



**UNIVERSIDAD PEDAGOGICA
NACIONAL**

Educadora de educadores

**ACTUALIZACIÓN DEL HARDWARE Y SOFTWARE DEL PORTAL AIRE
CIUDADANO PARA EL MONITOREO DE LA CALIDAD DEL AIRE.**

Darwin Yahir Murillo Gamarra

Universidad Pedagógica Nacional
Facultad de Ciencia y Tecnología
Departamento de Tecnología
Licenciatura en Electrónica
Bogotá, Colombia
2024

**ACTUALIZACIÓN DEL HARDWARE Y SOFTWARE DEL PORTAL AIRE
CIUDADANO PARA EL MONITOREO DE LA CALIDAD DEL AIRE.**

Darwin Yahir Murillo Gamarra

Proyecto presentado como requisito para optar al título de:
Licenciado en Electrónica

Asesor:

Jimmy William Ramirez Cano

**Universidad Pedagógica Nacional
Facultad de Ciencia y Tecnología
Departamento de Tecnología
Licenciatura en Electrónica
Bogotá, Colombia**

2024

Agradecimientos

En primer lugar, quiero agradecerles a todas las personas que me han acompañado y apoyado a lo largo de este camino. Quiero comenzar por mi familia, que ha estado presente en cada momento, brindándome sus consejos, su apoyo y amor, impulsándome a seguir adelante. Agradezco a la vida por darme la oportunidad de vivir este proceso de crecimiento personal y aprendizaje, a las personas de buen corazón que, con pequeños gestos, me ayudaron a avanzar.

Agradezco a la educadora de educadores... la Universidad Pedagógica Nacional y a cada uno de los maestros que han sido parte de mi proceso de formación académica. En especial, extendiendo mi agradecimiento al Dr. Jimmy William Ramírez Cano, mi asesor de proyecto de grado, por su guía y orientación durante este proceso de investigación. También agradezco al movimiento ambiental Aire Ciudadano, el antecedente principal de mi proyecto, que proporciono la inspiración para el desarrollo de esta investigación.

A mis amigos, a mis compañeros de la universidad, quienes estuvieron presente en la mayoría de los retos y dificultades que se presentaron durante nuestra formación, donde su compañía y amistad fueron fundamentales para superar los obstáculos que se presentan en esta experiencia de vida.

Finalmente, agradezco a los maestros de mi licenciatura, al maestro Guillermo Gómez, a la maestra Claudia Yaneth Rodríguez, a la maestra Carol Ivonn Rodríguez, y a todos los maestros con los que tuve la oportunidad de cruzarme en este camino. Agradeciendo los valiosos aprendizajes que obtuve de cada uno de ustedes, donde siempre se notó, que no fueron egoístas con su conocimiento, puesto que siempre me sentí acompañado de la experiencia, dedicación y el conocimiento de parte de todos ustedes.

A todos, gracias por ser parte de este camino dulce, aunque a veces amargo.

Tabla de contenido

| | |
|------------------------------------------------------------------|-----------|
| PLANTEAMIENTO DEL PROBLEMA | 9 |
| ANTECEDENTES | 11 |
| JUSTIFICACIÓN | 14 |
| OBJETIVO GENERAL | 16 |
| OBJETIVOS ESPECIFICOS | 17 |
| MARCO CONCEPTUAL | 17 |
| Diagramas UML | 18 |
| Diagrama de Casos de Uso UML | 19 |
| MARCO TEORICO | 20 |
| PROTOCOLO DE COMUNICACIÓN MQTT | 21 |
| MQTT + BROKER: | 22 |
| INTERNET DE LAS COSAS | 22 |
| GRAFANA | 23 |
| NODE.JS | 23 |
| NODE-RED | 24 |
| INFLUXDB | 24 |
| INDICE DE MEDICIONES DE CALIDAD DEL AIRE | 25 |
| METODOLOGÍA | 31 |
| Fase 1: Definición de análisis de requerimientos | 32 |
| Fase2: Diseño del sistema | 32 |
| Fase 3: Implementación del sistema | 32 |
| Fase 4: Pruebas y evaluación | 32 |
| RESULTADOS | 33 |
| DESARROLLO DE LA FASE 1. ANALISIS Y REQUERIMIENTOS | 33 |
| DESARROLLO DE LA FASE 2. DISEÑO DEL SISTEMA | 38 |
| DISEÑO DEL SISTEMA | 38 |
| Diagrama de Casos de Uso UML | 38 |
| Casos de Uso: | 39 |
| Diagrama de Clases UML | 40 |
| DESCRIPCIÓN DE LA ARQUITECTURA DEL SISTEMA | 41 |
| Sensores y comunicación: | 41 |
| DESARROLLO DE LA FASE 3. IMPLEMENTACIÓN DEL SISTEMA | 45 |
| ESQUEMA DEL SISTEMA DE MONITOREO | 46 |
| LECTURA DE LOS PARAMETROS AMBIENTALES | 46 |

| | |
|-----------------------------------------------------------------------------------------------------|-----------|
| CONFIGURACIÓN DE BRÓKER EMQX | 47 |
| PROGRAMACIÓN Y CONFIGURACIÓN DEL PROTOTIPO | 49 |
| CONFIGURACIÓN EN NODE-RED | 53 |
| CONFIGURACIÓN EN INFLUXDB | 55 |
| CONFIGURACIÓN EN GRAFANA | 58 |
| DESARROLLO DE LA FASE 4. PRUEBAS Y EVALUACIÓN..... | 60 |
| INTEGRACIÓN MODULOS LORA..... | 61 |
| VISUALIZACIÓN DE DATOS EN NODE-RED..... | 63 |
| INTEGRACIÓN DE SENSORES ADICIONALES | 64 |
| CONFIGURACIÓN DE LA ESTACIÓN DE MONITOREO..... | 65 |
| VISUALIZACIÓN DE DATOS EN GRAFANA | 65 |
| CONCLUSIONES..... | 75 |
| RECOMENDACIONES..... | 76 |
| REFERENCIAS..... | 78 |
| ANEXOS | 80 |
| <i>Anexo 1. Topics MQTT implementados en código de programación de la etapa receptora.....</i> | <i>80</i> |
| <i>Anexo 2. Esquema de conexión y sus etiquetas de identificación de la etapa receptora.....</i> | <i>80</i> |
| <i>Tabla. Etiquetas de conexiones etapa receptora.....</i> | <i>80</i> |
| <i>Anexo 3. Esquemático de conexión y etiquetas de identificación de la Etapa Transmisora</i> | <i>81</i> |
| <i>Tabla. Etiquetas de conexión de componentes en la etapa Transmisora.....</i> | <i>81</i> |
| <i>Anexo 4. Tabla de componentes con sus especificaciones, costo y recomendaciones de uso.....</i> | <i>82</i> |
| <i>Anexo 5. Procesos de calibración de los sensores conectados en la etapa transmisora.....</i> | <i>85</i> |
| Calibración de los Sensores MQ..... | 85 |
| Calibración del sensor PMS3003..... | 92 |
| Calibración del sensor SHT31-D | 93 |
| Calibración del sensor CCS811..... | 93 |
| <i>Anexo 6. Configuración para la implementación grafica en Grafana.....</i> | <i>94</i> |
| <i>Anexo 7. Guía para la Configuración para el inicio de InfluxDB.....</i> | <i>95</i> |
| <i>Anexo 8. Guía para el ingreso de las variables de almacenamiento en InfluxDB.....</i> | <i>96</i> |
| <i>Anexo 9. Evidencias de las mediciones realizadas en Ecoparque Sabana.....</i> | <i>96</i> |
| <i>Anexo 10. Evidencias de las mediciones realizadas en la Universidad Pedagógica Nacional.....</i> | <i>98</i> |
| <i>Anexo 11. Archivo para la impresión 3D para la estación receptora y transmisora.....</i> | <i>99</i> |
| <i>Anexo 12. Guía para la implementación y desarrollo del sistema de monitoreo.....</i> | <i>99</i> |

INDICE DE FIGURAS

| | |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| <i>Figura 1. Diagrama del protocolo MQTT + Broker. Fuente: EMQ Technologies</i> | <i>22</i> |
| <i>A Comprehensive Comparison of Open Source MQTT Brokers 2023 by EMQ Technologies Medium..</i> | <i>22</i> |
| <i>Figura 2. Colores asociados al índice de calidad del aire. Fuente: Portal Aire Ciudadano.</i> | <i>30</i> |
| <i>Figura 3. Estructura de la metodología escogida Waterfall. Fuente: Elaboración Propia.</i> | <i>31</i> |
| <i>Figura 4. Diagrama de casos de uso UML. Fuente: elaboración propia, realizado en: https://staruml.io/</i> | <i>39</i> |
| <i>Figura 5. Diagrama de lenguaje de modelado Unificado UML. Fuente: Elaboración Propia.</i> | <i>41</i> |
| <i>Figura 6. Esquema de la etapa receptora y transmisora del sistema de monitoreo.</i> | <i>43</i> |
| <i>Fuente: elaboración propia.</i> | <i>43</i> |
| <i>Figura 7. Esquema grafico del sistema de monitoreo. Fuente: Elaboración propia.....</i> | <i>46</i> |
| <i>Figura 8. Código para la definición de librerías y pines utilizados para medir las variables ambientales. Fuente: Elaboración Propia.</i> | <i>47</i> |
| <i>Figura 9. Código para la obtención de las variables ambientales. Fuente: Elaboración Propia.....</i> | <i>47</i> |
| <i>Figura 10. Usuario de autenticación asignado en el bróker. Fuente: Elaboración Propia.</i> | <i>48</i> |
| <i>Figura 11. Certificado de conexión EMQX. Fuente:https://cloud-intl.emqx.com/console/deployments/nb191fbe/overview</i> | <i>48</i> |
| <i>Figura 12. Conexión con el Bróker MQTT. Fuente: Elaboración Propia.....</i> | <i>49</i> |
| <i>Figura 13. Librerías utilizadas en estación trasmisora del prototipo. Fuente: Elaboración Propia.</i> | <i>50</i> |
| <i>Figura 14. Librerías utilizadas en el programa receptora de la estación. Fuente: elaboración propia. ..</i> | <i>50</i> |
| <i>Figura 16. Configuración para gestionar redes WiFi desde el portal web. Fuente: elaboración propia. .</i> | <i>51</i> |
| <i>Figura 17. Parámetros de configuración del servidor MQTT + Bróker. Fuente: elaboración propia. ...</i> | <i>51</i> |
| <i>Figura 18. Método de envío de datos cada 5 segundos a la etapa receptora. Fuente: elaboración propia.</i> | <i>52</i> |
| <i>Figura 19. Método de envío y publicación de los datos cada 5 segundos. Fuente: elaboración propia... </i> | <i>53</i> |
| <i>Figura 20. Opción en node-RED para añadir y configurar nodos. Fuente: elaboración propia.</i> | <i>53</i> |
| <i>Figura 21. Estado del servicio Node-RED. Fuente: elaboración propia.</i> | <i>53</i> |
| <i>Figura 22. Flujo para la lectura de los sensores en Node-RED. Fuente: elaboración propia.....</i> | <i>54</i> |
| <i>Figura 23. Configuración de usuario inicial en InfluxDB. Fuente: elaboración propia.....</i> | <i>55</i> |
| <i>Figura 24. API generado en InfluxDB. Fuente: elaboración propia.....</i> | <i>56</i> |
| <i>Figura 25. Modificación del tiempo de almacenamiento en InfluxDB. Fuente: elaboración propia.</i> | <i>56</i> |
| <i>Figura 26. Servidor InfluxDB configurado en Node-RED. Fuente: elaboración propia.</i> | <i>57</i> |
| <i>Figura 27. Configuración de nodo para almacenar mediciones en InfluxDB. Fuente: elaboración propia.</i> | <i>57</i> |
| <i>Figura 28. Mediciones de los sensores en InfluxDB. Fuente: elaboración propia.....</i> | <i>58</i> |

| | |
|-----------------------------------------------------------------------------------------------------------------------------------|----|
| <i>Figura 29. Integración del servidor Grafana. Fuente: elaboración propia.</i> | 59 |
| <i>Figura 30. Inicio de conexión como fuente de datos en Grafana. Fuente: elaboración propia.</i> | 59 |
| <i>Figura 31. Conexión a la base de datos InfluxDB desde Grafana. Fuente: elaboración propia.</i> | 59 |
| <i>Figura 32. Selección de datos y configuración de gráficos en Grafana. Fuente: elaboración propia.</i> | 60 |
| <i>Figura 33. Verificación del funcionamiento del protocolo y recepción de datos. Fuente: elaboración propia.</i> | 61 |
| <i>Figura 34. Conexión de nodos de entrada y de salida del protocolo MQTT. Fuente: elaboración propia.</i> | 62 |
| <i>Figura 35. Componentes utilizados para el periodo de integración del protocolo y módulos LoRa. Fuente: elaboración propia.</i> | 63 |
| <i>Figura 36. Visualización grafica por medio de nodos Gauge en Node-Red.</i> | 64 |
| <i>Figura 37. Verificación y visualización en el Dashboard en InfluxDB y Node-RED.</i> | 64 |
| <i>Figura 38. Receptor y trasmisor de la estación de monitoreo.</i> | 65 |
| <i>Figura 39. Visualización grafica de los datos almacenados en InfluxDB desde Grafana.</i> | 66 |
| <i>Figura 40. Verificación de funcionamiento de la visualización en Grafana.</i> | 66 |
| <i>Figura 41. Visualización Grafica de las variables en una Dashboard dentro de Grafana.</i> | 67 |
| <i>Figura 42. Visualización grafica de las mediciones realizadas en Ecoparque Sabana.</i> | 69 |
| <i>Figura 43. Mediciones en tiempo presente de la estación fija Aire Ciudadano.</i> | 69 |
| <i>Figura 44. Mediciones en tiempo presente por parte de la estación de la CAR en el parque Jaime Duque.</i> | 70 |
| <i>Figura 45. Esquemático de conexión etapa receptora.</i> | 80 |
| <i>Figura 46. Esquemático de conexión de etapa Transmisora.</i> | 81 |
| <i>Figura 47. Curva típica de sensibilidad del sensor MQ135.</i> | 85 |
| <i>Figura 48. Curva de calibración del sensor MQ135.</i> | 86 |
| <i>Figura 49. Función para obtener CO₂ en el software de la etapa Transmisora del sistema.</i> | 89 |
| <i>Figura 50. Curva de calibración del sensor MQ7.</i> | 90 |
| <i>Figura 51. Implementación del código de la etapa transmisora.</i> | 92 |
| <i>Figura 52. Función para la lectura de material particulado PM2.5.</i> | 93 |
| <i>Figura 53. Código aplicado en el software del sistema para mediciones de COV.</i> | 94 |

INDICE DE TABLAS.

| | |
|--------------------------------------------------------------------------------------------------------------------------------|-----------|
| <i>Tabla 1. Escala de medición de la EPA (AQI) para concentración de PM2.5.....</i> | <i>26</i> |
| <i>Tabla 2. Escala de medición de la EPA (AQI) para concentración de CO.....</i> | <i>27</i> |
| <i>Tabla 3. Escala de medición del ICA en Colombia para concentración de PM2.5.....</i> | <i>27</i> |
| <i>Tabla 4. Niveles máximos permisibles de contaminantes criterio en el aire</i> | <i>28</i> |
| <i>Tabla 5. Niveles máximos de contaminantes en el aire permisibles para el año 2030.....</i> | <i>29</i> |
| <i>Tabla 6. Concentraciones ($\mu\text{g}/\text{m}^3$) para los niveles de prevención, alerta o emergencia.....</i> | <i>29</i> |
| <i>Tabla 7. Descripción general del Índice de Calidad del Aire.....</i> | <i>30</i> |
| <i>Tabla 8. Evaluación para la selección de los componentes.....</i> | <i>34</i> |
| <i>Tabla 9. Tipos de interfaz de comunicación.....</i> | <i>36</i> |
| <i>Tabla 10.....</i> | <i>44</i> |
| <i>Dispositivos que pertenecen en la etapa receptora.</i> | <i>44</i> |
| <i>Tabla 11.....</i> | <i>44</i> |
| <i>Dispositivos que pertenecen en la etapa transmisora.....</i> | <i>44</i> |
| <i>Tabla 12. Valores máximos obtenidos de cada parámetro monitoreado en Ecoparque Sabana.....</i> | <i>72</i> |
| <i>Tabla 13.....</i> | <i>73</i> |
| <i>Valores máximos obtenidos de cada parámetro monitoreado.....</i> | <i>73</i> |
| <i>Tabla 14. Topics MQTT. Fuente: Elaboración Propia.....</i> | <i>80</i> |
| <i>Tabla 15. Etiquetas de conexiones etapa receptora.....</i> | <i>80</i> |
| <i>Tabla 16. Etiquetas de conexión de componentes en la etapa Transmisora.</i> | <i>81</i> |
| <i>Tabla 17. Componentes electrónicos seleccionados.</i> | <i>82</i> |
| <i>Tabla 18. Datos de referencia para el proceso de calibración del sensor MQ135.</i> | <i>88</i> |
| <i>Tabla 19. Ejes de referencia para el proceso de calibración del sensor MQ7.</i> | <i>90</i> |

PLANTEAMIENTO DEL PROBLEMA

Uno de los puntos cruciales que interfieren directamente en el bienestar medioambiental y la salud pública, que muy probablemente influye en el rendimiento de las personas, y de todos aquellos que viven en zonas urbanas, y rurales, es el monitoreo en tiempo presente de la calidad del aire de forma constante o de manera regular (Aire Ciudadano, 2022). El monitoreo ambiental de los factores de la calidad del aire tanto de forma regular como a lo largo del tiempo es fundamental para abarcar todos los factores que influyen en la contaminación del aire, tales como, el incorrecto uso de desechos, construcciones y otras actividades que producen y tienen un alto impacto de manera negativa la calidad del aire (Aire Ciudadano, 2022). En ese sentido, los indicadores de la plataforma Aire Ciudadano que es un portal de monitoreo ambiental, donde se reporta en tiempo real, las mediciones de variables ambientales, tales como, temperatura, humedad y material particulado de 2.5 micrómetros. El portal de monitoreo del Aire ciudadano posee limitaciones debido a una importante falta de medición de otros gases, como lo son , el dióxido de carbono, monóxido de carbono y compuestos orgánicos volátiles. (Rodríguez Mosquera 2023; Holguín Pedraza y Andrade Forero 2021).

En otras palabras, actualmente información incompleta sobre la calidad del aire, afectando negativamente la capacidad de la población de tomar decisiones informadas para adoptar medidas preventivas, decisiones relacionadas con la salud pública o ambientales (Peñaranda Arcentales, 2024), ya que las mediciones que se realizan son solo de variables como material particulado de 2.5 micrómetros (PM2. 5), temperatura y humedad, No obstante, los compuestos volátiles como los compuestos orgánicos volátiles, el dióxido y monóxido de carbono, son críticos para la salud humana, particularmente, genera problemas respiratorios y cardiovasculares (Organización Mundial de la Salud, Citado en Roldan Martínez. 2022), tales

como, asma, bronquitis, enfisema o neumonía, teniendo síntomas como tos, flema, falta de aire, opresión en el pecho y respiración sibilante. El actual sistema del portal Aire Ciudadano no proporciona una visión completa de su estado de la calidad del aire que impide a los usuarios, instituciones y autoridades locales tomar medidas preventivas necesarias para contrarrestar los efectos adversos de dicha contaminación (Aire Ciudadano). Según Holguín Pedraza y Andrade forero en su proyecto realizado para ver la polución del aire por medio de una plataforma en la nube argumentan que la falta de datos sobre diferentes compuestos complica la capacidad de reaccionar a tiempo ante la contaminación, puesto que pone en riesgo la salud de la población al no contar con información suficiente para diseñar sobre la base de enfoques integrales el problema de la contaminación (Rodríguez Mosquera, 2023). Además, la falta de monitoreo de estos contaminantes reduce la conciencia ciudadana sobre la realidad del Estado de la calidad del aire obstaculizando la movilización de acciones y toma de decisiones que promuevan la mitigación monitoreo y control de la calidad del aire (Peñaranda Arcentales, 2024)

El presente trabajo de grado tiene como propósito, actualizar e implementar mejoras al sistema de monitoreo del portal Aire Ciudadano mediante la integración de diferentes sensores interconectados, como los sensores para medir COV, CO y CO₂. Mediante la implementación de protocolos de comunicación como LoRa y MQTT, posibilitando una comunicación a larga distancia (ThinkIOT, 2023; E. M. Q. Technologies, 2024). Con esta actualización, permitirá proporcionar datos de diferentes gases y en tiempo real sobre la calidad del aire, lo que permitirá a la ciudadanía y a las autoridades locales tomar decisiones informadas y proactivas para mejorar la salud pública y proteger el medio ambiente (Roldán Martínez, 2022; Código IoT, n.d).

ANTECEDENTES

En el ámbito de interés de este proyecto, se encuentran diversos trabajos y proyectos disponibles en sitios web oficiales y repositorios institucionales. Estos recursos abordan información clave sobre la calidad ambiental, sensores y herramientas digitales o TIC aplicadas en sistemas de monitoreo y control. A continuación, se presentan algunos antecedentes que permiten abordar información clave sobre la calidad ambiental, sensores y herramientas digitales o TIC aplicadas en sistemas de monitoreo y control, tomándolos como base para el desarrollo del presente trabajo de grado.

Uno de los antecedentes principales es el Proyecto Aire ciudadano, creado en 2017 por el ingeniero Daniel Bernal como una iniciativa de ciencia ciudadana, enfocándose en el monitoreo de la calidad del aire. Este proyecto, basado en el activismo ambiental, utiliza diferentes tecnologías accesibles para permitir a los ciudadanos monitorear variables ambientales en sus comunidades. A través de la implementación de sensores de bajo costo, los ciudadanos interesados en el proyecto pueden medir parámetros como el material particulado (PM2.5), la temperatura y la humedad, con la opción de compartir estos datos en la nube. Sin embargo, el proyecto actual se centra principalmente en las variables ya mencionadas, lo que deja fuera la medición de otros compuestos volátiles y gases nocivos, como el dióxido y monóxido de carbono, variables críticas para evaluar la calidad del aire. Ahora bien, Aire ciudadano recomienda cubrir la necesidad de actualizar los sistemas de monitoreo para incluir sensores adicionales que midan variables nocivas para la calidad del aire. Por esto, estas recomendaciones se alinean directamente con el objetivo del presente

El segundo proyecto de investigación titulado "WSN para el monitoreo continuo de variables ambientales", realizado por Carlos Daniel Cadena y Leidy Carolina Pulido, tiene como

objetivo implementar esta red WSN utilizando una plataforma IoT en la nube. El proyecto busca solucionar los desafíos relacionados con la recolección de información en tiempo real y la comunicación bidireccional entre la red de sensores y la plataforma. Se destaca que el sistema operativo Contiki y la plataforma de conexión de nodos Node-RED se consideró adecuado debido a su compatibilidad con diversas arquitecturas y su eficiencia al trabajar con recursos limitados de procesamiento en plataformas de hardware. Además, se menciona que la plataforma Ubidots ofrece valiosas herramientas para el control de dispositivos y la visualización de datos, aunque se identificaron limitantes de almacenamiento en su versión STEM para aplicaciones con múltiples nodos y variables.

En base a los hallazgos del proyecto, se proponen recomendaciones para mejorar la implementación de la WSN. Estas recomendaciones incluyen la incorporación de mecanismos de seguridad para proteger los datos de posibles capturas por parte de terceros, la optimización del código para reducir el consumo de corriente y prolongar la vida útil de los nodos, la utilización de baterías con mayor capacidad en aplicaciones industriales, y el diseño de una caja protectora para los nodos sensores expuestos a condiciones climáticas adversas (Cadena, 2021).

Para este se proyectó se tomarán en cuenta las lecciones aprendidas y las recomendaciones de dicho proyecto para desarrollar una solución que favorezca la eficiencia del sistema, utilizando la plataforma de conexión de flujos de datos Node-Red para garantizar seguridad y durabilidad de los nodos sensores en diferentes condiciones ambientales. Por otro lado, la optimización del código para reducir el consumo energético de los sensores, y la utilización de adaptadores o baterías de mayor capacidad para asegurar la durabilidad del sistema en diferentes entornos. Estas recomendaciones son aplicables al presente proyecto, puesto que

busca mejorar la autonomía y eficiencia de los sensores mediante el uso de tecnologías como LoRa y la implementación de un sistema de transmisión de datos optimizado.

En la tercera investigación titulada "Plataforma en la nube para la visualización y análisis de datos provenientes de dispositivos para el monitoreo de la polución del aire", trabajo realizado por Forero y Marcos Andrade (2021). Se centraron en el desarrollo de una plataforma basada en la nube que permite visualizar de manera interactiva los niveles de contaminación en la ciudad de Bogotá, utilizando dispositivos IoT para la recolección de datos sobre las variables de material particulado de 2.5 y 10 micrómetros. El estudio destacó la importancia de utilizar una interfaz interactiva como Grafana, puesto que facilita el análisis de los datos por parte de los usuarios, permitiendo informar sobre los niveles de contaminación de manera gráfica. Para este sistema también se identificó la necesidad de incluir más variables de medición para una evaluación integral de la calidad del aire.

En el presente proyecto se tomarán en cuenta diversas recomendaciones clave para la mejora del sistema de monitoreo del portal Aire Ciudadano. Entre ellas, destaca la optimización de la visualización de datos mediante la incorporación de nuevas funcionalidades en la interfaz gráfica de Grafana, lo cual es uno de los objetivos principales de la actualización. Asimismo, se priorizará el uso de herramientas de bajo costo para garantizar la accesibilidad del sistema y su capacidad de replicación en diferentes contextos. Esto permitirá desarrollar una solución escalable, económica y eficiente tanto en términos de consumo energético como en la transmisión de datos. Estas recomendaciones y antecedentes proporcionan una base sólida para el desarrollo del presente proyecto de grado, asegurando la implementación de un sistema integral, eficiente y accesible para el monitoreo de la calidad del aire.

