

Plataforma integral de monitoreo ambiental inteligente para la toma de decisiones en ciudades sostenibles.

Comprehensive smart environmental monitoring platform for decision-making in sustainable cities.

Oliver Onasis Romero Altamirano (1).
Estudiante del Tecnológico Nacional de México. Instituto Tecnológico de Zacatepec.
20091169@zacatepec.tecnm.mx.

Jazmin Vázquez Hernández (2). Estudiante del Tecnológico Nacional de México. Instituto Tecnológico de Zacatepec,
20091201@zacatepec.tecnm.mx.

Eugenio César Velázquez Santana* (3). Instituto Tecnológico de Zacatepec. eugenio.vs@zacatepec.tecnm.mx.

José Antonio Velázquez Santana (4). Instituto Tecnológico de Zacatepec. jose.vs@zacatepec.tecnm.mx.

José Luis Pérez Estudillo (5). Instituto Tecnológico de Zacatepec. jose.pe1@zacatepec.tecnm.mx.

Gonzalo Rodríguez Mora (6). Instituto Tecnológico de Zacatepec. gonzalo.rm@zacatepec.tecnm.mx.

*corresponding author.

Artículo recibido en septiembre 09, 2026; aceptado en octubre 23, 2026.

Resumen.

Este artículo presenta los resultados de un proyecto de investigación aplicada para el sector de las Tecnologías de la Información y Comunicación, específicamente para el monitoreo ambiental inteligente en ciudades sostenibles. El trabajo de investigación fue realizado con enfoque en el desarrollo de un sistema integral de monitoreo y control ambiental para ciudades inteligentes que combina una aplicación de escritorio en PyQt6 con un sitio web. El sistema está conformado por un módulo Arduino con sensores ambientales, una interfaz moderna con visualización de datos en tiempo real, gráficos interactivos y capacidad de análisis histórico. Los resultados demuestran que el sistema desarrollado tuvo un impacto positivo en la gestión ambiental urbana, permitiendo el monitoreo continuo de condiciones ambientales y la toma de decisiones informadas para mejorar la calidad de vida en entornos urbanos. La plataforma incluye visualización mediante gráficas, mapa y generación de reportes ambientales.

Palabras claves: Análisis de datos, calidad del aire, ciudades inteligentes, internet de las cosas, monitoreo ambiental, sensores ambientales.

Abstract.

This article presents the results of an applied research project for the Information and Communication Technologies sector, specifically focused on intelligent environmental monitoring in sustainable cities. The research was conducted with a focus on developing a comprehensive environmental monitoring and control system for smart cities that combines a PyQt6 desktop application with a website. The system consists of an Arduino module with environmental sensors, a modern interface featuring real-time data visualization, interactive charts, and historical

analysis capabilities. The results demonstrate that the developed system had a positive impact on urban environmental management, enabling continuous monitoring of environmental conditions and informed decision-making to improve the quality of life in urban environments. The platform includes visualization through charts, a map, and the generation of environmental reports.

Keywords: Air quality, data analysis, environmental monitoring, environmental sensors, internet of things, smart cities.

1. Introducción.

Este proyecto es un sistema integral completamente implementado, compuesto por un módulo *Arduino* con sensores ambientales, una aplicación de escritorio en PyQt6 y un sitio *web* funcional para visualización y análisis de datos, que juntos conforman un prototipo que puede contribuir a que cualquier persona pueda conocer las condiciones ambientales del espacio que se encuentra. En los últimos años se han hecho esfuerzos considerables para que el monitoreo ambiental esté al alcance de todos, y es gracias al avance de la tecnología y la ciencia, que se han desarrollado nuevos sistemas de sensores, interfaces accesibles y plataformas de visualización que benefician a la población general, sin necesidad de conocimientos técnicos especializados.

Es por ello, que, con estos avances, se busca democratizar el acceso a la información ambiental mediante un prototipo de sistema automatizado, el cual supervise las condiciones ambientales del entorno, para que cualquier usuario pueda tener información en tiempo real del ambiente que lo rodea, mediante dispositivos con acceso a *Internet* (Mwende, 2025, Tai-hoo *et al*, 2025).

Con este proyecto se busca lograr que las personas tengan acceso a la información relevante de su entorno inmediato, como lo es: la temperatura, humedad, calidad del aire, niveles de ruido e iluminación, para que puedan tomar decisiones informadas sobre su bienestar personal y confort ambiental, ya sea en su hogar, oficina, escuela o cualquier espacio público.

