

Arquitectura Cloud-IoT para el monitoreo de la calidad del aire en interiores



Colaboración

Orlando Valentín Javier Silva; Daniel Fajardo Delgado; Ramón Chávez Bracamontes; Raúl Rentería Toriz; Fernando Mancilla Peña; María Guadalupe Sánchez Cervantes, Tecnológico Nacional de México / Instituto Tecnológico de Ciudad Guzmán

Fecha de recepción: 13 de marzo de 2024

Fecha de aceptación: 11 de julio de 2024

RESUMEN: Actualmente, la atención hacia la calidad del aire en espacios interiores (IAQ, por sus siglas en inglés) se ha vuelto importante dado su impacto directo en la salud y el bienestar de las personas. En respuesta a esta creciente inquietud, en este artículo se presenta una arquitectura Cloud-IoT que fusiona tecnologías de Internet de las cosas y cómputo en la nube para el monitoreo funcional de la IAQ. Esta arquitectura integra dispositivos equipados con sensores de variables ambientales (temperatura, humedad, dióxido de carbono y compuestos orgánicos volátiles) que en conjunto permiten evaluar la IAQ. Adicionalmente, la arquitectura implementa servicios en la nube para un procesamiento/almacenamiento seguro y escalable de los datos generados por los dispositivos. Asimismo, la arquitectura contempla una aplicación móvil que permite a los usuarios administrar uno o más dispositivos y acceder a los servicios en la nube. La aplicación también permite a los usuarios visualizar la IAQ en tiempo actual o a través de históricos de datos de sus dispositivos.

PALABRAS CLAVE: Aplicación Móvil, Calidad del Aire en Interiores, Cloud-IoT, Cómputo en la Nube, Internet de las Cosas.

ABSTRACT: Currently, attention towards Indoor Air Quality (IAQ) has become important given its direct impact on people's health and well-being. In response to this growing concern, this article presents a Cloud-IoT architecture that fuses Internet of Things and cloud computing technologies for functional IAQ monitoring. This architecture integrates devices equipped with sensors of environmental variables (temperature, humidity, carbon dioxide, volatile organic compounds) that together allow the IAQ to be evaluated. Additionally, the architecture implements cloud services for secure and scalable processing/storage of the data generated by the devices. Likewise, the architecture includes a mobile application that allows users to manage one or more devices and access cloud services. The application also allows users to view the IAQ in current time or through historical data from their devices.

KEYWORDS: Cloud-IoT, Cloud Computing, Indoor Air Quality, Internet of Things, Mobile App.

INTRODUCCIÓN

En la actualidad, la relevancia de la calidad del aire en entornos interiores se erige como un tema de vital importancia, influyendo directamente en la salud y el bienestar de las personas. A medida que la sociedad moderna pasa la mayor parte de su tiempo en espacios cerrados, la comprensión profunda de los factores que influyen en la calidad del aire se torna esencial [1]. Recientemente, estudios como los realizados en [2] y [3], sugieren que la IAQ es uno de los factores vinculados al incremento de la tasa de fatalidad de enfermedades respiratorias como el COVID-19. A medida que la comunidad científica profundiza en estos vínculos, surge la necesidad de un monitoreo eficiente y escalable de las variables ambientales que componen la IAQ. Es en este contexto que la tecnología de Internet de las cosas (Internet of Things, IoT) se presenta como una herramienta útil, permitiendo la recopilación de datos ambientales y ofreciendo información crítica para informar políticas de salud pública [4].

El IoT consiste en un conjunto de dispositivos interconectados entre sí para medir y transferir información sin necesidad de una intervención hu-

mana constante [5]. La información que se recaba de estos dispositivos generalmente requiere de una gran demanda de servicios de almacenamiento, procesamiento y gestión de servicios en los centros de datos basados en la nube [6]. A la integración de los servicios de cómputo en la nube con el IoT se le conoce como Cloud-IoT [7, 8]. Adicionalmente, el uso de aplicaciones móviles para la interacción con sistemas Cloud-IoT permite a los usuarios tener un acceso ubicuo y pertinente. La computación móvil en la integración con el Cloud-IoT permite la portabilidad, ubicación, conocimiento y accesibilidad de los datos para los usuarios [9].