JUSTIFICACIÓN

En la reserva natural Ecoparque Sabana, que se encuentra ubicada en el municipio de Tocancipá, exactamente en el Parque Jaime Duque, el tema principal que abarca dicha entidad es la restauración ecológica y el cuidado del medio ambiente, por tal motivo, el monitoreo y control de la calidad del aire ha tomado una mayor relevancia, puesto que tiene un impacto directo en comunidades y el ecosistema. Actualmente la reserva natural cuenta con una estación fija de monitoreo y control, la cual, solo se enfoca en obtener datos de material particulado de 25 micrómetros, temperatura y humedad. (Aire Ciudadano, 2022). Por tal motivo, existe la necesidad de actualizar y mejorar dicha estación de monitoreo, con la integración de dispositivos que permitan medir diferentes gases, tales como, el dióxido y monóxido de carbono y los compuestos volátiles orgánicos, ya que, según la Organización Mundial de la Salud (OMS), un alto nivel de los gases mencionados, influyen directamente al aumento de enfermedades respiratorias, cardiovasculares y otros problemas crónicos de salud que se pueden presentar, teniendo una presencia alta de contaminantes (Roldan Martínez, 2022; Holguín Pedraza & Andrade Forero, 2021), los problemas cardiovasculares pueden aumentar enfermedades y muerte prematura causada por dolor en el pecho, palpitaciones y la falta de aire producido por la baja oxigenación de glóbulos rojos, ritmos cardiacos anormales y alteración de la actividad cardiaca controlada por el sistema nervioso autónomo (Aire ciudadano, 2017). Por otro lado, si bien el uso de la estación portal Aire Ciudadano de la reserva natural Ecoparque La Sabana, como recurso único para el monitoreo participativo en tiempo real de PM2.5, temperatura y humedad contribuye significativamente al conocimiento completo de la calidad del aire, una sólida competencia limita su análisis e interpretación integral. Ocurrentemente, la falta de mediciones de CO₂, CO, y los compuestos orgánicos volátiles (COV) crea un faltante importante en la

información, dificultando su uso para informar decisiones e implementar medidas preventivas por parte de la comunidad (Aire Ciudadano, 2022; Rodríguez Mosquera, 2023).

La incorporación de sensores para medir estas partículas y compuestos ampliaría enormemente el conocimiento a disposición y daría poder a los ciudadanos con información crítica para proteger su bienestar en entornos tanto interiores como exteriores. Por otro lado, la presente investigación además pretende, no solo mejorar la calidad de monitoreo ambiental, sino promover conciencia colectiva sobre el valor de la calidad del aire y sus consecuencias en el bienestar público aportando la ciencia ciudadana, promoviendo la producción de conocimiento científico por medio de la participación e investigación colectiva. En consecuencia, se tiene la necesidad de actualizar y mejorar el sistema de Aire Ciudadano para medir la calidad del aire surge de la necesidad de dar información sobre los datos que permitan tomar decisiones informadas. Además de incorporar sensores para medir compuestos volátiles y gases nocivos ya mencionados para ofrecer una visión global de los riesgos que provoca la contaminación del aire. Esta mejora permitirá a los ciudadanos y a las instituciones tomar decisiones para adoptar medidas preventivas y evitar quedar expuestos en momentos críticos o en áreas afectadas (ThinkIOT, 2023).

El objetivo intrínseco de esta propuesta es fomentar un paradigma de monitoreo ambiental integral y proactivo, que coloca a la comunidad como un agente activo en la preservación del entorno y su salud. No sólo aterriza esta iniciativa en respuestas directas a las recomendaciones internacionales, sino que también representa una oportunidad real para el fortalecimiento de la resiliencia de la comunidad sobre riesgos ambientales (Rodríguez Mosquera 2023). Esta propuesta es relevante porque recoge la capacidad de generar un efecto positivo sobre la salud pública y la concienciación medioambiental al proporcionar una

plataforma de información fiable y accesible, en la que, dentro de la información, se promulgarán acciones que contribuyan a entornos más saludables y sostenibles basados en la responsabilidad.

Desde el punto de vista tecnológico, la propuesta encaja dentro de las cartas de la ciencia ciudadana y el uso de tecnologías accesibles. Mediante la implementación de sensores de bajo costo y protocolos de comunicación eficientes como LoRa y MQTT, el sistema de monitoreo se convertirá en una herramienta replicable en diversas áreas geográficas (E. M. Q. Technologies, 2024). Esto permitirá la creación de redes comunitarias de monitoreo que contribuyan a la recolección de datos a gran escala y fomenten una toma de decisiones informada (Código IoT, n.d.).

En otras palabras, el presente proyecto, no solo tiene influencia en el aspecto tecnológico del sistema de monitoreo del portal Aire ciudadano, sino que también, impulsa al fortalecimiento y preservación del medio ambiente, como también, al bienestar social y la salud pública. La importancia de contar con datos ambientales en tiempo real reside en su capacidad para analizar y procrear las respuestas de las personas y autoridades frente a los riesgos ambientales, impulsando a tener un entorno saludable y sostenible para las generaciones actuales y futuras.

OBJETIVO GENERAL

- Implementar las mejoras tecnológicas para la red de monitoreo y control de calidad del aire para visualizar el comportamiento de compuestos volátiles COV, los niveles de dióxido (CO) y monóxido de carbono (CO₂).

OBJETIVOS ESPECIFICOS

- Actualizar la estación fija de monitoreo del Portal Aire Ciudadano mediante la programación e implementación del protocolo de comunicación LoRA que integre y opere sensores de compuestos volátiles (COV), dióxido (CO) y monóxido de carbono (CO₂).
- Integrar protocolos de comunicación como LoRa y MQTT en el sistema existente del portal Aire Ciudadano para presentar y registrar la calidad del aire en tiempo real, empleando la plataforma Grafana.
- Crear una guía didáctica que simplifique la instalación, configuración y uso del sistema de monitoreo. Esta guía estará diseñada con el propósito de tener un instructivo base, de cómo implementar una estación de monitoreo de la calidad del aire.

MARCO CONCEPTUAL

El presente proyecto de grado se centra en usar diferentes tecnologías emergentes para la actualización del sistema de monitoreo de calidad del aire del portal Aire Ciudadano, realizando una actualización tanto el hardware como en el software del sistema actualmente implementado. La renovación del sistema de monitoreo por medio de la integración de componentes como sensores ambientales, protocolos de comunicación tales como el LoRA y el MQTT, y plataformas que permiten la visualización grafica de los datos, como Grafana. Además, se requiere una descripción profunda para su implementación y replica, según sea el caso, se decide

tomar como alternativa el Lenguaje Unificado de Modelado (UML) y las opciones que ofrece para el diseño y visualización del sistema.

Diagramas UML.

Los diagramas UML pueden representar la arquitectura del sistema, ya que son herramientas de modelado gráfico que permiten representar el sistema mediante su comportamiento y estructura de manera concisa, clara y visual. Para la presente investigación, estos diagramas son importantes, puesto que facilitan la planificación y comunicación entre los dispositivos que conforman el sistema de monitoreo. Los dos tipos de diagramas UML utilizados en este proyecto son los diagramas de clases y los diagramas de casos de usos [21].

Diagrama de Clases UML.

El diagrama de clases UML son figuras de ayuda utilizada dentro de la comunidad de ingeniería de software, usado, para representar la estructura estática de un sistema, que en este caso es un sistema de monitoreo de variables ambientales. Permite mostrar el sistema de manera detallada (conjuntos de clases, atributos, métodos, y relaciones entre ellos). Por ello, favorece visualizar cómo los componentes distintos se relacionan entre sí: cómo influyen y de qué dependen unos de otros permitiendo una mayor comprensión de la arquitectura del sistema y de cómo ésta deberá planificarse, con el objetivo de favorecer su implementación.

En el presente proyecto, el diagrama de clases se utiliza para realizar las relaciones entre los distintos componentes, así como para realizar la organización de los elementos que forman parte y completan el sistema. Por tal motivo, el diagrama permite identificar las interacciones y relaciones que existen entre los componentes, siendo esto importante para la planificación y la

implementación del software y el hardware del sistema, es decir, que permite la comprensión de la relación entre los sensores que permiten mediciones de variables ambientales, tales como temperatura, humedad, compuestos volátiles orgánicos (COV), dióxido de carbono (CO₂), monóxido de carbono (CO) y material particulado de 2.5 micrómetros y los módulos de comunicación (LoRa) con su respectiva placa microcontroladora sESP32, utilizadas para realizar la programación de las etapa receptora y transmisora de la estación de monitoreo.

El diagrama es útil porque permite describir gráficamente cómo los sensores reciben datos y los transmiten a través del sistema. Por ejemplo, sustrae a lo que hace el módulo LoRa en la estación transmisora que envía los datos recolecta por los sensores a la estación receptora que entonces los envía a una plataforma de almacenamiento y visualización. De manera adicional, sirve para modelar cada dispositivo como una clase, especificando los atributos (los datos de medición) y métodos (funciones para la transmisión y procesado de estos datos), mejorando así la comunicación entre dispositivos mediante protocolos tan simples como I²C, analógico o mediante el módulo LoRa con MQTT. Como conclusión, el diagrama de clases UML es fundamental en este proyecto porque permite estructurar de manera clara y lógica cómo los diferentes elementos del sistema se relacionan, lo que facilita su diseño, implementación y futura expansión [7].

Diagrama de Casos de Uso UML.

El diagrama de casos de uso UML describe los casos de uso que tienen lugar entre el sistema de monitoreo de la calidad del aire y el usuario (los actores). Este diagrama es útil para describir las particulares funcionalidades de los sistemas y escenarios de uso en los cuales los usuarios interfieren estos sistemas. En este proyecto se detalla, a través de diagrama de casos de uso, como los ciudadanos pueden consultar los datos de calidad del aire y contemplarlos en una

aplicación móvil (IoT MQTT Panel) o por medio de Grafana con conexión a la estación de monitoreo localmente. Asimismo, identifica a otro actor clave, el administrador del sistema, quien tiene privilegios para modificar parámetros críticos del sistema, como la frecuencia de medición o la incorporación de nuevos sensores [21].

Este diagrama puede ser de ayuda porque le brinda una visión general de cómo actúan diferentes tipos de usuarios en el sistema y así podrán ver cuáles son las funciones que necesitan ser conferidas. También ayuda a planificar el flujo de datos en el sistema, describiendo qué operaciones deben aplicarse a cada interacción. Esto es crucial para garantizar que tanto los ciudadanos como los administradores puedan obtener datos en tiempo real y modificar la configuración en caso necesario, como también, el nivel de acceso a la información y en sí, en el sistema. Los diagramas serán mostrados en el desarrollo de la fase 2, puesto que, es la encargada del diseño del sistema.

MARCO TEORICO

Actualmente existen muchas opciones de conectividad de dispositivos inalámbricos como WiFi, Bluetooth, LTE, SigFox, Zigbee y LoRa con características variadas que lo convierten en alternativa según aplicativo. Entre esta tecnología se resalta LoRa, por su tecnología de modulación favorece cubrir grandes superficies. Esta tecnología se presta para usar en dispositivos de bajo costo que usan batería, buscando cobertura de áreas con distancias más largas a lo que demandan datos en transmisión al mismo tiempo

Por otro lado, el protocolo MQTT, diseñado para facilitar la comunicación entre máquinas (M2M), se aplica en un modelo de publicación / suscripción a tópicos que lo hace altamente eficiente. Al utilizar mensajes pequeños, MQTT consume energía de forma eficiente,

una característica crucial para los dispositivos dependientes de la batería. Aunque este protocolo se ejecuta sobre TCP/IP, su arquitectura lo hace la oferta de baja latencia y consumo energético mínimo. Así mismo, incluye control de acceso mediante autenticación de usuario y contraseña o token, cifrado (SSL/TLS), lo que lo hace una opción fiable para aplicaciones que requieren alta seguridad, eficiencia energética y baja latencia en sistemas alimentados a batería.

Como fue mencionado, “Aire Ciudadano”, que es fundamentalmente el antecedente principal del proyecto de investigación y desarrollo que aquí se lleva a cabo, permite la medición de las partículas de material, la temperatura y la humedad mediante un conjunto de sensores. Los datos ambientales realizados son mostrados en tiempo real para un servidor de computadora que es un servidor y conecta a la interfaz Grafana. Así, hay que establecer una comunicación entre los dispositivos se han debido usar módulos LoRa de la capa física y protocolo MQTT para la transmisión de datos al servidor web. El tomar esta elección consiste en que el ancho de banda para enviar la información no es alto, ya que se trata de mensajes pequeños que notifiquen la información sobre el estado de la calidad del aire.

PROTOCOLO DE COMUNICACIÓN MQTT.

El protocolo MQTT (Message Queue Telemetry Transport) es una forma de comunicarse entre un dispositivo y un sistema usando redes de datos. Es un protocolo ligero por tanto es ideal para situaciones donde se requiere un bajo consumo de recursos que tiene directamente el sistema, por ejemplo, en dispositivos IoT. La baja latencia del MQTT permite que los mensajes se entreguen rápidamente para garantizar un tiempo mínimo entre el envío y la recepción de los datos. Se complementa con un modelo de publicación/suscripción, lo que implica que los

dispositivos no solo se comunican punto a punto, sino que también se pueden comunicar entre sí de manera bidireccional para lo que es esencial en redes de IoT [5, 8].

MQTT + BROKER:

El protocolo MQTT y su arquitectura, es un sistema de comunicación de red en estrella donde todos los dispositivos están unidos a un único punto central conocido como Broker. Por tanto, el Broker se comporta como un intermediario que une los mensajes emitidos por los dispositivos emisores y los distribuye a aquellos que se han suscrito a los temas de interés. El formato basado en eventos lo envía a todos los clientes conectados que se han suscrito a estos temas específicos [5, 8], teniendo un MQTT Broker principal, donde se realiza el proceso de publicación y suscripción, como se muestra en la figura 1.



Figura 1. Diagrama del protocolo MQTT + Broker. Fuente: EMQ Technologies

[A Comprehensive Comparison of Open Source MQTT Brokers 2023 | by EMQ Technologies | Medium](#)

INTERNET DE LAS COSAS

Sin la Internet de Cosas (IoT) nunca más los datos se adquirirán y serán controlados de la misma manera. Esta tecnología ayuda a la conexión de dispositivos y sistemas de recolección, transmisión, análisis datos en tiempo real. Conectividad inalámbrica en IoT facilita la información que viajar a través de los sensores al hardware o software y agregarle funcionalidad que los convierte con el fin de automatizar y optimizar los procesos. También, las interacciones entre dispositivos en el IoT no son simplemente ambas partes que intercambian información

entre iguales, sino una plataforma para supervisar y controlar desde distancia una gran cantidad de aplicaciones como control inteligente de medio ambiente hasta la administración inteligente de infraestructuras [20, 21, 4].

El objetivo principal del IoT es establecer una red interconectada que utilice internet como medio de comunicación para integrar dispositivos que transmiten datos en tiempo real, lo cual resulta fundamental para mejorar la eficiencia de sistemas de vigilancia, seguridad y monitoreo ambiental [20, 2].

GRAFANA

Grafana es una plataforma de visualización de datos avanzada y personalizable que permite analizar y monitorizar estas grandes cantidades de datos en tiempo real. Se utiliza casi exclusivamente en aplicaciones para IoT para extraer métricas de sistemas distribuidos y visualizarlas en gráficos y paneles interactivos. Es apto para muchos escenarios ya que propone integrarse con muchas fuentes de datos tales como InfluxDB, Prometheus, bases de datos SQL y por último sumar algunas de las habituales visualizaciones de datos [6,11].

Además, Grafana permite a los usuarios configurar gráficos en distintos formatos (barras, líneas, mapas de calor), facilitando la interpretación de patrones y tendencias en los datos. Asimismo, cuenta con la capacidad de generar notificaciones y alertas en tiempo real, lo que es esencial en sistemas de monitoreo continuo para la detección de eventos o fallas importantes [19, 3].

NODE.JS

Node.js es una tecnología que contiene entorno de ejecución en tiempo presente, incluyendo bibliotecas y programas realizados mediante el lenguaje de programación, JavaScript,

ejecutando subprogramas de manera independiente, utilizando un modelo entrada y salida sin bloqueo que es controlado por eventos, es decir, que se pueden realizar diferentes procesos, como operaciones, lecturas y escritura, como también realizar solicitudes por medio de HTTP, permitiendo así, el funcionamiento correcto de Node-RED, ya que proporciona librerías y motor de desarrollo necesario para el inicio y conexión de los nodos de datos.

NODE-RED

Node-RED es una herramienta de programación visual diseñada para crear flujos de trabajo que integran dispositivos, servicios y aplicaciones de manera intuitiva. Su interfaz de arrastrar y soltar permite a los usuarios conectar nodos que representan tareas o funciones específicas, facilitando la creación de soluciones IoT complejas sin necesidad de escribir grandes cantidades de código. Al ser una plataforma de código abierto, Node-RED cuenta con una extensa biblioteca de nodos predefinidos que soportan diversos protocolos de comunicación, como MQTT, HTTP y WebSockets, lo que lo convierte en una herramienta ideal para integrar y gestionar dispositivos IoT de manera eficiente [16, 5].

Node-RED simplifica la creación de aplicaciones IoT al permitir que los desarrolladores conecten y gestionen dispositivos y servicios en un solo entorno visual, facilitando la automatización de procesos y la integración de flujos de datos entre dispositivos distribuidos.

INFLUXDB

InfluxDB es una plataforma de base de datos de series temporales de código abierto creada por InfluxData. Esta base de datos se ha optimizado para quedarse con grandes volúmenes de datos de crecimiento crónico como eventos y mediciones marcadas con tiempo.

InfluxDB se adapta bastante bien al tipo de datos que se producen a intervalos regulares o irregulares (utilizados para la monitorización y el control de IoT [5, 6]).

Una de las mejores cosas de InfluxDB es que puedes definir políticas para retener datos, lo que se encarga de los datos antiguos y mantiene la interacción humana con la gestión de los datos al mínimo. Además, es una arquitectura fácil de soportar altas velocidades de escritura y consulta y es una herramienta adecuada para aplicaciones que necesitan realizar transmisión de datos en tiempo real.

INDICE DE MEDICIONES DE CALIDAD DEL AIRE

Para el análisis e interpretación de los datos recolectados en las mediciones realizadas sobre la calidad del aire, se tuvieron en cuenta diferentes áreas de estudios, haciendo uso de los estándares y escalas establecidos por la Oficina de Planificación y Estándares de Calidad del Aire – EPA, el ICA (Índice de la Calidad del Aire), y la Resolución 2254 de 2017 del Ministerio de Medio Ambiente de Colombia, con el propósito de establecer y tener en cuenta las escalas de mediciones de manera internacional, nacional y local. Por otro lado, las mediciones de PM_{2.5} y CO serán comparados con los rangos establecidos que se muestran en el Índice de calidad del aire (AQI) de la Agencia de Protección Ambiental de Estados Unidos (EPA) , como también, los criterios nacionales establecidos por medio de la resolución 2254. Esto permitió evaluar la calidad del aire en una Reserva natural como Ecoparque Sabana, como en el área urbana, medición realizada en la Universidad Pedagógica Nacional de Colombia, con el objetivo de establecer un análisis apoyado en normas nacionales e internacionales para respaldar y argumentar la interpretación y análisis de los niveles de contaminantes, teniendo así, una

información detallada sobre sus posibles implicaciones sobre un entorno ambiental y la salud (Ministerio de Medio Ambiente, 2017).

Las mediciones de material particulado de 2.5 micrómetros, dióxido y monóxido de carbono, se alinea a los estándares federales recomendados por la EPA, siguiendo la escala permisible y establecidos en la Resolución 2254, donde también el uso de la escala ICA permite interpretar los resultado de manera sencilla y accesible, con el propósito de alertar y facilitar información sobre los efectos de la contaminación en la salud pública, como también, la comparación entre áreas urbanas y reservas naturales, impulsando la ciencia ciudadana, con la toma decisiones informadas.

Ahora bien, como se mencionó anteriormente, la EPA utiliza el índice de calidad del Aire (AQÍ) para analizar, interpretar e informar los niveles de contaminación de manera accesible (EPA, 2023) para los contaminantes de partículas PM_{2.5} y monóxido de carbono (CO), como se muestra en la tabla 1. teniendo la categoría, el rango y las concentraciones de PM_{2.5} ($\mu\text{g}/\text{m}^3$) según el AQI (Escala de medición de la EPA), por otro lado, en la tabla 2, se muestra las categorías, rango y concentraciones de partículas por millón de concentración de CO.

Tabla 1.
Escala de medición de la EPA (AQI) para concentración de PM_{2.5}

| Categoría de AQI | Rango de AQI | Concentración de PM_{2.5} ($\mu\text{g}/\text{m}^3$) |
|---------------------------------|---------------------|------------------------------------------------------------------------------------|
| Buena | 0-50 | 0.0-12.0 |
| Moderada | 51-100 | 12.1-35.4 |
| Insalubre para grupos sensibles | 101-150 | 35.5-55.4 |
| Insalubre | 151-200 | 55.5-150.4 |
| Muy insalubre | 201-300 | 150.5-250.4 |
| Peligrosa | 301-500 | 250.5 o más |

Nota: Esta tabla muestra la escala de medición de la EPA teniendo en cuenta las categorías, rangos y concentración de material particulado. Fuente: U.S. Environmental Protection Agency, 2023.

Tabla 2.

Escala de medición de la EPA (AQI) para concentración de CO.

| Categoría de AQI | Rango de AQI | Concentración de CO |
|---------------------------------|---------------------|----------------------------|
| Buena | 0-50 | 0.0-4.4 |
| Moderada | 51-100 | 4.5-9.4 |
| Insalubre para grupos sensibles | 101-150 | 9.5-12.4 |
| Insalubre | 151-200 | 12.5-15.4 |
| Muy insalubre | 201-300 | 15.5-30.4 |
| Peligrosa | 301-500 | 30.5 o más |

Nota: Esta tabla muestra la escala de medición de la EPA teniendo en cuenta las categorías, rangos y concentración de material. Fuente: U.S. Environmental Protection Agency, 2023.

Para la escala de medición del ICA en Colombia, se tienen categorías similares a las que proporciona el AQI de la EPA, sin embargo, es importante tener en cuenta, que se tienen diferentes ajustes, con el propósito de adaptarse a las condiciones que proporciona Colombia, teniendo variaciones según el contaminante. El ICA proporciona las siguientes categorías, rangos y concentraciones de material particulado 2.5, mostradas en la tabla 3.

Tabla 3.

Escala de medición del ICA en Colombia para concentración de PM2.5

| Categoría de AQI | Rango de AQI | Concentración de PM2.5 ($\mu\text{g}/\text{m}^3$) |
|---------------------------------|---------------------|-------------------------------------------------------------------------|
| Buena | 0-50 | 0.0-12.0 |
| Moderada | 51-100 | 12.1-37.0 |
| Insalubre para grupos sensibles | 101-150 | 37.1-55.4 |
| Insalubre | 151-200 | 55.5-150.4 |
| Muy insalubre | 201-300 | 150.5-250.4 |
| Peligrosa | 301-500 | 250.5 o más |

Nota: Esta tabla muestra la escala de medición para concentración de PM2.5 según el ICA en Colombia. Fuente: Ministerio de Ambiente y Desarrollo Sostenible, 2022.

Por otro lado, el ICA también realiza una clasificación del CO en categorías similares, pero es importante tener en cuenta, que pueden variar dependiendo los estándares aplicados de manera local, regional o ciudad. La escala de medición que proporciona la Resolución 2254 de 2017 del Ministerio de Medio Ambiente de Colombia, establece diferentes límites permisibles para ciertos contaminantes específicos, asociando un color dependiendo su concentración, facilitando su comprensión y análisis, como se muestra en la tabla 4, donde se describen los niveles máximos permisibles de contaminantes criterio en el aire, como también, los niveles máximos permisibles de contaminantes en el aire para el 2030 como se muestra en la tabla 5, siguiendo la línea de los niveles máximos permisibles de contaminantes, el ICA también proporciona las concentraciones ($\mu\text{g}/\text{m}^3$) para los niveles de prevención, alerta o emergencia como se muestra en la tabla 6, tomando como referencia la resolución 2254 de 2017, que tiene como objetivo principal establecer el ámbito de aplicación y niveles máximos permisibles de contaminantes en el aire.

Tabla 4.

Niveles máximos permisibles de contaminantes criterio en el aire

| Contaminante | Nivel máximo Permissible ($\mu\text{g}/\text{m}^3$) | Tiempo de Exposición |
|--------------|-------------------------------------------------------|----------------------|
| PM10 | 30 | Anual |
| PM25 | 15 | Anual |
| SO2 | 20 | 24 horas |
| NO2 | 40 | Anual |

Nota: Esta tabla muestra los niveles máximos permisibles de material particulado criterio en el aire.

Fuente: Ministerio de Ambiente y Desarrollo Sostenible, 2017.

Tabla 5.

Niveles máximos de contaminantes en el aire permisibles para el año 2030

| Contaminante | Nivel máximo Permisible ($\mu\text{g}/\text{m}^3$) | Tiempo de Exposición |
|-----------------|------------------------------------------------------|----------------------|
| PM10 | 50 | Anual |
| | 100 | 24 horas |
| PM 2.5 | 25 | Anual |
| | 50 | 24 horas |
| SO ₂ | 50 | 24 horas |
| | 100 | 1 hora |
| NO ₂ | 60 | Anual |
| | 200 | 1 hora |
| O ₃ | 100 | 8 horas |
| CO | 5.000 | 8 horas |
| | 35.000 | 1 hora |

Nota: Esta tabla muestra los niveles máximos de contaminantes en el aire permisibles para el año 2030 en Colombia. Fuente: Ministerio de Ambiente y Desarrollo Sostenible, 2017.

Tabla 6.

Concentraciones ($\mu\text{g}/\text{m}^3$) para los niveles de prevención, alerta o emergencia

| Contaminante | Tiempo de Exposición | Prevención | Alerta | Emergencia* |
|----------------------|----------------------|-------------|-------------|--------------|
| PM10 | 24 horas | 155-254 | 255-354 | ≥ 355 |
| PM _{2.5} ** | 24 horas | 38-55 | 56-150 | ≥ 151 |
| O ₃ | 8 horas | 139-167 | 168-207 | ≥ 208 |
| SO ₂ | 1 hora | 198-486 | 487-797 | ≥ 798 |
| NO ₂ | 1 hora | 190-677 | 678-1221 | ≥ 1222 |
| CO | 8 horas | 10820-14254 | 14255-17688 | ≥ 17689 |

Nota: Esta tabla muestra las concentraciones para los niveles de prevención o emergencia y el tiempo de exposición del contaminante. Fuente: Ministerio de Ambiente y Desarrollo Sostenible, 2017.