Dado lo anterior, se realizó una búsqueda bibliográfica sobre las tecnologías encaminadas a mejorar el acceso a la información ambiental para la población general, que históricamente ha estado limitada a especialistas o dependía de mediciones manuales que en ocasiones no están disponibles para el ciudadano común; todo lo anterior, puede provocar que las personas tomen decisiones sin tener datos objetivos sobre su entorno, exponiéndose a condiciones que pueden afectar su confort, productividad o salud.

Por ello, nace la iniciativa de elaborar un diseño de un sistema integral automatizado en el que intervengan un microcontrolador, una aplicación de escritorio y una plataforma *web*, que le puedan ayudar a cualquier persona a estar informado de las condiciones ambientales que lo rodean. El sistema automatizado fue pensado para ser desarrollado con una tecnología simple y accesible para el usuario promedio, esto con el fin de buscar eficiencia y funcionalidad completa, es por eso por lo que se utilizó el microcontrolador *Arduino* y la plataforma de programación de *Arduino*, junto con una interfaz moderna en PyQt6. Enseguida, se discutirán aspectos sobre su funcionamiento y algunos fundamentos teóricos relevantes para el proyecto.

En los últimos años, la exploración en el área de la informática ha dado inicio a la innovación en la tecnología del monitoreo ambiental personal, que ofrece a los usuarios la posibilidad de conocer las condiciones de su entorno inmediato, mejorar su confort personal, y tomar decisiones informadas sobre su bienestar. De acuerdo con diversos estudios, el monitoreo ambiental personal es un enfoque accesible, que reúne, maneja y examina datos ambientales de representación temporal y espacial y los combina con información relevante, para apoyar la toma de decisiones personales.

Por esto, el monitoreo ambiental personal es una habilidad moderna, que promueve que las personas puedan entender y controlar su entorno, con ayuda del uso de la tecnología. La tecnología de monitoreo personal mejora la conciencia ambiental del individuo, también su capacidad de respuesta, su confort personal, su salud y su bienestar

general. Con esta tecnología se consigue información a detalle de las particularidades del microambiente personal, optimizando las decisiones individuales, desde la perspectiva del confort, la salud y la productividad personal.

Como lo mencionan diversos estudios, lo que permite el monitoreo ambiental personal es utilizar servicios accesibles y en tiempo real, apoyados en elementos simples como: el tipo de sensor, su ubicación personal, las preferencias individuales, el espacio de vida o trabajo y el nivel de detalle deseado.

Por otro lado, el Internet de las Cosas (*IoT*), es un conjunto de objetos físicos, con sensores incorporados con la finalidad de conectarse a Internet y poder recopilar e intercambiar datos de manera perspicaz con otros dispositivos.

El *IoT* es inducido por un conjunto de procesos y métodos que utilizan software, sensores y otras tecnologías para una comunicación en tiempo real entre procesos y personas.

Al hacer mención del *IoT* que se dirige al monitoreo personal, no sólo se habla de dispositivos móviles como *tablets* o computadoras, sino también incluye una interconectividad de técnicas de comunicación personales en colaboración y compatibles que intercambian datos de varios objetos como sensores ambientales, monitores de confort, sistemas de iluminación y dispositivos de medición. Las tecnologías del Internet de las Cosas que son utilizadas en el monitoreo personal se clasifican en tres niveles o capas: la capa de percepción, la capa de red y la capa de aplicación, las cuales no están sujetas a un modelo específico para el desarrollo de sistemas.

La capa de percepción está compuesta de diversos componentes o dispositivos que se encargan de recabar la información del entorno personal; estos dispositivos son los encargados de modificar las mediciones de los sensores que se implementaron. La capa de red incluye las redes de datos, que otorgan comunicación entre los dispositivos conectados, en esta capa, se declaran los diversos protocolos y medios de red requeridos.

La capa de aplicación es la encargada de la comunicación entre hombre-máquina, es decir, el sistema obtiene directamente los datos del usuario final, ya sean de entrada y/o de salida, con lo cual se mantiene la comunicación directa con el usuario, a través de las capas inferiores. Finalmente, la solución propuesta para el presente proyecto de monitoreo ambiental personal busca integrar estas tecnologías para crear un sistema accesible que permita a cualquier persona conocer y entender las condiciones ambientales de su entorno inmediato.