El presente trabajo propone una arquitectura Cloud-IoT para el monitoreo efectivo, escalable y confiable de la IAQ, así como el almacenamiento y acceso a los datos a través de una aplicación móvil. A diferencia de otros escenarios de implementación del monitoreo de la IAQ en México, que se centran en el uso de tecnología IoT y su interacción con algunos servicios en la nube, el presente trabajo explora la integración e interoperabilidad de ambas tecnologías permitiendo un efecto positivo contra los problemas de seguridad, escalabilidad, gestión y mantenimiento.

MATERIAL Y MÉTODOS

Metodología

La arquitectura propuesta se desarrolló a través de dos etapas principales. La primera etapa se enfocó en el análisis y diseño de la integración de los dispositivos IoT con los servicios en la nube, así como en la selección de las tecnologías implicadas. La segunda etapa consistió en la implementación de la arquitectura, lo que requirió la configuración detallada de los dispositivos, la conexión efectiva a los servicios en la nube y la funcionalidad de las aplicaciones de software para el sistema.

La primera fase consideró aspectos claves como la integración segura, efectiva y escalable de los dispositivos con los servicios en la nube. En lo concerniente a seguridad, se contempló la autenticación de dispositivos y usuarios utilizando certificados de confianza y conexiones cifradas. Para la lograr la efectividad en la comunicación entre dispositivos y los servicios en la nube, se planteó el uso de protocolos de mensajería ligeros diseñados para minimizar el ancho de banda y el uso de recursos. Además, a través de los servicios de la nube, se garantiza la escalabilidad ante el aumento o disminución de la carga de trabajo. Finalmente, se estableció como requisito en el diseño del modelo relacional que cada dispositivo IoT se administre solamente por un usuario, mientras que cada usuario puede tener asignados uno o más dispositivos.

Por otro lado, la segunda fase inició con la programación de los dispositivos IoT para la medición de las variables ambientales y la conectividad con la nube. Adicionalmente, se contempló la capacidad para que el usuario pueda personalizar la configuración de la conexión inalámbrica para su conexión al Internet. Así, cada dispositivo estaría ligado a un usuario en particular. Posteriormente, se configuraron diversos servicios en la nube que, en conjunto, permiten

la virtualización de cada dispositivo y el procesamiento y filtrado a una base de datos no relacional. Finalmente, se utilizó el Modelo-Vista-Controlador para separar operaciones de gestión y análisis de datos (modelo) de la aplicación móvil (vista), y se implementó un servicio Web alojado en la nube (controlador) como intermediario entre ambas. A continuación, se describe la arquitectura propuesta.

Arquitectura Cloud-IoT

La arquitectura Cloud-IoT propuesta se compone de dispositivos, servicios de la nube y una aplicación móvil. La Figura 1 muestra la conectividad de estos tres componentes y su estructura física dentro del sistema. La arquitectura contempla que uno o más dispositivos recolectan información ambiental y la transmiten constantemente a un servidor intermediario o bróker IoT, el cuál es una entidad de la nube con la capacidad de admitir múltiples conexiones y de controlar y redireccionar mensajes eficientemente. El bróker IoT administra la comunicación entre los dispositivos y concentra las mediciones ambientales en una base de datos. Por otro lado, debido a la cantidad de datos masivos que se almacenan en la base de datos, la arquitectura también considera un servicio de la nube para el tratamiento y análisis estadístico de grandes volúmenes de información. Finalmente, el resumen de la información, así como un seguimiento de las mediciones actuales se despliegan a través de un servicio Web para su visualización en una aplicación móvil. A continuación, se describe la implementación de cada uno de los tres módulos de la arquitectura propuesta.