Por otro lado, en la tabla 7, se puede observar una descripción general del índice de calidad del aire de manera local, teniendo en cuenta que se tiene una asociación de color y concentración de partículas de PM_{2.5}, siendo esto, importante para obtener una mayor comprensión e interpretaciones de los datos obtenidos.

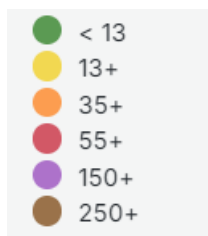
Tabla 7.
Descripción general del Índice de Calidad del Aire

| Rango | Color | Estado de la calidad del aire | Efectos |
|-----------|----------|---------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0-50 | Verde | Buena | La contaminación atmosférica supone un riesgo bajo para la salud. |
| 51-100 | Amarillo | Aceptable | Posibles síntomas respiratorios en grupos poblacionales sensibles. |
| 101-150 | Naranja | Dañina a la salud de grupos sensibles | Los grupos poblacionales sensibles pueden presentar efectos sobre la salud. 1) Ozono Troposférico: Las personas con enfermedades pulmonares, niños, adultos mayores y las que constantemente realizan actividad física al aire libre, deben reducir su exposición a los contaminantes del aire. 2) Material Particulado: Las personas con enfermedad cardíaca o pulmonar, los adultos mayores y los niños se consideran sensibles y por lo tanto en mayor riesgo. |
| 151-200 | Rojo | Dañina para la salud | Todos los individuos pueden comenzar a experimentar efectos sobre la salud. Los grupos sensibles pueden experimentar efectos más graves para la salud. |
| 201-300 | Púrpura | Muy Dañina para la salud | Estado de alerta que significa que todos pueden experimentar efectos más graves para la salud. |
| 301 – 500 | Marrón | Peligroso | Advertencia sanitaria. Toda la población puede presentar efectos adversos graves en la salud humana y están propensos a verse afectados por graves efectos sobre la salud. |

Nota: Esta tabla muestra una descripción general del ICA según el rango y su color asociado dependiendo la calidad del aire. Fuente: Ministerio de Ambiente y Desarrollo Sostenible, 2022.

Esto permite tener una mayor comprensión sobre la cantidad de material particulado que se encuentre en el ambiente, teniendo las siguientes relaciones entre concentración de material particulado y color. El color verde indica buena calidad del aire, siendo este inferior a 12 ppm, por otro lado, el color amarillo proporciona información sobre una calidad del aire aceptable. El color naranja y rojo indican una mayor concentración de partículas, teniendo un estado de calidad del aire dañina, y por último, el color púrpura y marrón, indican que la cantidad de PM2.5 es muy dañina y peligrosa para la salud, ingresando a un estado de alerta y advertencia sanitaria, as relaciones de color y concentración de PM2.5 se muestran en la figura 2, fortaleciendo la capacidad de interpretación y análisis de las mediciones realizadas.

Figura 2. Colores asociados al índice de calidad del aire. Fuente: Portal Aire Ciudadano.



METODOLOGÍA

En esta etapa del diseño del sistema, se ha optado por emplear la metodología en cascada para la implementación y desarrollo de la estación de monitoreo. La metodología en cascada es un enfoque estructurado y secuencial, requiere que cada fase debe completarse antes de pasar a la siguiente. La metodología en cascada se seleccionó para este proyecto, considerando que su estructura secuencial y organizada permite un desarrollo claro y sistemático para cada fase del proyecto, ya que es adecuada para trabajos donde se integra hardware y software, porque proporciona un desarrollo y marco claro en cada fase del proyecto, favoreciendo seguir el progreso.

Por tal razón las fases que complementan el presente proyecto de investigación y desarrollo se muestran en la figura 3, teniendo en cuenta la estructura de la metodología Waterfall, como también las fases establecidas con el propósito de que se siga una secuencia definida, sin la posibilidad de avanzar hasta que se haya completado la fase anterior.

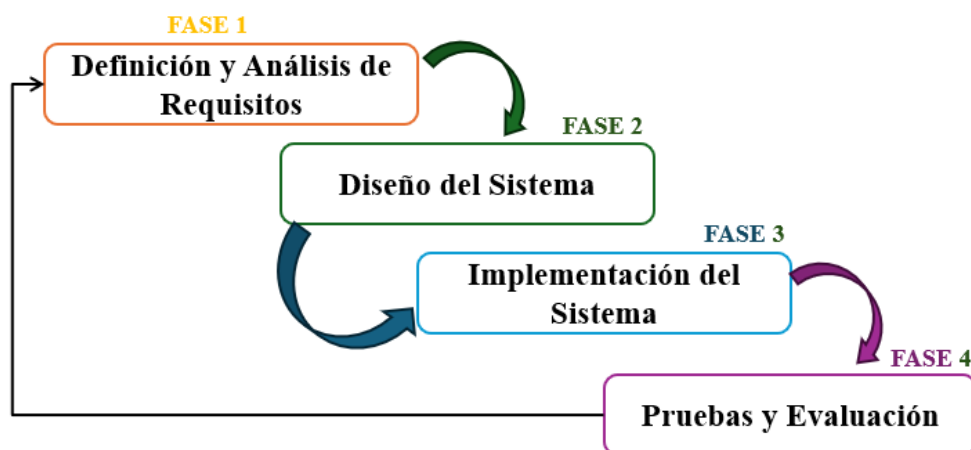


Figura 3. Estructura de la metodología escogida Waterfall.

Fuente: Elaboración Propia.

Fase 1: Definición de análisis de requerimientos

Durante esta etapa, se procede a la identificación y análisis de los componentes, dispositivos y sensores necesarios para el proyecto. Esto incluye la evaluación de los requisitos del sistema y la selección de los elementos esenciales para la estación de monitoreo.

Fase2: Diseño del sistema

Basándose en los requisitos definidos en la Fase 1, se crea el diseño del sistema. Esto implica la selección de sensores adicionales considerando criterios de costo, beneficio y funcionalidad. Se desarrollan estrategias de integración y se elaboran los diagramas necesarios para organizar el sistema y proporcionar una comprensión clara de cómo se implementarán e integrarán el hardware y el software de la estación de monitoreo.

Fase 3: Implementación del sistema

En esta fase se lleva a cabo la implementación del sistema de acuerdo con lo que se elaboró en la Fase 2, incluyendo el desarrollo y codificación del sistema, la integración de componentes y la configuración de sensores y módulos LoRa; garantizando de esta forma que todos los elementos del sistema funcionen de forma concertada y eficiente.

Fase 4: Pruebas y evaluación

Finalmente, se realizan pruebas y una evaluación detallada del sistema por cada sensor y modulo configurado para validar su funcionalidad. Se verifica que el sistema cumple con los requisitos establecidos, se realizan ajustes y optimizaciones necesarias, y se prueba el funcionamiento de la estación de monitoreo en diversas condiciones para asegurar su fiabilidad y precisión.

RESULTADOS

DESARROLLO DE LA FASE 1. ANALISIS Y REQUERIMIENTOS.

El significado de esta primera fase es encontrar y documentar todas las necesidades del sistema de monitoreo de calidad del aire. En esta fase se definen los requerimientos que necesitan las variables que deben medirse. Se recopilan: temperatura, humedad, compuestos volátiles (COV), dióxido de carbono (CO₂), monóxido de carbono (CO), y material particulado (PM_{2.5}). Además, se determinan los requisitos técnicos y funcionales a los que deberá guiar la implementación del sistema.



Se realizó una revisión exhaustiva de los sensores disponibles en el mercado, seleccionando aquellos que se ajustan mejor a los objetivos del proyecto, considerando factores como precisión, costo y compatibilidad con la plataforma ESP32. Asimismo, se decidió utilizar los protocolos de comunicación LoRa y MQTT para la transmisión de datos, y las plataformas InfluxDB y Grafana para el almacenamiento y visualización de la información en tiempo real.





Con el objetivo de mejorar el sistema actual de la plataforma Aire Ciudadano, se instalaron sensores programados para medir diversos parámetros de calidad del aire. La integración del protocolo LoRa permitió la transmisión de datos desde la estación emisora (conectada a la red WiFi) hasta la estación receptora, que actuó como repetidor y se encargó de enviar los datos a una interfaz web accesible. Los activos y materiales de la actual estación de monitorización permanente del Portal Ciudadano del Aire, así como los conocimientos adquiridos con su funcionamiento, se utilizarán para facilitar las actualizaciones de hardware y software. Además, se prestó especial atención a la calibración de los sensores y las tarjetas de control para garantizar su correcta instalación y funcionamiento en el sistema.

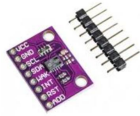
A continuación, en la tabla 1, se muestra una comparación de los componentes seleccionados para el sistema de monitoreo, justificando su elección en función de su relación costo-beneficio y compatibilidad técnica. Se evaluaron diversas opciones para cada sensor, evaluaciones mostradas en el [anexo 4](#), donde se eligieron aquellos que mejor se ajustan al proyecto, como insumo, se tuvo en cuenta la precisión, funcionalidad y costo. Además, en el [anexo 4](#) se adjunta una tabla donde se tienen los componentes seleccionados, sus comparaciones, costos y recomendaciones de uso, con el propósito de tener más información de los dispositivos que complementan el sistema de monitoreo.

Tabla 8.

Evaluación para la selección de los componentes

| Componente | Comparación | Razón de Selección |
|------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Tarjeta Controladora ESP32</p>  | <p>La Tarjeta Controladora ESP8266 fue considerada como una opción, pero se eligió el ESP32 debido a su menor costo, funcionalidad garantizada y compatibilidad con el código existente.</p> | <p>Se seleccionó debido a su integración con WiFi y compatibilidad con IoT.</p> |
| <p>Módulos LoRa SX1278</p>  | <p>Se evaluaron otros módulos LoRa, como los SX1276 o SX1272, pero se seleccionó el SX1278 debido a su mejor relación costo-beneficio y rendimiento adecuado para la comunicación LoRa.</p> | <p>El módulo LoRa SX1278 fue seleccionado por su rendimiento adecuado y su relación costo-beneficio favorable. Es capaz de proporcionar comunicación LoRa de manera efectiva y confiable. Trabajado en un rango de frecuencia de 450 MHz, con una distancia de 5 Km a distancia abierta. Su elección se basa en su capacidad para cumplir con los requisitos del proyecto a un costo razonable.</p> |

| | | |
|----------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Sensor SHT31D</p>  | <p>Aunque existen otros sensores de temperatura y humedad en el mercado, como el DHT22 o el BME280, se optó por el SHT31D debido a su precisión, confiabilidad y facilidad de integración con el código existente.</p> | <p>Se seleccionó porque ya está presente en el código y esquema de conexión del antecedente.</p> |
| <p>Sensor Plantower PMS7003 PM2.5</p>  | <p>Aunque hay otros sensores de partículas populares, como el SDS011 o el Nova PM SDS011, se eligió el PMS7003 por su precisión, compatibilidad con el proyecto existente y su costo beneficio en comparación con otros sensores.</p> | <p>Se seleccionó porque está presente en el antecedente y es compatible con la medición de partículas.</p> |
| <p>Sensor MQ7 (CO)</p>  | <p>Aunque otros sensores de CO, como el MQ-135 o el Fígaro TGS2620, podrían cumplir con la medición de CO, se seleccionó el MQ7 debido a su bajo costo y capacidad para proporcionar mediciones precisas de CO, lo que lo hace adecuado para el proyecto.</p> | <p>Se seleccionó por su precio y capacidad para medir CO, cumpliendo con los requisitos del proyecto.</p> |
| <p>Sensor MQ135 (CO₂).</p>  | <p>El sensor MQ135 se seleccionó debido a su capacidad para medir de manera precisa y confiable una amplia gama de gases en el aire. Es especialmente adecuado para monitoreo de la calidad del aire y detección de gases tóxicos. Su salida analógica permite una fácil integración con la tarjeta controladora ESP8266, lo que garantiza una interoperabilidad efectiva en el sistema. El uso del protocolo MQTT facilita la transmisión eficiente de datos a través de Internet para su posterior análisis y visualización.</p> | <p>El sensor MQ135 fue elegido debido a su capacidad para detectar una amplia gama de gases, incluyendo amoníaco, benceno, óxidos de nitrógeno, dióxido de azufre y compuestos orgánicos volátiles (COV). Su salida analógica proporcional permite medir la concentración de los gases detectados. Es adecuado para monitoreo de la calidad del aire y detección de gases tóxicos. La integración con la tarjeta controladora ESP8266 y el uso del protocolo MQTT garantizan una fácil integración y transmisión de datos eficiente para su análisis y visualización. En resumen, el MQ135 ofrece funcionalidad versátil y es una elección sólida para proyectos de monitoreo de la calidad del aire.</p> |

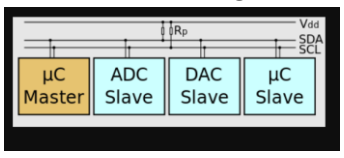
| | | |
|---------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------|
| <p>Sensor CCS811 (COV)</p>  | <p>El ENS160 y el SGX MICS-5524 son otros sensores de COV que podrían ser considerados, pero se eligió el CCS811 debido a su precio accesible y su capacidad para medir una amplia gama de COV, lo que lo hace adecuado para el proyecto.</p> | <p>Se eligió por su precio y características para medir COV, necesarias para ampliar la medición del proyecto.</p> |
|---------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------|

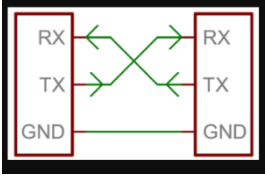
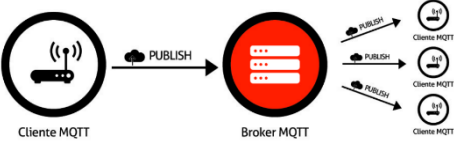
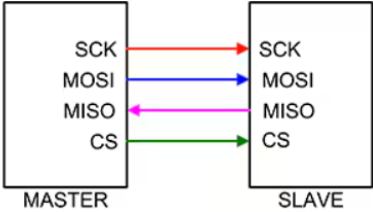
Nota: Esta tabla muestra la razón de selección de los componentes utilizados en el sistema. Fuente: Elaboración Propia.

Además, se detallan los tipos de interfaz de comunicación utilizados en el sistema, tales como I2C y UART, describiendo brevemente su funcionalidad y utilidad en la transmisión de datos entre los componentes. Estos protocolos permiten una conexión eficiente entre los sensores y la tarjeta controladora, asegurando la comunicación necesaria para el correcto funcionamiento del sistema, en la tabla 2 se describen las especificaciones de los tipos de interfaces que se usaron en el proceso de desarrollo y actualización del software.

Tabla 9.

Tipos de interfaz de comunicación.

| Tipo de Interfaz de Comunicación | Especificaciones |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p style="text-align: center;"><i>I2C</i></p>  <p style="text-align: center;"><i>Diagrama de conexión I2C</i> Imagen tomada de: 425px-I2C.svg .png (425×150) (wp.com)</p> | <p>Inter-Integrated Circuit. El tipo de comunicación conocido como I2C, es de manera síncrona, con una velocidad de datos hasta 3.4Mbbs, aunque en ocasiones puede alcanzar los 1Mbbs. En el sistema I2C, se utiliza una señal de rejoy compartida entre múltiples maestros y esclavos, dando la opción de que todos los maestros se comuniquen con todos los esclavos, teniendo una señal reloj común entre múltiples masters y esclavos, lo que permite la conexión de hasta 27 esclavos en un solo bus de comunicación.</p> |

| | |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p style="text-align: center;"><i>UART</i></p>  <p style="text-align: center;"><i>Diagrama de interfaz de conexión UART.</i> tomada de: 50e1ce8bce395fb62b000000.png (394×239) (wp.com)</p> | <p>Universal Asynchronous Receiver-Transmitter. La comunicación UART es un estándar de comunicación utilizado para la transferencia entre dispositivos de manera digital, enviándose de manera asincrónica, utilizando una velocidad de baudios predefinida para sincronizar la comunicación. Donde se tienen en cuenta dos señales principales:</p> <p>Transmisión (Tx): Señal que se encarga de llevar los datos desde un dispositivo de transmisión, en este caso, es la tarjeta controladora, hacia el receptor, en este caso es un módulo de comunicación, puesto que también puede ser otro microcontrolador.</p> <p>Recepción (Rx): Esta señal tiene la función de recibir los datos del dispositivo receptor, que proviene desde el trasmisor.</p> |
| <p style="text-align: center;"><i>MQTT</i></p>  <p style="text-align: center;"><i>Diagrama del protocolo MQTT.</i> tomada de: https://www.smartnet.com.co/mqtt-y-la-escalabilidad-del-broker/</p> | <p>MQTT (Message Queuing Telemetry Transport). es un protocolo diseñado para permitir la comunicación en tiempo real en redes con ancho de banda limitado. Una de las características principales del MQTT es su Publicación/Suscripción. Es decir que los dispositivos (clientes) pueden realizar la publicación de mensajes a un “topic” (tema), como también realizar suscripción a los temas para recibir mensajes. La calidad del servicio (QoS) de MQTT incluye tres niveles: QoS 0 (Best-effort), que entrega el mensaje al menos una vez; QoS 1 (garantizado), que también entrega el mensaje al menos una vez; y finalmente, QoS 2, que garantiza la entrega exactamente una vez, a pesar de ser más costoso en términos de recursos.</p> |
| <p style="text-align: center;"><i>SPI</i></p>  <p style="text-align: center;"><i>Conexión dúplex SPI.</i> Tomada de: https://www.digikey.com/es/articles/why-how-to-use-serial-peripheral-interface-simplify-connections-between-multiple-devices</p> | <p>SPI (Serial Peripheral Interface), protocolo de comunicación síncrona, es decir que utiliza un reloj compartido para sincronizar la transmisión de datos entre dispositivos. SPI es utilizado para la transferencia de datos entre un microcontrolador (maestro) y uno o más periféricos que serían los microcontroladores esclavos. La transferencia de datos que realiza el maestro al esclavo, la realiza a través de MOSI, es decir múltiples salidas, con solo una entrada, donde una vez completada la transferencia, el maestro desactiva el esclavo.</p> |

Nota: Esta tabla muestra los tipos de interfaz utilizados en el sistema teniendo en cuenta sus especificaciones. Fuente: Elaboración Propia.

DESARROLLO DE LA FASE 2. DISEÑO DEL SISTEMA

El objetivo principal de esta fase es crear e implementar un diseño detallado de la estación de monitoreo de calidad del aire, basándonos en los requisitos establecidos en la Fase 1. En esta etapa, se desarrollan estrategias de implementación e integración de los componentes, y se elaboran los diagramas UML para proporcionar una comprensión clara de la arquitectura del sistema, tanto en términos de hardware como de software.

DISEÑO DEL SISTEMA

Una vez definidos los requisitos, se procede a la fase de diseño, donde se estructura la arquitectura del sistema y se elaboran los diagramas UML, como el diagrama de clases y el diagrama de casos de uso. En esta fase, se define cómo interactuarán los sensores, los módulos de comunicación (como LoRa) y el ESP32 con las plataformas de almacenamiento y visualización, como InfluxDB y Grafana. El diseño del sistema incluye el hardware como el software. En la parte de hardware, se establecen las conexiones físicas entre los sensores y la placa de control (ESP32), utilizando protocolos como I2C o UART para la comunicación entre sensores y LoRa para la transmisión de datos a largas distancias. En la parte de software, se desarrolla la estructura del código que capturará, procesará y transmitirá los datos de los sensores hacia la base de datos InfluxDB, y luego hacia la interfaz de visualización Grafana.

Diagrama de Casos de Uso UML

El diagrama de casos de uso UML, describe las interacciones entre los usuarios y el sistema de monitoreo. Este diagrama es clave para comprender los requisitos del sistema desde la perspectiva del usuario, visualizando las dependencias y asociaciones como se describe gráficamente en la figura 4. En el diagrama se identifican dos actores principales:

- **Usuario:** Interactúa con el sistema para acceder a los datos sobre la calidad del aire.
- **Administrador:** Tiene privilegios adicionales para configurar parámetros, realizar mantenimiento y gestionar el sistema.

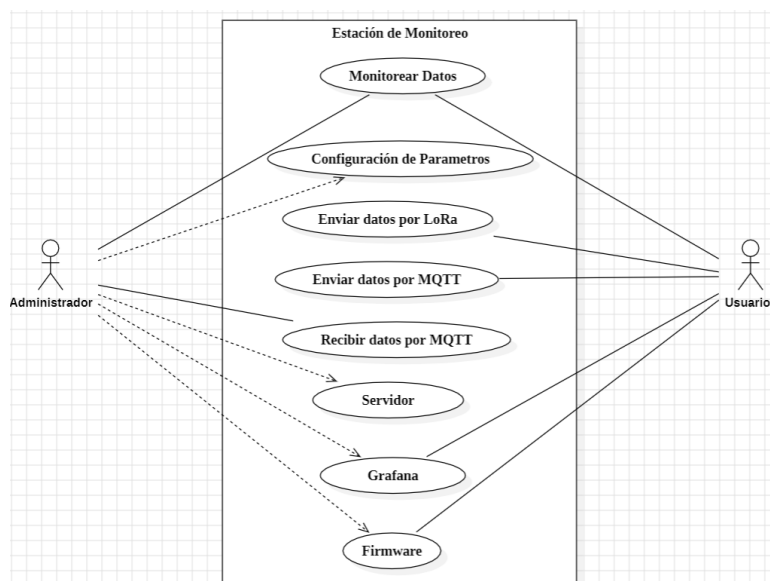


Figura 4. Diagrama de casos de uso UML.

Fuente: elaboración propia, realizado en: <https://staruml.io/>

El diagrama de la figura 3, es importante para comprender los requisitos del sistema desde una perspectiva del usuario, visualizando dependencias y asociaciones mediante líneas rectas y flechas. En la sección de actores, se identifica al usuario, quien interactúa con el sistema para acceder a los datos y realizar acciones específicas. Además, está el administrador, que posee privilegios adicionales para la administración, configuración y mantenimiento eficiente del sistema. Ahora bien, los casos de usos son:

Casos de Uso:

1. **Monitoreo de variables ambientales:** El usuario puede acceder a la plataforma y visualizar las mediciones en tiempo real de variables como temperatura, humedad,

PM2.5, CO₂, CO, y COV. El sistema recopila estos datos y la muestra de forma gráfica a través de la interfaz de usuario.

2. **Configuración de parámetros de monitoreo:** El administrador tiene la capacidad de modificar la frecuencia de muestreo, ajustar los umbrales de alerta, y seleccionar qué variables se deben monitorear. Esto permite adaptar el sistema a diferentes entornos y necesidades de monitoreo.
3. **Transmisión de datos a través de LoRa:** El sistema puede transmitir los datos recolectados mediante la red LoRa a una estación receptora. Tanto el usuario como el administrador pueden iniciar esta acción desde la interfaz de usuario, asegurando la transmisión de los datos a largas distancias.
4. **Almacenamiento de datos en el servidor:** Una vez recolectados, los datos se almacenan automáticamente en un servidor remoto. Esta acción no requiere intervención directa del usuario o administrador, asegurando que los datos se mantengan disponibles para análisis futuros.
5. **Visualización de datos en Grafana:** Tanto los usuarios como los administradores pueden acceder a los datos almacenados a través de la interfaz Grafana, que proporciona gráficos interactivos y detallados para el análisis y monitoreo de las condiciones ambientales.

Diagrama de Clases UML

El diagrama de clases UML, tiene como objetivo describir visualmente cómo la estación de monitoreo recopila datos de los diferentes sensores, utilizando una combinación de protocolos de comunicación, y los transmite a un servidor mediante MQTT para su almacenamiento y visualización en Grafana. En la figura 5 se muestra el diagrama de clases realizado con el objetivo de facilitar la comprensión de cómo se integran los componentes de hardware y software:

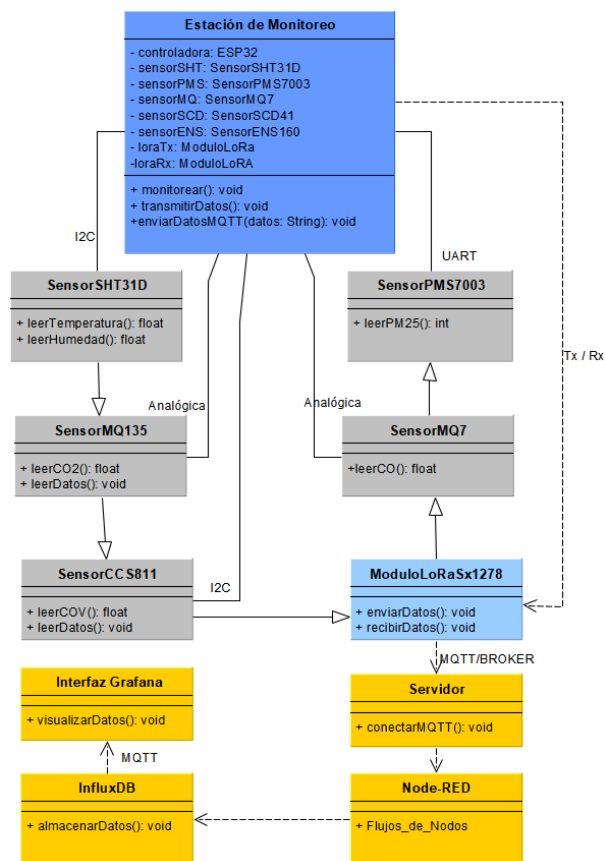


Figura 5. Diagrama de lenguaje de modelado Unificado UML.
Fuente: Elaboración Propia.

DESCRIPCIÓN DE LA ARQUITECTURA DEL SISTEMA

El diseño del sistema de monitoreo se centra en la estación de monitoreo, que actúa como el núcleo del sistema, conectada a varios sensores y módulos de comunicación. A continuación, se detallan los principales componentes.

Sensores y comunicación:

- El **SHT31D** mide temperatura y humedad, conectado mediante I2C, un protocolo de comunicación serial que permite la transmisión de datos entre dispositivos cercanos.
- El **SCD41** mide la concentración de dióxido de carbono (CO₂), también conectado mediante I2C.