2. Métodos.

La presente investigación es un estudio aplicado con base cualitativa complementada por evaluación técnica, orientado a diseñar y validar un sistema que cualquier persona pueda usar para consultar condiciones ambientales del lugar donde se encuentra. La recolección de requisitos se llevó a cabo mediante observación directa para identificar variables relevantes (temperatura, humedad, luz, calidad del aire, ruido), frecuencia de consulta y necesidades de visualización y accesibilidad; estos insumos guiaron la definición de funcionalidades y prioridades del sistema.

Para la gestión del desarrollo se empleó *Scrum* con iteraciones menores a cuatro semanas, ejecutando en cada *sprint* actividades de planificación, análisis, diseño, codificación y pruebas. Se priorizaron entregables funcionales y se incorporó retroalimentación de usuarios al *backlog*, con el objetivo de reducir riesgos, acelerar el aprendizaje y asegurar que las funcionalidades respondieran a las necesidades reales de consulta ambiental.

El diseño técnico del *hardware* se basó en *Arduino Uno* y sensores ambientales del *kit Grove*: DHT11 para temperatura y humedad, sensor de luz conectado a 'D2', sensor de gas 'A1', sensor de calidad de aire a 'A0' y sensor de sonido a 'A2', con conversión aproximada a decibelios para facilitar su interpretación.

En el *software* de escritorio se implementó una aplicación en PyQt6 que realiza lectura continua del puerto serial a 9600 baudios, parsea mensajes en formato JSON y muestra visualizaciones en tiempo real mediante gráficas y una vista de "estado del clima". Se incorporó un mapa interactivo con puntos de monitoreo configurables (mediante 'config_puntos.json'), navegación por un *sidebar*, estilos modernos y animaciones, además de simulación

de datos y almacenamiento histórico en memoria con exportación a CSV. En paralelo, se inició un prototipo de sitio web utilizando tecnologías estándar (HTML5, CSS3, JavaScript) y un *backend* en *Python* (*Flask/Django*) previsto para ofrecer acceso remoto a las mismas métricas mediante *endpoints* y autenticación básica.

La comunicación del sistema se basa en mensajes JSON emitidos por *Arduino* y consumidos por la aplicación, garantizando un intercambio de datos simple y robusto. Para persistencia y análisis se probó la subida de históricos hacia una base de datos MySQL mediante un túnel seguro, manteniendo los datos anonimizados; se contempló también el uso de archivos CSV para exportación y portabilidad de registros, facilitando su análisis en herramientas externas.

El protocolo de pruebas incluyó validaciones de lectura continua y resiliencia ante datos faltantes o atípicos, verificación de reconexión del puerto, calibración de rangos operativos (por ejemplo, mapeo del sensor de sonido a decibelios y umbrales de temperatura/humedad), evaluación del rendimiento de las gráficas y del mapa interactivo, así como ejercicios de usabilidad con usuarios finales para medir comprensión de métricas y alertas y ajustar textos, colores y jerarquías visuales.

Las métricas observadas contemplaron latencia de captura y visualización, tasa de actualización estable cercana a 1 Hz en los paneles principales, robustez ante errores de parseo y desconexiones del puerto, precisión relativa frente a instrumentos de referencia en ensayos controlados y satisfacción percibida por los usuarios en tareas de consulta y comprensión de alertas. Estas métricas guiaron ajustes iterativos para mejorar la experiencia y la confiabilidad del sistema.

En materia de ética y privacidad se definió no recolectar datos personales identificables, limitar la recopilación a lecturas ambientales y ofrecer control explícito al usuario sobre la exportación de sus registros. Se documentó de forma transparente el propósito del sistema y el manejo de datos, reforzando la confianza y el uso responsable de la plataforma.

Respecto a limitaciones y reproducibilidad, las pruebas se realizaron con un conjunto específico de sensores *Grove* y *Arduino* Uno, por lo que los resultados pueden variar con otros modelos o condiciones ambientales. Se documentó la estructura del proyecto y sus dependencias para facilitar su réplica, recomendándose mantener versiones consistentes de librerías de *Arduino* y *Python* y seguir las guías de configuración incluidas para asegurar un comportamiento equivalente.