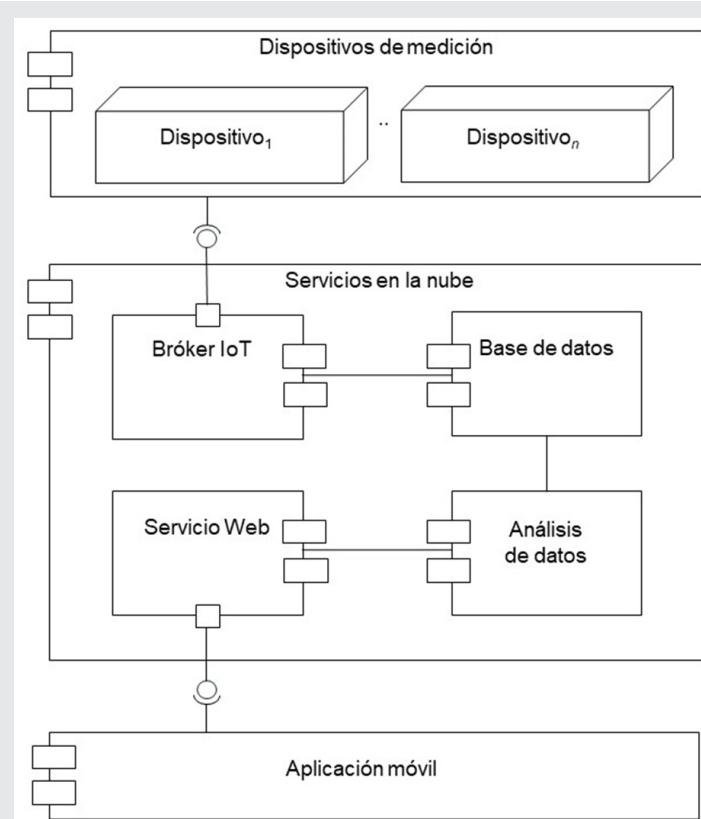


Figura 1. Arquitectura Cloud-IoT.
Fuente: Elaboración propia.

Dispositivos de medición

Son dispositivos de medición que se componen principalmente de un sistema embebido ESP32, que incluye un chip SoC de bajo costo y consumo de energía, conexión inalámbrica por Wifi y un conjunto de sensores montados en una tarjeta PCB interconectados como se muestra en la Figura 2. Los sensores pueden variar en cada dispositivo y pueden ser de: humedad, temperatura, ozono (O₃), dióxido de carbono (CO₂) y compuestos orgánicos volátiles (VOC), que en conjunto permiten evaluar la IAQ [10].

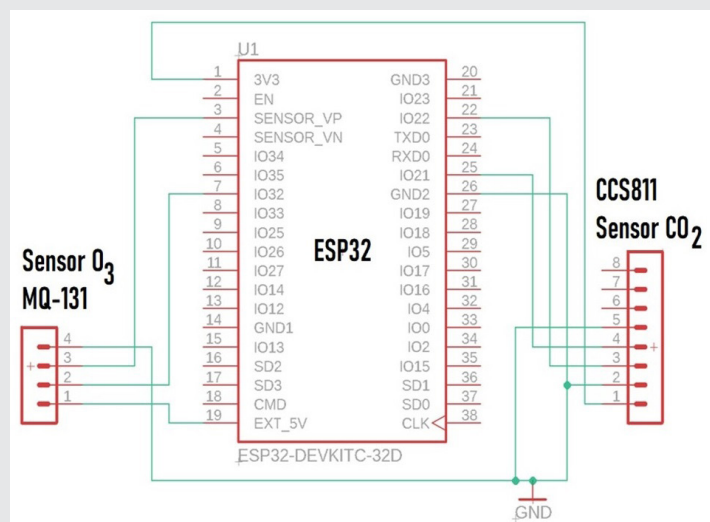


Figura 2. Esquema electrónico de los dispositivos de medición.

Fuente: Elaboración propia.

Cada dispositivo establece una comunicación segura con los servicios de la nube mediante el uso de certificados de confianza y el protocolo de mensajería ligera MQTT (Message Queuing Telemetry Transport). El MQTT coordina los mensajes con base en el modelo publicador/suscriptor, donde cada dispositivo figura como cliente y envía/solicita información al bróker IoT. Los mensajes están compuestos por texto plano definido en formato JSON.

La programación de cada dispositivo se hizo en Arduino, una plataforma de hardware libre y código abierto. Para realizar la conexión de los dispositivos con el bróker IoT, fue necesario implementar algunas librerías adicionales de Arduino para el ESP32. Una de ellas fue WiFiManager, que permite personalizar la configuración de la red inalámbrica del ESP32 sin necesidad de modificar el código del dispositivo. Otra de ellas fue PubSubClient, que permite la conexión y comunicación con AWS IoT Core Broker utilizando MQTT. Finalmente, se incluyó un archivo que contiene el URL del punto final de la nube (o cloud endpoint), el certificado de confianza y una llave privada para la conexión y autenticación del dispositivo.

