

Integración de una red de sensores con una plataforma IoT para control inteligente de aulas

Lucas Gómez D’Orazio ⁽¹⁾, Santiago Medina ⁽¹⁾ , Diego Montezanti ⁽¹⁾ 

¹ Instituto de Investigación en Informática LIDI (III-LIDI),
Facultad de Informática, Universidad Nacional de La Plata – Comisión de Investigaciones
Científicas de la Provincia de Buenos Aires

lucas.dorazio@alu.ing.unlp.edu.ar
{smedina,dmontezanti}@lidi.info.unlp.edu.ar

Resumen. Los recientes avances en las tecnologías de conectividad y sensado constituyen un escenario propicio para las soluciones basadas en IoT. Actualmente existe una variedad de herramientas que ayudan a explotar el potencial de estas soluciones, entre ellas las Plataformas IoT, complementadas con niveles de procesamiento cercanos a los bordes de la red. En este artículo se describe la implementación de un sistema para monitoreo y control inteligente de parámetros de interés en un edificio universitario basado en una plataforma IoT, y se brindan los resultados preliminares de la funcionalidad desplegada y las pruebas de conectividad realizadas entre los nodos sensores y el servidor.

Palabras claves: Plataforma IoT, Edge y Fog Computing, CoAP, MQTT, monitorización, control inteligente

1 Introducción y objetivos

En los últimos años, la Internet de las Cosas (IoT) ha ganado protagonismo debido a su característica de permitir la conexión de una gran cantidad de dispositivos sensores que obtienen información del entorno y enviarla a servicios en la nube para su procesamiento [1]. Esto se ha utilizado extensamente para desarrollar aplicaciones inteligentes, como gestión de tráfico, hogares inteligentes, monitoreo de fenómenos naturales y salud humana [2].

Debido a la creciente popularidad de IoT, la cantidad de dispositivos conectados a Internet se ha incrementado significativamente. Como resultado, se genera una enorme cantidad de tráfico de red, lo que conduce a cuellos de botella y puede producir limitaciones respecto a las latencias de comunicación con el *cloud* y al ancho de banda de la red [3], por lo que las infraestructuras tradicionales basadas en el *cloud* no resultan suficientes para las demandas actuales de las aplicaciones de IoT [4]. Para lidiar con estos inconvenientes, en los últimos años han surgido los paradigmas de *Fog Computing* y *Edge Computing*, que alivian estas sobrecargas trasladando parte de la potencia de cómputo cerca de los bordes de la red y lejos de los servidores centrales en la nube [1]. De esta forma, el cómputo de los datos de IoT se distribuye, permitiendo reducir las latencias de comunicación [3].

Una plataforma de IoT es un conjunto de servicios o sistemas de software que trabajan conjuntamente con nodos sensores y actuadores conectados mediante Internet. La plataforma brinda las herramientas para capturar, almacenar, procesar y presentar los datos obtenidos de los sistemas embebidos, haciendo uso de protocolos de comunicación IoT. Para ello, debe proporcionar servicios de administración de nodos finales, administración de redes y conectividad, procesamiento y análisis de datos, desarrollo de aplicaciones, seguridad, almacenamiento en bases de datos, herramientas de visualización, monitoreo y control de acceso, entre otros [4][5].

La funcionalidad de una plataforma de IoT viene dada por la interacción de sus seis bloques constitutivos: el bloque de identificación, el bloque de sensado, el bloque de comunicación (que contiene los protocolos que sirven para intercambiar datos entre los objetos conectados y el sistema de administración), el bloque computacional, el bloque de servicios y el bloque semántico [6].

También, dependiendo de la funcionalidad necesaria para una aplicación particular, se puede desplegar una solución a medida que implemente los servicios necesarios, como el almacenamiento, la visualización o la analítica de datos, prescindiendo de la utilización de una plataforma de IoT comercial.