- El **PMS7003** (PM2.5) y el ENS160 (COV) utilizan el protocolo UART para la comunicación serie con la estación de monitoreo. Este protocolo permite la transmisión de datos de manera asíncrona.
- El **MQ7** (monóxido de carbono) se conecta directamente a un pin analógico de la estación, proporcionando una salida proporcional a la concentración de CO.
- Módulos **LoRa SX1278**: Estos módulos permiten la transmisión de datos a largas distancias utilizando LoRa, un protocolo de comunicación de largo alcance y bajo consumo. Establecen una conexión entre el transmisor y el receptor, asegurando que los datos recolectados por la estación lleguen al servidor.
- Protocolo **MQTT**: El MQTT se utiliza para enviar los datos recolectados al servidor. Es un protocolo ligero diseñado para dispositivos con ancho de banda limitado, lo que lo hace ideal para este proyecto. Los datos son almacenados en InfluxDB y posteriormente visualizados en Grafana.
- **Visualización de datos en Grafana**: La interfaz Grafana recibe los datos almacenados en el servidor a través de MQTT, permitiendo a los usuarios monitorear y analizar los datos de forma gráfica e intuitiva. Grafana facilita la creación de paneles personalizados, lo que permite a los usuarios observar tendencias y obtener reportes detallados sobre las condiciones ambientales.

Ahora bien, con base a los diagramas de casos de uso, se procede a realizar el sistema de monitoreo de manera tangible, teniendo en cuenta la etapa receptora y trasmisora del sistema. Para realizar la programación del software para obtener las lecturas ambientales se tiene la etapa trasmisora, la cual consiste en la placa controladora ESP32 principal y los sensores que permiten medir variables ambientales. Como se puede visualizar en la figura 6, se considera la etiqueta de los pines de conexión entre la tarjeta microcontroladora y los sensores para tener una mayor comprensión del circuito en el momento de ensamble. Por otro lado, la etapa receptora, se compone de una ESP32 y un módulo LoRa, funcionando como etapa que realiza la conexión

WiFi para posteriormente visualizar los datos vía Grafana. El esquema de conexión de las etapas receptora y transmisora se muestran en la figura 6. Además, en anexos se encuentran las tablas donde se especifica cada etiqueta de conexión entre las placas controladoras, módulos LoRa y los sensores teniendo en cuenta la etapa receptora y transmisora del sistema, con el propósito de avanzar a la implementación y programación de los dispositivos y el protocolo MQTT.

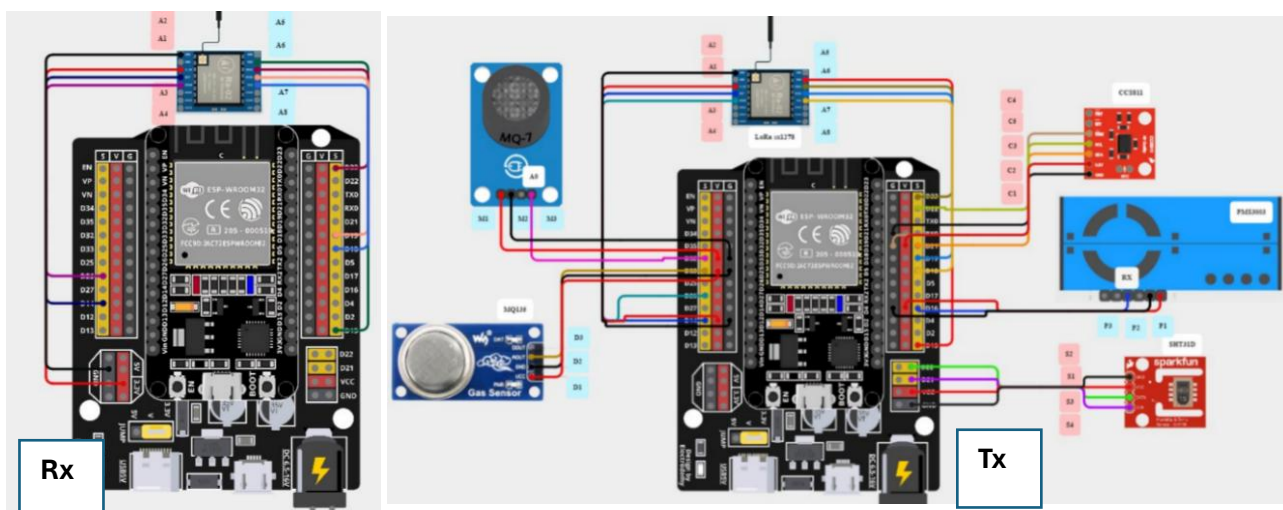


Figura 6. Esquema de la etapa receptora y transmisora del sistema de monitoreo.



Fuente: elaboración propia.

Como se observa en la Figura 5, la etapa receptora del sistema utiliza una placa ESP32 y un módulo LoRa SX1278, cuya función es recibir las mediciones de los sensores a través del enlace LoRa. Luego, el sistema establece una conexión WiFi a una red de 2.4 GHz, ingresando las credenciales necesarias para conectarse a internet. Esta conexión permite transmitir los datos recolectados hacia el almacenamiento en la nube y, posteriormente, visualizar las mediciones a través de la interfaz gráfica. Los componentes utilizados en la etapa receptora se presentan de forma individual en la tabla 3. Por otro lado, en la tabla 4 se muestran los dispositivos utilizados para la etapa transmisora detallando cada uno para facilitar su comprensión. Por otro lado, en el

[anexo 2](#) y en el [anexo 3](#), se tiene más información de las conexiones realizadas en cada etapa de la estación de monitoreo.

Tabla 10.

Dispositivos que pertenecen en la etapa receptora.

| | | |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Placa ESP32</p>  <p>La ESP32 es el microcontrolador principal de la etapa receptora. Es un dispositivo de bajo costo con conectividad WiFi y Bluetooth, ideal para aplicaciones IoT.</p> | <p>Función:</p> <ul style="list-style-type: none"> ✓ Actúa como el cerebro del sistema receptor, gestionando la recepción de los datos de los sensores enviados por el módulo LoRa. ✓ Establece la conexión a internet para transmitir los datos a la plataforma de almacenamiento y visualización. | <p>Especificaciones Claves:</p> <ul style="list-style-type: none"> - Procesador: Dual-core, 240 MHz. -Conectividad: WiFi 2.4 GHz y Bluetooth. -Entradas/Salidas: Soporta comunicación I2C, SPI, UART. -Voltaje de operación: 3.3V. |
| <p>Modulo LoRa sx1278</p>  <p>El módulo LoRa SX1278 permite la transmisión de datos a larga distancia mediante la tecnología LoRa (Long Range), ideal para aplicaciones que requieren comunicación a grandes distancias con bajo consumo de energía.</p> | <p>Función:</p> <ul style="list-style-type: none"> -Recibe las mediciones de los sensores que son transmitidas desde la etapa transmisora. -Envía estos datos a la ESP32 para que puedan ser procesados y transmitidos a la nube. | <p>Especificaciones Clave:</p> <ul style="list-style-type: none"> -Frecuencia de operación: 433 MHz (puede variar según la región). -Alcance: Hasta 10 km en condiciones óptimas. -Modulación: LoRa (baja tasa de transmisión, larga distancia). |



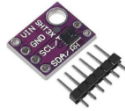

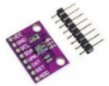

Nota: Esta tabla muestra la descripción de cada componente utilizado en la etapa receptora del sistema.

Fuente: Elaboración Propia.

Tabla 11.

Dispositivos que pertenecen en la etapa transmisora.

| Sensores | Descripción | Especificaciones clave |
|----------|-------------|------------------------|
|----------|-------------|------------------------|

| | | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>En la etapa transmisora, se encuentran los sensores que recopilan datos de variables como temperatura, humedad, CO, CO₂, COV y PM2.5.</p>  <p>PMS3003 (PM2.5)</p>  <p>MQ7 (CO)</p>  <p>SHT31D (Temperatura y Humedad)</p>  <p>MQ135 (CO₂)</p>  <p>CCS811 (COV)</p>  <p>Modulo LoRa sx1278</p> | <p>Los datos que se recopilan en esta etapa por medio de la implementación y programación de los sensores por medio de otra placa ESP32. Para ser enviados mediante un módulo LoRa sx1278 a la etapa receptora. Con el objetivo de realizar el siguiente paso, el cual es almacenar y visualizar los datos de manera gráfica.</p> | <p>SHT31D (Temperatura y Humedad): Precisión $\pm 2\%$ RH, $\pm 0.3^\circ\text{C}$.</p> <p>PMS3003 (PM2.5): Detección de partículas en suspensión en el aire.</p> <p>MQ7 (CO), MQ135 (CO₂) y CCS811(COV): Detectores de gases.</p> <p>-Modulación: LoRa (baja tasa de transmisión, larga distancia).</p> |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

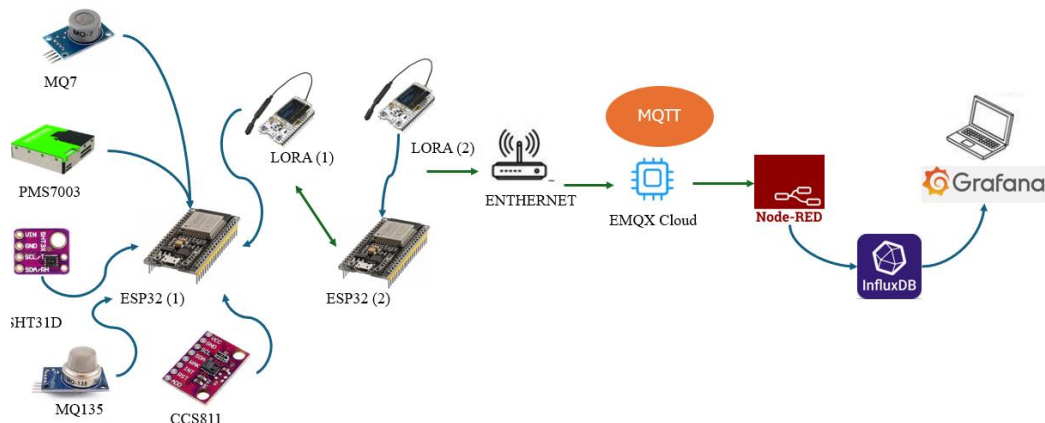
Nota: Esta tabla muestra los sensores utilizados en la etapa transmisora del sistema. Fuente: Elaboración propia.

DESARROLLO DE LA FASE 3. IMPLEMENTACIÓN DEL SISTEMA

En esta fase, se implementó el sistema de monitoreo ambiental utilizando el protocolo MQTT para la comunicación entre los diferentes componentes. El sistema fue diseñado e implementado de manera local, utilizando la herramienta Grafana para la visualización de los datos y InfluxDB para su almacenamiento. El servidor MQTT se configuró para gestionar la publicación y suscripción de los "topics", mientras que Node-RED se implementó para la conexión de los nodos y la integración de los sensores con las herramientas de visualización. A continuación, se describe el proceso de implementación y configuración del sistema.

ESQUEMA DEL SISTEMA DE MONITOREO

El sistema está compuesto por una estación de monitoreo que se comunica con un servidor a través del protocolo MQTT, enviando las mediciones de las variables ambientales. La visualización de estos datos se realiza en Grafana, y el almacenamiento de estos, en InfluxDB. En la figura 7, se muestra el esquema gráfico de la arquitectura del sistema de monitoreo



*Figura 7. Esquema grafico del sistema de monitoreo.
Fuente: Elaboración propia.*

LECTURA DE LOS PARAMETROS AMBIENTALES

Para medir las variables ambientales, se emplearon diversas librerías en el microcontrolador ESP32:

- Para la temperatura y humedad, se utilizó la librería <Adafruit_SHT31.h>.
- Para el dióxido de carbono (CO₂), se empleó la librería <MQ135.h>, configurada en el pin 33.
- Para el material particulado (PM2.5), se utilizó la librería <PMS.h>, configurando los pines 16 y 17 para la comunicación serial (SoftwareSerial).
- La librería <SPI.h> y <LoRa.h> se utilizaron para la transmisión y recepción de datos a través del bus SPI y el módulo LoRa SX1278.
- La conectividad WiFi en la ESP32 se habilitó con la librería <WiFi.h>.
- Las lecturas de los sensores se programaron para realizarse cada 5 segundos utilizando la librería <Ticker.h>.

En la figura 8, se muestra el código utilizado para la configuración de pines y librerías en el ESP32 de la etapa transmisora, para la obtención de las variables ambientales

```

void temperaturaHumedad() {
  temperatura = sht31.readTemperature();
  humedad = sht31.readHumidity();
}

void dioxidodecarbono() {
  rawValueMQ7 = analogRead(MQ7_PIN);
  rawValueMQ135 = analogRead(MQ135_PIN);
  ppmMQ7 = convertToPPM(rawValueMQ7);
}

void leerPMS3003() {
  pms.wakeUp(); // Asegúrate de que el sensor esté despierto
  delay(3000); // Espera 3 segundos para estabilizar la lectura
  pms.requestRead(); // Solicita una lectura

  if (pms.readUntil(data)) {
    pm25 = data.PM_AE_UG_2_5;
  } else {
    Serial.println("Error al leer los datos del PMS3003");
  }
  pms.sleep(); // Pone el sensor en modo de suspensión para ahorrar energía
}

```

Figura 8. Código para la definición de librerías y pines utilizados para medir las variables ambientales. Fuente: Elaboración Propia.

Para obtener las mediciones de temperatura, humedad, material particulado, compuestos volátiles, dióxido y monóxido de carbono se construyeron unas funciones en Arduino para facilitar la lectura de las variables ambientales. En la figura 9 se puede visualizar el código de funciones para la obtención de las variables ambientales.

```

2 #include <Wire.h>
3 #include <SPI.h>
4 #include <LoRa.h>
5 #include <MQ135.h>
6 #include <Ticker.h>
7 #include <Adafruit_SHT31.h>
8 #include <PMS.h>
9 #include <SoftwareSerial.h>
10
11 #define LORA_SS 15 // Pin SS
12 #define LORA_RST 14 // Pin RST
13 #define LORA_DIO0 26 // Pin DIO0
14
15 // Definición de los pines para los sensores
16 #define SDA_PIN 21
17 #define SCL_PIN 22
18 #define MQ7_PIN 32
19 #define MQ135_PIN 33
20 #define PMS_RX 16 // RX del PMS3003 conectado a TX de la ESP32 (GPIO 16)
21 #define PMS_TX 17 // No se utiliza, pero se define para la inicialización de SoftwareSerial
22

```

Figura 9. Código para la obtención de las variables ambientales. Fuente: Elaboración Propia.

CONFIGURACIÓN DE BRÓKER EMQX

El bróker MQTT se configuró utilizando EMQX, desplegado en la nube. Se creó una cuenta gratuita en EMQX y se configuraron las credenciales de acceso, como usuarios y

contraseñas, en la sección de autenticación del portal. El bróker se encarga de gestionar la publicación y suscripción de "topics", lo que permite que los sensores envíen sus datos al servidor y los clientes (como Node-RED y Grafana) reciban la información. En la Figura 10, se muestra el usuario (darwinm) creado para realizar la configuración del Broker en EMQX.

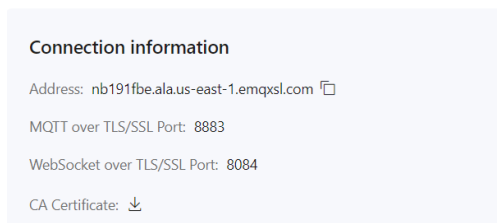


Figura 10. Usuario de autenticación asignado en el bróker. Fuente: Elaboración Propia.

La ESP32 se configuró como cliente MQTT, utilizando el bróker EMQX para la transmisión de datos de los sensores. Cada sensor publica sus mediciones en diferentes "topics", organizados jerárquicamente, lo que permite la suscripción a esos topics desde cualquier cliente compatible con MQTT. La conexión al bróker se realizó en el puerto 1883. Previamente, se instala y certifica el centro de datos EMQX como se muestra en la figura 11, proporcionando los datos necesarios para la implementación de clientes MQTT, como el puerto, el serial y el certificado de conexión.

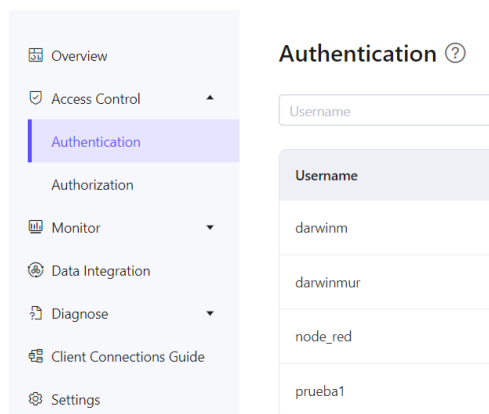


Figura 11. Certificado de conexión EMQX. Fuente: <https://cloud-intl.emqx.com/console/deployments/nb191fbe/overview>

Al instalar y configurar EMQX, en puerto por defecto 1883, se generó la opción de conexión al bróker, los dispositivos y aplicaciones para realizar el suscripción y publicación de datos. Para verificar el funcionamiento del bróker en el localhost, se puede utilizar el símbolo del sistema (CMD) o la terminal, como también, la descarga de la aplicación MQTTX que se encuentra disponible para dispositivos móviles o computadores, A continuación, se presenta el estado de funcionamiento del envío y recibimiento de los topics, variables que genera el protocolo MQTT para la transmisión y recepción de los datos en el servidor cliente. La siguiente figura muestra la publicación y suscripción de los topics con el protocolo MQTT y el bróker EMQX como se muestra en la figura 12.

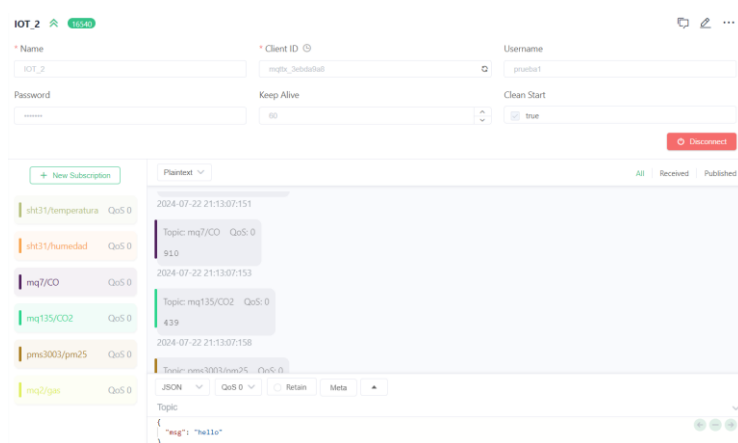


Figura 12. Conexión con el Bróker MQTT. Fuente: Elaboración Propia.

PROGRAMACIÓN Y CONFIGURACIÓN DEL PROTOTIPO

Para la implementación del prototipo, se añadieron conexiones adicionales a la estación de monitoreo y se integraron las librerías correspondientes para la conexión con el bróker MQTT y la lectura de los sensores. Las librerías incluidas permiten gestionar el tiempo de muestreo, manejar el almacenamiento en la memoria no volátil y controlar la conectividad WiFi. En la

figura 13, se muestran las librerías que fueron implementadas en la programación de recepción de la estación de monitoreo.

```

1  #include <Wire.h>
2  #include <SPI.h>
3  #include <LoRa.h>
4  #include <MQ135.h>
5  #include <Ticker.h>
6  #include <Adafruit_SHT31.h>
7  #include <PMS.h>
8  #include <SoftwareSerial.h>
^

```

Figura 13. Librerías utilizadas en estación transmisora del prototipo. Fuente: Elaboración Propia.

En la etapa receptora de la estación de monitoreo, la ESP32 se configuró para conectarse a una red WiFi utilizando <WiFiManager>. Esto permite cambiar la red a la que está conectada la estación presionando el botón de reinicio durante 5 segundos. El acceso a la interfaz de configuración se realiza a través de la IP 192.168.4.1. En la figura 14, se muestran las librerías utilizadas en la etapa receptora de la estación de monitoreo.

```

1  #include <WiFi.h> // Biblioteca para WiFi en ESP32
2  #include <WiFiClientSecure.h> // Biblioteca para conexiones seguras (SSL/TLS)
3  #include <PubSubClient.h> // Biblioteca para MQTT
4  #include <SPI.h>
5  #include <LoRa.h> //Biblioteca para uso de los modulos LoRA
6  #include <WiFiManager.h> // Incluye la biblioteca WiFiManager

```

Figura 14. Librerías utilizadas en el programa receptora de la estación. Fuente: elaboración propia.

Desde monitor serial del IDE de Arduino se puede visualizar el estado de conexión y habilitación al portal web, en la figura 15 se muestra el resultado desde el monitor serial.

```

13:38:09.396 -> *wm:connectTimeout not set, ESP waitForConnectResult...
13:38:11.872 -> *wm:AutoConnect: FAILED for 3099 ms
13:38:11.911 -> *wm:StartAP with SSID: MiEstacionMonitoreo
13:38:12.445 -> *wm:AP IP address: 192.168.4.1
13:38:12.445 -> *wm:Starting Web Portal

```

Figura 15. Estado de acceso al Porta Web. Fuente: elaboración propia.

Por otro lado, desde el navegador predeterminado se puede conectar a un punto de acceso, introduciendo las credenciales correctas para realizar la configuración de la red, teniendo la posibilidad de visualizar las redes disponibles para posteriormente realizar el ingreso con las credenciales correctas, que tienen en cuenta que las redes hábiles para conectarse desde la placa controladora son redes con banda de frecuencia de 2.4 GHz, como se muestra en la figura 16.



Figura 16. Configuración para gestionar redes WiFi desde el portal web. Fuente: elaboración propia.

Para configurar el Bróker desde la etapa receptora, se deben incluir las credenciales correspondientes, como la dirección, usuario, contraseña, puerto y certificado de seguridad. Esto implica descargar el certificado del bróker oficial y asignarlo al programa como una cadena de caracteres distribuidos en varias líneas de código, como se muestra en la figura 17.

```

23 // Parámetros del servidor MQTT
24 const char* mqtt_server = "nb191fbc.ala.us-east-1.emqxsl.com";
25 const int mqtt_port = 8883;
26 const char* MQTT_USER = "darwin";
27 const char* MQTT_PASSWORD =
28 const char* mqtt_server_cert = "-----BEGIN CERTIFICATE-----\n"
29 | "MIIDrzCCApegAwIBAgIQCDVgVpBCRRGhdWrJWZHHSjANBgkqhkiG9w0BAQUFADBh\n"
30 | "MQswCQYDVQQGEwJVUzEVMBMGA1UEChMMR6l1naUN1cnQgSW5jMRkwFwYDVQQLExB3\n"
31 | "d3cuZG1naW11cnQuYy9tMSAwHgYDVQDEdEaWdpQ2VydCBHbG9iYWwUM9vdCBD\n"
32 | "QTAEFw0wNjExMTAwMDAwMDBaFw0zMTEwMTAwMDAwMDBaMGExCzAJBgNVBAYTA1VT\n"
33 | "MRUwEwYDVQQKEwxEaWdpQ2VydCBJb250dXN0ZXN0ZXN0ZXN0ZXN0ZXN0ZXN0\n"
34 | "b20xIDAeBgNVBAMTF0RpZ21DZXJ0IEEdsb2JhbCBzY290IENBMIIIBIjANBgkqhkiG\n"
35 | "9w0BAQEFAAOCAQ8AMIIBCgKCAQEAA4jvhEXLeqKTT01eqUKPC3eQyAK17hL011sB\n"
36 | "CSDMAZ0nTjC3U/dDxGkAV53ijSLdhwZAAIEJzs4bg7/fzTtxRuLWZscFs3YnFo97\n"
37 | "nh6Vfe63SKMI2tavegW5BmV/S10fvBf4q77uKNd0f3p4mVmFAG5cIzJL07A6Fpt\n"
38 | "43c/dx//AH2hdmoRBBYmQ11GNXRor5H4idq9Joz+EkIYIVUX7Q6hL+hqkMf77P\n"

```

Figura 17. Parámetros de configuración del servidor MQTT + Bróker. Fuente: elaboración propia.

Para la emisión de topics, a través del microcontrolador, se utiliza la secuencia de control definida en Node-RED. La frecuencia de publicación de los datos mediante topics se ha configurado en intervalos de 5 segundos, lo cual es adecuado para detectar cambios ambientales y mantener un control efectivo sobre los mensajes publicados y suscritos mediante el protocolo MQTT. En la figura 18, se ilustra el método de publicación de los topics de cada variable medida en el programa, donde la etapa transmisora envía los paquetes de datos por medio del módulo LoRa, y la etapa receptora recibe y gestiona la publicación y suscripción de los topics con el protocolo. Como también, en el [anexo 1](#), se pueden visualizar el nombre de los topics utilizados en la programación de la etapa receptora del sistema, con el objetivo de realizar el proceso de suscripción y publicación mediante el protocolo MQTT.

```

111 void loop() {
112   // Envío de datos LoRa cada 5 segundos
113   if ((millis() - tiempoAnterior) >= 5000) {
114     String datos_Sensores = String(temperatura) + "," + String(humedad) + "," + String(ppmMQ7) + "," + String(rawValueMQ135) + "," + String(pm25) +
115     LoRa.beginPacket();
116     LoRa.print(datos_Sensores);
117     LoRa.endPacket();
118     Serial.println(datos_Sensores);
119     tiempoAnterior = millis();
120   }
121 }

```

Figura 18. Método de envío de datos cada 5 segundos a la etapa receptora. Fuente: elaboración propia.

Para la recepción de los datos se utilizó el método de publicación y suscripción que permite establecer el protocolo, añadiendo las diferentes variables de medición, como también estableciendo la condición del tiempo de envío de los datos en el programa. Empleando el método “callback”, el cual consisten en controlar la recepción de los tópicos, se reconstruyen los mensajes a los bytes recibidos y posteriormente la conexión con Node-RED, InfluxDB y finalmente en Grafana, para la visualización de los tópicos. En la figura 19, se muestra el método de envío y publicación que se realizó en el software de la etapa receptora.

```

137     if ((millis() - tiempoAnterior) >= 5000) {
138         client.publish("sht31/temperatura", String(temperatura).c_str());
139         client.publish("sht31/humedad", String(humedad).c_str());
140         client.publish("mq7/CO", String(ppmMQ7).c_str());
141         client.publish("mq135/CO2", String(ppmMQ135).c_str());
142         client.publish("pms3003/pm25", String(pm25).c_str());
143         client.publish("mq2/gas", String(ppmMQ2).c_str()); // Añadido para MQ2
144         tiempoAnterior = millis();
145     }
146 }
147 client.loop();
148 }
149 }

```

Figura 19. Método de envío y publicación de los datos cada 5 segundos.

