

UNIVERSIDAD LAICA ELOY ALFARO DE MANABÍ



TRABAJO DE TITULACIÓN
MODALIDAD DE PROYECTO INTEGRADOR
PREVIO A LA OBTENCIÓN DEL TÍTULO DE:
INGENIERO EN TECNOLOGÍAS DE LA INFORMACIÓN

TEMA DEL PROYECTO:
“DESARROLLO DE UNA APLICACIÓN MÓVIL HÍBRIDA PARA MONITOREO DE CALIDAD DEL AIRE UTILIZANDO MQTT Y WIFI-DIRECT CON RASPBERRY PI EN LA UNIVERSIDAD LAICA ELOY ALFARO DE MANABÍ, MATRIZ MANTA.”

AUTOR:
ANCHUNDIA TRIVIÑO TOMY ARNOL

TUTOR:
ING. WILLIAN JESÚS ZAMORA MERO, PHD

FACULTAD DE CIENCIAS DE LA VIDA Y TECNOLOGÍAS
CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN
MANTA – MANABÍ - ECUADOR-2026

CERTIFICACIÓN DEL DIRECTOR DE TRABAJO DE GRADUACIÓN

 Uleam UNIVERSIDAD LAICA ELOY ALFARO DE MANABÍ	NOMBRE DEL DOCUMENTO: CERTIFICADO DE TUTOR(A).	CÓDIGO: PAT-04-F-004
	PROCEDIMIENTO: TITULACIÓN DE ESTUDIANTES DE GRADO BAJO LA UNIDAD DE INTEGRACIÓN CURRICULAR	REVISIÓN: 1
		Página 1 de 1

CERTIFICACIÓN

En calidad de docente tutor(a) de la Facultad de Ciencias de la Vida y Tecnología de la Universidad Laica "Eloy Alfaro" de Manabí, CERTIFICO:

Haber dirigido, revisado y aprobado preliminarmente el Trabajo de Integración Curricular bajo la autoría del estudiante Anchundia Triviño Tomy Arnol, con CI: 1313364414, legalmente matriculado en la carrera de Ingeniería en Tecnologías de la Información, período académico 2025-2026, cumpliendo el total de 384 horas, cuyo tema del proyecto es **"DESARROLLO DE UNA APLICACIÓN MÓVIL HÍBRIDA PARA MONITOREO DE CALIDAD DEL AIRE UTILIZANDO MQTT Y WIFI-DIRECT CON RASPBERRY PI EN LA UNIVERSIDAD LAICA ELOY ALFARO DE MANABÍ, MATRIZ MANTA"**.

La presente investigación ha sido desarrollada en apego al cumplimiento de los requisitos académicos exigidos por el Reglamento de Régimen Académico y en concordancia con los lineamientos internos de la opción de titulación en mención, reuniendo y cumpliendo con los méritos académicos, científicos y formales, y la originalidad del mismo, requisitos suficientes para ser sometida a la evaluación del tribunal de titulación que designe la autoridad competente.

Particular que certifico para los fines consiguientes, salvo disposición de Ley en contrario.

Manta, 05 de febrero de 2026.

Lo certifico,



Ing. Wilber Jesús Zamora Mero

Docente Tutor(a)

DECLARACIÓN DE AUTORÍA DEL ESTUDIANTE

Declaro que el presente trabajo de titulación que tiene como tema “DESARROLLO DE UNA APLICACIÓN MÓVIL HÍBRIDA PARA MONITOREO DE CALIDAD DEL AIRE UTILIZANDO MQTT Y WIFI-DIRECT CON RASPBERRY PI EN LA UNIVERSIDAD LAICA ELOY ALFARO DE MANABÍ, MATRIZ MANTA” es de mi autoría original y que ha sido elaborado como resultado de mi propio trabajo académico, cumpliendo con los principios de honestidad intelectual y ética profesional.

Declaro además que el contenido de este documento no ha sido presentado previamente para la obtención de otro título o grado académico, y que todas las fuentes de información utilizadas han sido debidamente citadas y referenciadas conforme a las normas establecidas.



Anchundia Triviño Tomy Arnol

C.I. 131336441-4

DEDICATORIA

A Dios.

A mis padres y hermanos que son pilar fundamental en mi vida

A mis demás familiares, por la confianza y apoyo.

A hermanos de vida, amistades por la motivación.

A la vida por ayudarme a convertirme en quien soy.

A mi tutor y docentes que formaron parte de mi formación.

Anchundia Triviño Tomy Arnol

AGRADECIMIENTO

A Dios, por los dones como la fortaleza, sabiduría y perseverancia para culminar esta etapa académica.

A mamá y papá, por su apoyo incondicional, por los consejos y sobre todo por nunca permitirme rendir en momentos cuando todo tornaba se gris.

A mis hermanos, por su acompañamiento y por ser fuente de motivación.

A mis demás familiares por la confianza y apoyo infinito.

A mis hermanos de vida, amistades que confían y que siempre tienen un consejo palabras de motivación.

Finalmente, quiero agradecerme, por permitirme dejar guiar de todas las personas mencionadas, por la constancia y por la determinación.

Anchundia Triviño Tomy Arnol

ÍNDICE DE CONTENIDOS

Resumen	10
Abstract	11
Introducción	12
Capítulo I.....	14
1 Planteamiento del problema	14
1.1 Ubicación y contextualización del problema	14
1.2 Planteamiento del problema.....	15
1.3 Formulación del problema	16
1.4 Diagrama causa – efecto del problema	16
1.5 Objetivos	18
1.6 Justificación	18
1.7 Impactos esperados	19
Capítulo II	20
2 Marco teórico de la investigación	20
2.1 Antecedentes teóricos de la investigación	20
2.2 Bases teóricas.....	24
Capítulo III.....	40
3 Marco investigativo.....	40
3.1 Metodología aplicada en la investigación.....	40
3.2 Métodos y técnicas de investigación.....	41
3.3 Metodología de desarrollo: Metodología ágil Scrum	41
Capítulo IV.....	43
4 Marco propositivo	43
4.1 Descripción general de la solución propuesta.....	43

4.2	Determinación de recursos del proyecto.....	44
4.3	Etapas de desarrollo de la propuesta.....	46
Capítulo V		66
5	Resultados y validación.....	66
5.1	Resultados de la implementación.....	66
5.2	Resultados del backend y servicios expuestos.....	67
5.3	Resultados de la aplicación móvil.....	69
5.4	Resultados del sistema de notificaciones y eventos.....	74
5.5	Resultados de la conectividad y comunicación	77
5.6	Validación de los requisitos del sistema	82
5.7	Análisis del desempeño del sistema.....	84
5.8	Discusión de resultados.....	85
5.9	Limitaciones identificadas	87
Capítulo VI.....		89
6	Conclusiones	89
7	Recomendaciones.....	90
Bibliografía		91
Anexos.....		95

ÍNDICE DE TABLAS

Tabla 1	Características de tecnologías de comunicación en IoT	28
Tabla 2	Características de protocolos de comunicación en IoT.....	30
Tabla 3	Aplicaciones móviles existentes para el monitoreo de la calidad del aire.....	39
Tabla 4	Definición de roles Scrum	42
Tabla 5	Identificación de Stakeholders y actores.....	47
Tabla 6	Requisitos funcionales del sistema	48
Tabla 7	Requisitos no funcionales del sistema	49
Tabla 8	Planificación de Sprints del proyecto	51
Tabla 9	Comparación entre conexión local y conexión remota.....	81
Tabla 10	Validación de requisitos funcionales	82
Tabla 11	Validación de requisitos no funcionales	83

ÍNDICE DE FIGURAS

Figura 1 Ubicación satelital de la Universidad Laica Eloy Alfaro de Manabí	15
Figura 2 Diagrama de Ishikawa	17
Figura 3 Origen y evolución del Internet de las Cosas (IoT).....	25
Figura 4 Niveles recomendados de las directrices sobre la calidad del aire y metas intermedias.....	26
Figura 5 Modelo de publish/subscribe.....	31
Figura 6 Diagrama arquitectónico y tecnológico del sistema.....	53
Figura 7 Máquina virtual configurada en la plataforma Azure	56
Figura 8 Acceso y configuración del sistema operativo Ubuntu en la máquina virtual	57
Figura 9 Verificación del entorno de producción previo al despliegue	58
Figura 10 Backend desplegado y gestionado mediante PM2	59
Figura 11 Diagrama ERD de la base de datos en PostgreSQL operativa en el entorno de producción.....	60
Figura 12 Broker activo y publicando datos	60
Figura 13 Servicio del broker de IoT Uleam	61
Figura 14 Dispositivo configurado y parámetros agregados	61
Figura 15 Logs de la aplicación conectándose y recibiendo datos	62
Figura 16 Mecanismo de autenticación y control de acceso al sistema.....	64
Figura 17 Funcionamiento general del sistema de monitoreo de la calidad del aire	67
Figura 18 Documentación de la API REST del sistema mediante Swagger	68
Figura 19 Pantalla principal de la aplicación móvil	70
Figura 20 Consulta histórica de datos desde la aplicación móvil	71
Figura 21 Funcionamiento de la aplicación en modo local	72
Figura 22 Notificación generada ante un evento de calidad del aire	73
Figura 23 Aplicación móvil ejecutándose en plataformas Android e iOS.	74

Figura 24	Notificación generada ante evento de calidad del aire	76
Figura 25	Recepción de mensajes MQTT durante la operación del sistema.....	78
Figura 26	Funcionamiento del sistema en modo de conexión local	79
Figura 27	Sistema de IoT Uleam categoría Calidad del aire - mapa	80
Figura 28	Sistema de IoT Uleam categoría Calidad del aire - Datos en tiempo real.....	80
Figura 29	Diseño de la app en Figma	95
Figura 30	Revisión del correcto funcionamiento del sistema.....	95
Figura 31	Sprints en Notion para gestionar y monitorear el proceso de las tareas.....	96

RESUMEN

El presente proyecto aborda el desarrollo de una aplicación móvil híbrida orientada al monitoreo de la calidad del aire en el campus de la Universidad Laica Eloy Alfaro de Manabí, matriz Manta. La solución integra un microcomputador Raspberry Pi conectado a sensores ambientales capaces de medir distintos parámetros relacionados con la calidad del aire, como material particulado, temperatura, humedad y gases, entre otros.

Los datos recolectados son procesados localmente y transmitidos a la aplicación móvil mediante el protocolo MQTT, permitiendo su visualización en tiempo real. Adicionalmente, se implementa un mecanismo de comunicación local directa entre el dispositivo de adquisición de datos y la aplicación móvil, lo que garantiza el acceso a la información incluso en ausencia de conexión a internet.

El sistema se enmarca dentro del paradigma de Internet de las Cosas (IoT), permitiendo una monitorización continua y precisa del entorno. La aplicación ofrece una interfaz intuitiva para la consulta del estado de la calidad del aire, contribuyendo a la toma de decisiones informadas relacionadas con el cuidado de la salud y evidenciando el potencial de las tecnologías móviles e IoT en la gestión ambiental de entornos universitarios.

PALABRAS CLAVE

Calidad del aire, Internet de las cosas (IoT), Aplicación móvil, MQTT, Raspberry Pi, Monitoreo ambiental.

ABSTRACT

This project addresses the development of a hybrid mobile application aimed at monitoring air quality on the campus of the Universidad Laica Eloy Alfaro de Manabí, Manta campus. The proposed solution integrates a Raspberry Pi microcomputer connected to environmental sensors capable of measuring various parameters related to air quality, such as particulate matter, temperature, humidity, and gases, among others.

The collected data are processed locally and transmitted to the mobile application through the MQTT protocol, allowing real-time visualization. Additionally, a local direct communication mechanism is implemented between the data acquisition device and the mobile application, ensuring access to information even in the absence of an internet connection.

The system is framed within the Internet of Things (IoT) paradigm, enabling continuous and accurate environmental monitoring. The mobile application provides an intuitive interface for consulting air quality status, contributing to informed decision-making related to health care and demonstrating the potential of mobile and IoT technologies in environmental management within university environments.

KEYWORDS

Air quality, Internet of Things (IoT), Mobile application, MQTT, Raspberry Pi, Environmental monitoring.

INTRODUCCIÓN

La calidad del aire es un factor directamente relacionado con el bienestar humano y el equilibrio de los ecosistemas. La presencia de contaminantes atmosféricos, originados principalmente por el tránsito vehicular, la actividad industrial y los incendios forestales, ha incrementado la concentración de gases nocivos y material particulado en el ambiente. Esta situación representa un riesgo constante para la salud de la población y ha motivado el interés por comprender y controlar el comportamiento de la contaminación atmosférica mediante el uso de herramientas tecnológicas (OMS, 2024).

En el contexto ecuatoriano, la contaminación del aire constituye una preocupación creciente, especialmente en áreas urbanas donde la expansión de actividades productivas y el aumento de automóviles generan mayores emisiones contaminantes. La ciudad de Manta, caracterizada por su actividad portuaria, su desarrollo económico y su crecimiento urbano acelerado, presenta condiciones que pueden incidir negativamente en la calidad del aire. Estas circunstancias evidencian la necesidad de contar con mecanismos que permitan observar y analizar de forma continua los niveles de contaminación atmosférica, con el fin de apoyar la gestión ambiental y la toma de decisiones a nivel local.

Como parte de los esfuerzos institucionales para abordar esta problemática, la Universidad Laica Eloy Alfaro de Manabí (ULEAM) desarrolló el proyecto de investigación titulado Infraestructura Tecnológica aplicando dispositivos IoT y redes de sensores móviles: Monitoreo y evaluación del impacto ambiental en entornos urbanos: Caso de estudio Manta. En el marco de este proyecto se implementó un dispositivo basado en una Raspberry Pi, conectado a sensores capaces de medir concentraciones de gases como CO₂, CO, NO₂, SO₂ y O₃, así como material particulado PM_{2.5} y PM₁₀, además de variables climáticas como temperatura, humedad y presión atmosférica (Zamora, Machuca, Zambrano, & Ponce, 2023). No obstante, la información obtenida por este sistema se almacena de manera local, lo que limita la visualización de los datos en tiempo real y dificulta el acceso a registros históricos por parte de los usuarios.

Frente a esta limitación, el presente trabajo de titulación propone el desarrollo de una aplicación móvil híbrida orientada a la visualización y monitoreo de la calidad del aire, estableciendo una comunicación directa con los dispositivos de adquisición de datos basados en Raspberry Pi, así como la integración con servicios en la nube para la consulta de información histórica. La aplicación busca facilitar la interpretación de los datos ambientales

mediante una interfaz accesible e intuitiva, permitiendo a los usuarios conocer el estado de la calidad del aire de forma oportuna.

El documento se estructura en seis capítulos. En el Capítulo I se presenta la problemática abordada y los objetivos del proyecto; el Capítulo II desarrolla el estado del arte y el marco conceptual; el Capítulo III describe la metodología aplicada para el desarrollo del sistema; el Capítulo IV detalla la propuesta tecnológica implementada; el Capítulo V expone los resultados obtenidos; y el Capítulo VI recoge las conclusiones y recomendaciones derivadas del estudio.

CAPÍTULO I

1 Planteamiento del problema

1.1 Ubicación y contextualización del problema

La Universidad Laica Eloy Alfaro de Manabí (ULEAM) se encuentra ubicada en la ciudad de Manta, provincia de Manabí, dentro de la Ciudadela Universitaria situada en la vía San Mateo. Este campus concentra una alta afluencia diaria de estudiantes, docentes, personal administrativo y visitantes, lo que genera una dinámica constante de actividades académicas, administrativas y de movilidad en su entorno inmediato.

En los últimos años, la ULEAM ha incorporado diversas soluciones tecnológicas orientadas a la gestión institucional, tales como plataformas académicas, sistemas de información administrativa, mecanismos de control de acceso y herramientas digitales de comunicación. Dentro de este contexto, la universidad dispone actualmente de una solución para la visualización de información relacionada con la calidad del aire, a través del sitio web institucional IoT ULEAM, el cual permite consultar datos ambientales desde una plataforma web.

No obstante, dicha solución presenta limitaciones en cuanto a su alcance y accesibilidad. La plataforma se encuentra orientada exclusivamente a la visualización web y muestra únicamente información asociada al material particulado $PM_{2.5}$ y PM_{10} , restringiendo el acceso a otros parámetros ambientales relevantes, así como la consulta desde dispositivos móviles. Esta situación reduce la posibilidad de que la comunidad universitaria acceda de forma ágil y oportuna a información ambiental mientras se desplaza dentro del campus.

La ausencia de una aplicación móvil dedicada para el monitoreo de la calidad del aire limita el aprovechamiento de los datos recolectados y dificulta su interpretación en tiempo real. Esta condición adquiere mayor relevancia si se considera el crecimiento urbano de la ciudad de Manta, el incremento del tránsito vehicular en zonas aledañas al campus universitario y la presencia de actividades económicas que pueden influir en las condiciones atmosféricas del entorno.

En este escenario, se evidencia la necesidad de contar con una herramienta tecnológica que permita consultar y visualizar información ambiental de manera continua y accesible, adaptada a escenarios de movilidad dentro del campus universitario. Como parte del proceso de contextualización territorial, esta sección se complementa con una representación geoespacial del área de estudio, obtenida mediante herramientas cartográficas, acompañada de las coordenadas geográficas correspondientes, con el fin de delimitar con precisión el espacio en el que se desarrolla el presente trabajo.

Figura 1

Ubicación satelital de la Universidad Laica Eloy Alfaro de Manabí



Nota. Obtenido de Google Earth con las coordenadas: 0°57'10"S 80°44'42"W

1.2 Planteamiento del problema

La calidad del aire en espacios educativos constituye un factor determinante para la salud y el desempeño de quienes desarrollan actividades académicas y administrativas. De acuerdo con la Agencia de Protección Ambiental (EPA, 2025), mantener un entorno saludable no depende únicamente de la infraestructura física o del confort de las instalaciones, sino también de condiciones ambientales adecuadas, entre las que destaca la calidad del aire. La exposición prolongada a contaminantes atmosféricos, tanto en ambientes interiores como exteriores, representa un riesgo para la salud de estudiantes, docentes y personal administrativo.

Diversos estudios señalan que contaminantes como el material particulado $PM_{2.5}$ y PM_{10} , así como gases tales como NO_2 , SO_2 , O_3 y CO , pueden afectar el sistema respiratorio y cardiovascular, incrementando la probabilidad de enfermedades crónicas y eventos adversos para la salud (OMS, 2024). No obstante, la problemática asociada a la calidad del aire no se limita únicamente a la presencia de contaminantes, sino también a la disponibilidad y accesibilidad de información que permita comprender su comportamiento y evaluar posibles riesgos.

En el contexto universitario de la Universidad Laica Eloy Alfaro de Manabí, si bien existen iniciativas orientadas a la medición de parámetros ambientales mediante dispositivos y

sensores, la información generada no se encuentra plenamente disponible para la comunidad universitaria a través de herramientas accesibles y adaptadas a escenarios de movilidad. La ausencia de una aplicación móvil que permita visualizar los datos ambientales de forma clara y en tiempo real dificulta la interpretación de la información recolectada y limita su uso práctico para la toma de decisiones informadas.

Adicionalmente, aunque existen aplicaciones móviles de alcance internacional para el monitoreo de la calidad del aire, como IQAir, AirVisual o Plume Labs, estas soluciones se basan principalmente en estimaciones globales o regionales y no están diseñadas para integrar datos provenientes de sensores instalados en ubicaciones específicas. En consecuencia, la información que ofrecen no refleja con precisión las condiciones ambientales particulares del campus universitario de la ULEAM.

Esta situación evidencia una problemática concreta: la falta de una herramienta tecnológica móvil que permita monitorear, visualizar y comprender la calidad del aire del campus universitario a partir de datos generados localmente, de manera accesible, confiable y en tiempo real. La ausencia de este tipo de solución limita el aprovechamiento de la información ambiental existente y reduce su impacto potencial en la gestión institucional, la investigación y la protección de la salud de la comunidad universitaria.

1.3 Formulación del problema

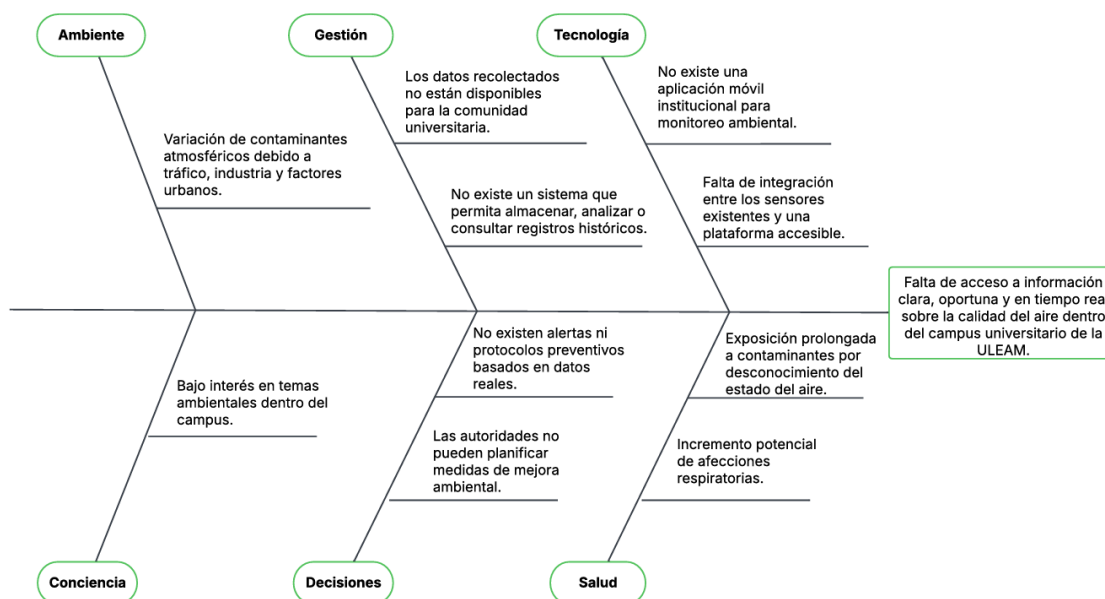
¿Cómo desarrollar una aplicación móvil que permita monitorear la calidad del aire en tiempo real dentro del campus de la Universidad Laica Eloy Alfaro de Manabí, a partir de datos generados por sensores instalados en el entorno universitario, facilitando el acceso y la visualización de la información para la comunidad universitaria?

1.4 Diagrama causa – efecto del problema

El presente apartado muestra un diagrama de causa-efecto, también conocido como diagrama de Ishikawa o de espina de pescado, con el objetivo de representar de manera visual y estructurada los factores que originan el problema central identificado: la falta de un sistema tecnológico accesible que permita monitorear en tiempo real la calidad del aire dentro del campus de la Universidad Laica Eloy Alfaro de Manabí. Este tipo de diagrama facilita la identificación, organización y análisis de las causas que intervienen en la problemática, contribuyendo a una mejor comprensión de su naturaleza y de la relación existente entre sus distintos elementos.

Figura 2

Diagrama de Ishikawa



Nota. Diagrama causa efecto diseñado en la plataforma de LucidChart. Elaboración propia

A partir del análisis representado en el diagrama de causa-efecto, se identifican diversos factores interrelacionados que dan origen al problema central. Entre ellos se encuentran los factores tecnológicos, asociados a la ausencia de una aplicación móvil institucional y a la limitada integración entre los sensores ambientales y plataformas accesibles para los usuarios, así como los factores de gestión, vinculados a la falta de mecanismos que permitan almacenar, consultar y difundir la información ambiental generada.

De igual forma, intervienen condiciones ambientales externas relacionadas con la variación de contaminantes atmosféricos presentes en el entorno del campus, junto con aspectos asociados a la toma de decisiones y al nivel de conciencia ambiental dentro de la comunidad universitaria. La interacción de estos factores produce efectos directos, tales como la exposición prolongada a contaminantes, riesgos potenciales para la salud, la dificultad para establecer alertas o estrategias preventivas basadas en datos reales y un acceso limitado a información confiable y actualizada sobre la calidad del aire.

