



**UNIVERSIDAD NACIONAL DE CHIMBORAZO  
FACULTAD DE INGENIERÍA  
CARRERA DE ELECTRÓNICA Y TELECOMUNICACIONES**

Implementación de un sistema para el monitoreo de parámetros ambientales en tiempo real, basado en IoT Y LoRaWAN en una plataforma Cloud.

**Trabajo de Titulación para optar al título de Ingeniero en Electrónica y Telecomunicaciones**

**Autor:**

Tirado Rodríguez, Karen Alexandra

**Tutor:**

Mgs. Cuzco Silva, Edgar Giovanni

**Riobamba, Ecuador. 2025**

## DERECHO DE AUTORÍA

Yo, **Karen Alexandra Tirado Rodríguez**, con cédula de ciudadanía **155011162-7**, autor del trabajo de investigación titulado: **“Implementación de un sistema para el monitoreo de parámetros ambientales en tiempo real, basado en IoT y LoRaWAN en una plataforma Cloud”**, certifico que la producción, ideas, opiniones, criterios, contenidos y conclusiones expuestas son de mí exclusiva responsabilidad.

Asimismo, cedo a la Universidad Nacional de Chimborazo, en forma no exclusiva, los derechos para su uso, comunicación pública, distribución, divulgación y/o reproducción total o parcial, por medio físico o digital; en esta cesión se entiende que el cesionario no podrá obtener beneficios económicos. La posible reclamación de terceros respecto de los derechos de autora de la obra referida, será de mi entera responsabilidad; librando a la Universidad Nacional de Chimborazo de posibles obligaciones.

En Riobamba, el 23 de junio del 2025.



Karen Alexandra Tirado Rodríguez

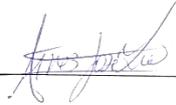
**C.I:** 155011162-7

## **DICTAMEN FAVORABLE DEL TUTOR Y MIEMBROS DEL TRIBUNAL**

Quienes suscribimos, catedráticos designados Tutor y Miembros del Tribunal de Grado para la evaluación del trabajo de investigación “**Implementación de un sistema para el monitoreo de parámetros ambientales en tiempo real, basado en IoT y LoRaWAN en una plataforma Cloud**”, presentado por **Karen Alexandra Tirado Rodríguez**, con cédula de identidad número **155011162-7**, certificamos que recomendamos la **APROBACIÓN** de este con fines de titulación. Previamente se ha asesorado durante el desarrollo, revisado y evaluado el trabajo de investigación escrito y escuchada la sustentación por parte de su autor; no teniendo más nada que observar.

De conformidad a la normativa aplicable firmamos, en Riobamba el 23 de junio del 2025.

Ing. José Luis Jinez  
**PRESIDENTE DEL TRIBUNAL DE GRADO**



---

PhD. Pedro Fernando Escudero  
**MIEMBRO DEL TRIBUNAL DE GRADO**



---

Mgs. Klever Hernán Torres  
**MIEMBRO DEL TRIBUNAL DE GRADO**



---

Dr. Edgar Giovanni Cuzco Silva  
**TUTOR**



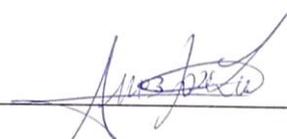
---

## **CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL**

Quienes suscribimos, catedráticos designados Miembros del Tribunal de Grado para la evaluación del trabajo de investigación “**Implementación de un sistema para el monitoreo de parámetros ambientales en tiempo real, basado en IoT y LoRaWAN en una plataforma Cloud**” presentado por **Karen Alexandra Tirado Rodríguez**, con cédula de identidad número **155011162-7**, bajo la tutoría del Mgs. **Edgar Giovanni Cuzco Silva**; certificamos que recomendamos la **APROBACIÓN** de este con fines de titulación. Previamente se ha evaluado el trabajo de investigación y escuchada la sustentación por parte de su autor; no teniendo más nada que observar.

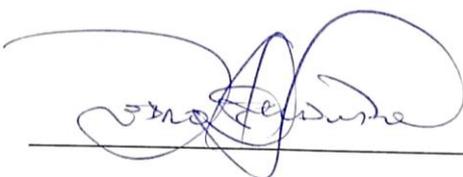
De conformidad a la normativa aplicable firmamos, en Riobamba el 23 de junio del 2025.

**Ing. José Luis Jinez**  
**Presidente del Tribunal de Grado**



---

**PhD. Pedro Fernando Escudero**  
**Miembro del Tribunal de Grado**



---

**Mgs. Klever Hernán Torres**  
**Miembro del Tribunal de Grado**

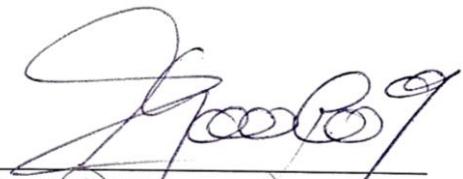


---

## **CERTIFICADO ANTIPLAGIO**

Que, **TIRADO RODRIGUEZ KAREN ALEXANDRA** con CC: **1550111627**, estudiante de la Carrera **ELECTRÓNICA Y TELECOMUNICACIONES, NO VIGENTE**, Facultad de **INGENIERÍA**; ha trabajado bajo mi tutoría el trabajo de investigación titulado ” **IMPLEMENTACIÓN DE UN SISTEMA PARA EL MONITOREO DE PARÁMETROS AMBIENTALES EN TIEMPO REAL, BASADO EN IoT Y LoRaWAN EN UNA PLATAFORMA CLOUD**”, cumple con el 10 %, de acuerdo al reporte del sistema Anti plagio **COMPILATIO MAGISTER**, porcentaje aceptado de acuerdo a la reglamentación institucional, por consiguiente autorizo continuar con el proceso.

Riobamba, 23 de junio de 2025



---

Mgs. Edgar Giovanni Cuzco Silva  
**TUTOR TRABAJO DE INVESTIGACIÓN**

## DEDICATORIA

A mi Dios eterno, en el inicio de este trabajo, elevo mi gratitud hacia Ti, fuente de sabiduría y entendimiento. Con humildad, reconozco que, sin tu guía y fortaleza, este logro no habría sido posible. Tú has sido mi roca y mi inspiración a lo largo de este arduo camino académico. Cada paso dado, cada obstáculo superado, ha estado marcado por tu inquebrantable amor y misericordia.

Gracias, mi amado Señor, en medio de las dudas y los momentos de debilidad, encontré consuelo en tu promesa de que todo, obra para bien a aquellos que te aman. Esta tesis no solo representa un logro personal, sino también un recordatorio de que, con fe en Ti, todas las cosas son posibles.

A mis padres, por su inmovible soporte, sabiduría y amor incondicional. Gracias por creer en mí en cada paso del camino y enseñarme el valor de la educación, por brindarme todo su apoyo para alcanzar mis metas. Cada paso en este difícil camino lo he dado con la certeza de que contaba con su constante presencia y su apoyo. Este logro es el resultado de su sacrificio y dedicación, y les dedico este trabajo con profundo agradecimiento.

A mis queridos hermanos, en este momento de celebración, quiero expresarles mi profundo agradecimiento y amor a través de esta dedicatoria. Su complicidad, su amistad y su constante respaldo han enriquecido mi vida y mi camino académico.

A ti Cristyan, esta tesis de grado es el fruto de mi esfuerzo, pero también del inmenso apoyo que has brindado en cada etapa de mi formación académica. Tu presencia inmutable, tu amor absoluto y tu apoyo desinteresado han sido la fuerza que me ha impulsado a alcanzar mis metas.

A usted MsC. Raúl U. Sanchez M, cuya orientación, conocimiento y ánimo han sido fundamentales en mi formación académica. Sus enseñanzas han iluminado mi camino y enriquecido mi desarrollo intelectual. Agradezco su compromiso y guía a lo largo de esta travesía educativa.

A mis amigos, por su constante soporte, comprensión y aliento en los momentos desafiantes. Agradezco su presencia ilimitada y su alegría, la cual ha hecho más ligero este camino.

## **AGRADECIMIENTO**

Quiero expresar mi más profundo agradecimiento a todas las personas e instituciones que hicieron posible la realización de esta tesis de grado, cuya dedicación y apoyo fueron fundamentales para el desarrollo de mi proyecto.

Agradezco al MsC. Raúl U. Sanchez M, por su invaluable apoyo y generosidad al brindarme acceso al Laboratorio de física de la Universidad Nacional de Chimborazo (UNACH) para la calibración de los sensores, proporcionarme sensores industriales y científicos de alta precisión, permitirme realizar mediciones iniciales en la terraza del bloque B y facilitar la calibración y toma de datos de la estación meteorológica de la universidad. Su apertura y compromiso fueron esenciales para garantizar la calidad de este trabajo.

A la Escuela Superior Politécnica de Chimborazo (ESPOCH), en especial al PhD. Arquimidez Haro y al PhD. Romel Palaguachi, por su disposición y colaboración al permitirme ubicar mi estación LoRaWan junto a la estación meteorológica Lizarzaburu. Su apoyo en la evaluación estadística de los sensores y la entrega de los datos correspondientes al período de medición conjunta fueron cruciales para validar los resultados de este proyecto.

Asimismo, extiendo mi gratitud al Ing. Jorge Pulgar, encargado del Auditorio del bloque L, por su amabilidad y autorización para instalar mi estación LoRaWan en la terraza junto al Auditorio, lo que permitió realizar las mediciones de señal necesarias para evaluar el rendimiento del sistema.

Este trabajo no habría sido posible sin el aporte de cada uno de ustedes, cuya confianza y colaboración fueron un pilar fundamental en este proceso. Mi más sincero agradecimiento por su tiempo, conocimientos y compromiso.

# ÍNDICE GENERAL

<b>DERECHO DE AUTORÍA .....</b>	<b>.....</b>
<b>DICTAMEN FAVORABLE DEL TUTOR Y MIEMBROS DEL TRIBUNAL .....</b>	<b>.....</b>
<b>CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL .....</b>	<b>.....</b>
<b>CERTIFICADO ANTIPLAGIO.....</b>	<b>.....</b>
<b>DEDICATORIA.....</b>	<b>.....</b>
<b>AGRADECIMIENTO .....</b>	<b>.....</b>
<b>ÍNDICE GENERAL.....</b>	<b>.....</b>
<b>INDICE DE TABLAS.....</b>	<b>.....</b>
<b>ÍNDICE DE FIGURAS.....</b>	<b>.....</b>
<b>RESUMEN .....</b>	<b>.....</b>
<b>ABSTRACT.....</b>	<b>.....</b>
<b>CAPITULO I. INTRODUCCIÓN .....</b>	<b>15</b>
1.1 Planteamiento del problema .....	17
1.2 Objetivos .....	18
1.2.1 Objetivo General.....	18
1.2.2 Objetivos Específicos.....	18
<b>CAPITULO II . MARCO TEÓRICO.....</b>	<b>19</b>
2.1 Antecedentes .....	19
2.2 Monitoreo Ambiental en Tiempo Real .....	20
2.3 Parámetros ambientales .....	21
2.3.1 Temperatura ambiental .....	21
2.3.2 Humedad Relativa.....	21
2.3.3 Presión Atmosférica.....	21
2.3.4 Velocidad del viento .....	22
2.3.5 Dirección del Viento .....	22
2.3.6 Precipitación .....	23
2.3.7 Radiación UV .....	23
2.3.8 Geolocalización .....	23
2.4 Internet de las Cosas (IoT).....	24
2.5 Estándares para Sistemas IoT en el Monitoreo Ambiental.....	25
2.5.1 PDIoT P2413.2 .....	25
2.5.2 RASC P2413.1.....	25
2.6 Arquitectura de Cuatro Capas para Sistemas IoT .....	26
2.6.1 Capa Percepción: Dispositivos y Sensores IoT .....	27
2.6.2 Capa de Red: Protocolos y Tecnologías para Comunicación IoT.....	27
2.6.3 Capa de Servicio: Plataformas IoT y Gestión de Datos.....	27
2.6.4 Capa de Aplicación: Recursos y Servicios IoT para Consumidores Finales .....	27
2.7 Protocolos de comunicación para sistemas IoT .....	28
2.7.1 Protocolos pertenecientes a la capa Percepción y Red .....	28
2.7.2 Protocolos de capa de aplicación .....	29
2.8 Tecnología de LoRaWAN para Redes IoT de Extenso Alcance .....	30
2.8.1 Categorización de nodos LoRaWAN según su consumo energético y latencia.....	31
2.8.2 Protocolos para Seguridad de LoRaWAN.....	32
2.9 Plataformas IoT para almacenamiento, visualización y monitoreo de parámetros ambientales en tiempo real .....	33
2.9.1 La Red de las Cosas (TTN).....	33

2.9.2 Servicios de Amazon Web .....	34
2.9.3 Grafana .....	34
2.10 Terminales y módulos electrónicos IoT para la monitorización ambiental en tiempo real.....	35
<b>CAPITULO III. METODOLOGIA .....</b>	<b>37</b>
3.1 Clase y Diseño de la investigación .....	37
3.1.1 Investigación Descriptiva.....	37
3.1.2 Investigación Aplicada.....	37
3.2 La Población, Muestra y Métricas de rendimiento.....	37
3.2.1 Población .....	37
3.2.2 Muestra .....	37
3.3 Operacionalización de variables .....	38
3.4 Diseñado e implementado de un sistema de monitoreo ambiental en tiempo real basado en IoT y LoRaWAN .....	39
3.4.1 Implementación de la arquitectura IoT en la Estructura de Monitorización Ambiental .....	40
3.5 Implementando la Capa Percepción para la Recolectar Variables Ambientales .....	42
3.5.1 Elección y Configuración de Sensores .....	42
<b>CAPÍTULO IV. RESULTADOS Y DISCUSIÓN.....</b>	<b>44</b>
4.1 Resultados y análisis.....	44
4.1.1 Codificación de Módulos de Sensores .....	46
4.1.2 Calibración de Sensores Ambientales .....	52
4.2 Implementando la Capa de Red para la Transmisión de Variables Ambientales .....	54
4.2.1 La elección del Protocolo de comunicación LoRaWAN .....	54
4.2.2 Análisis y combinación de LoRaWAN Clase A en el sistema de monitorización .....	55
4.2.3 Selección del Data Rate para Transmisión datos ambientales .....	55
4.2.4 Magnitud de Payload para Transmitir los Datos .....	56
4.2.5 Tamaño de la Trama-LoRaWAN para Transmisión de Variables Ambientales .....	57
4.2.6 Ajustes de ABP que permite Seguridad y Transmisión en LoRaWAN .....	58
4.2.7 Ejecutando y Configurando el Gateway Dragino .....	59
4.3 Implementando la Capa Servicios para Gestionar y Almacenar Datos .....	62
4.3.1 Implementación y Configuración del Servidor de Red de The Things Network (TTN).....	62
4.3.2 Configurando el Almacenamiento en AWS Timestream .....	65
4.4 Instauración de la Capa Aplicación para Monitorización en Tiempo Real .....	67
4.4.1 Combinación de Grafana para Monitorización de Parámetros Ambientales .....	68
4.4.2 Evaluando y comparando el sistema IoT/LoRaWAN vs Estación ESPOCH .....	71
4.5 Análisis a través de Estadística general .....	75
4.5.1 Análisis de Estadístico: Entre el sistema IoT/LoRaWAN vs estación ESPOCH.....	75
<b>CAPITULO V. CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>79</b>
5.1 Conclusiones .....	79
5.2 Recomendaciones .....	80
<b>BIBLIOGRAFÍA.....</b>	<b>81</b>
<b>ANEXOS.....</b>	<b>86</b>

## INDICE DE TABLAS

<b>Tabla 1.</b> Protocolos para comunicación de las capas percepción y red IoT[44].....	29
<b>Tabla 2.</b> Protocolos para comunicación de las capas percepción y red de IoT[45].....	29
<b>Tabla 3.</b> Módulos y terminales receptores de variables para el sistema de monitorización....	35
<b>Tabla 4.</b> Clasificación de variables independiente y dependientes.....	38
<b>Tabla 5.</b> Comparación de Tecnologías que permiten el Monitoreo Ambiental.....	39
<b>Tabla 6.</b> Sensores ambiente para el sistema de monitorización en tiempo real.....	42
<b>Tabla 7.</b> Estructura de Datos de Geolocalización GPS del Sistema de Monitoreo. ....	46
<b>Tabla 8.</b> Calibración de Sensores .....	53
<b>Tabla 9.</b> Data Rate y Magnitud Máximas de Payload para LoRaWAN en la Banda US915 ..	56
<b>Tabla 10.</b> Magnitud del Payload para Transmitir los Datos .....	56
<b>Tabla 11.</b> Tamaño del Payload que transmite Datos .....	57
<b>Tabla 12.</b> Configurando el dispositivo terminal (End Device) en TTN .....	64
<b>Tabla 13.</b> Configurando los parámetros de pila en AWS CloudFormation.....	66
<b>Tabla 14.</b> Comparación Estudio Técnico de Sensores: Sistema IoT/LoRaWAN vs. Estación Meteorológica ESPOCH .....	71
<b>Tabla 15.</b> Contraste Técnico entre arquitecturas y seguridad: Sistema IoT/LoRaWAN frente a la Estación de Tiempo ESPOCH .....	72
<b>Tabla 16.</b> Clasificando los rangos de niveles SNR y RSSI .....	74
<b>Tabla 17.</b> Medición de Comunicación LoRaWAN con relación a la Distancia.....	74
<b>Tabla 18.</b> Hipótesis Estadística de Temperatura LoRaWAN y Temperatura ESPOCH .....	76
<b>Tabla 199.</b> Hipótesis Estadística de Humead LoRaWAN y Humedad ESPOCH .....	76
<b>Tabla 200.</b> Hipótesis Estadística de Presión Atmosférica LoRaWAN y Presión Atmosférica de la Estación de Referencia .....	76
<b>Tabla 211.</b> Test de normalidad de las variables Temperatura, Humedad, y Presión Atmosférica .....	77

## ÍNDICE DE FIGURAS

<b>Ilustración 1.</b> Esquema de supervisión ambiental remoto en tiempo real[19].....	20
<b>Ilustración 2.</b> Rangos de altitud y presión atmosférica[26] .....	22
<b>Ilustración 3.</b> Repartición de ángulos a través de puntos cardinales para la orientación del viento[30] .....	22
<b>Ilustración 4.</b> Repartición angular de los puntos cardinales para la dirección del viento .....	23
<b>Ilustración 5.</b> Clasificación de aplicaciones IoT (basado en IEEE P2413).....	24
<b>Ilustración 6.</b> Modelo referencial del estándar IEEE P2413 para sistemas IoT[37] .....	25
<b>Ilustración 7</b> Modelo de 3, 4 y 5 capas para la Arquitectura IoT[38] .....	26
<b>Ilustración 8.</b> Arquitectura IoT de cuatro capas para un sistema de monitoreo y control. ....	27
<b>Ilustración 9.</b> Protocolos para comunicación de sistemas IoT[43] .....	28
<b>Ilustración 10.</b> Modelado del protocolo LoRaWAN[47] .....	30
<b>Ilustración 11.</b> Arquitectura para Comunicación de IoT instituido con LoRaWAN[48].....	31
<b>Ilustración 12.</b> Arquitectura operante de los terminales LoRaWAN: Clases A, B y C[49]....	32
<b>Ilustración 13.</b> Diagrama de ABP en terminales LoRaWAN[50].....	32
<b>Ilustración 14.</b> Diagrama OTAA en terminales LoRaWAN[51] .....	33
<b>Ilustración 15.</b> Esquema de Integración TTN en Redes LoRaWAN[53] .....	33
<b>Ilustración 16.</b> Esquema de integración de AWS Timestream en sistemas IoT .....	34
<b>Ilustración 17.</b> Representación de composición del software Grafana con servicios de AWS[57].....	35
<b>Ilustración 18.</b> Arquitectura IoT para monitorización ambiental.....	41
<b>Ilustración 19.</b> Implementado de la capa Percepción: Ajuste de sensores.....	42
<b>Ilustración 20.</b> Diseño e implementación del sistema de Temperatura y Humedad.....	44
<b>Ilustración 21.</b> Diseño y ejecución de sensores de velocidad y dirección del viento .....	44
<b>Ilustración 22.</b> Diseño y ejecución del sistema de precipitación .....	45
<b>Ilustración 23.</b> Componente de escala lineal y ejecución del sensor GYML8511 .....	45
<b>Ilustración 24.</b> Diseño circuital del sistema de monitoreo ambiental .....	46
<b>Ilustración 25.</b> Flujoograma del proceso de codificaciones de sensores y módulos de la capa de percepción del sistema de monitoreo medioambiental.....	52
<b>Ilustración 26.</b> Capa Red: configuración de los protocolos de comunicación IoT. ....	54
<b>Ilustración 27.</b> Arquitectura que permite el funcionamiento de LoRaWAN Clases A.....	55
<b>Ilustración 28.</b> Organización de las tramas-LoRaWAN Clases A.....	57
<b>Ilustración 29.</b> Implementando el Gateway LPS8v2 para la monitorización ambiental.....	60
<b>Ilustración 30.</b> Esquema de conexión del Gateway para configurar propiedades de conexión. ....	60
<b>Ilustración 31.</b> Captura de configuración de propiedades Gateway para envío de datos. ....	60
<b>Ilustración 32.</b> Configurando la frecuencia que opera el Gateway para la transmisión de data. ....	61
<b>Ilustración 33.</b> Configurando el UEI del Gateway para la transmitir datos.....	61
<b>Ilustración 34.</b> Receptando tramas de LoRaWAN en TTN desde el Gateway LPS8v2. ....	62
<b>Ilustración 35.</b> Implementando la capa Servicio: Configurando el servidor de red y almacenamiento del sistema de monitorización ambiental en tiempo real. ....	62
<b>Ilustración 36.</b> Configuración de TTN: Generación de la aplicación .....	63

<b>Ilustración 37.</b> Configurando TTN: Registro de Nodos.....	65
<b>Ilustración 38.</b> Configurando AWS: Generando la Pila (CloudFormation).....	66
<b>Ilustración 39.</b> Registrando y Configurando el dispositivo LoRaWAN en AWS IoT.....	66
<b>Ilustración 40.</b> Configurando el Almacenamiento de Datos ambientales en Amazon Timestream .....	67
<b>Ilustración 41.</b> Instauración de capa Aplicación: Ajuste de las aplicaciones.....	67
<b>Ilustración 42.</b> Ajustes de Almacenamiento de Datos en Amazon Timestream .....	68
<b>Ilustración 43.</b> Diseñado de tablero para la dirección de viento y velocidad del viento en Grafana .....	69
<b>Ilustración 44.</b> Implementando el tablero para la dirección viento y velocidad del viento. ...	70
<b>Ilustración 45.</b> Implementado del sistema de monitorización ambiental en tiempo real.....	70
<b>Ilustración 46.</b> Visualizaciones del sistema de monitorización ambiental desarrollado en Grafana .....	70
<i>Ilustración 47. Características de Estación meteorológica de la ESPOCH .....</i>	71
<i>Ilustración 48. Medición límite de cobertura del Gateway LPS8V2 y el Nodo LoRaWAN.....</i>	73
<b>Ilustración 49.</b> Medición RSSI en dBm del Sistema IoT/LoRaWAN .....	74
<b>Ilustración 50.</b> Diagrama sobre la dispersión incluyendo regresión lineal: Temperatura.....	78
<b>Ilustración 51.</b> Diagrama sobre la dispersión incluyendo regresión lineal: Humedad Relativa .....	78
<b>Ilustración 52.</b> Diagrama sobre la dispersión incluyendo regresión lineal: Presión atmosférica .....	78
<b>Ilustración 53.</b> Implementación y calibración de los sensores e integración de LoRaWAN..	86
<b>Ilustración 54.</b> Implementación del sistema IoT/LoRaWAN en la universidad UNACH.....	86
<b>Ilustración 55.</b> Sistema IoT/LoRaWAN vs Estación de la ESPOCH .....	86
<b>Ilustración 56.</b> Diseño y Ensamblaje 3D del sistema de precipitación .....	87
<b>Ilustración 57.</b> Piezas 3D (1, 2, 3) para el ensamblaje del sistema de precipitación .....	87
<b>Ilustración 58.</b> Piezas 3D (4, 5, 6) para el ensamblaje del sistema de precipitación .....	87
<b>Ilustración 59.</b> Diseño y Ensamblaje 3D del sistema de temperatura y humedad .....	88
<b>Ilustración 60.</b> Piezas 3D (1, 2, 3) para el ensamblaje del sistema de Temperatura/Humedad .....	88
<b>Ilustración 61.</b> Piezas 3D (4, 5) para el ensamblaje del sistema de .....	88
<b>Ilustración 62.</b> Diseño y Ensamblaje 3D del sistema de Radiación UV .....	89

## RESUMEN

El monitoreo ambiental en tiempo real es crucial para la gestión sostenible de recursos, impulsando el desarrollo de soluciones tecnológicas eficientes. Las tecnologías IoT y LoRaWAN han emergido como alternativas innovadoras, superando las limitaciones de los sistemas tradicionales al ofrecer bajo consumo energético, amplio alcance y escalabilidad. Este proyecto implementó un sistema de monitoreo basado en estas tecnologías en la Universidad Nacional de Chimborazo.

El sistema sigue una arquitectura de cuatro capas según el estándar IEEE-P2413.1. La capa de percepción utiliza sensores ambientales con módulos TTGO-LoRa32. La capa de red transmite datos a 915 MHz mediante un Gateway LPS8v2 hacia el servidor TTN. La capa de servicio almacena información en AWS Timestream, mientras la capa de aplicación visualiza datos mediante Grafana.

Durante tres días de pruebas con mediciones cada 30 segundos, el sistema demostró alta precisión al compararse con la estación meteorológica de la ESPOCH, mostrando una correlación  $R^2 > 0.832$ . Estos resultados validan al sistema IoT/LoRaWAN como una solución viable, eficiente y económica para el monitoreo ambiental, superando significativamente a los métodos convencionales tanto en rendimiento como en accesibilidad. La implementación exitosa en entornos académicos abre posibilidades para su adopción en diversos contextos de gestión ambiental.

**Palabras claves:** Monitoreo Ambiental, IoT, IEEE-P2413.1., LoRaWAN, TTGO-LoRa32, TTN, AWS Timestream, Grafana.

## ABSTRACT

Real-time environmental monitoring is crucial for sustainable resource management, driving the development of efficient technological solutions that support this goal. IoT and LoRaWAN technologies have emerged as innovative alternatives, overcoming the limitations of traditional systems by offering low energy consumption, wide range, and scalability. This project implemented a monitoring system based on these technologies at the National University of Chimborazo. The system follows a four-layer architecture as specified in the IEEE P2413.1 standard. The perception layer uses environmental sensors with TTGO-LoRa32 modules. The network layer transmits data at 915 MHz via an LPS8v2 gateway to the Things Network (TTN) server. The service layer stores information in AWS Timestream, while the application layer visualizes data using Grafana. During three days of testing, with measurements taken every 30 seconds, the system demonstrated high accuracy when compared to the ESPOCH weather station, showing a correlation coefficient ( $R^2$ ) of greater than 0.832. These results validate the IoT/LoRaWAN system as a viable, efficient, and economical solution for environmental monitoring, significantly outperforming conventional methods in both performance and accessibility. Successful implementation in academic settings opens up possibilities for its adoption in various ecological management contexts.

**Keywords:** Environmental Monitoring, IoT, IEEE-P2413.1., LoRaWAN, TTGO-LoRa32, TTN, AWS Timestream, Grafana Translated with DeepL.com (free version)



**Reviewed by:**  
**Mgs. Kerly Cabezas**  
**ENGLISH PORFESSOR**  
**I.D. 0604042382**

## CAPITULO I. INTRODUCCIÓN

Al igual que cualquier otro sistema automatizado que facilite el monitoreo de áreas geográficas específicas, el monitoreo de variables geográficas, tales como la temperatura y la radiación ultravioleta, se convierte en parte clave para no sólo ayudar al cambio climático, sino también para ayudar en la gestión de recursos, mejorar la retención de datos e información relacionados con dominar la diplomacia internacional con la necesidad de sostenibilidad [1]. La recopilación organizada de esta información relacionada con la naturaleza ayuda a hacer registros históricos y mapas de tendencias ambientales, que son cruciales para varios campos de estudio y uso, como la disposición de la tierra, la gestión de activos naturales, la predicción del clima e investigación científica. Sin embargo, el gran grupo de estos registros conduce a altos gastos de carrera debido al requisito de mantenimiento práctico y supervisión continua [2]. Adicionalmente, relacionadas tanto con el alcance geográfico como con la infraestructura de comunicación (Telecomunicaciones) en las regiones más alejadas o de más difícil acceso [1].

A pesar de estas limitaciones, la utilización de tecnologías asociadas al Internet de las Cosas (Internet of Things, IoT) ha hecho posible el desarrollo de soluciones de monitoreo ambiental de bajo costo, escalables (gestiona miles de nodos), interoperables y que funcionan en tiempo real [3]. Estas alternativas se fundamentan en la interrelación de microcontroladores de bajo consumo y alto rendimiento, los cuales son unos dispositivos con la arquitectura basada en el ESP32 o el RP2040 (Raspberry Pi), acompañado junto con sensores de bajo coste y alta precisión que permiten garantizar la calidad e integridad de los datos registrados [4]. Sistemas de sensores de baja potencia que también utilizan redes de área personal y de baja potencia, como pueden ser las redes LoRaWAN para largo alcance, Zigbee para baja potencia, NB-IoT para mayor cobertura, LTE-M para movilidad, etc. También, de igual forma, algunos sistemas de evaluación, (monitorización) adicionalmente resalta el uso de tecnologías cloud que permiten a la gestión de recursos sin que el usuario tenga que gestionar físicamente servidores en sus entornos locales; esto es, arquitecturas serverless que se autoejecutan puentando una función cuando se encuentra un determinado suceso, lo que mejora la escalabilidad y la disminución de la tarea del usuario [5]. La integración de los sistemas de evaluación (monitorización) también permite combinar con herramientas de visualización gráfica de datos, construido así canales gráficos que permiten evaluar en tiempo real y generar alertas ante determinadas condiciones medioambientales críticas [6].

