

Ciencia Latina Revista Científica Multidisciplinar, Ciudad de México, México.
ISSN 2707-2207 / ISSN 2707-2215 (en línea), julio-agosto 2025,
Volumen 9, Número 4.

https://doi.org/10.37811/cl_rcm.v9i2

SISTEMA IOT LORAWAN-CHIRPSTACK- GRAFANA PARA MONITOREO REMOTO DE POZOS PETROLEROS: ESTUDIO DE CASO EN TABASCO, MÉXICO

**IOT LORAWAN-CHIRPSTACK-GRAFANA SYSTEM
FOR REMOTE MONITORING OF OIL WELLS: A CASE
STUDY IN TABASCO, MEXICO**

Jesús Alayón Domínguez

Universidad Juárez Autónoma de Tabasco, México

Arturo Corona Ferreira

Universidad Juárez Autónoma de Tabasco, México

Carlos Arturo Custodio Izquierdo

Universidad Juárez Autónoma de Tabasco, México

Carlos González Zacarías

Universidad Juárez Autónoma de Tabasco, México

DOI: https://doi.org/10.37811/cl_rem.v9i4.19526

Sistema IoT LoRaWAN-ChirpStack-Grafana para Monitoreo Remoto de Pozos Petroleros: Estudio de Caso en Tabasco, México

Jesús Alayón Domínguez¹202h17011@egresado.ujat.mx<https://orcid.org/0009-0003-3075-3070>Universidad Juárez Autónoma de Tabasco
México**Arturo Corona Ferreira**arturo.corona@ujat.mx<https://orcid.org/0000-0001-8245-0838>Universidad Juárez Autónoma de Tabasco
México**Carlos Arturo Custodio Izquierdo**carlos.custodio@ujat.mx<https://orcid.org/0000-0002-8639-3858>Universidad Juárez Autónoma de Tabasco
México**Carlos González Zacarías**Carlos.gonzalez@ujat.mx<https://orcid.org/0000-0001-5307-3332>Universidad Juárez Autónoma de Tabasco
México

RESUMEN

El monitoreo remoto de pozos petroleros en regiones tropicales presenta desafíos significativos debido a condiciones ambientales extremas, aislamiento geográfico y limitaciones económicas de los enfoques de monitoreo convencionales. Este estudio presenta el desarrollo y validación de un prototipo tecnológico que integra la tecnología LoRaWAN con infraestructura en la nube mediante orquestación de microservicios para el monitoreo remoto de pozos petroleros en condiciones tropicales del sureste de México. Se empleó una metodología de prototipo tecnológico, implementando una arquitectura distribuida utilizando sensores IoT RAKwireless, comunicación LoRaWAN, plataforma ChirpStack, y sistemas de visualización Grafana-InfluxDB desplegados en Amazon Web Services usando contenerización Docker. El prototipo fue validado bajo condiciones operacionales reales caracterizadas por alta humedad relativa (70-90%) y temperaturas elevadas (28-35°C) típicas del trópico húmedo de Tabasco, México. Los resultados demostraron la integración exitosa de todos los componentes tecnológicos con operación estable bajo condiciones tropicales, alcanzando rangos de comunicación LoRaWAN superiores a 15 kilómetros en áreas rurales mientras se mantiene la eficiencia energética apropiada para dispositivos alimentados por batería. El prototipo implementó exitosamente siete microservicios contenerizados con visualización de datos en tiempo real mediante dashboards Grafana. El sistema demostró monitoreo confiable de presión, temperatura, humedad y concentraciones de gas. Este estudio contribuye conocimiento práctico para la integración de tecnología IoT en ambientes industriales tropicales.

Palabras clave: LoRaWAN, IoT industrial, monitoreo remoto

¹ Autor principal

Correspondencia: 202h17011@egresado.ujat.mx

IoT LoRaWAN–ChirpStack–Grafana System for Remote Monitoring of Oil Wells: A Case Study in Tabasco, Mexico

ABSTRACT

Remote monitoring of oil wells in tropical regions presents significant challenges due to extreme environmental conditions, geographical isolation, and economic limitations of conventional monitoring approaches. This study presents the development and validation of a technological prototype that integrates LoRaWAN technology with cloud infrastructure through microservices orchestration for remote monitoring of oil wells in tropical conditions of southeastern Mexico. A technological prototype methodology was employed, implementing a distributed architecture using RAKwireless IoT sensors, LoRaWAN communication, ChirpStack platform, and Grafana-InfluxDB visualization systems deployed on Amazon Web Services using Docker containerization. The prototype was validated under real operational conditions characterized by high relative humidity (70-90%) and elevated temperatures (28-35°C) typical of the humid tropics of Tabasco, Mexico. Results demonstrated successful integration of all technological components with stable operation under tropical conditions, achieving LoRaWAN communication ranges exceeding 15 kilometers in rural areas while maintaining appropriate energy efficiency for battery-powered devices. The prototype successfully implemented seven containerized microservices with real-time data visualization through Grafana dashboards. The system demonstrated reliable monitoring of pressure, temperature, humidity, and gas concentrations. This study contributes practical knowledge for IoT technology integration in tropical industrial environments.

Keywords: LoRaWAN, industrial iot, remote monitoring

Artículo recibido 04 Agosto 2025

Aceptado para publicación: 29 Agosto 2025



INTRODUCCIÓN

El presente artículo se enfoca en el desarrollo y validación de un prototipo tecnológico que integra la tecnología LoRaWAN con infraestructura en la nube para el monitoreo remoto de pozos petroleros en condiciones tropicales del sureste de México. El desafío de la investigación radica en las limitaciones de los sistemas de monitoreo convencionales para operar eficientemente bajo condiciones ambientales extremas, aislamiento geográfico y restricciones económicas características de las operaciones petroleras en regiones remotas. Este vacío en el conocimiento impide a las operaciones de pequeña y mediana escala acceder a tecnologías avanzadas de monitoreo que optimicen la supervisión de variables críticas como presión, temperatura, humedad y concentraciones de gas de manera eficaz y económicamente viable.

La orquestación de servicios IoT mediante contenedores representa un paradigma emergente para la escalabilidad y eficiencia de sistemas de monitoreo industrial. Khalilnasl (2025) evaluó el rendimiento de la virtualización de nodos LoRaWAN utilizando contenedores, demostrando que las tecnologías de contenerización como Docker pueden optimizar significativamente el uso de recursos computacionales mientras mantienen la flexibilidad operacional. El estudio reveló mejoras sustanciales en la eficiencia de recursos comparado con implementaciones tradicionales, validando la arquitectura de microservicios para sistemas de monitoreo distribuido. Esta aproximación permite la gestión independiente y escalable de servicios como ChirpStack, bases de datos temporales e interfaces de visualización, facilitando el mantenimiento y la evolución del sistema sin afectar la operación global.

El Internet Industrial de las Cosas (IIoT) ha transformado las prácticas de mantenimiento y monitoreo en la industria del petróleo y gas. Aragonés (2025) desarrolló una arquitectura IIoT sin batería alimentada por calor con conectividad NB-IoT para mantenimiento predictivo en la industria petrolera y gasífera. Esta investigación demostró que los sistemas autoalimentados pueden operar de forma continua en entornos industriales hostiles, eliminando la necesidad de mantenimiento de baterías y reduciendo significativamente los costos operacionales. La implementación de sensores IoT en pozos petroleros presenta desafíos únicos relacionados con las condiciones ambientales extremas y los requisitos de confiabilidad crítica, donde los sistemas de monitoreo deben mantener precisión en la



medición de parámetros como presión, temperatura, humedad y concentraciones de gas bajo condiciones de alta variabilidad ambiental.

La evaluación de rendimiento en redes IoT industriales requiere consideración de múltiples métricas incluyendo latencia, rendimiento y confiabilidad. Yoon (2025) propuso un sistema de recolección de datos de doble modo para transmisión periódica y urgente en redes de sensores inalámbricos con recolección de energía, logrando optimización significativa en la gestión de transmisiones críticas mientras mantiene la eficiencia energética para operaciones de rutina. Complementariamente, Alghamdi (2025) desarrolló algoritmos adaptativos para estimación de canal en tiempo real y ajuste de parámetros en redes LoRa para entornos IoT dinámicos. La investigación demostró mejoras sustanciales en la calidad de servicio (QoS) bajo condiciones de canal variables, alcanzando niveles de confiabilidad apropiados para entornos industriales complejos donde la variabilidad de las condiciones de transmisión puede afectar la integridad de los datos.