1.5 Objetivos

1.5.1 *Objetivo general*

Desarrollar e implementar una aplicación móvil que permita el monitoreo de la calidad del aire en la Universidad Laica Eloy Alfaro de Manabí, mediante el uso de protocolos de comunicación MQTT y conectividad inalámbrica para la transmisión de datos ambientales.

1.5.2 *Objetivos específicos*

- Identificar los parámetros atmosféricos relevantes para el monitoreo de la calidad del aire dentro del contexto universitario, considerando estándares nacionales, internacionales y normativas locales aplicables.
- Diseñar la arquitectura tecnológica del sistema, contemplando la comunicación mediante MQTT, la conectividad inalámbrica y la integración con servicios en la nube para la gestión de registros históricos.
- Desarrollar la aplicación móvil incorporando funcionalidades de visualización de datos en tiempo real, generación de alertas y almacenamiento de información ambiental.
- Integrar la aplicación móvil con el sistema de hardware disponible, utilizando protocolos de comunicación adecuados que garanticen la interoperabilidad, estabilidad y correcta recepción de los datos generados por los sensores.
- Evaluar el funcionamiento, la usabilidad y el rendimiento de la aplicación a través de pruebas controladas, considerando la experiencia de los usuarios y el análisis de los resultados obtenidos.

1.6 Justificación

La calidad del aire en espacios educativos influye directamente en el bienestar de estudiantes, docentes y personal administrativo, quienes desarrollan la mayor parte de sus actividades diarias dentro del campus universitario. Diversos organismos internacionales han señalado que la exposición a contaminantes atmosféricos puede generar efectos adversos en la salud, tales como afecciones respiratorias y cardiovasculares, así como impactos negativos en el rendimiento físico y cognitivo (OMS, 2024). En este contexto, contar con información ambiental confiable permite comprender mejor las condiciones del entorno y adoptar medidas orientadas a la prevención de riesgos.

En la Universidad Laica Eloy Alfaro de Manabí, el acceso a datos sobre la calidad del aire no se encuentra plenamente disponible a través de herramientas móviles que permitan una consulta directa y oportuna. Aunque existen plataformas comerciales de monitoreo ambiental, estas soluciones no están diseñadas para integrar mediciones obtenidas a partir de sensores instalados en el campus ni responden a las particularidades del entorno institucional. Esta limitación reduce la utilidad práctica de la información ambiental para la comunidad universitaria.

Desde una perspectiva tecnológica y académica, el desarrollo de una aplicación móvil adaptada al contexto institucional se justifica como una alternativa que permite aprovechar los datos generados localmente y presentarlos de forma accesible. La propuesta contribuye al fortalecimiento del uso de tecnologías emergentes en el ámbito universitario y sienta las bases para futuras iniciativas relacionadas con la investigación, la salud y la gestión ambiental.

1.7 Impactos esperados

La implementación del presente proyecto permitirá disponer de una herramienta tecnológica orientada a la visualización de información ambiental generada dentro del campus universitario. A través de la aplicación móvil, se facilitará el acceso a datos actualizados sobre la calidad del aire, lo que contribuirá a una mayor comprensión de las condiciones ambientales por parte de la comunidad universitaria.

Desde el ámbito social y educativo, se espera que el acceso a información clara y oportuna fomente una mayor conciencia sobre la importancia de un entorno saludable, promoviendo prácticas responsables y una toma de decisiones informada (UNESCO, 2022). Asimismo, la disponibilidad de datos ambientales confiables puede apoyar iniciativas relacionadas con la prevención de riesgos y el bienestar institucional.

En el plano tecnológico, la solución propuesta presenta un enfoque adaptable y escalable, lo que permitirá su integración con futuros proyectos, la ampliación a nuevos puntos de medición o la incorporación progresiva de funcionalidades adicionales. De esta manera, el proyecto aporta una base tecnológica que puede ser aprovechada para fortalecer procesos de investigación, sostenibilidad y desarrollo académico dentro de la universidad.

CAPÍTULO II

En este capítulo se desarrolla el marco teórico que sustenta el proyecto, proporcionando los conceptos, fundamentos y tecnologías relacionadas con la solución propuesta para el monitoreo de la calidad del aire. Aquí se abordan los antecedentes y bases científicas sobre la contaminación atmosférica, los parámetros ambientales relevantes y su importancia para la salud y el medio ambiente. Además, se describen las tecnologías utilizadas, incluyendo sensores ambientales, microcontroladores, comunicación inalámbrica, bases de datos, servicios en la nube y el desarrollo de aplicaciones móviles. Este marco conceptual permitirá comprender el contexto técnico y científico necesario para la ejecución del proyecto, facilitando la interpretación y justificación de la solución planteada.

2 Marco teórico de la investigación

2.1 Antecedentes teóricos de la investigación

En los últimos años, la investigación en monitoreo ambiental ha incorporado tecnologías IoT para capturar variables atmosféricas y presentar la información mediante plataformas digitales. En este apartado se revisan trabajos previos relacionados con sistemas de monitoreo de calidad del aire, arquitecturas IoT, transmisión de datos en tiempo real y mecanismos de visualización orientados a usuarios, con el fin de identificar aportes y criterios útiles para la presente investigación.

2.1.1 Sistemas de monitoreo de calidad del aire en entornos urbanos

Diversos estudios han abordado el monitoreo de la calidad del aire en entornos urbanos mediante el uso de sistemas tecnológicos que integran sensores ambientales y plataformas digitales para la visualización de datos. Estas investigaciones se han enfocado en medir contaminantes atmosféricos como material particulado y gases nocivos, con el propósito de analizar su comportamiento y evaluar posibles impactos en la salud de la población.

En este contexto, (Zamora, Machuca, Zambrano, & Ponce, 2023) desarrollaron un sistema de monitoreo ambiental orientado a entornos urbanos, en el cual se emplearon sensores para la medición de parámetros como material particulado y gases contaminantes. El sistema permitió recopilar información ambiental de forma continua y almacenarla para su posterior análisis. Este trabajo resulta relevante para la presente investigación, ya que evidencia la viabilidad de implementar soluciones tecnológicas para la medición de la calidad del aire en contextos locales.

De manera similar, (Kumar, Morawska, & Martuzzi, 2022) proponen un sistema de monitoreo de calidad del aire en tiempo real basado en tecnologías IoT, destacando la importancia de la recopilación continua de datos para comprender la variación de contaminantes atmosféricos en zonas urbanas. Los autores señalan que el acceso oportuno a esta información facilita la identificación de condiciones ambientales adversas y contribuye al diseño de estrategias preventivas.

Los estudios revisados demuestran que el uso de sistemas de monitoreo ambiental en entornos urbanos constituye una herramienta eficaz para la observación y análisis de la calidad del aire. Sin embargo, muchas de estas propuestas se centran en plataformas web o soluciones de alcance general, lo que limita su adaptación a contextos institucionales específicos y a escenarios que requieren acceso móvil y visualización directa de la información.

2.1.2 Uso de tecnologías IoT en el monitoreo ambiental

El uso de tecnologías basadas en Internet de las Cosas (IoT) se ha consolidado como una alternativa eficiente para el monitoreo ambiental, debido a su capacidad para integrar sensores, dispositivos de procesamiento y mecanismos de comunicación en sistemas distribuidos. Este enfoque ha permitido la recopilación continua de datos ambientales, facilitando su transmisión y disponibilidad para distintos tipos de usuarios.

En este sentido, (Gubbi, Buyya, Marusic, & Palaniswai, 2013) describen una arquitectura IoT orientada a la interconexión de dispositivos físicos y plataformas de procesamiento, destacando el papel de los sensores como elementos fundamentales para la adquisición de información del entorno. Los autores resaltan que este tipo de arquitectura permite escalar sistemas de monitoreo y adaptarlos a diferentes escenarios, incluyendo aplicaciones ambientales.

De igual forma, (Kumar, Morawska, & Martuzzi, 2022) analizan la implementación de sistemas IoT aplicados al monitoreo ambiental, enfatizando la importancia de la transmisión de datos en tiempo real para la observación del comportamiento de variables atmosféricas. El estudio señala que la integración de sensores con plataformas IoT contribuye a mejorar la precisión del análisis ambiental y a fortalecer la toma de decisiones basada en datos.

Los antecedentes revisados evidencian que las tecnologías IoT ofrecen una base sólida para el desarrollo de sistemas de monitoreo ambiental, al permitir la automatización de la recolección de datos y su disponibilidad mediante plataformas digitales. No obstante, muchos de estos sistemas se orientan a soluciones generales, lo que abre la posibilidad de adaptar este enfoque a contextos específicos, como el monitoreo ambiental en entornos universitarios.

2.1.3 Plataformas digitales para la visualización de datos ambientales

La visualización de datos ambientales constituye un componente clave dentro de los sistemas de monitoreo, ya que permite interpretar la información recolectada por los sensores y facilitar su comprensión por parte de los usuarios. En los últimos años, diversas investigaciones han desarrollado plataformas digitales orientadas a presentar datos ambientales mediante interfaces gráficas, tableros de control y aplicaciones web.

En este contexto, (Segura, 2016) proponen una plataforma web para la visualización de datos ambientales obtenidos a partir de sensores distribuidos, destacando el uso de gráficos en tiempo real y el acceso remoto a la información almacenada. Este tipo de soluciones permite centralizar los datos y facilita su consulta desde distintos dispositivos, principalmente a través de navegadores web.

De forma similar, otras investigaciones han implementado dashboards ambientales integrados a servicios en la nube, los cuales permiten almacenar grandes volúmenes de información y representar la evolución temporal de variables atmosféricas mediante gráficos y reportes históricos. Estos sistemas han demostrado ser útiles para el análisis ambiental y el apoyo a la toma de decisiones en contextos urbanos y organizacionales.

No obstante, gran parte de estas plataformas se encuentran orientadas a la visualización web y requieren el acceso a un navegador para consultar la información, lo que puede limitar su uso en escenarios de movilidad. Esta característica reduce la inmediatez con la que los usuarios pueden acceder a los datos ambientales, especialmente en espacios donde se requiere una consulta rápida y directa, como entornos educativos o institucionales (Rodríguez Ortiz, 2024).

2.1.4 Aplicaciones móviles orientadas al monitoreo ambiental

El uso de aplicaciones móviles para el monitoreo ambiental ha cobrado relevancia debido a la necesidad de acceder a información de manera inmediata y desde distintos espacios. Estas aplicaciones permiten a los usuarios consultar datos sobre la calidad del aire mediante interfaces adaptadas a dispositivos móviles, facilitando la interpretación de la información en escenarios de movilidad.

Diversos estudios han señalado que las aplicaciones móviles orientadas al monitoreo ambiental contribuyen a mejorar el acceso a datos atmosféricos y a fortalecer la conciencia ambiental de los usuarios, al presentar la información de forma clara y comprensible (UNESCO, 2022). En muchos casos, estas aplicaciones se apoyan en plataformas centralizadas que recopilan datos provenientes de estaciones de monitoreo distribuidas a nivel regional o nacional.

No obstante, una limitación recurrente de estas soluciones es que suelen basarse en estimaciones globales o regionales, sin integrar datos generados por sensores instalados en ubicaciones específicas. Esta característica reduce su capacidad para reflejar con precisión las condiciones ambientales de espacios particulares, como campus universitarios u otras áreas institucionales.

En este sentido, los antecedentes revisados evidencian que, si bien las aplicaciones móviles representan una herramienta eficaz para la consulta de información ambiental, existe una oportunidad para desarrollar soluciones adaptadas a contextos locales que permitan integrar mediciones propias y ofrecer una visualización directa de los datos generados en el entorno de estudio.

2.1.5 Tecnologías de comunicación para sistemas de monitoreo en tiempo real

Los sistemas de monitoreo ambiental requieren mecanismos de comunicación que permitan la transmisión eficiente de datos desde los sensores hacia las plataformas de procesamiento y visualización. En investigaciones recientes, se ha destacado la importancia de emplear tecnologías de comunicación capaces de soportar la transferencia continua de información, especialmente en aplicaciones donde se requiere acceso a datos en tiempo real.

Diversos estudios han analizado el uso de protocolos de comunicación ligeros en sistemas de monitoreo basados en IoT, resaltando su capacidad para operar en entornos con recursos limitados y redes de comunicación variables. Estos protocolos permiten reducir el consumo de ancho de banda y mejorar la estabilidad en la transmisión de datos ambientales, aspectos clave en aplicaciones distribuidas (Pincay Ponce & García Silva, 2024).

En trabajos orientados al monitoreo ambiental en tiempo real se señala que la comunicación eficiente entre dispositivos sensores y plataformas de análisis contribuye a una mejor respuesta ante cambios en las condiciones atmosféricas. La disponibilidad inmediata de los datos facilita su interpretación y permite apoyar procesos de análisis continuo en contextos urbanos e institucionales (Kumar, Morawska, & Martuzzi, 2022).

Los antecedentes revisados evidencian que la selección de tecnologías de comunicación adecuadas constituye un componente fundamental en el diseño de sistemas de monitoreo ambiental en tiempo real, ya que influye directamente en la confiabilidad, disponibilidad y utilidad de la información generada por los sensores.

2.2 Bases teóricas

2.2.1 *Internet de las cosas*

El concepto de Internet de las Cosas (IoT) se refiere a la interconexión de dispositivos físicos capaces de recopilar, procesar y transmitir información a través de redes digitales, permitiendo automatización, análisis y toma de decisiones basadas en datos. Aunque la idea tecnológica detrás del IoT comenzó a explorarse desde las primeras etapas del internet y la computación embebida, el término *Internet of Things* fue introducido formalmente en 1999 por Kevin Ashton, durante su trabajo en el Auto-ID Center del Massachusetts Institute of Technology (MIT), al relacionar sistemas de identificación por radiofrecuencia con plataformas de internet para mejorar procesos logísticos (González, 2023).

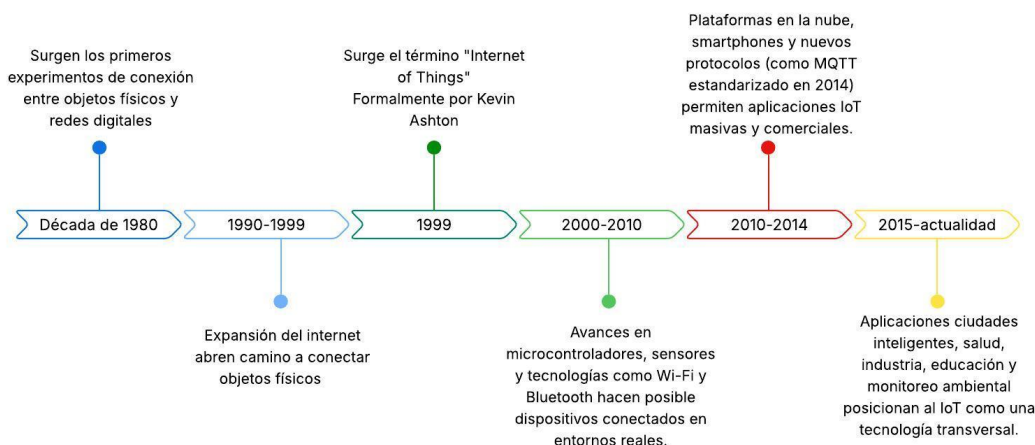
El principio del IoT tienen rastros desde la década de 1980, cuando surgieron los primeros dispositivos conectados a redes fuera del contexto computacional tradicional, uno de los pioneros en implementarlo para esos tiempos fue la reconocida empresa de Coca-Cola, con su máquina expendedora conectada a una red de la universidad Carnegie Mellon en el 1982, esta máquina enviaba información del inventario y temperatura mediante un sistema computacional remoto (Campbell, Aspray, Yost, Tinn, & Díaz, 2023). A partir de estos experimentos iniciales, la evolución tecnológica, junto con el crecimiento y desarrollo de microcontroladores, redes inalámbricas, sensores y protocolos ligeros, contribuyeron a la materialización del IoT como una tecnología global.

Para el 2010, el IoT experimentó un crecimiento significativo, el cloud computing empezó a tomar fuerza, empezaron a surgir nuevos dispositivos móviles inteligentes y la reducción de los costos en hardware de electrónica embebida (Gubbi, Buyya, Marusic, & Palaniswai, 2013). También dieron inicios las arquitecturas estandarizadas, plataformas de integración de datos y protocolos optimizados para la comunicación en entornos con recursos limitados, como lo es MQTT, IBM fue quien lo desarrolló inicialmente en 1999 y más tarde adoptado como estándar por OASIS en 2014 para entornos IoT (Alipour, Wang, & Krishnamurthy, 2021).

En la actualidad, el Internet de las Cosas se hace presente en aplicaciones para diversos sectores: ciudades inteligentes, infraestructura industrial (IIoT), salud digital, agricultura de precisión, transporte inteligente y monitoreo ambiental (Alay, Mendez, & Erazo, 2023). En el contexto educativo, universidades y centros académicos han implementado soluciones IoT orientadas al control energético, seguridad, automatización de espacios y monitoreo ambiental, sirviendo como plataformas reales de innovación aplicada, investigación y formación profesional.

Figura 3

Origen y evolución del Internet de las Cosas (IoT)



Nota. Gráfico de línea de tiempo elaborada en LucidChart. Elaboración propia

La figura 3 muestra la línea de tiempo de la evolución del Internet de las cosas tomando como referencia la visión histórica descrita por (Gubbi, Buyya, Marusic, & Palaniswai, 2013), complementada con hitos tecnológicos ampliamente documentados en el desarrollo del IoT.

El uso de IoT en el monitoreo ambiental ha permitido desarrollar sistemas distribuidos capaces de operar de forma continua, con bajo consumo de recursos y alta escalabilidad. Estas características hacen que el enfoque IoT sea adecuado para escenarios donde se requiere supervisar variables ambientales en tiempo real y acceder a la información desde diferentes plataformas.

2.2.2 Sensores ambientales

Los sensores ambientales son dispositivos diseñados para medir variables físicas y químicas del entorno, permitiendo la recopilación de datos que describen las condiciones ambientales de un espacio determinado. En sistemas de monitoreo, estos sensores constituyen el punto de partida para la adquisición de información, ya que transforman magnitudes ambientales en señales que pueden ser procesadas por sistemas digitales.

En el ámbito del monitoreo de la calidad del aire, se emplean sensores capaces de detectar concentraciones de gases contaminantes, como dióxido de carbono, monóxido de carbono, dióxido de nitrógeno y ozono, así como sensores destinados a la medición de material particulado, principalmente PM_{2.5} y PM₁₀ (Catiblanco Avedaño & Cañón Alfonso, 2019). De manera complementaria, también se utilizan sensores climáticos para registrar variables como

temperatura y humedad, las cuales influyen en el comportamiento de los contaminantes atmosféricos.

La integración de sensores ambientales en sistemas basados en IoT permite la recolección continua de datos y su posterior transmisión hacia plataformas de procesamiento y visualización. Esta característica facilita el análisis del comportamiento ambiental a lo largo del tiempo y posibilita la detección de variaciones que pueden resultar relevantes para la evaluación de la calidad del aire en contextos urbanos o institucionales.

2.2.3 Calidad del aire

La calidad del aire se refiere al estado de la atmósfera en relación con la presencia y concentración de contaminantes que pueden afectar la salud humana, los ecosistemas y el entorno en general. La evaluación de la calidad del aire se basa en la medición de distintos contaminantes atmosféricos y en la comparación de sus concentraciones con valores de referencia establecidos por organismos especializados.

Según (Moreira Romero, 2018), entre los principales contaminantes del aire se encuentran los gases como el dióxido de nitrógeno (NO_2), el dióxido de azufre (SO_2), el monóxido de carbono (CO) y el ozono (O_3), así como el material particulado, especialmente las fracciones $\text{PM}_{2.5}$ y PM_{10} . Estas partículas, debido a su tamaño reducido, pueden permanecer suspendidas en el aire e ingresar al sistema respiratorio, lo que las convierte en uno de los contaminantes de mayor interés en estudios de calidad del aire.

Figura 4

Niveles recomendados de las directrices sobre la calidad del aire y metas intermedias

Contaminante	Tiempo promedio	Meta intermedia				Nivel de las directrices sobre la calidad del aire
		1	2	3	4	
$\text{MP}_{2.5}, \mu\text{g}/\text{m}^3$	Anual	35	25	15	10	5
	24 horas ^a	75	50	37,5	25	15
$\text{MP}_{10}, \mu\text{g}/\text{m}^3$	Anual	70	50	30	20	15
	24 horas ^a	150	100	75	50	45
$\text{O}_3, \mu\text{g}/\text{m}^3$	Temporada alta ^b	100	70	–	–	60
	8 horas ^a	160	120	–	–	100
$\text{NO}_2, \mu\text{g}/\text{m}^3$	Anual	40	30	20	–	10
	24 horas ^a	120	50	–	–	25
$\text{SO}_2, \mu\text{g}/\text{m}^3$	24 horas ^a	125	50	–	–	40
$\text{CO}, \text{mg}/\text{m}^3$	24 horas ^a	7	–	–	–	4

Nota. Imagen propiedad de (OMS, 2024), donde presentan los principales contaminantes del aire y sus niveles

Como se puede apreciar en la figura 4 y lo aseguran organismos internacionales de salud afirmando que la exposición prolongada a concentraciones elevadas de contaminantes atmosféricos puede generar efectos adversos en la salud, tales como afecciones respiratorias y cardiovasculares, además de agravar enfermedades preexistentes (OMS, 2024). Por esta razón, el monitoreo continuo de la calidad del aire constituye una herramienta importante para identificar posibles riesgos y apoyar la adopción de medidas preventivas en distintos contextos, incluidos los entornos educativos.

2.2.4 Protocolos de comunicación

En el ecosistema del Internet de las Cosas, la comunicación es uno de los pilares fundamentales, siendo esta quien permite que los dispositivos, sensores, plataformas y usuarios intercambien información de forma eficiente, segura y en tiempo real. Según (Alay, Mendez, & Erazo, 2023), la comunicación en IoT se sustenta en una arquitectura conformada por sensores, redes de comunicación, plataformas middleware y aplicaciones finales, donde se utilizan protocolos específicos para cada capa según el requerimiento energético, alcance, velocidad de transmisión e interoperabilidad del sistema. Siendo este modelo comunicacional el que ha permitido que los dispositivos conectados evolucionen desde sistemas aislados hasta redes

inteligentes con capacidades de análisis e incluso capaces de automatizar la toma de decisiones.

2.2.4.1 Tecnologías de comunicación inalámbricas para IoT

Según (Mora Olivedo, Macías Lara, Rodríguez Vizute, & Sacón Klinger, 2021), indican que, a lo largo de los años, diferentes tecnologías inalámbricas se han consolidado como medios esenciales para la comunicación en IoT. Protocolos como Wi-Fi, Bluetooth Low Energy (BLE), Zigbee, LoRaWAN y 5G han sido adoptados en función de sus características técnicas y del contexto de aplicación, ejemplo, Wi-Fi destaca por su alta velocidad y disponibilidad en entornos educativos y urbanos, mientras que BLE y Zigbee se utilizan donde se requiere bajo consumo energético y redes de corto alcance. Por su parte, LoRaWAN y redes móviles como 4G/5G responden a necesidades de gran cobertura e infraestructura distribuida, siendo especialmente útiles en proyectos de ciudades inteligentes y monitoreo ambiental remoto.

La diversidad de tecnologías de comunicación disponibles responde a la necesidad de cubrir distintos escenarios de operación, desde redes de corto alcance y bajo consumo energético hasta

sistemas de transmisión de datos a gran escala. Cada tecnología presenta características particulares en cuanto a alcance, velocidad, estabilidad, costo y requisitos de infraestructura (Greer, 2023), lo que influye directamente en su adopción dentro de proyectos específicos. En la Tabla 1 se presentan algunas de las tecnologías de comunicación más utilizadas en el ámbito del Internet de las Cosas, permitiendo comparar sus principales atributos y comprender su adecuación según el tipo de aplicación, el entorno y las necesidades técnicas del sistema.