Fuente: elaboración propia.

CONFIGURACIÓN EN NODE-RED

Node-RED se utilizó para gestionar los datos recibidos de los sensores y enviar esta información a la base de datos InfluxDB. Se instalaron nodos de entrada, procesamiento y salida que permitieron la suscripción a los "topics" publicados por la ESP32. La configuración se realizó utilizando el paquete "node-red-contrib-influxdb".

Para lograr la interacción con el usuario, se agregaron nodos desde el "Manage palette" como se ilustra en la figura 20. También se instaló el paquete denominado "node-red-dashboard" para mejorar la funcionalidad de la interfaz. La herramienta Node-RED, posee nodos de interconexión que puedes relacionarse fácilmente al proyecto por medio de la incorporación de la llamada Dashboard, que la es la encargada de crear la etapa visual de la interfaz, teniendo la posibilidad de visualizar el estado del servicio de Node-Red, por medio del símbolo del sistema, como se muestra en la figura 21.

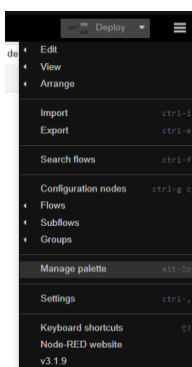


Figura 20. Opción en node-RED para añadir y configurar nodos.

```

node-red
19 Jun 18:33:04 - [info]
Welcome to Node-RED
=====
19 Jun 18:33:04 - [info] Node-RED version: v3.1.9
19 Jun 18:33:04 - [info] Node.js version: v20.12.2
19 Jun 18:33:04 - [info] Windows_NT 10.0.22631 x64 LE
19 Jun 18:33:08 - [info] Loading palette nodes
19 Jun 18:33:14 - [info] Dashboard version 3.6.5 started at /ui
19 Jun 18:33:14 - [info] Settings file : C:\Users\darwi\.node-red\settings.js
19 Jun 18:33:14 - [info] Context store : 'default' [module=memory]
19 Jun 18:33:14 - [info] User directory : \Users\darwi\.node-red
19 Jun 18:33:14 - [warn] Projects disabled : editorTheme.projects.enabled=false
19 Jun 18:33:14 - [info] Flows file : \Users\darwi\.node-red\flows.json
19 Jun 18:33:14 - [info] Server now running at http://127.0.0.1:1880/
19 Jun 18:33:14 - [warn]

```

Figura 21. Estado del servicio Node-RED.

Fuente: elaboración propia.

Para configurar Node-RED, fue necesario establecer nodos de entrada que reciben los datos de los sensores conectados a la ESP32 y al módulo LoRA (receptor). Los nodos de procesamiento filtran estos datos según sea necesario, y los nodos de salida envían los datos ya procesados a la base de datos InfluxDB, presentándolos de manera gráfica en la interfaz de Grafana. El uso de Node.js es esencial, ya que implementa el servidor backend, encargado de gestionar la comunicación entre la ESP32 y los servicios de almacenamiento y visualización de datos. Node.js se comunica directamente con el bróker MQTT (EMQX) para recibir los datos de los sensores y almacenarlos en InfluxDB.

Node-RED, que incluye nodos de comunicación MQTT preinstalados, facilita la suscripción y publicación de los datos. Esto permite que el software Node-RED acceda al localhost y se suscriba a los topics establecidos en el firmware. La principal ventaja de Node-RED es su orientación a la programación de flujos de datos IoT, lo que simplifica la distribución de la interfaz y la visualización gráfica de los datos, añadiendo los nodos para acceder a los registros almacenados, como se muestra en la figura 22, donde se establecen los nodos de conexión y flujo de datos.

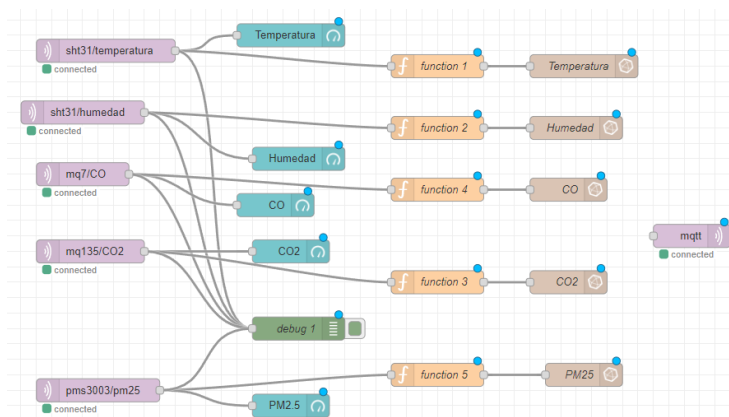


Figura 22. Flujo para la lectura de los sensores en Node-RED. Fuente: elaboración propia.

CONFIGURACIÓN EN INFLUXDB

InfluxDB se utilizó como herramienta para el almacenamiento de datos, operando como un servidor local. El acceso a InfluxDB se realizó mediante el puerto 8086, donde fue necesario realizar algunas configuraciones iniciales. Estas configuraciones incluyeron la creación de una organización, un usuario y un espacio de almacenamiento en el puerto Localhost 8086 de InfluxDB, como se puede visualizar en la figura 23.

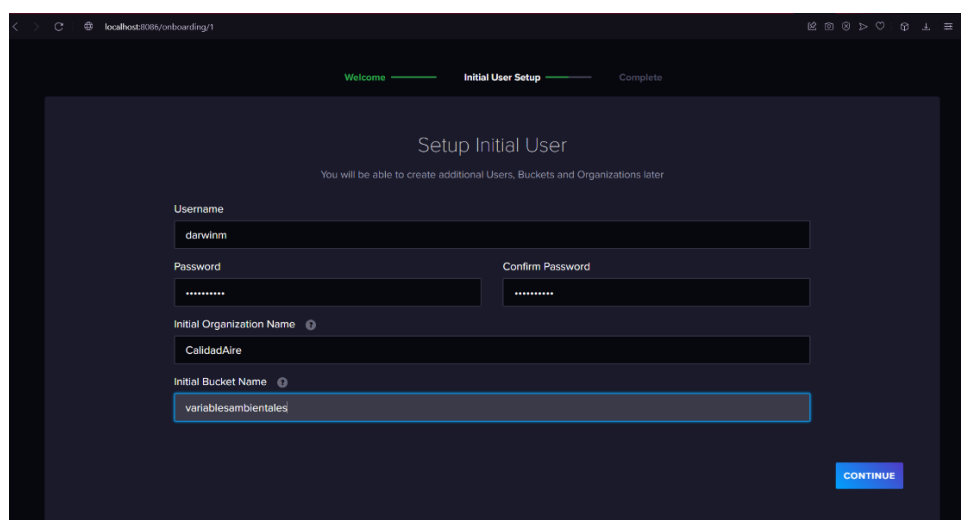


Figura 23. Configuración de usuario inicial en InfluxDB. Fuente: Elaboración Propia.

En el siguiente paso, se generó un token de API para acceder a los datos almacenados en la herramienta. Dentro de la pestaña "Load Data", se encuentra la opción para crear una API con permisos completos o personalizados. El token de API generado se asoció con el espacio de almacenamiento previamente creado, denominado variables_ambientales, configurándolo con todos los permisos necesarios, como se muestra en la figura 24.

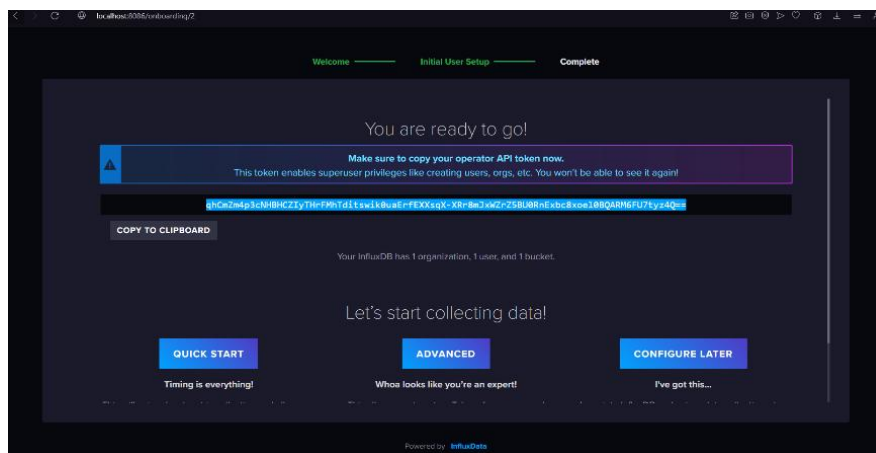


Figura 24. API generado en InfluxDB. Fuente: elaboración propia.

En la interfaz de InfluxDB encontramos la pestaña “Load Data”, donde es posible personalizar el periodo de tiempo durante el cual se deseaba almacenar los datos en el espacio de almacenamiento “variables_ambientales”. En primera instancia, la configuración estaba en “never”, lo que indicaba que los datos no se eliminarían de manera automática; configurándolo de tal manera que el tiempo de almacenamiento sea de 30 días, para que el mismo sistema procesa a borrar los datos almacenados. Para dicha configuración, es necesario editar el Bucket de InfluxDB como se muestra en la figura 25.

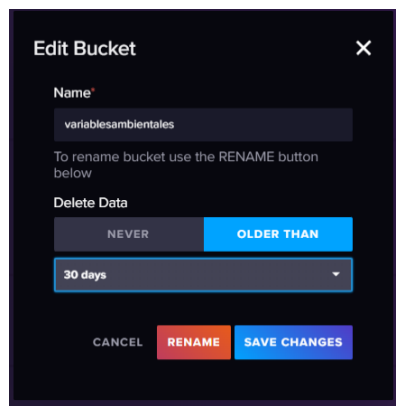
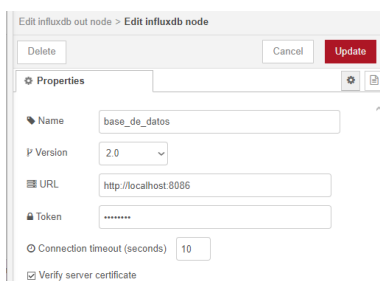


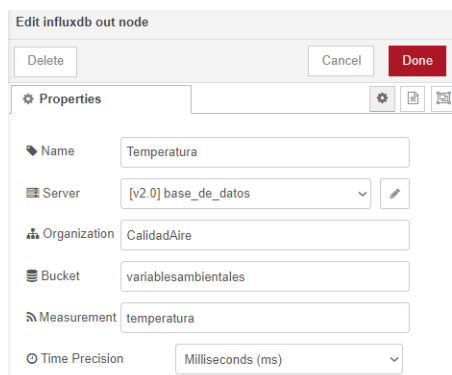
Figura 25. Modificación del tiempo de almacenamiento en InfluxDB. Fuente: elaboración propia.

Para integrar los datos, se añadieron nodos en la herramienta Node-RED, permitiendo la conexión a la base de datos y la recepción de las variables ambientales. Los nodos se añadieron mediante la opción "Manage palette", utilizando el paquete "node-red-contrib-influxdb". Los nodos de salida se configuraron con la dirección de la base de datos y el token necesario para realizar las escrituras de los valores, como se ilustra en la figura 26, donde se realiza la edición del nodo, que realiza la conexión con la base de datos.



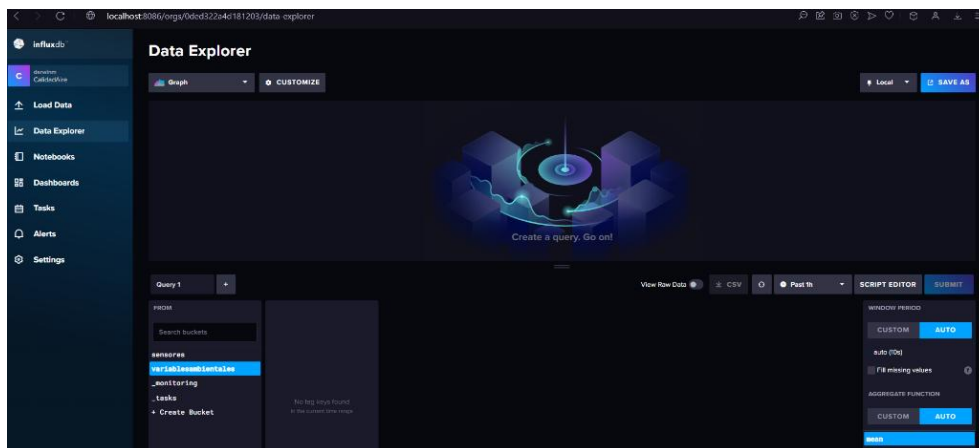
*Figura 26. Servidor InfluxDB configurado en Node-RED.
Fuente: elaboración propia.*

Fue indispensable añadir información adicional a los nodos, como el nombre de la organización, el espacio de almacenamiento de los datos y una designación para cada tipo de medición. En este caso, las mediciones incluyeron temperatura, humedad, dióxido de carbono, monóxido de carbono, material particulado y compuestos orgánicos volátiles. En la figura 27, se puede visualizar la edición que se realizó en cada nodo de entrada y salida de los sensores, para establecer la conexión con la base de datos.



*Figura 27. Configuración de nodo para almacenar mediciones en InfluxDB.
Fuente: elaboración propia.*

Una vez completadas las configuraciones, se verificó que la conexión con la base de datos fue exitosa. Las mediciones realizadas ahora se pueden visualizar en las gráficas de InfluxDB, por medio de la configuración del Data Explorer de InfluxDB como se muestra en la figura 28. Por otro lado, en el [anexo 7](#) se muestra una guía por medio de imágenes, para realizar la configuración e inicio de InfluxDB-



*Figura 28. Mediciones de los sensores en InfluxDB.
Fuente: elaboración propia.*

CONFIGURACIÓN EN GRAFANA

Finalmente, para la visualización de los datos, se configuró Grafana como servidor local, como se muestra en la figura 29, estableciendo la conexión con la base de datos InfluxDB y posteriormente se crearon paneles personalizados para cada variable medida (temperatura, humedad, PM2.5, CO₂, CO y COV). Se ajustaron los gráficos y las escalas para mejorar la interpretación de los datos, configuraciones que se realizan para agregar una nueva conexión como se ilustra en la figura 30.

Figura 29. Integración del servidor Grafana. Fuente: elaboración propia.

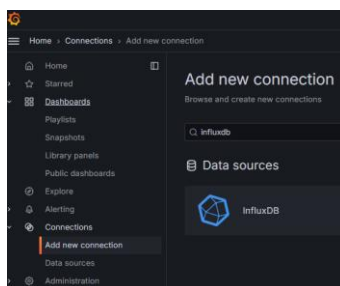


Figura 30. Inicio de conexión como fuente de datos en Grafana. Fuente: elaboración propia.

Para establecer la conexión, fue necesario configurar y seleccionar el lenguaje de consulta adecuado, además de agregar la dirección de la base de datos, como también se muestra en el [anexo 6](#), donde se tiene una guía por medio de imágenes para realizar el inicio del servidor de Grafana de manera local. Posteriormente, se identificó el nombre del bucket creado en InfluxDB, utilizando las credenciales de usuario y contraseña de la organización para la asignación final. En este caso, se optó por usar una API y se seleccionó el método POST como se muestra en la figura 31, donde se realiza la conexión a la base de datos desde Grafana.

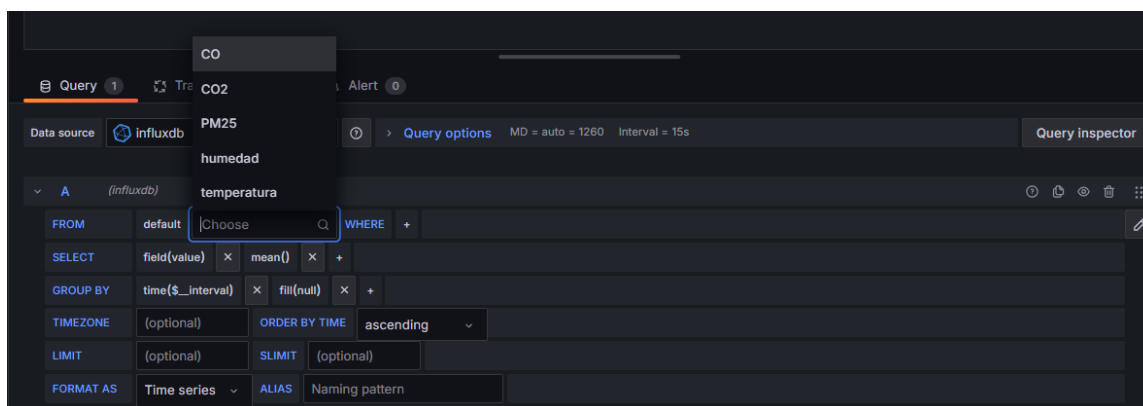
| | |
|-------------------|-------------------------------|
| Database | variablesambientales |
| User | darwinm |
| Password | configured Reset |
| HTTP Method | POST |
| Min time interval | 10s |
| Max series | 1000 |

Delete Save & test

Figura 31. Conexión a la base de datos InfluxDB desde Grafana. Fuente: elaboración propia.

Para la visualización de los datos de manera gráfica, se añadieron las visualizaciones en un nuevo dashboard, asignando la fuente de datos previamente configurada en cada gráfico. Por otro lado, dentro de las demás configuraciones se seleccionaron las mediciones creadas (Temperatura, Humedad, Material Particulado 2.5, Compuestos Volátiles, Dióxido y Monóxido de carbono). En la personalización del panel de visualización, se ajustó el tipo de visualización, configurando títulos, colores y leyendas para mejorar la claridad, el análisis y la interpretación de los datos. Además, se modificaron los ejes y las escalas de los gráficos para asegurar una visualización precisa y fácil de comprender.

Para las demás variables monitoreadas, se siguió el mismo procedimiento, creando múltiples paneles dentro del mismo dashboard para lograr una visualización completa e integral del sistema de monitoreo, como se puede visualizar en la figura 32, donde se realiza la selección y configuración de los datos y gráficos en Grafana.



*Figura 32. Selección de datos y configuración de gráficos en Grafana.
Fuente: elaboración propia.*

DESARROLLO DE LA FASE 4. PRUEBAS Y EVALUACIÓN.

En esta última fase, se llevó a cabo un proceso riguroso de pruebas y evaluación del funcionamiento de la estación de monitoreo, con el objetivo de validar la transmisión de datos y

el rendimiento de cada componente del sistema. Las pruebas se realizaron de manera progresiva, comenzando con la configuración e integración del protocolo MQTT y el sensor SHT31D, que mide las variables de temperatura y humedad. Este paso fue esencial para verificar la correcta transmisión de datos a través de la red, asegurando el correcto funcionamiento de la placa ESP32, el sensor, y la integración del protocolo MQTT.

Se comprobó que los datos obtenidos por el sensor SHT31D se transmitieran adecuadamente, verificando su precisión y visualización a través del monitor serial del IDE de Arduino. Además, se utilizó la aplicación web MQTTX para confirmar la transmisión y recepción de los datos del sensor, así como para visualizar el estado de la conexión con el protocolo MQTT como se ilustra en la figura 33.

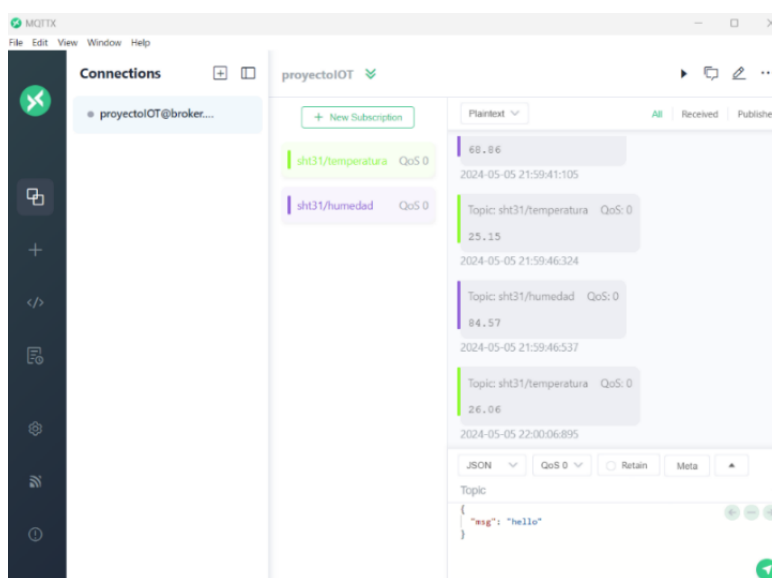
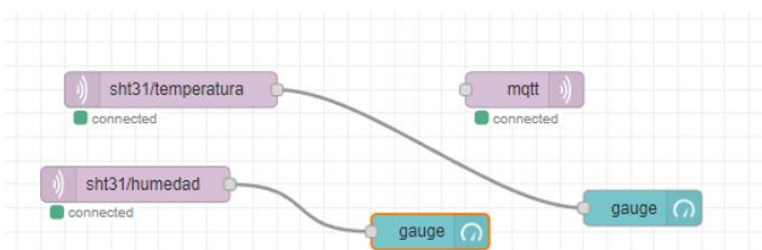


Figura 33. Verificación del funcionamiento del protocolo y recepción de datos.
Fuente: elaboración propia.

INTEGRACIÓN MODULOS LORA

A continuación, se integraron los módulos LoRa para establecer un sistema de transmisión y recepción de datos a largo alcance, utilizando dos placas ESP32. Se realizaron

pruebas para garantizar que los datos de temperatura y humedad provenientes del sensor SHT31D fueran transmitidos y recibidos de manera confiable entre los módulos LoRa. Para gestionar la entrada y salida de datos, se utilizaron los nodos mqtt_in y mqtt_out en Node-RED, logrando una comunicación continua y fluida. En la figura 34, se puede observar el primer nodo que se configuró para la establecer conexión por medio del protocolo MQTT.



*Figura 34. Conexión de nodos de entrada y de salida del protocolo MQTT.
Fuente: elaboración propia.*

Los módulos LoRa fueron seleccionados por su capacidad para transmitir datos a largas distancias con un bajo consumo energético. Estos módulos fueron configurados para operar en una red punto a punto, donde uno actúa como emisor de los datos recolectados por los sensores, y otro como receptor en la estación central.

Los datos monitoreados por los sensores se empaquetan y se transmiten a través de una frecuencia de radio específica, previamente configurada para maximizar el alcance y evitar interferencias. El módulo receptor procesa los datos recibidos y los transmite mediante el protocolo MQTT para su almacenamiento y análisis. En la Figura 35, se muestran los componentes utilizados durante las pruebas de integración de los módulos LoRa.

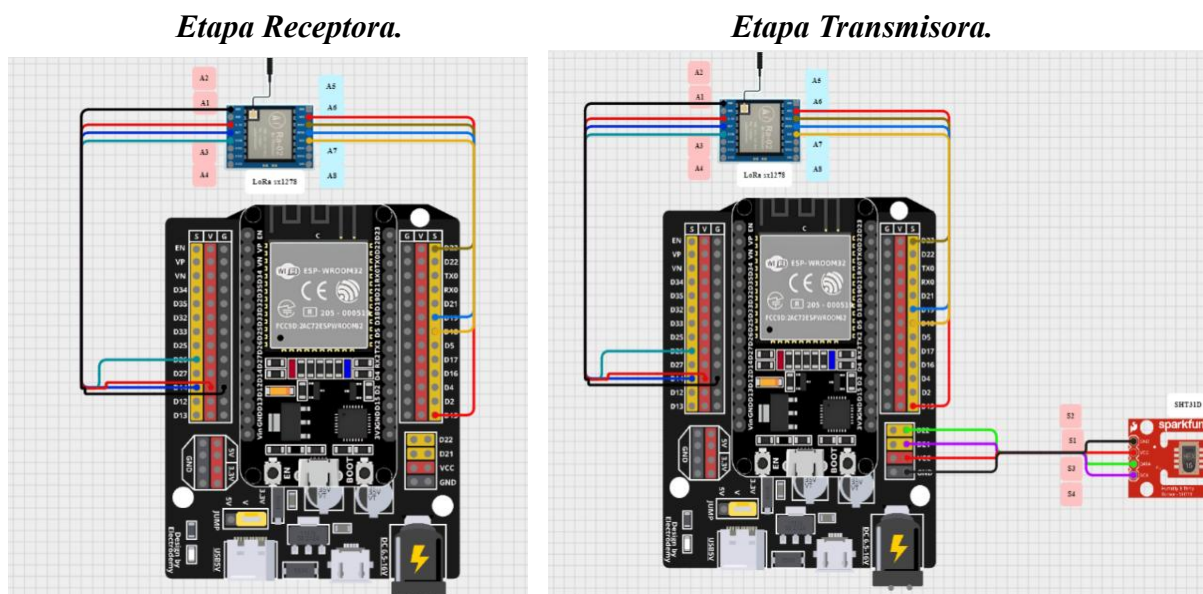


Figura 35. Componentes utilizados para el periodo de integración del protocolo y módulos LoRa.
Fuente: elaboración propia.

Además, en los anexos 2 y 3, se muestran los componentes utilizados en la etapa receptora y transmisora, teniendo como ayuda una tabla de etiquetas de identificación para tener como guía, el proceso de conexión de los componentes a cada etapa del sistema.

VISUALIZACIÓN DE DATOS EN NODE-RED

Para visualizar los datos obtenidos, se conectó la salida a un nodo Gauge en Node-RED, asignando etiquetas a las variables de temperatura y humedad. Esto permitió personalizar los colores y valores del medidor. Se accedió al gráfico mediante la dirección localhost:1880/ui, como se le ilustra en la figura 36, lo que generó una página web adaptativa que mostraba los datos en tiempo real.