Los sistemas de visualización de datos representan un componente crítico para la operación efectiva de sistemas de monitoreo industrial. Bonavolontà (2025) implementó un sistema de medición IoT para monitoreo en tiempo real de contribuciones energéticas en comunidades de energía renovable, utilizando dashboards interactivos que permitieron la toma de decisiones operacionales inmediatas. El sistema integró sensores distribuidos con plataformas de visualización, logrando actualización de datos en intervalos reducidos para facilitar el monitoreo continuo. La gestión de datos temporales constituye un desafío fundamental en aplicaciones de monitoreo continuo, requiriendo capacidades de almacenamiento escalable y consultas eficientes para el análisis de tendencias históricas y detección de anomalías en tiempo real. Zeta (2025) desarrolló un sistema de monitoreo de pH de bajo costo para acuicultura en una red de sensores inalámbricos multicapa, demostrando que las soluciones IoT pueden mantener precisión de medición mientras proporcionan visualización en tiempo real de parámetros críticos en aplicaciones industriales.

La forma en que se comunican los dispositivos IoT es mediante el protocolo MQTT (Message Queuing Telemetry Transport) el cual es un estándar fundamental para comunicación en redes de ancho de banda limitado. Mota (2025) implementó una arquitectura IoT para monitoreo de calidad del aire interior utilizando MQTT como protocolo principal de comunicación, demostrando alta confiabilidad en



transmisión de datos críticos durante períodos extendidos de operación. La arquitectura publish-subscribe del protocolo MQTT facilita la integración de múltiples sensores distribuidos con sistemas centralizados de gestión de datos, permitiendo el escalamiento horizontal de las redes de monitoreo sin comprometer la estabilidad del sistema. La implementación de brokers MQTT redundantes asegura la continuidad operacional en aplicaciones industriales críticas donde la pérdida de conectividad puede resultar en consecuencias operacionales significativas.

La tecnología LoRaWAN (Long Range Wide Area Network) se ha consolidado como una solución fundamental para aplicaciones IoT industriales debido a su capacidad de transmisión de largo alcance y bajo consumo energético. Sánchez (2025) implementó exitosamente una estación de monitoreo remoto utilizando tecnología LoRaWAN para medición de niveles de radiación UV, demostrando la viabilidad de esta tecnología para aplicaciones de monitoreo ambiental en tiempo real. La investigación evidenció que LoRaWAN puede alcanzar rangos de transmisión superiores a 15 kilómetros en áreas rurales, manteniendo una eficiencia energética óptima para dispositivos alimentados por baterías, lo cual resulta especialmente relevante para aplicaciones de monitoreo petrolero en ubicaciones remotas. Los estudios identificados se enfocan en componentes individuales sin abordar la integración holística de sistemas complejos para aplicaciones industriales específicas, dejando vacíos significativos de reportes que validen soluciones IoT completas en condiciones operacionales reales de la industria petrolera, incluyendo factores ambientales del trópico húmedo del sur de México. De igual manera existen escasos estudios que abordan el comportamiento de sistemas IoT bajo cargas operacionales reales con múltiples dispositivos distribuidos geográficamente, y las investigaciones actuales no abordan suficientemente los desafíos de integración con infraestructura industrial existente y estándares de comunicación establecidos. Esta brecha representa una oportunidad significativa que busca abordar el presente trabajo de investigación, particularmente relevante para futuras implementaciones que combinen las tecnologías LoRaWAN, infraestructura en la nube y sistemas de visualización para el monitoreo petrolero, validadas bajo condiciones reales de operación en ambientes tropicales.

El uso de sistemas IoT en pozos petroleros para su monitoreo es fundamental debido a las fallas que pueden generarse por las variaciones en la presión, temperatura, concentraciones de gas y humedad.



Cualquier irregularidad en estos indicadores críticos podría ocasionar fallas operacionales significativas. El monitoreo actual requiere que los operadores revisen estos indicadores de manera presencial, registrando datos de cada zona manualmente. Esta metodología presenta limitaciones, ya que pueden ocurrir eventos críticos en un punto mientras el operador inspecciona otra área, resultando en tiempos de respuesta inadecuados para la detección temprana de anomalías.

La red LoRaIN significa una alternativa eficiente para el monitoreo, esta consiste en sensores, gateways y un servidor que recolecta la información para los interesados, diseñado con el fin de utilizar poca energía y conectar inalámbricamente dispositivos en redes regionales, nacionales o globales, opera en bandas de radiofrecuencia sin licencia y puede alcanzar hasta un alcance de 15 kilómetros en áreas rurales (Bankov, Khorov, Lyakhov, 2016).

Objetivo general y Objetivos específicos

El monitoreo de pozos petroleros en regiones tropicales presenta desafíos únicos derivados de las condiciones ambientales extremas, la ubicación remota de las instalaciones, y las limitaciones económicas de los sistemas de supervisión tradicionales. Las altas temperaturas (28-35°C), humedad relativa (70-90%), y precipitaciones intensas del sureste mexicano generan condiciones adversas para equipos electrónicos convencionales, mientras que los costos de implementación y mantenimiento de sistemas SCADA tradicionales resultan prohibitivos para operaciones de pequeña y mediana escala.

La integración de tecnologías IoT basadas en LoRaWAN con infraestructura cloud representa una oportunidad para democratizar el acceso a sistemas de monitoreo avanzados, reduciendo significativamente los costos de implementación mientras se mejora la capacidad de supervisión remota, por lo cual esta iniciativa tiene por objetivo Desarrollar una arquitectura de microservicios distribuidos que integre sensores IoT RAKwireless, comunicación LoRaWAN, plataforma ChirpStack, y sistemas de visualización Grafana-InfluxDB, optimizada para las condiciones ambientales y operacionales del monitoreo petrolero tropical.

El alcance que se espera con este prototipo es demostrar la viabilidad técnica de la integración tecnológica propuesta, con el fin de lograr una validación práctica de la arquitectura en condiciones reales de campo, la evaluación de rendimiento bajo condiciones tropicales específicas, y la generación de conocimiento transferible para implementaciones similares.



El presente prototipo tecnológico busca realizar una contribución multidimensional al conocimiento científico y técnico mediante la validación práctica de la integración LoRaWAN-cloud bajo las condiciones tropicales extremas del sureste mexicano, caracterizadas por alta humedad relativa y temperaturas elevadas que no han sido suficientemente documentadas en la literatura especializada. El prototipo genera datos empíricos sobre el rendimiento de sistemas IoT petroleros en entornos operacionales reales, desarrolla un framework metodológico replicable que facilita la transferencia de conocimiento a proyectos similares, e identificar mejores prácticas específicas para el despliegue de tecnologías IoT en ambientes adversos. Adicionalmente, proporciona un análisis económico integral que compara los costos del sistema propuesto con alternativas tradicionales, generando información cuantitativa que facilita la toma de decisiones de inversión tecnológica en operaciones petroleras de pequeña y mediana escala.

METODOLOGÍA

La metodología de prototipo tecnológico se define como un proceso sistemático de desarrollo, implementación y validación de soluciones tecnológicas emergentes mediante la construcción de sistemas funcionales que demuestran la viabilidad técnica y operacional de conceptos innovadores (López et al., 2023). Este enfoque metodológico resulta particularmente apropiado para investigaciones que buscan validar la integración de múltiples tecnologías en aplicaciones industriales específicas, donde la contribución principal radica en la demostración práctica de su viabilidad técnica y no en el desarrollo de nuevos marcos teóricos.

Fase I: Análisis y Diseño

La primera fase de la metodología se centró en el análisis detallado de los requisitos técnicos y operacionales del monitoreo petrolero en condiciones tropicales, así como en el diseño de la arquitectura tecnológica del prototipo. Esta fase siguió las recomendaciones establecidas por López et al. (2023) para el desarrollo rápido de prototipos IoT basados en LoRaWAN, adaptando su framework de prototipado visual a las necesidades específicas del monitoreo industrial petrolero.

El proceso de análisis de requisitos incluyó la identificación de parámetros críticos de monitoreo (presión, temperatura, humedad, concentraciones de gas), especificaciones de comunicación remota (alcances superiores a 15 kilómetros en áreas rurales), y restricciones ambientales del entorno tropical



húmedo del sureste mexicano. La selección de tecnologías se basó en criterios de eficiencia energética, confiabilidad de comunicación, y escalabilidad del sistema, considerando las recomendaciones de Sánchez (2025) para implementaciones de estaciones de monitoreo remoto en condiciones ambientales extremas.