Tabla 1

Características de tecnologías de comunicación en IoT

Tecnología	Alcance aproximado	Consumo energético	Velocidad de transmisión	Topología	Costo de Implementación	Aplicaciones
Wi-fi	50-100 m	Alto	54 Mbps – 600 Mbps	Estrella	Medio	Hogares inteligentes, laboratorios, campus, aplicaciones que requieren ancho de banda alto
Bluetooth Low Energy (BLE)	10–50 m	Muy bajo	1 Mbps – 2 Mbps	Estrella / Malla (BLE 5)	Bajo	Wearables, sensores personales, dispositivos móviles
Zigbee	10–100 m	Bajo	250 Kbps	Malla	Bajo	Domótica, automatización industrial, redes sensoriales
LoRaWAN	2–15 km urbano / 15–40 km rural	Muy bajo	0.3–50 Kbps	Estrella	Medio–Alto	Monitoreo ambiental, agricultura inteligente, ciudades inteligentes
NB-IoT	1–10 km	Muy bajo	200 Kbps – 250 Kbps	Celular	Medio–Alto (depende de operador)	Sensores remotos, medidores inteligentes, infraestructura pública

5G IoT	1–5 km	Variable	1–20 Gbps	Celular	Alto	Telemedicina, vehículos inteligentes, aplicaciones críticas de baja latencia
RFID/NFC	10 cm – 10 m	Muy bajo	Bajo	Punto a punto	Bajo	Control de acceso, inventario, identificación
Ethernet (IoT industrial)	Cableado	Bajo	100 Mbps – 10 Gbps	Estrella	Medio–Alto	Industria 4.0, automatización y procesamiento intensivo de datos

Nota. Tabla comparativa elaborada con información del sitio de (Greer, 2023)

2.2.4.2 *Protocolos de comunicación en IoT*

En IoT los protocolos de comunicación forman parte de la base que garantiza que los dispositivos puedan intercambiar datos sin importar los contextos de uso, a diferencia de las tecnologías de comunicación, que se enfocan en el medio físico o inalámbrico por el cual viaja la información, los protocolos establecen reglas, formatos y mecanismos que permiten que los datos se transmitan correctamente entre sensores, plataformas y aplicaciones.

En el caso de aplicaciones de IoT orientadas al monitoreo ambiental, la comunicación no solo se limita a la conectividad entre sensores y dispositivos finales, sino que también se debe considerar la transmisión eficiente hacia las plataformas locales o en la nube.

Por otra parte (Guananga Narváez & Jaya Duche, 2022), mencionan que, la evolución en cuanto a protocolos para comunicación de aplicaciones de IoT ha permitido hoy por hoy sea más robusta, flexible y accesible, facilitando la creación de aplicaciones para cualquier tipo de enfoque. Gracias a las capacidades de estas comunicaciones, el IoT ha dejado de ser una tecnología emergente para convertirse en un componente clave dentro de la infraestructura digital moderna.

En la tabla 2, podemos analizar algunos protocolos que son usados para aplicaciones de IoT, los cuales han sido diseñados para responder a diferentes necesidades de comunicación según las capacidades del dispositivo, el entorno de despliegue y los requisitos de la aplicación. Cada protocolo posee características particulares en cuanto a su tipo, modelo de intercambio de datos, consumo energético, nivel de seguridad y velocidad de transmisión, factores que

influyen directamente en su desempeño dentro de sistemas distribuidos. Esta comparación permite comprender mejor las ventajas y limitaciones de cada alternativa, y facilita la selección del protocolo más adecuado para escenarios específicos, como monitoreo ambiental, control industrial, automatización del hogar o transmisión de datos en tiempo real.

Tabla 2

Características de protocolos de comunicación en IoT

Protocolo	Tipo	Modelo de comunicación	Consumo de recursos	Seguridad	Velocidad / desempeño	Escenario Ideal
MQTT	Protocolo de mensajería	Publish–Subscribe (Broker)	Muy bajo	Medio–Alto (TLS opcional)	Excelente para envío frecuente y ligero de datos	Sensores, monitoreo ambiental, IoT industrial, campus inteligentes
HTTP/HTTPS	Protocolo web	Cliente–Servidor	Alto	Alto	Alto, depende de conexión	Aplicaciones web, APIs, dispositivos con buena energía
CoAP	Protocolo optimizado IoT	Cliente–Servidor (UDP)	Bajo	Medio	Alto desempeño en redes inestables	Redes de sensores, LoRaWAN, NB-IoT
WebSockets	Conexión persistente bidireccional	Cliente–Servidor	Medio	Alto	Muy alta para tiempo real	Apps en tiempo real: mensajería, dashboards, IoT interactivo
AMQP	Protocolo de mensajería avanzado	Publish–Subscribe	Alto	Alto	Excelente pero pesado	Aplicaciones críticas, financieras, empresariales
DDS (Data Distribution Service)	Middleware distribuido	Peer-to-peer	Medio–Alto	Muy alto	Ultra bajo retardo	Vehículos autónomos, robótica, IoT crítico

Nota. Tabla comparativa de protocolos de comunicación basados en información de (Guananga Narváez & Jaya Duche, 2022)

2.2.4.3 MQTT como protocolo principal

El protocolo MQTT (Message Queuing Telemetry Transport), se han convertido en estándares para estos tipos de sistemas, especialmente en escenarios donde se necesita la transmisión continua de datos, bajos consumos energéticos y tolerancia a conexiones inestables. Fue desarrollado por Andy Standford Clark (IBM) y Arlen Nipper (Arcom) en 1999 con el objetivo de permitir la comunicación entre sensores de petróleo ubicados en zonas remotas mediante enlaces satelitales de baja velocidad. Convirtiéndose con el tiempo en los pilares del IoT moderno gracias a su diseño ligero y eficiente. (Alipour, Wang, & Krishnamurthy, 2021)

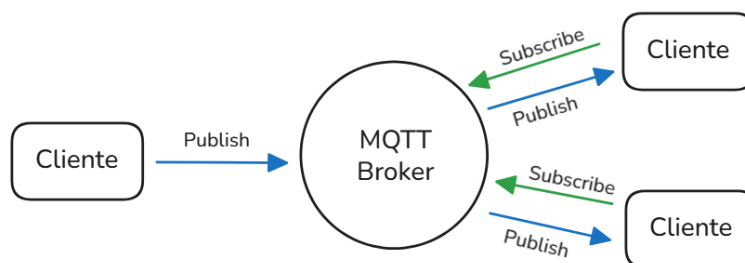
2.2.4.4 Modelo Publish/Subscribe

A diferencia de otros protocolos orientados a la comunicación directa entre cliente y servidor, MQTT implementa un modelo de mensajería Publish/Subscribe, donde existe un intermediario conocido como bróker entre los dispositivos. El flujo con este modelo funciona de la siguiente manera; un cliente puede publicar mensajes sobre un “mensaje” (topic), mientras que otros clientes pueden suscribirse a ese mismo tema para recibir actualizaciones en tiempo real. (Alipour, Wang, & Krishnamurthy, 2021)

Esta arquitectura desacopla el envío y recepción de información, permitiendo que los dispositivos no necesiten conocerse entre sí, lo que hace que mejore la escalabilidad y reduce notoriamente la complejidad del sistema o entorno donde se lo aplique.

Figura 5

Modelo de publish/subscribe



Nota. Ilustración recreada con el flujo como lo explica Alipour

En la figura 4 se muestra el flujo del modelo publish/subscribe basado en como lo explica (Alipour, Wang, & Krishnamurthy, 2021) donde un cliente publica y el intermediario o bróker es quien permite ser replicada la publicación para que otros clientes puedan suscribirse al mismo y puedan ver la información en tiempo real.

2.2.5 Plataformas de almacenamiento y procesamiento

Las plataformas de procesamiento y almacenamiento desempeñan un papel central en los sistemas de monitoreo ambiental, ya que permiten recibir, procesar y conservar la información generada por los sensores a lo largo del tiempo. En este contexto, la computación en la nube se ha consolidado como una solución que facilita el acceso remoto a los datos, la escalabilidad de los sistemas y la disponibilidad continua de la información (Moreira Sánchez, Cruz Felipe, Coterá Ramírez, & Medranda Cobeña, 2025).

La computación en la nube permite centralizar el procesamiento de datos provenientes de múltiples dispositivos, evitando la dependencia de infraestructura local compleja. Este enfoque resulta especialmente útil en sistemas de monitoreo ambiental, donde se generan grandes volúmenes de datos que deben ser almacenados y consultados de manera eficiente desde diferentes plataformas.

El almacenamiento de datos históricos constituye un componente fundamental para el análisis de la calidad del aire, ya que posibilita la observación de tendencias, la comparación de registros en distintos periodos y la identificación de variaciones en el comportamiento de los contaminantes. Disponer de registros históricos facilita la elaboración de reportes, el apoyo a procesos de investigación y la toma de decisiones basadas en información acumulada a lo largo del tiempo.

2.2.5.1 Bases de Datos

Como parte de los componentes que conforman las aplicaciones, encontramos las bases de datos, son quienes responden a la necesidad de manejar información dinámica que vienen desde distintas fuentes, como servicios externos, sistemas conectados o funciones con las que el usuario interactúa. Las bases de datos son las que permiten almacenar las configuraciones, registros históricos, resultados de algún flujo de procesos, acceso a datos temporales, todo esto para garantizar al usuario la experiencia de consistencia y persistencia de información.

Va a depender de la perspectiva arquitectónica la forma en que las bases de datos puedan operar, ya que existe la forma local, remota e incluso se pueden usar las dos formas, esto va a depender de los requerimientos de la aplicación. La elección de los modos de operar, permite

la flexibilidad de diseñar aplicaciones que sean escalables y adaptables, capaces de funcionar con o sin conexión a internet (Mercado, 2023)

En cuanto a aplicaciones de IoT nos referimos, las bases de datos cumplen un rol estratégico al servir como almacén de información para evitar pérdidas de datos ante fallos de interrupciones y dependiendo de las configuraciones de la aplicación no existiría pérdida de conectividad.

SQLite

SQLite es una base de datos relacional de código abierto, embebido que está diseñado para trabajar de forma autónoma sin requerir un servidor (Grant & Owens, 2010). Se diferencia de los demás gestores de bases de datos ya que almacena toda la información en un archivo local, facilitando la implementación, portabilidad y mantenimiento en aplicaciones móviles o en sistemas que cuentan con bajos recursos. Esta característica es que lo convierte en la mejor opción en las aplicaciones móviles por el bajo consumo de memoria y rapidez, tanto así que prácticamente es un estándar para el almacenamiento local en las aplicaciones móviles sin importar el sistema operativo.

Para este proyecto, se usa esta base de datos en el módulo de Raspberry pi como almacenamiento local para luego mandarlo a otros servicios. Integrándose con facilidad con los recursos que presta este pequeño computador sin trabajos forzados ni pesados.

PostgreSQL

Este gestor de base de datos relacional es una de los más conocidos y usados, su característica principal es que es de tipo relacional, orientada a objetos, y de código abierto.

Datos de (Momjiam, 2004) destacan que su fuerte es el soporte para transacciones ACID (Atomicidad, consistencia, aislamiento y durabilidad). Lo que asegura la fiabilidad de la información almacenada incluso si se presentan fallos en el sistema.

En sistemas de monitoreo ambiental, para ser más específico en IoT, resulta ser la mejor opción ya que por las funciones que presta, existe la opción de que mediante las configuraciones podamos detectar cuando los sensores dejan de leer datos, y definir parámetros para que se envíen las alertas cuando existan escenarios que se consideren riesgosos.

Además de que, por su capacidad de manejar datos estructurales y temporales, facilita la generación de reportes, análisis de tendencias y apoyo a la toma de decisiones basadas en evidencias.

2.2.5.2 *Tecnologías de desarrollo de software*

JavaScript

Es uno de los lenguajes considerado como ligeros, de alto nivel, interpretado y orientado a eventos, su principal uso se encuentra en las aplicaciones web y móviles, durante mucho tiempo se ha considerado una de las tecnologías más importantes dentro del ecosistema de desarrollo de aplicaciones modernas incluyendo soluciones basadas en internet de las cosas (Pacheco, Escobar, & Trujillo, 2021).

Entre las ventajas que presenta JavaScript es la capacidad para integrarse con frameworks y librerías que facilitan el desarrollo frontend o interfaces de usuarios dinámicas e interactivas, además de total compatibilidad con entornos multiplataformas.

En cuanto al desarrollo de aplicaciones móviles, permite construir interfaces capaces de consumir datos en tiempo real provenientes de otros servicios que no se encuentran en el desarrollo o de sistemas de mensajería.

Otro de los puntos a favor que tiene JavaScript es la facilidad para manejar la asincronía, lo que presenta alta ventaja en aplicaciones IoT donde los datos llegan de manera continua y no de distintas fuentes.

En este proyecto, se emplea el uso de este lenguaje de programación con la Librería React Native misma que permite el desarrollo de aplicaciones móviles híbridas junto con el uso de otros módulos y servicios para consolidar una aplicación en condiciones para el monitoreo de la calidad del aire.

Python

Python es un lenguaje de programación de propósito general, considerado de alto nivel, actualmente utilizado para proyectos grandes como lo es la Inteligencia Artificial, minería de datos y aplicaciones comunes, su principal enfoque está en el backend, con su sintaxis legible y clara permite al desarrollador y al software escalar de forma que no tenga límites (Paguay, Paucar, & Flores, 2019)

En el contexto de Internet de las cosas, toma la posición del lenguaje más utilizado para el desarrollo de servicios que interactúan con sensores, dispositivos embebidos y sistemas de mensajerías.

En este proyecto se integra perfectamente ya que la compatibilidad con plataformas como Raspberry Pi y su soporte con MQTT permiten la implementación y resultados esperados para el procesamiento y envíos de datos en tiempo real

API REST

Las Interfaces de Programación de Aplicaciones, conocidas como API, permiten la comunicación entre diferentes sistemas de software mediante un conjunto de reglas y definiciones previamente establecidas. En particular, las API de tipo REST (Representational State Transfer) se basan en un estilo arquitectónico que facilita el intercambio de información a través de servicios web, utilizando protocolos estándar de comunicación (Amoedo, 2013). Este enfoque promueve la interoperabilidad entre aplicaciones, independientemente del lenguaje de programación o la plataforma utilizada.

Las API REST se caracterizan por su simplicidad y flexibilidad, ya que emplean el protocolo HTTP para realizar operaciones sobre recursos identificados mediante URLs. Estas operaciones suelen representarse a través de métodos estándar como GET, POST, PUT y DELETE, los cuales permiten consultar, enviar, actualizar o eliminar información (IBM, 2025). Asimismo, el uso de formatos de intercambio de datos como JSON o XML favorece la transmisión de información de manera estructurada y comprensible tanto para sistemas como para desarrolladores.

El uso de API REST resulta especialmente adecuado en sistemas distribuidos, donde es necesario separar la lógica de procesamiento de datos de las interfaces de visualización (Red Hat, 2023). Esta arquitectura facilita la escalabilidad, el mantenimiento y la reutilización de los servicios, permitiendo que distintas aplicaciones, como plataformas web o móviles, accedan a la misma fuente de información. Por estas razones, las API REST se han consolidado como una solución ampliamente adoptada en el desarrollo de aplicaciones modernas y sistemas que requieren comunicación eficiente entre múltiples componentes.

Frameworks para desarrollo backend

Los frameworks para desarrollo backend proporcionan una estructura organizada que facilita la construcción de aplicaciones del lado del servidor. Estas herramientas ofrecen componentes y convenciones que permiten gestionar de forma eficiente aspectos como el manejo de solicitudes, la lógica de negocio, la validación de datos y la comunicación con bases

de datos. Al apoyarse en un framework, los desarrolladores pueden concentrarse en la funcionalidad del sistema, reduciendo la necesidad de implementar soluciones desde cero.

En el contexto de sistemas basados en servicios y arquitecturas distribuidas, los frameworks backend suelen incorporar soporte nativo para la creación de API, la gestión de rutas y el procesamiento de solicitudes HTTP (MDN, 2025). Este enfoque favorece la separación de responsabilidades dentro del sistema y permite que distintas aplicaciones, como plataformas web o móviles, interactúen con los servicios de manera estructurada y controlada. Además, muchos frameworks incluyen mecanismos de seguridad, control de errores y escalabilidad que resultan esenciales en aplicaciones modernas.

El uso de frameworks backend contribuye a la mantenibilidad y extensibilidad de los sistemas de software, ya que promueve el uso de buenas prácticas de desarrollo y patrones arquitectónicos ampliamente aceptados. Estas características los convierten en una opción adecuada para aplicaciones que requieren crecimiento progresivo, integración con múltiples servicios y adaptación a diferentes entornos tecnológicos.

2.2.6 Desarrollo de aplicaciones móviles

Aplicaciones móviles híbridas

Dentro del ecosistema digital moderno, las aplicaciones móviles forman parte de los avances significativos, convirtiéndose en el principal medio de interacción entre los usuarios y servicios tecnológicos. De acuerdo con (Acosta, León, & Sanafria, 2022), una aplicación móvil es un sistema de software diseñado para operar en dispositivos con recursos limitados, como son los teléfonos inteligentes y tabletas, repotenciando la optimización, usabilidad, rendimiento y la interacción que tiene el usuario con el dispositivo. Los lanzamientos de las primeras tiendas de aplicaciones fueron en el 2008, dando así el paso a la distribución masiva del software, dejando abierto el camino al crecimiento acelerado en la gran variedad de aplicaciones, alcance y complejidad funcional que brindan hoy por hoy las aplicaciones.

Las primeras aplicaciones solían ser sencillas, eran herramientas simples de propósitos específicos, luego fueron evolucionando, se presentaban las funciones como conexión a plataformas con servicios en la nube y sistemas distribuidos. En la actualidad incorporan arquitecturas híbridas y nativas, patrones de diseños y mecanismos de comunicación remota mediante APIs REST, WebSockets o protocolos IoT, permitiendo que trasciendan sus

funciones, dejando de ser solo funciones de consumo de información y convertirse en herramientas fuertes en áreas como la educación, salud, seguridad, transporte, productividad, automatización y todas las áreas que nos podamos imaginar. Siendo las aplicaciones el enlace más accesible entre los usuarios y el entorno digital, especialmente en sistemas donde se requiere acceso a la información en tiempo real o interacción inmediata.

Visualización de datos ambientales

No es sorpresa que los datos son el eje principal de todo tema y en especial los datos medioambientales, de hecho son factor clave en los sistemas de monitoreo basados en IoT, permitiendo transformar grandes volúmenes de datos numéricos en información comprensible y accesible para usuarios finales. De hecho la representación de estos datos en forma de gráficos con las variantes ambientales, como temperatura, humedad, concentración de partículas o gases contaminantes facilita la interpretación y toma de decisiones en entornos educativos y urbanos.

Desde el enfoque tecnológico, la visualización de datos se apoya en técnicas de representación gráfica que incluyen gráficos de líneas, barras, indicadores, mapas de calor y dashboards interactivos. Estas herramientas permiten identificar tendencias, variaciones temporales y posibles anomalías en los datos recolectados, mejorando la comprensión del comportamiento ambiental en tiempo real o a lo largo del tiempo. En aplicaciones móviles, la visualización debe además considerar criterios de usabilidad, claridad y adaptación a distintos tamaños de pantalla, garantizando una experiencia intuitiva para el usuario (Llamocca, 2016).

En los sistemas de monitoreo ambiental, además de dar la función de visualizar, también puede tomarse de forma preventiva, mediante la representación de los niveles de calidad del aire, usando índices de colores, escalas o alertas, facilitando la identificación de condiciones desfavorables, así mismo con los datos históricos registrados ayudando al análisis comparativo y al apoyo de procesos de investigaciones futuras.

Las tecnologías móviles con el auge de su evolución y análisis de datos han permitido que la visualización de información ambiental sea cada vez más interactiva y accesible (Pascual, 2024). Por lo que se considera que la visualización de datos en sistemas de monitoreo ambiental sea elemento fundamental y sirva como puente entre datos técnicos generados por los sistemas mencionados y el conocimiento práctico que necesitan los usuarios finales.

Accesibilidad e interfaz de usuario

La accesibilidad y el diseño de la interfaz de usuario influyen directamente en la efectividad de una aplicación móvil, ya que determinan la forma en que los usuarios interactúan con la información presentada. Una interfaz bien diseñada facilita la comprensión de los datos, reduce la carga cognitiva y permite que usuarios con distintos niveles de experiencia tecnológica puedan utilizar la aplicación de manera eficiente (Ortiz Zambrano, Chavez Cujilan, Toapanta Bernabe, & Lino Castillo, 2017).

En aplicaciones que presentan información técnica o ambiental, la claridad visual, la organización de los elementos y el uso adecuado de colores y símbolos son aspectos fundamentales para garantizar una correcta interpretación de los datos. El diseño centrado en el usuario busca adaptar la interfaz a las necesidades reales de las personas, promoviendo una experiencia de uso intuitiva y funcional.

Desde una perspectiva académica, diversos estudios resaltan que la incorporación de principios de usabilidad y accesibilidad mejora la aceptación de las aplicaciones móviles y amplía su alcance, permitiendo que la información sea accesible para un mayor número de usuarios y contextos de uso.

Aplicaciones de monitoreo de calidad del aire

En términos generales un software de monitoreo de calidad del aire nos permite la visualización de datos en tiempo real, pero cuando nos encontramos con aplicaciones móviles para el monitoreo de la calidad del aire, damos un paso más, debido a que los dispositivos móviles casi siempre lo llevamos a todos lados, por lo que quiere decir que tenemos acceso a estos datos desde cualquier lugar que nos encontremos, este tipo de aplicaciones se caracterizan por consumir datos provenientes de sensores locales, estaciones de monitoreos o servicios en la nube, mismos datos que son procesados y presentados gráficamente, implementado índices estandarizados para la calidad del aire a nivel mundial, regional o local.

Muchas de las aplicaciones existentes están destinadas para contexto urbano o a gran escala, lo que quiere decir que los datos no son tan fiables dependiendo del punto donde nos encontremos; si queremos datos de un lugar en específico como por ejemplo una unidad educativa en particular, los datos proporcionados a nivel regional no son fiables ni precisos.

En la tabla 3 se presentan algunas aplicaciones móviles existentes para el monitoreo de la calidad del aire, considerando sus principales características, tipo de datos utilizados, alcance

de monitoreo y limitaciones. Esta comparación permite identificar tendencias comunes, carencias y habilidades, mismas que sirven para el análisis del enfoque tecnológico del proyecto.

Tabla 3

Aplicaciones móviles existentes para el monitoreo de la calidad del aire

Aplicación	Fuente de datos	Visualización en tiempo real	Historial de datos	Enfoque principal	Limitaciones
AirVisual	Estaciones públicas y sensores	Si	Si	Monitoreo urbano global	Dependencia de datos externos
Plume Labs (Plume Air Report)	Estaciones y modelos predictivos	Si	Parcial	Predicción de contaminación	No integra sensores propios
BreezoMeter	Estaciones oficiales y modelos IA	Si	Si	Calidad del aire y salud	Uso limitado sin suscripción
IQAir	Redes globales de monitoreo	Si	Si	Información ambiental global	Enfoque no local
Air Matters	APIs públicas	Si	Si	Salud y clima	Escasa personalización

Nota. Elaboración propia con información de reseñas de usuarios.

CAPÍTULO III

3 Marco investigativo

En este capítulo se presenta el marco investigativo que sustenta el desarrollo del proyecto, describiendo la metodología empleada, los métodos, técnicas e instrumentos de investigación utilizados, así como el enfoque aplicado para la construcción de la solución tecnológica propuesta, orientada al monitoreo de la calidad del aire mediante el uso de tecnologías IoT.

3.1 Metodología aplicada en la investigación

3.1.1 Tipo de Investigación

La presente investigación será de tipo aplicada, ya que se orienta a la solución de un problema real mediante el desarrollo de una aplicación para el monitoreo de la calidad del aire basado en tecnologías IoT. Este enfoque permite transformar los conceptos teóricos abordados en una solución tecnológica concreta, donde su diseño e implementación buscan generar resultados funcionales y verificables, validados a través de su desempeño y funcionamiento en un entorno real.