Este trabajo se enmarca dentro de la Práctica Profesional Supervisada del alumno avanzado en la carrera Ingeniería en Computación Lucas Gómez D'Orazio. En el mismo se describen y analizan los resultados iniciales de la integración de una red de sensores con una plataforma de IoT específica (ThingsBoard [7]), y con un conjunto de servicios individuales, en la búsqueda de desplegar un sistema de monitoreo y control inteligente de aulas en un edificio universitario. La experimentación realizada apunta a generar capas de *Edge* y *Fog Computing* para la gestión de parte de la funcionalidad del proyecto.

2 Plataformas, servicios y protocolos

Thingsboard es una plataforma IoT de código abierto que permite almacenar, visualizar y analizar datos con diferentes servicios, como múltiples motores de base datos, generación de alertas, llamados a procedimientos remotos, flujos de trabajo basados en eventos y diseño de tableros dinámicos, entre otros [7]. Su bloque de comunicaciones permite la transmisión de datos entre los dispositivos conectados y el sistema de administración, a través de los protocolos estándar para IoT que se utilizan en la industria, como MQTT, CoAP y HTTP.

CoAP [8] (*Constrained Application Protocol*) es un protocolo basado en HTTP que usa comunicaciones uno a uno, y se utiliza en el hardware de IoT. Debido a esto, debe ser liviano y generar poco tráfico, por lo que utiliza UDP sobre IP. En tanto, MQTT [9] (*Message Queue Telemetry Transport*) es un protocolo de comunicaciones liviano implementado sobre TCP/IP. Usa un servidor *broker* en medio de los dispositivos que se comunican, por lo que no es comunicación M2M. Consiste en tres elementos: publicador, suscriptor y el *broker*. Los clientes publican y se suscriben a tópicos en el *broker*.

En el caso de una red WAN, MQTT se comporta mejor debido a la existencia del *broker* que se encuentra entre los dispositivos que se comunican, por lo que es útil cuando hay ancho de banda limitado. En tanto, CoAP es una buena opción para

aplicaciones basadas en servicios web. Se utiliza cuando los dispositivos necesitan transmitir y recibir a gran velocidad, ya que UDP tiene soporte para *multicast* y *broadcast* [6].

En tanto, en el bloque computacional existen diferentes plataformas de hardware diseñadas para ejecutar específicamente aplicaciones de IoT, como Intel Galileo, Raspberry PI o Arduino. Del mismo modo, hay plataformas de software que proveen las funcionalidades requeridas por las aplicaciones de IoT; entre ellas, están las plataformas a nivel de la nube, que permiten el procesamiento de los datos en tiempo real y ayudan al usuario final a obtener conocimiento extraído de grandes volúmenes de datos.

3 Desarrollo, experimentación y resultados preliminares

El objetivo del trabajo es el despliegue y configuración de una red de sensores de CO2 en el contexto de un edificio universitario cuyas aulas se van a monitorear y controlar remotamente [10]. Los sensores de CO2 en cada aula están conectados vía WiFi a una Raspberry Pi (que corresponde al nivel de Edge), que recolecta los datos de cada nodo a través de peticiones HTTP y los pre-procesa, identificando los valores provenientes de cada sensor, adaptándolos al formato adecuado (JSON) y encapsulándolos como mensajes MQTT para enviarlos al servidor ThingsBoard, tal como se muestra en la Figura 1. Este servidor centraliza el monitoreo de los niveles de CO2 de todo el edificio, evaluando la coherencia de los valores recibidos y configurando el motor de reglas de la plataforma para emitir alertas en caso de valores no deseados. Un ejemplo del panel de control y visualización (o *dashboard*) de ThingsBoard se puede observar en la Figura 2. Adicionalmente, en el nodo de nivel de Edge se desplegaron servicios individuales de base de datos de series temporales (Influx DB) y visualizador de datos (Grafana) que brindan alternativas locales a algunas de las funcionalidades provistas por la plataforma de IoT.

