

UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO

**FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA,
INFORMATICA Y MECÁNICA**

ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA



TESIS

**IMPLEMENTACIÓN DE UN PROTOTIPO PARA LA LECTURA
AUTOMATIZADA DE MEDIDORES DE AGUA POTABLE
DOMICILIARIA MEDIANTE UNA RED IOT**

PRESENTADO POR:

Br. SALUSTIO RELY MERCADO HANCCO

**PARA OPTAR AL TITULO PROFESIONAL
DE INGENIERO ELECTRÓNICO**

ASESOR:

Mg. MILTON JHON VELASQUEZ CURO

CUSCO-PERÚ

2025



Universidad Nacional de San Antonio Abad del Cusco

INFORME DE SIMILITUD

(Aprobado por Resolución Nro.CU-321-2025-UNSAAC)

El que suscribe, el Asesor MILTON JHON VELASQUEZ CURO
..... quien aplica el software de detección de similitud al
trabajo de investigación/tesis titulada: IMPLEMENTACION DE UN PROTOTIPO PARA
LA LECTURA AUTOMATIZADA DE MEDIDORES DE AGUA POTABLE DOMICILIARIA
MEDIANTE UNA RED IOT

Presentado por: SALUSTIO RELY MERCADO HANCCO DNI N° 70353074 ;
presentado por: DNI N°:
Para optar el título Profesional/Grado Académico de
INGENIERO ELECTRÓNICO

Informo que el trabajo de investigación ha sido sometido a revisión por 01 veces, mediante el Software de Similitud, conforme al Art. 6° del **Reglamento para Uso del Sistema Detección de Similitud en la UNSAAC** y de la evaluación de originalidad se tiene un porcentaje de 09%.

Evaluación y acciones del reporte de coincidencia para trabajos de investigación conducentes a grado académico o título profesional, tesis

Porcentaje	Evaluación y Acciones	Marque con una (X)
Del 1 al 10%	No sobrepasa el porcentaje aceptado de similitud.	X
Del 11 al 30 %	Devolver al usuario para las subsanaciones.	
Mayor a 31%	El responsable de la revisión del documento emite un informe al inmediato jerárquico, conforme al reglamento, quien a su vez eleva el informe al Vicerrectorado de Investigación para que tome las acciones correspondientes; Sin perjuicio de las sanciones administrativas que correspondan de acuerdo a Ley.	

Por tanto, en mi condición de Asesor, firmo el presente informe en señal de conformidad y adjunto las primeras páginas del reporte del Sistema de Detección de Similitud.

Cusco, 19 de SETIEMBRE de 2025


Firma

Post firma MILTON JHON VELASQUEZ CURO

Nro. de DNI 41053102

ORCID del Asesor 0000-0001-7521-8846

Se adjunta:

1. Reporte generado por el Sistema Antiplagio.
2. Enlace del Reporte Generado por el Sistema de Detección de Similitud: oid:
<https://unsaac.turnitin.com/viewer/submissions/oid:27259:500855019?locale=es-MX>

Rely Mercado

Tesis-RelyMercado.pdf

 Universidad Nacional San Antonio Abad del Cusco

Detalles del documento

Identificador de la entrega
trn:oid::27259:500855019

138 páginas

Fecha de entrega
19 sep 2025, 10:32 a.m. GMT-5

20.593 palabras

Fecha de descarga
19 sep 2025, 10:37 a.m. GMT-5

116.713 caracteres

Nombre del archivo
Tesis-RelyMercado.pdf

Tamaño del archivo
5.6 MB

9% Similitud general

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para ca...

Filtrado desde el informe

- Bibliografía
- Texto citado
- Texto mencionado
- Coincidencias menores (menos de 9 palabras)

Fuentes principales

- 6%  Fuentes de Internet
- 1%  Publicaciones
- 7%  Trabajos entregados (trabajos del estudiante)

Marcas de integridad

N.º de alerta de integridad para revisión

-  **Caracteres reemplazados**
40 caracteres sospechosos en N.º de páginas
Las letras son intercambiadas por caracteres similares de otro alfabeto.