3.1.2 Niveles de Investigación

3.1.2.1 Investigación Descriptiva

Esta investigación presenta un nivel descriptivo, la esencia es describir las características del caso de estudio, así como el funcionamiento y comportamiento del sistema desarrollado, con este nivel también se detallan los componentes del sistema, la adquisición, forma de proceso y visualización de los datos destacando que para este nivel aún no se establecen relación causa efecto.

3.1.2.2 Investigación Exploratoria

Del mismo modo encontramos el nivel exploratorio, debido al contexto específico donde no existen soluciones ampliamente documentadas con características similares, este nivel permite explorar formas de integración entre sensores ambientales, comunicación IoT y aplicaciones móviles, con el fin de evaluar la viabilidad y utilidad como herramienta de monitoreo ambiental.

3.2 Métodos y técnicas de investigación

En esta sección, se presenta lo necesario para la recolección de información, antes del diseño y desarrollo de la aplicación, para lograr esto se utilizaron métodos y técnicas como:

Revisión documental: Como lo vimos en el Capítulo II, aplicando método analítico, se buscó en distintas fuentes bibliográficas de trabajos con temas similares al de este proyecto, se identificaron los que se asemejan un poco más, con el fin de captar mejores prácticas y enfocarlas en nuestro contexto de app para el monitoreo de la calidad del aire con tecnologías IoT.

Observación Directa: Se buscaron las aplicaciones existentes que cumplan las características de monitorear el aire y que tengan tecnologías IoT, con la finalidad de reconocer reseñas, basarnos en experiencias de usuarios (UX) ya existentes, mejorar en falencias, y sobre todo en adaptar lo que ya existe, pero en una app única y dedicada con sensores que van a estar dentro del campus de la ULEAM.

3.3 Metodología de desarrollo: Metodología ágil Scrum

Para el desarrollo de la app, se tomó la decisión de emplear la metodología ágil Scrum, debido a su enfoque iterativo e incremental, mismo que resulta adecuado para proyectos tecnológicos que requieren adaptación continua, validaciones al instante y mejoras constantes. Es aquí donde Scrum permite organizar el trabajo en ciclos de desarrollo relativamente cortos, facilitando la planificación, implementación y evaluación constante de cada funcionalidad de la aplicación.

Para aplicar Scrum se basó en la definición de sprints, donde se establecen objetivos claros y orientados al desarrollo de las funcionalidades específicas del sistema, como es la lectura de los datos desde los sensores, la comunicación mediante el protocolo MQTT, procesamiento de la información en el backend y la visualización de los datos en la aplicación móvil. Cada Sprint permitió obtener incrementos funcionales del sistema, estos fueron evaluados y ajustados con los resultados obtenidos.

De la mano de lo ya mencionado, se priorizaron tareas mediante un producto backlog, en el que se registraban los requerimientos y funcionalidades del proyecto, organizando según el nivel de importancia. Esta priorización permitió enfocar los esfuerzos en los componentes críticos del sistema y asegurar una evolución ordenada del desarrollo. La metodología Scrum facilitó la detección temprana de errores, la integración progresiva de los módulos y la validación continua del sistema, contribuyendo al cumplimiento de los objetivos planteados en la investigación.

3.3.1 Roles del Scrum

Durante el desarrollo del proyecto se consideraron los roles propuestos por la metodología Scrum, adaptándose al contexto académico y al alcance del trabajo. El rol de Product Owner fue desempeñado por el tutor de tesis, Ing. Willian Jesús Zamora Mero, PhD, quien orientó la definición y priorización de los requerimientos del sistema, asegurando que las funcionalidades desarrolladas se alineen con los objetivos académicos y con la solución del problema planteado.

El rol de Scrum Master fue asumido por el investigador, quien se encargó de facilitar el proceso de desarrollo, promover el cumplimiento de la metodología y organizar las actividades correspondientes a cada sprint. Así mismo, el equipo de desarrollo estuvo conformado por el propio investigador, siendo responsable del diseño, implementación, integración y pruebas de los distintos componentes del sistema, incluyendo el nodo IoT, el backend y la aplicación móvil. Esta asignación de roles es coherente con la naturaleza individual del proyecto y permite una correcta aplicación de los principios de la metodología Scrum.

Tabla 4

Definición de roles Scrum

Rol	Encargado	Responsabilidad
Product owner	Ing. Willian Jesús Zamora Mero, PhD (Tutor).	Definir y priorizar los requerimientos del sistema, orientar el alcance del proyecto y asegurar que las funcionalidades desarrolladas estén alineadas con los objetivos académicos y la solución del problema planteado.
Scrum Master	Tomy Arnol Anchundia Triviño (autor)	Facilitar el proceso de desarrollo bajo la metodología Scrum, organizar y dar seguimiento a los sprints, gestionar el cumplimiento de la metodología y apoyar en la resolución de impedimentos técnicos durante el desarrollo del proyecto.
Equipo de desarrollo	Tomy Arnol Anchundia Triviño (autor)	Diseñar, implementar, integrar y probar los distintos componentes del sistema, incluyendo el nodo IoT, el backend y la aplicación móvil, asegurando el correcto funcionamiento de la solución propuesta.

Nota. Elaboración propia

CAPÍTULO IV

4 Marco propositivo

A continuación, se expone la propuesta de solución tecnológica desarrollada para el monitoreo de la calidad del aire, detallando el diseño y funcionamiento del sistema implementado. Se describe la arquitectura general, los componentes que conforman la solución y la forma en que interactúan para la captura, transmisión, procesamiento y visualización de los datos ambientales, con el fin de dar respuesta al problema planteado y cumplir los objetivos establecidos en la investigación.

4.1 Descripción general de la solución propuesta

La solución propuesta consiste en el diseño e implementación de un sistema integral de monitoreo de la calidad del aire basado en tecnologías de Internet de las Cosas (IoT), orientado a la recolección, transmisión, almacenamiento y visualización de datos ambientales en tiempo real. Este sistema surge como respuesta a la necesidad de contar con información confiable y accesible sobre las condiciones del aire en un entorno determinado, permitiendo evaluar su comportamiento y apoyar la toma de decisiones relacionadas con el cuidado del ambiente y la salud.

El sistema está conformado por un nodo IoT encargado de la captura de datos ambientales a través de sensores especializados, los cuales permiten medir variables asociadas a la calidad del aire. Estos datos son procesados inicialmente a nivel del dispositivo y posteriormente transmitidos mediante una infraestructura de comunicación basada en el protocolo MQTT, seleccionado por su ligereza, eficiencia y adecuación para entornos IoT.

La información recolectada es enviada a un servidor donde se gestiona el almacenamiento y procesamiento de los datos, garantizando su disponibilidad para consultas posteriores y análisis histórico. Sobre esta base, se desarrolla una aplicación móvil que actúa como interfaz principal con el usuario, permitiendo la visualización de los datos de manera clara, intuitiva y organizada. A través de la aplicación, los usuarios pueden acceder a información actualizada sobre la calidad del aire, facilitando su interpretación mediante gráficos y valores representativos.

La arquitectura de la solución ha sido diseñada de forma modular, lo que permite la escalabilidad del sistema y la posibilidad de incorporar nuevos sensores, funcionalidades o mejoras futuras sin afectar su funcionamiento general. Asimismo, la propuesta contempla la

integración entre hardware, software y comunicaciones, asegurando un flujo continuo de información desde la captura de los datos hasta su visualización final.

En conjunto, la solución propuesta no solo demuestra la aplicación práctica de tecnologías IoT en el monitoreo ambiental, sino que también constituye una herramienta tecnológica funcional que valida los conceptos desarrollados a lo largo de la investigación, cumpliendo con los objetivos planteados y aportando una alternativa viable para el seguimiento de la calidad del aire en tiempo real.

4.2 Determinación de recursos del proyecto

La correcta ejecución de la propuesta planteada requiere la identificación y organización de los recursos necesarios, tanto humanos como tecnológicos. Estos recursos permiten garantizar el desarrollo, implementación y funcionamiento adecuado del sistema de monitoreo de la calidad del aire, asegurando su sostenibilidad y correcto desempeño dentro del contexto para el cual ha sido diseñado.

4.2.1 Recursos humanos

Los recursos humanos involucrados en el desarrollo de la presente propuesta corresponden a los actores responsables de la planificación, supervisión y ejecución del proyecto. Considerando que se trata de un trabajo de titulación de carácter académico, los roles definidos dentro de la metodología Scrum son asumidos de manera específica, manteniendo la correcta distribución de responsabilidades.

- **Autor del proyecto, Scrum Master y Equipo de desarrollo**

El autor del proyecto es el principal responsable del diseño, desarrollo, implementación y validación del sistema de monitoreo de la calidad del aire. Asimismo, asume el rol de Scrum Master, encargándose de aplicar la metodología Scrum, planificar las iteraciones de desarrollo, gestionar las tareas y asegurar el cumplimiento de los objetivos establecidos en cada etapa del proyecto.

Entre sus responsabilidades se incluyen el desarrollo de la aplicación móvil, la implementación de los servicios backend, la integración con los dispositivos IoT, el manejo de bases de datos y la realización de pruebas funcionales del sistema.

- **Tutor de Tesis, Product Owner**

El tutor de tesis Ing. Willian Jesús Zamora Mero, PhD, cumple el rol de Product Owner, siendo el responsable de orientar la visión general del proyecto y validar que la propuesta cumpla con los objetivos académicos, técnicos y metodológicos establecidos.

Dentro de sus funciones se encuentran la revisión y aprobación de los requerimientos del sistema, la retroalimentación sobre los avances del proyecto, la priorización de funcionalidades y la supervisión del correcto enfoque de la solución propuesta.

- **Usuarios Finales**

Como recurso humano complementario, se considera la participación indirecta de los usuarios finales, quienes aportan información relevante para la validación del funcionamiento del sistema, especialmente en la interpretación de los datos de calidad del aire y la usabilidad de la aplicación. Su participación permite evaluar el desempeño del sistema en un entorno real.

4.2.2 Recursos tecnológicos

Los recursos tecnológicos, tanto a nivel de hardware como de software, permitieron implementar de manera integral el sistema de monitoreo de la calidad del aire. La selección de estas tecnologías se realizó considerando criterios como disponibilidad, compatibilidad, escalabilidad y adecuación al contexto académico del proyecto.

En cuanto al hardware, se utilizó una Raspberry Pi como dispositivo principal para la adquisición y procesamiento inicial de los datos provenientes de los sensores ambientales. Este dispositivo permitió ejecutar servicios locales, almacenar información de manera temporal y gestionar la comunicación con el sistema central mediante el protocolo MQTT. Asimismo, se emplearon sensores de calidad del aire para la medición de variables como material particulado, temperatura, humedad y gases contaminantes.

Respecto a los recursos de software, se desarrollaron servicios en Python para la lectura, procesamiento y almacenamiento local de los datos en una base de datos SQLite, así como para su posterior envío hacia un broker MQTT. Para la gestión y visualización de la información, se implementó un backend desarrollado con NestJS, conectado a una base de datos PostgreSQL alojada en la nube, lo que permitió centralizar y consultar los datos históricos del sistema.

La aplicación móvil fue desarrollada utilizando React Native con Expo, permitiendo la visualización de los datos en tiempo real y el acceso a información histórica desde dispositivos Android. Además, se empleó el protocolo MQTT, mediante el broker Mosquitto, como

mecanismo principal de comunicación entre la Raspberry Pi, el backend y la aplicación, tanto en entornos locales como remotos.

Finalmente, para el despliegue del sistema se utilizó una máquina virtual en la nube, donde se configuraron los servicios necesarios para el funcionamiento del backend, incluyendo la gestión de entornos de desarrollo y producción, variables de entorno y medidas básicas de seguridad. El uso de estas tecnologías permitió construir una solución funcional, escalable y acorde a los objetivos planteados en el proyecto.

4.3 Etapas de desarrollo de la propuesta

El desarrollo de la propuesta se llevó a cabo mediante un conjunto de etapas organizadas, las cuales permitieron estructurar el trabajo de forma progresiva y ordenada. Estas etapas abarcan desde la planificación inicial del proyecto hasta la implementación y validación de la solución, considerando tanto los aspectos técnicos como metodológicos necesarios para el desarrollo de un sistema de monitoreo de la calidad del aire basado en tecnologías IoT.

4.3.1 Fase de planificación y definición del proyecto

Esta representó el inicio del proyecto y fue fundamental para establecer una visión clara de la solución que se buscaba desarrollar. Como estudiante y autor del proyecto, en esta etapa se analizó el problema relacionado con la necesidad de monitorear la calidad del aire de manera continua y accesible, identificando las limitaciones de las soluciones existentes y la oportunidad de aplicar tecnologías IoT como alternativa.

Durante la planificación se definió el alcance del sistema, estableciendo las principales funcionalidades que debía cumplir la propuesta, tales como la recolección de datos mediante sensores ambientales, el procesamiento de la información y su visualización a través de una aplicación móvil. Asimismo, se tomaron en cuenta las condiciones reales del entorno, como la disponibilidad de conexión a internet, lo que motivó la implementación de mecanismos de comunicación tanto locales como remotos.

Finalmente, se decidió aplicar la metodología Scrum para organizar el desarrollo del proyecto, ya que permitió dividir el trabajo en etapas manejables, priorizar actividades y realizar ajustes conforme avanzaba el desarrollo, facilitando así un mejor control del proceso y de los resultados obtenidos.

4.3.2 Identificación de stakeholders y actores del sistema

En esta etapa se identificaron los principales stakeholders involucrados en el desarrollo de la propuesta, considerando únicamente a las personas e instituciones que influyen directa o indirectamente en el proyecto. Este análisis permitió reconocer los roles, responsabilidades y niveles de participación de cada actor humano, facilitando la organización del trabajo y la validación de la solución desarrollada. Los componentes tecnológicos del sistema no se consideran como stakeholders, ya que su descripción y análisis se abordan posteriormente dentro del diseño arquitectónico de la solución.

Tabla 5

Identificación de Stakeholders y actores

Actor / Stakeholder	Tipo	Rol dentro del proyecto	Interacción con el sistema
Autor del proyecto	Interno	Desarrollo e implementación de la solución	Diseño, desarrollo, pruebas y mantenimiento
Tutor de tesis	Interno	Orientación académica y validación del proyecto	Revisión, retroalimentación y aprobación
Institución académica	Externo	Evaluación del proyecto de titulación	Validación académica
Usuarios finales	Externo	Uso y validación funcional del sistema	Consulta de datos y retroalimentación

Nota. Elaboración propia

4.3.3 Levantamiento y análisis de requisitos

Después de identificar los actores, se realizó el levantamiento y análisis de requisitos, mismo que permitió identificar de manera clara las funcionalidades y características que debía cumplir el sistema de monitoreo de la calidad del aire. Esta etapa fue fundamental para definir el comportamiento esperado de la solución y asegurar que el desarrollo responda a las necesidades planteadas desde el inicio del proyecto. Los requisitos se obtuvieron a partir del análisis del problema, la revisión del estado del arte, la observación directa del funcionamiento del sistema y las pruebas realizadas durante el desarrollo. Para una mejor organización, los requisitos se

clasificaron en requisitos funcionales y requisitos no funcionales, lo que facilitó su implementación y posterior validación.

4.3.3.1 *Requisitos funcionales*

Aquí se describen las acciones y servicios que el sistema debe proporcionar a los usuarios. En el caso del presente proyecto, estos requisitos están orientados principalmente a la adquisición, procesamiento, almacenamiento y visualización de los datos de calidad del aire, así como a la comunicación entre los distintos componentes del sistema.

Tabla 6

Requisitos funcionales del sistema

Código	Requisito funcional	Descripción	Módulo asociado
RF-SIS-01	Adquisición de datos	El sistema debe capturar datos de calidad del aire mediante sensores IoT	IoT
RF-SIS-02	Procesamiento local	El sistema debe procesar y almacenar datos localmente en la Raspberry Pi	IoT
RF-SIS-03	Envío de datos MQTT	El sistema debe enviar los datos mediante el protocolo MQTT	Comunicación
RF-SIS-04	Almacenamiento central	El sistema debe almacenar los datos en una base de datos centralizada	Backend
RF-SIS-05	Visualización en tiempo real	La aplicación debe mostrar datos de calidad del aire en tiempo real	App móvil
RF-SIS-06	Consulta histórica	La aplicación debe permitir la consulta de datos históricos	App móvil
RF-SIS-07	Autenticación de usuarios	El sistema debe permitir el acceso mediante autenticación	Seguridad
RF-SIS-08	Conectividad local y remota	El sistema debe adaptarse a conexiones locales o con internet	Comunicación

Nota. Elaboración propia.

4.3.3.2 Requisitos no funcionales

Aquí se describen las acciones y servicios que el sistema debe proporcionar a los usuarios. En el caso del presente proyecto, estos requisitos están orientados principalmente a la adquisición, procesamiento, almacenamiento y visualización de los datos de calidad del aire, así como a la comunicación entre los distintos componentes del sistema.

Tabla 7

Requisitos no funcionales del sistema

Código	Requisito no funcional	Categoría
RNF-SIS-01	Disponibilidad del sistema	Disponibilidad
RNF-SIS-02	Seguridad de la información	Seguridad
RNF-SIS-03	Tiempo de respuesta adecuado	Rendimiento
RNF-SIS-04	Facilidad de uso de la aplicación	Usabilidad
RNF-SIS-05	Escalabilidad del sistema	Escalabilidad
RNF-SIS-06	Compatibilidad con Android e iOS	Portabilidad
RNF-SIS-07	Integridad de los datos	Confiabilidad

Nota. Elaboración propia

4.3.4 Elaboración del Producto Backlog

El Product Backlog se construyó identificando todas las funcionalidades y tareas necesarias para el desarrollo completo del sistema, transformando los requisitos en historias de usuario claras y directas.

4.3.4.1 Historias de usuarios del Product Backlog

- **HU-01:** Configurar la Raspberry Pi, actualización del sistema operativo y revisión de compatibilidad.
- **HU-02:** Verificación de instalación y configuración de los sensores de calidad del aire (PM, gases, temperatura, humedad) en la Raspberry Pi para asegurar la correcta captura de datos.

- **HU-03:** Crear un servicio en Python para leer datos de los sensores y almacenarlos en SQLite, manteniendo un registro local.
- **HU-04:** Implementar un servicio de envío de datos mediante MQTT hacia un broker local o remoto para garantizar la comunicación con la aplicación y el backend.
- **HU-05:** Diseñar y configurar la base de datos en PostgreSQL para almacenar de manera centralizada los datos enviados desde la Raspberry Pi.
- **HU-06:** Implementar el backend en NestJS para procesar los datos, exponerlos mediante API y gestionar usuarios y autenticación.
- **HU-07:** Implementar el almacenamiento de datos históricos en el backend para permitir consultas y generación de reportes en la aplicación.
- **HU-08:** Autenticar usuarios en la aplicación mediante Google Authentication teniendo a Clerk como proveedor para asegurar el acceso seguro a los datos.
- **HU-09:** Visualizar los datos de calidad del aire en tiempo real desde la aplicación móvil para conocer el estado actual del ambiente.
- **HU-10:** Consultar los datos históricos en la aplicación para analizar la evolución de la calidad del aire.
- **HU-11:** Implementar la comunicación adaptativa local/remota (Wi-Fi Direct o internet) para garantizar la funcionalidad de la app sin conexión a internet.
- **HU-12:** Configurar la máquina virtual en la nube con backend y servicios de despliegue para asegurar la disponibilidad remota del sistema.
- **HU-13:** Realizar pruebas de integración y validación de servicios para asegurar el correcto funcionamiento de todos los componentes.
- **HU-14:** Implementar medidas de seguridad y control de acceso para proteger los datos y la comunicación entre dispositivos.
- **HU-15:** Generar documentación técnica y manuales de usuario para facilitar el uso y mantenimiento del sistema.

4.3.5 Desarrollo iterativo mediante Sprints

Siguiendo la metodología Scrum, cada sprint tuvo objetivos claros y definidos, tomando como referencia las historias de usuario del Product Backlog. Este enfoque permitió avanzar de manera progresiva, asegurando que cada funcionalidad fuera implementada, probada y validada antes de pasar a la siguiente etapa.

A continuación, se presenta como se distribuyeron los sprints con las historias de usuarios asociadas y los resultados obtenidos en cada iteración.

Tabla 8

Planificación de Sprints del proyecto

Sprint	Historias de usuarios desarrolladas	Descripción	Tiempo dedicado
Sprint 1	HU-01	Configuración inicial	1 semana
	HU-02		(15/09/2025 - 19/09/2025)
	HU-03		
Sprint 2	HU-04	Comunicación y almacenamiento central	2 semanas
	HU-05		(22/09/2025 - 03/10/2025)
	HU-06		
Sprint 3	HU-08	Visualización y autenticación	3 semanas
	HU-09		(06/10/2025 - 24/10/2025)
Sprint 4	HU-07	Consulta histórica y adaptabilidad	2 semanas
	HU-10		(27/10/2025 - 07/11/2025)
	HU-11		
Sprint 5	HU-12	Configuración remota y despliegue en la nube	1 semana
	HU-13		(10/11/2025 - 14/11/2025)
Sprint 6	HU-14	Pruebas, integración y seguridad	3 semanas
	HU-14		(17/11/2025 - 10/12/2025)
Sprint 7	HU-15	Documentación y cierre	3 semanas
			(08/01/2026 - 30/01/2026)

Nota. Elaboración propia.

4.3.6 *Diseño arquitectónico y tecnológico*

En la fase de Diseño arquitectónico y tecnológico aborda de manera clara la estructura del sistema, como interactúan sus componentes, que tecnologías intervienen en cada nivel. El diseño arquitectónico constituye la base sobre la cual se desarrolló e integró toda la solución, garantizando su correcto funcionamiento, escalabilidad y mantenimiento.

El objetivo principal de esta etapa fue organizar la solución de manera clara y modular, permitiendo un adecuado flujo de información desde la captura de los datos ambientales hasta su visualización final en la aplicación móvil.

El sistema está bajo una arquitectura distribuida, en la que cada componente cumple una función específica dentro del proceso de monitoreo de la calidad del aire. Este enfoque facilita la escalabilidad, el mantenimiento y la integración de nuevas funcionalidades, además de permitir el funcionamiento del sistema tanto en entornos locales como en escenarios con conexión a internet.

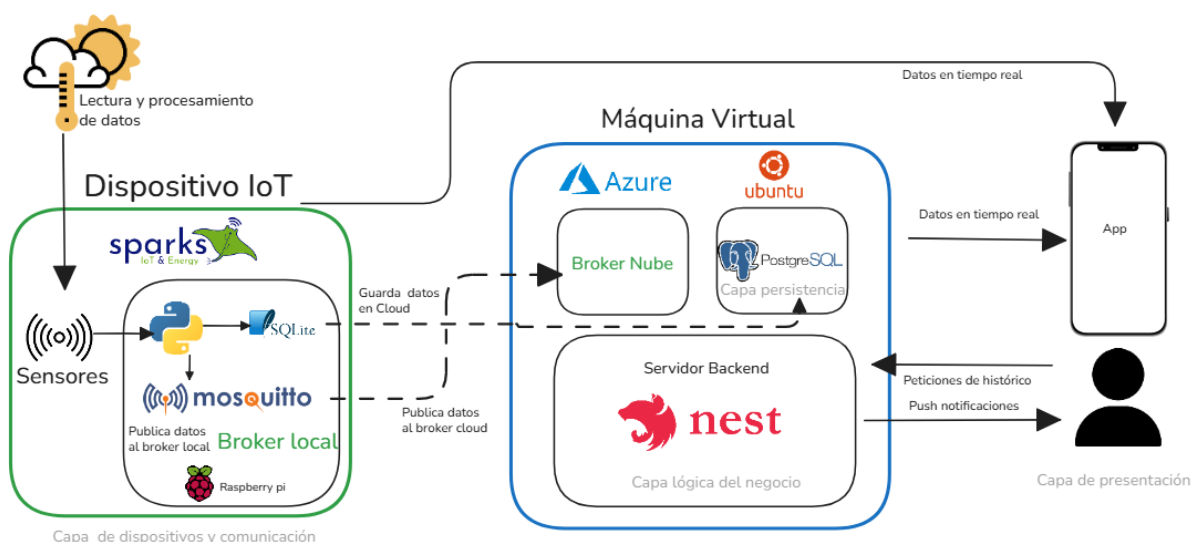
4.3.6.1 *Diseño arquitectónico general*

En la figura 5, se presenta el diseño arquitectónico y tecnológico, mediante un diagrama que permite visualizar de forma clara la interacción entre los distintos componentes tecnológicos que conforman la solución. En este esquema se identifica el flujo de datos desde los sensores de calidad del aire, pasando por los servicios de comunicación y procesamiento, hasta llegar a la aplicación móvil utilizada por los usuarios finales.