El diseño arquitectónico del prototipo se estructuró siguiendo un enfoque de microservicios distribuidos en la nube, aprovechando las ventajas de contenerización documentadas por Khalilnasl (2025). La arquitectura propuesta integra dispositivos finales RAKwireless con conectividad LoRa WAN, gateway de comunicación, plataforma de gestión ChirpStack desplegada en contenedores Docker, y sistemas de almacenamiento y visualización basados en InfluxDB y Grafana respectivamente.

La definición de métricas de evaluación se estableció considerando los estándares de rendimiento documentados en la literatura especializada. Se adoptaron las metodologías de evaluación de latencia propuestas por Sisinni et al. (2023), incluyendo mediciones de tiempo de respuesta end-to-end, calidad de señal (RSSI, SNR), y tasas de entrega de paquetes (PDR). Adicionalmente, se definieron métricas específicas para la evaluación del sistema en condiciones tropicales, incluyendo estabilidad térmica, resistencia a la humedad, y confiabilidad operacional extendida.

Fase II: Desarrollo del Prototipo

La segunda fase se enfocó en la implementación práctica de la infraestructura tecnológica y el desarrollo de los componentes del sistema de monitoreo. Esta etapa comprendió el despliegue de la arquitectura en nube utilizando los recursos de AWS con las tecnologías seleccionadas, lo que incluyó la configuración de dispositivos IoT, la integración de microservicios y el establecimiento de las comunicaciones entre todos los elementos del prototipo.

La implementación del sistema de monitoreo se fundamentó en el despliegue de dispositivos IoT basados en placas de montaje integradas con antenas LoRaWAN y Bluetooth Low Energy (BLE). El núcleo de procesamiento se configuró mediante módulos RAK4631, los cuales proporcionan conectividad dual LoRaWAN/BLE y capacidades de procesamiento distribuido. Para la adquisición de variables ambientales críticas, se incorporó el sensor multivariable RAK19007, diseñado específicamente para el monitoreo continuo de temperatura, presión barométrica, humedad relativa y concentraciones de gases en entornos industriales.



La arquitectura de red se centralizó mediante un gateway WisGate Edge Ultra Full-duplex, seleccionado por sus características de cobertura extendida y operación bidireccional simultánea. Esta configuración garantiza la integridad de las comunicaciones en las condiciones climáticas adversas características del estado de Tabasco, manteniendo la robustez del enlace radioeléctrico incluso en escenarios de alta humedad y variabilidad térmica.

Los requisitos del proyecto establecieron la necesidad de implementar una arquitectura cloud-native basada en Amazon Web Services (AWS) y tecnologías de contenerización Docker. Esta decisión arquitectónica permite la escalabilidad horizontal del sistema y facilita el mantenimiento distribuido de la infraestructura.

Como intermediario de comunicaciones (broker) entre la red LoRaWAN y los servicios en la nube, se seleccionó ChirpStack, plataforma que proporciona las funcionalidades de Network Server y Application Server necesarias para la gestión centralizada de los dispositivos IoT y el procesamiento de los datos telemétricos.

La totalidad de la solución fue diseñada bajo la arquitectura de microservicios en la nube. Este tipo de soluciones permite que el total del diseño sea escalable por la distribución de servicios mediante dos instancias de máquinas virtuales en AWS, estableciendo una separación lógica y funcional de los componentes del sistema. Esta configuración permite optimizar el rendimiento y garantizar la continuidad operacional del sistema de monitoreo.

La primera instancia se dedicó exclusivamente a la gestión de comunicaciones IoT, alojando los servicios de conectividad LoRaWAN y la interfaz de integración con los dispositivos de campo. Esta máquina virtual concentra las funcionalidades de recepción, procesamiento inicial y enrutamiento de datos telemétricos, actuando como puente entre la red de sensores y la infraestructura de análisis.

La segunda instancia se configuró como servidor de visualización y almacenamiento temporal, integrando InfluxDB como base de datos de series temporales y Grafana como plataforma de generación de dashboards interactivos. Esta separación permite el procesamiento en tiempo real de las métricas ambientales y facilita la creación de interfaces de usuario especializadas para el análisis de tendencias y patrones operacionales.



Esta arquitectura distribuida establece un sistema de monitoreo descentralizado que proporciona visibilidad integral tanto del comportamiento de los sensores como del estado operacional de los dispositivos IoT. La configuración permite la detección temprana de anomalías en el funcionamiento de los sensores, identificación de patrones de degradación en las comunicaciones y análisis simultáneo de múltiples variables ambientales, eliminando la dependencia de un único punto de falla y mejorando la resiliencia general del sistema.

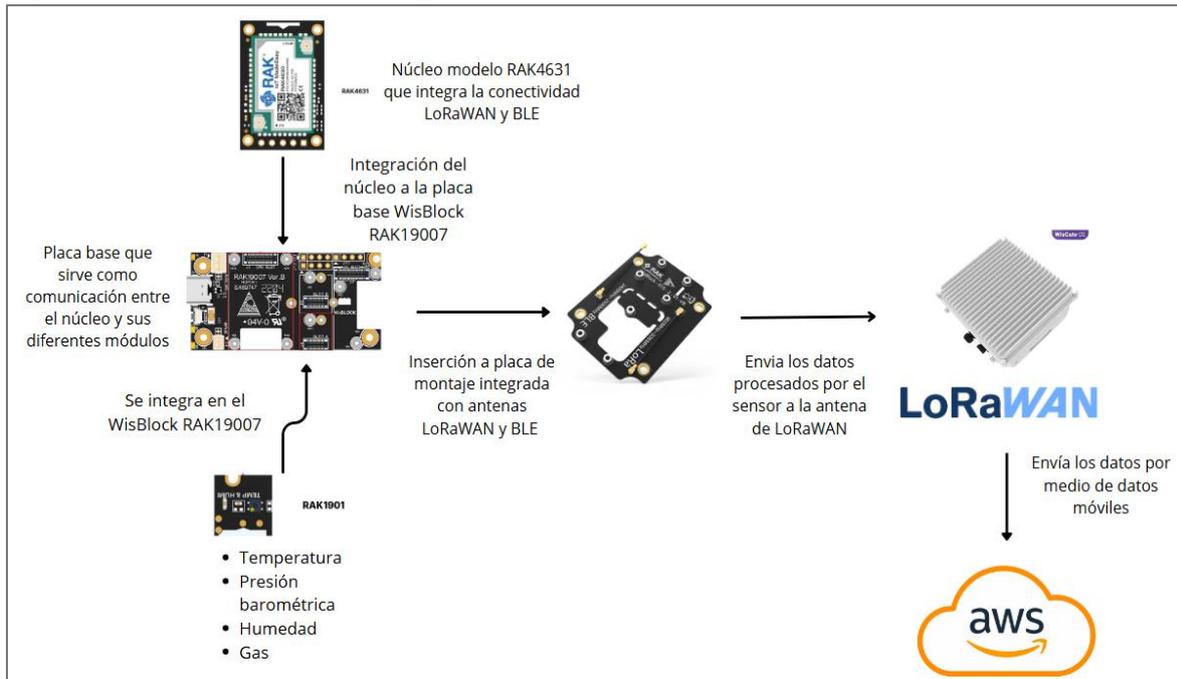
Respecto a la captura de datos del contexto, se consideró emplear un modelado de flujo de datos basado en IoT. IoT permite implementar un esquema de implementación donde los datos de contexto como temperatura, humedad, vibración se capturan mediante dispositivos electrónicos autónomos que transmiten de manera periódica información mediante antenas LoRaWAN, las cuales habitualmente se encuentran integradas en la cápsula del dispositivo, operando bajo intervalos de muestreo configurables. Es fundamental destacar que el alcance efectivo de la red está determinado por las características de radiación y potencia de transmisión de las antenas de los nodos sensores, no por las especificaciones del gateway receptor.

El gateway WisGate Edge Ultra actúa como concentrador de datos, recibiendo las tramas LoRaWAN provenientes de los dispositivos distribuidos en el área de monitoreo. Una vez procesadas y validadas estas señales, el gateway establece la conectividad hacia la infraestructura en la nube mediante un módulo de comunicaciones celulares integrado, utilizando la red de datos móviles como medio de transporte principal. Esta configuración garantiza la independencia de infraestructura de red fija y proporciona flexibilidad operacional en ubicaciones remotas.

Los datos telemétricos, una vez recibidos en la infraestructura AWS, ingresan al pipeline de procesamiento donde son validados, normalizados y distribuidos hacia los servicios de almacenamiento y visualización correspondientes.