3. Desarrollo.

La información recolectada en tiempo real se obtiene mediante los sensores ambientales conectados a un microcontrolador *Arduino*, enviando lecturas de temperatura, humedad, iluminación, calidad del aire y niveles de ruido en formato JSON por comunicación serial. Estas mediciones se consumen desde la aplicación de escritorio en PyQt6, donde se procesan, visualizan y almacenan para su posterior uso y análisis; adicionalmente, se habilita la exportación a archivos CSV y la integración con una base de datos MySQL para escenarios de persistencia y explotación estadística. El diseño de la arquitectura prioriza la consulta sencilla por parte de cualquier persona, permitiendo solicitar y ver la información del entorno inmediato mediante gráficos en tiempo real y un mapa interactivo con puntos de monitoreo configurables.

Una vez definidos los componentes del sistema y su relación funcional, se construyó el firmware en *Arduino* Uno con sensores del *kit Grove*, realizando calibraciones básicas, conversión del valor crudo de sonido a decibelios y validaciones de rangos operativos antes de emitir los datos. Paralelamente, se implementó la interfaz moderna en PyQt6, con navegación por *sidebar*, vistas de estado del clima y mapa interactivo, estilos oscuros y animaciones, además de un modo de simulación que permite verificar el comportamiento de las visualizaciones cuando no se cuenta con el *hardware* conectado. Esta combinación aceleró la iteración del prototipo, redujo riesgos de integración y permitió afinar la experiencia de consulta ambiental.

Para validar el diseño electrónico de manera segura, se recurrió a la simulación previa de circuitos y flujos de señal, utilizando herramientas que permiten probar la interacción entre microcontrolador y sensores antes de ensamblar físicamente el prototipo.

Este enfoque apoya al desarrollador en la identificación de requerimientos eléctricos, revisión de conexiones y comprobación de la respuesta esperada de los sensores; una vez que el esquema se comporta conforme a lo esperado, se procede a la implementación sobre el *hardware* real, minimizando la probabilidad de daños a los componentes y asegurando un montaje más confiable.

En la capa de *software*, la aplicación de escritorio incorpora lectura continua del puerto serial, manejo robusto de errores y reconexión, actualización de gráficos a una tasa estable cercana a 1 Hz, y almacenamiento histórico para análisis de tendencias. El mapa interactivo, inspirado en plataformas tipo “*Google Maps*”, carga imágenes de fondo y permite configurar puntos de monitoreo con nombre, color, tamaño y posición, ofreciendo una visualización geoespacial clara y adaptable al tamaño de ventana. En paralelo, se avanza en un sitio *web* que replica las principales funcionalidades de consulta y visualización mediante tecnologías estándar (HTML5, CSS3 y JavaScript) y un *backend* en *Python* (Flask/Django), con el objetivo de habilitar acceso remoto seguro a las métricas ambientales.

Finalmente, el sistema contempla flujos de trabajo para subir históricos hacia la base de datos y generar reportes ambientales, así como pruebas de usabilidad con usuarios finales para evaluar comprensión de métricas y alertas. Este ciclo de desarrollo iterativo, sustentado por simulación, prototipado electrónico y validación continua en *software*, permitió consolidar un sistema confiable y accesible que facilita a cualquier persona solicitar y conocer las condiciones ambientales del lugar en el que se encuentra. El sistema está conformado por un microcontrolador *Arduino UNO* que integra sensores de temperatura y humedad (DHT11), calidad del aire (MQ-2), iluminación (LDR + 10k) y ruido (Sound Detector). El *Arduino* realiza el muestreo periódico de cada canal, y publica las lecturas en formato JSON por el puerto serie a 9600 baudios. La aplicación de escritorio en PyQt6 escucha el puerto serie de la computadora, parsea el JSON y reenvía las métricas a un servicio *web* mediante una *Web API REST*, que persiste los datos y los expone a un sitio *web* para consulta, visualización y análisis. La arquitectura se refleja en el diagrama por capas y en el diagrama de cableado disponibles en la Figura 1.