Los datos ambientales son capturados por los sensores conectados a la Raspberry Pi, la cual actúa como dispositivo central de adquisición. Posteriormente, esta información es enviada a través del protocolo MQTT hacia los servicios backend, donde es procesada y almacenada en la base de datos correspondiente. Finalmente, la aplicación móvil consume estos datos para su presentación en tiempo real y de forma histórica.

Figura 6

Diagrama arquitectónico y tecnológico del sistema



Nota. Elaboración propia con la herramienta Excalidraw.

4.3.6.2 Capas del sistema

Con el fin de mantener una estructura clara y ordenada, el sistema se organizó en capas, cada una con responsabilidades bien definidas. A continuación, se describen las principales capas que conforman la arquitectura del proyecto.

Adquisición de datos

Esta capa corresponde al nivel más bajo del sistema y está encargada de la captura de la información ambiental. Está conformada por los sensores de calidad del aire conectados a la Raspberry Pi, los cuales miden variables como material particulado, temperatura, humedad y gases contaminantes.

La Raspberry Pi actúa como nodo central de adquisición, encargándose de leer los datos de los sensores, realizar un preprocesamiento básico y preparar la información para su envío hacia las capas superiores del sistema.

Comunicación

La capa de comunicación se encarga de transmitir los datos capturados desde la Raspberry Pi hacia los servicios de procesamiento. Para este propósito se emplea el protocolo MQTT, el

cual permite una comunicación ligera, eficiente y orientada a mensajes, adecuada para sistemas IoT.

Esta capa garantiza el envío de los datos en tiempo real o casi en tiempo real, permitiendo que el sistema funcione tanto en escenarios con conexión a internet como en entornos locales, según la configuración establecida.

Procesamiento y servicios

Aquí se encuentran los servicios encargados de recibir, procesar y gestionar los datos enviados desde la capa de comunicación. Aquí se realiza la validación de la información, el procesamiento necesario y la preparación de los datos para su almacenamiento y posterior consulta.

Esta capa constituye el núcleo lógico del sistema, ya que permite centralizar las reglas de negocio, facilitar la integración con otros componentes y asegurar la consistencia de la información.

Persistencia de datos

Esta capa es la responsable del almacenamiento de la información generada por el sistema. En ella se gestionan las bases de datos donde se registran los datos históricos de calidad del aire, permitiendo su consulta, análisis y visualización posterior.

El diseño de esta capa permite mantener la integridad de los datos y soportar consultas tanto en tiempo real como históricas, lo cual resulta fundamental para el análisis del comportamiento de la calidad del aire a lo largo del tiempo.

Presentación

Esta capa corresponde a la aplicación móvil desarrollada, la cual actúa como interfaz entre el sistema y los usuarios finales. A través de esta capa, los usuarios pueden visualizar los datos de calidad del aire de manera clara y organizada, acceder a información en tiempo real e histórica, y recibir retroalimentación del estado del entorno monitoreado.

Esta capa se comunica con los servicios backend para obtener la información necesaria, asegurando una experiencia de usuario adecuada y alineada con los objetivos del proyecto.

Con este enfoque, el diseño arquitectónico y tecnológico del sistema establece una base sólida para las siguientes fases del desarrollo, donde se detallan las configuraciones de

infraestructura, servicios específicos, despliegue, seguridad y pruebas, permitiendo una implementación ordenada y coherente de la propuesta.

4.3.7 Configuración de la máquina virtual y entorno de ejecución del sistema

En esta fase se realizó la configuración de la infraestructura tecnológica necesaria para el funcionamiento del sistema propuesto. Para el desarrollo del proyecto se utilizó una única máquina virtual alojada en la plataforma Microsoft Azure, la cual permitió centralizar los servicios principales del sistema, facilitando su administración y garantizando un entorno estable para la ejecución de la solución.

La máquina virtual fue configurada con el sistema operativo Ubuntu, sobre el cual se desplegaron los servicios backend, la base de datos y el broker de mensajería utilizados por el sistema. Esta decisión permitió contar con un entorno flexible, escalable y alineado con las necesidades del proyecto

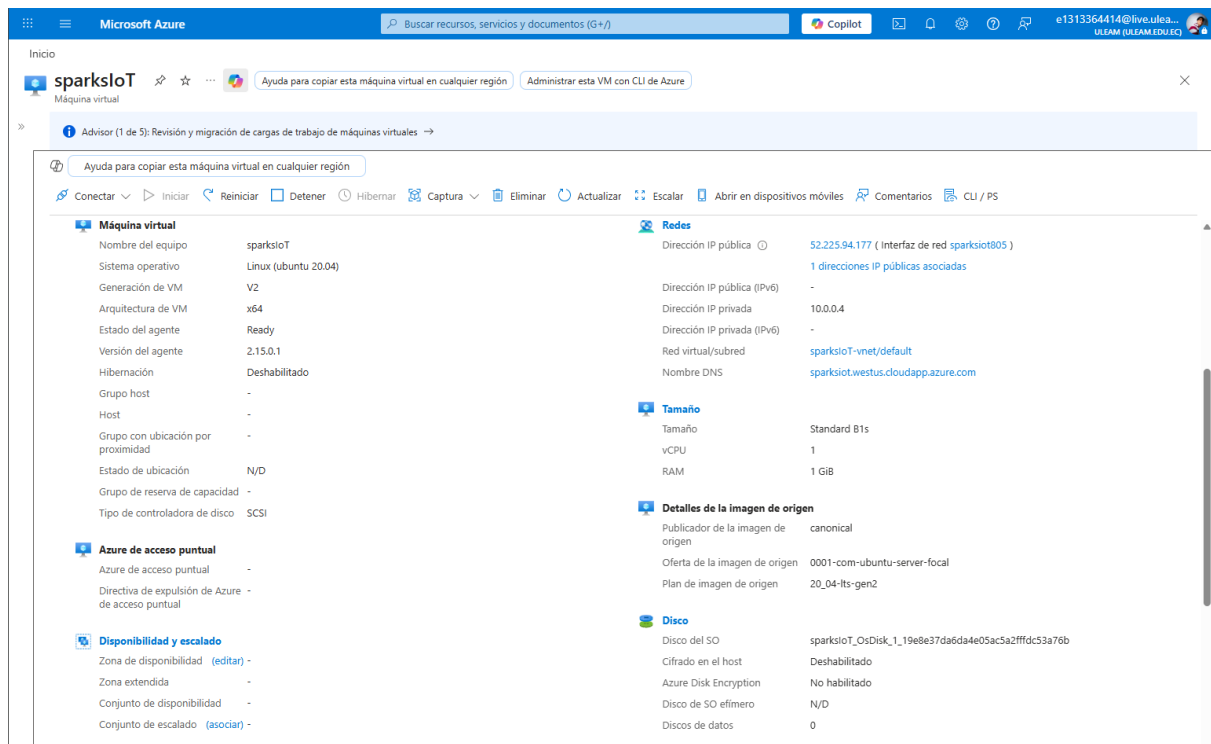
4.3.7.1 Configuración de la máquina virtual en Azure

Se empleó una máquina virtual gestionada desde el portal de Azure, donde se definieron los parámetros básicos de infraestructura tales como el sistema operativo, recursos de procesamiento y almacenamiento necesarios para soportar la carga del sistema.

El acceso a la máquina virtual se realizó mediante conexión segura por protocolo SSH, lo que permitió administrar el servidor, instalar dependencias y realizar las configuraciones necesarias para el correcto funcionamiento de los servicios. En la figura 6 podemos ver las características desde el portal de Azure las propiedades de la máquina virtual.

Figura 7

Máquina virtual configurada en la plataforma Azure



Nota. Captura de pantalla de las especificaciones de la máquina virtual desde Azure.

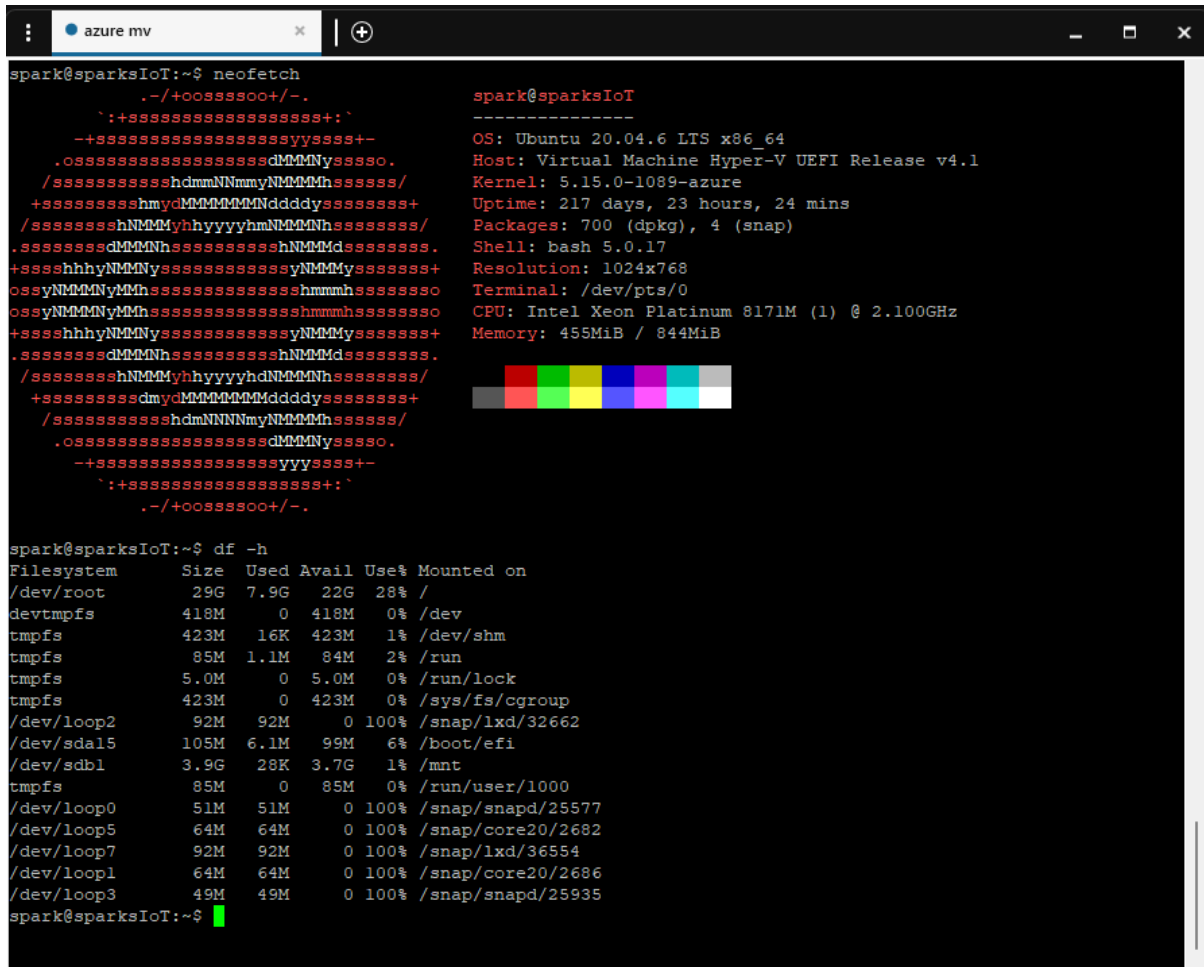
4.3.7.2 Configuración del sistema operativo Ubuntu

Una vez creada la máquina virtual, se procedió a la configuración del sistema operativo Ubuntu, el cual sirvió como base para la ejecución de los distintos servicios del sistema. En esta etapa se realizaron tareas como la actualización del sistema, la instalación de paquetes necesarios y la configuración básica del entorno de trabajo.

Ubuntu permitió contar con un entorno estable y ampliamente compatible con las tecnologías utilizadas en el proyecto, facilitando la instalación y gestión de los servicios backend, la base de datos y el broker de mensajería. En la figura 7 podemos ver las características y especificaciones de la máquina virtual con el sistema operativo Ubuntu y el entorno donde se va a ejecutar el proyecto.

Figura 8

Acceso y configuración del sistema operativo Ubuntu en la máquina virtual



```
spark@sparkIoT:~$ neofetch
./+oossssoo+/-
`:+ssssssssssssssssss+:`
  -+ssssssssssssssssssyyss+-
  .osssssssssssssssssdMMNysso.
  /ssssssssssshdmmNNmmyNMMNhsssss/
  +ssssssssshmydMMMMMMNdddyssssss+
  /ssssssssshNMMMyhhyyyhmNMMNhsssss/
  .ssssssssdMMNhsssssssshNMMdssssss.
  +ssssshhNMMNysssssssssyNMMysssss+
  ossyNMMNymMhssssssssshmmhssssso
  +ssssshhNMMNysssssssssyNMMysssss+
  .ssssssssdMMNhsssssssshNMMdssssss.
  /ssssssshNMMMyhhyyyhdNMMNhsssss/
  +ssssssssdmydMMMMMMNdddyssssss+
  /ssssssssshdmmNNmmyNMMNhsssss/
  .osssssssssssssssdMMNysso.
  -+ssssssssssssssssyyss+-
  `:+ssssssssssssssss+:`
  ./+oossssoo+/-

spark@sparkIoT
-----
OS: Ubuntu 20.04.6 LTS x86_64
Host: Virtual Machine Hyper-V UEFI Release v4.1
Kernel: 5.15.0-1089-azure
Uptime: 217 days, 23 hours, 24 mins
Packages: 700 (dpkg), 4 (snap)
Shell: bash 5.0.17
Resolution: 1024x768
Terminal: /dev/pts/0
CPU: Intel Xeon Platinum 8171M (1) @ 2.100GHz
Memory: 455MiB / 844MiB

spark@sparkIoT:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        29G   7.9G   22G   28% /
devtmpfs         418M    0   418M    0% /dev
tmpfs            423M   16K   423M    1% /dev/shm
tmpfs            85M    1.1M   84M    2% /run
tmpfs            5.0M    0    5.0M    0% /run/lock
tmpfs            423M    0   423M    0% /sys/fs/cgroup
/dev/loop2       92M    92M    0 100% /snap/lxd/32662
/dev/sda15       105M   6.1M   99M    6% /boot/efi
/dev/sdb1        3.9G   28K   3.7G    1% /mnt
tmpfs            85M    0    85M    0% /run/user/1000
/dev/loop0       51M    51M    0 100% /snap/snapd/25577
/dev/loop5       64M    64M    0 100% /snap/core20/2682
/dev/loop7       92M    92M    0 100% /snap/lxd/36554
/dev/loop1       64M    64M    0 100% /snap/core20/2686
/dev/loop3       49M    49M    0 100% /snap/snapd/25935
spark@sparkIoT:~$
```

Nota. Captura de pantalla de las características de la máquina virtual mediante la conexión con Sola Putty

4.3.8 Gestión de mensajería

Dentro del entorno de ejecución del sistema, se configuró un broker MQTT dentro de la misma máquina virtual. Este componente cumple un rol fundamental en la comunicación entre la Raspberry Pi y el backend, permitiendo la publicación y suscripción de mensajes de manera eficiente.

El broker MQTT fue configurado para recibir los datos enviados desde los dispositivos IoT y permitir que los servicios backend se suscriban a los tópicos correspondientes. De esta manera, se garantiza un flujo de información continuo y en tiempo real.

En esta fase se estableció el funcionamiento general del broker, mientras que los detalles relacionados con tópicos, seguridad y pruebas de comunicación se abordan en secciones posteriores. Sin olvidar que el sistema tiene un modo de conexión local, que prácticamente consiste en recibir datos conectados a la red de la raspberry, para esto de la misma manera se configuró un bróker local.

4.3.9 Proceso de despliegue del sistema

En esta etapa se pone en funcionamiento todos los componentes desarrollados en el punto anterior. El despliegue permite la validación de que todos los servicios se hayan integrado correctamente, garantizando la disponibilidad del sistema para su uso en condiciones reales.

Para el despliegue se sigue un orden, considerando la dependencia entre los distintos componentes del sistema y asegurando que cada servicio se encuentre operando antes de que se integre con los demás.

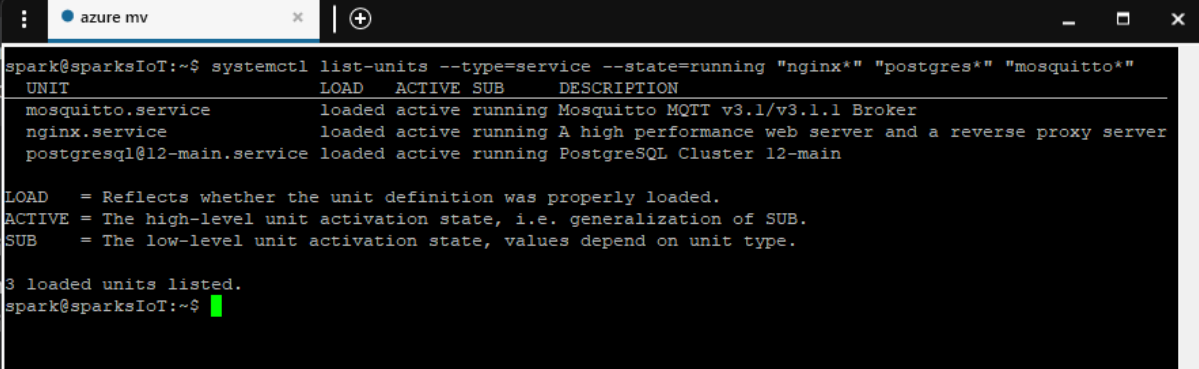
4.3.9.1 Preparación del entorno

El primer paso, es verificar que la máquina virtual cuente con todas las dependencias necesarias para la ejecución del sistema. Esto incluye la validación del sistema operativo completamente actualizado, entorno de ejecución del backend con su servidor levantado y respondiendo a las peticiones, la base de datos activa y persistente y bróker escuchando publicaciones del MQTT.

Con esto se asegura reducir errores durante el despliegue, además de que se garantiza que el entorno de producción se encuentre alineado con las configuraciones previamente definidas, como se muestra en la figura 9, el estado de los servicios configurados se encuentra activos.

Figura 9

Verificación del entorno de producción previo al despliegue



```
spark@sparksIoT:~$ systemctl list-units --type=service --state=running "nginx*" "postgres*" "mosquitto*"
UNIT                                LOAD    ACTIVE SUB    DESCRIPTION
mosquitto.service                  loaded active running Mosquitto MQTT v3.1/v3.1.1 Broker
nginx.service                       loaded active running A high performance web server and a reverse proxy server
postgresql@12-main.service          loaded active running PostgreSQL Cluster 12-main

LOAD    = Reflects whether the unit definition was properly loaded.
ACTIVE  = The high-level unit activation state, i.e. generalization of SUB.
SUB     = The low-level unit activation state, values depend on unit type.

3 loaded units listed.
spark@sparksIoT:~$
```

Nota. Captura de pantalla del estado del servicio MQTT desde Solar Putty

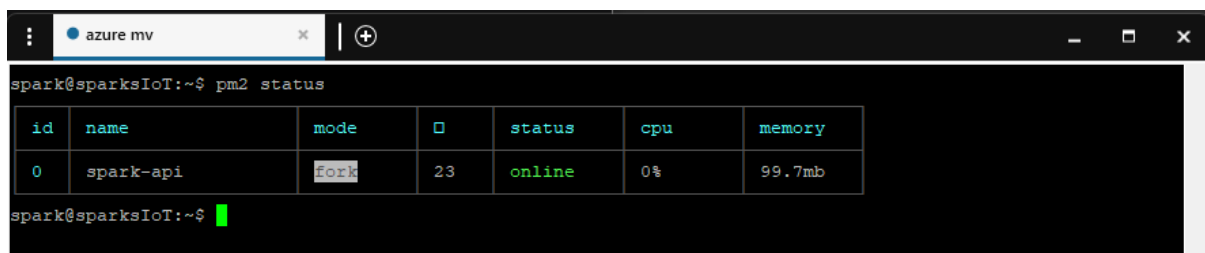
4.3.9.2 Despliegue del backend

Ya preparado el entorno, se continúa al despliegue del backend desarrollado con NESTJS. El código fuente fue transferido al servidor mediante git, clonamos el repositorio, se instalan las dependencias del proyecto, se hace el build para su correcta ejecución.

Se inicializa el backend y mediante el gestor de procesos de PM2 se asegura la ejecución continua y su disponibilidad ante posibles reinicios del sistema. Como podemos apreciar en la figura 9, el backend se encuentra listo para recibir datos desde el bróker y atender las solicitudes provenientes de la aplicación móvil.

Figura 10

Backend desplegado y gestionado mediante PM2



The screenshot shows a terminal window with a dark background. At the top, there's a tab labeled 'azure mv'. The terminal prompt is 'spark@sparksIoT:~\$'. The command 'pm2 status' has been executed, resulting in a table with the following data:

id	name	mode	□	status	cpu	memory
0	spark-api	fork	23	online	0%	99.7mb

Below the table, the terminal prompt 'spark@sparksIoT:~\$' is visible with a green cursor.

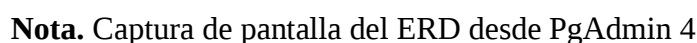
Nota. Captura de pantalla del administrador del backend pm2, desde Solar Putty

4.3.9.3 Despliegue de la base de datos

Se verifica el correcto funcionamiento de la base de datos PostgreSQL configurada en la máquina virtual. Se validó la estructura de las tablas necesarias y la conectividad entre el backend y la base de datos.

Este paso fue fundamental para garantizar que los datos recibidos durante la operación del sistema puedan ser almacenados de manera correcta y consultados posteriormente.

Diagrama ERD de la base de datos en PostgreSQL operativa en el entorno de producción



Para el bróker, este fue puesto en funcionamiento como parte del proceso de despliegue, asegurando que se encuentre disponible para recibir los mensajes publicados desde la Raspberry Pi. Se realizaron pruebas básicas de publicación y suscripción para validar la correcta recepción de los datos.

Figura 12

```

● servicio_no_connect_mqtt.service - Sparks ULEAM Conexión cuando el dispositivo no cuenta con internet
   Loaded: loaded (/etc/systemd/system/servicio_no_connect_mqtt.service; disabled; vendor preset: enabled)
   Active: active (running) since Mon 2026-02-02 11:29:50 -05; 4h 0min ago
     Main PID: 1428 (python3)
       Tasks: 1 (limit: 3720)
    CGroup: /system.slice/servicio_no_connect_mqtt.service
            └─1428 /usr/bin/python3 /home/pi/codigo/servicio_no_connect_mqtt.py

feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 185998, 'timestamp': 1770063738, 'tem': 26.39, 'hum': 60.13,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 185999, 'timestamp': 1770063798, 'tem': 26.42, 'hum': 59.89,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 185999, 'timestamp': 1770063798, 'tem': 26.42, 'hum': 59.89,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 186000, 'timestamp': 1770063858, 'tem': 26.46, 'hum': 59.8,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 186000, 'timestamp': 1770063858, 'tem': 26.46, 'hum': 59.8,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 186001, 'timestamp': 1770063918, 'tem': 26.51, 'hum': 59.65,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 186001, 'timestamp': 1770063918, 'tem': 26.51, 'hum': 59.65,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 186002, 'timestamp': 1770063978, 'tem': 26.55, 'hum': 59.54,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 186002, 'timestamp': 1770063978, 'tem': 26.55, 'hum': 59.54,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 186002, 'timestamp': 1770063978, 'tem': 26.55, 'hum': 59.54,
lines 1-18/18 (END)

```

60

4.3.9.5 Despliegue del bróker de IoT Uleam

Para este apartado, se agregó un nuevo servicio, publicando datos hacia el bróker de IoT Uleam. Este despliegue permite que los datos lleguen al dashboard de la categoría Calidad del aire.