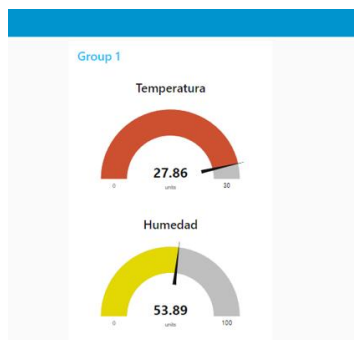


Figura 36. Visualización grafica por medio de nodos Gauge en Node-Red.
Fuente: elaboración propia.

INTEGRACIÓN DE SENSORES ADICIONALES

Una vez validada la integración básica del sistema, se incorporaron sensores adicionales como el PMS3003 (para medir PM2.5), el MQ7 (para CO), y el MQ135 (para CO₂). Se configuraron nodos en Node-RED para recibir y procesar estos datos, verificando que la transmisión de la información mediante MQTT fuera correcta y que los datos se almacenaran adecuadamente en InfluxDB. En el [anexo 8](#), se muestra por medio de imágenes el proceso para la integración de las variables en la base de datos. Posteriormente, se realizaron ajustes para asegurar la precisión de las lecturas, configurando correctamente los algoritmos de los sensores y ajustando variables como la resistencia y el tiempo de estabilización como se ilustra en la figura 37.

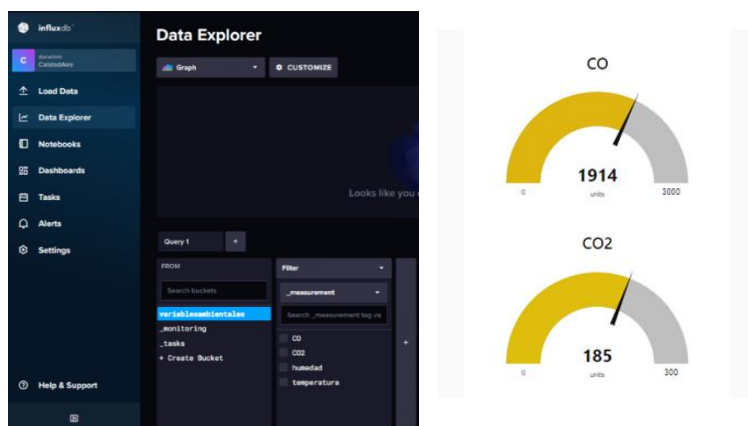


Figura 37. Verificación y visualización en el Dashboard en InfluxDB y Node-RED. Fuente: elaboración propia.

Dada por terminada la etapa la integración de los sensores adicionales al sistema, se ha logrado cumplir el primer objetivo específico del proyecto, teniendo en cuenta que la incorporación de los sensores se realizó de manera satisfactoria, puesto que, las variables obtenidas por medio de cada sensor se pueden visualizar en InfluxDB para su almacenamiento de manera local.

CONFIGURACIÓN DE LA ESTACIÓN DE MONITOREO

Durante la integración de los sensores a la placa ESP32, se prestó atención a la correcta selección de pines y materiales de conexión para garantizar una instalación física robusta, minimizando el riesgo de desconexiones o fallas de alimentación, como también el proceso de calibración que tiene cada sensor, procedimiento que se describe en el [anexo 5](#). Además, en la figura 38, se muestra la estación de monitoreo, teniendo en cuenta la etapa de recepción y de transmisión.

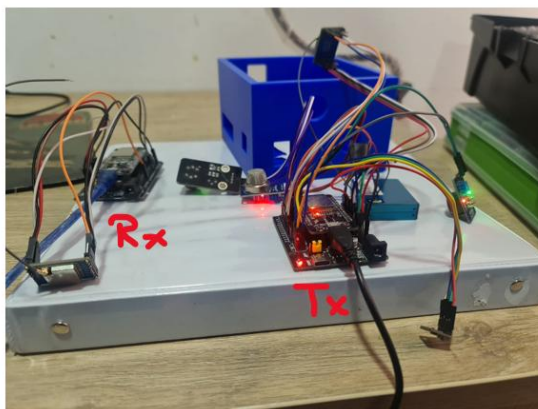


Figura 38. Receptor y transmisor de la estación de monitoreo. Fuente: elaboración propia.

VISUALIZACIÓN DE DATOS EN GRAFANA

En la etapa final, se configuró Grafana para visualizar los datos almacenados en InfluxDB. Se estableció la conexión con la base de datos y se crearon paneles personalizados para monitorear en tiempo real las variables de temperatura, humedad, PM2.5, CO, CO₂, y COV. A lo largo de esta fase, se realizaron ajustes en la configuración de los gráficos para optimizar la

visualización de los datos como se ilustra en la figura 39, asegurando que todos los sensores funcionaran de manera precisa y sincronizada.

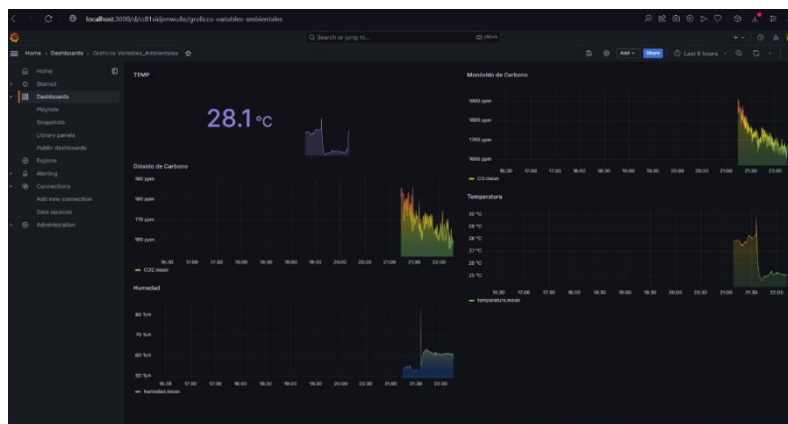


Figura 39. Visualización gráfica de los datos almacenados en InfluxDB desde Grafana. Fuente: elaboración propia.

Para garantizar un funcionamiento óptimo de la visualización gráfica, se llevaron a cabo diversas pruebas y modificaciones en el firmware de la estación de monitoreo. Estas pruebas confirmaron que los datos de los sensores se transmitían correctamente a MQTT y posteriormente a InfluxDB, asegurando que finalmente pudieran visualizarse en Grafana. Se verificó la accesibilidad del sistema desde cualquier dispositivo con conexión a internet. Además, se realizaron ajustes y optimizaciones para asegurar la estabilidad y seguridad del sistema de monitoreo. En la figura 40, se puede visualizar el primer dato obtenido de manera gráfica, siguiendo los mismos pasos para los demás nodos de conexión que se estipularon en Node-RED.

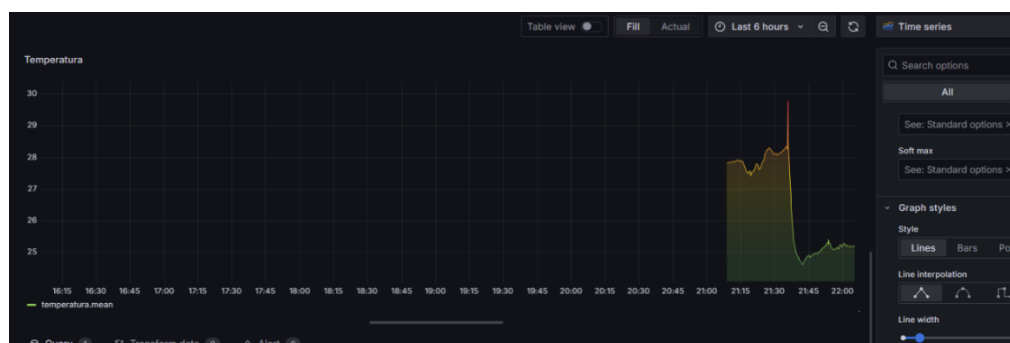


Figura 40. Verificación de funcionamiento de la visualización en Grafana. Fuente: elaboración propia.

Finalmente, De la misma manera, siguiendo la metodología en cascada, cada vez que se realizó el vínculo de los sensores, con el protocolo de comunicación MQTT y la conexión establecida por la interfaz Grafana, se realizaron pruebas de evaluación, antes de proceder con integración del siguiente sensor. En otras palabras, todas las variables ambientales se integraron en el sistema y se pudo visualizar en diferentes paneles en un solo dashboard en Grafana. Aparte de las mencionadas, esto abarcaba la temperatura, la humedad, el PM2.5, el monóxido de carbono, el dióxido de carbono y los compuestos orgánicos volátiles. Cada una de ellas se presentó en un panel, lo que permitía ver el estado completo y detallado del monitoreo de las variables ambientales, como se puede visualizar en la siguiente figura 41, Obteniendo la visualización grafica de cada variable medida cada 5 minutos, configuración implementada en el software y en la arquitectura del sistema, cumpliendo así, el segundo objetivo específico del presente proyecto, estableciendo la integración del protocolo LoRa y MQTT en el sistema.



*Figura 41. Visualización Grafica de las variables en una Dashboard dentro de Grafana.
Fuente: Elaboración Propia.*

Ahora bien, para realizar el contraste y comparaciones de mediciones de calidad del aire en diferentes entornos, se toma en primera instancia como referencia Ecoparque Sabana, debido a que su característica principal, la cual es, una reserva natural, por otro lado, se compara con un

área urbana de la Universidad Pedagógica Nacional. Las mediciones realizadas en Ecoparque tienen como función principal, establecer una referencia de la calidad del aire, siendo este, un área, donde se tiene menos impacto por actividades humanas intensivas, como el tráfico vehicular o industrias, sin embargo, se logró evidenciar que las industrias aledañas a la reserva natural generan una variación en la calidad del aire de Ecoparque.

Las mediciones realizadas en la reserva natural se establecieron siguiendo los estándares establecidos y descritos en la parte conceptual, teniendo en cuentas las guías de la EPA, el índice de calidad del aire (ICA) y la resolución 2254 de 2017 del Ministerio de Ambiente, donde se monitorearon los parámetros establecidos en el diseño e implementación de la estación de monitoreo.

- Monóxido de Carbono (CO)
- Dióxido de Carbono (CO₂)
- Compuestos Volátiles Orgánicos (COV)
- Material Particulado (PM_{2.5})
- Temperatura y Humedad.

Para realizar la medición de cada variable ambiental, se utilizó como referencia la estación fija de monitoreo del Portal Aire Ciudadano, ubicado en la casa de la Tingua de Ecoparque Sabana, como también, la estación certificada de la CAR que se encuentra en la reserva natural. Ubicando la estación de monitoreo a la misma altura y posición de la estación fija Aire Ciudadano. Las mediciones se registraron en intervalos de cinco minutos durante una hora y media, exactamente en las horas de 9:30 AM a las 11:00 AM.

Los resultados obtenidos de los niveles de concentración de cada variable ambiental en la reserva natural, se muestra a continuación en la figura 42. Además, en el [anexo 9](#), se muestran las evidencias de las medidas tomadas en diferentes intervalos de tiempo.



Figura 42 . Visualización grafica de las mediciones realizadas en Ecoparque Sabana. Fuente: Elaboración Propia.

Estas mediciones fueron comparadas en tiempo presente con la estación fija de Aire Ciudadano y la estación certificada de la CAR con la que cuenta la reserva natural, logrando evidenciar una tolerancia de precisión alta con los sensores de la estación certificado, ya que, la estación cuenta con sensores con mayor precisión y certificados para realizar las mediciones de calidad del aire. El índice de calidad del aire, teniendo en cuenta la medición de material particulado de la estación fija de Aire ciudadano se muestran en la figura 43.

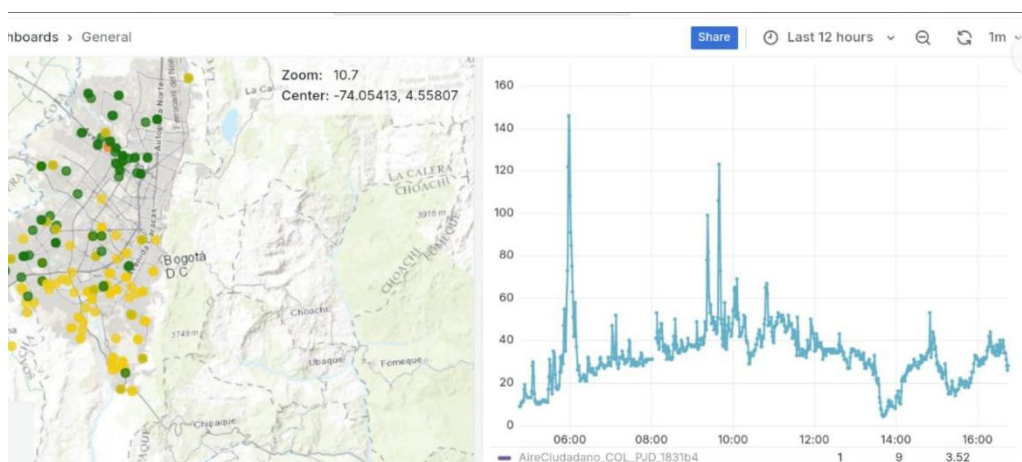


Figura 43. Mediciones en tiempo presente de la estación fija Aire Ciudadano. Fuente: Portal Aire Ciudadano.

Además, en la figura 44, se muestran las variables medidas en tiempo presente por la estación certificada de Ecoparque, logrando evidenciar las diferencias y tolerancias de las mediciones de la estación de monitoreo del presente proyecto

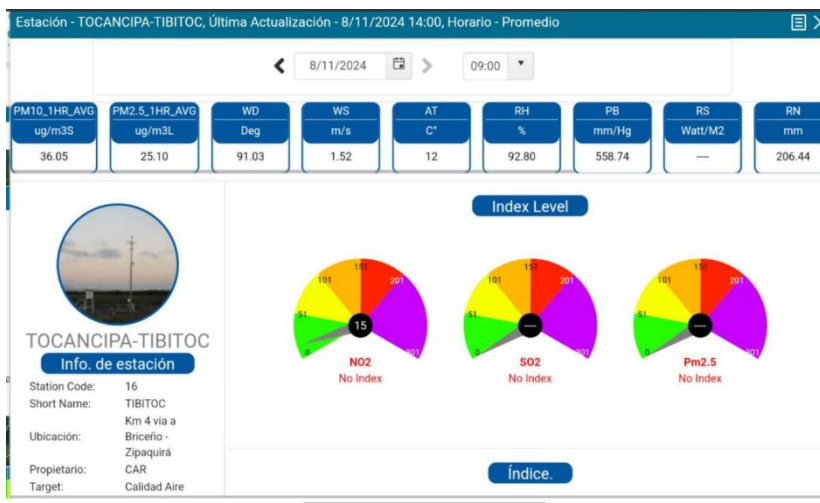


Figura 44. Mediciones en tiempo presente por parte de la estación de la CAR en el parque Jaime Duque.
Fuente: <https://sitedlia.car.gov.co/>

A continuación, se muestran los índices altos de las variables medidas en Ecoparque sabana, ya que, Grafana permite escoger el lapso de tiempo, como también, el poder observar el índice con mayor nivel relacionando el tiempo en el que se obtuvo como se muestra en la figura 45.



Figura 45. Los índices altos obtenidos en las mediciones en Ecoparque. Fuente: Elaboración Propia.

Estos resultados se analizaron para posteriormente realizar una comparación con las mediciones realizadas en el área urbana de la Universidad Pedagógica Nacional, evaluando así, las diferencias entre un área natural y una zona urbana, teniendo en cuenta la misma hora y altura de la posición de la estación de monitoreo realizada en ecoparque sabana. Los resultados realizados en la Universidad Pedagógica Nacional en la sede de la calle 72 No. 11-86, Bogotá, Colombia, como se muestra en la siguiente figura 46. Además, en el [anexo 10](#), se muestran las evidencias de las mediciones realizadas en la universidad Pedagógica Nacional.



Figura 46. Mediciones realizadas en la Universidad Pedagógica Nacional de la sede calle 72. Fuente: Elaboración Propia.

Posteriormente, se realiza el mismo filtro de lapso y de índices altos de cada medida para realizar las comparaciones, dicha relación se muestra en la figura 47, disponiendo de las opciones con las que cuenta la interfaz Grafana para la interpretación y análisis de los datos obtenidos, con el propósito de poder realizar comparaciones de los parámetros máximos que lograron obtener mediante las medidas que se realizaron en dos zonas diferentes.



Figura 47. Filtro de las mediciones realizadas en la Universidad Pedagógica Nacional de la sede calle 72. Fuente: Elaboración Propia.

Ahora bien, para el análisis, la interpretación y las conclusiones de las medidas realizadas en las dos zonas presentadas anteriormente, se realiza una tabla de comparación de las variables medidas por medio del sistema de monitoreo. En Ecoparque Sabana, las mediciones de la calidad del aire que se realizaron durante hora y media, en donde se obtuvieron los siguientes valores, teniendo en cuenta cada parámetro monitoreado, en la tabla 12 se muestran los índices altos obtenidos.

Tabla 12.

Valores máximos obtenidos de cada parámetro monitoreado en Ecoparque Sabana.

| Parámetro | Valor máximo obtenido | Unidad |
|-----------------|-----------------------|--------|
| Temperatura | 15.7 | °C |
| Humedad | 82.5 | % |
| COV | 5000 | ppm |
| PM2.5 | 150 | µg/m |
| CO | 1700 | ppm |
| CO ₂ | 3100 | ppm |

Nota: Esta tabla muestra los diferentes valores obtenidos teniendo en cuenta cada parámetro ambiental monitoreado en Ecoparque Sabana. Fuente: Elaboración Propia.

Tabla 13.

Valores máximos obtenidos de cada parámetro monitoreado.

| Parámetro | Valor máximo obtenido | Unidad |
|------------------|------------------------------|---------------|
| Temperatura | 17.8 | °C |
| Humedad | 300 | % |
| COV | 12500 | ppm |
| PM2.5 | 3000 | µg/m |
| CO | 500 | ppm |
| CO ₂ | 3500 | ppm |

Nota: Esta tabla muestra los diferentes valores obtenidos teniendo en cuenta cada parámetro ambiental monitoreado en la universidad Pedagógica Nacional. Fuente: Elaboración Propia

Ahora bien, la comparación de los resultados obtenidos entre Ecoparque Sabana y Universidad Pedagógica Nacional, se realizaron mediante la configuración de los paneles y unidades de medidas en Grafana, con el propósito de facilitar la comparación de los parámetros críticos de las variables medidas. Donde se destaca, que los valores de PM2.5 en la universidad Pedagógica Nacional muestran una alta concentración de material particulado en comparación con la reserva natural, donde posiblemente se debe al tráfico y la actividad humana en áreas urbanas, lo cual podría representar riesgos para la salud humana en el sector, de acuerdo a los estándares de la EPA y la resolución 2254 del año 2017 que proporciona el Ministerio de Ambiente. Por otro lado, los niveles de dióxido de carbono, y compuestos orgánicos volátiles, en el área urbana, también se pueden visualizar que los valores son notablemente altos en un área urbana, lo cual, se encuentra relacionado con la actividad de combustión de vehículos en áreas urbanas, como también, la presencia de productos industriales y materiales de construcción. Según la EPA y el ICA es importante subrayar la importancia de controlar estos compuestos debido a las enfermedades respiratorias que se pueden contraer por medio del impacto ambiental de dichos compuestos.

Por otra parte, en las mediciones obtenidas de los niveles de concentración de monóxido de carbono (CO), se logró visualizar, que se tuvo mayor concentración en la reserva natural de Ecoparque Sabana, donde posiblemente se puede presentar por medio de la descomposición natural de materia orgánica, menor dispersión de gases en el terreno, como también, la concentración de gases que generan el tráfico vehicular y las empresas industriales que se encuentran alrededor de Ecoparque Sabana.

Para dar cumplimiento al tercer objetivo específico del presente proyecto, se realiza una guía didáctica, con el propósito de tener una orientación base para la implementación de la estación de monitoreo, describiendo desde los componentes que complementan la estación de monitoreo, hasta la configuración e implementación del sistema. En el anexo [] se puede visualizar la guía didáctica por medio de un hipervínculo que redirecciona al drive donde se encuentra el archivo de la guía. Por otra parte, para el diseño de la guía, se integraron los estándares básicos de competencias, teniendo en cuenta las habilidades que desarrolla un estudiante desde grado noveno. Por otro lado, también se implementaron algunas metas de aprendizaje del enfoque STEM, con el propósito de no solo adquirir conocimientos en electrónica y programación, sino también, el desarrollo de habilidades en ciencia, tecnología, ingeniería y matemáticas, teniendo una relación entre teoría y práctica sobre la importancia de monitorear la calidad del aire.

Finalmente, con el cumplimiento de cada objetivo específico se ha logrado realizar el objetivo general del presente proyecto, teniendo así, la actualización de la estación de monitoreo, por medio de la integración de nuevos sensores, y protocolos de comunicación MQTT y LoRa para la visualización en tiempo presente a través de la interfaz de Grafana, por último, se realizó la creación de una guía didáctica para facilitar su implementación. Esto demuestra que la

implementación del sistema de monitoreo de la calidad del aire es una herramienta accesible y práctica, donde se promueve la ciencia ciudadana por medio de concienciación ambiental y la educación en tecnologías que promueven el monitoreo y control de variables.

CONCLUSIONES.

La estación de monitoreo de la calidad del aire del portal Aire Ciudadano, permitió la integración e implementación de los sensores que miden dióxido de carbono (CO_2) monóxido de carbono (CO) y los compuestos volátiles orgánicos (COV). Teniendo la capacidad de obtener datos de diferentes gases en tiempo real, permitiendo monitorear y visualizar más información sobre la calidad del aire.

Se logro evidenciar que se puede realizar una red de múltiples estaciones de monitoreo con diferentes sensores (IoT), puesto que tecnologías como MQTT y LoRA han demostrado ser eficiente para realizar dicha conexión, posibilitando la transmisión de datos a largas distancia y la implementación de sistemas de monitoreo en entornos exteriores o interiores, teniendo la posibilidad de replicar la estación de monitoreo en diferentes áreas geográficas.

Para la actualización de los paneles gráficos realizado en la interfaz Grafana, es fundamental realizar una configuración previa del entorno gráfico, ya que, es necesario elegir correctamente la unidad de medida, las variables de mediciones y la frecuencia de medición para obtener finalmente un panel grafico que permita realizar una mejor interpretación y análisis de los datos en tiempo presente.

La visualización grafica se realizó de manera correcta en Grafana, puesto que permite realizar diferentes configuraciones para las visualizaciones de las variables ambientales en

tiempo presente, teniendo la posibilidad de tomar medidas preventivas a los ciudadanos y a las entidades locales, como también, fomentar la concientización por el cuidado del medio ambiente y la importancia del monitoreo y control de la calidad del aire, con el objetivo de reducir el impacto negativo que genera la contaminación ambiental.

Este proyecto de grado no solo cumplió los objetivos de investigación establecidos sobre la actualización, implementación y medición de la calidad del aire, sino que también fue reconocido en el ámbito interuniversitario, puesto que fue presentado en el concurso Tesis In-Bite de la Universidad Pedagógica Nacional, bajo la modalidad del formato In-Bite 12X25, ocupando el segundo puesto entre los participantes de diferentes universidad y carreras profesionales. Este logro refleja no solo la relevancia del proyecto como contribución a la ciencia ciudadana y al monitoreo ambiental, sino también, la relevancia del tema de investigación y la claridad en su comunicación.

RECOMENDACIONES.

Es importante que, para futuros trabajos, se recomienda agregar un dispositivo que permita medir la dirección y velocidad del viento, ya que complementa la estación de monitoreo, puesto que, no solo se obtendrán datos de la contaminación ambiental, sino también, el área y dirección de propagación de las emisiones, asistiendo la planificación de medidas preventivas.

Se recomienda realizar programas educativos que relacione las medidas de la estación de monitoreo y la comunidad, con el propósito de sensibilizar a la población con observaciones argumentadas, resaltando la importancia del monitoreo de la calidad del aire, como también, la realización de materiales educativos que asistan el análisis e interpretación de los datos que se pueden observar a través de Grafana.

Para futuros trabajos se tiene la opción de actualizar el hardware y software de la estación de monitoreo, integrando sensores con mayor precisión en sus medidas, con el objetivo de obtener datos con menor porcentaje de error, puesto que, las actividades humanas y los factores ambientales que se pueden presentar en diversos lugares, ocasionando variaciones de la calidad del aire con mayor frecuencia.

Se recomienda que, para futuras implementaciones, es relevante utilizar notificaciones de alerta en el sistema, con el propósito de informar cuando se detecten niveles de contaminación perjudiciales para la salud humana, impulsando a la protección contra enfermedades cardiovasculares y respiratorias, como también, previniendo altos niveles de contaminación.

REFERENCIAS.