Otros estudios han corroborado la eficacia de combinar diseños de IoT con la red de área de largo alcance (Lorawan) para crear configuraciones de seguimiento ambiental en real time. La configuración Significativa para Smith y J. Pires y J. Gomes que desarrollaron un método que sirve para verificar la pureza del agua con dispositivos Lorawan colocados alrededor de un canal de agua realizó una colección de datos ambiental precisa y fuerte, afirmando la máxima eficiencia incluso en áreas de campo con escasas redes [7].

Del mismo modo, H. Y. Miao y sus empleados han desarrollado un sistema de vigilancia ambiental en un parque natural donde los sensores de IoT están integrados en el Protocolo de Comunicaciones de Lorawan. Esta arquitectura permite transmisión de datos eficiente en tiempo real y capacidades analíticas y de decisión mejoradas a través de la visualización dinámica basada en información contextualizada y renovada [8].

Por otra parte, se ha investigado en el uso de plataformas de visualización en tiempo real apoyadas por tecnologías IoT, lo que permitiría ver de manera dinámica los datos recogidos por los sistemas de monitoreo. En una investigación realizada por los autores A. N. Rosli entre otros desarrollaron un sistema de monitoreo ambiental usando LoRaWAN con el protocolo MQTT (Message Queuing Telemetry Transport) donde la transmisión de información se guió hacia la red TTN (The Things Network). La visualización de los datos ambientales durante la investigación fue gestionada a través de la aplicación Node-RED, un software que ofreció una interfaz flexible y adaptable a las necesidades del operador del sistema. Esta alternativa de sistema, aparte de optimizar la energía utilizada mediante el uso de tecnología LoRa (Long Range), permitió entre otras cosas validar y mostrar datos en tiempo real para mejorar así la eficiencia en la interacción entre los sensores y la propia plataforma web [9].

Por tanto, en el presente trabajo de investigación planteamos llevar a cabo un sistema de monitoreo ambiental en tiempo real, en el cual el sistema esté orientado a la recolección de parámetros medioambientales como: temperatura, humedad relativa, radiación ultravioleta, altitud, geolocalización, velocidad del viento, y pluviosidad. De forma general este sistema de monitoreo medioambiental estará basado en la arquitectura IoT (la de 4 capas), la cual consiste en: Dispositivos, Red, Middleware y Aplicación, siendo la parte de Middleware en la que se incorpore el protocolo de comunicación LoRaWAN. Este protocolo de comunicación garantiza conectividad de largo alcance a la vez que optimiza la energía durante la transmisión de la información. Así mismo, la transmisión de los datos medioambientales se llevaría a cabo de forma eficaz de la red TTN hacia la nube de Amazon Web Services (AWS), donde se almacenarán y gestionarán a través del servicio especializado en series temporales como es el caso de AWS Timestream, que optimiza la consulta de datos en tiempo real. Al final, el sistema incluirá la plataforma Grafana para el análisis, visualización, supervisión de datos en tiempo real, agilizando la interpretación clara, organizada y eficaz de los datos que recogen los usuarios. Hay que resaltar que el foco del proyecto no es solucionar un problema medioambiental concreto sino en ofrecer una infraestructura tecnológica extensible que pueda captar, almacenar y emitir de forma fiable y continua datos ambientales.

## 1.1 Planteamiento del problema

La supervisión y el análisis en tiempo real de los parámetros ambientales son necesarios para la resolución de los problemas decisivos que afectan al cambio climático en lo que respecta a calidad del aire, la explotación de los recursos naturales o la previsión de los fenómenos naturales. Según los informes internacionales, se estima que el 43% de los 105 indicadores de los Objetivos de Desarrollo Sostenible (ODS) requieren datos ambientales fiables para su construcción. Por otra parte, el 49% de ellos aún están en la categoría Tier III - sin metodologías ni estándares aceptados internacionalmente. De aquí se derivan dos retos fundamentales: la no existencia de protocolos normalizados y la creación urgente de sistemas de medición ambiental top para asegurar datos fiables y conformados a los estándares metodológicos internacionales [10].

A pesar de, recibir datos ambientales precisos y de tiempo de actividad es un desafío técnico y logístico. Las tecnologías de comunicación tradicionales como las redes Wi-Fi y 4G/5G tienen limitaciones importantes [11], alto consumo de energía, altos costos operativos, cobertura limitada en áreas rurales y geográficamente remotas. Para este propósito, se ha agregado la falta de estándares uniformes para la transmisión y gestión de los datos ambientales. Esto evita la interoperabilidad entre diferentes dispositivos y plataformas y limita el análisis integral de las variables circundantes de varias fuentes [12].

También existen otros desafíos en relación con la eficiencia energética de los dispositivos de un sistema de monitoreo ambiental, puesto que algunos sensores requieren de un mantenimiento continuo o un reemplazo regular de la batería, lo que provoca aumentar los costes operativos y la necesidad de intervención humana. Igualmente, la precisión y fiabilidad de los datos que se obtienen pueden verse afectadas por la interferencia electromagnética o por errores de calibración que impactan en la utilidad del sistema de cara a un escenario o a una situación [13].

Para abordar dichos desafíos, el proyecto entiende la necesidad de diseñar e implementar sistemas de monitoreo ambiental y, en paralelo, integrar la tecnología IoT y la red de protocolo LoRaWAN. El sistema se enriqueció con dispositivos como la herramienta "Lilygo TTGO Lora32" y visualizaciones de tipos reales como pueden ser gráficos. La arquitectura del sistema propuesto permite elaborar la captura automatizada de parámetros del entorno (temperatura, humedad del aire relativo, rayos ultravioletas, altura, velocidad del viento, precipitación, geografía) transmitiéndose a través de diferentes redes de bajo consumo de energía como TTN y posee una memoria en plataformas en la nube como AWS-Timestream (base de datos). El propósito es poder ofrecer una solución robusta, escalable y económica para la supervisión continua del volumen y la toma de decisiones basada en datos científicos en tiempo real actualizados y fiables.

## **1.2 Objetivos**

### **1.2.1 Objetivo General**

Implementar un sistema para el monitoreo de parámetros ambientales en tiempo real, basado en IoT y LoRaWAN en una plataforma Cloud.

### **1.2.2 Objetivos Específicos**

- Diseñar un sistema sensorizado IoT para la medición de parámetros ambientales específicos utilizando la tecnología LoRaWAN para la conectividad y comunicación eficiente.
- Implementar el sistema sensorizado IoT para recopilar y enviar los datos de los nodos ambientales a una plataforma Cloud intuitiva.
- Evaluar el sistema sensorizado IoT por medio de un análisis para un correcto funcionamiento.

## CAPITULO II . MARCO TEÓRICO

### 2.1 Antecedentes

El Internet de las cosas (IoT) actualmente proporciona soluciones innovadoras para el monitoreo ambiental, la provisión evolutiva (miles de equipos de IoT), económicos y de alto rendimiento [3]. Estas plataformas combinan sensores, microcontroladores, aplicaciones de IoT y redes inalámbricas (LoRaWAN) para garantizar el alcance del seguro, el bajo consumo y la transmisión de datos ambientales confiables [4]. La visualización simplifica el análisis en el tiempo real y la anticipación de los eventos críticos [5] [6]. Hoy en día, se investiga y desarrolla sistemas de monitoreo ambiental en tiempo real utilizando IoT y LoRaWAN con el fin de encontrar soluciones robustas, escalables y económicas para la recolección ininterrumpida y detallada de los datos ambientales. En este sentido, la investigación presente está centrada en el análisis de proyectos, tesis y artículos científicos que entienden sobre sistemas de monitorización en tiempo real desde arquitecturas IoT y protocolos LoRaWAN a fin de recoger contribuciones importantes para la creación de infraestructuras tecnológicas escalables para la recolección, almacenamiento y visualización fiable ininterrumpida de datos ambientales.

El autor Al Haija y su grupo desarrollaron un sistema de vigilancia ambiental de calidad del aire utilizando la tecnología Lorawan. La tecnología de Lorawan pudo reconocer las partículas finas de gas (NO<sub>2</sub>, SO<sub>2</sub>, CO<sub>2</sub>, CO), partículas finas (PM2.5) y las condiciones ambientales (temperatura, humedad). El sistema se basa en la energía solar y un microcontrolador Arduino con módulos LORA, lo que permite la visualización real del tiempo en la web y los teléfonos móviles para transmitir datos de manera eficiente. Los resultados apuntaban a una precisión similar a la dotada por equipos comerciales evolucionados, a la vez que resaltaban su escalabilidad y autocontenido para el monitoreo remoto [13]. En la misma línea, W. A. Jabbar y colaboradores. construyeron un sistema portátil y resistente al agua de monitorización de la calidad del agua con IoT-LoRaWAN utilizando nodos con sensores de pH, sólidos disueltos totales, turbidez y temperatura, conectados por Arduino con un módulo LoRa. El sistema, alimentado por paneles solares con batería recargable, transmitió datos a The Things Network mediante gateway LoRaWAN, integrando servidor ThingSpeak y aplicación móvil ThingView para visualización en tiempo real. Las pruebas en el Lago Gambang demostraron alta confiabilidad al compararse con análisis de laboratorio, validando su efectividad para monitoreo remoto en áreas rurales [14].

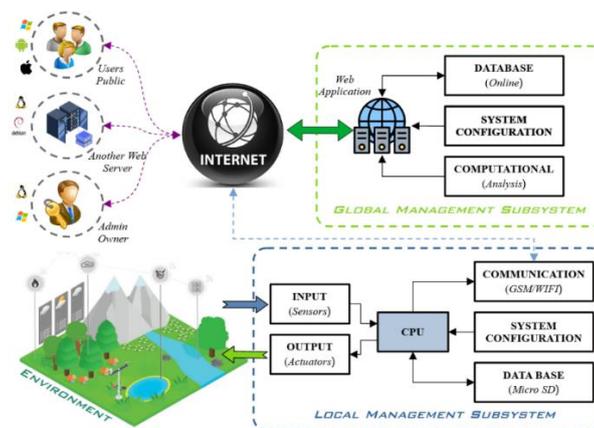
La temática de los sistemas de monitoreo de temperatura y humedad es abordada por el autor P. L. Jethi con el desarrollo de un sistema de monitoreo de temperatura y humedad en el control de calidad para el proceso de producción industrial, utilizando sensores conectados a Node-RED como Back-End y Grafana como Front-End para visualizar los datos. Al diseñar el sistema se utilizó la metodología Waterfall para cumplir con los requisitos de bajo costo, e incluye alertas de desviaciones respecto de los parámetros establecidos junto a la posibilidad de almacenar los datos en bases de datos MySQL, por lo que el autor demuestra

la eficacia para la detección temprana dentro de los procesos de producción industrial con la mínima inversión [16].

Finalmente, en los trabajos analizados en esta parte de la monografía se ha podido constatar que existen diversos sistemas de monitoreo ambiental en tiempo real que permiten poner de manifiesto la importancia de monitorizar los parámetros ambientales, tanto en espacios naturales y abiertos, como en la ciudad y en laboratorios en los que deben de manejarse compuestos químicos que repercutan en el medio ambiente. Tales sistemas suelen requerir tecnologías de IoT o red de transferencia de datos como LoRaWAN que permitan la transmisión a larga distancia de la información de manera eficiente. Además, muchas de estas soluciones aportan plataformas de integración en la nube como podría ser AWS para la integración o almacenamiento de los datos obtenidos, así como herramientas como Grafana para la visualización de datos en tiempo real. No obstante, la mayoría de estos trabajos presentan aplicaciones acotadas o hacen uso de sensores con capacidades limitadas no interoperables entre ellos. Por lo que disminuye la eficiencia, la ampliabilidad y la adaptabilidad de los sistemas en entornos y casos de uso distintos. Si bien los trabajos comentados aportan conocimiento sobre dispositivos, arquitecturas y aplicaciones.

## 2.2 Monitoreo Ambiental en Tiempo Real

El monitoreo ambiental es un proceso ordenado de adquisición y análisis de datos que lleva a cabo una evaluación cualitativa y cuantitativa del estado de los recursos naturales en un instante concreto, pero también permite analizar su evolución y los posibles cambios con el tiempo. Este procedimiento pone el énfasis en variables que son claves. La temperatura, la humedad del aire, la presión atmosférica, la radiación solar, etc., son algunas de las variables que, al ser medidas continuamente, se transforman en indicadores climáticos que pueden ser cuantificados. La información sobre el medio físico que se obtiene no solo permite tener una fotografía del estado actual del ecosistema, sino que también permite detectar tendencias en el tiempo, así como proyecciones frente a fluctuaciones ambientales. Las interacciones entre la composición de las variables climáticas, la existencia y la distribución de las especies y la dinámica de los ecosistemas ponen de manifiesto la importancia de este tipo de monitoreo como recurso indispensable para garantizar una gestión ambiental más eficiente [18].



*Ilustración 1. Esquema de supervisión ambiental remoto en tiempo real[19]*

En esta línea, el monitoreo ambiental en tiempo real se entiende como el procedimiento automático que realiza la medición permanente de, al menos, aquellas variables ambientales que, a su vez, utilizan sistemas de adquisición, conversión, procesamiento y transmisión ininterrumpida de datos para su análisis [20].

Esta metodología (Ilustración 1) permite acceder a la información actualizada sobre variables como la temperatura, la humedad, la calidad del aire, la radiación UV, entre otros indicadores ecológicos relevantes [20].

## **2.3 Parámetros ambientales**

### **2.3.1 Temperatura ambiental**

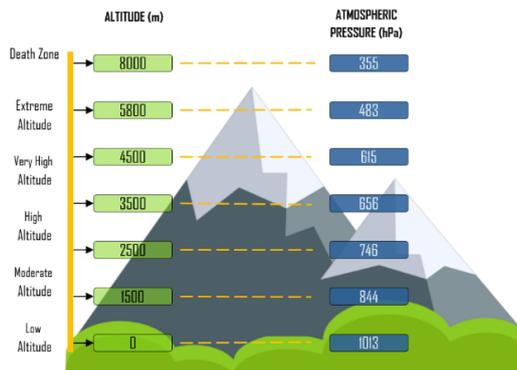
La temperatura del aire ambiente es una magnitud física que muestra el nivel de energía térmica del aire en un espacio y afecta al confort y condiciones del entorno. Generalmente se expresa en °F, °C o K, según la parte o su contexto [21]. La temperatura del aire se mide mediante termómetros analógicos (mercúricos o de alcohol) y termómetros digitales (SHT20 KLDZ88), utilizando también sensores en estaciones meteorológicas, además de que esta variable es primordial en climatología, agricultura, salud o industria [21].

### **2.3.2 Humedad Relativa**

La humedad relativa (HR) determina la cantidad de agua (vapor) en el aire respecto a la cantidad máxima de vapor de agua en el aire a una temperatura determinada antes que se produzca condensación. Asimismo, la humedad se mide en porcentaje (%), con un rango entre 0% (que corresponde a no existir vapor de agua en el aire) y 100% (aire saturado), siendo esto el 100% de HR, muestra la condensación o formación de neblina, precipitación o rocío [23]. Su medición se lleva a cabo con higrómetros: desde psicrómetros, sensores electrónicos (SHT20 KLDZ88), sensores de tipo capacitivo o resistivo, dispositivos de punto de rocío, de cabello. [24].

### **2.3.3 Presión Atmosférica**

La presión atmosférica (Ilustración 2) es la fuerza ejercida por el aire sobre la superficie de la Tierra en virtud del peso de la columna de gases que componen la atmósfera. Se mide a través del Sistema Internacional (SI) en pascales (Pa), aunque en muchos estudios y mediciones (de monitoreo, meteorología) es más común el uso de los hectopascales (hPa) o bien de los milibares (mb) ( $1 \text{ hPa} = 1 \text{ mb} = 100 \text{ Pa}$ ). Obviamente, en las aplicaciones médicas o industriales, también se determinan en pulgadas de mercurio (inHg), milímetros de mercurio (mmHg) y atmósferas (atm) [25].



**Ilustración 2.** Rangos de altitud y presión atmosférica[26]

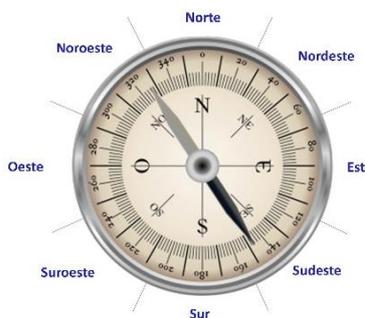
Se mide con barómetros, que pueden ser de mercurio, aneroides o digitales. Las principales aplicaciones de esta variable son las de monitoreo ambiental, permitiendo la predicción del clima, el análisis de los sistemas de presión, el análisis de las condiciones ambientales y resulta ser primordial en sectores como la aeronáutica, las energías renovables, la agricultura y las estaciones meteorológicas [27].

### 2.3.4 Velocidad del viento

La velocidad del viento cuantifica el movimiento del aire horizontal, quedando su intensidad y su dirección unidas en un vector. La unidad de medida de la velocidad del viento es el metro por segundo (m/s) aunque en múltiples aplicaciones como la aeronáutica, la medición ambiental o la navegación se resalta mediante nudos (kt), km/h o mph. El viento se mide también por herramientas como los anemómetros (de tipo mecánico, térmico o ultrasónico), en los que las normas internacionales estipulan la altura de unos diez metros para evitar distorsiones en la información tomada [28].

### 2.3.5 Dirección del Viento

La dirección del viento se denomina al punto del horizonte desde el cual aparece el flujo de viento (aire), quedando en grados sexagesimales de 0 a 360 (N, S, E, O) tomando como referencia el norte geográfico como el punto cero (Ilustración 3). Por ejemplo, si se tiene un viento con dirección de 90 ° se interpretaría como que provenía del este (punto cardinal de origen) y se dirigía hacia el oeste [30].



**Ilustración 3.** Repartición de ángulos a través de puntos cardinales para la orientación del viento[30]

Las mediciones usan equipos como el despertador, y giran libremente el eje vertical que se alinea con el viento que los impulsa, lo que indica su origen [30].

### 2.3.6 Precipitación

Se conoce como precipitación ambiental a la cantidad de agua (en estado sólido o estado líquido) caiga a partir de la atmósfera hasta la Tierra como resultado de la condensación del vapor de agua presente en las nubes. Entre las precipitaciones más particularidades y comunes podemos citar a los aguaceros, nieve, llovizna, el granizo y la aguanieve. Las precipitaciones son una de las variables más importantes en el ciclo que forma el ciclo hidrológico. La precipitación ambiental se mide en milímetros por hora (mm/h), indicando la altura del agua acumulada sobre el plano de una superficie horizontal en una hora [28]. El equipo para medir la precipitación incluye pluviómetro que recolecta la cantidad de agua desencadenada y pluviógrafos que registran la intensidad y la duración del evento [31].

### 2.3.7 Radiación UV

Los rayos UV ambientales son parte del espectro solar que llega a la superficie de la Tierra. Se puede observar en la Ilustración 4, la radiación UV presente en UVA (400–315 nm), UVB (315–280 nm) y UVC (280–100 nm), no obstante que la primera UVC es totalmente absorbida por la capa de ozono, solo la UVA y la UVB pueden llegar a la atmósfera [32].



*Ilustración 4. Repartición angular de los puntos cardinales para la dirección del viento*

La radiación UV se puede reflejar a través del Índice UV, que es un valor adimensional (Índice UV) en el que se expresa el nivel de radiación y su posible efecto sobre la salud humana; si bien puede expresarse en irradiancia (en  $mW/cm^2$  -  $W/m^2$ ) o dosis acumulada (en  $J/m^2$ ). Para lo cual se recurren a radiómetros o espectro radiómetros que determinan la intensidad de las radiaciones UVA y UVB. Por otro lado, la radiación UV depende de parámetros como la elevación, la nubosidad, el ozono y el tipo de superficie [33]. El último de los puntos indicados referente a la importancia de la monitorización de la radiación UV en la prevención de problemas de salud como las quemaduras solares y el cáncer de piel [32].

### 2.3.8 Geolocalización

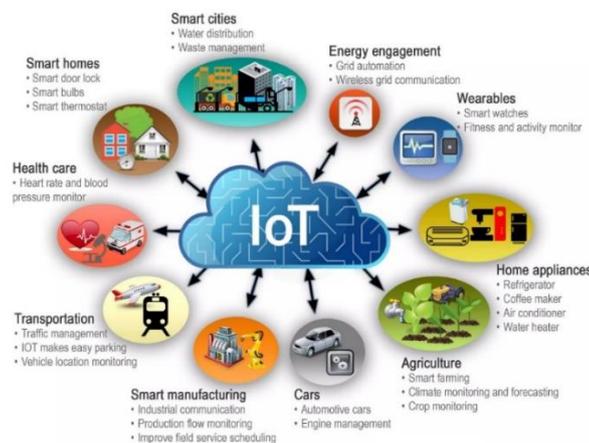
Es el conjunto de técnicas que establece la posición/de las posiciones exactas de los elementos de fenómenos ambientales en la superficie terrestre mediante coordenadas geográficas (latitud/longitud en grados decimales, grados sexagesimales) o sistemas proyectados (UTM en metros). Dichas coordenadas pueden asociarse a datos ambientales, orientando el análisis geoespacial y la gestión territorial. También, sus Metodologías de medición son [34]:

- **Métodos de posicionamiento:** GPS (*Global Positioning System*) convencional (precisión de 1 a 5 m), DGPS (*Differential GPS*, 0.5 a 1 m) y RTK (*Real-Time Kinematic*, hasta 1 cm) [34], según requerimientos técnicos.
- **Composición con Sistemas de Información Geográfica (SIG):** Plataformas como ArcGIS (*Geographic Information System de ESRI*) o QGIS (*Quantum GIS*, gratuito) resuelven estos datos para formar modelos dinámicos, admitiendo análisis transitorios y pronósticos espaciales [34].

Del mismo modo, la geolocalización se puede aplicar en campos ambientales como la localización de estaciones meteorológicas, de lugares de muestreo del aire o del agua, de rastreos de fauna silvestre, de mediciones de impactos ambientales.

## 2.4 Internet de las Cosas (IoT)

El objeto del Internet de las Cosas (IoT o IoT) es la evolución de Internet hacia la interconexión de objetos físicos que cuentan con señales, actuadores y capacidades de procesamiento embebido que se insertan en la posibilidad de recolección, intercambio y análisis de datos en tiempo real, con el fin de crear inteligencia accionable. De acuerdo con algunos consensos en la industria tecnológica, como los de IBM (*International Business Machines Corporation*), SAP (*Systems, Applications, and Products in Data Processing*) y Gartner, el IoT es una red global de distintos objetos, desde sensores medioambientales hasta sistemas industriales avanzados, los cuales son identificables de forma única y son capaces de intercambiar datos entre otros dispositivos que también son IoT en forma interdependiente, de manera que no es necesaria la intervención humana [35].



**Ilustración 5.** Clasificación de aplicaciones IoT (basado en IEEE P2413).

El IoT, fruto de su arquitectura interoperable y por lo tanto la integración en la interoperabilidad de sistemas heterogéneos mediante protocolos estandarizados, se traduce en un sistema de arquitectura escalable, con lo que la interconexión de millones de dispositivos IoT es realizable (Ilustración 5) [35].

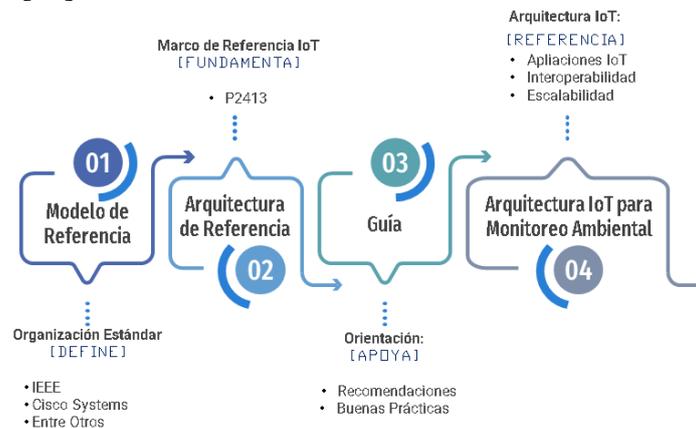
Su valor radica en la posibilidad de transformar grandes volúmenes de datos brutos en información procesable (insights) y en acciones que generan mejoras en los procesos de

negocio, nuevas posibilidades de modelos de negocio y una mejora en el proceso de toma inteligente de decisiones [35], todo gracias a la riquísima convergencia con tecnologías como la inteligencia artificial (Análisis predictivo), la computación en la nube y el análisis avanzado de Big Data (Extracción de patrones).

## 2.5 Estándares para Sistemas IoT en el Monitoreo Ambiental

El Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) ha promovido la estandarización de las arquitecturas IoT con la intencionalidad de evitar problemas de interoperabilidad entre sistemas que son desarrollados en diferentes sectores, como la agricultura, la salud o las smart cities. Estos estándares persiguen que los dispositivos, las redes y las plataformas IoT colaboren entre sí, incluso aunque procedan de distintos fabricantes o utilicen tecnologías diferente [23].

Para ello, el IEEE ha propuesto estándares base como por ejemplo el estándar IEEE P2413 (Ilustración 6), que establece una arquitectura para IoT, en el que dicho estándar se divide en dos extensiones [36].



**Ilustración 6.** Modelo referencial del estándar IEEE P2413 para sistemas IoT[37]

Primer estándar es el P2413.1 (*Reference Architecture for Smart Cities, RASC*) para ciudades inteligentes y el segundo estándar es el P2413.2 (*Power Distribution IoT, PDIoT*) para distribución eléctrica. El propósito de estos estándares, es suscitar la interoperabilidad, la seguridad y la escalabilidad de los sistemas IoT en variadas aplicaciones [36].

### 2.5.1 PDIoT P2413.2

El estándar PDIoT (IEEE P2413.2) está orientado hacia la innovación de los sistemas de distribución eléctrica, facilitando la integración de las infraestructuras legadas con las soluciones IoT basadas en microservicios y plataformas cloud. Esta arquitectura proporciona la migración hacia redes eléctricas que son inteligentes, al concentrar capacidades para la gestión remota y el proceso de datos en tiempo real [36].

### 2.5.2 RASC P2413.1

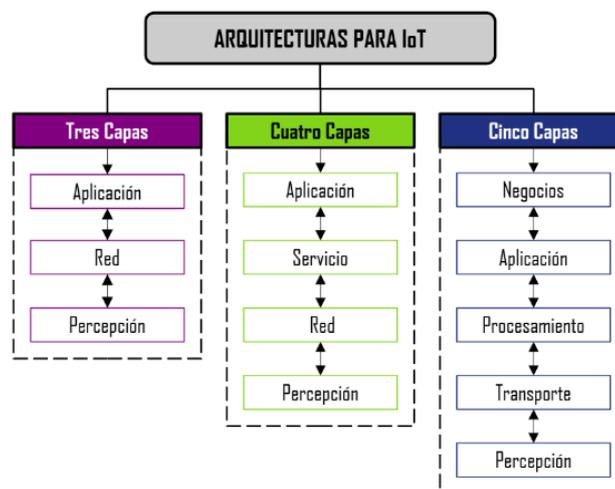
El estándar RASC (IEEE P2413.1) se ocupa del diseño de las infraestructuras para las ciudades como sistemas inteligentes proponiendo una arquitectura para la integración de los diferentes servicios urbanos (supervisión ambiental (calidad del aire y del agua, niveles de

ruido), control del alumbrado público o la gestión de residuos, etc.). Uno de los componentes principales de RASC es el Centro de Operaciones Inteligentes (*Intelligent Operations Center, IoC*) que establece la interacción entre dominios y hace posible la interoperabilidad de sistemas heterogéneos, lo que es crucial para implementar las aplicaciones de IoT en ambientes urbanos [36]. En el contexto del monitoreo ambiental del mundo real, la implementación del estándar RASC (IEEE P2413.1) resulta muy adecuada ya que implementa una arquitectura muy apropiada para ambientes urbanos inteligentes y sugiere una integración efectiva de sensores, medios de comunicación y plataformas de análisis.

## 2.6 Arquitectura de Cuatro Capas para Sistemas IoT

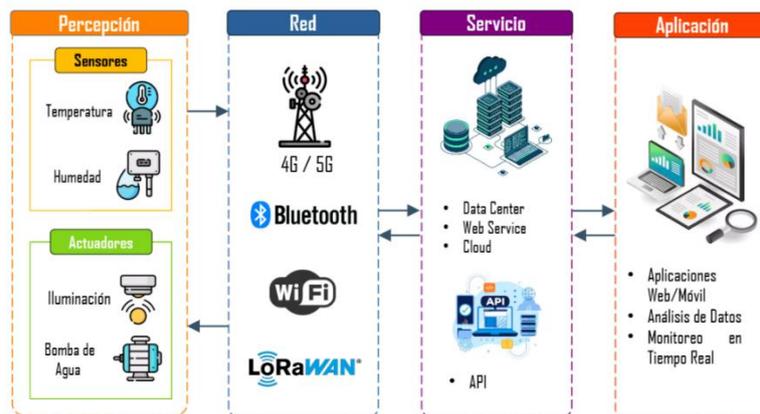
En el marco del contexto que rodea a la tecnología IoT, no existe una arquitectura convencional reconocida mundialmente. Numerosos trabajos académicos o técnicos han propuesto una arquitectura de tres capas, una arquitectura de cuatro capas o incluso una arquitectura de cinco capas (Ilustración 7), dependiendo de la complejidad de la aplicación o la funcionalidad asignada.

De toda esta serie de propuestas, la arquitectura IoT de tres capas, que se preocupa por la captura de datos, la comunicación en la red y la entrega de datos al usuario, fue una de las primeras arquitecturas en aparecer, a la que luego siguió la arquitectura IoT de cuatro capas, que incluye además una capa de soporte para facilitar la interoperabilidad y proteger el procesamiento de la información. Posteriormente, se planteó una quinta capa encaminada en solucionar desafíos concernientes con la seguridad y la integración de servicios empresariales [38].