Figura 13

Servicio del broker de IoT Uleam

```
pi@raspberrypi ~/codigo (0.191s)
sudo systemctl restart iot-publisher

pi@raspberrypi ~/codigo (0.139s)
sudo systemctl status iot-publisher

● iot-publisher.service - Sistema de Publicación IoT ULEAM
   Loaded: loaded (/etc/systemd/system/iot-publisher.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2026-02-02 15:57:35 -05; 6s ago
     Main PID: 3420 (python3)
        Tasks: 2 (limit: 3720)
      CGroup: /system.slice/iot-publisher.service
              └─3420 /usr/bin/python3 /home/pi/codigo/suscribe_public_iotv2.py

feb 02 15:57:35 raspberrypi systemd[1]: Started Sistema de Publicación IoT ULEAM.
```

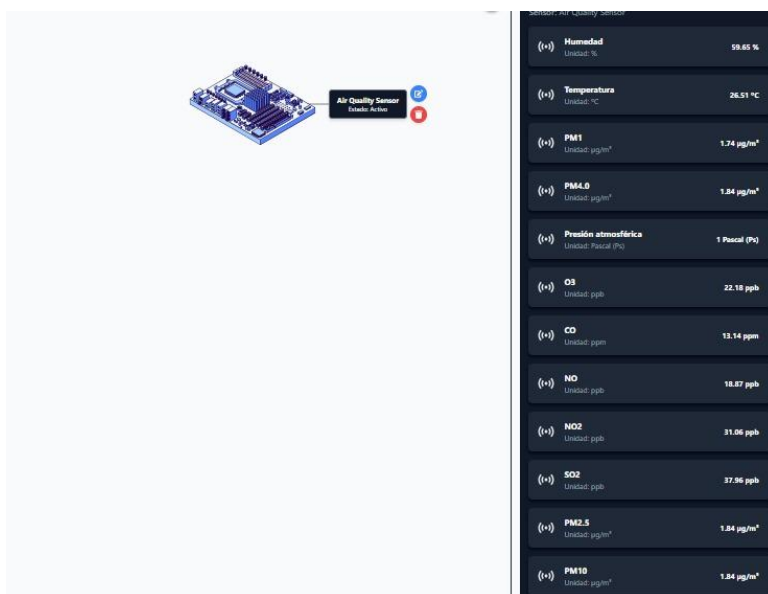
Nota. Elaboración propia, captura de pantalla desde Solar Putty

4.3.9.6 Integración de parámetros en IoT Uleam

Dentro del sistema de IoT Uleam, autenticados como administrador, se procede a agregar los parámetros con las unidades de medida, estos datos son los que va a recibir del publicador previamente configurado, en la figura 14 podemos apreciar el dispositivo configurado y los parámetros agregados.

Figura 14

Dispositivo configurado y parámetros agregados



Nota. Elaboración propia, captura de pantalla desde el sistema IoT Uleam

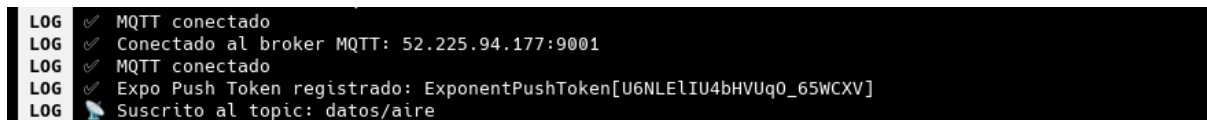
4.3.9.7 Integración con la aplicación móvil

Con los servicios del servidor ya desplegados, se configuró la aplicación móvil para conectarse al backend y consumir los datos proporcionados por el sistema. Se realizaron pruebas de conectividad para verificar que la información capturada por los sensores sea correctamente visualizada en la aplicación.

Esta integración permitió validar el funcionamiento completo del sistema, desde la captura de datos hasta su presentación al usuario final.

Figura 15

Logs de la aplicación conectándose y recibiendo datos



Nota. Captura de pantalla de los logs proporcionados por Expo Go

4.3.9.8 Validación de despliegue

Por último, se realizaron pruebas generales para confirmar que todos los componentes del sistema se encuentren operando de forma conjunta. Esta validación incluyó la revisión del flujo de datos, la estabilidad de los servicios y la correcta visualización de la información.

El proceso de despliegue permitió confirmar que la solución propuesta cumple con los objetivos planteados y se encuentra lista para su evaluación y uso.

4.3.9.9 Servicio de notificaciones y comunicación con la interfaz

El sistema cuenta con un servicio de notificaciones encargado de gestionar la comunicación entre los procesos internos del sistema y la interfaz de usuario. Este servicio permite informar al usuario sobre eventos relevantes que ocurren durante el funcionamiento de la aplicación, tales como cambios en la conectividad, recepción de nuevos datos o estados importantes del sistema.

El servicio actúa como un intermediario entre los procesos que se ejecutan en segundo plano y la capa de presentación, asegurando que la información sea transmitida de manera oportuna y organizada. De esta forma, la interfaz puede actualizarse dinámicamente sin necesidad de intervención directa del usuario.

La implementación de este servicio mejora la usabilidad y la percepción de confiabilidad del sistema, ya que el usuario es informado en tiempo real sobre el estado de la aplicación y de los datos de monitoreo de la calidad del aire.

4.3.9.10 Servicio de visualización y presentación de datos

Aquí se procesa y organiza la información obtenida desde las diferentes fuentes del sistema, con el fin de mostrarla de manera clara y comprensible al usuario final. Este servicio transforma los datos recibidos en información estructurada, facilitando su interpretación.

La información de calidad del aire se presenta de forma dinámica, permitiendo la actualización continua de los valores conforme se reciben nuevos datos. Este enfoque permite al usuario conocer el estado ambiental en tiempo real, apoyando la toma de decisiones y la concientización sobre la calidad del aire.

Este servicio prioriza la claridad, organización y facilidad de interpretación de los datos, asegurando que la información presentada sea útil y accesible para cualquier tipo de usuario, independientemente de su nivel técnico.

4.3.10 Seguridad del sistema

La solución garantiza la protección de la información, el control de acceso a los servicios y la confiabilidad de los datos de monitoreo de la calidad del aire. Durante el desarrollo e implementación del sistema se aplicaron diversas medidas de seguridad orientadas a minimizar riesgos y asegurar un funcionamiento adecuado en los distintos entornos de ejecución.

4.3.10.1 Autenticación de usuarios

El sistema tiene implementado Clerk, una plataforma para la autenticación de usuarios que permite controlar el acceso a las funcionalidades de la aplicación móvil. Este proceso asegura que únicamente usuarios autorizados puedan acceder a la información presentada por el sistema, protegiendo los datos frente a accesos no autorizados.

La autenticación se integra con los servicios del backend, permitiendo validar la identidad del usuario antes de conceder el acceso a los recursos disponibles, fortaleciendo así la seguridad general del sistema.

4.3.10.2 Control de acceso a los servicios

Además de la autenticación, se implementó un control de acceso a los servicios expuestos por el backend, garantizando que únicamente las solicitudes válidas y correctamente autorizadas puedan ser procesadas. Este control permite restringir el uso de los servicios según el rol o estado del usuario, evitando usos indebidos del sistema.

El control de acceso contribuye a mantener la integridad del sistema y a prevenir posibles vulnerabilidades derivadas del acceso no controlado a los recursos.

4.3.10.3 Seguridad en la comunicación

La comunicación entre los diferentes componentes del sistema, como el dispositivo IoT, el backend y la aplicación móvil, fue diseñada considerando principios básicos de seguridad. Se establecieron configuraciones que permiten una transmisión de datos confiable, reduciendo el riesgo de interceptación o manipulación de la información durante su envío.

En el caso del uso del protocolo MQTT, se definieron tópicos específicos y se gestionó el acceso a los mismos, asegurando que los datos de monitoreo sean transmitidos únicamente entre los componentes autorizados del sistema.

4.3.10.4 Protección de la información

La información almacenada en las bases de datos fue gestionada de manera segura, considerando buenas prácticas para el manejo de datos sensibles y operativos. El uso de bases de datos separadas para el entorno local y el entorno en la nube permite reducir riesgos ante fallos de conectividad, además de mejorar la disponibilidad y confiabilidad de los datos.

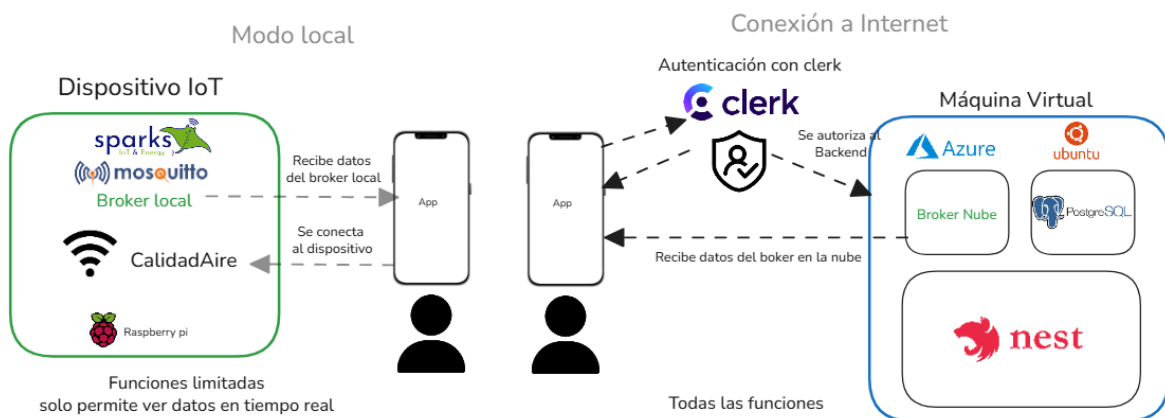
Asimismo, se cuidó la gestión de credenciales y configuraciones sensibles mediante el uso de variables de entorno, evitando la exposición directa de información crítica dentro del código fuente.

4.3.11 Buenas prácticas aplicadas

Se aplicaron buenas prácticas de seguridad, tales como la separación de entornos, el control de configuraciones, la validación de datos y la gestión adecuada de accesos. Estas prácticas contribuyeron a fortalecer la seguridad del sistema y a garantizar un funcionamiento estable y confiable.

Figura 16

Mecanismo de autenticación y control de acceso al sistema



Nota. Elaboración propia, diseñado con Excalidraw

4.3.11.1 Estrategias y pruebas realizadas

Para garantizar el funcionamiento de todo el sistema propuesto, se aplicaron estrategias de pruebas a lo largo del proceso de desarrollo e implementación. Estas pruebas permitieron validar la integración entre los distintos componentes del sistema, así como comprobar que la solución cumple con los requisitos funcionales y no funcionales definidos previamente.

Las estrategias de prueba se ejecutaron de manera progresiva, acompañando cada etapa del desarrollo, lo que facilitó la detección temprana de errores y la mejora continua del sistema.

4.3.11.2 Pruebas funcionales

Se realizaron pruebas comprobando cada una de las funcionalidades del sistema opere conforme a lo establecido en los requisitos funcionales. Se validó la correcta visualización de los datos de calidad del aire, la consulta de información histórica, el proceso de autenticación de usuarios y la interacción general de la aplicación móvil con el backend. Confirmando que las funciones principales del sistema responden de manera adecuada a las acciones del usuario y a los datos recibidos desde el dispositivo IoT.

4.3.11.3 Pruebas de conectividad

Como hemos visto en las etapas anteriores, el sistema tiene distintos escenarios de conectividad, se realizaron pruebas específicas para validar el comportamiento del sistema tanto en entornos con conexión a internet como en entornos locales mediante conexión directa con la Raspberry Pi. Se comprobó la correcta suscripción a los tópicos MQTT y el cambio automático de conexión según la red disponible. Garantizando la continuidad del servicio y la correcta recepción de los datos en tiempo real.

4.3.11.4 Pruebas de rendimiento

Para evaluar la capacidad del sistema para manejar la recepción y procesamiento de datos de manera continua. Se observó el comportamiento del backend y de la base de datos ante la recepción periódica de información, verificando que el sistema mantenga un tiempo de respuesta adecuado y un funcionamiento estable. Demostrando que la solución es capaz de operar en un entorno real sin afectar la experiencia del usuario.

4.3.11.5 Validación de resultados

Finalmente, los resultados obtenidos durante las pruebas permitieron validar el cumplimiento de los objetivos planteados para la solución tecnológica. El sistema demostró ser funcional, estable y capaz de proporcionar información confiable sobre la calidad del aire, tanto en tiempo real como de forma histórica.

CAPÍTULO V

5 Resultados y validación

5.1 Resultados de la implementación

Como resultado del desarrollo de la propuesta planteada, se logró implementar un sistema funcional para el monitoreo de la calidad del aire, el cual integra de manera efectiva componentes de hardware y software bajo un enfoque basado en tecnologías IoT. El sistema desarrollado permite la adquisición, transmisión, procesamiento, almacenamiento y visualización de datos ambientales, cumpliendo con los objetivos definidos en las etapas iniciales del proyecto.

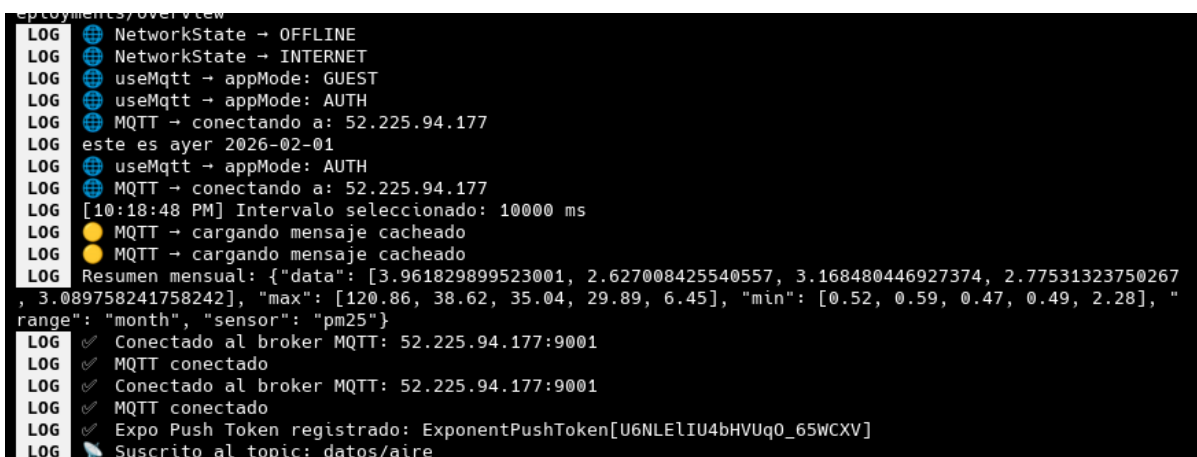
La implementación del sistema se compone de varios elementos interconectados: un dispositivo IoT encargado de la recolección de datos mediante sensores, un backend desplegado en la nube responsable del procesamiento y gestión de la información, un sistema de mensajería para la comunicación en tiempo real y una aplicación móvil que permite al usuario visualizar los datos y recibir notificaciones relevantes. Esta integración garantiza un flujo continuo de información desde el entorno físico hasta la interfaz de usuario final.

Durante la ejecución del sistema, los datos capturados por los sensores de calidad del aire son enviados de forma periódica al backend, utilizando mecanismos de comunicación adecuados según la disponibilidad de conectividad. El sistema fue diseñado para operar tanto en entornos locales, mediante conexión directa con el dispositivo IoT, como en entornos con acceso a internet, lo que incrementa su flexibilidad y aplicabilidad en escenarios reales.

Como resultado de esta implementación, se obtuvo una solución tecnológica operativa que permite visualizar información en tiempo real, consultar datos históricos, recibir notificaciones ante eventos relevantes y mantener la continuidad del servicio incluso ante limitaciones de conectividad. Estos resultados evidencian que la propuesta desarrollada no solo cumple con los requisitos planteados, sino que también demuestra su viabilidad y utilidad como herramienta de apoyo para el monitoreo de la calidad del aire.

Figura 17

Funcionamiento general del sistema de monitoreo de la calidad del aire



```
LOG [10:18:48 PM] NetworkState → OFFLINE
LOG [10:18:48 PM] NetworkState → INTERNET
LOG [10:18:48 PM] useMqtt → appMode: GUEST
LOG [10:18:48 PM] useMqtt → appMode: AUTH
LOG [10:18:48 PM] MQTT → conectando a: 52.225.94.177
LOG [10:18:48 PM] este es ayer 2026-02-01
LOG [10:18:48 PM] useMqtt → appMode: AUTH
LOG [10:18:48 PM] MQTT → conectando a: 52.225.94.177
LOG [10:18:48 PM] [10:18:48 PM] Intervalo seleccionado: 10000 ms
LOG [10:18:48 PM] MQTT → cargando mensaje cacheado
LOG [10:18:48 PM] MQTT → cargando mensaje cacheado
LOG [10:18:48 PM] Resumen mensual: {"data": [3.961829899523001, 2.627008425540557, 3.168480446927374, 2.77531323750267, 3.089758241758242], "max": [120.86, 38.62, 35.04, 29.89, 6.45], "min": [0.52, 0.59, 0.47, 0.49, 2.28], "range": "month", "sensor": "pm25"}
LOG [10:18:48 PM] Conectado al broker MQTT: 52.225.94.177:9001
LOG [10:18:48 PM] MQTT conectado
LOG [10:18:48 PM] Conectado al broker MQTT: 52.225.94.177:9001
LOG [10:18:48 PM] MQTT conectado
LOG [10:18:48 PM] Expo Push Token registrado: ExponentPushToken[U6NLElIU4bHVUq0_65WCXV]
LOG [10:18:48 PM] Suscrito al topic: datos/aire
```

Nota. Elaboración propia de captura de pantalla desde Expo Go

5.2 Resultados del backend y servicios expuestos

Parte fundamental de la propuesta es el backend, encargado de centralizar la lógica del sistema, procesar los datos provenientes del dispositivo IoT y exponer los servicios necesarios para su consumo por parte de la aplicación móvil. El backend desarrollado permitió gestionar de manera eficiente la información de la calidad del aire, garantizando su disponibilidad, integridad y acceso controlado.

El backend se encuentra desplegado en un entorno de ejecución en la nube, lo que permitió asegurar su accesibilidad desde distintos dispositivos y ubicaciones. A través de este componente, el sistema es capaz de recibir datos en tiempo real, almacenarlos en la base de datos, procesarlos según los requerimientos del sistema y ponerlos a disposición mediante servicios REST bien definidos.

5.2.1 Gestión de datos y comunicación con el sistema IoT

El backend actúa como intermediario entre el dispositivo IoT y la aplicación móvil, recibiendo los datos generados por los sensores de calidad del aire. Esta información es procesada y validada antes de ser almacenada, asegurando consistencia y confiabilidad en los datos registrados.

Asimismo, el backend mantiene comunicación constante con el sistema de mensajería MQTT, permitiendo la recepción de datos en tiempo real y la correcta integración con los distintos modos de conexión del sistema, tanto local como remoto.

5.2.2 Servicios REST para el acceso a la información

La implementación de servicios REST permiten a la aplicación móvil consultar y gestionar la información almacenada en el sistema. Estos servicios facilitan el acceso a datos en tiempo real, la consulta de información histórica y la obtención de parámetros relevantes para la visualización y análisis de la calidad del aire.

5.2.3 Documentación de la API mediante Swagger

Para facilitar la validación y comprensión de los servicios desarrollados, se implementó la documentación de la API utilizando la herramienta Swagger. Esta documentación permitió visualizar de forma clara los endpoints disponibles, los métodos utilizados, los parámetros requeridos y las respuestas generadas por cada servicio.

Como podemos observar en la figura 18, el uso de Swagger facilitó las pruebas de los servicios del backend, permitiendo verificar su correcto funcionamiento de manera independiente a la aplicación móvil. Además, esta documentación contribuye a la mantenibilidad y escalabilidad del sistema, al ofrecer una referencia clara de los servicios disponibles.

Figura 18

Documentación de la API REST del sistema mediante Swagger



Nota. Elaboración propia de captura de pantalla desde el navegador

5.2.4 Validación de funcionamiento del backend

Durante las pruebas realizadas, se verificó que el backend responde correctamente a las solicitudes realizadas por la aplicación móvil y que los datos recibidos desde el dispositivo IoT son procesados y almacenados de forma adecuada. Estas validaciones confirmaron la estabilidad del backend y su capacidad para operar de manera continua en un entorno real.

Los resultados obtenidos demuestran que el backend cumple con su función como componente central del sistema, garantizando la correcta gestión de la información y la comunicación entre los diferentes módulos de la solución.

5.3 Resultados de la aplicación móvil

La aplicación móvil constituye el principal medio de interacción entre el usuario y el sistema de monitoreo de la calidad del aire. A través de esta aplicación, los usuarios pueden visualizar la información recopilada por los sensores, recibir notificaciones ante eventos relevantes y acceder a los datos tanto en tiempo real como de forma histórica.

La aplicación fue desarrollada de manera multiplataforma, permitiendo su ejecución en dispositivos Android e iOS, lo que amplía el alcance y la accesibilidad del sistema. Su diseño se enfocó en la facilidad de uso, claridad visual y respuesta rápida ante los cambios en los datos.

5.3.1 Interfaz de usuario y experiencia de uso

La interfaz de la aplicación móvil fue diseñada para presentar la información de forma clara e intuitiva. Se prioriza la visualización de los valores de calidad del aire mediante componentes gráficos y textos descriptivos, facilitando la interpretación de los datos por parte del usuario.

La navegación entre las distintas secciones de la aplicación se realiza de manera sencilla, permitiendo acceder rápidamente a las funcionalidades principales del sistema sin requerir conocimientos técnicos avanzados.

Figura 19

Pantalla principal de la aplicación móvil



Nota. Elaboración propia captura de pantalla de la vista de inicio desde un celular Android

5.3.2 Visualización de datos en tiempo real

Uno de los principales resultados obtenidos con la aplicación móvil es la capacidad de mostrar los datos de calidad del aire en tiempo real. Esta funcionalidad permite al usuario observar de forma inmediata los cambios en los valores capturados por los sensores, lo cual es fundamental para la toma de decisiones oportunas.

Como pudimos observar en la figura 19, la aplicación recibe esta información a través de la comunicación establecida con el backend o directamente mediante el broker MQTT en modo local, garantizando la actualización continua de los datos visualizados.

5.3.3 Consulta histórica de datos

Además de la visualización en tiempo real, la aplicación móvil permite consultar registros históricos de la calidad del aire. Esta funcionalidad facilita el análisis de tendencias y el seguimiento de la evolución de los valores registrados a lo largo del tiempo.

Los datos históricos son obtenidos desde la base de datos del sistema a través de los servicios REST del backend, asegurando consistencia y confiabilidad en la información presentada.

Figura 20

Consulta histórica de datos desde la aplicación móvil



Nota. Elaboración propia de captura de pantalla del histórico desde un celular Android

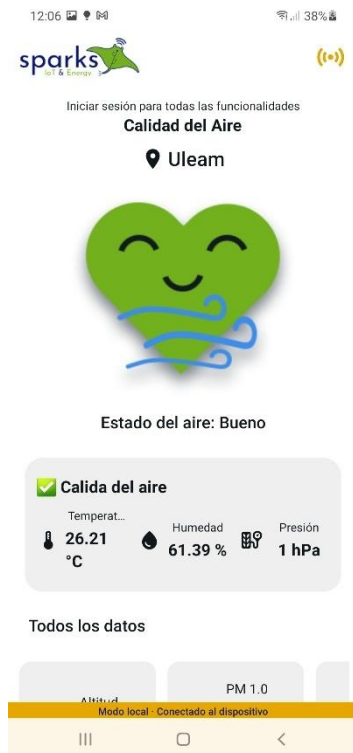
5.3.4 Modo de conexión local y remoto

La aplicación móvil fue diseñada para operar en dos modos de conexión: modo local y modo remoto. En el modo local, la aplicación se conecta directamente al sistema mediante una red local o Wi-Fi Direct, permitiendo la comunicación aun cuando no exista conexión a Internet.