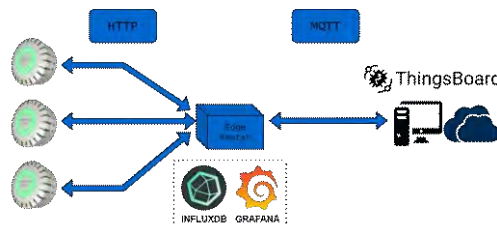


Fig 1. Arquitectura desplegada para la medición de CO2

De las pruebas realizadas con MQTT surge que el mínimo intervalo de tiempo entre envíos de mensajes que garantiza la recepción y el procesamiento de todos los mensajes es de 1.3 milisegundos. Al aumentar la frecuencia de envío, comienza a haber pérdida de mensajes. Sin embargo, las cantidades absolutas de mensajes recibidos continúan en incremento, hasta alcanzar el límite de frecuencia de envío de un mensaje cada 0.5 milisegundos. A partir de ese valor, la plataforma ya ha alcanzado la saturación en la recepción de mensajes, por lo que si se continúa con la reducción del intervalo,

comienzan a disminuir los mensajes recibidos, además del esperable aumento en las tasas de pérdidas. Las cantidades de mensajes recibidos y perdidos se observan en la Figura 3.

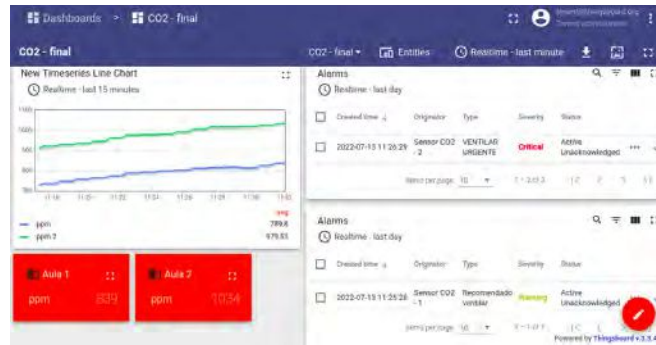


Fig. 2. Panel de monitoreo y control de ThingsBoard.

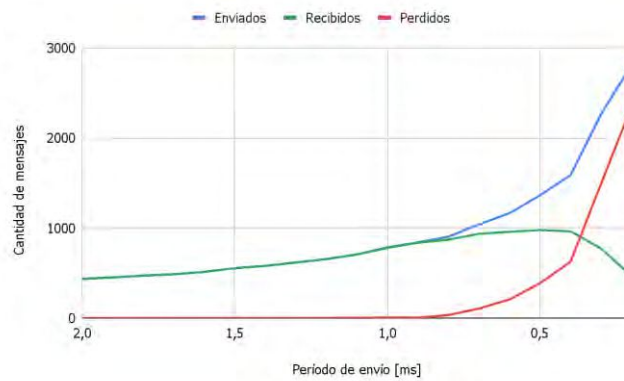


Fig. 3. Relación entre mensajes MQTT recibidos y perdidos al reducir el intervalo entre envíos.

Por último, es importante mencionar que para llegar a los máximos valores en la recepción es necesario desactivar el Algoritmo de Nagle de la comunicación TCP. Al enviarse cada mensaje en un paquete individual, se garantiza que se respeta el intervalo de tiempo que la plataforma requiere para procesar las recepciones, lo cual no ocurre cuando varios mensajes MQTT se encapsulan en un único paquete TCP.

4 Conclusiones y líneas futuras

En este trabajo, en vías de desarrollo, se han alcanzado algunos primeros resultados satisfactorios. Se ha logrado desarrollar una solución de IoT para el monitoreo de CO2 en espacios cerrados. Debido a la baja cantidad de recursos de infraestructura

necesarios para el despliegue, consideramos que la solución implementada es ligera, viable y escalable.