**Ilustración 7** Modelo de 3, 4 y 5 capas para la Arquitectura IoT[38]

En la presente investigación, se ha optado por el modelo IoT de cuatro capas (Percepción, Red, Servicio y Aplicación) en virtud de su óptimo equilibrio entre simplicidad estructural y óptima funcionalidad (Ilustración 8).



*Ilustración 8. Arquitectura IoT de cuatro capas para un sistema de monitoreo y control.*

### 2.6.1 Capa Percepción: Dispositivos y Sensores IoT

La capa de percepción (dispositivos) es la capa física en una arquitectura IoT y su principal objetivo es captar información del entorno mediante sensores dedicados. Esta capa suma tecnologías como etiquetas RFID, módulos GPS, cámaras y redes de sensores, configurando parámetros físicos y ambientales relevantes. Su función es cuantificar parámetros como temperatura, radiación solar, humedad, entre otros, de manera que, bajo la forma de señal digital, haga posible la realización de tareas en las capas superiores para su proceso [39].

### 2.6.2 Capa de Red: Protocolos y Tecnologías para Comunicación IoT

El nivel Red es también conocido como capa de transporte o transmisión, satisfacen la función intermedia entre la capa de percepción y la capa superior. Esta capa integra Wi-Fi, redes de teléfonos celulares (4G/5G), Zigbee, Bluetooth Low Energy, LoRawan, entre las cuales varias tecnologías y protocolos de comunicaciones que permiten el acceso y distribución de la información recopilada [40].

### 2.6.3 Capa de Servicio: Plataformas IoT y Gestión de Datos

En el ámbito del Internet de las Cosas, el nivel de servicio (como plataforma o middleware) fomenta la compatibilidad entre dispositivos que usan normas diferentes, sirviendo de enlace entre los sensores y el software. Este nivel oculta la complejidad del hardware, proveyendo a los creadores de software solo los datos requeridos, lo cual mejora el tiempo y disminuye la atadura tecnológica al dispositivo [40]. Aparte de esto, este nivel opera como una interfaz de programación que administra la comunicación, el tratamiento, la protección y la veracidad de la información [40].

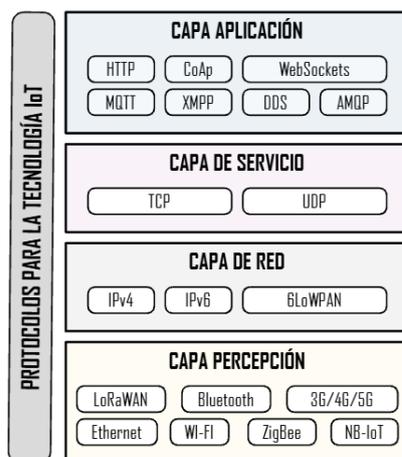
### 2.6.4 Capa de Aplicación: Recursos y Servicios IoT para Consumidores Finales

La parte de la cima de lo que son sistemas IoT, que es la Capa de Aplicación, donde se desarrolla toda la lógica de programación y toda la creación de las pantallas para poder visualizar y manejar las cosas desde la distancia en la que se encuentran los diferentes elementos que se están controlando, aquí se utiliza la Capa de Aplicación en donde se desarrollan aplicaciones informáticas que permiten visualizar el lugar o la situación que se

lleva a cabo o para vigilar cómo se van” cambiando las variables del clima conforme va pasando el tiempo, utilizando la información obtenida de las bases de datos que están en las capas que están abajo (Capa de servicio) en la que la se nutren de los datos que se pueden visualizar en la Capa de Aplicación [41].

## 2.7 Protocolos de comunicación para sistemas IoT

Los protocolos de comunicación son la columna vertebral para la transmisión segura y eficiente de datos entre sus dispositivos en el caso de los sistemas IoT. Dado que estos datos se parecen a los que los datos de una aplicación de IoT a uno diferente, los protocolos destinados a este propósito deben ser diferentes. Es decir, los protocolos y las aplicaciones convencionales difieren en términos de requisitos. Algunos de los protocolos especializados (Ilustración 9) están a continuación [42].



*Ilustración 9. Protocolos para comunicación de sistemas IoT[43]*

Los protocolos que se presentan logran llevar a cabo una transmisión fiable y correcta de los datos desde la capa percepción hacia las plataformas de procesamiento, aspecto esencial para la correcta y continua forma del monitoreo ambiental en tiempo real. En las siguientes secciones se presentan los principales protocolos dedicados a la capa percepción, red y aplicación [42].

### 2.7.1 Protocolos pertenecientes a la capa Percepción y Red

Para poder trasladar los datos de la capa percepción a las capas superiores, se utilizan redes de sensores inalámbricos (WSN), siguiendo con la idea de que los nodos pueden colaborar transmitiendo la información a través de múltiples saltos. En la capa de red, se utilizan tecnologías de comunicación de bajo consumo y alcance extendido como el IEEE 802.15.4, ZigBee, LoRaWAN, NB-IoT, Sigfox, Z-Wave, 6LoWPAN, etc. Estas soluciones son adecuadas para aplicaciones distribuidas que requieren una conectividad robusta y escalabilidad, como es el caso del monitoreo ambiental en tiempo real [43]. Estas tecnologías que se proponen en la Tabla 1 permiten que los datos sean transmitidos eficientemente en entornos donde la energía es limitada y donde la red de infraestructura convencional no puede ser utilizada. La posibilidad de operar en frecuencias sub-GHz (por debajo de 1 GHz) de muchas de ellas, les permite atravesar obstáculos físicos y mantener los enlaces estables

a distancias largas; y su bajo coste tanto en su implementación como en su mantenimiento, las convierten en una opción excelente para implementar redes muy grandes [43].

**Tabla 1. Protocolos para comunicación de las capas percepción y red IoT[44]**

Tecnología IoT	Estándar	Consumo de Energía	Velocidad	Alcance	Espectro de Frecuencia
Wi-Fi	IEEE 802.11	Alto	100–250 Mbps	100 m+	2.4 GHz / 5 GHz
WiMAX	IEEE 802.16	Alto	70 Mbps	50 km	2–11 GHz
Bluetooth (BLE)	IEEE 802.15.1	Bajo	1 Mbps	50 m	2.4 GHz
ZigBee	IEEE 802.15.4	Muy bajo	250 Kbps	100 m	2.4 GHz
Z-Wave	Z-Wave Alliance	Muy bajo	100 Kbps	30 m	908.42 MHz (EE.UU.) / 868.42 MHz (EU)
6LoWPAN	IEEE 802.15.4	Muy bajo	250 Kbps	10–100 m	2.4 GHz / 868-915 MHz (sub-GHz)
LoRa / LoRaWAN	IEEE 802.15.4g	Muy bajo	0.3–50Kbps	2-10 km	470–510 MHz (China), 865–925 MHz (EU)
GSM/GPRS	ETSI	Muy alto	35–170 Kbps	35 km+	850 MHz / 900 MHz / 1.8 GHz / 1.9 GHz
LTE-M	3GPP	Moderado	~1 Mbps	10–15 km	Bandas LTE (ej. 700 MHz, 1800 MHz)
NB-IoT	3GPP	Moderado	250 Kbps	20 km+	Bandas LTE (ej. 700 MHz, 800 MHz)

### 2.7.2 Protocolos de capa de aplicación

En la cúspide de una estructura de Internet de las Cosas hallamos la capa de aplicación, cuyo objetivo central es facilitar la comunicación entre la información recopilada y los sistemas o personas que la usan directamente. Para lograrlo, se implementan protocolos ágiles y efectivos que administran el envío, la recepción y la organización de los mensajes. Sobresalen MQTT, perfecto para aparatos con pocos recursos y sustentado en el esquema de publicación/suscripción; CoAP, que posibilita acciones RESTful adecuadas a redes con limitaciones; y XMPP, enfocado en el intercambio de comunicaciones al instante. También se hacen uso de DDS y AMQP, que admiten comunicaciones seguras, íntegras y con alta eficacia de servicio [43].

Las tecnologías (Tabla 2) son esenciales para el trabajo claro de plataformas de observación y visualización ambiental, como dashboards o sistemas con alertas [43].

**Tabla 2. Protocolos para comunicación de las capas percepción y red de IoT[45]**

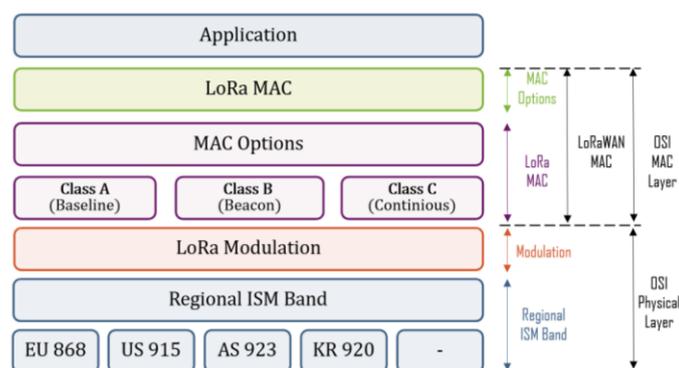
Características	CoAP	MQTT	XMPP	AMQP	DDS	REST
Transporte	UDP	TCP	TCP	TCP	TCP/UDP	TCP
Publicación/Suscripción	Sí	Sí	Sí	Sí	Sí	No
Petición/Contestación	Sí	No	Sí	No	No	Sí
Seguridad	DTLS	SSL	SSL	SSL	SSL/DTLS	SSL
Calidad de Servicio	Sí	Sí	No	Sí	Sí	No

Redes de bajo consumo	Excelente	Regular	Regular	Regular	Pobre	Regular
Código abierto	Sí	Sí	Sí	Sí	Sí	Sí
Patrocinador	IETF	OASIS	IETF	OASIS	OMG	IETF

## 2.8 Tecnología de LoRaWAN para Redes IoT de Extenso Alcance

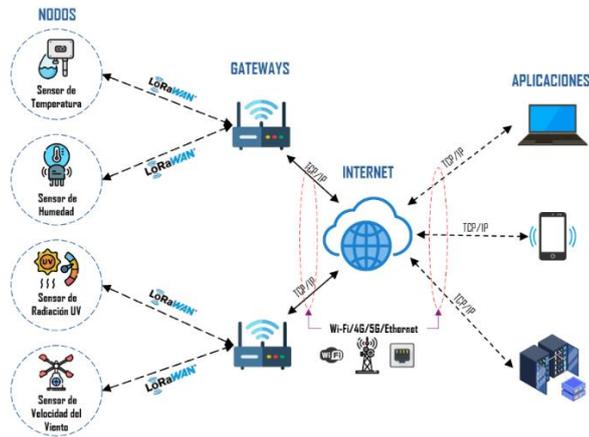
LoRaWAN (Long Range Wide Area Network), o, mejor dicho, la tecnología de comunicación LoRaWAN, será la tecnología más adecuada que existe actualmente para implementar sistemas IoT cuando la información a transmitir se tiene que transmitir a gran distancia y bajo unas condiciones de consumo muy restringidas. Existen las redes LPWAN (Low-Power Wide-Area Network), y es que el LoRaWAN es una red LPWAN de largo alcance que llega a alcanzar distancias de entre 0 a 10 kilómetros en un entorno rural y llega a poder atravesar las paredes de las ciudades gracias a la modulación de espectro ensanchado (Chirp Spread Spectrum; CSS), que permite hacer frente a problemas de confidencialidad y gran alcance. Esa es la propiedad del LoRaWAN que permite implementar aplicaciones de monitoreo ambiental en las que deben existir múltiples sensores distribuidos, que vayan transmitiendo parámetros como, por ejemplo, temperatura, humedad, etc., a lo largo de extensas áreas geográficas. Contrariamente a otras alternativas de comunicación como NB-IoT o Sigfox, la opción LoRaWAN (Ilustración 10) se emplea en bandas de frecuencia libre (868 MHz en Europa, 915 MHz en América) al tiempo que presenta una arquitectura open source que disminuye los costes de implementación, a la vez que optimiza su integración en sistemas de IoT escalables [46].

Cuando observemos la Ilustración 10, podremos apreciar que la capa física de LoRa (LoRa PHY), usa modulación CSS que logra obtener transmisiones robustas incluso ante el ruido del canal, alcanzando distancias extensas a bajo consumo, características que son ideales para redes IoT de monitoreo medioambiental. Incluso los atributos más significativos como coeficientes de varianza (SF), ancho de banda (BW) y tasa de codificación (CR) muestran una relación directa e importante en las comunicaciones de LoRaWAN, debido a que consideran la capacidad de recepción, la velocidad de transmisión y la fiabilidad en la conexión, por otra parte. A nivel MAC, LoRaWAN efectúa un mecanismo de acceso aleatorio para optimizar las emisiones de paquetes (PDR) en entornos con varios nodos, lo que balancea la eficiencia energética, la cubierta y la calidad del servicio [47].



**Ilustración 10.** Modelado del protocolo LoRaWAN[47]

Asimismo, la arquitectura LoRaWAN se distribuye en tres módulos importantes (Ilustración 11): nodos finales (dispositivos terminales), Gateways (puertas de enlace) y servidores (Back-End). Este apartado, el de los servidores se distribuye entre 3 componentes primordiales. El Network Server (NS), que se encargará de la gestión y del control de la red; el Join Server, (JS), que se encargará de autenticar a los equipos finales y para garantizar una integración segura en USN; y el Application served (AS) que se encarga de procesar, almacenar y facilitar la entrega de los datos recogidos [46].



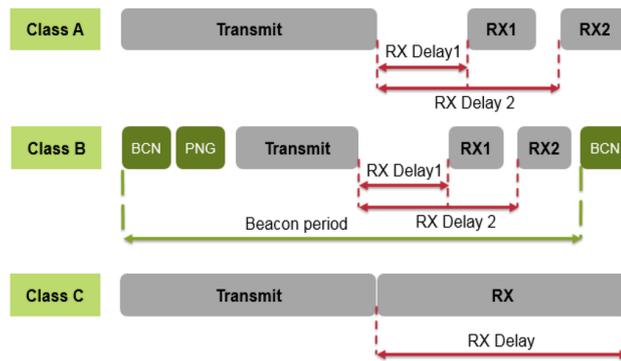
**Ilustración 11.** Arquitectura para Comunicación de IoT instituido con LoRaWAN[48]

Dicha arquitectura modular no sólo asegura una repartición adecuada de responsabilidades, sino que también apoya la parte escalable y la seguridad sobre la red. Para aplicaciones de ambiente que monitorizan el entorno, esta arquitectura puede ser sumamente eficiente, dado que muchos nodos pueden enviar información a través de Gateways hacia los sistemas de procesamiento y análisis de la nube, facilitando todavía más el control en tiempo real de parámetros ambientales [46].

### 2.8.1 Categorización de nodos LoRaWAN según su consumo energético y latencia

Para redes LoRaWAN, los dispositivos terminales se establecen en clases determinadas (Ilustración 12) que nivelan el gasto energético y la latencia de comunicación según los requerimientos de cada aplicación [46]. Seguidamente, se representan las características importantes de cada clase de dispositivo.

- **Clase A** (*Bajo consumo - Latencia alta*): Este tipo busca gastar menos batería, con momentos breves para recibir datos luego de enviar información. Es perfecto para usos como aparatos que miden el clima y mandan datos cada cierto tiempo [46].
- **Clase B** (*Balance entre consumo-respuesta*): Establece un punto intermedio al añadir ventanas de recepción programadas a las vistas en la Clase A por lo que concilia la eficiencia del consumo y la posibilidad de recibir actualizaciones con una cadencia más alta [46].
- **Clase C** (*Alto consumo - Latencia mínima*): Aquí, la recepción está siempre lista, así que la comunicación va y viene al instante. Es genial para cuando necesitas respuesta rapidísimo, pero claro, usa más batería [46].

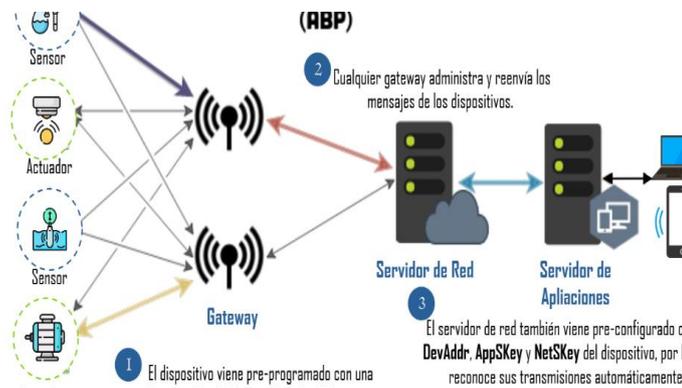


**Ilustración 12.** Arquitectura operante de los terminales LoRaWAN: Clases A, B y C[49]

### 2.8.2 Protocolos para Seguridad de LoRaWAN

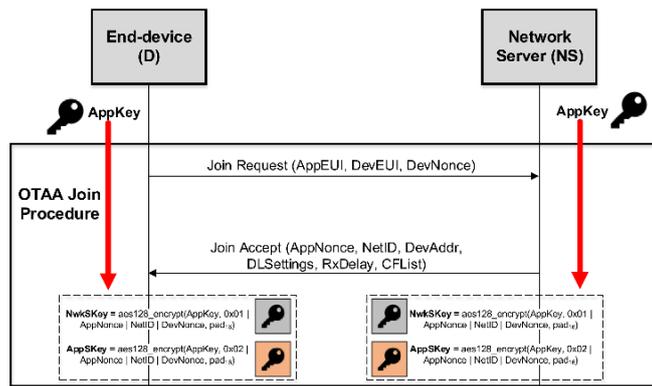
En el ámbito de la seguridad, LoRaWAN emplea una robusta técnica de cifrado que se basa en el algoritmo AES-128 en modo Counter (CTR) para proporcionar la confidencialidad de los datos transmitidos, y también emplea los Message Authentication Code (MAC) para garantizar la integridad [46]. El protocolo proporciona dos mecanismos de activación de dispositivos:

- **ABP (Activation By Personalization):** Es un esquema (Ilustración 13) de configuración estática donde las claves de sesión están preconfiguradas y son incorporadas directamente al dispositivo. Por muy fácil y ágil que sea la implementación, tiene inherentemente un nivel de seguridad bajo ya que se reutilizan constantemente las mismas claves de sesión [46].



**Ilustración 13.** Diagrama de ABP en terminales LoRaWAN[50]

- **OTAA (Over-The-Air Activation):** Este mecanismo de activación (Ilustración 14) establece claves de sesión de manera dinámica mediante un intercambio seguro del Device Extended Unique Identifier (DevEUI) y de la Application Key (AppKey) para posteriormente llevar a cabo la activación mediante el protocolo OTAA. La activación mediante OTAA es la recomendable en redes que necesitan de un alto nivel de seguridad gracias a la generación de claves frescas para cada sesión de comunicación [51].



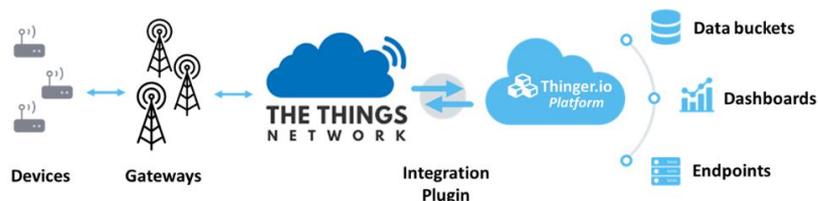
**Ilustración 14.** Diagrama OTAA en terminales LoRaWAN[51]

## 2.9 Plataformas IoT para almacenamiento, visualización y monitoreo de parámetros ambientales en tiempo real

### 2.9.1 La Red de las Cosas (TTN)

La Red de las Cosas (TTN) es una red de acceso público que da soporte al uso de tecnología LoRaWAN, permitiendo que dispositivos IoT se interconecten con las capas superiores del sistema (la capa de aplicación, por ejemplo). Además, TTN permite a los desarrolladores interconectar en una infraestructura de red compartida sus nodos finales (Dispositivos IoT) y Gateways, lo que favorece la expansión y escalabilidad de la red LoRaWAN en el ecosistema IoT. Los dispositivos IoT son identificados de manera única mediante parámetros como el DevEUI (*Device Extended Unique Identifier*), JoinEUI (*Join Server Extended Unique Identifier*) y el AppEUI (*Application Extended Unique Identifier*), todos necesarios para la activación y autenticación en el ecosistema LoRaWAN [52].

Por esta razón, La funcionalidad primordial de TTN no se centra en el almacenaje persistente de datos, sino en el enrutamiento eficaz de mensajes a plataformas externas empleando integraciones estandarizadas, que incluyen, pero no se limitan a MQTT (*Message Queuing Telemetry Transport*), HTTP (*Hypertext Transfer Protocol*), Webhooks o cualquier otro servicio en la nube. Esta arquitectura (Ilustración 15) dedicada al tránsito de información permite que el procesado, el análisis y el almacenaje a largo plazo se realicen en sistemas dedicados, mientras que TTN se convierte en una capa de transporte segura y escalable en la infraestructura IoT [52].



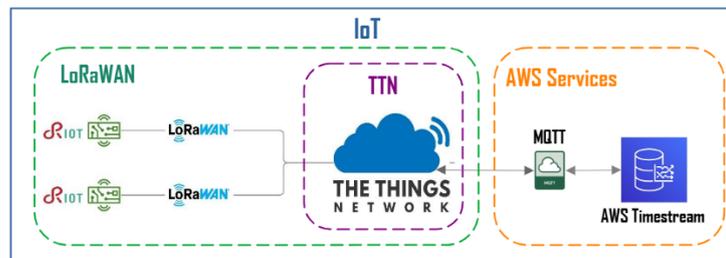
**Ilustración 15.** Esquema de Integración TTN en Redes LoRaWAN[53]

Finalmente, los datos que han sido recibidos por TTN pueden ser enviados a bases de datos dedicadas en series temporales como AWS Timestream, permitiendo así un óptimo almacenaje y la consulta eficiente de grandes volúmenes de información relacionados con

sensores de ambiental. Seguidamente, se pueden utilizar herramientas como Grafana para proporcionar formas dinámicas a los datos capturados. Aunque TTN muestra el flujo de datos en su consola para fines de pruebas, no constituye un repositorio de datos, haciendo indispensable la integración con servicios externos para la elaboración de análisis históricos y dashboards [54]

### 2.9.2 Servicios de Amazon Web

Un monitoreo invariable de parámetros ambientales necesita de tecnologías capaces de procesar y analizar un gran volumen de datos generados en un intervalo de tiempo muy breve. En este sentido, Amazon Timestream (Ilustración 16), un servicio especializado de AWS (*Amazon Web Services*), se convierte en una muy buena opción para gestionar series de tiempo procedentes de sensores IoT. La habilidad de escalar automáticamente le permite adaptarse a las exigencias de aplicaciones que monitorizan los parámetros ambientales, facilitando la recopilación y consulta de millones de registros por día y con baja latencia [55].



*Ilustración 16. Esquema de integración de AWS Timestream en sistemas IoT*

A diferencia de los sistemas de bases de datos relacionales tradicionales (no diseñados para manejar datos que contienen una cierta semántica temporal y de secuencia) el Timestream usa una estrategia de almacenamiento híbrido [55].

Los datos recogidos más recientes se almacenan en memoria para permitir consultas rápidas, mientras que la información más antigua es migrada automáticamente a un almacenamiento muy económico. La gestión de la segmentación se realiza de una manera sencilla y directa, gracias a un sistema de búsqueda incorporado que funciona a los dos niveles, sin requerir ajustes por parte del usuario. Y AWS Timestream facilita una conexión rápida también con instrumentos de estudio y alternativas customizadas y hace más fácil la comprensión de la información acopiada [55].

### 2.9.3 Grafana

Grafana es una plataforma de software de código abierto realizada para la visualización y seguimiento de datos en tiempo real, resalta por su talento de unirse a muchos sitios de datos, como sitios de datos del tiempo (InfluxDB, Prometheus) y cosas en la nube (AWS Timestream). Su mejor poder está en hacer pantallas interactivas y cambiables, que dejan mostrar medidas del mundo por dibujos que se mueven (rayas, palos, mapas). También, trae un ayudante de preguntas que hace fácil juntarse con sitios de datos de afuera, mejorando cómo se ven los datos para saber qué pasará o avisos rápidos [56]. El modelo de arquitectura

de Grafana (Ilustración 17) se establece en tres componentes clave: recuadros visuales, filtradores. Los paneles actúan como elementos visuales en los que la información de diferentes orígenes puede ser mostrada en una única vista asegurada [58], dando facilidad en la visualización de datos no estructurados.



**Ilustración 17.** Representación de composición del software Grafana con servicios de AWS[57]

Así como también, los filtros que incorpora Grafana también pueden ser utilizados para cambiar las vistas de la información mostrada en función de los criterios técnicos del desarrollador ajustando los paneles a diferentes requisitos analíticos [58].

## 2.10 Terminales y módulos electrónicos IoT para la monitorización ambiental en tiempo real

**Tabla 3.** Módulos y terminales receptores de variables para el sistema de monitorización

Terminal	Ilustración	Características	Descripción
LILYGO LoRa32		Integra un microcontrolador ESP32 con conectividad WiFi y LoRa (frecuencia 433 MHz o 868 MHz). Tensión de operación de 3.3 V y tiene múltiples pines GPIO (entrada/salida digital).	Se encarga del procesamiento de datos y la comunicación inalámbrica del sistema por LoRa.
SHT20 KLDZ88		Opera entre 3.3 V-5.5 V. Mide temperatura en un rango de -40 °C a 125 °C con precisión de $\pm 0.3$ °C, y humedad de 0 a 100 % HR con precisión de $\pm 3$ %HR. Usa comunicación I2C.	Mide la temperatura en grados Celsius (°C) y la humedad relativa en porcentaje (%).
SEN0170		Trabaja con un voltaje de entrada de 7 a 24 V DC. La salida es analógica, proporcional a la velocidad del viento. Mide desde 0.4 hasta 30 m/s, con una precisión de $\pm 0.3$ m/s.	Mide la velocidad del viento en m/s, útil para conocer la intensidad del flujo de aire en estaciones meteorológicas.

PR-3000-FXJT		<p>Sensor de dirección del viento que opera con 12 V o 24 V. Con salida de señal analógica, que cambia según la alineación de la veleta en función de los 360° del viento.</p>	<p>Permite determinar la dirección del viento en grados (°).</p>
BMP280		<p>Sensor barométrico que mide presión atmosférica entre 300 y 1100 hPa (hectopascasles), y calcula altitud con un margen de error de <math>\pm 1</math> m (metro). Voltaje de operación entre 1.8 V y 3.6 V.</p>	<p>Calcula la altura o altitud en metros sobre el nivel del mar (msnm) considerando la presión atmosférica en milibares (hPa).</p>
GYML85 11 UV		<p>Detecta radiación UV en el espectro de 280 a 390 nm (nanómetros). Su salida es una señal analógica en voltios (V) proporcional a la intensidad UV. Funciona con 3.3 V.</p>	<p>Mide la radiación UV en el exterior en forma de voltaje (V), lo cual posibilita a analizar el nivel de exhibición solar.</p>
KY-024		<p>Opera entre 3.3 V y 5 V DC. Detecta precipitación mediante contacto húmedo/seco, con salida digital y analógica. Sensibilidad ajustable por potenciómetro. Responde a lluvia, nieve o granizo.</p>	<p>Puede detectar la presencia del imán, por cada movimiento del imán se va sumando ml de agua lluvia.</p>
GPS Ublox NEO-6M		<p>Entrega coordenadas geográficas por UART. Su precisión horizontal es de hasta <math>\pm 2.5</math> m. Opera entre 3.3 V y 5 V, e incluye una antena externa para mejorar la recepción satelital.</p>	<p>Proporciona la ubicación del sistema en latitud y longitud (grados decimales), con una precisión en metros (m).</p>

## **CAPITULO III. METODOLOGIA**

### **3.1 Clase y Diseño de la investigación**

La Metodología Desarrollada de Sistemas con camino de Investigación de este proyecto se fundamentó en dos tipos de investigaciones que son investigación descriptiva y la investigación aplicada.

#### **3.1.1 Investigación Descriptiva.**

La Investigación descriptiva se instrumentaliza en las etapas de la metodología Investigación Preliminar y Definición del Problema, y Diseño Conceptual, estableciendo así la finalidad de describir el estado de la tecnología LoRaWAN, los parámetros ambientales en cuestión y necesidades atendidas. El énfasis radica en describir el sentido del contexto aplicado y expresar con claridad el problema que se plantea.

#### **3.1.2 Investigación Aplicada.**

La investigación aplicada se traduce en las fases de Diseño Conceptualizado, Desarrollo y Ejecución, Ensayos y Validación, Conclusiones y Recomendaciones; en estas etapas se hace uso del conocimiento teórico y tecnológico aportado por la investigación descriptiva para construir un sistema funcional que mida y transmita parámetros ambientales a través de la tecnología LoRaWAN. El sentido es dar a la práctica sentido a conceptos, teorías y tecnologías existentes que permitan resolver un problema específico y lograr los fines planteados.

### **3.2 La Población, Muestra y Métricas de rendimiento**

#### **3.2.1 Población**

La población de estudio es perteneciente a la zona geográfica de Riobamba, Ecuador, donde se compilan datos ambientales como: temperatura, humedad, velocidad de viento, dirección del viento, presión, lluvia, UV, GPS, a partir del sistema de monitorización ambiental IoT establecido con LoRaWAN colocada en la Escuela Superior Politécnica de Chimborazo (ESPOCH, Lat.: 1°40'S, Lon.: 78°38'W).