En el modo remoto, la aplicación se comunica con el backend desplegado en la nube, como lo hemos visto en la figura 19, accediendo a los datos a través de los servicios expuestos. Esta flexibilidad garantiza el funcionamiento del sistema en distintos escenarios de conectividad. Y en la figura 21, vemos la conexión con el modo local, además apreciamos que las funciones son limitadas.

Figura 21

Funcionamiento de la aplicación en modo local



Nota. Elaboración propia de captura de pantalla de la vista de home en modo local desde un celular Android

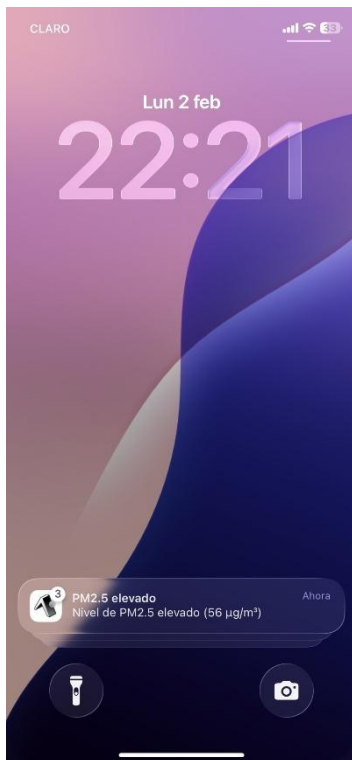
5.3.5 Sistema de notificaciones ante eventos

Como resultado del desarrollo de la aplicación móvil, se implementó un sistema de notificaciones que alerta al usuario cuando se detectan eventos relevantes, como valores fuera de los rangos establecidos para la calidad del aire.

Estas notificaciones permiten una reacción inmediata por parte del usuario, reforzando el carácter preventivo del sistema y mejorando su utilidad en contextos reales.

Figura 22

Notificación generada ante un evento de calidad del aire



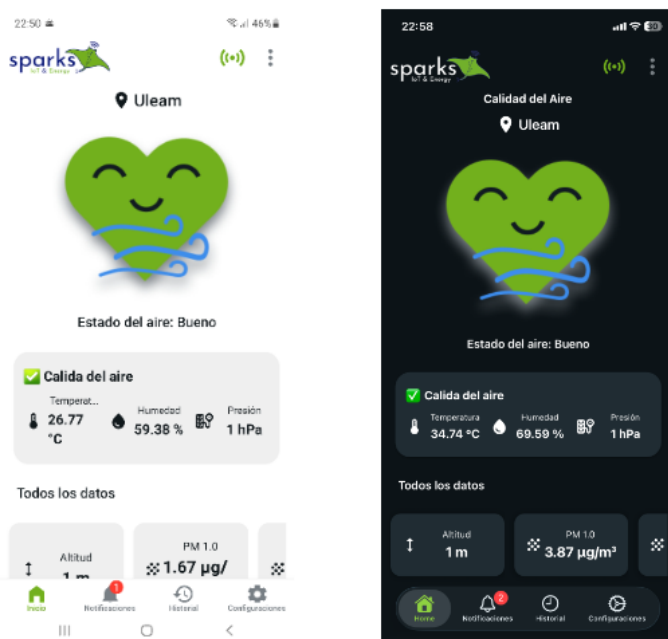
Nota. Elaboración propia de captura de pantalla de una notificación llegando a un dispositivo iOS

5.3.6 Compatibilidad con Android e iOS

La aplicación fue probada exitosamente en dispositivos con sistema operativo Android y iOS, verificando su correcto funcionamiento en ambas plataformas. Esta compatibilidad demuestra la versatilidad de la solución y su capacidad para adaptarse a diferentes entornos móviles.

Figura 23

Aplicación móvil ejecutándose en plataformas Android e iOS.



Nota. Elaboración propia, captura desde un dispositivo Android y otro iOS

5.3.7 Validación del Funcionamiento de la aplicación móvil

Las pruebas realizadas sobre la aplicación móvil confirmaron su correcta integración con el backend y el sistema IoT. Se verificó la recepción de datos en tiempo real, la consulta de información histórica, el funcionamiento de las notificaciones y el cambio adecuado entre los modos de conexión.

Los resultados obtenidos evidencian que la aplicación móvil cumple con los requisitos funcionales establecidos y se presenta como una herramienta confiable para el monitoreo de la calidad del aire.

5.4 Resultados del sistema de notificaciones y eventos

Tras la implementación y ejecución del sistema de notificaciones y gestión de eventos del sistema se presentan los resultados obtenidos. Este componente cumple un rol fundamental, ya que permite informar de manera oportuna a los usuarios sobre situaciones relevantes que pueden representar un riesgo para la salud o requerir atención inmediata.

5.4.1 Notificaciones ante eventos relevantes

Durante las pruebas realizadas, el sistema demostró la capacidad de generar notificaciones automáticas cuando se detectan eventos relevantes relacionados con la calidad del aire. Estos eventos se producen a partir del análisis de los datos recibidos desde los sensores y procesados por el backend.

Entre los eventos considerados se incluyen:

- Incremento significativo en los niveles de material particulado o gases.
- Variaciones anómalas en parámetros ambientales.
- Condiciones que superan los valores normales definidos para el sistema.

Los resultados evidencian que las notificaciones se generan de manera consistente cada vez que se cumple una condición de evento.

5.4.2 Alertas por valores críticos

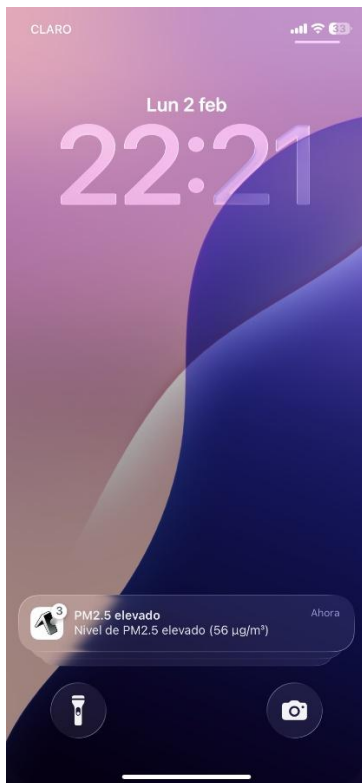
Uno de los principales resultados del sistema es la generación de alertas por valores críticos, las cuales se activan cuando los parámetros de calidad del aire superan umbrales predefinidos. Durante la validación se comprobó que:

- Las alertas se activan correctamente ante valores críticos.
- El contenido de la notificación permite identificar claramente el tipo de evento ocurrido.
- El usuario recibe la alerta de forma inmediata en su dispositivo móvil.

Este comportamiento confirma que el sistema responde adecuadamente ante escenarios de riesgo, contribuyendo a la prevención y concienciación del usuario.

Figura 24

Notificación generada ante evento de calidad del aire



Nota. Elaboración propia, captura de pantalla de una notificación llegando a un dispositivo

5.4.3 Comunicación en tiempo real

Los resultados obtenidos demuestran que el sistema de notificaciones opera bajo un esquema de comunicación en tiempo real, permitiendo que los eventos detectados en el sistema de adquisición de datos se reflejen casi de forma inmediata en la aplicación móvil.

Esta comunicación se logró mediante la integración del backend con los servicios de mensajería, lo que permitió:

- Transmitir eventos sin retrasos perceptibles.
- Mantener una sincronización adecuada entre los componentes del sistema.
- Garantizar la entrega de notificaciones incluso cuando la aplicación se encuentra en segundo plano.

5.4.4 Interacción con la aplicación móvil

El sistema de notificaciones se integra de manera directa con la aplicación móvil, permitiendo una interacción fluida con el usuario. Durante la ejecución del sistema se observó que:

- Al recibir una notificación, el usuario puede acceder a la aplicación para consultar más detalles.
- Las notificaciones refuerzan la experiencia de uso, manteniendo informado al usuario en todo momento.
- La aplicación responde correctamente a los eventos recibidos, mostrando información actualizada.

Estos resultados confirman que el sistema de notificaciones y eventos cumple su función como mecanismo de alerta y comunicación efectiva dentro de la solución propuesta.

5.4.5 Resultados generales del sistema de notificaciones

A partir de los resultados obtenidos, se concluye que el sistema de notificaciones y eventos:

- Funciona de manera estable y confiable.
- Responde oportunamente ante eventos relevantes y críticos.
- Se integra correctamente con la aplicación móvil.

Aporta un valor significativo al sistema al mejorar la capacidad de reacción del usuario.

En consecuencia, el sistema de notificaciones se considera correctamente implementado y validado, cumpliendo con los objetivos planteados para el proyecto.

5.5 Resultados de la conectividad y comunicación

La comunicación entre los dispositivos IoT, el backend y la aplicación móvil se realiza principalmente mediante el protocolo MQTT, el cual demostró ser eficiente y adecuado para sistemas de monitoreo en tiempo real. El sistema fue concebido con un enfoque híbrido de conectividad, permitiendo adaptarse dinámicamente a las condiciones de red disponibles. Esta característica aporta un valor significativo a la solución, ya que asegura la continuidad del servicio incluso en contextos donde la conectividad es limitada o inexistente.

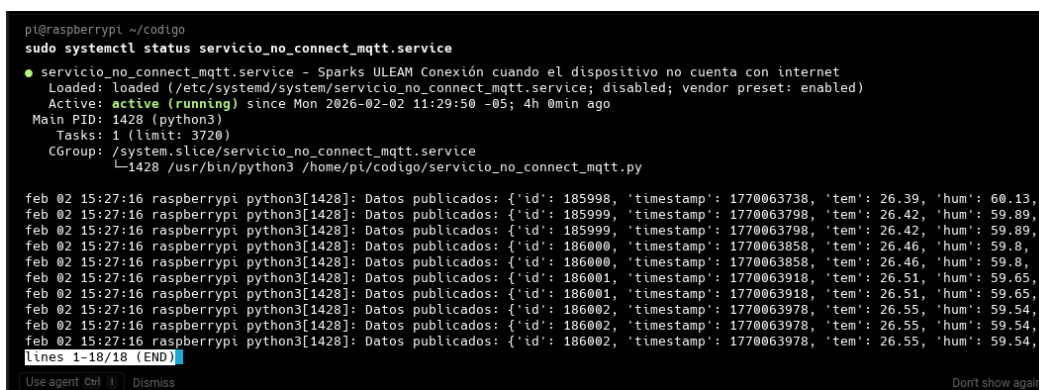
5.5.1 Comunicación mediante el protocolo MQTT

Los resultados obtenidos evidencian que MQTT permitió una transmisión eficiente y confiable de los datos de calidad del aire, gracias a su naturaleza ligera y orientada a mensajes.

Se verificó que los datos publicados por el dispositivo IoT son recibidos correctamente por el broker MQTT y posteriormente consumidos por los servicios correspondientes, sin pérdidas significativas de información. Este comportamiento permitió una actualización constante de los datos visualizados en la aplicación móvil, reforzando el carácter de monitoreo en tiempo real del sistema.

Figura 25

Recepción de mensajes MQTT durante la operación del sistema



```
pi@raspberrypi ~/codigo
sudo systemctl status servicio_no_connect_mqtt.service
● servicio_no_connect_mqtt.service - Sparks ULEAM Conexión cuando el dispositivo no cuenta con internet
   Loaded: loaded (/etc/systemd/system/servicio_no_connect_mqtt.service; disabled; vendor preset: enabled)
   Active: active (running) since Mon 2026-02-02 11:29:50 -05; 4h 0min ago
     Main PID: 1428 (python3)
       Tasks: 1 (limit: 3720)
    CGroup: /system.slice/servicio_no_connect_mqtt.service
            └─1428 /usr/bin/python3 /home/pi/codigo/servicio_no_connect_mqtt.py

feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 185998, 'timestamp': 1770063738, 'tem': 26.39, 'hum': 60.13,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 185999, 'timestamp': 1770063799, 'tem': 26.42, 'hum': 59.89,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 185999, 'timestamp': 1770063799, 'tem': 26.42, 'hum': 59.89,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 186000, 'timestamp': 1770063859, 'tem': 26.46, 'hum': 59.8,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 186000, 'timestamp': 1770063859, 'tem': 26.46, 'hum': 59.8,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 186001, 'timestamp': 1770063918, 'tem': 26.51, 'hum': 59.65,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 186001, 'timestamp': 1770063918, 'tem': 26.51, 'hum': 59.65,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 186002, 'timestamp': 1770063978, 'tem': 26.55, 'hum': 59.54,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 186002, 'timestamp': 1770063978, 'tem': 26.55, 'hum': 59.54,
feb 02 15:27:16 raspberrypi python3[1428]: Datos publicados: {'id': 186002, 'timestamp': 1770063978, 'tem': 26.55, 'hum': 59.54,
lines 1-18/18 (END)
```

Nota. Elaboración propia, captura desde Solar Putty

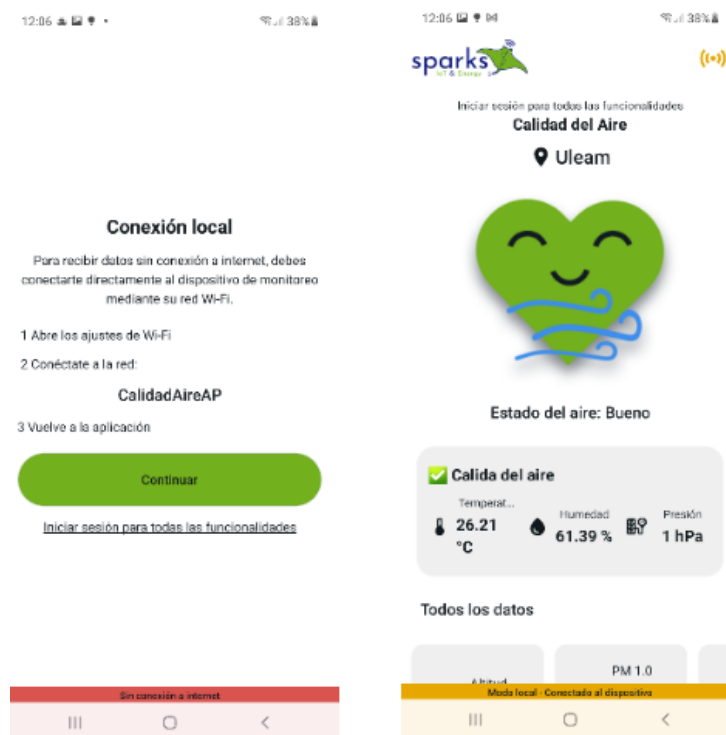
5.5.2 Funcionamiento en red Local

Durante las pruebas realizadas se comprobó que, al conectarse a la red local del dispositivo IoT, la aplicación móvil establece comunicación directa mediante el broker MQTT local.

En este modo de operación, los datos de los sensores son recibidos y visualizados en tiempo real, manteniendo la funcionalidad completa del sistema. Este resultado valida la correcta implementación del modo local y demuestra la viabilidad del sistema en entornos aislados o con conectividad restringida.

Figura 26

Funcionamiento del sistema en modo de conexión local



Nota. Elaboración propia, captura de pantalla desde un dispositivo Android

5.5.3 Funcionamiento en red con acceso a internet

Asimismo, se validó el funcionamiento del sistema en redes con acceso a Internet, donde la comunicación se realiza a través de los servicios desplegados en la nube. En este escenario, el dispositivo IoT envía los datos al backend remoto, el cual los procesa y los pone a disposición de la aplicación móvil mediante los servicios expuestos.

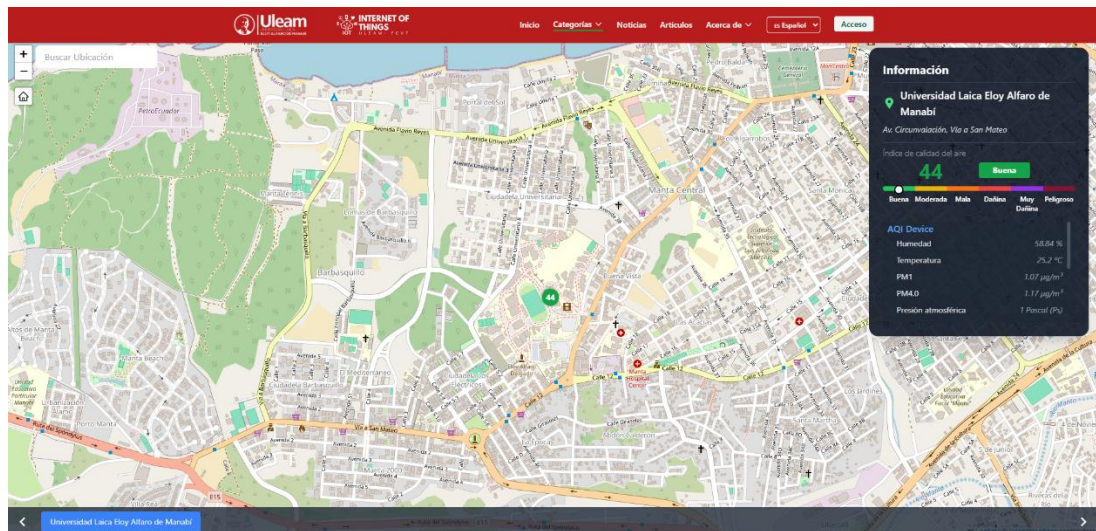
Los resultados muestran que el sistema mantiene un comportamiento estable en este modo de operación, así como lo podemos apreciar en la figura 23, permitiendo la recepción continua de datos y su correcta visualización desde cualquier ubicación con acceso a Internet. Esto confirma la flexibilidad del sistema para operar tanto a nivel local como remoto.

5.5.4 Visualización de datos en sistema de IoT Uleam

Los resultados obtenidos previo a la implementación del broker de IoT Uleam, permiten visualizar los datos de la categoría calidad del aire, se presentan en mapa o en tiempo real. Como podemos apreciar en la figura 27 y 28, se muestran los datos publicados desde el servicio que actualmente se ejecuta dentro de la raspberry pi.

Figura 27

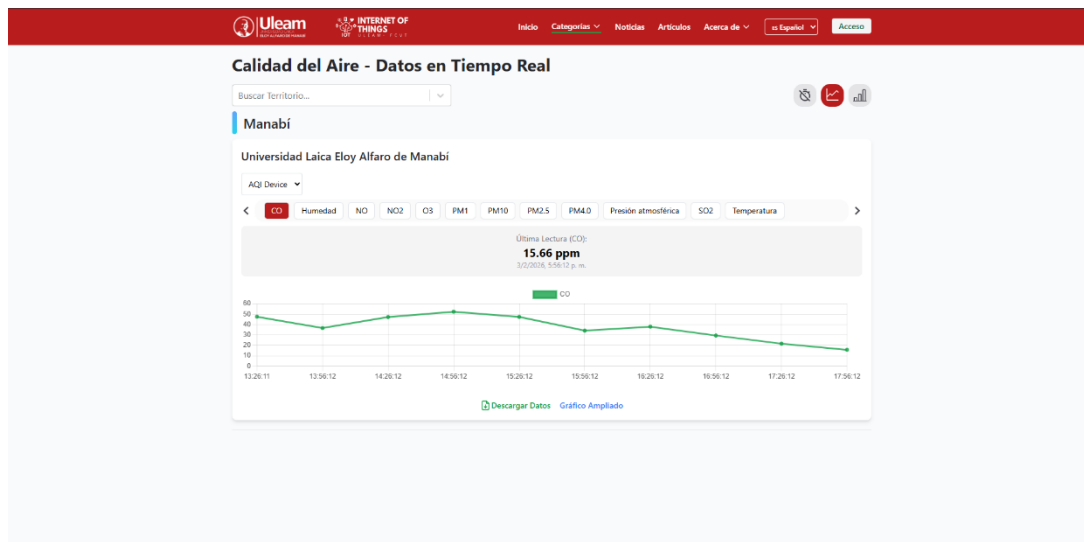
Sistema de IoT Uleam categoría Calidad del aire - mapa



Nota. Elaboración propia, captura de pantalla desde IoT Uleam

Figura 28

Sistema de IoT Uleam categoría Calidad del aire - Datos en tiempo real



Nota. Elaboración propia, captura de pantalla desde IoT Uleam

5.5.5 Estabilidad de la comunicación

Los resultados obtenidos indican que el sistema mantiene una comunicación estable, sin interrupciones frecuentes ni pérdidas de datos significativas, incluso durante la transmisión continua de información.

Además, se observó que el sistema es capaz de gestionar adecuadamente cambios en la conectividad, manteniendo el flujo de datos y evitando fallos críticos en la operación de la

aplicación móvil. Este comportamiento evidencia la robustez del diseño de comunicación implementado.

5.5.6 Recepción continua de datos

La información generada por los sensores es procesada y transmitida sin retrasos perceptibles, lo que se refleja en una actualización constante de los valores mostrados al usuario.

Este resultado es fundamental para un sistema de monitoreo, ya que garantiza que la información presentada sea actual y confiable, permitiendo al usuario conocer el estado de la calidad del aire en todo momento.

5.5.7 Comparación entre conexión local y remota

A partir de los resultados obtenidos, se realizó una comparación entre el funcionamiento del sistema en modo local y en modo remoto, con el fin de analizar su comportamiento en distintos escenarios de conectividad. En la tabla 9, podemos analizar las características de los modos de conexión.

Tabla 9

Comparación entre conexión local y conexión remota

Característica	Conexión local	Conexión remota
Dependencia de Internet	No	Sí
Latencia	Baja	Media
Estabilidad	Alta	Alta
Recepción de datos	En tiempo real	En tiempo real
Uso en entornos aislados	Sí	No

Nota. Elaboración propia

5.5.8 Resultados generales de la conectividad

En base a los resultados obtenidos, se concluye que el sistema de conectividad y comunicación cumple de manera satisfactoria con los objetivos planteados. La solución demuestra ser flexible, robusta y capaz de adaptarse a diferentes condiciones de red, garantizando la continuidad del monitoreo de la calidad del aire.

Este comportamiento reafirma la validez del enfoque híbrido adoptado y posiciona al sistema como una solución práctica y aplicable en escenarios reales.

5.6 Validación de los requisitos del sistema

La validación de los requisitos permite evidenciar, de forma clara y estructurada, que el sistema responde a las necesidades identificadas y que su funcionamiento es coherente con los objetivos del proyecto.

Para este análisis, los requisitos se clasifican en funcionales y no funcionales, indicando su estado de cumplimiento y la evidencia que respalda cada resultado. Esta validación se realizó a partir de las pruebas ejecutadas, la observación directa del sistema en funcionamiento y los resultados presentados en los apartados anteriores de este capítulo.

5.6.1 Validación de los requisitos funcionales

A continuación, se presenta la validación de dichos requisitos en función de los resultados obtenidos durante la implementación y pruebas del sistema.