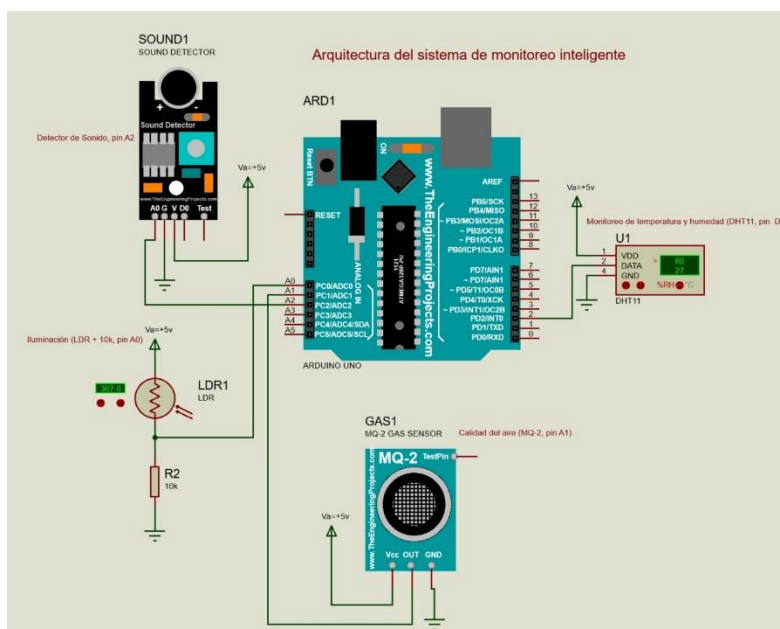


Figura 1. Arquitectura del proyecto en *Proteus*. Elaboración propia.

El *firmware* desarrollado en *Arduino* se programó utilizando el *IDE* oficial, implementando la lectura periódica de los sensores ambientales conectados a los pines analógicos y digitales del microcontrolador. El código realiza el muestreo continuo de temperatura, humedad, iluminación, calidad del aire y nivel de sonido, procesa los valores obtenidos y los envía en formato JSON a través del puerto serial configurado a 9600 baudios.

La conexión entre el *Arduino* y la computadora portátil se establece mediante un cable USB, el cual habilita la comunicación serial a través de un puerto COM. Una vez conectado, la aplicación desarrollada en *Python* detecta el puerto disponible, establece la velocidad de transmisión correspondiente y comienza la lectura continua de los datos enviados por el microcontrolador. De esta manera, el programa en la computadora recibe, interpreta y visualiza la información en tiempo real, garantizando la integración entre *hardware* y *software* dentro del sistema de monitoreo ambiental, como se muestra en la Figura 2.

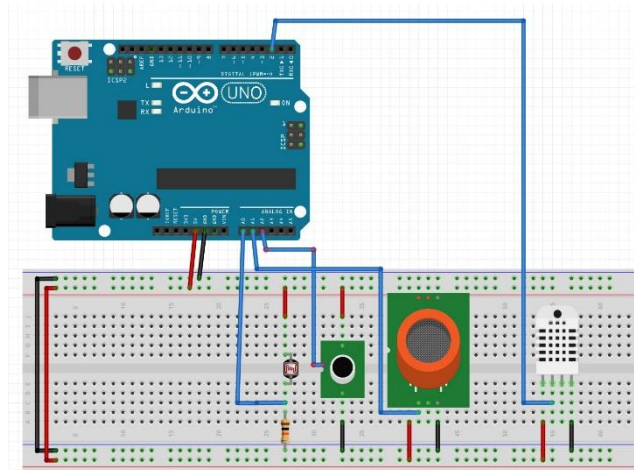


Figura 2. Conexión y programación de sensores.

Una vez realizada la configuración de los sensores en el *Arduino* y establecida la comunicación con la computadora, se procede a verificar la correcta interacción entre el microcontrolador y la aplicación de escritorio desarrollada en PyQt6. Esta aplicación cuenta con diversos módulos que permiten la gestión y visualización integral de la información ambiental.

Entre sus principales interfaces se encuentra el módulo de conexión Figura 3, donde el usuario puede seleccionar el puerto COM y la velocidad en baudios para establecer la comunicación con el *Arduino*. Asimismo, el sistema incorpora un panel principal o *Dashboard* que muestra en tiempo real las variables ambientales como temperatura, humedad, iluminación, calidad del aire y nivel de ruido, integradas en una vista general para facilitar la interpretación global del entorno.



Figura 3. Interfaz de conexión con *Arduino*.

De igual forma, la aplicación dispone de vistas individuales para cada variable, en las cuales se presentan gráficas dinámicas que permiten analizar el comportamiento histórico de los datos. Adicionalmente, se incluye un módulo de mapa interactivo que permite ubicar puntos de monitoreo configurables, así como una sección de análisis y tendencias donde se pueden observar patrones y variaciones a lo largo del tiempo. Estas interfaces permiten al usuario visualizar información relevante de manera clara, organizada y accesible, fortaleciendo la experiencia de monitoreo ambiental en tiempo real. En la figura 4, se muestran gráficamente los datos del sensor de temperatura.