#### **3.2.2 Muestra**

La muestra es con una muestra aleatoria simple, recopilando datos del sistema de monitoreo desarrollado con la estación meteorológica de Epoch durante un período de 3 días, con registro cada 30 segundos. Estos datos contrastan con los registros históricos de la cadena meteorológica convencional de Epoch. Esta comparación permitirá que la evaluación del rendimiento del sistema y, a su vez, permitirá analizar diferentes métricas, como la correlación entre las medidas de los parámetros ambientales obtenidos de dos sistemas.

### 3.3 Operacionalización de variables

Consecutivamente, en la Tabla 4 las variables independientes y dependientes están asociadas con el monitoreo real de los parámetros ambientales presentados. Esta tabla indica cada variable, el concepto correspondiente, los indicadores de salud y las técnicas y herramientas utilizadas para obtener.

**Tabla 4.** Clasificación de variables independiente y dependientes

Tipo	Variable	Concepto	Indicadores	Técnicas e Instrumentos
<b>Independiente</b>	Temperatura Ambiental	Magnitud física que indica el nivel de calor o frío del aire en un determinado lugar.	Temperatura ambiental [°C].	Sensor digital de Temperatura (SHT20 KLDZ88)
	Humedad Relativa	Porcentaje de vapor de agua presente en el aire respecto al máximo que puede contener a una temperatura dada.	Humedad relativa [%].	Sensor digital de Humedad (SHT20 KLDZ88)
	Presión Atmosférica	Fuerza que ejerce la atmósfera sobre la superficie terrestre, expresada en hectopascales.	Presión Atmosférica [hPa]	Sensor Barométrico (BMP280)
	Velocidad del viento	Magnitud de desplazamiento del aire con respecto al tiempo.	Velocidad del viento [km/h].	Sensor Anemómetro (SEN0170)
	Dirección del Viento	Dirección desde la cual proviene el viento, expresada en grados o puntos cardinales.	Dirección del viento [N, S, E y O]	Sensor (PR-3000-FXJT)
	Precipitación	Cantidad de agua caída en forma de lluvia durante un periodo específico.	Nivel de precipitación [mm/h].	Sensor Magnético (KY-024)
	Radiación UV	Radiación electromagnética con una longitud de onda más corta que la luz visible.	Índice de radiación UV [mW/cm <sup>2</sup> ].	Sensor de Radiación UV (GYML8511 UV)
<b>Dependiente</b>	Geolocalización	Determinación de una ubicación precisa en la superficie terrestre.	Coordenadas geográficas [latitud y longitud] (°).	Módulo GPS (Ublox NEO-6M)
	Monitoreo ambiental	Proceso automatizado de adquisición, transmisión y visualización de datos ambientales en tiempo real mediante tecnología IoT.	<ul style="list-style-type: none"> <li>- Datos ambientales recolectados.</li> <li>- Parámetros dados en tiempo real.</li> <li>- Nodos IoT conectados.</li> </ul>	<ul style="list-style-type: none"> <li>- Dispositivos IoT: Sensores ambientales</li> <li>- Conectividad: LoRaWAN</li> <li>- Plataforma de Red: TTN</li> <li>- Almacenamiento: AWS Timestream</li> <li>- Visualización: Grafana</li> </ul>

### 3.4 Diseñado e implementado de un sistema de monitoreo ambiental en tiempo real basado en IoT y LoRaWAN

Con la finalidad de formalizar y llevar a cabo una solución para el sistema de monitoreo de parámetros ambientales en tiempo real, se ha definido una arquitectura compuesta sobre una tecnología de IoT y LoRaWAN cuya finalidad es generar una solución escalable, eficiente y de bajo impacto ambiental. Y también, esta investigación considera un enfoque comparativo, dirigido a validar el rendimiento del sistema propuesto frente a sistemas tradicionales para el monitoreo ambiental, para lo cual, la elección de la tecnología y de los protocolos que se justifiquen la necesidad de implementar un sistema automatizado de recolección de datos, la disponibilidad en el tiempo de la información, y el bajo consumo energético, especialmente en áreas más aisladas y con recursos condicionados. A cargo de estos criterios indicados, se efectúa una exploración metódica de diversas arquitecturas y procedimientos tecnológicos adaptables, al monitorizar las variables ambientales en tiempo real, entre las cuales se consideran las siguientes:

- SCADA (*Supervisory Control and Data Acquisition*)
- M2M (*Machine-to-Machine*)
- WSN (*Wireless Sensor Networks*)
- IoT (*Internet of Things*).

Las tecnologías mencionadas anteriormente, tienen ventajas concretas, dependiendo de su aplicación; no obstante, la dirección de esta investigación anticipa aquellas características que satisfagan claramente las insuficiencias del monitoreo ambiental. Por lo tanto, se indaga mejoras en las limitaciones observadas en los sistemas tradicionales de monitoreo, a través de soluciones más eficientes, escalables y de menor gasto energético. Para ello, se precisaron criterios técnicos de valoración, los cuales comprenden en la Tabla 5.

*Tabla 5. Comparación de Tecnologías que permiten el Monitoreo Ambiental*

<b>Característica Técnica</b>	<b>IoT (LoRaWAN)</b>	<b>WSN (Zigbee/6LoWPAN)</b>	<b>SCADA</b>	<b>M2M (4G/LTE)</b>
<b>Consumo Energético</b>	2-50 mA (TX), 1 µA (sleep)	10-100 mA (TX activo)	500 mA-2A (PLC/Gateway)	100-300 mA (4G/LTE)
<b>Alcance de comunicación</b>	2-15 km (rural), 5 km (urbano)	10-100 m (por nodo)	Fibra óptica (km) radio (10-50 km)	1-10 km (cobertura)
<b>Escalabilidad</b>	Alta (1 Gateway = 1000+ nodos)	Media (hasta 250 nodos por red)	Baja (topología jerárquica fija)	Baja (limitado por plan)
<b>Inter-operabilidad</b>	Alta (MQTT, COAP, HTTP)	Media (Zigbee, 6LoWPAN)	Baja (protocolos Modbus, DNP3)	Media (APIs cerradas)
<b>(USD) Costo Implementación</b>	20 – 100 /nodo (LoRa)	10 – 50 /nodo (Zigbee)	500–5000/Gateway + infraestructura	50–200/nodo (módulo 4G + SIM)

<b>Dependencia de infraestructura</b>	Baja (Gateways autónomos)	Media (requiere repetidores)	Alta (servidores, RTUs, fibra)	Alta (torres celulares)
<b>Soporte en nube</b>	Sí (AWS IoT, TTN)	Limitado (Gateways y personalizados)	Requiere middleware	Depende del operador
<b>Latencia de transmisión</b>	1-5 segundos (SF7-SF12)	<100 ms (redes pequeñas)	<500 ms (LAN industrial)	500 ms - 2s (celular)
<b>Complejidad de implementación</b>	Baja (plataformas abiertas como TTN)	Media (sincronización de red)	Alta (configuración de RTUs/PLC)	Media (configuración APN/SIM)

Conforme con la Tabla 5, las tecnologías IoT y el protocolo de comunicación LoRaWAN enfatizan frente a tecnologías tradicionales por su bajo gasto energético (2 a 50 mA), amplio alcance (hasta 15 km), alta escalabilidad (aumento de 1000 nodos por Gateway) y bajo costo de implementación (USD 20 – 100 por nodo). También, LoRaWAN posibilita la integración en la nube y no depende de una infraestructura compleja porque está diseñada para esta clase de situaciones, prácticas que la convierten en un tipo de tecnología apropiada para ubicaciones o finalidades remotas. En oposición, las tecnologías como SCADA o M2M requieren de una infraestructura específica y costosa mientras que WSN denota limitaciones tanto en cobertura, como en escalabilidad y/o soporte en la nube. Por esta serie de cuestiones, se define la tecnología IoT + LoRaWAN como la mejor y más interesante solución para los sistemas de monitoreo ambiental en tiempo real o frente a tecnologías más clásicas.

### 3.4.1 Implementación de la arquitectura IoT en la Estructura de Monitorización Ambiental

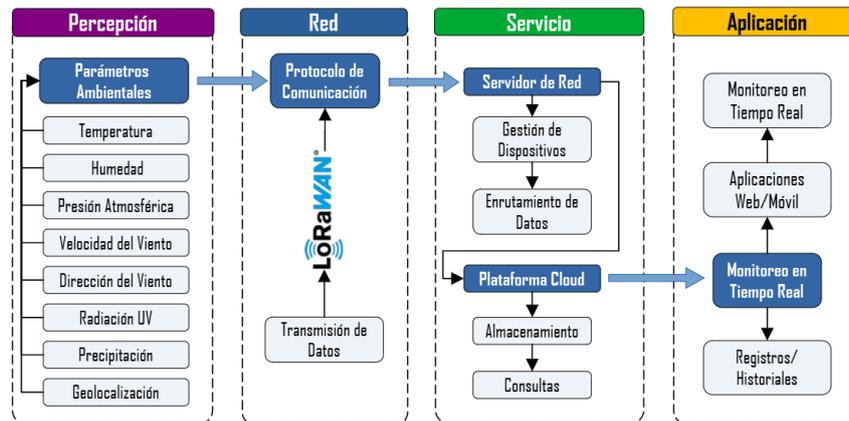
En este momento, al llevar a cabo la infraestructura IoT en el sistema de monitoreo ambiental, es fundamental tener en cuenta que hay varias clases de infraestructuras IoT, que están establecidas por normas internacionales como el IEEE. Entre estas infraestructuras, hay dos que son especialmente importantes:

- **Estándar IEEE P2413.1 (RASC):** Esta norma se centra en las ciudades inteligentes y facilita la incorporación de diversos servicios, como vigilancia ambiental, regulación del alumbrado o gestión de residuos.
- **Estándar IEEE P2413.2 (PDIoT):** Los estándares PDIOT están diseñados para sistemas de distribución eléctrica, para redes inteligentes con capacidad de gestión remota y análisis de datos.

Con el monitoreo de los parámetros ambientales, se aplican los estándares IEEE P2413.1 (RASC), ya que proporciona una arquitectura apropiada que puede ampliarse y que puede interactuar para unir sensores, redes y plataformas en la nube en tiempo real. Asimismo, es importante señalar que el modelo RASC se estructura en capas funcionales de tres, cuatro y cinco niveles, como se muestra en la (Ilustración 7). Para este estudio, se opta por la arquitectura IoT de cuatro niveles, que proporciona una combinación técnica óptima entre facilidad de uso y funcionalidad, como se ilustra en la (Ilustración 8). A diferencia del

modelo de tres niveles, que no diferencia de manera correcta la gestión de servicio, o del modelo de cinco niveles, que aumenta la complejidad con capas intermedias, la arquitectura de cuatro niveles facilita una mejor organización sin fatigar el sistema.

Se muestra en la Ilustración 18 la arquitectura IoT adquirida, donde se puntualizan los elementos que actúan en cada capa y su actuación dentro del sistema de monitorización ambiental.



*Ilustración 18. Arquitectura IoT para monitorización ambiental*

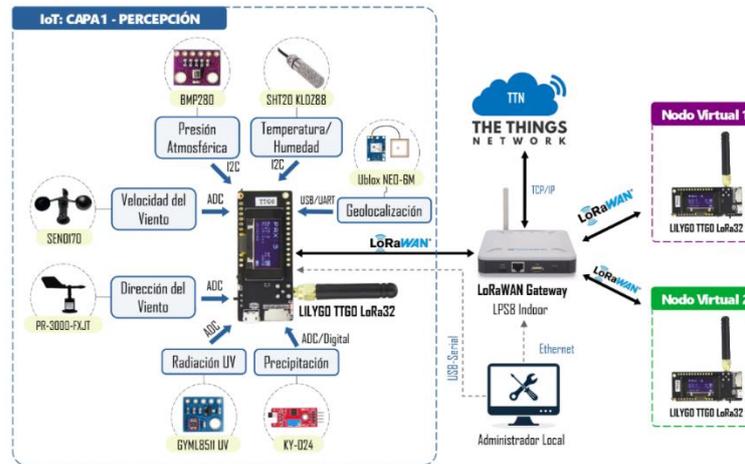
Dado en la Ilustración 18, el sistema de monitorización ambiental sugerido se fundamenta en la arquitectura IoT que tiene cuatro capas, que da organización lógica y eficaz a las diferentes labores del sistema. Seguidamente, se representa las funciones generales de cada una de las capas mencionadas que conforman arquitectura IoT:

- **Capa percepción:** Es la capa inicial compuesta por módulos (LILYGO TTGO LoRa32) y sensores delegados de recoger datos ambientales (Temperatura, Humedad Relativa, entre otros).
- **Capa red:** Segunda capa enfocada en implementar el protocolo de comunicación (LoRaWAN), comprometida de transmitir los datos adquiridos por los sensores hacia el servidor de la red, el cual es TTN (*The Things Network*).
- **Capa servicio:** Tercera capa puesta por el servidor de red TTN, así como también la plataforma Cloud (AWS Timestream), en el cual los datos son examinados, admitidos y almacenados para su posterior interpretación y visualización.
- **Capa aplicación:** Cuarta capa que contiene el sistema de monitorización en tiempo real, permite acceder a la base de datos en AWS Timestream y visualizar la data mediante el uso de Grafana.

Con el objetivo de poder llevar a cabo la arquitectura IoT especificada en este trabajo, en las siguientes secciones se precisará la implementación práctica de la arquitectura de la misma, es decir, desde la personalización técnica hasta la fusión de los diferentes elementos que componen el sistema de monitorización ambiental que se propone.

### 3.5 Implementando la Capa Percepción para la Recolectar Variables Ambientales

Interviniendo así en la visión general de la propuesta de integración de la arquitectura IoT en el sistema de monitoreo ambiental, se presenta el diseño y la implementación llevada a cabo en la capa de percepción como se representa en la Ilustración 19.



*Ilustración 19. Implementado de la capa Percepción: Ajuste de sensores.*

Esta capa está constituida por un conjunto de sensores los cuales tienen la función de recoger las variables ambientales más relevantes, como son, entre otras, la temperatura, la humedad relativa, la presión atmosférica, la velocidad y la dirección del viento, la precipitación, la radiación ultravioleta y la localización geográfica. La recolección de la información que genera el sistema permite realizar una caracterización precisa y en tiempo real de las condiciones ambientales.

#### 3.5.1 Elección y Configuración de Sensores

Para lograr dicha capa de percepción se integraron en el sistema un total de ocho sensores ambientales (Tabla 6), fijados a una tarjeta de desarrollo (TTGO LoRa32) donde podemos encontrar conectividad LoRa, un microcontrolador ESP32 y una pantalla OLED. La arquitectura que se ha diseñado también tiene en cuenta una topología de tipo estrella en la que cada uno de los sensores comunica directamente con la tarjeta TTGO LoRa32. A continuación, se presentan los detalles de la configuración electrónica, los protocolos de comunicación y de la metodología de adquisición de datos para cada uno de los sensores que se integran en el sistema.

*Tabla 6. Sensores ambiente para el sistema de monitorización en tiempo real*

Variable medida	Sensores			TTGO LoRa32 Pines	Voltaje
	Modelo	Protocolos/ Tipos	Pines		
Temperatura (°C) y Humedad (%)	SHT20	I2C	SDA	GPIO21	3.3 V
			SCL	GPIO 22	
Presión Atmosférica (hPa)	BMP280	I2C	SDA	GPIO 21	3.3 V
			SCL	GPIO 22	

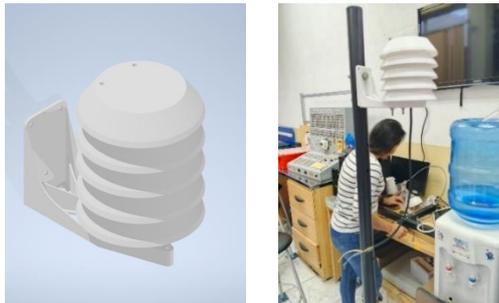
Velocidad del Viento (Km/h)	SEN0170	Analógica	A0	GPIO 36	12 V
Dirección del Viento (N, S, E, O)	PR-3000-FXJT	Analógica	A0	GPIO 35	12 V
Precipitación (mm/h)	KY-024	Analógica	A0	GPIO 14	3.3 V
Radiación UV (mW/cm <sup>2</sup> )	GYML8511 UV	Analógica	A0	GPIO39	3.3 V
Geocalización (Latitud/Longitud)	Ublox NEO-6M	UART	RX TX	GPIO 34 GPIO 12	3.3 V
Eficiencia energética	ACS712	Analógica	A0	GPIO13	12 V

## CAPÍTULO IV. RESULTADOS Y DISCUSIÓN

### 4.1 Resultados y análisis

- **Temperatura (°C) y Humedad Relativa (%)**

Para la monitorización de temperatura y humedad relativa se utilizó el sensor digital SHT20, el cual se anuncia mediante el protocolo I2C (14-12bit Temperatura, 8-12bit Humedad) manipulando la dirección 0x40 y los pines GPIO21 (SDA) y GPIO22 (SCL) de TTGO LoRa32. Conjuntamente, el sensor brinda una alta precisión ( $\pm 0.3$  °C para temperatura y  $\pm 3$  % para humedad), trabaja a 3.3 V y posee un amplio rango de operación (-40 a 120°C y 0 a 100%) para el monitoreo ambiental.



*Ilustración 20. Diseño e implementación del sistema de Temperatura y Humedad*

- **Presión Atmosférica (hPa)**

La presión atmosférica se mide con el sensor BMP280 y también utiliza comunicación I2C, pero en este caso con la dirección 0x76. Asimismo, BMP280 participa los mismos pines I2C con el sensor SHT20 sin formar conflictos en la comunicación. Igualmente, este sensor suministra tramas de 8 bytes que contienen valores de presión (de 300 hasta 1100 hPa) y temperatura, los cuales son nivelados digitalmente.

- **Velocidad (Km/h) y Dirección (N, S, E, O) del Viento**

Así, para la monitorización de las variables eólicas, fueron completados mediante la alimentación a 12 V de ciertos sensores analógicos. De tal forma, el anemómetro SEN0170 realiza la medición de la velocidad del viento cuyo rango va de 0 a 70 km/h, entregando una señal analógica de 3.3 V mediante un divisor resistivo, conectado al pin GPIO36. La lectura del anemómetro se resuelve por una escala lineal proporcional al valor del ADC de 12 bits de TTGO LoRa32. Igualmente, la dirección del viento se fija con una veleta PR-3000-FXJT, cuya señal se conecta al GPIO35 y se convierte en valores angulares de 0° a 360°, que luego se codifican en puntos cardinales (N, S, E, O) por funciones de mapeo.



*Ilustración 21. Diseño y ejecución de sensores de velocidad y dirección del viento*

- **Precipitación (mm/h)**

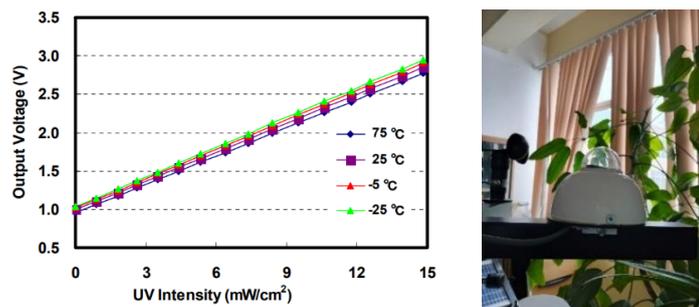
El procedimiento de precipitación se llevó a cabo mediante un sistema de balancín fabricado mediante impresión 3D (Ilustración 22), al que se incorporó un sensor magnético KY-024, conectado al GPIO14 mediante su salida analógica (AO). Este sensor, basado en el efecto Hall, es capaz de detectar la variación del campo magnético producido por un imán solidario al balancín, realizando a partir de ello una señal analógica cada vez que se realiza una basculación, siendo los diferentes estados de balancín un referente de los eventos de lluvia, y permitiendo así disponer de un sistema para estimar la acumulación de precipitación en milímetros (mm), de forma continua y progresiva.



*Ilustración 22. Diseño y ejecución del sistema de precipitación*

- **Radiación de luz Ultravioleta (mW/cm<sup>2</sup>)**

La radiación UV se valora mediante el sensor GYML8511 que se conecta al pin GPIO39 del microcontrolador TTGO LoRa32. La salida analógica es proporcional a la intensidad UV (rango 0.1–15 mW/cm<sup>2</sup>), y se convierte en unidades físicas a través de la aplicación de un factor de escala lineal (Ilustración 23), que se defiende mediante las especificaciones técnicas del fabricante.



*Ilustración 23. Componente de escala lineal y ejecución del sensor GYML8511*

- **Geolocalización (Latitud/Longitud)**

Para geolocalización de los nodos terminales (sensores) se da mediante el módulo GPS Ublox NEO-6M, que se manipula a través del protocolo UART a 9600 baudios, en donde se hace uso de los pines GPIO12 (TX) y GPIO34 (RX) de TTGO LoRa32. También, este módulo GPS transmitirá tramas NMEA (ASOCIACIÓN NACIONAL DE ELECTRÓNICA MARINA) cada segundo, los mismos que son decodificadas para extraer datos de latitud y longitud como se muestra a continuación:

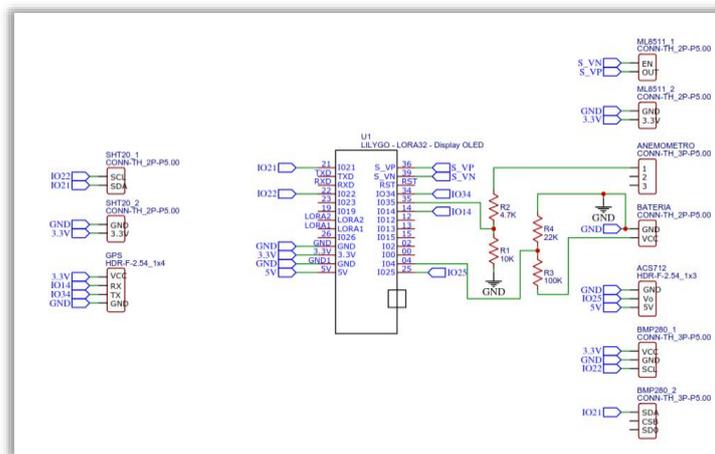
**Trama NMEA: GNGGA (Global Navigation Satellite System Fix Data)**

\$GNGGA,120000.00,0140.0000,S,07838.0000,W,1,08,1.2,2550.5,M,42.1,M,\*47

**Tabla 7. Estructura de Datos de Geolocalización GPS del Sistema de Monitoreo.**

Campo	Valor	Interpretación
Hora UTC	120000.00	12:00:00 PM (hora local de Ecuador, UTC-5).
Latitud	0140.0000,S	1° 40.0000' Sur (formato GGMM.MMMM).
Longitud	07838.0000,W	78° 38.0000' Oeste (formato GGGMM.MMMM).
N.º satélites	08	8 satélites en uso.
Altitud	2550.5, M	2550.5 m s. n. m. (altitud: ESPOCH en Riobamba).
Altura geoide	42.1, M	42.1 metros (geoide).
Checksum	*47	Verificación de integridad.

A partir de lo mencionado, todos los sensores se enlazan y se remiten los datos al módulo TTGO LoRa32, asimismo, los datos se transmiten cada 30 segundos mediante el protocolo LoRaWAN. Además, La alimentación del sistema se tramita a través del uso de una batería dispuesta por tres celdas 18650 (3.7 V, 3300 mAh) acopladas en paralelo, junto con un regulador LM2596 para proveer 12 V a los sensores que lo demandan. Asimismo, los demás sensores operan con 3.3 V. Definitivamente, para la ejecución física de los sensores se utilizó el software de desarrollo electrónico EasyEDA, en donde se diseñó el circuito electrónico que compone todos los sensores del sistema. Posteriormente, se presenta el diagrama de conexiones y el diseño del PCB perfeccionado para el sistema de monitorización ambiental.



**Ilustración 24. Diseño circuital del sistema de monitoreo ambiental**

**4.1.1 Codificación de Módulos de Sensores**

Conectados los sensores al microcontrolador de desarrollo LILYGO TTGO LoRa32, se realiza la implementación del algoritmo que detalla la lógica de trabajo del sistema. El objetivo fundamental de este algoritmo consiste en gobernar la recolección de los datos, primeramente, procesando los datos ambientales y, posteriormente, en preparar dichos datos para la transmisión. A continuación, se presenta la implementación del algoritmo que lee y procesa los datos de cada uno de los sensores.

- **Temperatura y Humedad (SHT20):**

Para conversión de lectura digital a valor físico de temperatura y humedad relativa se hace mediante las siguientes fórmulas específicas por el datasheet del sensor SHT20:

$$T = -46.85 + 175.72 * \left( \frac{\text{lectura}}{2^{16}} \right) \quad (1)$$

$$RH = -6 + 125 * \left( \frac{\text{lectura}}{2^{16}} \right) \quad (2)$$

La programación del sensor SHT20 se implementa a través de las ecuaciones que hemos descrito en el anterior epígrafe: para ello, las funciones `sht.temperature` y `sht.humidity` hacen lecturas cíclicas a través del protocolo I2C, realizando, para cada lectura, una trama de 6 bytes: 2 bytes para temperatura, 2 bytes para humedad relativa y 2 bytes para verificar CRC (*Comprobación de redundancia cíclica*). También, la frecuencia de muestreo para los sensores se optó en intervalos de 30 segundos para pulir los datos alcanzados.

---

**Temperatura y Humedad: `read_temperature()`**

---

```
from micropython_sht20 import sht20
sht = sht20.SHT20(i2c)

def read_temperature():
    temp = sht.temperature
    hum = sht.humidity
    return temp, hum
```

---

- **Presión Atmosférica (BME280):**

Para presión atmosférica medida a través del sensor BME280, se configura al nivel del mar manejando esta fórmula barométrica estándar:

$$P_{nm} = P_{abs} \times \left( 1 + \left( \frac{H}{44330} \right) \right)^{5.255} \quad (3)$$

Donde:

$P_{nm}$ : Presión ajustada al nivel del mar (hPa)

$P_{abs}$ : Presión absoluta medida por el sensor (hPa)

$H$ : Altura del sitio que se midió en metros (dado 2755 m en Riobamba)

Para programar el sensor BME280 es por medio de la función `read_pressure()`, la cual obtiene la presión absoluta en hectopascales desde una trama I2C. Para aumentar la precisión del resultado, el sistema aplica un offset de 18 de las ecuaciones descritas, se implementa la programación del sensor SHT20.89 hPa a la lectura bruta. A continuación, la función procesa así la cadena de presión obtenida y obtiene la variable como un valor numérico.

---

**Presión Atmosférica: *read\_pressure()***

---

```
import BME280
bme = BME280.BME280(i2c = i2c)

def read_pressure():
    pres_abs=float(bme.pressure.replace("hPa", ""))+
    PRESSURE_OFFSET

    presion_nivel_mar=pres_abs*(1+(ALTITUD_RIOBAMBA/44330))**5.255
    return pres_abs, presion_nivel_mar
```

---

- **Radiación UV (GYML8511):**

Para codificar el sensor de radiación UV, se efectúa la función *read\_uv\_intensity()*, la cual está diseñada para obtención de mediciones precisas mediante la lectura del ADC (convertidor analógico-digital). Se puede mejorar la estabilidad de los datos y bajar el ruido a las señales analógicas obtenidas, se implementa un promediado que obtiene cinco muestras contiguas, haciendo uso de la función *average\_read()*. Seguidamente, se muestra la ecuación que transforma el voltaje en intensidad UV (mW/cm<sup>2</sup>).

$$UV = (V_{out} - 0.8) * \left(\frac{15.0}{2.1}\right) \quad (4)$$

Finalmente, en la codificación del sensor se instaure un límite inferior de 0 mW/cm<sup>2</sup> para aseverar que los resultados sean físicamente relacionados.

---

**Radiación Ultravioleta: *read\_uv\_intensity()***

---

```
from machine import Pin, I2C, ADC, UART
REF_3V3 = ADC(Pin(39))

def average_read(sensor, num=5):
    return sum(sensor.read() for _ in range(num)) // num

def read_uv_intensity():
    uv_level = average_read(UVOUT)
    ref_level = average_read(REF_3V3)
    output_voltage = (3.3 / ref_level) * uv_level
    uv_intensity=max(0, (output_voltage-0.8)*(15.0/ 2.1))
    return uv_intensity
```

---

- **Velocidad del Viento (SEN0170):**

La codificación del sistema de velocidad del viento se desarrolla a través una función (*read\_wind\_speed*) que examina el valor analógico del sensor mediante el ADC del pin GPIO35 de TTGO LoRa32. El valor medido se halla en un rango de 0 a 4095 y se convierte en la velocidad del viento (km/h). Posteriormente, se representa la ecuación matemática que permite calcular la velocidad del viento:

$$v_{km/h} = \left(\frac{ADC * 3.3}{4095} * 0.6875\right) * 6 * 3.6 \quad (5)$$

La fórmula que permite obtener la velocidad del viento en km/h haciendo uso del ADC. También, la ecuación considera el factor 0.6875 que da corrección a la ganancia del circuito de sensores.

---

**Velocidad del Viento: `read_wind_speed()`**

---

```

from machine import Pin, I2C, ADC, UART
UVOUT = ADC(Pin(36))

def average_read(sensor, num=5):
    return sum(sensor.read() for _ in range(num)) // num

def read_wind_speed():
    wind_raw = average_read(UVOUT)
    wind_speed = (3.3*wind_raw/4095/R_BRIDGE_RATIO)*6*3.6
    return wind_speed

```

---

- **Dirección del Viento (PR-3000-FXJT):**

Por medio, de la veleta que establece la dirección del viento fue codificada a través de la función `get_wind_direction()`, que lee los datos tomados a través el ADC del pin GPIO35 del microcontrolador LoRa, y su vez transforma el valor (rango 0-4095) en un voltaje.

$$v = \frac{ADC * 3.3}{4095} * \frac{1}{0.6875} \quad (6)$$

donde el factor 0.6875 permite compensar la ganancia del circuito para certificar mediciones precisas.