Figura 1: Infraestructura de los dispositivos IOT



La segunda fase de la metodología se enfocó en la implementación de la infraestructura cloud necesaria para soportar el prototipo tecnológico. Esta fase siguió las mejores prácticas establecidas por Khalilnasl (2025) para la virtualización de nodos LoRaWAN mediante contenedores, adaptando su framework de orquestación a las necesidades específicas del monitoreo petrolero industrial.

La configuración del entorno cloud se realizó utilizando instancias EC2 de Amazon Web Services, aprovechando las capacidades de escalabilidad y confiabilidad de la plataforma. Se implementaron dos máquinas virtuales dedicadas: una para servicios de comunicación IoT (ChirpStack, brokers MQTT, bases de datos operacionales) y otra para servicios de gestión de datos y visualización (InfluxDB, Grafana). Esta segregación de servicios siguió los principios de arquitectura de microservicios documentados en la literatura, permitiendo escalabilidad independiente y mantenimiento simplificado. La orquestación de microservicios se implementó utilizando Docker como plataforma de contenerización, siguiendo las recomendaciones de Khalilnasl (2025) para optimización de recursos computacionales. Se desarrollaron contenedores específicos para cada servicio: ChirpStack Gateway Bridge para comunicación con dispositivos LoRaWAN, Mosquitto como broker MQTT, PostgreSQL para gestión de metadatos, Redis para almacenamiento temporal, e InfluxDB para series temporales.

La configuración de contenedores se optimizó para maximizar la eficiencia de recursos mientras se mantiene la flexibilidad operacional requerida para aplicaciones industriales.

La implementación de la plataforma ChirpStack se realizó considerando las especificaciones técnicas de la red LoRaWAN y los requisitos de integración con sistemas de visualización. Se desarrollaron perfiles de dispositivos específicos para sensores RAKwireless, incluyendo decodificadores JavaScript personalizados para el procesamiento de datos de sensores industriales. La configuración de aplicaciones ChirpStack se estructuró para facilitar la gestión de múltiples dispositivos distribuidos geográficamente, implementando estrategias de agrupación y categorización apropiadas para entornos petroleros.

La activación y parametrización de sensores IoT se realizó siguiendo protocolos específicos para entornos industriales petroleros. Se configuraron sensores para medición de presión barométrica, temperatura ambiente, humedad relativa, y concentraciones de gases característicos de operaciones petroleras. La calibración de sensores se ejecutó utilizando estándares industriales, estableciendo rangos de operación apropiados para las condiciones ambientales del sureste mexicano y los requisitos operacionales de monitoreo petrolero.

La integración de comunicación MQTT se implementó siguiendo las recomendaciones de Mota et al. (2025) para arquitecturas IoT de monitoreo ambiental. Se configuró el broker Mosquitto para gestionar la comunicación bidireccional entre dispositivos finales y la plataforma de gestión, implementando estrategias de calidad de servicio (QoS) apropiadas para aplicaciones industriales críticas. La arquitectura publish-subscribe se optimizó para soportar múltiples dispositivos distribuidos mientras se mantiene la latencia de comunicación dentro de rangos aceptables para monitoreo en tiempo real.

La integración ChirpStack-InfluxDB se configuró para asegurar la transferencia confiable de datos desde la plataforma IoT hacia el sistema de almacenamiento temporal. Se implementaron mecanismos de validación de datos y estrategias de recuperación ante fallos, garantizando la integridad y disponibilidad de información crítica para operaciones petroleras. La sincronización de datos se optimizó para minimizar la latencia entre la captura de sensores y la disponibilidad en dashboards de visualización.



La configuración de InfluxDB se optimizó para el manejo eficiente de series temporales generadas por sensores IoT distribuidos. Se implementaron estrategias de retención de datos apropiadas para aplicaciones industriales, balanceando los requisitos de almacenamiento histórico con las limitaciones de recursos computacionales. La estructura de base de datos se diseñó para facilitar consultas analíticas complejas, incluyendo agregaciones temporales, detección de anomalías, y análisis de tendencias históricas.

La implementación del proyecto resultó en el desarrollo exitoso de una arquitectura descentralizada en la nube para la gestión integral de dispositivos IoT en aplicaciones de monitoreo ambiental. El sistema logró establecer comunicaciones bidireccionales en tiempo real entre los nodos sensores distribuidos y la infraestructura de procesamiento en AWS, garantizando la continuidad operacional y la integridad de los datos telemétricos.

La plataforma desarrollada permite el monitoreo continuo y simultáneo de múltiples variables ambientales críticas, incluyendo temperatura, humedad relativa, concentraciones de gases y presión barométrica. Los dashboards implementados en Grafana proporcionan interfaces interactivas para la visualización de tendencias temporales para el análisis de correlaciones entre variables.

Las pruebas de campo demostraron la viabilidad técnica de la tecnología LoRaWAN para aplicaciones de monitoreo en entornos industriales. Se logró establecer comunicaciones estables en un radio de cobertura de 9.8 km alrededor del gateway central, valor que representa un parámetro de referencia significativo para el despliegue de redes IoT en regiones con condiciones climáticas de alta humedad, como las características del estado de Tabasco.

Este alcance operacional confirma la efectividad de la tecnología LoRaWAN para superar las limitaciones de propagación radioeléctrica inherentes a entornos tropicales, estableciendo un estándar de cobertura aplicable a proyectos similares en la región del Golfo de México.

La implementación del sistema se materializó mediante una arquitectura de dos máquinas virtuales sobre servicios de cómputo en la nube usando AWS, donde dicho recurso en nube permite “facilitar la realización de tareas, el trabajo colaborativo, reducción de costes y el desempeño eficiente” (Rodríguez & Andrés, 2024, p. 9), cada una diseñada para optimizar funciones específicas dentro del flujo de datos del proyecto.



La primera máquina virtual se configuró como servidor dedicado para la gestión integral de las comunicaciones LoRaWAN, alojando exclusivamente la suite completa de ChirpStack y sus dependencias críticas. Esta instancia integra Mosquitto como broker MQTT para la gestión de mensajería asíncrona, PostgreSQL como sistema de gestión de base de datos relacional para el almacenamiento de metadatos de dispositivos y configuraciones de red, y Redis como sistema de caché distribuido para optimizar el rendimiento de las consultas frecuentes y la gestión de sesiones.

La segunda máquina virtual se especializó en los servicios de almacenamiento temporal y visualización de datos, integrando InfluxDB como base de datos de series temporales optimizada para el manejo de métricas ambientales de alta frecuencia, y Grafana como plataforma de monitoreo y seguimiento de indicadores, para ello se generó un dashboards interactivos y sistemas de alertas.

El flujo de información opera mediante un pipeline unidireccional donde los dispositivos IoT establecen comunicación directa con la instancia de ChirpStack, la cual procesa y valida los datos antes de transmitirlos hacia InfluxDB. Posteriormente, Grafana consume estos datos para la generación de visualizaciones en tiempo real y análisis históricos.

Ambas instancias fueron implementadas utilizando una arquitectura de microservicios basada en contenedores Docker, lo que garantiza la portabilidad, escalabilidad horizontal y facilita el mantenimiento distribuido del sistema, permitiendo actualizaciones independientes de cada componente sin afectar la operación global de la plataforma.

Para lograr una articulación de todos los servicios se abordó una estrategia de orquestación de contenedores, los cuales se instalaron dentro de dos máquinas virtuales. La primera máquina virtual implementa una arquitectura de microservicios orquestada mediante Docker Compose, integrando siete servicios especializados que operan de manera coordinada. El servicio principal ChirpStack v4 se configura como el núcleo del Network Server, exponiendo el puerto 8080 para la interfaz web de administración y estableciendo dependencias directas con los servicios de soporte.

La configuración incluye ChirpStack Gateway Bridge en dos modalidades: el servicio estándar que opera en el puerto UDP 1700 para la recepción de paquetes LoRaWAN, y una instancia especializada para Basic Station en el puerto 3001, ambas configuradas para la banda de frecuencia US915_1.