- [1] @grafana/Create-Plugin. (3 de mayo de 2024). Npm.
<https://www.npmjs.com/package/@grafana/create-plugin>.
- [2] ¿Qué es IoT y cómo funciona? (n.d.). SAP. <https://www.sap.com/spain/products/artificial-intelligence/what-is-iot.html>
- [3] ¿Qué es UML? (n.d.). <http://profesores.fi-b.unam.mx/carlos/aydoo/uml.html>
- [4] Aire Ciudadano – Ciencia ciudadana por la Calidad del Aire. (17 de septiembre de 2022).
<https://aireciudadano.com/>.
- [5] *Build a Data Source Plugin | Grafana Plugin Tools*.
<https://grafana.com/developers/plugin-tools/tutorials/build-a-data-source-plugin>.
- [6] Código IoT. (n.d.). Introducción a Node-Red en Internet de las Cosas.
<https://www.codigoiot.com/curso/introduccion-a-node-red/>
- [7] Data Logging with MQTT, Node-RED, InfluxDB and Grafana. (n.d.).
<https://docs.arduino.cc/tutorials/portenta-x8/datalogging-iot/>.
- [8] Data Source Management | Grafana Documentation. (n.d.). Grafana Labs.
<https://grafana.com/docs/grafana/latest/administration/data-source-management/>
- [9] Diagrama de clases: Qué es, cómo hacerlo y ejemplos | Miro. (n.d.). Miro.
<https://miro.com/es/diagrama/que-es-diagrama-clases-uml/>.
- [10] E. M. Q. Technologies. (13 de junio de 2024). A Comprehensive Comparison of Open Source MQTT Brokers 2023. Medium. <https://emqx.medium.com/a-comprehensive-comparison-of-open-source-mqtt-brokers-2023-e70257cc5b75>
- [11] Eclipse Mosquitto. (27 de octubre de 2021). Download.
<https://mosquitto.org/download/>
- [12] Especificaciones del módulo ESP32. (2021, julio).
<https://vasanza.blogspot.com/2021/07/especificaciones-del-modulo-esp32.html>.
- [13] Grafana: The open observability platform. (n.d.). Grafana Labs.
<https://grafana.com/>
- [14] Guías prácticas. (n.d.). Repositorio UCC.
<https://repository.ucc.edu.co/server/api/core/bitstreams/a7ea175f-de13-49fd-afba-c021baf7a6d0/content>

- [15] Holguín Pedraza, J. A., & Andrade Forero, M. S. (2021, diciembre). Plataforma en la nube para la visualización y análisis de datos provenientes de dispositivos para el monitoreo de la polución del aire. <http://repository.javeriana.edu.co/handle/10554/6207>
- [16] M. H. Roldán Martínez. (julio de 2022). Diseño de sistema monitorización de calidad del aire mediante tecnologías y sensores de IoT y domótica. Design of an air quality monitoring system using IoT and home automation technologies and sensors. <http://repositorio.upct.es/handle/10317/11636>
- [17] Mejores Trabajos de Grado Facultad de Ingeniería - Javeriana I-2022. (28 de marzo de 2022). PUJaveriana - Issuu. https://issuu.com/pujaveriana/docs/mejores_trabajos_de_grado_ingenier_a_javeriana_i-2
- [18] Ministerio de Ambiente y Desarrollo Sostenible. (2017). *Resolución 2254 de 2017, por la cual se establecen normas de calidad del aire y límites de contaminación en Colombia*. Bogotá, Colombia: Ministerio de Ambiente y Desarrollo Sostenible.
- [19] Ministerio de Ambiente y Desarrollo Sostenible. (2022). *Índice de Calidad del Aire (ICA): Normas y procedimientos en Colombia*. Bogotá, Colombia: Ministerio de Ambiente y Desarrollo Sostenible.
- [20] Nur Hanani Ahmad Puaad. (2020). Beginner for Node-RED, InfluxDB & Grafana. YouTube. <https://www.youtube.com/watch?v=mvIuB1N0U7I>
- [21] Peñaranda Arcentales, B. G. (2024). Desarrollo de un sistema IoT para el monitoreo remoto de variables en un huerto agrícola. <http://dspace.ups.edu.ec/handle/123456789/27496>
- [22] Probando más nodos en Nodered – Prometec. (n.d.). <https://www.prometec.net/probando-mas-nodos-en-nodered/>
- [23] Qué es Grafana y primeros pasos | OpenWebinars. (n.d.). OpenWebinars.net. <https://openwebinars.net/blog/que-es-grafana-y-primeros-pasos/>
- [24] Rodríguez Mosquera, S. A. (noviembre de 2023). Automatización de Centro de Datos: Infraestructura de Centros de Datos Virtuales. <http://bibdigital.epn.edu.ec/handle/15000/25020>
- [25] ThinkIOT. (2023). Smart Servo Control: Node-RED Dashboard & ESP32 MQTT Integration. YouTube. <https://www.youtube.com/watch?v=veKMM0pgEHY>
- [26] U.S. Environmental Protection Agency. (2023). *Air Quality Index (AQI): A Guide to Air Quality and Your Health*. Recuperado de <https://www.epa.gov/air-quality-index>

ANEXOS

Anexo 1. Topics MQTT implementados en código de programación de la etapa receptora.

| Dispositivos de Medición | Topic |
|-------------------------------------------------------|-------------------|
| Sensor SHT31D (Temperatura) | sht31/temperatura |
| Sensor SHT31D (Humedad) | sht31/humedad |
| Sensor MQ7 (monóxido de Carbono) | mq7/CO |
| Sensor MQ135 (Dióxido de Carbono) | mq135/CO2 |
| Sensor PMS7003 (Material Particulado 2.5 micrómetros) | pms3003/pm25 |
| Sensor CCS811 (Compuestos Orgánicos Volátiles) | ccs811/cov |

Tabla 14. Topics MQTT. Fuente: Elaboración Propia.

Anexo 2. Esquema de conexión y sus etiquetas de identificación de la etapa receptora.

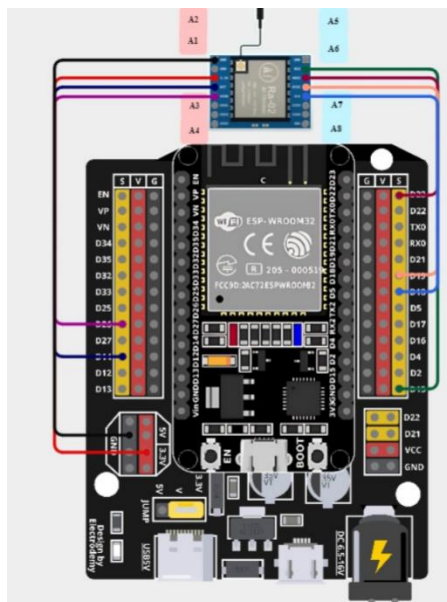


Figura 45. Esquemático de conexión etapa receptora. Fuente: elaboración propia.

Tabla 15. Etiquetas de conexiones etapa receptora.

| Etiqueta | Pin de Conexión en la ESP32 | Componente (Pin del componente) |
|----------|-----------------------------|---------------------------------|
| A1 | VCC | LoRa(5v) |
| A2 | GND | LoRa(GND) |

| | | |
|----|-----|-------------|
| A3 | D14 | LoRa (RST) |
| A4 | D26 | LoRa (DIO0) |
| A5 | D15 | LoRa (NSS) |
| A6 | D23 | LoRa (MOSI) |
| A7 | D19 | LoRa (MISO) |
| A8 | D18 | LoRa (SCK) |

Tabla 15. Etiquetas de conexiones de la etapa receptora. Fuente: Elaboración Propia.

Anexo 3. Esquemático de conexión y etiquetas de identificación de la Etapa Transmisora:

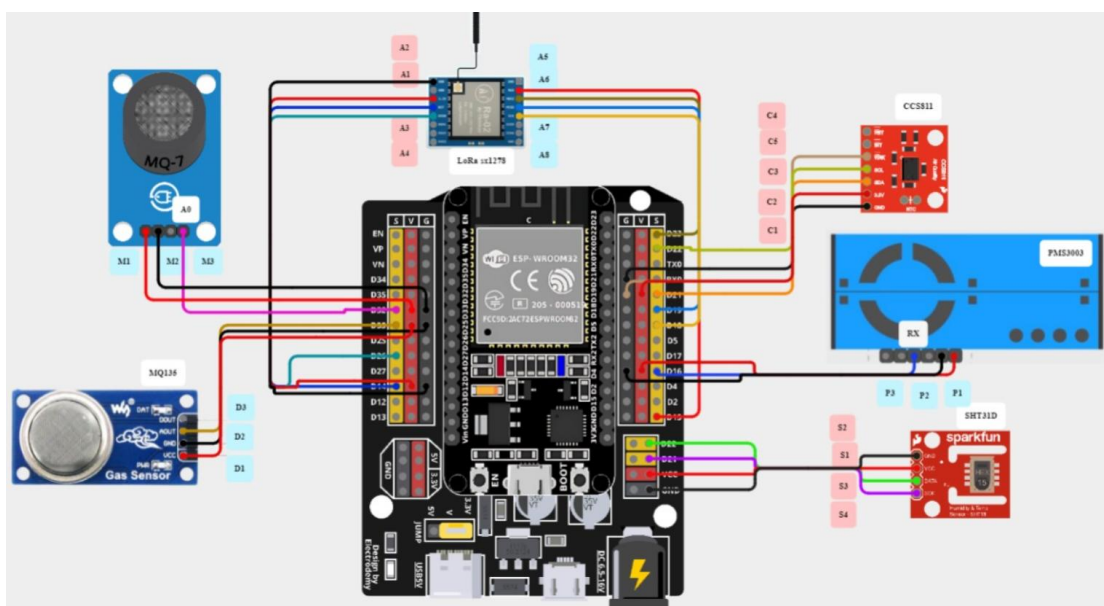


Figura 46. Esquemático de conexión de etapa Transmisora. Fuente: elaboración propia.

Tabla 16. Etiquetas de conexión de componentes en la etapa Transmisora.


| Etiqueta | Pin de Conexión en la ESP32 | Componente (Pin del componente) |
|----------|-----------------------------|---------------------------------|
| A1 | VCC | LoRa(5v) |
| A2 | GND | LoRa(GND) |
| A3 | D14 | LoRa (RST) |
| A4 | D26 | LoRa (DIO0) |
| A5 | D15 | LoRa (NSS) |
| A6 | D23 | LoRa (MOSI) |
| A7 | D19 | LoRa (MISO) |
| A8 | D18 | LoRa (SCK) |
| C1 | GND | CCS811(GND) |
| C2 | VCC | CCS811(3.3V) |
| C3 | D21 | CCS811(SDA) |


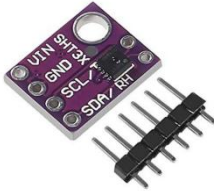


| | | |
|----|-----|-------------|
| C4 | GND | CCS811(WAK) |
| C5 | D22 | CCS811(SCL) |
| P1 | VCC | PMS(VCC) |
| P2 | GND | PMS(GND) |
| P3 | D16 | PMS(TX) |
| S1 | VCC | Sht(VCC) |
| S2 | GND | Sht(GND) |
| S3 | D22 | Sht(DATA) |
| S4 | D21 | Sht(SCK) |
| D1 | VCC | MQ135(VCC) |
| D2 | GND | MQ135(GND) |
| D3 | D33 | MQ135(AOUT) |
| M1 | VCC | MQ7(VCC) |
| M2 | GND | MQ7(GND) |
| M3 | D32 | MQ7(A0) |


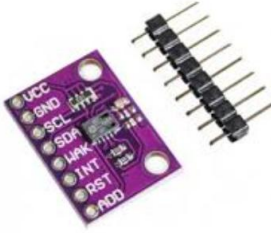
Tabla 16. Etiquetas de conexión de componentes en la etapa Transmisora. Fuente: Elaboración Propia.

Anexo 4. Tabla de componentes con sus especificaciones, costo y recomendaciones de uso.

Tabla 17. Componentes electrónicos seleccionados. Fuente: Elaboración Propia.

| Componente | Especificaciones | Costo | Recomendaciones |
|---------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Tarjeta Controladora ESP832:  | El módulo ESP32 (ESP-WROOM-32) es una poderosa plataforma para proyectos y aplicaciones IoT. Con un microprocesador Tensilica Xtensa de 32 bits LX6 que alcanza hasta 240 MHz, ofrece conectividad Wi-Fi y Bluetooth integrada. | \$32.000 | La tarjeta ESP32 ha sido elegida para este proyecto de investigación debido a su historial en aplicaciones IoT y su capacidad para conectarse con otros dispositivos utilizando diversos protocolos de comunicación, incluido MQTT para mensajes y suscripciones. Se programa fácilmente a través del IDE Arduino utilizando el lenguaje de programación C++, lo que la hace ideal para proyectos que requieren conexión Wi-Fi y acceso a servidores web como Grafana. |
| Módulos LoRa sx1278: | integración: Los módulos LoRa SX1278 deben ser conectados a las tarjetas controladoras ESP32 a través de los pines adecuados y configurarse para la | \$30.000 | El módulo LoRa SX1278 se seleccionó por su capacidad de comunicación a larga distancia y bajo consumo de energía, ideal para el monitoreo remoto de la calidad del aire. Su |

| | | | |
|-----------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  | <p>comunicación LoRa. Se requerirá código para establecer la comunicación entre los módulos LoRa, así como para enviar y recibir datos a través de la red LoRa. Esto puede implicar la gestión de la potencia de transmisión y la configuración de los canales de comunicación.</p> <p>Funcionamiento: Durante el funcionamiento, los módulos LoRa deben estar configurados para enviar regular y confiablemente datos de calidad del aire a la estación receptora.</p> | | <p>tecnología LoRa permite una transmisión eficiente de datos a distancias considerables, garantizando cobertura para un sistema de monitoreo ambiental distribuido. Su interfaz SPI facilita la integración con la tarjeta controladora ESP32, asegurando una interoperabilidad efectiva en el sistema. La compatibilidad con el protocolo MQTT permite una transmisión eficiente de datos a través de Internet para su análisis y visualización.</p> |
| <p>Sensor SHT31D:</p>  | <p>El sensor SHT31D se conecta a la tarjeta ESP32 para medir la temperatura y humedad del aire. Requiere programación para obtener lecturas precisas y se debe ubicar en un lugar representativo.</p> | <p>\$32.000</p> | <p>El sensor SHT31D es altamente preciso y estable en la medición de temperatura y humedad. Es ideal para monitoreo ambiental y se integra fácilmente con la tarjeta ESP32. Utiliza la interfaz I2C y el protocolo MQTT para una transmisión eficiente de datos.</p> |
| <p>Señor PM2.5 PMS7003:</p>  | <p>El sensor PMS7003 se conecta a la tarjeta Arduino Wemos D1 a través de los pines RX y TX. Requiere programación personalizada para configurar la comunicación y interpretar los datos. Se debe colocar en una posición adecuada y calibrar para mediciones precisas.</p> | <p>\$130.000</p> | <p>El sensor Plantower PMS7003 es preciso y confiable en la medición de partículas finas (PM2.5) en el aire. Es ideal para monitorear la calidad del aire en interiores y exteriores. Se integra fácilmente con la tarjeta ESP8266 a través de UART y utiliza MQTT para la transmisión eficiente de datos.</p> |
| <p>Sensor MQ7 (CO):</p>  | <p>El sensor MQ7 detecta la concentración de monóxido de carbono (CO) en el aire. Tiene un rango de detección de 20 a 2000 ppm y responde rápidamente a cambios en la concentración. Proporciona una salida analógica que varía según la concentración de CO. Es adecuado para monitorear la calidad del aire y detectar fugas de gas.</p> | <p>\$12.000</p> | <p>El sensor MQ7 es preciso y confiable en la medición de monóxido de carbono (CO) en el aire. Se utiliza para monitoreo de la calidad del aire en diferentes entornos. Se integra fácilmente con la tarjeta ESP32 a través de su interfaz analógica y utiliza MQTT para la transmisión eficiente de datos.</p> |

| | | | |
|------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Sensor MQ135 (CO₂)</p>  | <p>El sensor MQ135 es un sensor de calidad del aire que detecta una amplia gama de gases, incluyendo amoníaco (NH₃), benceno (C₆H₆), óxidos de nitrógeno (NO_x), dióxido de azufre (SO₂) y compuestos orgánicos volátiles (COV). Tiene un rango de detección configurable y proporciona una salida analógica proporcional a la concentración de gases detectados. Es sensible y selectivo a diferentes tipos de gases. Funciona mediante un elemento sensible basado en óxido metálico semiconductor. Es adecuado para aplicaciones de monitoreo de la calidad del aire en interiores y exteriores, así como para la detección de gases tóxicos.</p> | <p>\$16.000</p> | <p>El sensor MQ135 ha sido seleccionado debido a su capacidad para medir de manera precisa y confiable una amplia gama de gases en el aire. Su diseño y rendimiento avanzados lo hacen adecuado para aplicaciones de monitoreo de la calidad del aire en diversos entornos. Además, su salida analógica permite una fácil integración con la tarjeta controladora ESP8266, asegurando una interoperabilidad efectiva en el sistema. La ventaja de utilizar el protocolo MQTT permite una transmisión eficiente de datos para su posterior análisis y visualización. El sensor MQ135 es una elección ideal para el monitoreo de la calidad del aire y la detección de gases tóxicos.</p> |
| <p>Sensor CCS811 (COV)</p>  | <p>El sensor CCS811 detecta una amplia gama de compuestos orgánicos volátiles (COV). Tiene un rango de detección configurable y proporciona una salida digital para indicar la presencia o ausencia de COV. Es altamente sensible y selectivo a diferentes tipos de gases. Tiene un tiempo de respuesta rápido y consume poca energía. Es ideal para monitoreo de la calidad del aire, control de contaminación y detección de fugas químicas.</p> | <p>\$40.000</p> | <p>El sensor CCS811 es preciso y confiable en la medición de compuestos orgánicos volátiles (COV) en el aire. Se utiliza para monitorear la calidad del aire en entornos interiores y exteriores. Se integra fácilmente con la tarjeta ESP32 a través de la interfaz UART y utiliza MQTT para la transmisión eficiente de datos.</p> |

Anexo 5. Procesos de calibración de los sensores conectados en la etapa transmisora.

Calibración de los Sensores MQ

El sensor utilizado para la medición de Dióxido de carbono fue el MQ135, que necesita de un proceso de calibración para obtener medidas coherentes y ajustadas a la recepción y entorno ambiental a la que se encuentre ubicado la estación de monitoreo. En primera instancia, el sensor MQ135 necesita estar encendido mediante un lapso de 24 horas, ya que necesita tener un proceso de calentamiento para dar inicio al proceso de calibración. Por otro lado, se realiza la configuración y conversión en el código de programación de la etapa receptora de la estación de monitoreo. Posteriormente, para verificar el funcionamiento de las mediciones del sensor, se procede a realizar la curva de calibración del sensor, donde iremos variando la resistencia del módulo y tomando en cuenta las mediciones obtenidas. Por otro lado, es importante tener en cuenta la gráfica de caracterización del sensor que nos proporciona su propio DataSheet, como se muestra en la figura 47.

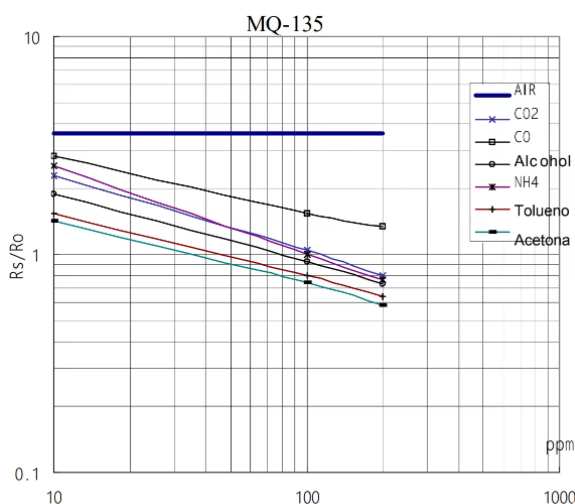


Figura 47. Curva típica de sensibilidad del sensor MQ135. Fuente: DataSheet Winsen Electronics Technology Co., Ltd.

Para finalizar la calibración del sensor primero necesitamos medir R_0 en aire limpio, ajustando R_0 exactamente en 10.000 Ohmios, posteriormente se realiza una medición de la resistencia R_s , en diferentes cantidades de concentraciones de CO_2 , utilizando botellas que generan un cambio de partículas por millón (ppm), como también, midiendo con el multímetro la salida del pin análogo del sensor, obteniendo resultados en intervalos de tiempo diferentes, por ejemplo:

- Para 115 ppm se obtiene un R_s de $20\text{ K}\Omega$
- Para 250 ppm se obtiene un R_s de $14\text{ K}\Omega$
- Para 325 ppm se obtiene un R_s de $9\text{ K}\Omega$

El siguiente paso, es solucionar la división entre R_s y R_0 para hallar la relación entre las resistencias y así, proceder a realizar la gráfica que genera la curva de calibración, teniendo en el eje x la concentración que se tienen por partículas por millón y en el eje y, asignar los diferentes valores que se obtuvieron mediante la relación de R_s y R_0 , por ejemplo:

- Para 115 ppm: $R_s/R_0 = 20\text{ K}\Omega / 10\text{ K}\Omega = 2$
- Para 250 ppm: $R_s/R_0 = 14\text{ K}\Omega / 10\text{ K}\Omega = 1.4$
- Para 325 ppm: $R_s/R_0 = 9\text{ K}\Omega / 10\text{ K}\Omega = 0.9$

Obteniendo la siguiente curva realizada en Excel mostrada en la figura 48, teniendo en cuenta los intervalos de R_s obteniendo mediante la variación de diferentes ambientes, logrando tener una gráfica que nos permite interpretar y analizar el comportamiento de medición del sensor, puesto que será una gráfica guía para realizar la configuración en el software y hardware del sistema.

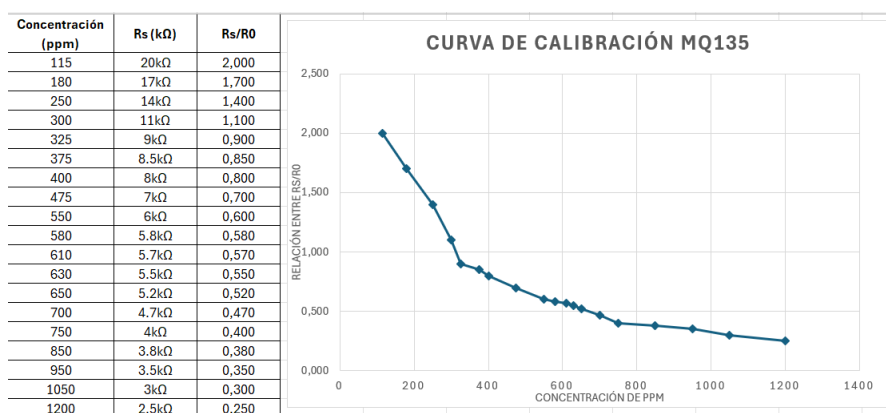


Figura 48 . Curva de calibración del sensor MQ135. Fuente: Elaboración Propia.

Ahora bien, la curva mostrada en la figura 48 se tiene en cuenta la relación en R_s/R_0 y la concentración de ppm en diferentes ambientes, para llevar a cabo una medición de R_s en tiempo real, con el objetivo de estimar la concentración que existe de dióxido de carbono. Para la medición en tiempo real, se realiza una lectura de salida del voltaje que proporciona el sensor, con la intención de calcular la resistencia R_s en tiempo real, utilizando la siguiente fórmula para obtener R_s en tiempo real:

$$R_s = \left(\frac{V_{CC} - V_{out}}{V_{out}} \right) * R_0 \quad (\text{Ecuación 1})$$

Donde VCC es el voltaje de alimentación, en este caso son aproximadamente 5 voltios , Por otro lado, Vout es la medida que se obtiene mediante la salida del pin analógico del sensor y el R0 es la resistencia que se implementa en el código de programación en la placa ESP32, asignando el valor de 10 KΩ. El siguiente paso, después de obtener el valor de Rs en tiempo presente, se calcula la relación Rs/R0, teniendo en cuenta que R0 es la medida que se obtuvo sin ninguna afectación en el ambiente. Con base en el valor de la relación obtenida por medio de la curva, se utiliza el grafico de calibración para obtener medidas coherentes y estimar la concentración de gases aplicando la ecuación 1.

$$R_s = \left(\frac{V_{CC} - V_{out}}{V_{out}} \right) * R_0 \quad (\text{Ecuación 1})$$

$$R_s = \left(\frac{4.92 - 1.2}{1.2} \right) * 10.000 = 30 \text{ K}\Omega \quad (\text{Ecuación 2})$$

Teniendo en cuenta que acaremos datos que nos proporciona la curva de la figura [] elaborada para el sensor MQ135, se realiza el siguiente procedimiento:

- Punto inicial en X=115 , Y=2.0 Rs/R0
- Punto final en X= 1200 , Y=0.25 Rs/R0
- X referencia = $\log_{10}(115) = 2.06$
- Y referencia = $\log_{10}(2.0) = 0.30$

$$C_{uva} = \frac{\log_{10}(0.25) - \log_{10}(2.0)}{\log_{10}(1200) - \log_{10}(115)} = -0.8867 \quad (\text{Ecuación 3})$$

Posteriormente, los datos obtenidos, son los parámetros de calibración a ingresar en el programa de la etapa receptora, teniendo en cuenta que el sensor MQ135 tiene un Rs de 30KΩ teniendo los siguientes datos de referencia que se muestra en la tabla 16 , tomando como referencia la curva realizada en la figura 48.

Tabla 18.

Datos de referencia para el proceso de calibración del sensor MQ135. Fuente: Elaboración Propia.

| Dióxido de Carbono | Eje X | Eje Y | Curva |
|---------------------------|--------------|--------------|--------------|
| MQ 135 | 2.06 | 0.30 | -0.886 |

En último lugar, se realizan el siguiente procedimiento matemático para encontrar la concentración de gas (CO₂) por partículas por millón (ppm). Como se muestra en ecuación 4.

$$Gas\ ppm = 10^{\left(\frac{Relación-GasEjeY}{GasCurva}\right)} + GasEjeX \quad (Ecuación\ 4)$$

El primer calculo numérico es realizar una división para encontrar la relación Rs/R0 teniendo en cuenta que ya obtuvieron los datos anteriormente mediante la ecuación 1.

$$Relación = \frac{Rs}{R0} = \frac{30\ K\Omega}{10\ K\Omega} = 3 \quad (Ecuación\ 5)$$

El primer calculo numérico que se realiza es la resta del algoritmo natural de la relación obtenida anteriormente en la ecuación 4, menos la contante encontrada del eje Y en este caso 0.30.

$$\ln(3) - 0.30 = 0.7986 \quad (Ecuación\ 6)$$

El siguiente paso es realizar la división entre los resultados obtenidos en la ecuación 5 entre el resultado de la curvatura obtenida para el dióxido de carbono.

$$\frac{0.7986}{-0.886} = -0.9013 \quad (Ecuación\ 7)$$

Para completar los datos que hacen para para realizar el procedimiento matemático de la ecuación 4, se calcula el exponente a la expresión, sumándole al resultado de la ecuación (7) la constante del eje X.

$$-0.9013 - 2.06 = -2.9613 \quad (Ecuación\ 8)$$

Por último, se calcula la concentración de PMM del gas (O₂, utilizando la ecuación de la 4.

$$\text{Gas ppm} = 10^{\left(\frac{3-0.30}{-0.8867}\right)} + 2.06 = 2.0608 \text{ PPM} \quad (\text{Ecuación 9})$$

En la figura 49 se puede evidenciar la implementación en el código de programación de la etapa transmisora de la estación de monitoreo.

```

54 // Función para obtener la concentración de CO2 utilizando el MQ135
55 void dioxidodecarbono() {
56     rawValueMQ7 = analogRead(MQ7_PIN);
57     rawValueMQ135 = analogRead(MQ135_PIN);
58     // Convertir la salida analógica en resistencia Rs
59     float Vout = (rawValueMQ135 / 4095.0) * VCC; // Calcular el voltaje de salida del MQ135
60     float Rs = ((VCC - Vout) / Vout) * R0; // Calcular Rs en tiempo real
61     // Calcular la relación Rs/R0
62     float ratio = Rs / R0;
63     // Usar la curva de calibración para estimar la concentración de CO2 en ppm
64     float co2_concentration = pow(10, ((log10(ratio) - log10(2.0)) / (-0.8867)) + log10(115));
65     ppmMQ7 = convertToPPM(rawValueMQ7); // Para el MQ7
66     Serial.print("Concentración de CO2 (MQ135): ");
67     Serial.println(co2_concentration);
68 }

```

Figura 49. Función para obtener CO₂ en el software de la etapa Transmisora del sistema. Fuente: Elaboración Propia.