Figura 4. Variable de temperatura en tiempo real.

En la figura 5, se puede apreciar las diferentes variables que se gestionan en la interfaz y de las cuales se extraen los datos para graficarlos en su respectivo *Dashboard*. En esta misma figura, se aprecia si hay conexión con los sensores, además de la escala que se está empleando para graficar los datos recolectados, en algunos casos es porcentaje, en otros grados centígrados, en otros ppm, en otros decibeles (dB).



Figura 5. Análisis en conjunto.

Dentro de las funciones que fueron requeridas, están las de generar reportes por fecha y por tipo de sensor. Estas interfaces permiten al usuario visualizar información relevante sobre las variaciones que se han tenido de cada uno de los sensores, como se puede apreciar en la figura 6.

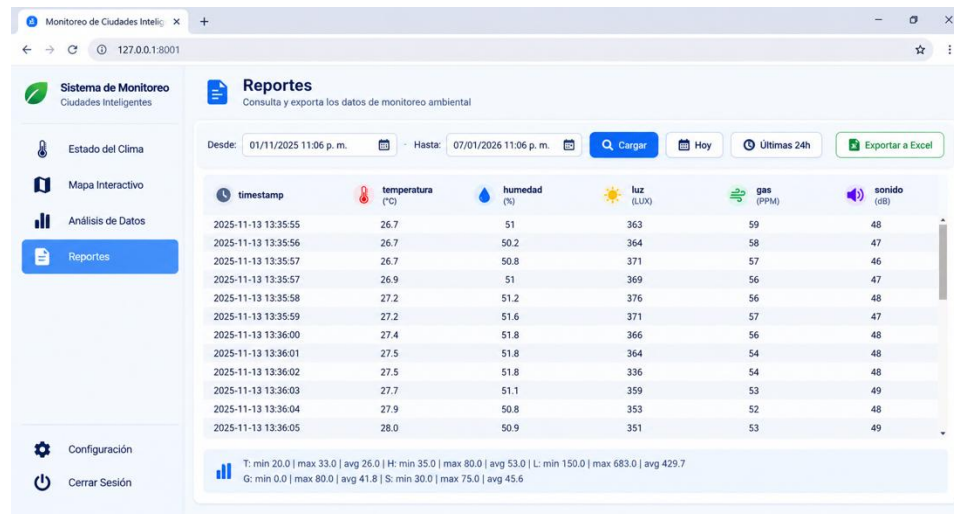


Figura 6. Generador de reportes de los diferentes sensores.

Asimismo, la aplicación web permite al usuario consultar la información ambiental en tiempo real, como es el caso de la figura 7, en donde se puede visualizar registros históricos mediante gráficas dinámicas y acceder a reportes generados por el sistema.

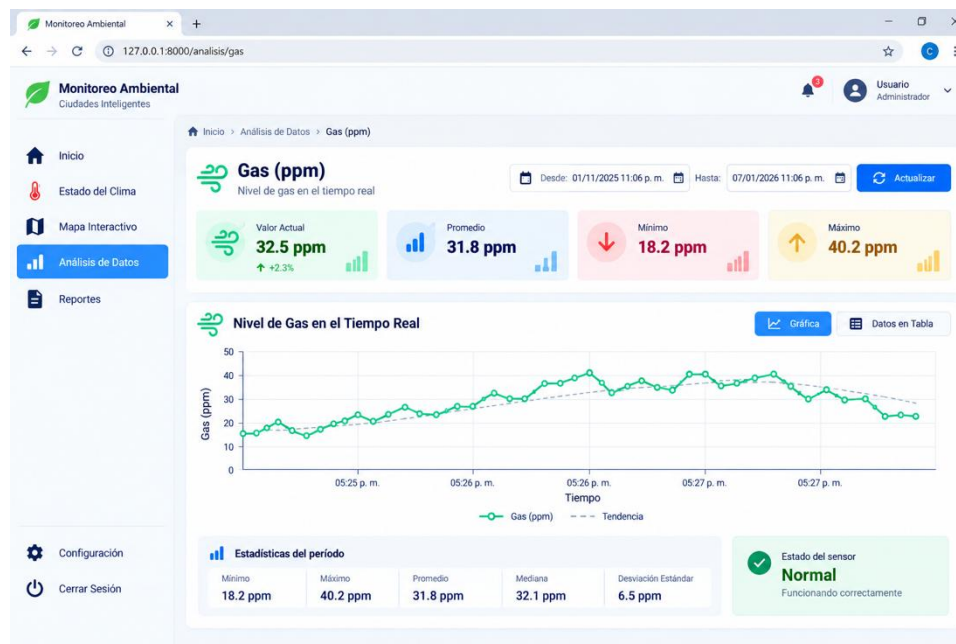


Figura 7. Generador de reportes históricos.