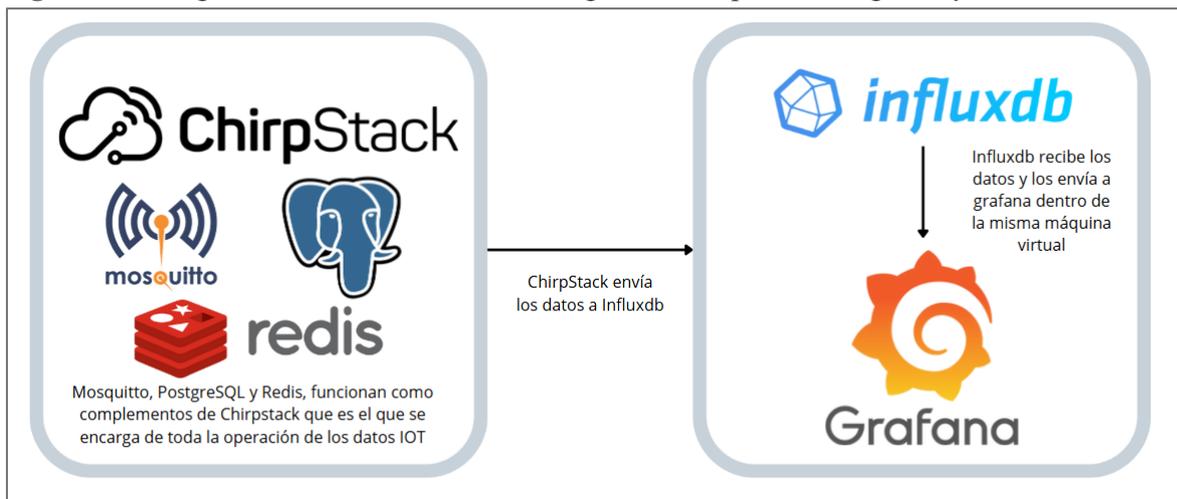


El sistema incorpora además ChirpStack REST API en el puerto 8090, proporcionando interfaces programáticas para la integración con sistemas externos.

Los servicios de soporte incluyen PostgreSQL 14 Alpine como sistema de gestión de base de datos principal, Redis 7 Alpine configurado como sistema de caché con persistencia optimizada mediante snapshots periódicos, y Eclipse Mosquitto v2 como broker MQTT en el puerto 1883. La persistencia de datos se garantiza mediante volúmenes Docker dedicados para PostgreSQL y Redis.

La segunda máquina virtual aloja una arquitectura simplificada de dos servicios críticos: InfluxDB como base de datos de series temporales, configurada con una base de datos denominada "Grafana" y credenciales administrativas predefinidas, expuesta en el puerto 8086. El servicio Grafana se ejecuta en el puerto 3000 con dependencia directa del servicio InfluxDB, configurado con credenciales administrativas y volúmenes persistentes para la conservación de dashboards y configuraciones.

Figura 2: Configuración de red LoRaWAN: Integración ChirpStack con gateway RAKwireless



La integración de dispositivos IoT en la plataforma ChirpStack requiere la configuración previa de perfiles de dispositivo específicos, los cuales definen las capacidades técnicas y parámetros operacionales de cada tipo de sensor.

El registro de dispositivos se realiza a través del módulo "Devices" de la interfaz web de ChirpStack, donde se completan los campos obligatorios de identificación y se asocia el perfil de dispositivo correspondiente. El sistema solicita posteriormente la configuración de claves de seguridad, diferenciando entre root-keys para dispositivos OTAA (Over-The-Air Activation) y session-keys para dispositivos ABP (Activation By Personalization).

En configuraciones donde ChirpStack opera con un join-server dedicado, los dispositivos OTAA pueden utilizar este servidor para la activación automática, eliminando la necesidad de introducir manualmente las root-keys en la plataforma.

La validación del registro se ejecuta mediante el monitoreo simultáneo de las pestañas "Device data" y "LoRaWAN frames" en la interfaz web. Para dispositivos OTAA, el proceso de activación genera inicialmente mensajes JoinRequest seguidos de JoinAccept en el registro de tramas LoRaWAN. Una vez establecida la conectividad, la transmisión de datos genera eventos "Join" y "Up" en la pestaña de datos del dispositivo, confirmando la correcta integración del nodo sensor en la red LoRaWAN y su capacidad para transmitir información telemétrica hacia la infraestructura de procesamiento.

El núcleo de procesamiento del sistema se implementó mediante el módulo RAK4631, el cual integra el microcontrolador Nordic nRF52840 como unidad central de procesamiento. Los chips de la familia nRF se caracterizan por su arquitectura multiprotocolo, proporcionando compatibilidad nativa con múltiples estándares de comunicación inalámbrica, incluyendo Bluetooth Low Energy (BLE), Bluetooth Mesh, Thread, Zigbee, IEEE 802.15.4 y ANT, entre otros protocolos emergentes.

La integración del sistema se realizó sobre la placa base RAK19007, diseñada específicamente para aplicaciones IoT modulares. Esta plataforma proporciona cinco slots de expansión compatibles con el ecosistema RAK, permitiendo la incorporación de módulos especializados para sensores adicionales, sistemas de almacenamiento local, interfaces de usuario y otros periféricos según los requerimientos específicos de cada aplicación.

Para la adquisición de variables ambientales, se implementó el módulo sensor RAK1906, el cual integra cuatro transductores especializados en un único encapsulado. Este módulo multi-sensor permite la medición simultánea de temperatura, humedad, presión barométrica y concentraciones de gases, proporcionando un conjunto completo de parámetros ambientales críticos para aplicaciones de monitoreo industrial.

Las pruebas de integración se diseñaron para validar la funcionalidad punto a punto el sistema, donde se tuviera una prueba de concepto en un clima tropical para evaluar la comunicación bidireccional entre dispositivos IoT con comunicaciones LoRaWAN con la infraestructura cloud de AWS que aloja los servicios de abonados mediante ChirpStack y reportando resultados en el servidor de Grafana.

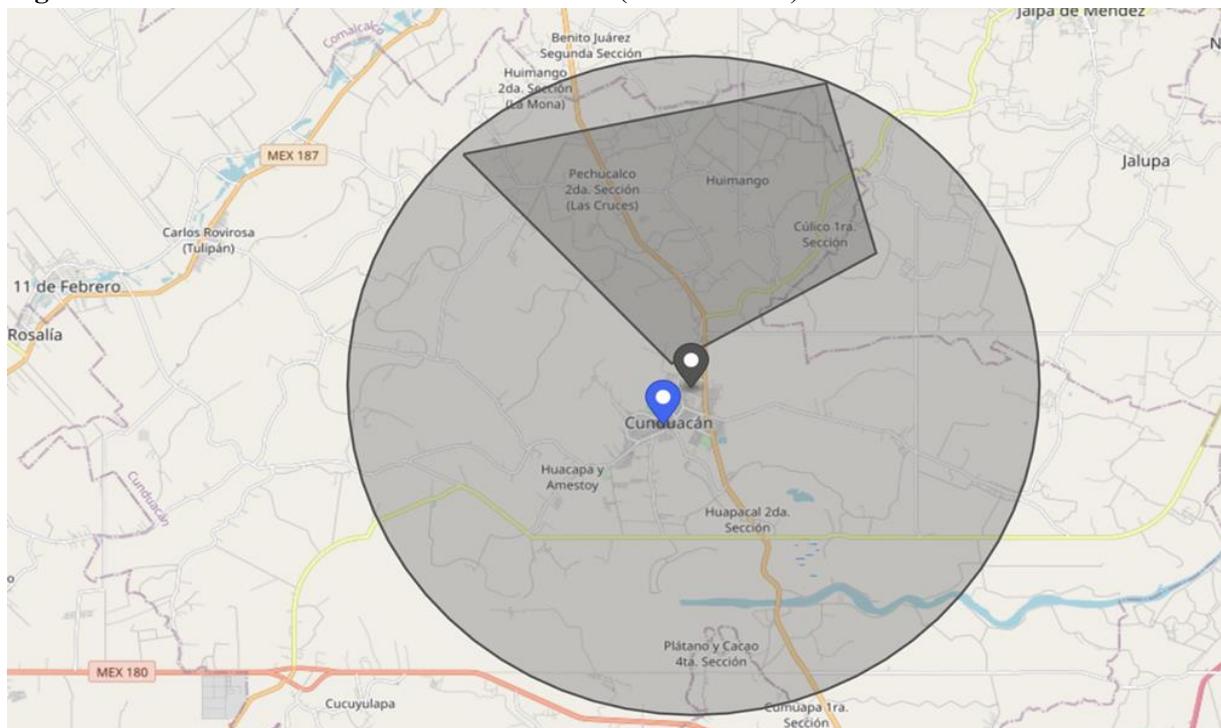


El protocolo de pruebas incluyó la verificación del flujo completo de datos telemétricos (temperatura, gas, presión barométrica y humedad), desde la adquisición en los nodos sensores hasta su procesamiento y visualización en la plataforma Grafana.