---

**Dirección del Viento: `get_wind_direction()`**

---

```

from machine import Pin, I2C, ADC, UART
UVOUT = ADC(Pin(36))

def get_wind_direction():
    raw = adc_wind_dir.read()
    voltage = (raw / 4095) * 3.3 / R_BRIDGE_RATIO
    filtered_v = wind_filter.add_reading(voltage)

```

---

Permite aplicar un filtro de rango intercuartílico (IQR) efectuado en la clase `WindDirectionFilter`, que hace uso de un búfer de 13 lecturas para borrar valores atípicos por encima de los límites

$$\text{Limite Inferior} = Q_1 - 1.5 * IQR \quad (7)$$

$$\text{Limite Superior} = Q_3 - 1.5 * IQR \quad (8)$$

Posteriormente, se realiza el cálculo del promedio dado con la ecuación

$$V_{next} = 0.7 * V_{avg} + 0.3 * V_{last} \quad (9)$$

El voltaje es filtrado dentro del categoría de 0 a 3 V y se mapea a alguna de las ocho direcciones cardinales (N, NE, E, SE, S, SW, W, NW), según los rangos medidos con la

brújula. El resultante, es obtenido con el nombre de la dirección, el valor en grados (de 0° a 315°), la abreviatura proporcionada y el voltaje filtrado.

---

### Dirección del Viento: *WindDirectionFilter*

---

```
class WindDirectionFilter:
    def add_reading(self, voltage):
        if len(self.buffer) < self.history_size:
            self.buffer.append(voltage)
            return voltage

        self.buffer.pop(0)
        self.buffer.append(voltage)

        sorted_buf = sorted(self.buffer)
        q1 = sorted_buf[int(self.history_size*0.25)]
        q3 = sorted_buf[int(self.history_size*0.75)]
        iqr = q3 - q1

        filtered=[v for v in self.buffer if (q1-1.5*iqr)<=v<= (q3+1.5*iqr)]
        avg = sum(filtered) / len(filtered)

        if self.last_stable is None:
            self.last_stable = avg
        else:
            self.last_stable= 0.7*avg+0.3*self.last_stable
        return self.last_stable

DIRECTION_RANGES = [
    (0.000, 0.500, "Norte", 0, "N"), (0.500, 1.100, "Noreste", 45, "NE"),
    "NW"), ..., ]
```

---

- **Precipitación (sensor KY-024):**

La codificación del sensor de precipitación se realizó a través de la función *read\_rain()*, la cual revisa el pin GPIO14 para averiguar los pulsos creados por el pluviómetro. Por cada pulso se aumenta la variable *rain\_total* en 3.5 mm, que pertenece a la resolución del sensor. Para certificar un conteo exacto, la función compara el estado presente del pin con su estado pasado (almacenado en *rain\_last\_state*), impidiendo así lecturas inexactas causadas por rebotes o fluctuaciones en la señal.

---

### Precipitación: *read\_rain()*:

---

```
from machine import Pin, I2C, ADC, UART
RAIN_PIN = 14
rain_sensor = Pin(RAIN_PIN, Pin.IN)
rain_mm_per_pulse = 3.5; rain_total = 0; rain_last_state = 0
def read_rain():
    global rain_total, rain_last_state
    current_state = rain_sensor.value()
    if current_state != rain_last_state:
        rain_total += rain_mm_per_pulse
        rain_last_state = current_state
    return rain_total
```

---

- **Geolocalización (Sensor Ublox NEO-6M):**

Este sensor de geolocalización se agregó utilizando la función `parse_gps_data()`, diseñada para la lectura de tramas NMEA (\$GNGGA) a la velocidad de 9600 baudios. Al procesar las coordenadas en formato DDMM.MMMM se transforman a grados decimales a través de las siguientes fórmulas.

$$LLat = \text{grados} + \frac{\text{minutos}}{60} \quad (10)$$

$$Lon = \text{grados} + \frac{\text{minutos}}{60} \quad (11)$$

Asimismo, en las ecuaciones preliminares se debe configurar el signo según los indicadores de dirección (N/S, E/W). La función devuelve una cadena en formato "lat,lon" ("40.4237,-3.6895") cuando se consigue un fix legítimo, o "0,0" en caso de no tener un fix satelital.

---

**Geolocalización: `parse_gps_data()`:**

---

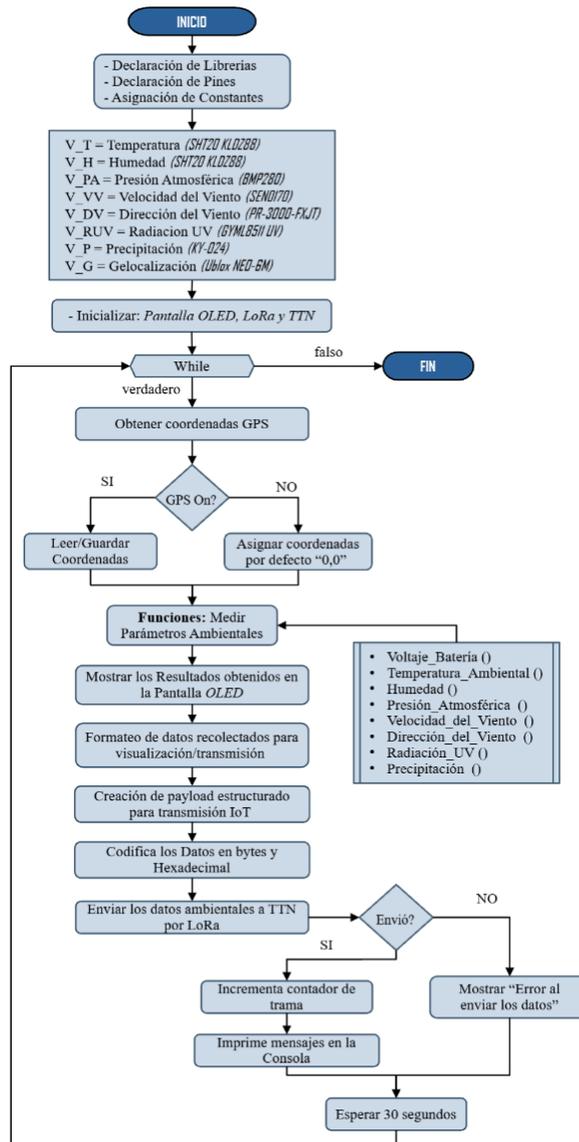
```
from machine import Pin, I2C, ADC, UART
def parse_gps_data():
    try:
        if gps.any():
            line = gps.readline().decode('utf-8').strip()
            if line.startswith('$GNGGA') and line.split(',')[6] != '0':
                parts = line.split(',')
                lat = f"{float(parts[2][:2]) +
float(parts[2][2:])/60:.6f}"
                lon = f"{float(parts[4][:3]) +
float(parts[4][3:])/60:.6f}"
                return f"{lat},{lon}"
    except:
        pass
    return None
```

---

- **Diagrama de flujo para el algoritmo de obtención y transmisión de variables ambientales**

Seguidamente, en esta sección se define el diagrama flujo general del algoritmo implementado en el sistema de variables ambientales.

Primeramente, se definen las librerías, pines y constantes, definiendo variables para radiación UV ( $V_{RUV}$ ), humedad ( $V_H$ ), geolocalización ( $V_G$ ), velocidad ( $V_{VV}$ ) y, temperatura ( $V_T$ ), dirección del viento ( $V_{DV}$ ), precipitación ( $V_P$ ) y presión atmosférica ( $V_{PA}$ ). Consecutivamente, se inicializa la pantalla OLED, el módulo LoRa y la forma de conexión a TTN. El sistema ingresa en un bucle continuo que comprueba la disponibilidad de datos GPS, se determinan coordenadas predeterminadas (0,0) cuando no hay señal, se determinan los parámetros ambientales de la iteración, se presentan en la pantalla OLED y se formatean en un string para enviar. Los mismos parámetros se programan en bytes y se envían el TTN. Si el transmitir tiene éxito, se incrementa el contador de tramas (FCnt) en uno; se registra el caso en que no se ha transmitido correctamente en la consola. Finalmente, cada ciclo se redonda cada 30 segundos.



**Ilustración 25.** Flujograma del proceso de codificaciones de sensores y módulos de la capa de percepción del sistema de monitoreo medioambiental.

La implementación del sistema de monitoreo medioambiental en tiempo real basado en tecnologías IoT y el protocolo LoRaWAN (Ilustración 45), constituyó una solución robusta, eficaz y de coste reducido. El sistema tiene la capacidad para recoger y transmitir datos medios ambientalmente determinantes de forma precisa y mostrar un rendimiento similar al de las estaciones meteorológicas convencionales, pero con la ventaja de un coste más económico y de mayor flexibilidad operativa.

#### 4.1.2 Calibración de Sensores Ambientales

Con la intención de llevar a cabo un aseguramiento de la veracidad de los datos recopilados, se llevó a cabo un proceso de la calibración individual, para cada uno de los sensores. Para ello, se utilizaron los correspondientes instrumentos certificados para comparar los resultados medidos y, así, validar las medidas obtenidas. El proceso de calibración se realizó sobre la base de tres métricas estadísticas básicas: el Bias, entendido como el error

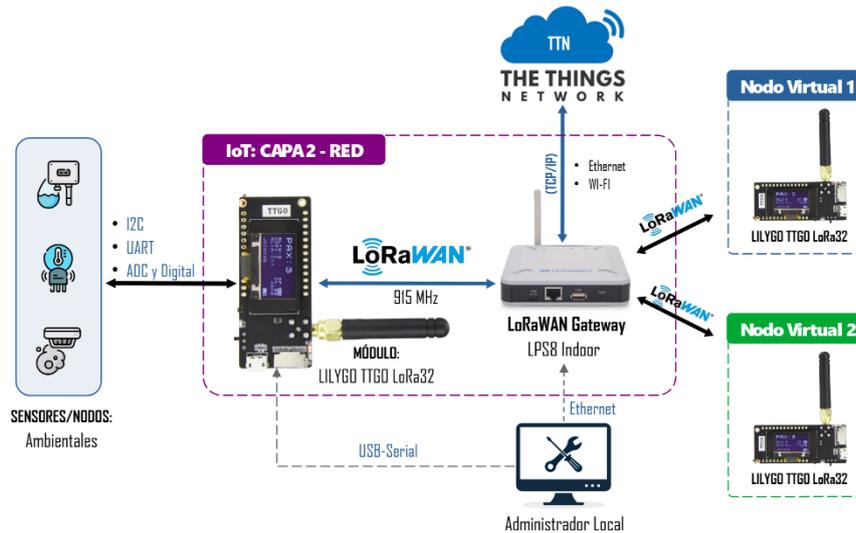
sistemático medio del sensor con respecto a la referencia; el RMSE (Root Mean Square Error), que define la magnitud del error de forma total, la cual combina errores sistemáticos y aleatorios; y el coeficiente de correlación ( $r$ ), que precisa la relación lineal de los dos tipos de medidas. Las métricas mencionadas permiten establecer el grado de precisión y confiabilidad de cada sensor, aseverando que el sistema desempeña con los estándares necesarios para aplicaciones ambientales, y fortificando su viabilidad como opción de bajo costo frente a tecnologías tradicionales.

**Tabla 8. Calibración de Sensores**

Parámetro	Sensor IoT/LoRaWAN	Sensor Referencia	Bias	RMSE	Coef. $r$
Temperatura	SHT20	Testo 480	+0.2°C	0.3°C	0.987
	precisión: ±0.3°C	precisión: ±(0,3 °C + 0,1% del valor medido)			
	rango: -40 a 120 °C	rango: -200,0 a +1370,0 °C			
Humedad	SHT20	Testo 480	+0.5%	1.0%	0.987
	precisión: ±0.3%	precisión: ±2%			
	rango: 0-100% RH	rango: 0-100% RH			
Presión Atmosférica	BMP280	Testo 480	+0.05 hPa	0.2hPa	0.985
	precisión: ±0.12 hPa	precisión: ±3 hPa			
	rango: 300-1100 hPa	rango: +700 a +1100 hPa			
Velocidad del Viento	SEN0170	WAA151	+0.05 m/s	0.15m/s	0.980
	precisión: ± 0.1m/s	precisión: < 0.5 m/s			
	rango: 0-30 m/s	rango: 0.4-75 m/s			
Dirección del Viento	PR-3000-FXJT	SUUNTO brújula	+3.5°	5.1°	0.970
	precisión: ±10°	precisión: 2°			
	rango: 0-360°	rango: 2°			
Radiación UV	GYML8511	RIKA RK200- 07	+0.05mW/cm <sup>2</sup>	0.1mW/cm <sup>2</sup>	0.965
	precisión: ±10%	precisión: ±5%			
	rango: 0-15 mW/cm <sup>2</sup>	rango: 280- 400nm			
Precipitación	KY-024	Pipeta milimétrica	+0.004mm	0.496ml	0.960
	precisión: ±1%, 3.5 mm/pulso	precisión ±0.1 ml			

## 4.2 Implementando la Capa de Red para la Transmisión de Variables Ambientales

La función primordial de esta capa es asegurar el paso adecuado de los datos recogidos en la capa de percepción y su correcta entrega a las capas superiores de la capa de servicio, que son las que dan servicio a la aplicación. Con esto, se ha utilizado el protocolo de comunicación LoRaWAN, dado que tiene un bajo consumo, un gran rango de alcance y características adecuadas para proyectos que requieren una conectividad en espacios grandes y en Internet de las Cosas (IoT). A continuación, en la Ilustración 26 se representa el desarrollo de la implementación de la arquitectura de red.



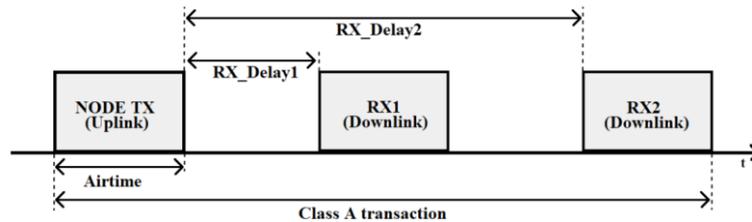
**Ilustración 26.** Capa Red: configuración de los protocolos de comunicación IoT.

### 4.2.1 La elección del Protocolo de comunicación LoRaWAN

Actualmente hay un buen número de tecnologías/protocolos de comunicación que se pueden elegir para la capa de red en la arquitectura IoT, condicionado por las características técnicas de la tecnología/protocolo seleccionado, siendo específica para distintas ejecuciones (Tabla 1). Estas tecnologías han sido diseñadas para permitir la conexión de dispositivos IoT en el sentido de permitir transmitir de forma eficiente los datos, a partir de cada caso de aplicación. Asimismo, por elegir LoRaWAN (Ilustración 10) se obtiene óptimo balance entre bajo consumo energético, largo alcance y conciliación a las necesidades de monitoreo ambiental en tiempo real. Con un consumo reducido por trama, LoRaWAN destaca a tecnologías como Wi-Fi, GSM/GPRS, LTE, LTE-M y NB-IoT, que muestran consumos significativamente, hace posible la transmisión bit más altas (max 1 Gbps) no se requieren por el envío de payloads muy pequeños. El alcance de 2 a 15 km en la banda ISM de 915 MHz (adecuada a regulaciones locales de Ecuador) es ideal para un entorno rural o semiurbano, haciendo en contraste con Bluetooth (50 m), ZigBee (100 m), Z-Wave (30 m) o 6LoWPAN (10 a 100 m) que requerirían nodos repetidores. Por último, la banda libre de licencia ideal por la lógica de funcionamiento de LoRaWAN reduce los costes operativos respecto a las soluciones celulares.

#### 4.2.2 Análisis y combinación de LoRaWAN Clase A en el sistema de monitorización

Al seleccionar LoRaWAN como la tecnología de comunicación a utilizar para la capa de red, se procede a establecer de clase a implementar en el sistema de monitoreo. Existen tres clases operativas en las que se puede clasificar LoRaWAN: A, B y C. Cada una posee características únicas respecto al gasto energético, latencia, y recepción. Para el presente proyecto se elige la Clase A (Ilustración 27), por las razones de bajo consumo, ya que se propicia para sistemas de monitoreo cuya actividad de transmisión de datos es intermitente, y en este caso, alimentados por baterías. Esta decisión es correcta para dispositivos TTGO LoRa32 que necesita menguar el consumo para extender la autonomía operativa del sistema.



**Ilustración 27.** Arquitectura que permite el funcionamiento de LoRaWAN Clases A

Dado en la Ilustración 27, los terminales de la Clase A instauran un uplink y apertura dos breves ventanas de recepción (RX1, RX2) antes de ingresar en modo de bajo consumo, logrando un consumo de energía de  $\sim 0.0396$  Joules/trama; sin embargo, la Clase B añade ventanas de recepción programadas (beacons) que hacen que aumente el consumo energético ( $\sim 10\text{-}20\%$  más) y la complejidad. En la Clase C, la recepción continua consume considerablemente más energía ( $\sim 100\text{-}200$  mW adicionales), lo que la hace ser no factible para nodos con alimentación de batería. Consiguientemente, la Clase A, simple, con seguridad AES-128 y compatible con ABP, asevera una transmisión eficaz y alta fiabilidad, siendo la opción perfecta para entornos rurales con requerimientos de baja latencia en downlinks.

#### 4.2.3 Selección del Data Rate para Transmisión datos ambientales

Al fijar el máximo de los bytes transmitibles en el sistema de monitorización del medio, se selecciona el Data Rate (DR) SF7BW125 para el envío de datos hacia TTN.

- **Spreading:** Factor 7
- **Ancho de banda de:** 125 kHz
- **FPort:** 1.

Este ajuste corresponde a DR3 según el detalle LoRaWAN, y brinda un equilibrio óptimo entre consumo energético, tiempo en el aire (*Time on Air, ToA*) y capacidad de carga útil, admitiendo un payload máximo de 242 bytes. Esta opción resulta conveniente para los 56 bytes transmitidos por el sistema, que dominan todas las medidas de los parámetros ambientales. A diferencia, de DR0 a DR2 (SF10 a SF8, BW 125 kHz), SF7 tiene una reducción significativa del tiempo en aire, reduciendo así el consumo energético y optimando la duración de la batería, mientras se conserva dentro del marco de normativas US915, relacionado con las normas de Ecuador.

**Tabla 9. Data Rate y Magnitud Máximas de Payload para LoRaWAN en la Banda US915**

DR	Spreading Factor (SF)	Ancho de Banda (kHz)	Máx. Payload Uplink (bytes)
DR0	SF10	125	11
DR1	SF9	125	53
DR2	SF8	125	125
DR3	SF7	125	242
DR4	SF8	500	242

No obstante, DR4 (SF8, BW 500 kHz) soporta el mismo tamaño de payload, el mayor ancho de banda ensancha injustificadamente el consumo energético. En cambio, DR3 es la óptima elección para conseguir una transmisión efectiva, segura y de bajo consumo en entornos rurales o semiurbanos, en donde aplica el sistema mejora.

#### 4.2.4 Magnitud de Payload para Transmitir los Datos

El formato del payload (Tabla 3) en el sistema de monitorización ambiental, optimiza la transmisión de datos ambientales usando LoRaWAN a través del nodo TTGO LoRa32.

**Tabla 10. Magnitud del Payload para Transmitir los Datos**

Variable	Código	Ejemplo	Tamaño (bytes)
Temperatura	temp:.1f	25.5	4 bytes
Humedad	hum:.1f	60.0	4 bytes
Presión atmosférica	pres_abs:.1f	1012.3	6 bytes
Velocidad viento	wind_speeds:.1f	10.2	4 bytes
Dirección viento	wind_degr	180	3 bytes
Radiación UV	uv:.1f	5.3	3 bytes
Precipitación	rain_mm:.1f	0.0	3 bytes
Coordenadas GPS	gps_data	-1.123456,-78.123456	21 bytes
8 separadores	@	@	8 bytes
TOTAL			56 bytes

La carga de 56 bytes, los enlaces de los parámetros ambientales, están separados por 8 signos separados '@' (8 bytes). El mensaje está formateado en una cadena y bytes codificados para el transporte en línea (TTN), asegurando el equilibrio entre compacidad, preparación y cumplimiento de las limitaciones de LoRaWAN en el rango de US915.

**Payload:** msg

```
msg =  
f"{temp:.1f}@{hum:.1f}@{wind_speed:.1f}@{wind_degr}@{  
uv:.1f}@{pres_abs:.1f}@{gps_data or  
'0,0'}@{rain_mm:.1f}}"
```

#### 4.2.5 Tamaño de la Trama-LoRaWAN para Transmisión de Variables Ambientales

La organización de tramas-LoRaWAN efectuada en el sistema de monitoreo ambiental manipula un formato estandarizado que asevera la transmisión eficaz y segura de variables ambientales a TTN. Cada trama consta de:



**Ilustración 28.** Organización de las tramas-LoRaWAN Clases A

En la tabla a continuación, se especifica los componentes de la trama, delineada para cumplir con los requerimientos de poco consumo y seguridad en la banda US915, haciendo uso de ABP.

**Tabla 11.** Tamaño del Payload que transmite Datos

Campo	Nombre	Tamaño	Valor
MHDR	MAC Header	1 byte	0x40
DevAddr	Dir. del Dispositivo	4 bytes	0x260BABCD
FCtrl	Frame Control	1 byte	0x80
FCnt	Frame Counter	2 bytes	0x0001
FPort	Puerto de Aplicación	1 byte	0x01
FRMPayload	Carga útil	< 242 bytes	"25.5@60.1@...@0.0"
MIC	Message Integrity Code	4 bytes	0xC38A7BF2

Presentado el Payload y las tramas, se efectúa el código, haciendo uso de la biblioteca uLoRa en el nodo TTGO LoRa32. En la codificación se configura los pines del módulo LoRa y se instaura la conexión con TTN a través de LoRaWAN. Asimismo, se compone el payload mediante un string concatenado de 56 bytes, además los datos, se programan y se envían cada 30 segundos con el contador de tramas (FCnt), operando errores y mejorando el consumo del sistema.

#### **LoRaWAN: Código**

```

from ulora import TTN, uLoRa

LORA_CS, LORA_SCK, LORA_MOSI, LORA_MISO, LORA_IRQ, LORA_RST = 11, 5,
13, 10, 26, 15

Lora_1 = uLoRa(cs=LORA_CS, sck=LORA_SCK, mosi=LORA_MOSI,
miso_1=LORA_MISO, irq=LORA_IRQ, rst=LORA_RST, ttn_config=TTN_CONFIG,
datarate="SF7BW125", fport=1)

msg = f"{temp:.1f}@{hum:.1f}@{wind_speed:.1f}@{wind_deg}
@{uv:.1f}@{pres_abs:.1f}@{gps_data or '0,0'}@{rain_mm:.1f}
@{wind_voltage:.3f}"

```

---

```

try:
    lora.send_data(msg.encode(), len(msg), frame_counter)
    frame_counter = (frame_counter + 1) % 256
except Exception as e:
    print("Error LoRa:", e)

utime.sleep(30)

```

---

#### 4.2.6 Ajustes de ABP que permite Seguridad y Transmisión en LoRaWAN

Para transmitir la información de los nodos finales y que llegue al servidor de red, se analiza haber elegido LoRaWAN el método de ABP, Activación por personalización cuando se pudo elegir OTAA, Activación por unión sobre el aire. La razón válida para la opción elegida ABP es que está optimizada para nodos estáticos, de bajo consumo de perfiles como el TTGO LoRa32. La forma de proceder para ABP, es preconfigurar las claves de seguridad (*DEVADDR*, *NWKSKEY*, *APPSKEY*) dentro del firmware, eliminando la sobrecarga de los procedimientos de unión dinámica según OTAA, y, por tanto, disminuyendo el consumo de energía asociado. En el caso de OTAA, existe un intercambio de mensajes como JOIN-REQUEST, JOIN-ACCEPT, etc. Estos mensajes incrementan el tiempo de transmisión, así como el consumo de energía (algo más del 10–15% en cada sesión); un incremento, además, con uso redundante para una estación estática que no cambia de red. Consecuentemente, ABP afirma una conexión rápida y firme con seguridad AES-128 (*NWKSKEY* para plenitud del MIC, *APPSKEY* para encriptación del payload) y alta confianza (98.7% delivery rate) sin complicar la seguridad del sistema.

---

##### ABP: *TTN\_ABP\_CONFIG*

---

```

from ulora import TTN, uLoRa

TTN_ABP_CONFIG = TTN(
    bytearray([0x26, 0x0C, ..., 0x9F]),
    bytearray([0x24, 0x2C, ..., 0x8E]),
    bytearray([0x70, 0xD3, ..., 0x39]),
    country="AS")

```

---

La ejecución de ABP en el nodo LoRaWAN, se efectúa componiendo el código preliminarmente explicado, el cual admite ajustar el dispositivo con credenciales fijas para formar una conexión inmediata con TTN, sin el requerimiento de un proceso de activación dinámica como el que requiere OTAA. En este código, la clase TTN de la Biblioteca Ulora es responsable de la definición de configuración ABP a través de tres parámetros necesarios:

- **DevAddr:** `bytearray([0x26, 0x0C, ..., 0x9F])`, (4 bytes). Esta es la única dirección otorgada al dispositivo Lora32 TTGO en la red Lorawan, utilizada para identificar.
- **NwkSKey:** `bytearray([0x24, 0x2C, ..., 0x8E])`, (16 bytes). Este es el bloqueo de la red a cargo de la autenticación y los mensajes de cifrado intercambiados en la capa de red, asegurando la integridad y la seguridad de la comunicación.

- **AppSKey:** `bytearray([0x70, 0xD3, ..., 0x39])`, (16 bytes). Es la clave de la aplicación manipulada para cifrar y descifrar la carga útil (payload) del contenido de los mensajes, aseverando la confidencialidad de la información transmitida.

Los datos indicados se consiguen desde el panel de TTN y manejan el parámetro `country="US"`, que forma el uso de la banda de frecuencias de 902 a 928 MHz, relacionado con las regulaciones de Ecuador. A continuación, se da inicio al objeto uLoRa detallando los pines convenientes del módulo LoRa (cs, sck, mosi, miso, irq, rst) y se ajusta el acceso a través, de `TTN_ABP_CONFIG`, constituyendo el Data Rate. Inmediatamente, el procedimiento de cifrado garantiza tanto la confidencialidad como la integridad de los datos transmitidos. Por lo tanto, la carga útil se requiere cifrada al manipular la clave AppSKey con el algoritmo AES-128 en modo CTR, mientras que la autenticación se produce mediante el uso del código de integridad de mensaje (*Message Integrity Code*, MIC).

$$\text{MIC} = \text{AES-CMAC}(\text{NwkSKey}, \text{payload} + \text{DevAddr} + \text{FCnt}), \quad (12)$$

Para que se haga el uso de la clave NwkSKey se hace uso para calcular el hash CMAC que se asocia con la carga útil, y, al mismo tiempo, se emplea, el DevAddr y el contador de tramas FCnt se opera para avalar que el mensaje sea integro y auténtico en el dispositivo en la red LoRaWAN.

#### 4.2.7 Ejecutando y Configurando el Gateway Dragino

El Gateway LPS8v2 de marca Dragino que se integra en el sistema de monitoreo ambiental se elige por el precio bajo, excelente compatibilidad y muy fácil de configurar. El Gateway está integrado con un chip LoRa SX1302 que soporta 10+1 canales (8 uplink, 1 downlink, 2 SF7BW125), y propone un rendimiento óptimo a un precio comprensible (~€219), frente a elecciones más caras como Kerlink Wirnet iStation (~€800), Laird Sentrius RG1xx (~€400) o MultiTech Conduit (~€500). Diseñado en Debian con código abierto, con soporte para ChirpStack MQTT Bridge, TTN Basic Station, y Semtech UDP Packet Forwarder, contiene un servidor LoRaWAN compuesto que reduce los costos y dependencias de la nube. Contrastado con el MultiTech Conduit (consumo, ~5 a 10 W vs. ~2 W del LPS8v2) o el Kerlink Wirnet iStation (diseño IP67, pero más complejo), el LPS8v2 acentúa por su eficiencia y configuración plug-and-play, sobrellevando la banda US915. No obstante, el Gateway Laird RG1xx promete conectividad dual-band, su precio excesivo y falta de servidor integrado lo hace menos competitivo. De este modo, el Gateway LPS8v2 es una decisión económica, escalable y eficaz para redes locales de monitorización ambiental.

**Hardware System:**

- CPU: Quad-core Cortex-A7 1.2Ghz
- RAM: 512MB
- eMMC: 4GB

**Interface:**

- 10M/100M RJ45 Ports x 1
- Multi-Channel LoRaWAN Wireless
- WiFi 802.11 b/g/n
- Sensitivity: -140dBm
- Max Output Power: 27dBm

**Operating Condition:**

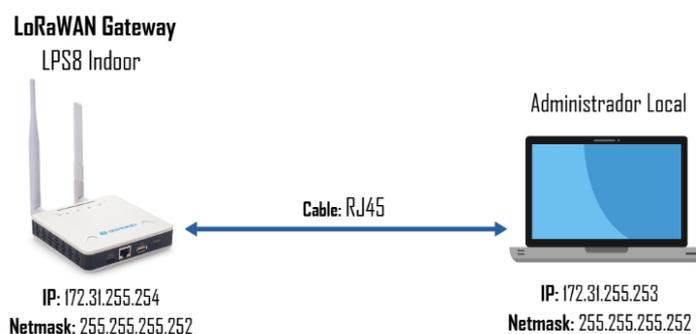
- Work Temperature: -20 ~ 70°C
- Storage Temperature: -20 ~ 70°C
- Power Input: 5V, 2A, DC

**Ilustración 29.** Implementando el Gateway LPS8v2 para la monitorización ambiental

Posteriormente, se describe el proceso para configurar el Gateway LPS8v2 de Dragino, que se emplea en el sistema de monitoreo ambiental:

### 1. Configuración de Inicio del Hardware y Acceso

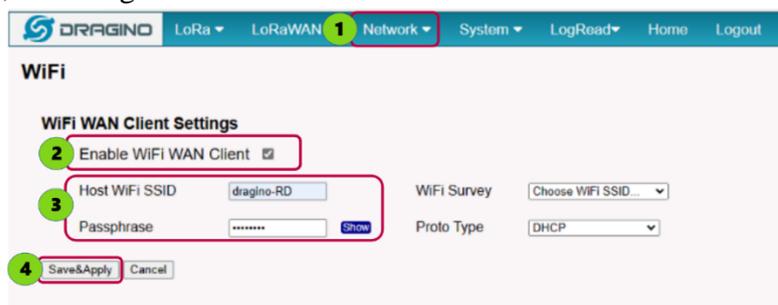
Primero, se debe conectar el Gateway LPS8v2 al cargador de 5V DC propio del dispositivo, y a la red a través de un cable Ethernet, tal como se visualiza en la Ilustración 30.