Además, incorpora un módulo de notificaciones que alerta, sobre posibles variaciones fuera de los rangos establecidos. En caso necesario, el usuario puede realizar configuraciones remotas para ajustar parámetros operativos, fortaleciendo así el control y supervisión del entorno monitoreado. El sistema también incorpora un mapa interactivo que muestra los puntos de monitoreo configurados, permitiendo una visualización geográfica de los datos recopilados.

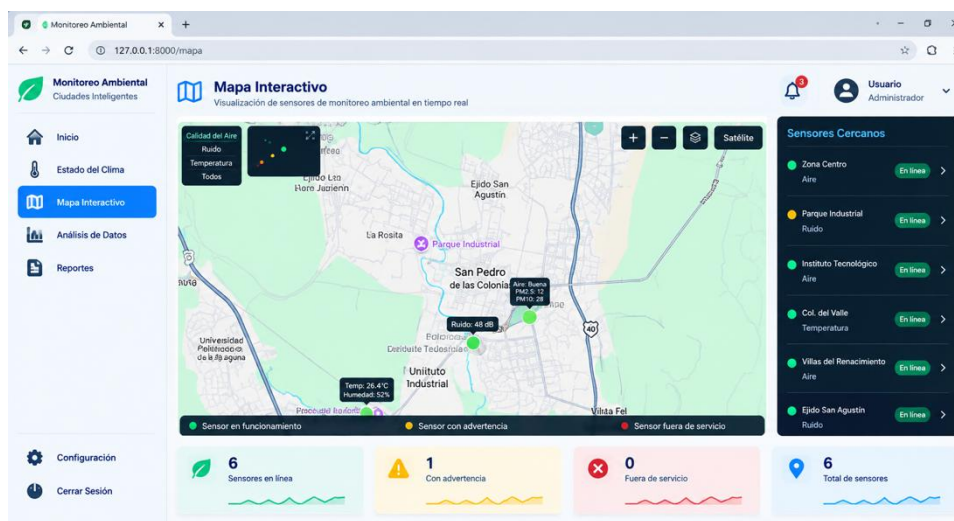


Figura 8. Mapa de ubicación de sensores.

De esta manera, el usuario puede acceder a la información desde cualquier navegador con conexión a *Internet*, fortaleciendo la accesibilidad y el alcance del sistema, como se muestra en la figura 8. El módulo de escritorio actúa como núcleo de adquisición y almacenamiento de datos, mientras que la aplicación *web* funciona como plataforma de consulta remota, integrando ambos componentes en una arquitectura distribuida.

Conclusiones.

El sistema inteligente de monitoreo y control ambiental integra un módulo *Arduino* con sensores, una aplicación de escritorio PyQt6 y un sitio *web* para visualización y análisis en tiempo real, aportando una base sólida para la gestión urbana y el bienestar ciudadano en *smart cities*. La arquitectura *IoT* adoptada —con sensorización distribuida, comunicación eficiente y análisis de datos— se alinea con los principios y beneficios documentados para ciudades inteligentes (Gubbi *et al.*, 2013; Zanella *et al.*, 2014). La visualización inmediata de indicadores, los gráficos interactivos y el análisis histórico permiten identificar tendencias, anomalías y riesgos ambientales, información clave para la toma de decisiones y la salud pública (World Health Organization, 2021).

Además, el uso de *hardware* accesible como *Arduino* y *software* multiplataforma como PyQt6 reduce barreras de costo y despliegue, facilitando la adopción por instituciones y ciudadanía (Arduino, n.d.; Riverbank Computing, 2024). En conjunto, la solución democratiza el acceso a la información ambiental y establece una base técnica para futuras extensiones, como control automático de actuadores y analítica predictiva, manteniendo interoperabilidad y escalabilidad en entornos urbanos (Zanella *et al.*, 2014).