Tabla 10

Validación de requisitos funcionales

Código	Requisito funcional	Estado de cumplimiento	Evidencia
RF-SIS-01	Visualizar datos de calidad del aire en tiempo real desde la aplicación móvil	Cumple	Sección 5.3.2 – Visualización en tiempo real
RF-SIS-02	Consultar datos históricos de calidad del aire	Cumple	Sección 5.3.3 – Consulta histórica
RF-SIS-03	Permitir la autenticación de usuarios en la aplicación móvil	Cumple	Sección 5.3 – Resultados de la aplicación móvil

RF-SIS-04	Recibir datos desde el dispositivo IoT mediante MQTT	Cumple	Sección 5.5.1 – Comunicación MQTT
RF-SIS-05	Operar el sistema en modo local sin conexión a Internet	Cumple	Sección 5.5.2 – Funcionamiento en red local
RF-SIS-06	Operar el sistema en modo remoto con acceso a Internet	Cumple	Sección 5.5.3 – Funcionamiento en red remota
RF-SIS-07	Generar notificaciones ante eventos relevantes	Cumple	Sección 5.4 – Sistema de notificaciones
RF-SIS-08	Mostrar alertas por valores críticos de calidad del aire	Cumple	Sección 5.4.2 – Alertas por valores críticos

Nota. Elaboración propia

A partir de los resultados obtenidos, se evidencia que los requisitos funcionales definidos fueron implementados y validados satisfactoriamente, permitiendo que el sistema cumpla con las funciones principales para las cuales fue diseñado.

5.6.2 Validación de los requisitos no funcionales

Los requisitos no funcionales están relacionados con las características de calidad del sistema, tales como rendimiento, disponibilidad, seguridad y usabilidad. La validación de estos requisitos se realizó mediante la observación del comportamiento del sistema durante su ejecución y las pruebas realizadas en distintos escenarios.

Tabla 11

Validación de requisitos no funcionales

Código	Requisito no funcional	Estado de cumplimiento	Evidencia
RNF-SIS-01	Disponibilidad del sistema	Cumple	Sección 5.5 – Funcionamiento en red local y remota

RNF-SIS-02	Seguridad de la información	Cumple	Sección 4.3.10 y 5.4 – Seguridad y autenticación
RNF-SIS-03	Tiempo de respuesta adecuado	Cumple	Secciones 5.3 y 5.5 – Visualización y comunicación
RNF-SIS-04	Facilidad de uso de la aplicación	Cumple	Sección 5.3.1 – Interfaz y experiencia de usuario
RNF-SIS-05	Escalabilidad del sistema	Cumple parcialmente	Sección 4.3.6 – Diseño arquitectónico
RNF-SIS-06	Compatibilidad con Android iOS	Cumple	Sección 5.3.6 – Compatibilidad multiplataforma
RNF-SIS-07	Integridad de los datos	Cumple	Secciones 5.2 y 5.5 – Backend y comunicación

Nota. Elaboración propia

5.6.3 Análisis general de la validación de requisitos

El análisis global de la validación permite concluir que la solución desarrollada cumple de manera satisfactoria tanto con los requisitos funcionales como con los no funcionales definidos en la etapa de planificación. No se identificaron requisitos que no hayan sido cumplidos, y el sistema demostró un comportamiento acorde a lo esperado durante las pruebas realizadas.

Esta validación confirma que la propuesta tecnológica desarrollada responde adecuadamente al problema planteado y que el sistema se encuentra en condiciones de ser utilizado como una herramienta efectiva para el monitoreo de la calidad del aire.

5.7 Análisis del desempeño del sistema

Este análisis se basa en la observación directa del funcionamiento del sistema, las pruebas realizadas en los distintos módulos y la interacción continua entre el dispositivo IoT, el backend y la aplicación móvil.

A continuación, se describen los principales aspectos evaluados en relación con el desempeño del sistema.

5.7.1 *Tiempo de respuesta*

Los datos capturados por los sensores fueron transmitidos y procesados de forma casi inmediata, permitiendo su visualización en la aplicación móvil sin retrasos perceptibles para el usuario. Tanto en el modo de conexión local como en el modo remoto, la aplicación mostró los datos de manera oportuna, lo que indica que el tiempo de respuesta del sistema es consistente con los objetivos planteados para la solución.

5.7.2 *Estabilidad del sistema*

No se presentaron fallos críticos ni interrupciones que impidieran el funcionamiento normal del sistema durante las pruebas realizadas.

Asimismo, el sistema fue capaz de mantener su operación frente a cambios en la conectividad, adaptándose de forma adecuada a los distintos escenarios de red. Este comportamiento evidencia que la arquitectura implementada contribuye a la estabilidad general del sistema.

5.7.3 *Manejo de datos en tiempo real*

Los resultados muestran que el sistema es capaz de recibir, procesar y visualizar los datos de calidad del aire de manera continua, sin pérdidas significativas de información.

La integración entre el protocolo MQTT, el backend y la aplicación móvil permitió una actualización constante de los valores presentados al usuario, reforzando la utilidad del sistema como herramienta de monitoreo en tiempo real.

5.7.4 *Comportamiento general del sistema*

La interacción entre los distintos componentes se realizó de forma fluida, permitiendo cumplir con las funciones principales del sistema sin afectar la experiencia del usuario.

El desempeño observado durante las pruebas confirma que la solución desarrollada es funcional, estable y adecuada para su aplicación en un entorno real, cumpliendo con los objetivos establecidos para el proyecto.

5.8 *Discusión de resultados*

Desde el inicio del proyecto, se planteó como problema la dificultad para acceder a información confiable, continua y en tiempo real sobre la calidad del aire, especialmente en contextos donde la conectividad a Internet puede ser limitada. Frente a esta situación, se

propuso el desarrollo de un sistema basado en tecnologías IoT que permita la adquisición, transmisión y visualización de datos ambientales de forma flexible y accesible.

Comparación entre lo esperado y lo logrado

Al comparar los resultados esperados con los resultados obtenidos, se evidencia que el sistema desarrollado cumple de manera satisfactoria con lo planteado inicialmente. Se esperaba contar con una solución capaz de monitorear la calidad del aire en tiempo real, almacenar información histórica y notificar al usuario ante eventos relevantes, y estos objetivos fueron alcanzados durante la implementación y validación del sistema.

Asimismo, uno de los aspectos más importantes considerados desde la etapa de planificación fue la capacidad del sistema para operar tanto en modo local como en modo remoto. Los resultados obtenidos demuestran que esta funcionalidad se implementó correctamente, permitiendo la continuidad del monitoreo incluso en escenarios sin acceso a Internet, lo cual refuerza el cumplimiento de los objetivos del proyecto.

Aportes del sistema desarrollado

El sistema desarrollado aporta una solución tecnológica funcional y adaptable para el monitoreo de la calidad del aire, integrando de manera efectiva componentes de hardware y software. Entre los principales aportes se destacan:

- La integración de un dispositivo IoT para la adquisición de datos ambientales.
- El uso de un backend centralizado que permite procesar y gestionar la información.
- La implementación de una aplicación móvil multiplataforma como interfaz principal para el usuario.
- La incorporación de un sistema de notificaciones que mejora la capacidad de respuesta ante eventos relevantes.
- La posibilidad de operar en distintos escenarios de conectividad, lo que incrementa la aplicabilidad del sistema.

Estos aportes permiten que la solución no solo cumpla con los requisitos técnicos, sino que también resulte útil y práctica en contextos reales.

Valor del proyecto

El valor del proyecto radica en su enfoque aplicado y en la solución concreta que ofrece frente a un problema real. El sistema desarrollado demuestra que es posible implementar una

herramienta de monitoreo de la calidad del aire utilizando tecnologías accesibles, manteniendo un equilibrio entre funcionalidad, usabilidad y adaptabilidad.

Desde el punto de vista académico, el proyecto permitió aplicar los conocimientos adquiridos durante la formación universitaria en un caso práctico, integrando conceptos de IoT, desarrollo de software, comunicaciones y sistemas distribuidos. Desde el punto de vista social y tecnológico, la solución propuesta contribuye a la concienciación sobre la calidad del aire y sienta las bases para futuras mejoras o ampliaciones del sistema.

En conjunto, los resultados obtenidos validan la propuesta desarrollada y evidencian que el proyecto cumple con los objetivos planteados, aportando una solución viable y funcional al problema identificado.

5.9 Limitaciones identificadas

Durante el desarrollo e implementación del sistema de monitoreo de la calidad del aire se identificaron ciertas limitaciones, las cuales están relacionadas principalmente con aspectos técnicos, de infraestructura y de alcance del proyecto. Estas limitaciones no invalidan los resultados obtenidos, pero es importante mencionarlas para contextualizar adecuadamente el funcionamiento y los alcances de la solución desarrollada.

Dependencia de la conectividad

Si bien el sistema fue diseñado para operar tanto en modo local como en modo remoto, su funcionamiento depende en cierta medida de la disponibilidad de una red de comunicación. En escenarios donde la conectividad es inestable o inexistente, algunas funcionalidades, como la sincronización de datos con el backend en la nube o el acceso remoto a la información, pueden verse limitadas.

Limitaciones de Hardware

El sistema utiliza hardware específico para la adquisición de los datos ambientales, lo cual implica ciertas restricciones en cuanto a capacidad de procesamiento, almacenamiento y disponibilidad de sensores. Estas limitaciones pueden influir en la frecuencia de muestreo de los datos o en la posibilidad de ampliar el número de variables monitoreadas sin realizar modificaciones adicionales al hardware utilizado.

Alcances del proyecto

El proyecto fue desarrollado con un enfoque académico y aplicado, por lo que su alcance se limita a la implementación de un sistema funcional de monitoreo de la calidad del aire. No se contemplaron aspectos como la implementación a gran escala, la integración con sistemas gubernamentales o la validación mediante estudios ambientales de largo plazo, los cuales podrían ser considerados en trabajos futuros.

Posibles líneas de mejora

Durante el desarrollo del proyecto se identificaron oportunidades de mejora relacionadas con la ampliación de funcionalidades, optimización del rendimiento y fortalecimiento de la seguridad del sistema. No obstante, estas mejoras no fueron abordadas en el presente trabajo, ya que exceden el alcance definido inicialmente dando la posibilidad a aplicar estas mejoras en una segunda versión o actualización de la app.

CAPÍTULO VI

6 Conclusiones

El desarrollo del presente proyecto permitió cumplir con el objetivo principal de diseñar e implementar un sistema de monitoreo de la calidad del aire basado en tecnologías IoT, capaz de adquirir, procesar y visualizar datos ambientales de manera confiable y accesible. A lo largo del trabajo se logró integrar de forma efectiva componentes de hardware y software, dando como resultado una solución funcional y operativa en escenarios reales.

El sistema desarrollado demostró ser capaz de recopilar datos desde un dispositivo IoT, transmitirlos mediante protocolos adecuados, almacenarlos de forma segura y presentarlos al usuario final a través de una aplicación móvil multiplataforma. La correcta integración entre el dispositivo de adquisición, el backend, la base de datos, el broker de mensajería y la aplicación móvil evidencia la viabilidad técnica de la propuesta planteada.

Uno de los aspectos más relevantes del proyecto es la capacidad del sistema para operar tanto en modo local como en modo remoto, lo cual permite mantener el monitoreo de la calidad del aire incluso en entornos con conectividad limitada. Esta característica fortalece la aplicabilidad del sistema y responde directamente al problema identificado en el capítulo inicial.

Asimismo, la implementación de un sistema de notificaciones y eventos permitió alertar al usuario ante situaciones relevantes, mejorando la utilidad práctica de la solución y aportando un valor adicional en términos de prevención y concienciación ambiental. Los resultados obtenidos durante las pruebas y validaciones confirman que el sistema cumple con los requisitos funcionales y no funcionales definidos, presentando un comportamiento estable y coherente con los objetivos del proyecto.

Desde el punto de vista académico, el proyecto permitió aplicar de manera práctica los conocimientos adquiridos durante la formación universitaria, integrando conceptos relacionados con IoT, desarrollo de aplicaciones móviles, backend, bases de datos y comunicaciones. En conjunto, los resultados alcanzados permiten concluir que la propuesta desarrollada constituye una solución válida y funcional para el monitoreo de la calidad del aire.

7 Recomendaciones

Como resultado del análisis realizado y de la experiencia obtenida durante el desarrollo del proyecto, se identifican varias recomendaciones que pueden servir como base para futuros trabajos o mejoras del sistema.

Se recomienda ampliar el sistema para incorporar nuevos sensores que permitan medir un mayor número de variables ambientales, lo que contribuiría a obtener un análisis más completo de la calidad del aire. De igual forma, podría considerarse la implementación de mecanismos avanzados de análisis de datos, como la generación de reportes automáticos o el uso de técnicas de análisis predictivo.

Otra recomendación importante es fortalecer los mecanismos de seguridad del sistema, incorporando configuraciones adicionales de protección en la comunicación y en el acceso a los servicios, especialmente en escenarios de despliegue a mayor escala. Asimismo, se podría optimizar la gestión de la base de datos para soportar un mayor volumen de información sin afectar el rendimiento del sistema.

Desde el punto de vista de la aplicación móvil, se recomienda continuar mejorando la interfaz de usuario y la experiencia de uso, incorporando nuevas funcionalidades que faciliten la interpretación de los datos y el acceso a la información histórica. También sería recomendable realizar pruebas con un mayor número de usuarios para obtener retroalimentación que permita perfeccionar el sistema.

Finalmente, se sugiere considerar la implementación del sistema en entornos reales de mayor alcance, como instituciones educativas o espacios públicos, con el fin de evaluar su comportamiento a largo plazo y su impacto en la concienciación sobre la calidad del aire.

BIBLIOGRAFÍA

- Acosta, J., León, A., & Sanafria, W. (2022). Las aplicaciones móviles y su impacto en la sociedad. *Revista Universidad y Sociedad*, 4(2). doi:http://scielo.sld.cu/scielo.php?script=sci_isoref&pid=S2218-36202022000200237&lng=es&tlng=en
- Alay, M., Mendez, M., & Erazo, O. (2023). Tecnología de Internet de las Cosas en el monitoreo de cultivos agrícolas. *Internet of things technology in agricultural crop monitoring*, 4(3), 69-93. doi:<https://doi.org/10.35290/ro.v4n3.2023.939>
- Alipour, N., Wang, M., & Krishnamurthy, D. (2021). Oasis: Performance Matching IoT System Emulation,. En 2. I. (CLOUD), *2021 IEEE 14th International Conference on Cloud Computing (CLOUD)* (págs. 378-386). IEEE. doi:10.1109/CLOUD53861.2021.00051.
- Amoedo, E. (2013). *Principios de diseño de APIs REST*. Obtenido de https://d1wqtxts1xzle7.cloudfront.net/43461896/introduccion_apis_rest-sample-libre.pdf?1457378275=&response-content-disposition=inline%3B+filename%3DIntroduccion_apis_rest_sample.pdf&Expires=1770158522&Signature=JYH2re1WhScao96BfVT2mWFiepQGJ5per6sx4U~cHHY
- Campbell, M., Aspray, W., Yost, J., Tinn, H., & Díaz, G. (2023). *Computer*. New York: Taylor & Francis Group. doi:<https://doi.org/10.4324/9781003263272>
- Catiblanco Avedaño, C., & Cañón Alfonso, N. (2019). *PROTOTIPO DE BAJO COSTO PARA MONITOREO DE CALIDAD DEL AIRE EN. UNIVERSIDAD PILOTO DE COLOMBIA. UNIPILOTO*. Obtenido de <http://repository.unipiloto.edu.co/handle/20.500.12277/4880>
- EPA. (23 de Junio de 2025). *Agencia de protección ambiental de Estados Unidos*. Obtenido de Por qué la calidad del aire de los interiores es importante para las escuelas.
- González, J. (2023). El Internet de las Cosas (IoT), la Ciudad Inteligente(SmartCity) y el Medio Ambiente. En U. d. Huelva, *Fomento de la Cultura Científica, Tecnológica y de Innovación en Ciudades Inteligentes* (págs. 58-61). Dialnet.
- Grant, A., & Owens, M. (2010). *The definitive guide to SQLite*. Apress. doi:<https://izbicki.me/public/cs/databases-lectures/SQLite.pdf>
- Greer, F. (19 de Junio de 2023). *6 tipos de tecnología IoT inalámbrica*. Obtenido de Zipit: <https://www.zipitwireless.com/blog/5-types-of-wireless-iot-technology>

- Guananga Narváez, E., & Jaya Duche, M. (2022). *Comparación de protocolos de comunicación para internet de las cosas (IoT)*. Universidad Politécnica Salesiana. doi:<http://dspace.ups.edu.ec/handle/123456789/23315>
- Gubbi, J., Buyya, R., Marusic, S., & Palaniswai, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7), 3-27. doi:<https://doi.org/10.1016/j.future.2013.01.010>
- IBM. (24 de Abril de 2025). *¿Qué es la API REST?* . Obtenido de IBM: <https://www.ibm.com/es-es/think/topics/rest-apis>
- Kumar, Morawska, & Martuzzi. (2022). Real-time air quality monitoring systems and their role in public health. *Environmental Monitoring and Technology Journal*, 112-129.
- Llamocca, P. (2016). Integración y Visualización de Datos Abiertos. *Trabajo Fin de Máster en Ingeniería de Computadores*. UNIVERSIDAD COMPLUTENSE DE MADRID, Madrid. doi:<https://hdl.handle.net/20.500.14352/25145>
- MDN. (2025). *Introducción al desarrollo web del lado del servidor*. Obtenido de Mozilla Developer Network: https://developer.mozilla.org/es/docs/Learn_web_development/Extensions/Server-side
- Mercado, R. (2023). Bases de datos en aplicaciones móviles para sistema operativo Android. *Dialnet*, 10(1), 17-37. doi:<https://dialnet.unirioja.es/servlet/articulo?codigo=9274840>
- Momjian, B. (2004). *PostgreSQL Introduccion y conceptos*. Addison Wesley. doi:<https://momjian.us/main/writings/pgsql/other/bookfigs.pdf>
- Mora Olivedo, A., Macías Lara, R., Rodríguez Vizueté, J., & Sacón Klinger, H. (2021). Estudio de la tecnología de comunicación inalámbrica en el estándar IEEE 802.11ax orientada al despliegue en Ecuador para el desarrollo del internet de las cosas. *Dominio de las ciencias*, 7, 12. doi:<http://dx.doi.org/10.23857/dc.v7i4.2447>
- Moreira Romero, Á. (2018). Contaminación del aire en el medio ambiente por las emisiones de gases tóxicos de empresas industriales en Ecuador. *Revista multidisciplinaria de innovación y estudios aplicados*, 3, 1- 8. doi: <https://www.polodelconocimiento.com/>
- Moreira Sánchez, W., Cruz Felipe, M., Coterá Ramírez, G., & Medranda Cobeña, G. (2025). *Herramientas y plataformas de procesamiento: Un análisis sistemático en el contexto de IoT y Big Data*. Revista ODIGOS. doi:https://www.researchgate.net/profile/Wilmer-Moreira-3/publication/390670619_Herramientas_y_plataformas_de_procesamiento_Un_analisis

- is_sistematico_en_el_contexto_de_IoT_y_Big_Data_Processing_tools_and_platforms
_A_systematic_analysis_in_the_context_of_IoT_and
- OMS. (24 de Octubre de 2024). *Organización Mundial de la salud*. Obtenido de Contaminación del aire ambiente (exterior) y salud: [https://www.who.int/es/news-room/fact-sheets/detail/ambient-\(outdoor\)-air-quality-and-health](https://www.who.int/es/news-room/fact-sheets/detail/ambient-(outdoor)-air-quality-and-health)
- Ortiz Zambrano, J., Chavez Cujilan, Y., Toapanta Bernabe, M., & Lino Castillo, K. (2017). La usabilidad y accesibilidad: Estudio de guías para aplicaciones en dispositivos. *Dominio de las ciencias*, 3, 1181 - 1209. doi:<http://dx.doi.org/10.23857/dom.cien.pocaip.2017.3.3.jun.1181-1209>
- Pacheco, A., Escobar, J., & Trujillo, E. (2021). USO DE PATRONES DE DISEÑO Y METAPROGRAMACIÓN. *División de Estudios de Posgrado e Investigación*. Instituto Tecnológico de Chihuahua, Chihuahua. doi:https://www.itchihuahua.mx/revista_electro/2021/C-Sub32.pdf
- Paguay, P., Paucar, R., & Flores, A. (2019). Evaluación del rendimiento de la transferencia de imágenes en aplicaciones IoT entre los lenguajes de programación Java y Python. *Ciencia Digital, desarrollo y crecimiento*, 3(3.4), 24-84. doi:<https://doi.org/10.33262/cienciadigital.v3i3.4..878>
- Pascual, P. (2024). Análisis y visualización de datos sobre. *Escuela Técnica Superior de*. Universidad Politécnica, Madrid. doi:https://oa.upm.es/83184/1/TFG_SERGIO_PASCUAL_PINO.pdf
- Pincay Ponce, F., & García Silva, J. (2024). *TECNOLOGÍAS DE INTERNET DE LAS COSAS APLICADAS A LOS SERVICIOS URBANOS DE LA CIUDAD DE JIPIJAPA*. UNESUM. Obtenido de <http://repositorio.unesum.edu.ec/handle/53000/6372>
- Red Hat. (31 de Julio de 2023). *¿Qué es una API de REST?* Obtenido de Red Hat: <https://www.redhat.com/es/topics/api/what-is-a-rest-api>
- Rodriguez Ortiz, G. A. (2024). *PLATAFORMA IOT PARA EL MONITOREO Y EVALUACION DE LA CALIDAD*. UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA. UNAB. doi:<http://hdl.handle.net/20.500.12749/25223>
- Salinas, Y., Galvan, D., Guzman, I., & Orrante, J. (2022). El impacto del internet de todas las cosas (IoT) en la vida cotidiana. *Revista multidisciplinaria*, 6(2), 1369-1378. doi:https://doi.org/10.37811/cl_rcm.v6i2.1959
- Segura, A. A. (2016). Arquitectura de Software de Referencia para Objetos Inteligentes en Internet de las Cosas. *Revista Latinoamericana de ingeniería de software*, 2, 73 - 110. doi:<https://doi.org/10.18294/relais.2016.73-110>

UNESCO. (2022). What you need to know about access to information. *International day for universal access to information* . Obtenido de <https://www.unesco.org/en/days/universal-access-information>

Zamora, W., Machuca, M., Zambrano, J. L., & Ponce, M. (2023). SparksAIR: A Low-Cost IoT Solution to Manage Air Pollution in Urban Areas. En S. N. Link, *International Conference on Applied Technologies* (págs. 105-116). Springer, Cham.

ANEXOS

Figura 29

Diseño de la app en Figma

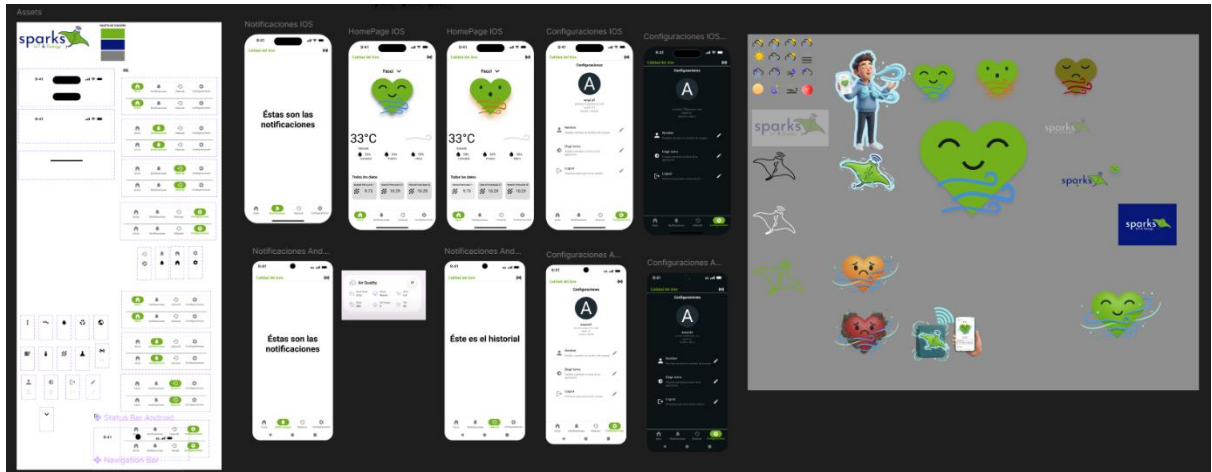


Figura 30

Revisión del correcto funcionamiento del sistema



Figura 31

Sprints en Notion para gestionar y monitorear el proceso de las tareas