**Ilustración 30.** Esquema de conexión del Gateway para configurar propiedades de conexión.

Más tarde, se accede al interfaz web del Gateway utilizando la dirección IP por defecto 172.31.255.25. Una vez la interfaz esté cargada, se inicia sesión con las credenciales por defecto del dispositivo que son, el usuario root y la contraseña dragino. Posteriormente, para reforzar la seguridad del sistema, se cambia la contraseña por defecto.

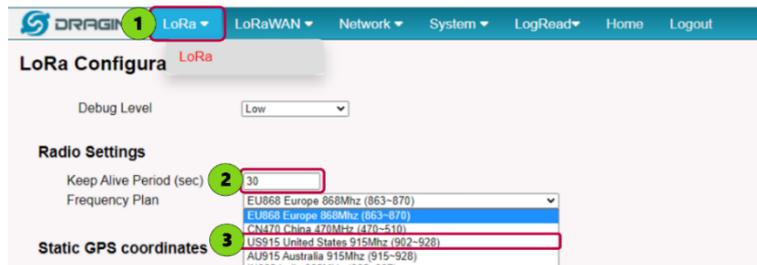
A continuación, se accede a la opción System > Network, donde se selecciona la pestaña WiFi para poder realizar la configuración de la conexión inalámbrica. Cabe señalar que en esta última sección se establece el SSID al que corresponde el nombre *Unach2025* y posteriormente, se le asigna la contraseña adecuada.



**Ilustración 31.** Captura de configuración de propiedades Gateway para envío de datos.

## 2. Configurando LoRaWAN y TTN

A partir de la interfaz web del Gateway, partimos a la sección LoRa > *LoRa Configuration* donde se configura los parámetros para el funcionamiento del protocolo LoRa. Inicialmente, seleccionamos el *Frequency Plan* correspondiente a “US915 United States 915MHz (902 a 928 MHz)”, que desempeña con las regulaciones de frecuencia determinadas para el área de despliegue. Asimismo, se concreta el *Keep Alive Period* de 30 segundos. Por medio de ese valor admite mantener activa la conexión entre el Gateway y la red de TTN, avalando así una comunicación perpetua y constante.



**Ilustración 32.** Configurando la frecuencia que opera el Gateway para la transmisión de data.

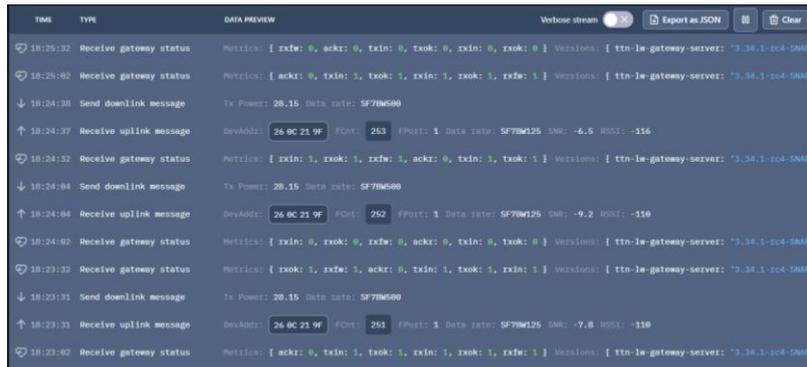
Para activación de la funcionalidad LoRaWAN, primeramente, en la plataforma TTN, se consiguió generar el identificador único del dispositivo (Gateway EUI). Dada esta información, se procedió a completar los campos que eran necesarios rellenar en la configuración del Gateway, donde se encontraba el EUI y la dirección de correo del propietario; de la misma manera, se determinó qué servidor era el apropiado en función de la región que seleccionamos, eligiendo por *nam1.cloud.thethings.network*, como servidor que trabajaba para Norteamérica, el cual resulta compatibles con las distintas bandas de frecuencia que se utilizan y se encuentran en Ecuador.



**Ilustración 33.** Configurando el UEI del Gateway para la transmitir datos.

## 3. Pruebas y Funcionamiento:

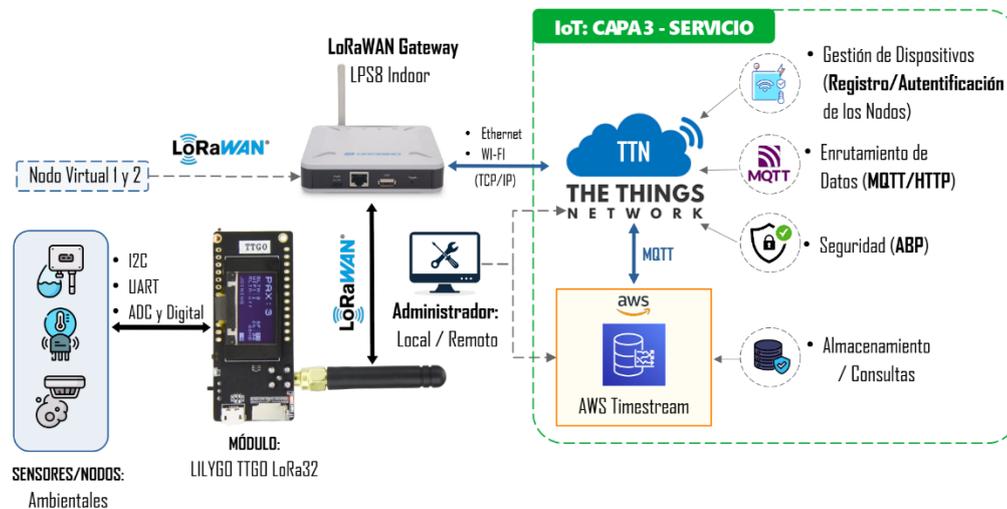
El Gateway LPS8v2 se ha probado y validado con respecto a la consola del TTN, donde se reciben tramas uplink enviadas desde el nodo TTGO LoRa32 con dirección “26 C2 21 9F” con incrementos en el FCnt desde 250 hasta 253. A su vez, todas las tramas tienen una tasa de entrega del 100%, en condiciones de calidad aceptables: en rango de RSSI entre -110 y -116 dBm, y SNR de entre -6.5 dB a -9.2 dB utilizando la modulación SF7BW125. Con esto, se valida el funcionamiento del Gateway LPS8v2 con las configuraciones realizadas en la red LoRaWAN y permite determinar que tanto los dispositivos, como el servidor de red tienen un funcionamiento efectivo.



**Ilustración 34.** Recaptando tramas de LoRaWAN en TTN desde el Gateway LPS8v2.

### 4.3 Implementando la Capa Servicios para Gestionar y Almacenar Datos

Para gestionamiento y almacenamiento los datos provenientes de los sensores, se efectúa la capa de servicio (Ilustración 35) adentro de la arquitectura IoT del sistema de monitorización. En la etapa, se compone el servidor de red de TTN, el mismo que actúa como mediador entre la capa de red y aplicación. El servidor TTN recoge los paquetes de data enviados por los sensores mediante la red LoRaWAN, valida su integridad y consecutivamente, los enruta hacia el sistema que almacena en la nube en este contexto una plataforma Cloud.



**Ilustración 35.** Implementando la capa Servicio: Configurando el servidor de red y almacenamiento del sistema de monitorización ambiental en tiempo real.

#### 4.3.1 Implementación y Configuración del Servidor de Red de The Things Network (TTN)

Seleccionar TTN como la plataforma para la capa de servicio de IoT en el sistema de monitoreo ambiental se debe a su compatibilidad, bajo costo y capacidades de manejo de datos en relación con el nodo TTGO LoRa32 y la puerta de enlace LPS8v2 operando en la banda US915 (902–928 MHz). TTN proporciona una infraestructura global de acceso abierto sin restricciones internacionales de servicio, lo que significa que los usuarios pueden utilizarlo para fines no comerciales, ofreciendo servicios de red gratuitos que incluyen la

conexión sin problemas de puertas de enlace y nodos a través de la activación ABP, a diferencia de otras plataformas como Lorient que requieren tarifas de suscripción.

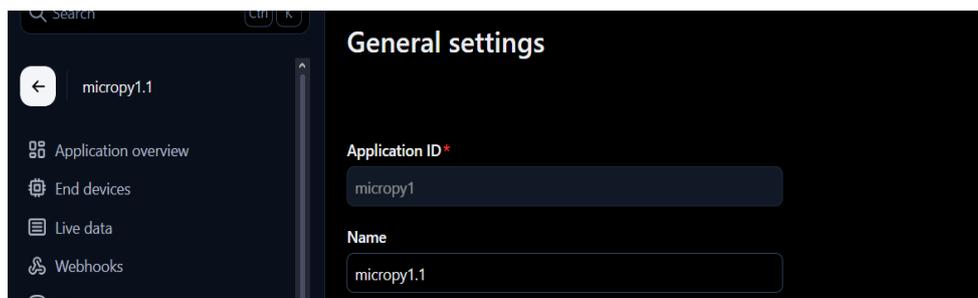
A diferencia de otros servidores como ChirpStack, que necesita la puesta en marcha y el mantenimiento de los servidores propios, TTN es capaz de ofrecer una aplicación en la nube preconfigurada y de alta disponibilidad. Además, la interfaz de TTN es capaz de decodificar payloads de forma sencilla mediante la utilización de funciones personalizadas con el uso de JavaScript y facilitar una exportación de datos mediante MQTT o HTTP que permite simplificar la integración de las aplicaciones destinadas a la visualización y el análisis. A continuación, se expone la descripción del método técnico continuo para la creación de la aplicación, al registrar el nodo y la estructura del decodificador del payload. El procedimiento parte del supuesto que se utiliza el Gateway LPS8v2 y el nodo TTGO LoRa32 configurados para realizar la transmisión de datos cada 30 segundos.

### 1. Generación de la Aplicación en TTN.

Se accede a la consola de TTN en el sitio [console.thethingsnetwork.org](https://console.thethingsnetwork.org). Dentro del apartado Applications, deberemos hacer clic en *Adding application* y se configura las siguientes propiedades:

- **Application ID:** micropy1
- **Application name:** micropy1.1

Tan pronto como se completa los campos anteriores, se guardó la aplicación. Ya que parte de estos campos, además, se verificó que el *Application EUI* resultara de forma automática por el sistema.



**Ilustración 36.** Configuración de TTN: Generación de la aplicación

Después, se fue a la parte de *Payload formatters* > *Uplink*, seleccionando como tipo de formateador *Custom JavaScript formatter*. y, en este campo, ubicamos el código correspondiente a la decodificación del payload en variables individuales (parámetros ambientales), con las validaciones de error de la decodificación del payload para garantizar que el análisis de los datos recibidos estuviese correctamente realizado.

---

**TTN: Formater code**

---

```
function Decoder(bytes, port) {
  var decoded = {};
  try {
    var str = String.fromCharCode.apply(null, bytes);
    var values = str.split('@');
    if (values.length < 7) {
      throw new Error(`Expected at least 7 values, got
${values.length}`);
    }
    decoded.temperature = {
      value: parseFloat(values[0]),
      unit: "°C" };
    decoded.humidity = {
      value: parseFloat(values[1]),
      unit: "%"};
    // Demás sensores y validaciones se visualizan en el Anexo 2
  } catch (error) {
    decoded.error = {
      message: "Decoding failed: " + error.message;};
  }
  return decoded;
}
```

---

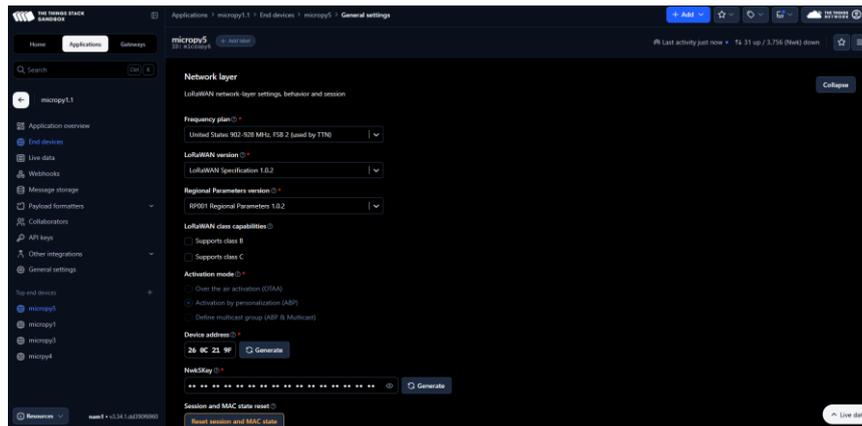
## 2. Registro del Nodo (End Device):

Internamente en la aplicación micropy1.1, se accede al componente *End devices* y se selecciona el campo *Add end device*. Posteriormente, se elige *Manually enter end device details* para incorporar manualmente la información del nodo y ajustar las siguientes propiedades:

**Tabla 12.** Configurando el dispositivo terminal (End Device) en TTN

Parámetro	Valor
End device ID	micropy5
DevEUI	70*****
End device name	name_nodo_unach1
Frequency plan	United States 902–928 MHz, FSB 2 (TTN)
LoRaWAN version	LoRaWAN Specification 1.0.2
Regional Parameters version	RP001 Regional Parameters 1.0.2
Activation mode	Activation by Personalization (ABP)
DevAddr	260C****
NwkSKey	2420****
AppSKey	2020****

Posteriormente, se guarda la configuración del dispositivo y se verifica el registro exitoso en la consola de TTN.



**Ilustración 37.** Configurando TTN: Registro de Nodos

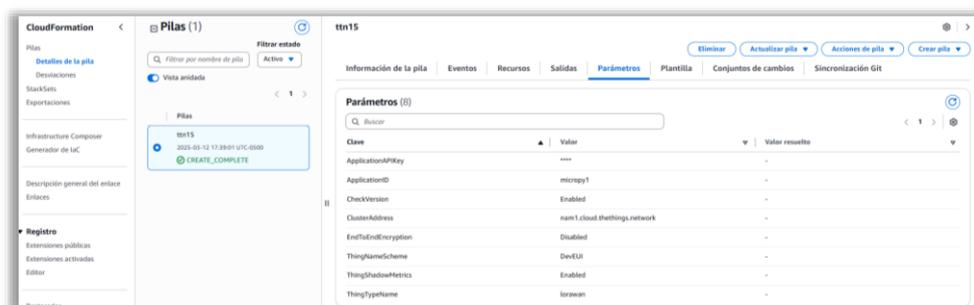
### 4.3.2 Configurando el Almacenamiento en AWS Timestream

Al elegir Amazon Timestream como la base de datos para almacenar variables ambientales se establece un diseño perfeccionado para la manipulación de series temporales de alta frecuencia, puesto que las tramas son entregadas cada 30 segundos por el nodo TTGO LoRa32, a través de TTN y el gateway LPS8v2, con un payload promediado de 56 bytes (SF7BW125) en este caso. A diferencia de las soluciones como Amazon RDS ( $\approx 0.10$  USD / GB) o DynamoDB. La compresión integrada y la estructura de costos se escriben más efectiva ( $\sim 0.03$  USD / GB, se escribe  $\sim 0.01$  USD cada mil consejos SQL).

Enfrentados con otras opciones especializadas como influxDB, requiere implementación e infraestructura especializada (o registro de nube desde  $\sim$  USD / GB), TimimetStream permite requisitos basados en SQL e integrado directamente en el grafano a través del complemento de datos AWS oficial, creando condiciones favorables Visitar y analizar medidas ambientales en tiempo real sin administrar servidores. A continuación, se detalla el procedimiento técnico efectuado para manipulación de los datos ambientales, donde la información decodificada desde TTN se transmite a través del protocolo MQTT a AWS IoT Core y a su vez se va almacenando de forma eficaz en Amazon Timestream.

#### 1. Configurando el Entorno en AWS a través de la plantilla CloudFormation

Primeramente, se accede a la consola de AWS y se selecciona el servicio CloudFormation, dentro de la región *US East (N. Virginia)*. Inmediatamente, se procede a creación de una nueva pila manipulando la plantilla prediseñada para automatizar la creación de los recursos obligatorios para integrar TTN con AWS IoT Core y Amazon Timestream.



### Ilustración 38. Configurando AWS: Generando la Pila (CloudFormation)

La pila fue establecida con el nombre *ttn15*, y los parámetros que se utilizaron son los siguientes:

**Tabla 13.** Configurando los parámetros de pila en AWS CloudFormation

Aspecto	Configuración
ApplicationAPIKey	(Clave secreta proporcionada por TTN)
ApplicationID	micropy1
ClusterAddress	nam1.cloud.thethings.network
ThingNameScheme	DevEUI
ThingShadowMetrics	Enabled
ThingTypeName	lorawan

Ya desplegado, el estado de la pila cambia a *CREATE\_COMPLETE*, ratificando la correcta creación del recurso.

## 2. Registro y Configuración del Dispositivo LoRaWAN en AWS IoT Core

En esta parte, un nuevo dispositivo (*Thing*) se almacena en el núcleo de AWS IoT y se transforma en la identidad principal. La característica asignada al objeto se especifica seguidamente:

- **DevEUI:** 70B3D57ED006B6B7
- **stackName:** ttn\_15
- **deviceID:** micropy5

Este ajuste admite la gestión personal del dispositivo en el ambiente AWS IoT.



**Ilustración 39.** Registrando y Configurando el dispositivo LoRaWAN en AWS IoT

## 3. Integrando TTN con AWS a través de MQTT

Por consiguiente, se genera una regla de ingesta de mensajes MQTT llamada *HandleUplinkRule\_tJSRvQskzaE*, que filtra los mensajes uplink provenientes de TTN. Para consiguiente, se manipuló la siguiente instrucción SQL:

---

**AWS: HandleUplinkRule\_tJSRvQskzaE**

---

```
SELECT * FROM 'thethings/lorawan/ttn15/uplink'
```

---

Aquella regla admite la capturan de los datos receptados por los nodos LoRaWAN que se envían a TTN y se procesa por dentro del ambiente de AWS.

#### 4. Almacenando Datos ambientales en Amazon Timestream(Base de datos)

Ingresando al servicio Amazon Timestream, se escoge la base de datos ttndb3 y se creó la tabla llamada tabla2, configurada con los siguientes factores:

- **Retención en memoria:** 12 horas
- **Retención en almacenamiento magnético:** 10 años
- **Claves de partición:** personalizadas
- **Escrituras:** habilitadas

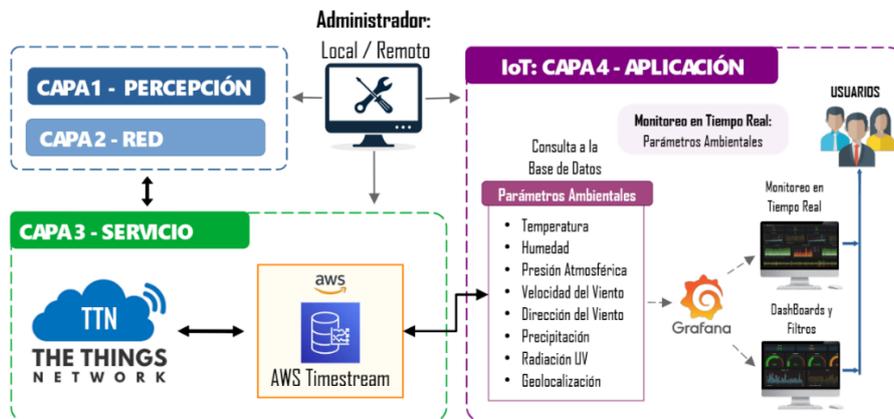


**Ilustración 40.** Configurando el Almacenamiento de Datos ambientales en Amazon Timestream

Este procedimiento permite garantizar un acopio eficaz de datos de series temporales, que es almacenado convenientemente tanto a breve como a extenso plazo.

#### 4.4 Instauración de la Capa Aplicación para Monitorización en Tiempo Real

Finalmente, con esta capa de la arquitectura IoT se establecen los programas informáticos responsables de la representación y administración de los datos recolectados. En este proyecto, se pone en marcha la plataforma Grafana, que facilita el seguimiento en tiempo real de los parámetros ambientales mediante representaciones gráficas dinámicas.



**Ilustración 41.** Instauración de capa Aplicación: Ajuste de las aplicaciones.

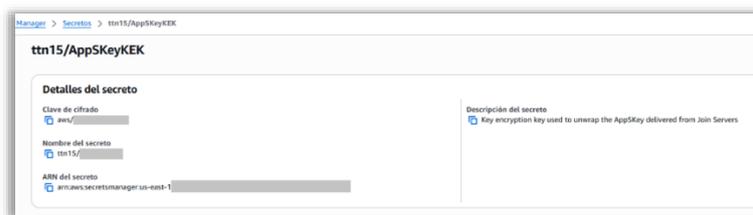
#### 4.4.1 Combinación de Grafana para Monitorización de Parámetros Ambientales

La decisión de optar por Grafana como instrumento para visualizar los datos ambientales guardados en Amazon Timestream se fundamenta en su habilidad para crear paneles interactivos, altamente personalizables y con integración directa con Timestream. Esto facilita consultas de baja latencia (cerca de 100 ms por consulta) y una visualización eficaz de series de tiempo. Grafana promete dos expectativas significantes: la versión tramitada por AWS y la versión autotramitada en instancias EC2, las mismas que ofrecen un ambiente configurable y escalable. A diferencia, de herramientas como es Kibana, que posee máxima complejidad para tratar datos temporales, también, Power BI, que se enfoca en estudio de negocios, Grafana es más correcto para circunstancias de monitorización ambiental.

El establecimiento del complemento Amazon Timestream Data Source facilita efectuar consultas SQL de forma directa en la base de datos `ttn15.tabla2` (región US East - N. Virginia), utilizando credenciales dispuestas de seguramente a través del gestor de secretos AWS. Así, la integración aclara la representación de métricas procedentes de TTN (formato JSON decodificador), deslizando la complicación operativa de soluciones como InfluxDB + Chronograf, que solicitan infraestructura extra y un especialista en el área. Adicionalmente, Grafana facilita el desarrollo de dashboards interactivos, la integración de alertas en tiempo real y envío de gráficos, permitiendo que sea accesible, escalable y una experiencia excelente al usuario. Todo esto lo hace una opción apropiada para contextos rurales donde se implementa el sistema de vigilancia, en concordancia con los objetivos de accesibilidad, eficacia y análisis del entorno establecidos en este proyecto de estudio. Para hacer funcionar Grafana y visualizaciones de la información ambiental almacenada en Amazon Timestream, se realiza las siguientes acciones:

##### 1. Estableciendo la Procedencia de Datos que llegan a Grafana:

Para generar una vinculación positiva entre Grafana y Amazon Timestream, se aprovechó el sistema que administra de credenciales de AWS Secrets Manager. A partir de la consola de AWS, en la región US East (N. Virginia), se generó un nuevo secreto eligiendo la opción “*Other type of secret*”. En el mismo, se almacenan las claves de ingreso con permisos determinados (`timestream:Select`, `timestream:DescribeTable`), asignándole el nombre `ttn15/AppSKeyKEK` y agregando la descripción que permite identificar el objetivo para autenticar en Grafana.



**Ilustración 42.** Ajustes de Almacenamiento de Datos en Amazon Timestream

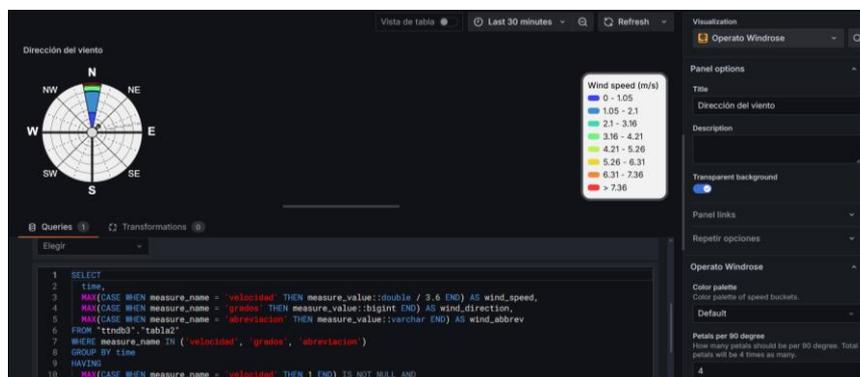
En el proceso de creación, se establecieron las políticas de acceso para limitar la utilización del secreto a los servicios autorizados, garantizando de esta manera el acatamiento de buenas

prácticas de seguridad. Al concluir, se extrajo el ARN del secreto, que luego se empleó en la configuración del origen de datos en Grafana, lo que permitió la realización de consultas sobre Timestream sin revelar las credenciales directamente en el ambiente de visualización.

## 2. Diseñando el Tablero(Dashboard) y Visualizando Datos:

Dada esta sección, se manifiesta la creación de un tablero de control personalizando la variable de dirección del viento, tomándolo como una muestra representativa, dado que el proceso es similar para las demás visualizaciones, se cambia el tipo de panel y se codifica como se requiere el gráfico. En este tablero, se creó un panel en Grafana manipulando el plugin Operato Windrose, que proporciona la representación en tiempo real de una rosa de los vientos.

En esta ilustración se visualiza la dirección absoluta y la fuerza del viento a través áreas circulares y escalas cromáticas.



**Ilustración 43.** Diseño de tablero para la dirección de viento y velocidad del viento en Grafana

Desde TTN se envía los datos ambientales, se receipta directamente de AWS Timestream, haciendo uso del consultor SQL que obtiene las variables de velocidad, dirección en grados y abreviatura cardinal, las ordena y grafica.

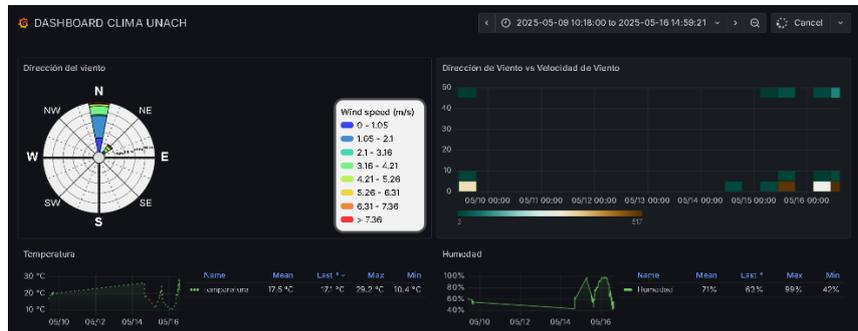
### **Grafana: Dirección de viento y Velocidad del viento - SQL**

```

SELECT
  time,
  MAX(CASE WHEN measure_name = 'velocidad' THEN measure_value::double /
3.6 END) AS wind_speed,
  MAX(CASE WHEN measure_name = 'grados' THEN measure_value::bigint END) AS
wind_direction,
  MAX(CASE WHEN measure_name = 'abreviacion' THEN measure_value::varchar
END) AS wind_abbrev
FROM "ttndb3"."tabla2"
WHERE measure_name IN ('velocidad', 'grados', 'abreviacion')
GROUP BY time
HAVING
  MAX(CASE WHEN measure_name = 'velocidad' THEN 1 END) IS NOT NULL AND
  MAX(CASE WHEN measure_name = 'grados' THEN 1 END) IS NOT NULL AND
  MAX(CASE WHEN measure_name = 'abreviacion' THEN 1 END) IS NOT NULL
ORDER BY time DESC

```

Así, el panel se ajusta para dar información reciente, establecida en categorías de velocidad programados por color. La herramienta ofrece una visión nítida y eficaz del comportamiento del viento, provechosa para el análisis del clima y la toma de decisiones en directo.



**Ilustración 44.** Implementando el tablero para la dirección viento y velocidad del viento.



**Ilustración 45.** Implementado del sistema de monitorización ambiental en tiempo real

De esta manera, el sistema de monitorización IoT/LoRaWAN, compuesto con Grafana (Ilustración 46), posibilita la visualización en el momento, de las variables ambientales, exponiendo datos históricos que se almacenaron en AWS Timestream, lo que asevera la disponibilidad momentánea de información ambiental al instante.



**Ilustración 46.** Visualizaciones del sistema de monitorización ambiental desarrollado en Grafana

#### 4.4.2 Evaluando y comparando el sistema IoT/LoRaWAN vs Estación ESPOCH

En la determinación si el sistema de monitorización ambiental implementado cumple con las características concretas en la hipótesis, se ejecuta un análisis comparativo con una estación meteorológica perteneciente a la Universidad Politécnica de Chimborazo (ESPOCH).



<b>Nombre de la Estación Meteorológica:</b>	ESPOCH
<b>Parroquia:</b>	Lizarzaburu
<b>Cantón:</b>	Riobamba
<b>Latitud:</b>	9816965
<b>Longitud:</b>	758398
<b>Altura (m):</b>	2754

*Ilustración 47. Características de Estación meteorológica de la ESPOCH*

- **Comparación Estudio Técnico de los Sensores Ambientales entre el Sistema IoT/LoRaWAN y la Estación ESPOCH**

La Tabla 14 siguiente presenta la comparación entre los sensores del sistema IoT/LoRaWAN y la estación meteorológica de ESPOCH. Ambos registran variables tales como temperatura, humedad, presión, viento, lluvia y localización. No obstante, el sistema IoT incorpora un sensor de radiación UV, lo que supone un beneficio al expandir el espectro de vigilancia ambiental.