Agradecimientos.

Los autores agradecen al **Instituto Tecnológico de Zacatepec** por el apoyo académico y las facilidades brindadas para el desarrollo del presente proyecto de residencia profesional.

Referencias bibliográficas.

- Arduino.** (n.d.). Arduino Uno Rev3 — Documentation. <https://docs.arduino.cc/hardware/uno-rev3/>.
- Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013).** Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7), 1645–1660. <https://doi.org/10.1016/j.future.2013.01.010>.
- Mwende, W. (2025).** The Internet of things: Smart cities and their infrastructure. Vol. 12(2). Pp. 47-52. <https://www.inosr.net/inosr-scientific-research/>.
- Riverbank Computing.** (2024). PyQt6 Documentation. <https://www.riverbankcomputing.com/static/Docs/PyQt6/>.
- Tai-hoo, K., Ramos, C., & Mohammed, S. (2017).** Smart City and IoT. Vol. 76. Pp. 159-162. <https://doi.org/10.1016/j.future.2017.03.034>.
- World Health Organization.** (2021). WHO global air quality guidelines: Particulate matter (PM2.5 and PM10), ozone, nitrogen dioxide, sulfur dioxide and carbon monoxide. <https://www.who.int/publications/i/item/9789240034228>.
- Zanella, A., Bui, N., Castellani, A., Vangelista, L., & Zorzi, M. (2014).** Internet of Things for Smart Cities. *IEEE Internet of Things Journal*, 1(1), 22–32. <https://doi.org/10.1109/JIOT.2014.2306328>.

Información de los autores.



Oliver Onasis Romero Altamirano, es egresado de la carrera de Ingeniería en Sistemas Computacionales del Instituto Tecnológico de Zacatepec, cursó la especialidad de Desarrollo Web Empresarial. Realizó su residencia profesional con el proyecto “Plataforma de Monitoreo Ambiental con Enfoque en Ciudades Inteligentes-Módulo Front-end”. Su área de interés, es el desarrollo de software web orientado a la Inteligencia Artificial y el análisis de datos para la toma de decisiones empresariales.



Jazmín Vázquez Hernández, es egresada de la carrera de Ingeniería en Sistemas Computacionales del Instituto Tecnológico de Zacatepec, cursó la especialidad de Desarrollo Web Empresarial. Realizó su residencia profesional con el proyecto “Plataforma de Monitoreo Ambiental con Enfoque en Ciudades Inteligentes-Módulo Back-end”. Su área de interés, es el desarrollo de software web orientado a la Inteligencia Artificial y el análisis de datos para la toma de decisiones empresariales.



Eugenio César Velázquez Santana, es Doctor en Planeación Estratégica y Dirección de Tecnología. Cuenta con más de 20 años en la docencia nivel superior y posgrado. Actualmente es Responsable del Cuerpo Académico ante PRODEP con el nombre “Aprendizaje Automático”, además, es Presidente del Consejo de Maestría en Inteligencia Artificial en el Instituto Tecnológico de Zacatepec.



José Antonio Velázquez Santana, es Licenciado en Informática y Maestro en Tecnologías de la Información, egresado del Instituto Tecnológico de Zacatepec. Se ha desempeñado como Profesor de Tiempo Completo en esta institución durante 32 años, contribuyendo a la formación de diversas generaciones de profesionales en el área de las tecnologías de la información. Su trayectoria se distingue por su compromiso con la docencia, la actualización tecnológica y la formación académica de los

estudiantes.



José Luis Pérez Estudillo, es Ingeniero en Sistemas Computacionales egresado del Instituto Tecnológico de Zacatepec en el año 2000. Actualmente es profesor de tiempo completo del departamento de Sistemas y Computación. Su área de interés es la electrónica, redes de computadoras y docencia.



Gonzalo Rodríguez Mora, es Licenciado en Informática, egresado del TecNM/ Instituto Tecnológico de Zacatepec, con una trayectoria profesional de 20 años como docente, impartiendo clases en Ingeniería en Sistemas Computacionales y afines, dando conferencias y cursos de capacitación a los docentes de la misma institución. Su perfil va enfocado a redes de computadoras y lenguajes de programación. Trabajó en el Centro Nacional de Metrología (CENAM), durante 3 año bajo el cargo de Profesional Dictaminador de Servicios Especializados, desarrollando Sistemas de Automatización.