Servicios en la nube

Se utilizan los servicios de la nube de Amazon Web Services (AWS) para llevar a cabo la implementación del bróker IoT. Se optó por AWS debido a su capacidad para

gestionar grandes volúmenes de datos generados por dispositivos IoT, proporcionando una infraestructura altamente escalable bajo un modelo elástico de precios. En particular, se utilizaron los servicios de AWS IoT Core Broker y AWS Lambda

a) Bróker IoT

Los dispositivos envían las mediciones ambientales en tiempo actual al AWS IoT Core Broker, el cuál administra los mensajes en canales de comunicación, denominados temas o tópicos. El servicio del bróker IoT de AWS [11] tiene la capacidad de administrar hasta billones de dispositivos y trillones de mensajes, y definir para cada dispositivo un canal de comunicación que puede ser bidireccional.

Los dispositivos se autentican con AWS IoT Core Broker a través de certificados x509 mediante conexiones seguras cifradas con el protocolo TLS/SSL (transport layer security/secure sockets layer). Para ello, se define una entidad certificadora que autentica la provisión de dispositivos mediante los servicios just-in-time provisioning (JITP) y just-in-time registration (JITR). El JITP se refiere al proceso de aprovisionamiento de dispositivos justo en el momento en que se conectan al AWS IoT Core Broker; de esta forma, se permite que los dispositivos obtengan su configuración y credenciales de seguridad en el momento de la conexión [12]. Por otro lado, el JITR se relaciona con el registro y virtualización de dispositivos dentro del servicio AWS IoT Core. Cuando un dispositivo se conecta a este servicio puede registrarse automáticamente en lugar de requerir un registro estático previo [13]. Cada dispositivo debe contar con un objeto de virtualización en la nube, que permita acceder individualmente a los temas correspondientes para su monitoreo, ya que ambos tipos de aprovisionamiento (JITP y JITR) buscan simplificar la gestión y la seguridad de los dispositivos IoT, especialmente en implementaciones a gran escala. Finalmente, se utiliza el servicio de AWS Lambda para hacer el registro automático de nuevos certificados bajo una entidad certificadora, y con ello generar el objeto de virtualización. Una vez que el AWS IoT Core Broker recibe la información desde los dispositivos, reenvía los mensajes con una base a través de una regla, una entidad del AWS para filtrar, transformar y redireccionar los mensajes de la cola MQTT. Así, es función del AWS IoT Core Broker reenviar los mensajes de cada canal de comunicación a la base de datos.

b) Base de datos

La arquitectura propuesta considera una combinación de dos tipos de bases de datos, una relacional para la administración de usuarios y dispositivos y otra no relacional (o NoSQL) para el registro histórico de los valores de medición. Se decidió utilizar una base de datos NoSQL de clave-valor con la finalidad de tratar eficientemente y de forma escalable grandes volúmenes de información. La Figura 3 muestra el diagrama entidad-relación de la base de datos, donde el signo de X indica las tablas relacionales con las que interactúan los administradores del

sistema, mientras que el signo + indica las tablas relacionales con las que interactúa el usuario. Por otro lado, la entidad datos_sensados pertenece a la base de datos NoSQL que contiene los históricos de las mediciones ambientales de los dispositivos. Note que existe una relación de la entidad datos_sensados con la tabla de dispositivo.

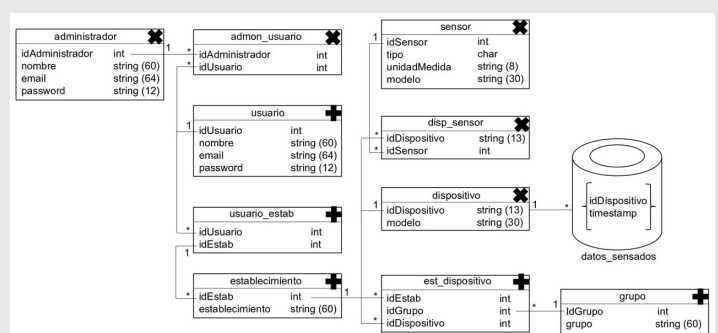


Figura 3: Diseño de la base de datos
Fuente: Elaboración propia.