**Tabla 14. Comparación Estudio Técnico de Sensores: Sistema IoT/LoRaWAN vs. Estación Meteorológica ESPOCH**

<b>Medición</b>	<b>Sistema de Monitoreo Ambiental IoT/LoRaWAN</b>	<b>Estación Meteorológica ESPOCH (Lizarzaburu)</b>
Temperatura	SHT20 (rango: -40 a 120 °C, $\pm 0.3$ °C)	Vaisala HMP155 (-80°C a +60°C, $\pm 0.1$ °C)
Humedad Relativa	SHT20 (rango: 0-100% RH, $\pm 3\%$ RH)	Vaisala HMP155 (0-100% HR, $\pm 1\%$ )
Presión Atmosférica	BMP280 (rango: 300-1100 hPa, $\pm 0.12$ hPa)	Vaisala BARO-1 (600-1100 hPa, $\pm 0.3$ hPa)
Velocidad del Viento	SEN0170 (rango: 0-70 km/h, $\pm 0.1$ m/s)	Vaisala WM30 (0-60 m/s, precisión $\pm 0.3$ m/s)
Dirección del Viento	PR-3000-FXJT (rango: 0-360°, $\pm 10^\circ$ )	Vaisala WM30 (0°-360°, precisión $\pm 3^\circ$ )
Radiación UV	GYML8511 (rango: 0.1-15 mW/cm <sup>2</sup> , $\pm 10\%$ )	No disponible
Precipitación	KY-024 (3.5 mm/pulso, $\pm 1\%$ )	Young 51203 (pluviómetro tipo cangilón, resolución 0.1 mm)
Geolocalización	Ublox NEO-6M (precisión: $\pm 2.5$ m)	No disponible (ubicación fija)

Sin embargo, la estación meteorológica de la ESPOCH posee sensores de alta gama con niveles de precisión altos, el sistema IoT/LoRaWAN consigue resultados semejantes haciendo uso de sensores de poco costo, lo que expone la viabilidad técnica y financiera de su establecimiento en circunstancias de recursos finitos.

- **Semejanza Técnica de arquitecturas entre la Estación IoT/LoRaWAN y Estación ESPOCH**

Por consiguiente, para realización de comparación técnica entre el Sistema de Monitoreo Ambiental IoT/LoRaWAN planteado y la Estación Meteorológica de la ESPOCH, se desarrolla la siguiente tabla con las particularidades técnicas de ambos sistemas.

*Tabla 15. Contraste Técnico entre arquitecturas y seguridad: Sistema IoT/LoRaWAN frente a la Estación de Tiempo ESPOCH*

<b>Característica</b>	<b>Sistema de Monitoreo Ambiental IoT/LoRaWAN</b>	<b>Estación Meteorológica ESPOCH (Lizarzaburu)</b>
<b>Arquitectura del sistema</b>	IoT (basada en IEEE P2413.1, 4 capas)	Convencional (sistema propietario: Vaisala)
<b>Protocolo de Comunicación</b>	LoRaWAN (banda ISM 915 MHz), largo alcance,	GSM, RS-232, RS-485
<b>Seguridad</b>	ABP (autenticación por claves preconfiguradas)	No especificado (típico: autenticación básica)
<b>Base de datos</b>	Nube (AWS Timestream)	Memoria interna 3.3 MB + CF externa hasta 2 GB
<b>Movilidad</b>	Móvil (diseño compacto)	Estación fija, mástil de 9 m
<b>Monitoreo en Tiempo Real</b>	Sí (visualización en Grafana)	limitada, con MAWS Terminal/YourView
<b>Precio aproximado</b>	Bajo (estimado: ~\$500 USD)	Alto (aproximado: >\$9900 USD)
<b>Dimensiones del Sistema</b>	Compacto (aprox. 2.40x0.80x0.50 m)	Grande (aproximado 15x30x2 m, incluye mástil)
<b>Consumo Energético</b>	Bajo	Variable, generalmente mayor

El Sistema IoT/LoRaWAN sugerido constituye una opción contemporánea y eficaz en contraposición a la Estación de la ESPOCH, especialmente en términos de su arquitectura, tecnología de comunicación y seguridad. Aunque el sistema desarrollado se fundamenta en una arquitectura de cuatro capas acorde al estándar IEEE P2413.1 y emplea LoRaWAN en la banda ISM de 915 MHz, perfecto para ambientes rurales debido a su largo alcance y bajo consumo de energía, la estación ESPOCH utiliza tecnologías propias como GSM y RS-232, que son menos eficaces y demandan una infraestructura más grande.

En el sector de seguridad y almacenamiento, el sistema IoT sobresale por su implementación de autenticación ABP y cifrado LoRaWAN, además de ofrecer una solución en la nube escalable a través de AWS Timestream. No obstante, la estación convencional no cuenta con sistemas de seguridad avanzados y posee capacidad de almacenamiento limitado que es a 3.3 MB de memoria interna y una tarjeta CF de 2 GB. Además, el sistema sugerido brinda beneficios considerables en términos de portabilidad, vigilancia en tiempo real, gastos y eficiencia energética. Su diseño reducido (2.40x0.80x0.50 m), facilita su despliegue en

diversos contextos y ofrece una visualización instantánea de datos mediante plataformas como Grafana.

En contraste, la estación ESPOCH, que tiene un mástil fijo de 9 metros y medidas que se aproximan a 15x30x2 m, cuenta con una instalación estática que imposibilita su desplazamiento y solo permite el análisis futuro de información a través de herramientas exclusivas como MAWS Terminal o YourView. En la parte de precio y sostenibilidad, el sistema IoT/LoRaWAN llega a un costo cercano a los 500 USD y un consumo energético notablemente reducido, en discordancia con el sistema convencional, que sobrepasa los 9900 USD y requiere energía de la red eléctrica, más baterías, lo que lo hace una opción menos accesible para implementaciones en áreas rurales.

- **Valoración del rendimiento del protocolo LoRaWAN de comunicación en el sistema IoT/LoRaWAN**

Para analizar de manera técnica el rendimiento del protocolo de comunicación LoRaWAN en el sistema de vigilancia ambiental sugerido, se llevaron a cabo ensayos experimentales enfocados en evaluar la eficacia en la transmisión de datos a partir de la capa de percepción incluso la capa de aplicación, (IoT - IEEE P2413.1). El Gateway (LPS8V2) se ubicó de manera firme en el campus Edison Riera de la Universidad Nacional de Chimborazo, situado en las longitudes  $-1.65344^\circ$  latitud y  $-78.64404^\circ$  longitud, a una altitud cercana de 2800 metros sobre el nivel del mar.



**Ilustración 48.** Medición límite de cobertura del Gateway LPS8V2 y el Nodo LoRaWAN

Consecutivamente, se procedió a trasladar el sistema IoT/LoRaWAN en distintos puntos estratégicos semiurbanos de la ciudad de Riobamba, con la finalidad de medir la conducta del enlace en diversos ambientes de distancia y entorno. En cada sitio, se tomaron 100 muestras por punto, inspeccionando los promedios de SNR (*Signal-to-Noise Ratio*) y RSSI (*Received Signal Strength Indicator*). En el cual, el SNR muestra la relación entre la señal útil y el ruido dado en el canal de comunicación, también, que el RSSI toma la medida de la potencia recibida de la señal. Consecuentemente, los dos parámetros admiten valorar la calidad y estabilidad del enlace LoRaWAN motivado por algunos rangos específicos (Tabla 16).

**Tabla 16. Clasificando los rangos de niveles SNR y RSSI**

Métrica	Óptimo	Aprobado	Pobre
SNR (dB)	> 10	10 a -7	< -7
RSSI (dBm)	≥ -80	-80 a -120	< -120

Como se visualiza en la Tabla 17, se evidencia que en trayectos cortos (hasta ~2.5 km), los valores promedio de SNR se conservan en niveles positivos, logrando hasta +12.8 dB (coordenada -1.645546, -78.651931), muestra una excelente relación señal/ruido. Juntamente, los valores de RSSI en esa categoría oscilan entre -79.3 dBm y -109 dBm, es estimado correcto para LoRaWAN, sin embargo, se visualizan atenuaciones más severas a medida que se aumenta la distancia.

**Tabla 17. Medición de Comunicación LoRaWAN con relación a la Distancia**

Coordenada: Gateway		Distancia (km)	Media	
Latitud	Longitud		SNR (dB)	RSSI (dBm)
-1.652723	-78.641178	0.33	10.1	-81.3
-1.650352	-78.645851	0.40	10.4	-86.0
-1.648840	-78.648698	0.73	11.3	-79.3
-1.645546	-78.651931	1.24	12.8	-86.0
-1.643596	-78.652108	1.41	9.3	-100.0
-1.642748	-78.658928	2.03	1.1	-106.0
-1.638302	-78.657367	2.24	11.1	-92.7
-1.641008	-78.663016	2.52	2.7	-103.0
-1.605110	-78.690415	7.43	-4.5	-108.0
-1.602195	-78.693534	7.90	-3.6	-115.0

No obstante, al exceder los 7 km, se nota una degradación considerable. En la ubicación más remota (-1.602195, -78.693534), el SNR disminuye a -3.6 dB y el RSSI baja hasta -115 dBm, lo que se aproxima al límite inferior sugerido para enlaces sólidos en LoRaWAN. Aunque, la transmisión no se dificultó, lo que permite validar la robustez del protocolo y como tolera los inferiores márgenes de señal, inclusive en condiciones urbanas con posibles interrupciones e impedimentos físicos.



**Ilustración 49. Medición RSSI en dBm del Sistema IoT/LoRaWAN**

Para finalizar, los datos obtenidos de SNR y RSSI (Ilustración 49), puestos en la Tabla 17, se hallan en su mayoría dentro de los rangos excelente y tolerable que se definió en la Tabla 16, lo que demuestra un rendimiento apropiado del sistema. Así, se establece que la calidad del enlace LoRaWAN, inclusive en entornos semiurbanos y a trayectos de llegan a 7.9 km, se queda dentro de los parámetros deseados para una transmisión eficaz, validando así la eficacia del protocolo en condiciones existentes de operación.

#### **4.5 Análisis a través de Estadística general**

Para admitir el funcionamiento del sistema de monitorización ambiental establecido en IoT/LoRaWAN, se efectuó un análisis estadístico de los datos medidos para los parámetros de temperaturas, humedades y presiones atmosféricas. A partir, el análisis tuvo una comparación directa con la información proporcionada por la estación meteorológica de la ESPOCH.

Es necesario, entender que algunos sensores de la ESPOCH, como los de velocidad del viento, dirección del viento y precipitación, fueron instalados a una altura cercana de 10 metros, por lo tanto, se crea discrepancias en las medidas respecto al sistema IoT/LoRaWAN. Por ende, la validación de estos sensores se hizo previamente mediante un proceso de calibración, explicado al empiece de la sección de metodología, haciendo uso de sensores estandarizados que son referencia.

##### **4.5.1 Análisis de Estadístico: Entre el sistema IoT/LoRaWAN vs estación ESPOCH**

Con la finalidad de obtener la eficacia en el sistema de monitoreo de temperatura a partir del sistema de referencia (ESPOCH) y apoyado en un sistema de monitoreo remoto LoRaWAN (tecnología de Internet de las cosas a baja potencia), se han obtenido los resultados necesarios a través de SPSS e inmediatamente validar la precisión del sensor IoT en condiciones reales, esta última sea una de las consideraciones importantes para el uso en protocolos de telecomunicaciones, agricultura de precisión o smart cities.

Tamaño de Muestra: es de Tipo determinístico formado por 432 datos de cada variable meteorológica: Temperatura, Humedad y Presión atmosférica que se toman cada 10 minutos en la ciudad de Riobamba.

##### **4.5.1.1 Evaluación de Correlación de Kendall**

La correlación Tau de Kendall es una medida no paramétrica que estudia la fuerza y dirección de la relación presente entre dos variables a partir de sus rangos, lo que le permite ser utilizada con datos escalados no normales.

**Tabla 18. Hipótesis Estadística de Temperatura LoRaWAN y Temperatura ESPOCH**

Establecer la hipótesis nula y alternativa: H0: No existe asociación significativa entre las variables ( $\tau = 0$ ) Ha: Existe asociación significativa entre las variables ( $\tau \neq 0$ )	
Nivel de significancia	$\alpha = 0.05$
P-valor	0.001 es menor a 0.05
$\tau$	0.8536193
Tomar la decisión: Se rechaza la hipótesis nula, existe asociación significativa entre las variables al 95% de confianza.	

**Interpretación:**

Al tener  $\tau = 0.8536193$ , se reconoce una asociación muy fuerte y estadísticamente significativa  $p < 0.001$  entre las mediciones de temperatura obtenidas por ESPOCH y LoRaWAN. Además, la concordancia en las tendencias es muy significativa, lo que revela que ambos dispositivos captan los cambios térmicos en similares proporciones en el tiempo.

**Tabla 199. Hipótesis Estadística de Humedad LoRaWAN y Humedad ESPOCH**

Establecer la hipótesis nula y alternativa: H0: No existe asociación significativa entre las variables ( $\tau = 0$ ) Ha: Existe asociación significativa entre las variables ( $\tau \neq 0$ )	
Nivel de significancia	$\alpha = 0.05$
P-valor	0.001 es menor a 0.05
$\tau$	0.9236928
Tomar la decisión: Se rechaza la hipótesis nula, existe asociación significativa entre las variables al 95% de confianza.	

**Interpretación:**

Se aprecia la asociación más fuerte de todas ( $\tau \approx 0.92$ ), mostrando una coincidencia casi perfecta en las mediciones de la humedad, lo que implica que ambos sistemas son muy estables para esta variable, incluso en situaciones de clima variable.

**Tabla 200. Hipótesis Estadística de Presión Atmosférica LoRaWAN y Presión Atmosférica de la Estación de Referencia**

Establecer la hipótesis nula y alternativa: H0: No existe asociación significativa entre las variables ( $\tau = 0$ ) Ha: Existe asociación significativa entre las variables ( $\tau \neq 0$ )	
Nivel de significancia	$\alpha = 0.05$
P-valor	0.001 es menor a 0.05
$\tau$	0.7622201

---

Tomar la decisión:  
Se rechaza la hipótesis nula, existe asociación significativa entre las variables al 95% de confianza.

---

**Interpretación:**

Correlación fuerte, con  $\tau=0.7622201$ , escasamente significancia  $p<0.001$ , un tanto inferior a las anteriores. Indica que, aunque se presenta una clara asociación, puede haber factores de tipo técnico o ambiental que puedan proporcionar más variabilidad en las mediciones de presión en el contexto de las mediciones tomadas en los diferentes dispositivos.

**4.5.1.2 Pruebas de Normalidad**

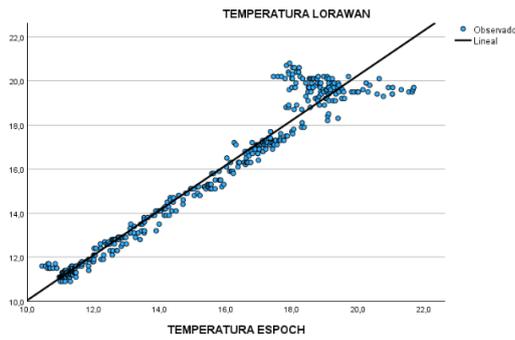
Una vez establecidas las hipótesis y propuestas, se llevó a cabo el test de Lilliefors (Kolmogorov-Smirnov) en R Commander para comprobar la normalidad de las variables y diferencias y si se requerían para su contraste métodos paramétricos o no paramétricos. Este paso es diagnóstico, ya que evalúa el cumplimiento de los supuestos de las pruebas estadísticas elegidas de acuerdo con las hipótesis.

***Tabla 211.** Test de normalidad de las variables Temperatura, Humedad, y Presión Atmosférica*

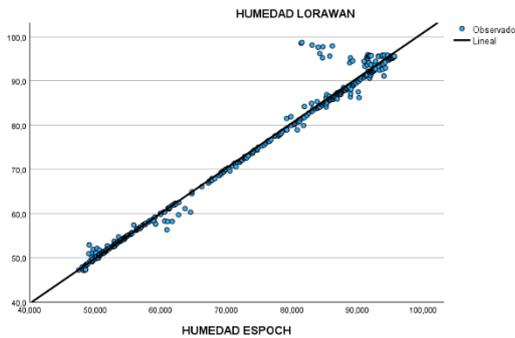
Variables	Estadístico de Lilliefors	P-valor	Decisión al 95% de confianza
Temperatura LoRaWAN	D = 0.12435	$<2.2e-16$	No se distribuye normalmente
Temperatura ESPOCH	D = 0.11077	$<7.499e-14$	No se distribuye normalmente
Humedad LoRaWAN	D = 0.15605	$<2.2e-16$	No se distribuye normalmente
Humedad ESPOCH	D = 0.15559	$<2.2e-16$	No se distribuye normalmente
Presión Atmosférica LoRaWAN	D = 0.037904	$<0.1377$	Se distribuye normalmente
Presión Atmosférica ESPOCH	D = 0.055942	$<0.002501$	No se distribuye normalmente

Las diferencias entre dispositivos presentaron distribuciones no-normales en todas las variables (Lilliefors  $p < 0.001$ ) con los valores  $\tau = 0.861$  (temperatura),  $\tau = 0.448$  (humedad) y  $\tau = 0.755$  (presión), justificando así la aplicación de pruebas no paramétricas, especialmente la correlación Tau de Kendall para las correlaciones.

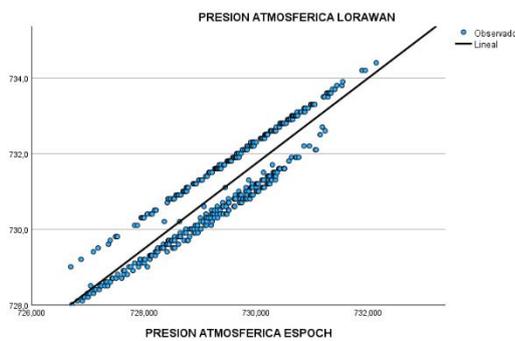
De igual modo, las gráficas que aportan ilustraciones de dispersión con ajuste lineal sirven para ilustrar una correlación positiva significativa, entre los datos obtenidos por parte del sistema LoRaWAN y la estación meteorológica de la ESPOCH, respecto de la temperatura, la humedad relativa, y la presión atmosférica. En todos los casos los puntos están muy cercanos a la línea de regresión, lo cual demuestra una elevada coincidencia de ambas fuentes de medición. La pendiente en torno a 1 en cada gráfico significa que existe una relación directa proporcional, y particularmente evidentes son las situaciones de temperatura y humedad, donde se aprecian mínimas desviaciones.



**Ilustración 50.** Diagrama sobre la dispersión incluyendo regresión lineal: Temperatura (LoRaWAN vs Estación ESPOCH)



**Ilustración 51.** Diagrama sobre la dispersión incluyendo regresión lineal: Humedad Relativa (LoRaWAN vs Estación ESPOCH)



**Ilustración 52.** Diagrama sobre la dispersión incluyendo regresión lineal: Presión atmosférica (LoRaWAN vs Estación ESPOCH)

## CAPITULO V. CONCLUSIONES Y RECOMENDACIONES

### 5.1 Conclusiones

- Al llevar a cabo el sistema de monitorización medioambiental en tiempo real utilizando el Internet de las cosas y LoRaWAN se fue desarrollando bajo la inclusión de la arquitectura IoT de cuatro capas indicada por el estándar IEEE P2413.1. Los resultados obtenidos contrastados mediante la realización del análisis estadístico comparativo correspondiente con la estación meteorológica de la ESPOCH, se han verificado evidenciando una alta correlación entre ambos sistemas analizados ( $R^2 = 0.982$  para las variables correspondientes a temperatura, humedad relativa y presión atmosférica). Dicho resultado queda confirmado y soporta el carácter técnicamente sólido que posee el sistema que se ha desarrollado, configurándose, así como una alternativa válida y eficaz frente a los sistemas tradicionales de monitoreo medioambiental (en términos de precisión, coste y escalabilidad).
- La integración de sensores de bajo coste con la combinación de protocolo de comunicación LoRaWAN ha permitido recoger y transmitir datos de parámetros ambientales con calidad semejante al de equipos comerciales de alta gama (temperatura, humedad relativa, presión atmosférica, velocidad y dirección del viento, precipitación, radiación UV y geolocalización) superando limitaciones de tecnologías convencionales (portabilidad, escalabilidad y bajo consumo de energía (20 a 50 mA en modo transmisión)).
- Las valoraciones experimentales del protocolo LoRaWAN, realizadas en ambientes semiurbanos de Riobamba, con distancias de hasta 7.9 km, muestran un rendimiento ideal con valores medios de SNR oscilando entre 1.1 y 12.8 dB y RSSI de entre -79.3 y -115 dBm, entre límites aceptables de transmisión fiable. Esto confirma su idoneidad para implementación en terrenos acondicionados y despoblados, mostrando su rango de operación (hasta 15 km) y su coste de implementación (cerca de USD 500 por nodo).
- Además, la puesta en práctica de la arquitectura del sistema IoT, que consta de la capa de percepción con los módulos TTGO LoRa32, la capa de red con el Gateway LPS8v2, la capa de servicio con TTN y AWS Timestream y la capa de la aplicación con Grafana, ha hecho posible una transmisión eficiente (carga de 56 bytes cada 30 segundos), almacenamiento escalable y visualización en tiempo real de datos ambientales. La calibración previa de los sensores con estándares comerciales asegura igualmente la integridad y equivalencia de las mediciones, concordando con el objetivo de modernizar el monitoreo ambiental en comparación con las estaciones tradicionales.
- Por su parte, el análisis comparativo demuestra que el sistema IoT/LoRaWAN presenta una mejor portabilidad que la estación ESPOCH (el diseño compacto de 15x30x2 cm frente a 2.40x0.80x0.50 m), un costo estimado de ~USD 500 en comparación con >USD 9900 o una mejor flexibilidad en las operaciones, sin perder competitividad, de modo que el sistema se presenta como una opción viable para aplicaciones en contextos con escasos recursos, especialmente en zonas rurales o de difícil acceso.

## 5.2 Recomendaciones

- Se propone realizar pruebas adicionales del sistema IoT/LoRaWAN en ambientes rurales con condiciones topográficas más difíciles, con el fin de analizar el efecto de los obstáculos físicos en los parámetros de calidad de señal, como el SNR y el RSSI. Esta evaluación optimizará tanto el modo de configuración del Data Rate (SF7BW125) como el diseño del posicionamiento del Gateway LPS8v2, para aumentar el alcance efectivo de la red, superando los 7.9 km que se han registrado en entornos semiurbanos.
- Se sugiere aumentar la funcionalidad de la capa de aplicación en Grafana añadiendo modelos predictivos basados en modelos predictivos fundamentados en el aprendizaje automático (regresión logística o redes neuronales recurrentes) para predecir tendencias del ambiente (picos de lluvia o radiación UV), utilizando los datos históricos almacenados en AWS Timestream y para optimizar el proceso de toma de decisiones en tiempo real.
- Paralelamente, se establece realizar un análisis de costes pormenorizado y un estudio de escalabilidad (de nodos adicionales >500) para determinar la rentabilidad y la viabilidad técnica de la implementación del sistema IoT/LoRaWAN en redes regionales de monitoreo ambiental, mediante la integración de múltiples Gateways LPS8v2 - suponiendo un rediseño de la infraestructura de TTN/AWS para gestionar volúmenes de datos superiores (payload >56 bytes).
- Se recomienda la integración de funciones de notificación(sms,email) ante condiciones críticas (radiación UV alta, lluvia excesiva) y la aplicación de modelos de machine learning sobre las series temporales almacenadas en AWS Timestream para la predicción de eventos.
- Para finalizar, se sugiere llevar a cabo un análisis espectral de la banda ISM 915 MHz en la ciudad de Riobamba para verificar interferencias específicas (señales Wi-Fi o dispositivos electrónicos), realizar un ajuste dinámico del Spreading Factor (SF7 a SF9) atendiendo a las condiciones locales, con la finalidad de robustecer el enlace LoRaWAN en entornos donde hay una alta densidad de dispositivos IoT.

## BIBLIOGRAFÍA

- [1] A. Fascista, “Toward Integrated Large-Scale Environmental Monitoring Using WSN/UAV/Crowdsensing: A Review of Applications, Signal Processing, and Future Perspectives,” *Sensors*, vol. 22, no. 5, 2022, doi: 10.3390/s22051824.
- [2] L. Lan, X. Le, H. Dong, J. Xie, Y. Ying, and J. Ping, “One-step and large-scale fabrication of flexible and wearable humidity sensor based on laser-induced graphene for real-time tracking of plant transpiration at bio-interface,” *Biosens. Bioelectron.*, vol. 165, p. 112360, 2020, doi: <https://doi.org/10.1016/j.bios.2020.112360>.
- [3] R. B. B., M. A. Haile, D. T. Haile, and D. Zerihun, “Real-Time Sensor Data Analytics and Visualization in Cloud-Based Systems for Forest Environment Monitoring,” *Int. J. Adv. Signal Image Sci.*, vol. 9, no. 1, pp. 29–39, 2023, doi: 10.29284/ijasis.9.1.2023.29-39.
- [4] M. A. López Muñoz *et al.*, “Wireless Dynamic Sensor Network for Water Quality Monitoring Based on the IoT,” *Technologies*, vol. 12, no. 11, 2024, doi: 10.3390/technologies12110211.
- [5] S. Ugwuanyi, G. Paul, and J. Irvine, “Survey of iot for developing countries: Performance analysis of lorawan and cellular nb-iot networks,” *Electron.*, vol. 10, no. 18, 2021, doi: 10.3390/electronics10182224.
- [6] M. C. Al Fajar and O. N. Samijayani, “Realtime Greenhouse Environment Monitoring Based on LoRaWANProtocol using Grafana,” in *2021 International Symposium on Electronics and Smart Devices (ISESD)*, 2021, pp. 1–5. doi: 10.1109/ISESD53023.2021.9501628.
- [7] L. M. Pires and J. Gomes, “River Water Quality Monitoring Using LoRa-Based IoT,” *Designs*, vol. 8, no. 6, 2024, doi: 10.3390/designs8060127.
- [8] H. Y. Miao, C. T. Yang, E. Kristiani, H. Fathoni, Y. S. Lin, and C. Y. Chen, “On Construction of a Campus Outdoor Air and Water Quality Monitoring System Using LoRaWAN,” *Appl. Sci.*, vol. 12, no. 10, 2022, doi: 10.3390/app12105018.
- [9] A. N. Rosli, R. Mohamad, Y. W. Mohamad Yusof, S. Shahbudin, and F. Y. Abdul Rahman, “Implementation of MQTT and LoRaWAN System for Real-time Environmental Monitoring Application,” in *ISCAIE 2020 - IEEE 10th Symposium on Computer Applications and Industrial Electronics*, 2020, pp. 287–291. doi: 10.1109/ISCAIE47305.2020.9108808.
- [10] CEPAL, “Acerca de Estadísticas ambientales y de los indicadores ambientales de los Objetivos de Desarrollo Sostenible (ODS),” *Naciones Unidas*, 2020. <https://www.cepal.org/en/node/41212> (accessed Feb. 03, 2025).
- [11] K. Shafique, B. A. Khawaja, F. Sabir, S. Qazi, and M. Mustaqim, “Internet of Things (IoT) for Next-Generation Smart Systems: A Review of Current Challenges, Future Trends and Prospects for Emerging 5G-IoT Scenarios,” *IEEE Access*, vol. 8, pp. 23022–23040, 2020, doi: 10.1109/ACCESS.2020.2970118.