Las mediciones de alcance efectivo establecieron una distancia máxima de transmisión de 9.8 km en condiciones de línea de vista directa entre dispositivos IoT y gateway. Este parámetro se determinó mediante pruebas sistemáticas de calidad de señal (RSSI) y relación señal-ruido (SNR) en diferentes ubicaciones geográficas.

Sin embargo, las evaluaciones en zonas con alta densidad de cobertura vegetal en un ambiente de lluvia tropical revelaron atenuaciones significativas en la potencia de señal recibida, fenómeno atribuible a los efectos de dispersión y absorción electromagnética inherentes a entornos forestales densos. A pesar de estas limitaciones de propagación, se validó que la ubicación estratégica de nodos en puntos con condiciones favorables de propagación garantiza la continuidad operacional del sistema de monitoreo, manteniendo la integridad de las comunicaciones dentro de los parámetros de calidad establecidos.

Figura 3: Área de cobertura alrededor de la antena (10 kilómetros)



RESULTADOS

Los resultados demostraron una integración exitosa de todos los componentes, manteniendo la integridad de los datos durante las transmisiones sin registrar pérdidas de paquetes o interrupciones en el servicio. La validación confirmó la interoperabilidad entre los módulos de hardware distribuidos y los servicios de procesamiento remoto, estableciendo la viabilidad técnica de la arquitectura propuesta. Las pruebas de concepto se ejecutaron bajo condiciones ambientales representativas de la región, incluyendo evaluaciones durante eventos de precipitación para determinar la robustez del sistema frente a factores climáticos adversos. Los resultados evidenciaron que el protocolo LoRaWAN mantiene la estabilidad de las comunicaciones incluso durante episodios de lluvia intensa, confirmando la resiliencia de la tecnología ante las condiciones meteorológicas características del clima tropical.

AWS al ser una plataforma de nube muy popular y completa debido a los múltiples servicios que ofrece; es una plataforma de servicios de nube que proporciona una variedad de servicios de cómputo llevando a esta solución a ser flexible y fácil de escalar (Castillo, 2020, p. 3). Por ello el uso AWS como la plataforma central para alojar todos los servicios requeridos para el desarrollo del proyecto, permite que bajo el esquema de máquinas virtuales y contenedores sea rápido de escalar los servicios virtuales que proporciona EC2.

Las dos máquinas virtuales, se ejecutan en todo momento con el sistema operativo Ubuntu, donde en su interior se habilitó para usar Docker a fin de orquestar microservicios con los cuales se realizaría todo el flujo de información para los objetivos deseados. Docker permite montar y orquestar microservicios en cuestión de segundos, lo cual permite enfocarse en el uso de los microservicios antes que programar alguna solución ya existente, además que permite el uso eficiente de los recursos de las máquinas virtuales, permitiendo utilizar infraestructura más económica permitiendo ahorrar costos (Arjona, C, 2020).

Partiendo de estas dos herramientas se planeó una infraestructura donde en una máquina virtual se pudieran alojar exclusivamente los microservicios relacionados con los sensores de IOT. “ChirpStack es un servidor de red LoRaWAN de código abierto que se puede utilizar para configurar redes LoRaWAN.

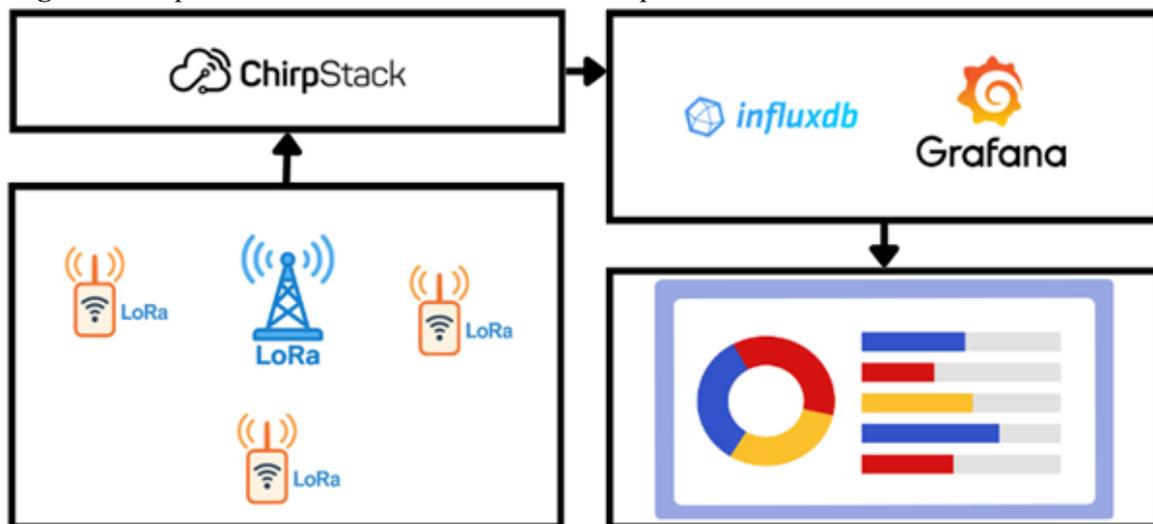


Proporciona una interfaz web para la gestión de gateways y dispositivos, así como para configurar integraciones de datos con los principales proveedores de la nube, bases de datos y servicios comúnmente utilizados para el manejo de datos de dispositivos.” (Lorenzo Castro, 2024, p. 10)

ChirpStack permite la comunicación con el gateway, el cual recibe los datos que envían los sensores IOT, permitiendo visualizar la información necesaria, monitorear el gateway, además de permitir el enlace de los datos con plataformas externas como en este caso con InfluxDB.

La segunda máquina virtual alojó toda la parte relacionada con el sistema de dashboards con InfluxDB y Grafana, gracias a ChirpStack permite enlazar los datos con InfluxDB, esta se encarga de almacenar los datos en tiempo real conforme los recibe ChirpStack, Grafana consume esos datos realizando consultas a InfluxDB y mostrándose en tiempo real de manera gráfica.

Figura 4: Arquitectura de la comunicación de los dispositivos.

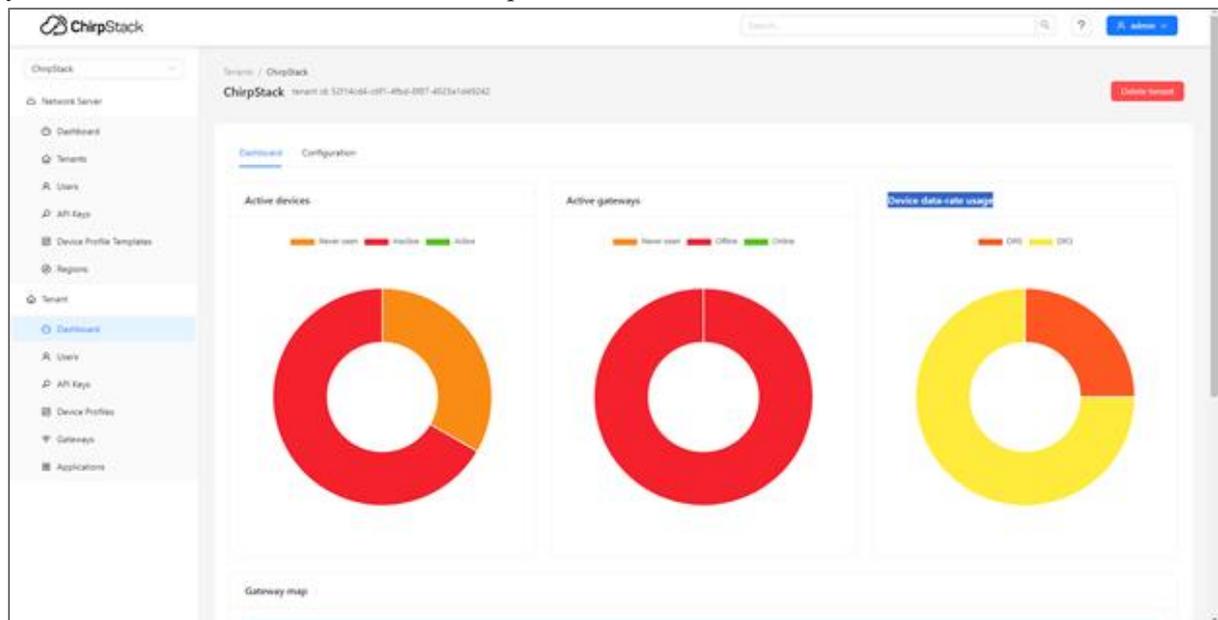


Para la máquina virtual de ChirpStack se montaron con Docker los servicios necesarios para la comunicación con el gateway LoRaWAN. ChirpStack es la interfaz web para la interacción con los servicios de ChirpStack, donde se configuran los datos necesarios para la interacción con el gateway, los sensores IOT y la gestión de perfiles para el acceso a la información. ChirpStack Rest Api permite realizar operaciones CRUD con el servidor de ChirpStack, integrada en este caso para futuras prácticas con dispositivos específicos. ChirpStack Gateway Bridge actúa como puente para la comunicación entre el gateway LoRa y el servidor ChirpStack.