Para realizar la calibración del sensor que nos permite obtener datos de monóxido de carbono (CO), se realiza el mismo procedimiento que se realizó anteriormente para el sensor MQ7. Iniciar su calentamiento del módulo por 24 horas mínimo, con la intención de obtener la curva que permite realizar la calibración y posteriormente iniciar con el procedimiento matemático, para la implementación algorítmica. En la figura [] se muestra el datasheet de sensor MQ7 para sacar la referencia de los ejes.

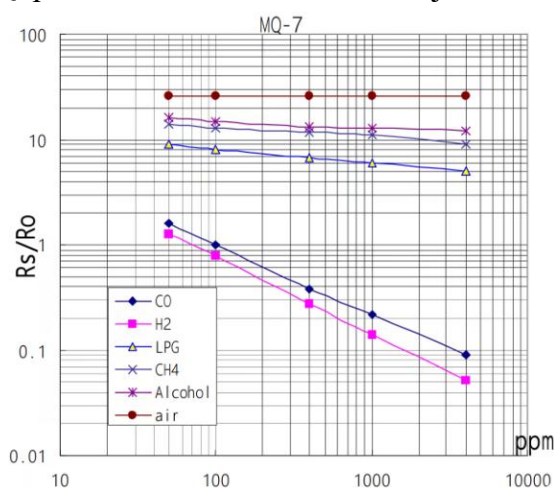


Figura . Curva típica de sensibilidad del sensor MQ7. Fuente: DataSheet Winsen Electronics Technology Co., Ltd.

Principalmente se realiza el cálculo matemático para hallar la resistencia al aire limpio R_s del sensor MQ7 utilizando la ecuación 1, cambiando los valores medidos para el presente sensor y manteniendo el mismo valor de R_0 .

$$R_s = \left(\frac{4.85 - 1.2}{1.2} \right) * 10.000 = 30.416 \Omega \quad (\text{Ecuación 1})$$

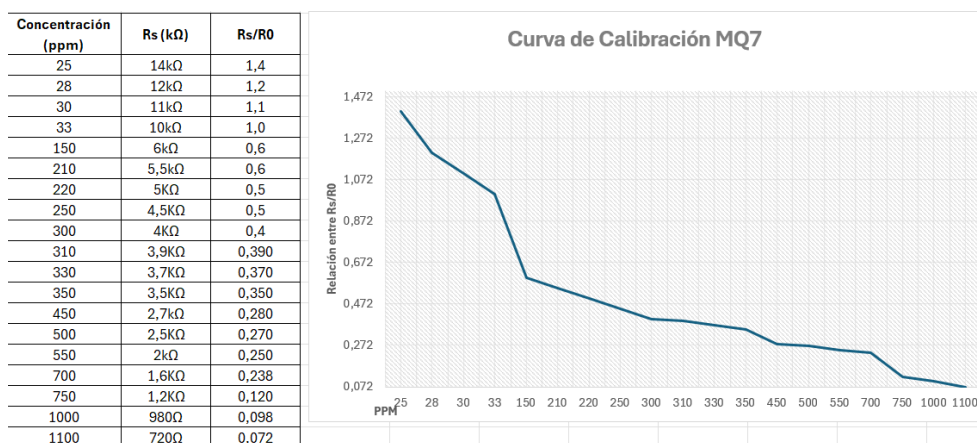


Figura 50. Curva de calibración del sensor MQ7. Fuente: Elaboración Propia.

Teniendo en cuenta que acaremos datos que nos proporciona la curva de la figura 50, elaborada para el sensor MQ135, se realiza el siguiente procedimiento:

- Punto inicial en $X=25$, $Y=1.35R_s/R_0$
- Punto final en $X=1100$, $Y=0.072 R_s/R_0$
- X referencia = $\log_{10}(25) = 1.397$
- Y referencia = $\log_{10}(1.35) = 0$

$$Cuva = \frac{\log_{10}(0.072) - \log_{10}(1.35)}{\log_{10}(1100) - \log_{10}(25)} = -0.774 \quad (\text{Ecuación 2})$$

| Monóxido de Carbono | Eje X | Eje Y | Curva |
|---------------------|-------|-------|--------|
| MQ 7 | 1.397 | 0.130 | -0.774 |

Tabla 19. Ejes de referencia para el proceso de calibración del sensor MQ7. Fuente: Elaboración Propia.

En último lugar, se realizan el siguiente procedimiento matemático para encontrar la concentración de gas (CO_2) por partículas por millón (ppm). Como se muestra en ecuación 4.

$$Gas\ ppm = 10^{\left(\frac{Relación-GasEjeY}{GasCurva}\right)} + GasEjeX \quad (Ecuación\ 3)$$

El primer calculo numérico es realizar una división para encontrar la relación Rs/R0 teniendo en cuenta que ya obtuvieron los datos anteriormente mediante la ecuación 1.

$$Relación = \frac{Rs}{R0} = \frac{30.416\ \Omega}{10\ K\Omega} = 0.0030 \quad (Ecuación\ 4)$$

El primer calculo numérico que se realiza es la resta del algoritmo natural de la relación obtenida anteriormente en la ecuación 4, menos la contante encontrada del eje Y en este caso 0.30.

$$\ln(0.0030) - 0.130 = -2.652 \quad (Ecuación\ 5)$$

El siguiente paso es realizar la división entre los resultados obtenidos en la ecuación 5 entre el resultado de la curvatura obtenida para el dióxido de carbono.

$$\frac{-2.652}{-0.774} = 3.426 \quad (Ecuación\ 6)$$

Para completar los datos que hacen para para realizar el procedimiento matemático de la ecuación 4, se calcula el exponente a la expresión, sumándole al resultado de la ecuación (6) la constante del eje X.

$$3.426 - 1.397 = 2.029 \quad (Ecuación\ 7)$$

Por último, se calcula la concentración de PMM del gas CO₂, utilizando la ecuación de la 4 del procedimiento de calibración del sensor MQ135.

$$Gas\ ppm = 10^{\left(\frac{0.0030-0.130}{-0.774}\right)} + 1.397 = 2.856\ PPM \quad (Ecuación\ 8)$$

En la figura 51, se puede evidenciar la implementación en el código de programación de la etapa transmisora de la estación de monitoreo.

```

75 // Función para obtener la concentración de CO utilizando el MQ7
76 void monoxidoDeCarbono() {
77   rawValueMQ7 = analogRead(MQ7_PIN);
78
79   // Convertir la salida analógica en resistencia Rs para el MQ7
80   float Vout_MQ7 = (rawValueMQ7 / 4095.0) * VCC; // Calcular el voltaje de salida del MQ7
81   float Rs_MQ7 = ((VCC - Vout_MQ7) / Vout_MQ7) * R0_MQ7; // Calcular Rs en tiempo real
82
83   // Calcular la relación Rs/R0 para MQ7
84   float ratio_MQ7 = Rs_MQ7 / R0_MQ7;
85
86   // Usar la curva de calibración para estimar la concentración de CO en ppm
87   float co_concentration = pow(10, ((log10(ratio_MQ7) - log10(2.0)) / (-0.32)) + log10(100));
88
89   Serial.print("Concentración de CO (MQ7): ");
90   Serial.println(co_concentration);
91
92   ppmMQ7 = co_concentration; // Guardar el valor calculado de CO para transmisión
93 }

```

Figura 51. Implementación del código de la etapa transmisora. Fuente: Elaboración Propia.

Calibración del sensor PMS3003

Para realizar el proceso de calibración del sensor PMS7003, para obtener mediciones de material particulado de 2.5 micrómetros, En primera instancia, el sensor fue calibrado, exponiéndolo a diferentes ambientes controlados, es decir, que se inicia en ambiente limpio, para operar en condiciones normales, con el objetivo de estabilizar las lecturas y asegurar una correcta medición. La siguiente etapa, es realizar mediciones en un ambiente donde se tenga concentración alta de partículas, en este caso se realizaron mediciones en una parte urbana, donde se tiene tránsito y obras de construcción cercanas, puesto que es, fundamental para reflejar mediciones en diferentes entornos.

Para finalizar con el proceso de calibración del sensor PMS7003, es importante realizar una comparación con un equipo de referencia, en este caso, se utiliza la estación de monitoreo del Portal Aire Ciudadano que se encuentra instalado en Ecoparque Sabana, para tener un dispositivo de referencia, colocándolos en el mismo lugar, para asegurar que las mediciones sean consistentes en las dos estaciones de monitoreo.

Se dejaron ambos dispositivos por una hora, para registrar los valores obtenidos en las dos estaciones y calcular el factor de corrección. Teniendo el factor de corrección, se aplicó el ajuste directamente en software de la estación de monitoreo, con el propósito de realizar diferentes mediciones en diversas condiciones de calidad de aire, verificando así, la fiabilidad del factor de corrección.

Calibración del sensor SHT31-D

Para la calibración del sensor que permite medir temperatura y humedad, el cual es, el sensor SHT31-D, es importante tener en cuenta que es un sensor que funciona de manera digital, teniendo la posibilidad de realizar un proceso de calibración sencillo, puesto que ya suelen venir calibrados de fábrica, sin embargo, se realiza el proceso de verificación de su precisión. En primera instancia, se realiza una comparación de medidas, con la estación de monitoreo que se encuentra ubicada en Ecoparque Sabana, ubicándolos en el mismo entorno y dejándolos funcionar mediante una hora, para obtener mayores registros de valores y así, poder realizar la calibración. Se realizó esto en varios entornos de ecoparque para observar y ajustar las lecturas en el software del sistema, como se muestra en la figura 52.

```

95 void leerPMS3003() {
96     pms.wakeUp();
97     delay(3000);
98     pms.requestRead();
99
100    if (pms.readUntil(data)) {
101        pm25 = data.PM_AE_UG_2_5;
102    } else {
103        Serial.println("Error al leer los datos del PMS3003");
104    }
105    pms.sleep();
106 }

```

Figura 52 . Función para la lectura de material particulado PM2.5.

Calibración del sensor CCS811

El sensor CCS811 que permite medir compuestos volátiles orgánicos (COV), necesita un proceso de calibración más cuidadoso, donde implica proporcionar medidas de referencia de temperatura y humedad, utilizando el sensor SHT31-D, de la estación de monitoreo de Ecoparque Sabana. En primera instancia, es importante estabilizar el sensor en un ambiente durante al menos 30 minutos, durante este proceso, es necesario evitar manipular el sensor. Se utilizó el sensor SHT31-D como se mencionó anteriormente, para realizar mediciones en un mismo ambiente, ayudando a calcular las mediciones de COV, puesto que estos gases, se ven afectados por los cambios que se tienen en la temperatura y humedad. Posteriormente se calcularon y se compararon las mediciones obtenidas, aplicando el factor de corrección en el código utilizando la librería que proporciona el sensor CCS811, como se muestra en la siguiente

imagen, donde fue necesario restaurar los valores base (baseline) del sensor, como se muestra en la figura 53.

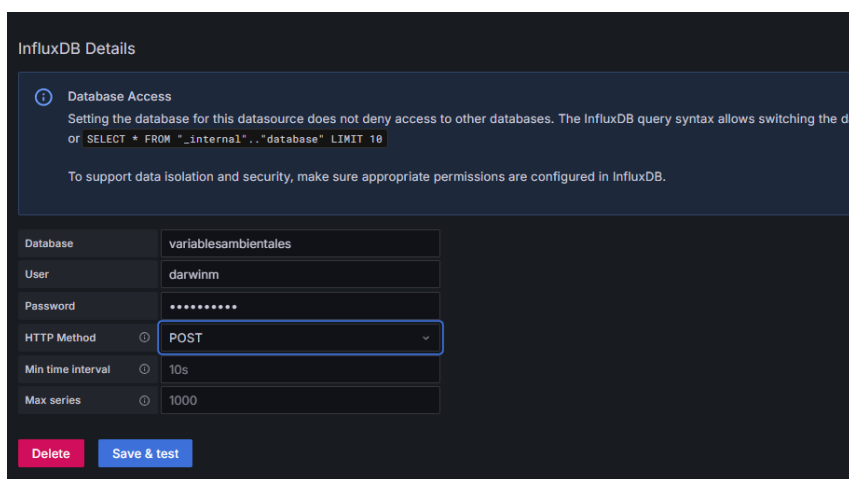
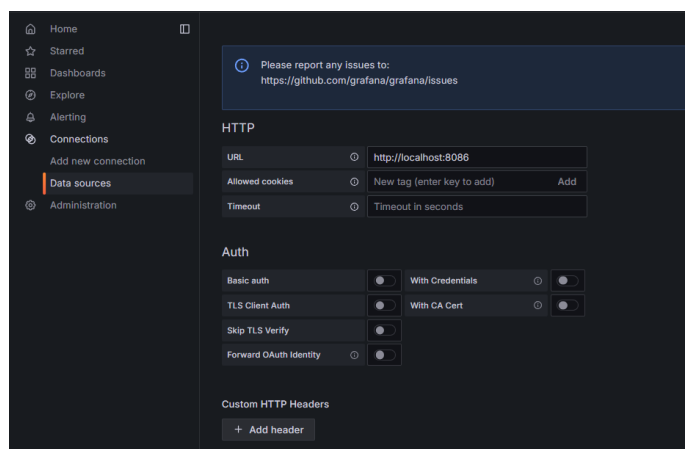
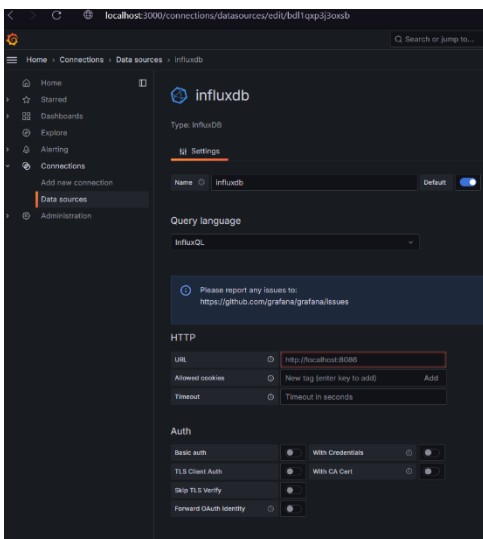
```

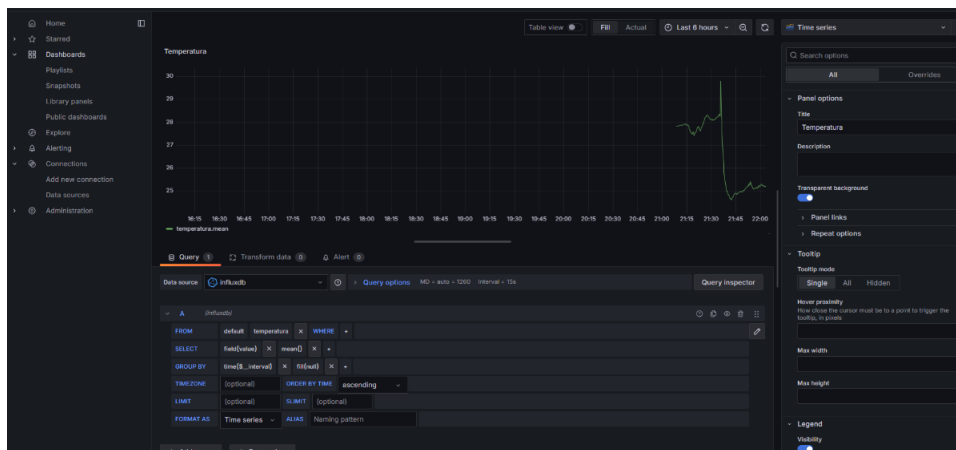
108 void leerCCS811() {
109     if(ccs.available()){
110         if(!ccs.readData()){
111             ccs811_co2 = ccs.geteCO2();
112             ccs811_tvoc = ccs.getTVOC();
113         } else {
114             Serial.println("ERROR al leer CCS811!");
115         }
116     }
117 }

```

Figura 53. Código aplicado en el software del sistema para mediciones de COV.

Anexo 6. Configuración para la implementación grafica en Grafana.





Anexo 7. Guía para la Configuración para el inicio de InfluxDB.

```

Microsoft Windows [Versión 10.0.22631.3527]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Program Files\InfluxData\influxdb>influxd
2024-05-05T02:09:40.007851Z info Welcome to InfluxDB {"log_id": "0oyd3veW000", "version": "v2.7.6", "commit":
"3c58c06206", "build_date": "2024-04-12T21:51:33Z", "log_level": "info"}
2024-05-05T02:09:40.0241196Z info Resources opened {"log_id": "0oyd3veW000", "service": "bolt", "path": "C:
\Users\darwi\\influxdbv2\influxdb.bolt"}
2024-05-05T02:09:40.024249Z info Resources opened {"log_id": "0oyd3veW000", "service": "sqlite", "path": "
C:\Users\darwi\\influxdbv2\influxdb.sqlite"}
2024-05-05T02:09:40.027766Z info Bringing up metadata migrations {"log_id": "0oyd3veW000", "service": "KV migrati
ons", "migration_count": 20}
2024-05-05T02:09:40.365137Z info Bringing up metadata migrations {"log_id": "0oyd3veW000", "service": "SQL migrat
ions", "migration_count": 8}
2024-05-05T02:09:40.493751Z info Using data dir {"log_id": "0oyd3veW000", "service": "storage-engine", "service":
"store", "path": "C:\Users\darwi\\influxdbv2\engine\data"}
2024-05-05T02:09:40.495341Z info Compaction settings {"log_id": "0oyd3veW000", "service": "storage-engine", "servi
ce": "store", "max_concurrent_compactions": 4, "throughput_bytes_per_second": 50331648, "throughput_bytes_per_secon
d_burst": 50331648}
2024-05-05T02:09:40.495858Z info Open store (start) {"log_id": "0oyd3veW000", "service": "storage-engine", "
service": "store", "op_name": "tsdb_open", "op_event": "start"}
2024-05-05T02:09:40.496479Z info Open store (end) {"log_id": "0oyd3veW000", "service": "storage-engine", "
service": "store", "op_name": "tsdb_open", "op_event": "end", "op_elapsed": "0.620ms"}
2024-05-05T02:09:40.496479Z info Starting retention policy enforcement service {"log_id": "0oyd3veW000", "servi
ce": "retention", "check_interval": "30m"}
2024-05-05T02:09:40.496479Z info Starting precreation service {"log_id": "0oyd3veW000", "service": "shard-prec
reation", "check_interval": "10m", "advance_period": "30m"}
2024-05-05T02:09:40.501812Z info Starting query controller {"log_id": "0oyd3veW000", "service": "storage-re
ads", "concurrency_quota": 1024, "initial_memory_bytes_quota_per_query": 9223372036854775807, "memory_bytes_quota_per_qu
ery": 9223372036854775807, "max_memory_bytes": 0, "queue_size": 1024}

```

```

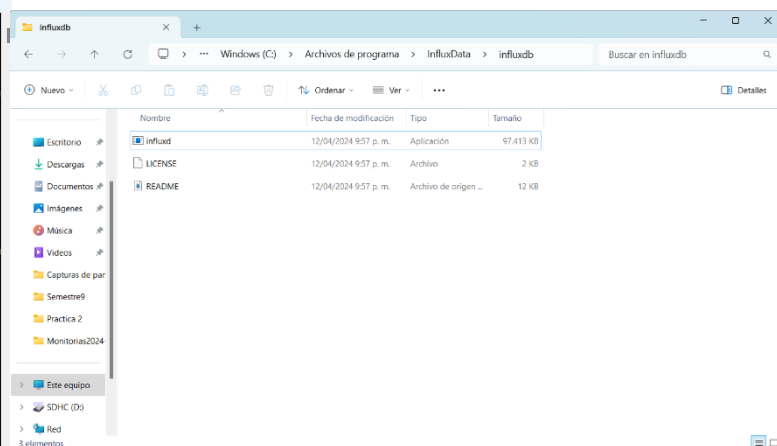
Administrador: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. https://aka.ms/PSWindows

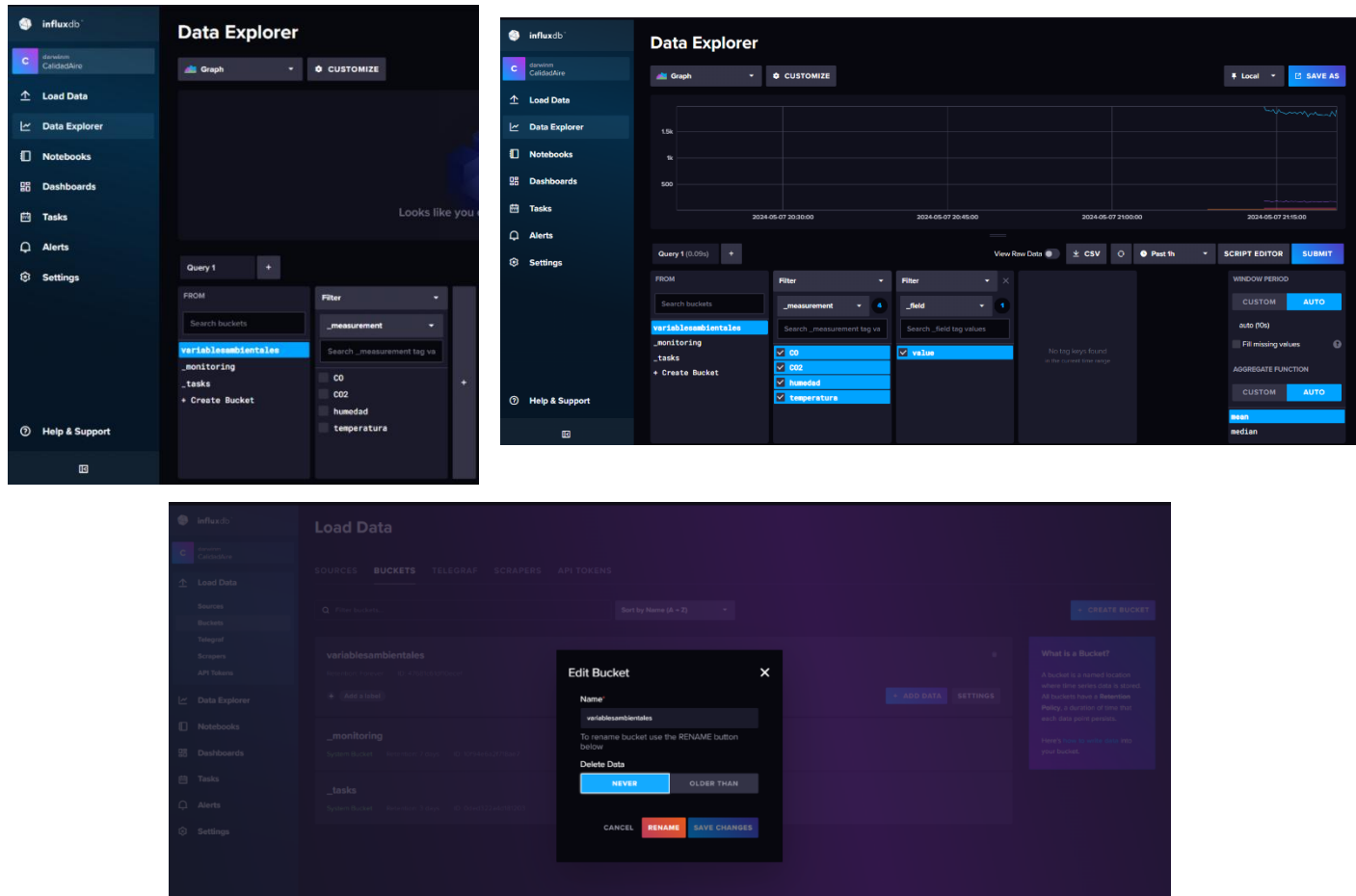
PS C:\WINDOWS\system32> Expand-Archive .\influxdb2-2.7.6-windows.zip -DestinationPath 'C:\Program Files\InfluxData\influxdb\
Expand-Archive : La ruta de acceso ".\influxdb2-2.7.6-windows.zip" no existe o no es una ruta de acceso válida del
sistema de archivos.
En línea: 1 Carácter: 1
+ Expand-Archive .\influxdb2-2.7.6-windows.zip -DestinationPath 'C:\Pro...
+ ~~~~~
+ CategoryInfo          : InvalidArgument: (.\influxdb2-2.7.6-windows.zip:String) [Expand-Archive], InvalidOperati
onException
+ FullyQualifiedErrorId : ArchiveCmdletPathNotFound_Expand-Archive

PS C:\WINDOWS\system32> wget https://dl.influxdata.com/influxdb/releases/influxdb2-2.7.6-windows.zip -UseBasicParsing -O
influxdb2-2.7.6-windows.zip
>> Expand-Archive .\influxdb2-2.7.6-windows.zip -DestinationPath 'C:\Program Files\InfluxData\influxdb\'
PS C:\WINDOWS\system32>
PS C:\WINDOWS\system32>

```



Anexo 8. Guía para el ingreso de las variables de almacenamiento en InfluxDB.



Anexo 9. Evidencias de las mediciones realizadas en Ecoparque Sabana.





Anexo 10. Evidencias de las mediciones realizadas en la Universidad Pedagógica Nacional.





Anexo 11. Archivo para la impresión 3D para la estación receptora y transmisora.

Enlace del archivo:

Anexo 12. Guía para la implementación y desarrollo del sistema de monitoreo.

Enlace de la guía: