es posible observar, en los parámetros globales del objeto se deben configurar, por ejemplo, el parámetro de QoS (en 1) y asignar una variable `<<ha/_temperature1>>` como string. Del código de programación anterior del módulo wifi ESP8266, observamos que se asigna el mismo nombre al servidor mqtt, en este contenedor se almacenan los resultados de las muestras del sensor. Si se requiere crear nuevos objetos, es necesario ir al apartado del cliente mqtt e indicar el tipo de datos a recibir. En consecuencia, se presenta la pantalla de creación de un nuevo dispositivo.

Existen una limitada cantidad de tipos de datos que se pueden recibir en el servidor, y estos administrarlos adecuadamente con la base de datos. Además, es posible realizar un `<<timeline>>`, que sirve para mostrar los resultados recibidos, guardarlos y visualizarlos en la interfaz gráfica.

A simple vista la interfaz de administración no es nada amigable hacia el usuario u operador de la central de monitoreo de la WSN. Es por esto que se configura el `<<frontend>>` del sistema de gestión, para lo cual escogemos el navegador IP del servidor y el puerto 40404.

Vale reconocer los puntos más importantes del portal de gestión de WSN. En la esquina inferior izquierda se muestra el botón de paneles o secciones, es en ella que se agregan las pantallas necesarias para la gestión del centro de monitoreo

Los resultados de la configuración y despliegue de la interfaz gráfica web están a la vista en la figura 7, aunque se puede implementar cualquier tipo de señalización para representar n sensores. Además, se puede incluir cuadros estadísticos de los datos obtenidos en un determinado rango de tiempo que pueden variar entre semanas, días y horas, tal como se muestra en la figura 8.

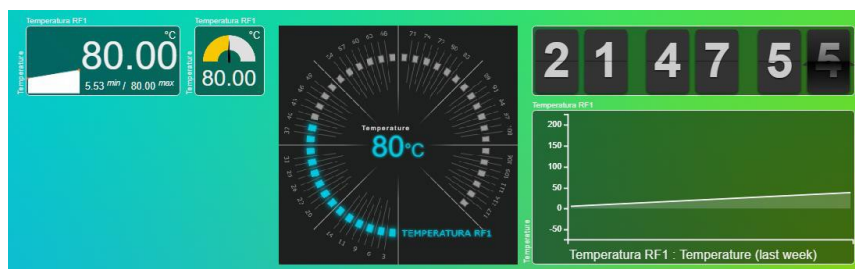


Figura 7: Interfaz gráfica de resultados de sensor de temperatura simulada

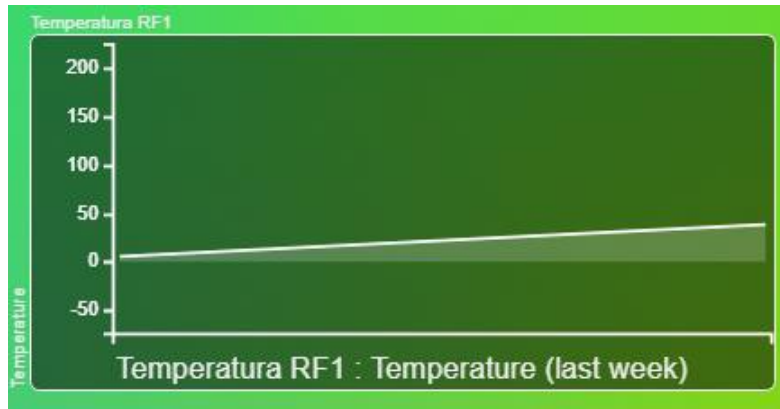


Figura 8: Dato estadístico de las muestras tomadas en días.

Además, es muy importante especificar qué tipo de sensor se utilizará en el campo ya que la interfaz necesita dicha información para especificar la unidad con la que estará trabajando, así como en la figura 9 muestra los datos obtenidos para el sensor de presión.

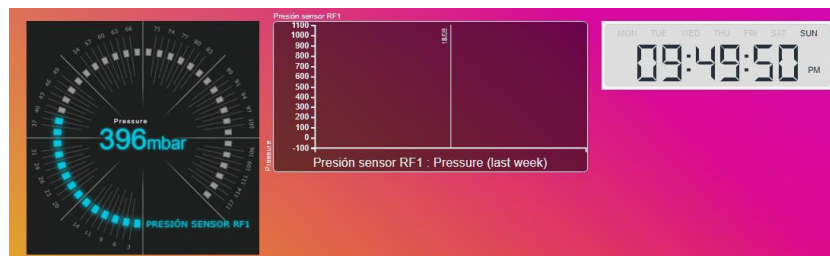


Figura 9: Interfaz gráfica de resultados de sensor de presión.

Así como en el panel de temperatura, este sistema gráfico web es capaz de acumular información por tiempo ilimitado y mostrarla en una tabla estadística.

Conclusiones

Este sistema de gestión implementado a pequeña escala para su demostración del funcionamiento puede reducir costos y ser a la vez muy eficiente ya que el tiempo de respuesta de la transmisión de los datos al servidor es inmediata, si existe un problema de conectividad de estos a la red automáticamente nos indica una alarma de fallo de conexión.

Referencias bibliográficas

Azuero, M. E. (2016). Repositorio univesidad nacional de loja. Obtenido de “DESARROLLO DE UN PROTOTIPO DE RED:
<http://dspace.unl.edu.ec/jspui/bitstream/123456789/10904/1/Ochoa%20Azuero,%20Mar%C3%ADa%20Elena.pdf>

Maya, E. (2014). Red WSN. Obtenido de
<http://repositorio.utn.edu.ec/bitstream/123456789/3526/2/04%20RED%20031%20ARTICULO%20TECNICO%20ESPA%C3%91OL%20-%20INGLES.pdf>

Palacios Meléndez, E., & Cordova Rivadeniera, L. (2015). Sistema de adquisición de señales digitales. Redalyc, 11-17.

Petersen, S. C. (diciembre de 2011). WirelessHART versus ISA100.11a: the format war hits the factory floor. IEEE Ind. Electronics Mag, 5(4), 23-34.

Sabando C., J. (2011). Fundamentos de Comunicaciones Inalámbricas. México: Pearson Educación.