El Gateway Bridge BasicStation funciona como una alternativa del gateway bridge, en caso de entornos con restricciones o gateways más básicos. Mosquitto es un broker MQTT de código abierto que sirve de intermediario para dispositivos de baja potencia (Amaguaya Ramos, 2020, p. 10). PostgreSQL es el gestor de base de datos, en este caso, enfocado en la gestión de dispositivos, aplicaciones, gateway y perfiles. Redis es el motor de base de datos enfocado en el almacenamiento temporal de los datos obtenidos de las aplicaciones y dispositivos alojados en ChirpStack.

El servidor web de ChirpStack ha sido la herramienta principal para el desarrollo del proyecto, ya que monitorea de primera mano los recursos utilizados para todas las prácticas, suministrando un dashboard para el monitoreo del estado de los dispositivos y la tasa de datos utilizada.

Figura 5: Interfaz inicial de ChirpStack donde se muestran los dispositivos activos, gateways activos y el uso de la velocidad de datos de los dispositivos.

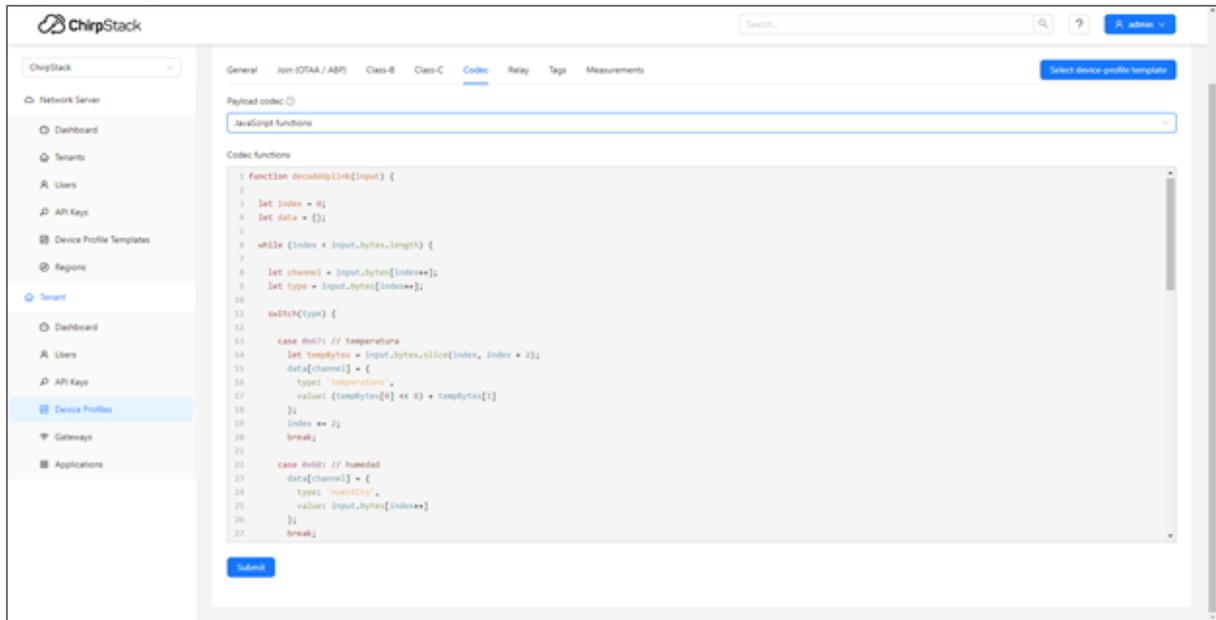


Los perfiles para los dispositivos funcionan para configurar los tipos de dispositivos que se utilizarán y la información que recibirá o transmitirá, en este caso los dispositivos enviaban información y el servidor solo recibía los datos, por ello se le implementó una configuración en la sección de Código donde se decodifican los datos enviados por los sensores de IOT, donde se transforman los datos recibidos que vienen en Bytes para la fácil lectura de estos en la interfaz web.

En este caso se le dio configuración extra relacionado con la forma en InfluxDB recibe y trabaja con los datos, esta parte, funciona para formatear los datos en la forma en que necesitamos trabajar con ellos, se pueden trabajar los datos de una manera más simplificada con una configuración por default

en el Código llamada Cayenne LPP, pero por las necesidades de formateo se optó por suministrar una función de JavaScript donde se especificó la forma en que se decodifican y como queríamos los datos para su uso posterior.

Figura 6: Interfaz donde se escribe el código para la codificación o decodificación de los datos que llegan al dispositivo.



Las aplicaciones se crean para establecer grupos de dispositivos con los que se desea trabajar, constituyendo una forma de agrupar los dispositivos de manera que se puedan tener todos los dispositivos de un mismo tipo juntos o incluso diferentes dispositivos con un mismo propósito. Dentro de las aplicaciones se agregan los dispositivos, para los cuales es importante proporcionar la información detallada y necesaria para su configuración. Los parámetros requeridos incluyen el Device EUI, que corresponde al número de serie proporcionado por cada dispositivo físico; el Device Profile, donde se selecciona el perfil de dispositivo configurado previamente para que la configuración se aplique al dispositivo que se desea agregar; y la Application Key, donde se debe proporcionar una clave de aplicación. En caso de ser el primer dispositivo de la aplicación, ChirpStack genera una clave aleatoriamente, la cual deberá ser utilizada para todos los dispositivos que se integren posteriormente a la aplicación.

Dentro de la parte de dispositivos en la aplicación se tiene un dashboard que proporciona información principalmente sobre el rendimiento del dispositivo, los datos se muestran en un apartado llamado

Events donde se muestran los datos que el dispositivo manda, este puede mandar datos o avisos, como es el caso cuando se activa el dispositivo donde avisa que se conectó con el gateway.

Figura 7: Interfaz donde se muestran gráficamente métricas relacionadas con el uso de los dispositivos.



Figura 8: Interfaz donde se muestran los datos que manda el dispositivo IOT a ChirpStack.

The screenshot shows the 'Events' tab for device 'Rak4'. The table lists events from 2024-06-27 12:22:06 to 2024-05-30 13:51:16. Each event includes a status (e.g., 'join', 'sep'), a device ID, and a data field containing DR, Data, FPort, and FPort2 values.

Time	Status	Device ID	Data	FPort	FPort2
2024-06-27 12:22:06	join	91277945			
2024-06-27 12:17:58	join	80948881			
2024-05-30 13:53:36	sep		DR: 3 Data: #26790x3924859897327489492488c	FPort: 13	FPort2: 3
2024-05-30 13:53:16	sep		DR: 3 Data: #26790x9020514537327489492288d	FPort: 12	FPort2: 3
2024-05-30 13:52:56	sep		DR: 3 Data: #26790x023859897327489492312a	FPort: 11	FPort2: 3
2024-05-30 13:52:36	sep		DR: 3 Data: #26790x902051453732748949232ca	FPort: 18	FPort2: 3
2024-05-30 13:52:16	sep		DR: 3 Data: #26790x910205145373274894922E3F	FPort: 9	FPort2: 3
2024-05-30 13:51:56	sep		DR: 3 Data: #26790x902051453732748949239f0	FPort: 8	FPort2: 3
2024-05-30 13:51:36	sep		DR: 3 Data: #26790x9020514537327489492354F	FPort: 7	FPort2: 3
2024-05-30 13:51:16	sep		DR: 3 Data: #26790x90205145373274894925cc8	FPort: 6	FPort2: 3

Estos datos se muestran formateados en como un arreglo de datos, dentro se muestran desplegados los datos en el formato que configuramos en el Códec, además de los datos del dispositivo como se muestran en las figuras.