La base de datos relacional se implementó en una instancia de Amazon RDS [14] (relational database service) donde se creó una base de datos en MySQL para la administración de usuarios y dispositivos, con su respectivo establecimiento (lugares como casas, oficinas, escuelas, etc) y grupo (ubicaciones específicas dentro de los establecimientos). Por otro lado, la base de datos NoSQL se implementó en Amazon DynamoDB [15], que ofrece una flexibilidad en el tamaño de los registros, un rendimiento de alto desempeño para grandes volúmenes de información y escalabilidad automática. En particular, se consideran una clave principal compuesta por los siguientes dos atributos: idDispositivo y timestamp. El atributo idDispositivo es la clave de partición y almacena la dirección física MAC de la tarjeta de red inalámbrica del ESP32. El atributo timestamp es la clave de ordenación y almacena, en un formato numérico, la fecha y hora en que AWS IoT Core Broker recibió el mensaje. El timestamp se refiere al número de segundos que han transcurrido a partir de las cero horas del 1 de enero de 1970 GMT.

c) Análisis de datos

El atributo timestamp es esencial en el análisis de datos, ya que proporciona información sobre el momento en que se registraron los datos. Este análisis es un servicio de la nube que se implementa exclusivamente para la base de datos NoSQL. Este servicio genera estadísticas básicas relacionadas con la IAQ en distintos rangos de tiempo. Actualmente, el análisis de datos se está explorando con el uso de Exponential Smoothing, un método de pronóstico de series de tiempo para datos univariados. Este método puede ampliarse para respaldar datos con una tendencia sistemática o un componente estacional, lo que lo hace adecuado para modelar y predecir patrones en los datos de IAQ a lo largo del tiempo.

d) Servicio Web

El servicio Web opera como un intermediario entre los servicios en la nube y los usuarios. Además de intervenir en las funciones estándar de autenticación, recuperación y envío de datos, este servicio establece una conexión continua con el IoT broker, asegurando una vigilancia constante y actualizada de los dispositivos IoT y sus datos asociados. Esto se integra con el módulo de análisis de datos para obtener estadísticas básicas sobre los datos recopilados, lo que brinda a los usuarios una mejor comprensión de la información obtenida. Esto incluye la generación de informes y gráficos que resumen tendencias y patrones significativos en los datos de la IAQ a lo largo del tiempo.

Aplicación móvil

La arquitectura propuesta contempla una aplicación móvil multiplataforma para que los usuarios interactúen con los servicios en la nube. Dicha aplicación fue desarrollada mediante React Native, un framework de JavaScript de software libre capaz de crear aplicaciones nativas para iOS y Android. La aplicación móvil provee de una interfaz intuitiva para el registro y autenticación de usuarios, la administración de dispositivos y el acceso a la medición en tiempo actual y a datos históricos de los dispositivos.

La comunicación de la aplicación con los servicios de la nube en AWS se lleva a cabo a través del servicio Web, que sirve como intermediario de la aplicación para realizar operaciones de envío y recepción de datos de manera eficiente. Con esto se aprovechan las capacidades de AWS sin depender directamente de las librerías específicamente de React Native. Este enfoque tiene varios beneficios, ya que se reduce la carga en el dispositivo móvil, se optimiza la velocidad de transferencia de datos y se garantiza una experiencia de usuario más fluida.

El servicio Web se aloja en una instancia EC2 [16], la cual ofrece una infraestructura confiable y escalable, asegurando un rendimiento constante incluso en situaciones de carga intensa. Además, al almacenar la lógica de comunicación con los servicios AWS en la instancia de EC2, se simplifica la gestión y actualización de la aplicación, ya que los cambios en la lógica del servidor no requieren modificaciones directas en el código de la aplicación móvil.

RESULTADOS

La implementación de la arquitectura Cloud-IoT demuestra ser funcional. La capacidad de procesamiento distribuido por la infraestructura en la nube proporciona un entorno para la gestión eficiente de datos y recursos. La naturaleza ligera del protocolo MQTT contribuye a la eficiencia del sistema, eliminando la pérdida de información y optimizando la velocidad de transmisión. Adicionalmente, la elec-

ción de una base de datos NoSQL fue una estrategia efectiva para gestionar la enorme cantidad de mediciones recolectada por los dispositivos, así como la flexibilidad y la escalabilidad inherentes a este tipo de bases de datos. También la implementación del servicio Web como intermediario entre la aplicación móvil y los servicios en la nube resultó útil para evitar la saturación de recursos en los dispositivos móviles.

La aplicación móvil sigue en desarrollo, pero cuenta con la mayoría de las funcionalidades. La Figura 4 muestra dos pantallas de la aplicación móvil: el inicio de sesión y la vista principal. El inicio de sesión (Figura 4a) solicita al usuario una cuenta de correo electrónico y una contraseña para autenticarse, las cuales se comparan con la información registrada en las tablas relacionales usuario. Esta pantalla también incluye las funciones de registrar usuario o generar una nueva contraseña en caso de olvidar la anterior. Por otro lado, la vista principal de la aplicación Figura 4b enlista los dispositivos registrados por el usuario y muestra un menú que enlaza a las pantallas de mediciones en tiempo actual, análisis de datos y de configuración.