La plataforma de IoT seleccionada para nuestra solución (ThingsBoard) provee una variedad de servicios (integrados entre sí) que facilitan el desarrollo de proyectos. En este desarrollo se exploraron los servicios de capa de transporte, el motor de reglas y las funcionalidades de almacenamiento y visualización de datos. Además, se evaluaron las limitaciones de la plataforma para recibir mensajes en función de la cantidad de memoria asignada a la cola de mensajes, así como también el máximo valor aceptable de frecuencia de envío para los requerimientos de la solución.

En cuanto a los trabajos futuros con la plataforma, se planea investigar su integración con otros servicios externos capaces de potenciar las soluciones IoT, como también investigar los beneficios y desventajas de utilizar una versión reducida orientada al *edge computing*. Además, en el presente trabajo se utilizó la versión de ThingsBoard basada en una arquitectura monolítica, por lo que está pendiente probar la versión basada en microservicios.

Respecto a las comunicaciones, resulta necesario realizar pruebas orientadas a hallar una relación de compromiso entre la utilización eficiente de la red (relacionada con el empaquetamiento de mensajes) y la maximización de la tasa de recepción. Además, hemos comenzado a realizar pruebas de comunicación entre los nodos sensores y la plataforma mediante CoAP, cuyos resultados nos permitirán comparar el desempeño del sistema con ambos protocolos.

Por último, y continuando con la línea de trabajo planteada en [11], se planifica ampliar la funcionalidad del sistema, agregando en las aulas nodos para la medición y control del consumo energético, conectados también al servidor ThingsBoard. De ese modo, se podrá contar con un sistema de monitoreo y control inteligente de parámetros de interés en los edificios objetivo, implementado como una solución IoT.

Referencias

1. Mohan, N., & Kangasharju, J. (2016, November). Edge-fog cloud: A distributed cloud for internet of things computations. In 2016 Cloudification of the Internet of Things (CIoT) (pp. 1-6). IEEE.
2. Tong, Y., Tian, L., Lin, L., & Wang, Z. (2020). Fault Tolerance Mechanism Combining Static Backup and Dynamic Timing Monitoring for Cluster Heads. *IEEE Access*, 8, 43277-43288.
3. Karagiannis, V., Desai, N., Schulte, S., & Punnekkat, S. (2020). Addressing the node discovery problem in fog computing. In 2nd Workshop on Fog Computing and the IoT (FogIoT 2020). Schloss Dagstuhl-Leibniz-Zentrum für InformatikAuthor, F.: Article title. *Journal* 2(5), 99–110 (2016).
4. Ullah, M., Nardelli, P. H., Wolff, A., & Smolander, K. (2020). Twenty-one key factors to choose an iot platform: Theoretical framework and its applications. *IEEE Internet of Things Journal*, 7(10), 10111-10119.
5. M. Fahmideh and D. Zowghi, "An exploration of IoT platform development", *Information Systems*, vol. 87, p. 101409, 2020.
6. Hejazi, H., Rajab, H., Cinkler, T., & Lengyel, L. (2018, January). Survey of platforms for massive IoT. In *2018 IEEE international conference on future IoT technologies (future IoT)* (pp. 1-8). IEEE.
7. ThingsBoard Homepage, <https://thingsboard.io>, accedido el 2022/08/20.

8. CoAP Homepage, <https://coap.technology/>, accedido el 2022/08/20.
9. MQTT Homepage, <https://mqtt.org/>, accedido el 2022/08/20.
10. De Antueno, J., Medina, S., De Giusti, L., & De Giusti, A. (2020). Analysis, Deployment and Integration of Platforms for Fog Computing. *Journal of Computer Science and Technology*, 20(2), e12-e12.
11. Medina, S., Montezanti, D. M., Gomez D'Orazio, L., Compagnucci, E., De Giusti, A. E., & Naiouf, M. (2022). Incorporating Resilience to Platforms based on Edge and Fog Computing. In *X Jornadas de Cloud Computing, Big Data & Emerging Topics* (La Plata, 2022).