Figura 9: Datos crudos que manda el dispositivo IOT a ChirpStack.

```
deduplicationId: "598038ab-38e7-40cd-84a6-e3d207e8ce1f"
time: "2024-05-30T19:53:36.408457+00:00"
deviceInfo: {} 10 keys
  tenantId: "52f14cd4-c6f1-4fbd-8f87-4025e1d49242"
  tenantName: "ChirpStack"
  applicationId: "659081e7-906f-4786-9ae9-ae04a2f3a237"
  applicationName: "APPUJAT01"
  deviceProfileId: "f735d0cb-af29-4f5d-8156-49e8044613a9"
  deviceProfileName: "PROFILEUJAT01"
  deviceName: "Rak4"
  devEui: "ac1f09fffe08e822"
  deviceClassEnabled: "CLASS_A"
  tags: {} 0 keys
  devAddr: "015d09b9"
  adr: true
  dr: 3
  fCnt: 13
  fPort: 3
  confirmed: true
  data: "AWcLowIoXQNzj2gEArAN"
object: {} 4 keys
  gas: {} 1 key
    1: 450.69
  temperature: {} 1 key
    1: 29.79
  humidity: {} 1 key
    1: 46.5
  pressure: {} 1 key
    1: 1008.8
rxInfo: {} 1 item
  0: {} 11 keys
    gatewayId: "ac1f09fffe0922e9"
    uplinkId: 1060246582
    gwTime: "2024-05-30T19:53:36.408457+00:00"
    nsTime: "2024-05-30T19:53:36.490988305+00:00"
    timeSinceGpsEpoch: "1401134034.408457s"
    rssi: -98
    snr: 8.5
    location: {} 0 keys
    context: "AAAAAAAAAAAAAwAerz3cA=="
  metadata: {} 2 keys
    region_common_name: "US915"
    region_config_id: "us915_1"
    crcStatus: "CRC_OK"
```

Las aplicaciones tienen la posibilidad de hacer integraciones con otros servicios como LoRaCloud, http, etc. En este caso se realizó una integración con InfluxDB como primer paso para pasar los datos a Grafana.

En la segunda máquina virtual se montaron dos servicios. Grafana es una herramienta que permite convertir los datos de series temporales en gráficos. InfluxDB es una base de datos para la gestión de series de datos temporales. Una vez creada la base de datos dentro de InfluxDB, los datos necesarios sobre la base de datos se registran en ChirpStack para la integración permitiendo que los datos se transmitan a InfluxDB en tiempo real. Automáticamente detecta los dispositivos y los va seccionando dentro de la base de datos en el momento en el que mandan por primera vez información. Grafana como ChirpStack cuenta con la posibilidad de realizar una integración con InfluxDB, de manera que Grafana consume los datos alojados en la base de datos y permite trabajar con estos para la creación de gráficos interactivos.

Grafana pide una sentencia para realizar la búsqueda de los datos deseados, se puede copiar la misma sentencia que se utiliza en InfluxDB, se accede a la información de forma gráfica en la interfaz de InfluxDB pero se extrae la sentencia para ingresarla en Grafana y que este realice una consulta en la base de datos, Grafana cada 5 segundos consultará los datos actualizando los datos en tiempo real desde InfluxDB, incluso en caso de no tener datos actualizados Grafana realiza un promedio de los datos obtenidos en el transcurso de tiempo seleccionado.



El diseño del dashboard fue personalizado según las necesidades específicas del proyecto. En este caso, se creó un conjunto de dashboards para cada parámetro que evalúa el sensor, donde se obtiene la información de cada dispositivo y se pueden comparar los datos que cada uno transmite. Grafana permite publicar los dashboards para que otras personas puedan visualizar los datos, facilitando la creación de un enlace público para acceder a la información.

CONCLUSIONES

La implementación del prototipo de monitoreo petrolero basado en LoRaWAN y arquitectura cloud-native demostró que es posible integrar de manera eficiente tecnologías IoT, orquestación de microservicios y visualización de datos en tiempo real, incluso bajo las condiciones ambientales extremas del trópico húmedo del sur de México. Los resultados obtenidos validan la viabilidad técnica del sistema, además de su resiliencia frente a factores climáticos adversos y sus capacidades para mantener comunicaciones estables en rangos operativos cercanos a los 10 kilómetros.

La arquitectura distribuida propuesta optimizó el uso de recursos computacionales, garantizó la continuidad operacional y facilitó el mantenimiento independiente de cada componente, cumpliendo con los principios de escalabilidad y portabilidad requeridos en entornos industriales críticos. Asimismo, la incorporación de métricas de desempeño y pruebas de campo en escenarios reales permitió obtener datos empíricos valiosos que fortalecen la evidencia sobre el comportamiento de redes LoRaWAN en contextos de alta humedad y vegetación densa.

Este trabajo aporta un marco metodológico replicable que puede servir como base para futuras implementaciones de monitoreo industrial en regiones remotas. La propuesta representa una alternativa económicamente viable frente a sistemas SCADA tradicionales, democratizando el acceso a tecnologías avanzadas para pequeñas y medianas operaciones petroleras. La investigación no sólo cierra una brecha en la literatura especializada sobre despliegues IoT en ambientes tropicales, sino que sienta un precedente para el desarrollo de soluciones industriales robustas, escalables y sostenibles.

REFERENCIAS BIBLIOGRÁFICAS

Alghamdi, F. (2025). Adaptive real-time channel estimation and parameter adjustment for LoRa networks in dynamic IoT environments. *Sensors*, 25(7), 2121. <https://doi.org/10.3390/s25072121>



- Amaguaya Ramos, R. X. (2020). Análisis comparativo a nivel transaccional de brokers MQTT (Mosquitto, Mosca y Emq) con respecto a la disponibilidad en infraestructuras IoT ante ataques DDoS (Master's thesis, Quito, 2020.).
- Aragónés, R. (2025). Enhanced heat-powered batteryless IIoT architecture with NB-IoT for predictive maintenance in the oil and gas industry. *Sensors*, 25(8), 2590. <https://doi.org/10.3390/s25082590>
- Arjona, C. (2020). Webinar: Docker, creando entorno de desarrollo seguro.
- Bankov, D., Khorov, E., & Lyakhov, A. (2016, November). On the limits of LoRaWAN channel access. In 2016 International conference on engineering and telecommunication (EnT) (pp. 10-14). IEEE.
- Bonavolontà, F., Capunzo, M., De Benedetto, E., Duraccio, L., Giordano, D., Lanzillo, S., Silvestri, S., & Tessitore, S. (2025). Real-time monitoring of energy contributions in renewable energy communities through an IoT measurement system. *Sensors*, 25(5), 1402. <https://doi.org/10.3390/s25051402>
- Castillo Barquero, R. (2020). Servicios de amazon en la nube: AWS.
- Khalilnasl, H. (2025). On the use of containers for LoRaWAN node virtualization: Practice and performance evaluation. *Electronics*, 14(8), 1568. <https://doi.org/10.3390/electronics14081568>
- López, J. J., Lamo, P., Penella-López, M. T., & Nieto-Peroy, C. (2023). Rapid IoT prototyping: A visual programming tool and hardware solutions for LoRa-based devices. *Sensors*, 23(17), 7511. <https://doi.org/10.3390/s23177511>
- Lorenzo Castro, Á. (2024). Sistema de monitorización agrícola mediante comunicación LoRa basado en Chirpstack.
- Mota, A. (2025). Implementation of an Internet of Things architecture to monitor indoor air quality: A case study during sleep periods. *Sensors*, 25(6), 1683. <https://doi.org/10.3390/s25061683>
- Rodríguez, C., & Andrés, E. (2024). Análisis comparativo de protocolos de seguridad en diferentes plataformas de nube, como AWS, Google Cloud y Microsoft Azure (Master's thesis, Quito, Ecuador: Universidad Tecnológica Israel).
- Sánchez, I. (2025). Implementation of a remote monitoring station for measuring UV radiation levels from solarimeters using LoRaWAN technology. *Sensors*, 25(10), 3110. <https://doi.org/10.3390/s25103110>



- Sisinni, E., Fernandes Carvalho, D., Deperassinge, C., Deprez, F., Ferrari, P., Flammini, A., Galimberti, G., Giuliano, R., Pavei, A., Rinaldi, S., & Roedig, U. (2023). Assessing a methodology for evaluating the latency of IPv6 with SCHC compression in LoRaWAN deployments. *Sensors*, 23(5), 2407. <https://doi.org/10.3390/s23052407>
- Yoon, I. (2025). Dual-mode data collection for periodic and urgent data transmission in energy harvesting wireless sensor networks. *Sensors*, 25(8), 2559. <https://doi.org/10.3390/s25082559>
- Zeta, B. M. A. (2025). A low-cost pH sensor for real-time monitoring of aquaculture systems in a multi-layer wireless sensor network. *Sensors*, 25(9), 2824. <https://doi.org/10.3390/s25092824>