Figura 4. Pantallas principales de la aplicación móvil; a) inicio de sesión; y b) vista inicial de la aplicación. Fuente: Elaboración propia.

La Figura 5 muestra un ejemplo de mediciones en tiempo actual. Allí se observan los detalles del dispositivo de quien se recibe información: dirección MAC, ubicación y modelo, y las variables ambientales que en ese momento se están recolectando. El valor medido cambiará de color de acuerdo con los rangos definidos para cada variable y que se pueden catalogar en tres niveles: bajo, aceptable o crítico. También, en la parte inferior de estas pantallas, se muestra una tabla con el código de colores que muestra los niveles ideales de las variables ambientales en espacios cerrados acorde a la agencia de protección ambiental de los Estados Unidos de América.

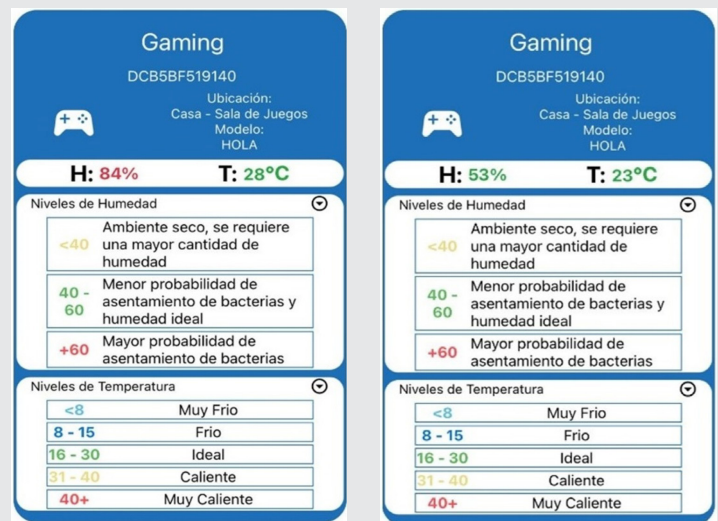


Figura 5. Pantallas de monitoreo con niveles críticos y aceptables de las variables ambientales. Fuente: Elaboración propia.

Finalmente, la Figura 6 muestra un ejemplo de una pantalla del apartado de históricos. Allí se pueden apreciar gráficas en un intervalo de una semana, así como estadísticos básicos por cada variable medida. Esta última pantalla contiene intervalos de tiempo que comprenden horas, semanas, meses y hasta un año por dispositivo.

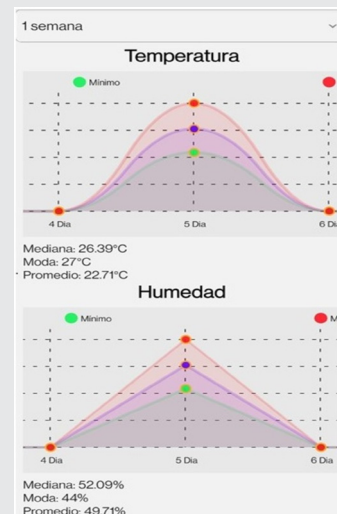


Figura 6. Pantalla de registros históricos. Fuente: Elaboración propia.

CONCLUSIONES

El presente trabajo muestra una arquitectura Cloud-IoT para proporcionar soluciones de monitoreo de la IAQ. Esta arquitectura es eficiente y escalable en términos de los servicios proporcionados por la nube. Además, la arquitectura permite un manejo y control para volúmenes masivos de datos, así como la gestión integral de usuarios y sus dispositivos, unificando todos los servicios involucrados. Adicionalmente se hizo una imple-

mentación de la arquitectura propuesta en los servicios de la nube de Amazon, principalmente por sus capacidades de escalabilidad para proyectos IoT, así como por la gratuidad de sus servicios por un tiempo limitado.

La arquitectura contempla la asignación de uno o varios dispositivos a un usuario, así como la asignación de uno o varios usuarios a un administrador. Como trabajo futuro se prevé la integración de interfaces de voz para dispositivos como Amazon Alexa o Google Home, como parte de una solución para hogares inteligentes. Finalmente, se plantea la inclusión de información de localización por GPS de los dispositivos dentro de la arquitectura, con el propósito de generar mapas de los establecimientos de cada usuario, indicando la ubicación específica de cada dispositivo de medición dentro de dicho entorno.