- [12] M. G. D. Gololo *et al.*, “Review of IoT Systems for Air Quality Measurements Based on LTE/4G and LoRa Communications,” *Internet of Things*, vol. 5, no. 4, pp. 711–729, 2024, doi: 10.3390/iot5040032.
- [13] S. L. Ullo and G. R. Sinha, “Advances in smart environment monitoring systems using iot and sensors,” *Sensors (Switzerland)*, vol. 20, no. 11, 2020, doi: 10.3390/s20113113.
- [14] W. A. Jabbar *et al.*, “Development of LoRaWAN-based IoT system for water quality monitoring in rural areas,” *Expert Syst. Appl.*, vol. 242, p. 122862, 2024, doi: <https://doi.org/10.1016/j.eswa.2023.122862>.
- [15] D. I. Tirtakusuma *et al.*, “Real-time Monitoring Using AWS Cloud Platform For Agriculture Soil,” in *2024 IEEE International Symposium on Consumer Technology (ISCT)*, 2024, pp. 510–516. doi: 10.1109/ISCT62336.2024.10791164.
- [16] K. Anam, D. N. Rofi, and R. Meiyanti, “Monitoring System for Temperature and Humidity Sensors in the Production Room Using Node-Red as the Backend and Grafana as the Frontend,” *J. Syst. Eng. Inf. Technol.*, vol. 2, no. 2, pp. 68–76, 2023, doi: 10.29207/joseit.v2i2.5222.
- [17] P. L. Jethi, F. H. Pin, and E. A. Kumar, “Engineering Laboratory IAQ Monitoring: A Real-Time Solution using Raspberry PI and Grafana Dashboard,” *J. Adv. Res. Fluid Mech. Therm. Sci.*, vol. 124, no. 2, pp. 53–70, 2024, doi: 10.37934/arfmts.124.2.5370.
- [18] S. A. Guillén Meza, “Diseño, Construcción y Aplicación de un Dispositivo Electrónico Basado en Arduino para Monitoreo en Tiempo Real de Parámetros Ambientales,” Universidad Estatal del Sur de Manabí, 2022. [Online]. Available: <http://repositorio.unesum.edu.ec/handle/53000/4351>
- [19] A. P. Nugroho, L. Sutiarto, and T. Okayasu, “Appropriate adaptation of precision agriculture technology in open field cultivation in tropics,” *IOP Conf. Ser. Earth Environ. Sci.*, vol. 355, no. 1, 2020, doi: 10.1088/1755-1315/355/1/012028.
- [20] M. F. Acevedo, *Real-Time Environmental Monitoring: Sensors and Systems*, 2nd ed. Boca Raton, 2023. doi: 10.1201/9781003425496.
- [21] D. L. Cuasquer Chiscueth, “Aplicación de la tecnología IoT (Internet of Things) para la medición de variables meteorológicas en la agricultura sostenible,” Escuela Politécnica Nacional Facultad, 2022. [Online]. Available: <http://bibdigital.epn.edu.ec/handle/15000/22946>
- [22] T. Boris, “Clasificación de los climas del mundo según la temperatura y las precipitaciones como Koppen, Caldas, Lang, Martone y Thorntwaite.,” *Ingeniería Ambiental*, 2020. <https://ingenieriaambiental.net/clasificacion-climas/> (accessed Jan. 06, 2025).
- [23] B. Warthon *et al.*, “Correlacion de material particulado PM 2.5 en funcion a la humedad relativa - precipitacion usando python, en la ciudad del cusco,” vol. 5, pp. 1–15, 2023, [Online]. Available: <http://arxiv.org/abs/2303.00526>
- [24] M. Alugubelly, K. R. Polepalli, S. Gade, and S. Ninomiya, “Analysis of similar

- weather conditions to improve reuse in weather-based decision support systems,” *Comput. Electron. Agric.*, vol. 157, no. February, pp. 154–165, 2019, doi: 10.1016/j.compag.2018.12.010.
- [25] G. A. Cordoba Calvache, “Diseño de un prototipo de ajuste de presión atmosférica de una cámara altimétrica para pruebas de altímetros sensitivos por medio de visualización en Labview,” Universidad Distrital Francisco José de Caldas, 2022. [Online]. Available: <http://hdl.handle.net/11349/30254>
- [26] G. Parati *et al.*, “Clinical recommendations for high altitude exposure of individuals with pre-existing cardiovascular conditions,” *Eur. Heart J.*, vol. 39, no. 17, pp. 1546–1554, 2019, doi: 10.1093/eurheartj/ehx720.
- [27] K. Cagua Colmenares and V. H. A. Tatiana, “Construcción de un CanSat para la Medición de Temperatura , Presión Atmosférica y Altitud,” Fundación Universitaria Los Libertadores, 2022. [Online]. Available: <http://hdl.handle.net/11371/5622>
- [28] P. F. Bolaños Logroño, J. L. Yaulema Castañeda, O. D. Gavilánez Álvarez, and B. L. Ayavaca Vallejo, “Neural networks in predicting wind speed,” *Polo del Conoc.*, vol. 5, no. 05, pp. 4–40, 2020, doi: 10.23857/pc.v5i5.1398.
- [29] I. H. Bucak, H. Tanrıverdi, and F. E. Kılıç, “An evaluation of childhood carbon monoxide intoxications in a rural area using the Beaufort wind scale,” *Environ. Monit. Assess.*, vol. 1, pp. 1–11, 2022, doi: 10.21203/rs.3.rs-2266081/v1.
- [30] HIBA, “Sensores de dirección del viento,” *Hub Iberia Agrotech*, 2023. <https://learningdata.hubiberiaagrotech.eu/sensores-de-direccion-del-viento/> (accessed Mar. 06, 2025).
- [31] R. Avanzato and F. Beritelli, “An innovative acoustic rain gauge based on convolutional neural networks,” *Inf.*, vol. 11, no. 4, pp. 1–16, 2020, doi: 10.3390/info11040183.
- [32] E. D. Llamuca Perugachi, “Repotenciación del Sistema de Medición de Radiación UV de La Fie Incorporando Funcionalidades de IoT y Autonomía Eléctrica para Registro de Datos en Servidor IoT,” Escuela Superior Politécnica de Chimborazo, 2023. [Online]. Available: <http://dspace.esPOCH.edu.ec/handle/123456789/21618>
- [33] J. H. Colcha Ulcuango, “Development of a solar radiation measurement device with IoT monitoring based on,” *Rev. Soc. Front.*, vol. 4, no. 5, pp. 1–15, 2024, doi: 10.59814/resofro.2024.4(5)e459.
- [34] R. G. E. Daniel, “Diseño e Implementación de un Prototipo de Recolección de Datos y Geolocalización de Propósito Industrial e IoT para Sistemas Fotovoltaicos,” Universidad de los Llanos, 2020. [Online]. Available: <https://repositorio.unillanos.edu.co/handle/001/4265>
- [35] F. Firouzi, K. Chakrabarty, and S. Nassif, *Intelligent Internet of Things: From Device to Fog and Cloud*. 2020. doi: 10.1007/978-3-030-30367-9.
- [36] T. Lynn, P. T. Endo, A. M. N. C. Ribeiro, G. B. N. Barbosa, and P. Rosati, “The internet of things: definitions, key concepts, and reference architectures,” *cloud-to-*

*thing Contin. Oppor. challenges cloud, fog edge Comput.*, pp. 1–22, 2020.

- [37] O. Logvinov, “Standard for an architectural framework for the Internet of Things (IoT) - IEEE P2413,” *IEEE Standards Association*. MEPTEC, pp. 67–75, 2020. [Online]. Available: <https://www.meptec.org/Resources/8 - Logvinov.pdf>
- [38] M. YILDIRIM, U. DEMİROĞLU, and B. ŞENOL, “An in-depth exam of IoT, IoT Core Components, IoT Layers, and Attack Types,” *Eur. J. Sci. Technol.*, no. 28, pp. 665–669, 2021, doi: 10.31590/ejosat.1010023.
- [39] L. Kakkar, D. Gupta, S. Saxena, and S. Tanwar, “IoT Architectures and Its Security: A Review,” in *Proceedings of the Second International Conference on Information Management and Machine Intelligence*, D. Goyal, A. K. Gupta, V. Piuri, M. Ganzha, and M. Paprzycki, Eds., Singapore: Springer Singapore, 2021, pp. 87–94.
- [40] B. B. Gupta and M. Quamara, “An overview of Internet of Things (IoT): Architectural aspects, challenges, and protocols,” *Concurr. Comput. Pract. Exp.*, vol. 32, no. 21, pp. 1–24, 2020, doi: 10.1002/cpe.4946.
- [41] D. Calderón, F. J. Folgado, I. González, and A. J. Calderón, “Implementation and Experimental Application of Industrial IoT Architecture Using Automation and IoT Hardware/Software,” *Sensors*, vol. 24, no. 24, 2024, doi: 10.3390/s24248074.
- [42] R. G. Bracho Guamaní and D. E. Guañuna Andi, “Comparación Experimental De Protocolos De Comunicación De Iot En Cuanto a La Seguridad De Transmisión De Datos En Dispositivos Iot De Cobertura Wlan,” Universidad Politécnica Salesiana, 2023. [Online]. Available: <http://dspace.ups.edu.ec/handle/123456789/24195>
- [43] M. Lombardi, F. Pascale, and D. Santaniello, “Internet of things: A general overview between architectures, protocols and applications,” *Inf.*, vol. 12, no. 2, pp. 1–21, 2021, doi: 10.3390/info12020087.
- [44] T. Harwood, “Internet of Things Protocols.” Postscapes, p. 5, 2020. [Online]. Available: <https://www.postscapes.com/internet-of-things-protocols/>
- [45] Q. I. Sarhan, “Internet of things: a survey of challenges and issues,” *Int. J. Internet Things Cyber-Assurance*, vol. 1, no. 1, p. 40, 2019, doi: 10.1504/ijitca.2018.10011246.
- [46] V. Bonilla, B. Campoverde, and S. G. Yoo, “A Systematic Literature Review of LoRaWAN: Sensors and Applications,” *Sensors*, vol. 23, no. 20, 2023, doi: 10.3390/s23208440.
- [47] D. H. Kim, E. K. Lee, and J. Kim, “Experiencing LoRa network establishment on a smart energy campus testbed,” *Sustain.*, vol. 11, no. 7, 2020, doi: 10.3390/su11071917.
- [48] M. Ragnoli, G. Barile, A. Leoni, G. Ferri, and V. Stornelli, “An autonomous low-power lora-based flood-monitoring system,” *J. Low Power Electron. Appl.*, vol. 10, no. 2, 2020, doi: 10.3390/jlpea10020015.
- [49] T. Bouguera, J. F. Diouris, J. J. Chaillout, R. Jaouadi, and G. Andrieux, “Energy

- consumption model for sensor nodes based on LoRa and LoRaWAN,” *Sensors (Switzerland)*, vol. 18, no. 7, pp. 1–23, 2018, doi: 10.3390/s18072104.
- [50] J. Bravo-Arrabal, J. J. Fernandez-Lozano, J. Serón, J. A. Gomez-Ruiz, and A. García-Cerezo, “Development and implementation of a hybrid wireless sensor network of low power and long range for urban environments,” *Sensors (Switzerland)*, vol. 21, no. 2, pp. 1–27, 2021, doi: 10.3390/s21020567.
- [51] R. Sanchez-Iborra *et al.*, “Enhancing LoRaWAN security through a lightweight and authenticated key management approach,” *Sensors (Switzerland)*, vol. 18, no. 6, pp. 1–18, 2018, doi: 10.3390/s18061833.
- [52] J. Lopez Vicario, “Red sensores multiservicio LPWAN, LoRa, LoRaWAN, TTN y MQTT,” Universitat Oberta de Catalunya, 2023. [Online]. Available: <http://hdl.handle.net/10609/147334>
- [53] J. Trincado Castán, “The Things Network Integration Plugin,” *Thinger.io*, 2025. <https://things.io/the-things-network-integration-plugin/> (accessed Apr. 01, 2025).
- [54] O. R. Marcelo Chuqui and D. M. Castro Castro, “Despliegue y Configuración de una Red LoRaWAN usando una Plataforma The Things Network (TTN) para Dispositivos de Internet de las Cosas (IoT),” Universidad Politécnica Salesiana, 2022. [Online]. Available: <http://dspace.ups.edu.ec/handle/123456789/23944>
- [55] C. A. Zambrano Macías and M. Á. Ortiz Villegas, “Diseño e Implementación de Módulo Didáctico IoT Aplicado a Telemedicina con Hardware de Bajo Costo y Servicios Web en la Nube,” Universidad Politécnica Salesiana, 2021. [Online]. Available: <http://dspace.ups.edu.ec/handle/123456789/22270>
- [56] M. Masiello, “Dashboarding per il monitoraggio di sistemi IoT = Dashboarding for IoT,” Politecnico di Torino, 2024. [Online]. Available: <https://webthesis.biblio.polito.it/31742/>
- [57] S. Konoplev and O. Milind, “Query and visualize Amazon Redshift operational metrics using the Amazon Redshift plugin for Grafana,” *AWS Big Data Blog*, 2022. <https://aws.amazon.com/es/blogs/big-data/query-and-visualize-amazon-redshift-operational-metrics-using-the-amazon-redshift-plugin-for-grafana/> (accessed Apr. 10, 2025).
- [58] I. Hindi, M. Alyaman, A. AboZenah, A. Zaid, and M. Shrara, “Smart Alarm IoT System: Monitoring Elevator Traffic and Meteorological Data on Job Sites Using MQTT and InfluxDB integrated with Grafana,” in *2024 15th International Conference on Information and Communication Systems (ICICS)*, 2024, pp. 1–6. doi: 10.1109/ICICS63486.2024.10638309.

## ANEXOS

**Anexo 1.** En esta sección se presentan los resultados obtenidos durante la implementación y operación del sistema de monitoreo ambiental en tiempo real, basado en tecnologías IoT y comunicación LoRaWAN.



**Ilustración 53.** Implementación y calibración de los sensores e integración de LoRaWAN



**Ilustración 54.** Implementación del sistema IoT/LoRaWAN en la universidad UNACH



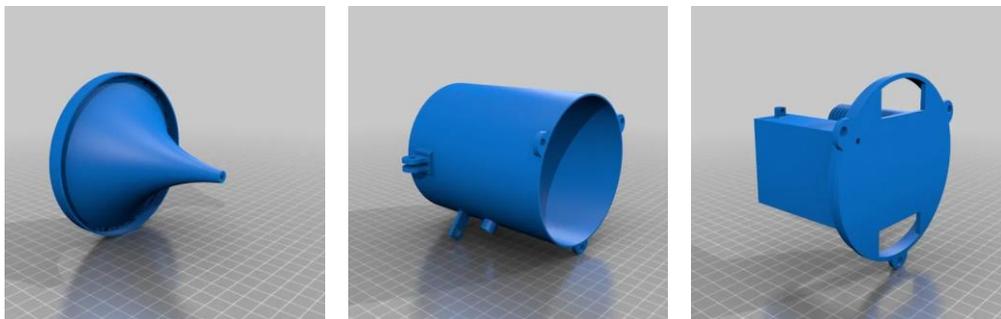
**Ilustración 55.** Sistema IoT/LoRaWAN vs Estación de la ESPOCH

**Anexo 2.** A continuación, se presentan los diseños y ensamblajes en 3D desarrollados para los sensores de precipitación, temperatura/humedad y radiación UV del sistema de monitoreo ambiental IoT. Estos diseños, fueron modelados con el objetivo de garantizar la protección de los sensores frente a las condiciones ambientales, como lluvia, radiación solar intensa y variaciones térmicas. De esta forma se garantiza el correcto funcionamiento y durabilidad de cada sensor.

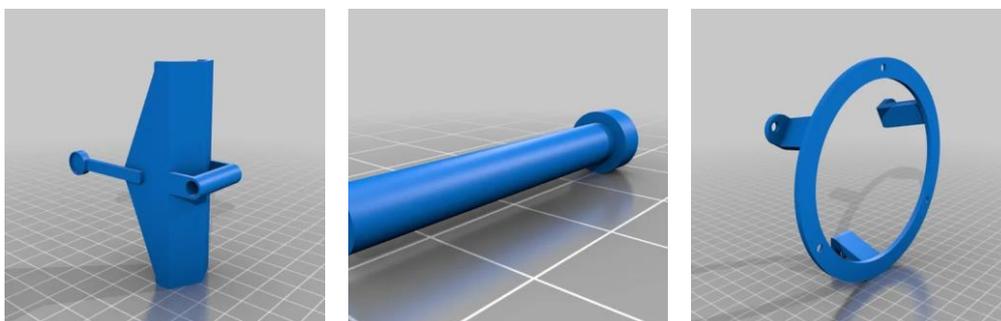
- Pluviómetro



**Ilustración 56.** Diseño y Ensamblaje 3D del sistema de precipitación

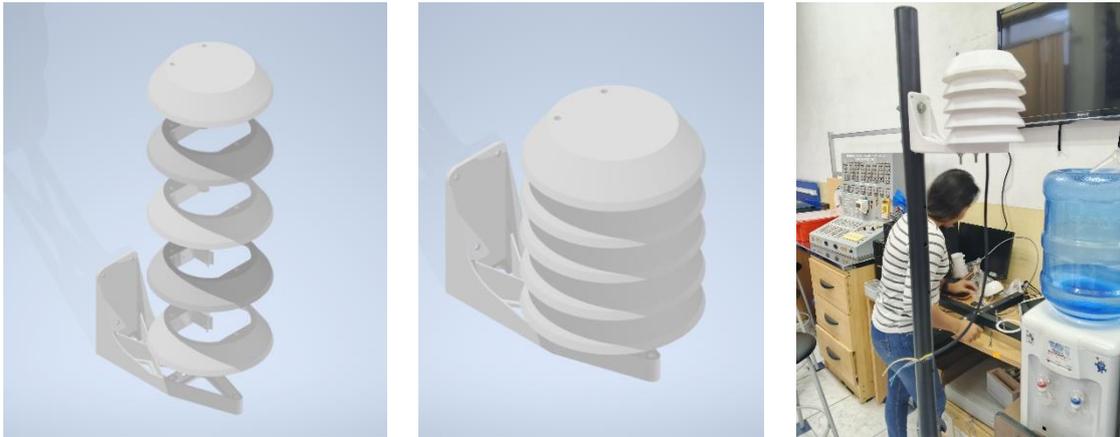


**Ilustración 57.** Piezas 3D (1, 2, 3) para el ensamblaje del sistema de precipitación

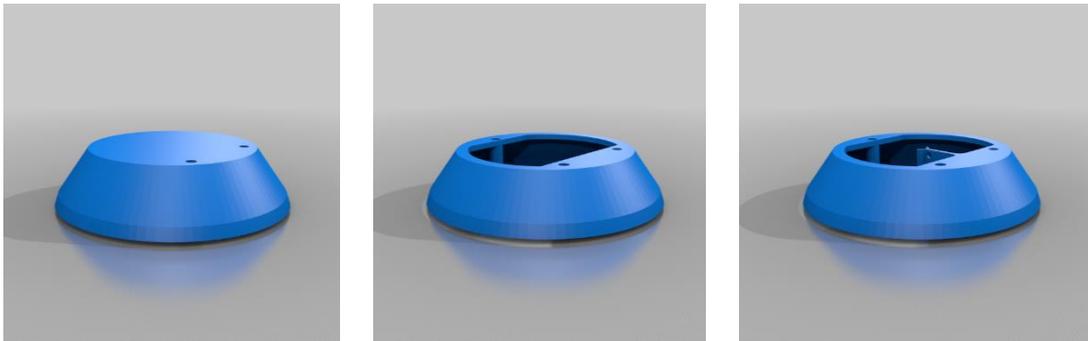


**Ilustración 58.** Piezas 3D (4, 5, 6) para el ensamblaje del sistema de precipitación

- Temperatura / Humedad Relativa



**Ilustración 59.** Diseño y Ensamblaje 3D del sistema de temperatura y humedad



**Ilustración 60.** Piezas 3D (1, 2, 3) para el ensamblaje del sistema de Temperatura/Humedad



**Ilustración 61.** Piezas 3D (4, 5) para el ensamblaje del sistema de Temperatura/Humedad

- Radiación UV



**Ilustración 62.** Diseño y Ensamblaje 3D del sistema de Radiación UV

**Anexo 3.** Contiene el código correspondiente a la implementación de cada sensor, incluyendo su configuración, adquisición de datos y procesamiento básico. Además, se incorpora la funcionalidad para el envío de la información recopilada hacia la red The Things Network (TTN).

```

import utime, time
import machine
import ssd1306
from ulora import TTN, uLoRa
from machine import Pin, I2C, ADC,
UART
from micropython_sht20 import sht20
import BME280

#--CONFIGURACIÓN GENERAL--
ALTITUD_RIOBAMBA = 2755
PRESSURE_OFFSET = 18.89

#--DECLARACIÓN DE PINES - OLED--
i2c = I2C(1, sda=Pin(21),
scl=Pin(22), freq=10000)
oled = ssd1306.SSD1306_I2C(128, 64,
i2c, addr=0x3C)

#-- PINES - SENSOR SHT20 Y BME280
sht = sht20.SHT20(i2c)
bme = BME280.BME280(i2c=i2c)

#--DECLARACIÓN DE PINES - GPS--
gps = UART(1, baudrate=9600, rx=34,
tx=12)

#--PINES - SENSOR UV --
UVOUT = ADC(Pin(36))
REF_3V3 = ADC(Pin(39))
UVOUT.atten(ADC.ATTN_11DB)
REF_3V3.atten(ADC.ATTN_11DB)

#--PINES - DIRECCIÓN DEL VIENTO--
WIND_DIR_PIN = 35
R_BRIDGE_RATIO = 0.6875
adc_wind_dir =
ADC(Pin(WIND_DIR_PIN))
adc_wind_dir.atten(ADC.ATTN_11DB)

#--PINES - PLUVIÓMETRO--
RAIN_PIN = 14
rain_sensor = Pin(RAIN_PIN, Pin.IN)
rain_mm_per_pulse = 3.5
rain_total = 0
rain_last_state = 0

#--FILTRADO DE DIR. DEL VIENTO--
class WindDirectionFilter:
    def __init__(self):
        self.buffer = []
        self.history_size = 13
        self.last_stable = None

DIRECTION_RANGES = [
    (0.000, 0.500, "Norte", 0,
"N"),
    (0.500, 1.100, "Noreste", 45,
"NE"),
    (1.100, 1.700, "Este", 90,
"E"),
    (1.700, 2.300, "Sureste", 135,
"SE"),
    (2.300, 3.000, "Sur", 180,
"S"),
    (3.000, 3.600, "Suroeste", 225,
"SW"),
    (3.600, 4.300, "Oeste", 270,
"W"),
    (4.300, 5.000, "Noroeste", 315,
"NW"),
]

#--FUNCIONES DE LECTURA --
def average_read(sensor, num=5):
    return sum(sensor.read() for _
in range(num)) // num

#--FUNCIONES TEMPERATURA Y HUMEDAD
def read_temperature_humidity():
    return sht.temperature,
sht.humidity

#--FUNCIONES PRESIÓN ATMOSFÉRICA
def read_pressure():
    pres_abs =
float(bme.pressure.replace("hPa",
"")) + PRESSURE_OFFSET
    presion_nivel_mar = pres_abs *
(1 + (ALTITUD_RIOBAMBA / 44330)) **
5.255
    return pres_abs,
presion_nivel_mar

#--FUNCIONES RADIACIÓN UV
def read_uv_intensity():
    uv_level = average_read(UVOUT)
    ref_level =
average_read(REF_3V3)
    output_voltage = (3.3 /
ref_level) * uv_level
    uv_intensity = max(0,
(output_voltage - 0.8) * (15.0 /
2.1))
    return uv_intensity

#--FUNCIONES DIRECCIÓN DEL VIENTO
def get_wind_direction():

```

```

    raw = adc_wind_dir.read()
    voltage = (raw / 4095) * 3.3 /
R_BRIDGE_RATIO
    filtered_v =
wind_filter.add_reading(voltage)

#--FUNCIONES VELOCIDAD DEL VIENTO
def read_wind_speed():
    wind_raw =
average_read(adc_wind_dir)
    wind_speed = (3.3 * wind_raw /
4095 / R_BRIDGE_RATIO) * 6 * 3.6
    return wind_speed

#--FUNCIONES PRESIPITACION
def read_rain():
    global rain_total,
rain_last_state
    current_state =
rain_sensor.value()
    if current_state !=
rain_last_state:
        rain_total +=
rain_mm_per_pulse
    rain_last_state = current_state
    return rain_total

#--FUNCIONES GEOLOCALIZACIÓN
def parse_gps_data():
    try:
        if gps.any():
            line =
gps.readline().decode('utf-
8').strip()

            if
line.startswith('$GNGGA') and
line.split(',')[6] != '0':
                parts =
line.split(',')
                lat =
f"{float(parts[2][:2]) +
float(parts[2][2:])/60:.6f}"
                lon =
f"{float(parts[4][:3]) +
float(parts[4][3:])/60:.6f}"
                return
f"{lat},{lon}"
            except:
                pass
            return None

#--CONFIGURACIÓN LoRa --
LORA_CS = 20
LORA_SCK = 1
LORA_MOSI = 5
LORA_MISO = 9
LORA_IRQ = 2
LORA_RST = 7

TTN_ABP_CONFIG = TTN(
    bytearray([0x26, 0x0C, ...,
0x9F]),
    bytearray([0x24, 0x2C, ...,
0x8E]),
    bytearray([0x70, 0xD3, ...,
0x39]),
    country="US"
)

lora = uLoRa(
    cs=LORA_CS,
    sck=LORA_SCK,
    mosi=LORA_MOSI,
    miso=LORA_MISO,
    irq=LORA_IRQ,
    rst=LORA_RST,
    ttn_config=TTN_ABP_CONFIG,
    datarate="SF7BW125",
    fport=1
)

#--FUNCIONES OLED --
def
mostrar_medicion(gps_data=None):
    oled.fill(0)
    y_pos = 0

    for i, msg in
enumerate([message, message1,
message2, message3, message4,
message5]):
        oled.text(msg, 0, y_pos)
        y_pos += 10 if i < 4 else
12

    if gps_data:
        oled.text("GPS:", 0, 50)
        oled.text(gps_data[:15],
30, 50)
        if len(gps_data) > 15:
oled.text(gps_data[15:], 0, 60)
        else:
            oled.text("GPS: Sin señal",
0, 50)

    oled.show()

#--BUCLE PRINCIPAL --
frame_counter = 0
while True:
    gps_data = parse_gps_data()
    temp, hum =
read_temperature_humidity()
    pres_abs, _ = read_pressure()
    uv = read_uv_intensity()
    wind_speed = read_wind_speed()
    wind_dir_name, wind_deg,
wind_abbrev, wind_voltage =
get_wind_direction()
    rain_mm = read_rain()

```

```

    #-- Mensajes OLED --
    message = f"Temp:{temp:.1f}C
Hum:{hum:.1f}%"
    message1 =
f"Viento:{wind_speed:.1f}km/h"
    message2 = f"Dir:{wind_abbrev}
{wind_deg}°"
    message3 = f"UV:{uv:.1f}mW/cm2"
    message4 =
f"Pres:{pres_abs:.1f}hPa"
    message5 =
f"Lluvia:{rain_mm:.1f}mm"

    #-- Mensaje para LoRa --
    msg =
f"{temp:.1f}@{hum:.1f}@{wind_speed:
.1f}@{wind_deg}@{uv:.1f}@{pres_abs:
.1f}@{gps_data or
'0,0'}@{wind_voltage:.3f}@{rain_mm:
.1f}"

    try:

lora.send_data(msg.encode() ,
len(msg) , frame_counter)
        frame_counter =
(frame_counter + 1) % 256
    except Exception as e:
        print("Error LoRa:", e)

    mostrar_medicion(gps_data)
    utime.sleep(30)

```

**Anexo 4.** Contiene el código fuente del decodificador personalizado en JavaScript utilizado en la consola de TTN para interpretar los payloads enviados por el nodo TTGO LoRa32. Este decodificador procesa los datos recibidos en formato de texto delimitado por el carácter @ y los convierte en variables individuales correspondientes a los sensores integrados

```
function Decoder(bytes, port) {
  var decoded = {};

  try {
    // Convert bytes to string
    var str =
String.fromCharCode.apply(null,
bytes);
    var values = str.split('@');

    // Basic validation
    if (values.length < 7) {
      throw new Error(`Expected at
least 7 values, got
${values.length}`);
    }

    // Main measurements
    decoded.temperature = {
      value: parseFloat(values[0]),
      unit: "°C",
      status:
isNaN(parseFloat(values[0])) ?
"invalid" : "ok"
    };

    decoded.humidity = {
      value: parseFloat(values[1]),
      unit: "%",
      status:
isNaN(parseFloat(values[1])) ?
"invalid" : "ok"
    };

    // Wind data
    var windDeg =
parseFloat(values[3]);
    decoded.wind = {
      speed: {
        value:
parseFloat(values[2]),
        unit: "km/h",
        status:
isNaN(parseFloat(values[2])) ?
"invalid" : "ok"
      },
      direction:
getWindDirection(windDeg),
      raw_voltage: values.length >
7 ? parseFloat(values[7]) : null
    };

    // UV radiation
    decoded.uv = {
      intensity:
parseFloat(values[4]),
      unit: "mW/cm²",
      risk_level:
getUVRiskLevel(parseFloat(values[4]
)),
      status:
isNaN(parseFloat(values[4])) ?
"invalid" : "ok"
    };

    // Pressure
    decoded.pressure = {
      absolute:
parseFloat(values[5]),
      unit: "hPa",
      status:
isNaN(parseFloat(values[5])) ?
"invalid" : "ok"
    };

    // GPS data
    decoded.gps =
parseGPS(values[6]);

    // System info
    decoded.metadata = {
      payload_format:
"lilygo_weather_v1",
      received_at: new
Date().toISOString(),
      data_quality:
calculateDataQuality(values)
    };
  }
  return decoded;
}

// Helper function: Convert wind
degrees to direction
function getWindDirection(degrees)
{
  if (isNaN(degrees)) return null;

  const directions = [
    { min: 0, max: 22.5, abbr: 'N',
name: 'North', es: 'Norte' },
    { min: 22.5, max: 67.5, abbr:
'NE', name: 'Northeast', es:
'Noreste' },
    { min: 67.5, max: 112.5, abbr:
'E', name: 'East', es: 'Este' },
  ]
}
```

```

    { min: 112.5, max: 157.5, abbr: 'SE', name: 'Southeast', es: 'Sureste' },
    { min: 157.5, max: 202.5, abbr: 'S', name: 'South', es: 'Sur' },
    { min: 202.5, max: 247.5, abbr: 'SW', name: 'Southwest', es: 'Suroeste' },
    { min: 247.5, max: 292.5, abbr: 'W', name: 'West', es: 'Oeste' },
    { min: 292.5, max: 337.5, abbr: 'NW', name: 'Northwest', es: 'Noroeste' },
    { min: 337.5, max: 360, abbr: 'N', name: 'North', es: 'Norte' }
  ];

  const normalizedDeg = (degrees % 360 + 360) % 360;
  const dir = directions.find(d => normalizedDeg >= d.min && normalizedDeg < d.max);

  // Helper function: Parse GPS coordinates
  function parseGPS(gpsStr) {
    if (!gpsStr || gpsStr === '0,0')
      return { status: "no_fix" };

    const parts = gpsStr.split(',');
    if (parts.length !== 2) return { status: "invalid_format" };

    const lat = parseFloat(parts[0]);
    const lon = parseFloat(parts[1]);

    if (isNaN(lat) || isNaN(lon))
      return { status: "invalid_values" };

    return {
      latitude: lat,
      longitude: lon,
      status: "valid",
      google_maps_url: `https://maps.google.com/?q=${lat},${lon}`,
      precision: {
        estimate: "±11m",
        decimals: 6,
        description: "6 decimal places (~11cm precision)"
      }
    };
  }

  // Helper function: UV risk level
  function getUVRiskLevel(uvIndex) {
    if (isNaN(uvIndex)) return

```

```

invalid"; if (uvIndex < 3) return "low";
    if (uvIndex < 6) return "moderate";
    if (uvIndex < 8) return "high";
    if (uvIndex < 11) return "very_high";
    return "extreme";
  }

  return {
    score: qualityScore,
    missing_fields: expectedValues
  - presentValues,
    completeness:
    `${presentValues}/${expectedValues}`
  };

```