BIBLIOGRAFÍA

[1] Shengjing, S., Zheng, X., Villalba-Díez, J., Y Ordieres-Meré, J. (2019). Indoor air-quality data-monitoring system: Long-term monitoring benefits. *Sensors*, 4157(19).

[2] Tzampoglou, P., Y Loukidis, D. (2020). Investigation of the importance of climatic factors in covid-19 worldwide intensity. *International Journal of Environmental Research and Public Health*, 17(21).

[3] Ferreira, A., Y Barros, N. (2022). Covid-19 and lockdown: The potential impact of residential indoor air quality on the health of teleworkers. *International Journal of Environmental Research and Public Health*, 19(10), 6079.

[4] Saini, J., Dutta, M., Y Marques, G. (2020). Indoor air quality monitoring systems based on internet of things: A systematic review. *International Journal of Environmental Research and Public Health*, 17(14), 4942.

[5] Nguyen, D., Ding, M., Pathirana, N., Seneviratne, A., Li, J., Niyato, D., Dobre, Poor, O., Y Vincent., H. (2022). 6g Internet of Things: A comprehensive survey. *IEEE Internet of Things Journal*, 9(1), 359-383.

[6] John, R., Kureshi, R., Thakker, D., & Mishra, K. (2021). Internet of Things (IoT) and Indoor Air Quality (IAQ) monitoring in the health domain. En *Procedimientos de la 11va conferencia internacional sobre el Internet de las cosas* (pp. 215-218). St.Gallen, Suiza.

[7] Sadeeq, M., Abdulkareem, M., Zeebaree, R., Ahmed, M., Sami, S., Y Zebari, R. (2021). IoT and Cloud computing issues, challenges and opportunities: A review. *Qubahan Academic Journal*, 1(2), 1-7.

[8] Bouakouk, M., Abdelli, A., & Mokdad., L. (2020). Survey on the Cloud-IoT paradigms: Taxonomy and architectures. En *Procedimientos de la IEEE Symposium on Computers and Communications - ISCC 2020* (pp. 1-6). Rennes, Francia.

[9] Mahbub, M., Hossain, Y Gazi, M. (2020). IoT-Cognizant cloud-assisted energy efficient embedded system for indoor intelligent lighting, air quality monitoring, and ventilation. *Internet of things*, 11, 100266.

[10] Corlan, R., Balogh, R., Ionel, I., Kilyeny, S. (2021). The importance of indoor air quality (IAQ) monitoring. En *Journal of Physics: Conference Series* (Eds.), volume 1781, 012062. IOP Publishing.

[11] AWS (2023). AWS IoT Core endpoints and quotas - Documentación oficial de Amazon Web Services. Obtenida el 5 de diciembre de 2023, de la página electrónica: <https://docs.aws.amazon.com/general/latest/gr/iot-core.html>.

[12] AWS (2023). Just-in-time provisioning. - Documentación oficial de Amazon Web Services. Obtenida el 5 de diciembre de 2023, de la página electrónica: <https://docs.aws.amazon.com/iot/latest/developerguide/jit-provisioning.html>.

[13] AWS (2023). Register a client certificate when the client connects to AWS IoT just-in-time registration. Documentación oficial de Amazon Web Services. Obtenida el 5 de diciembre de 2023, de la página electrónica: <https://docs.aws.amazon.com/iot/latest/developerguide/auto-register-device-cert.html>.

[14] AWS (2023). ¿Qué es amazon relational database service (Amazon RDS)? - Documentación oficial de Amazon Web Services. Obtenida el 5 de diciembre de 2023, de la página electrónica: https://docs.aws.amazon.com/es_es/AmazonRDS/latest/UserGuide/Welcome.html.

[15] AWS (2023). What is amazon DynamoDB? Documentación oficial de Amazon Web Services. Obtenida el 5 de diciembre de 2023, de la página electrónica: <https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Introduction.html>.

[16] AWS (2023). ¿Qué es amazon EC2? - Documentación oficial de Amazon Web Services. Obtenida el 5 de diciembre de 2023, de la página electrónica: https://docs.aws.amazon.com/es_es/AWSEC2/latest/UserGuide/concepts.html.