Los algoritmos de nuestro sistema analizan un documento en profundidad para buscar inconsistencias que permitirían distinguirlo de una entrega normal. Si advertimos algo extraño, lo marcamos como una alerta para que pueda revisarlo.

Una marca de alerta no es necesariamente un indicador de problemas. Sin embargo, recomendamos que preste atención y la revise.

DEDICATORIA

Esta tesis está dedicado a mis padres,
gracias por el apoyo incansable y la
confianza depositada durante la
ejecución de esta tesis.

AGRADECIMIENTOS

Agradecer a mi madre, por el cariño y soporte demostrado en cada etapa de mi vida; sin tu presencia nada de esto habría sido posible.

A mi padre, por el cariño y apoyo recibido durante todos estos años.

A mis hermanas, unas grandes personas a quienes quiero y admiro.

A mi asesor, por su invaluable apoyo y orientación durante todo el proceso, su apoyo fue fundamental para alcanzar este logro.

Al laboratorio LIISTI y a los docentes dictaminantes por los sabios consejos brindados para que la tesis vaya a buen puerto.

A la universidad, a la escuela profesional de Ingeniería Electrónica y a sus docentes por la educación brindada durante los años de formación profesional.

Finalmente, a las buenas personas que aparecieron en el camino.

Gracias a todos.

INDICE

CAPITULO I: GENERALIDADES	1
1.1. Ámbito geográfico	1
1.2. Descripción del problema.....	2
1.3. Formulación del problema.....	3
1.4. Justificación.....	7
1.5. Objetivos	8
1.5.1. Objetivo general	8
1.5.2. Objetivo específico	8
1.6. Alcances	8
1.7. Limitaciones	9
CAPITULO II: MARCO TEÓRICO.....	10
2.1. Antecedentes	10
2.2. Internet de las cosas – IoT	11
2.2.1. Medición inteligente - Smart metering	13
2.3. Sistemas de comunicación.....	14
2.4. Tecnologías de comunicación inalámbrica disponibles para redes IoT	14
2.4.1. Wi-Fi.....	15
2.4.2. Bluetooth	15
2.4.3. Redes de baja potencia área amplia (LPWAN)	15
2.4.4. Comparación de tecnologías inalámbricas	17
2.5. Lora.....	19
2.5.1. Ancho de Banda – BW (Bandwith)	21
2.5.2. Factor de dispersión – SF (Spreading Factor).....	21
2.5.3. Tasa de códigos - CR (Coding Rate).....	21
2.6. LoraWAN.....	21
2.7. Estructura de una red LoraWAN	22
2.7.1. Dispositivos Finales (End Devices)	23
2.7.2. Puerta de Enlace (Gateway).....	25
2.7.3. Servidor de Red (Network Server).....	26
2.7.4. Servidor de Aplicaciones (Application Server).....	26
2.8. Seguridad en redes LoraWAN	27
2.9. Activación de dispositivos finales.....	28
2.9.1. Activación por personalización – ABP	28
2.9.2. Activación por aire - OTAA	28
2.10. Seguridad de la red completa	30

2.11.	Sistema de medición de agua.....	31
2.11.1.	Métodos de medición de agua potable.....	31
2.11.2.	Integración en el internet de las cosas – IoT	34
2.11.3.	Smart Metering y la transformación de datos en información útil	35
CAPITULO III: DISEÑO E IMPLEMENTACIÓN DEL PROTOTIPO		36
3.1.	Introducción.....	36
3.2.	Elección de componentes del sistema IoT	36
3.2.1.	Nodo final	36
3.2.2.	Sensor	39
3.2.3.	Gateway LoraWAN.....	41
3.2.4.	Servidor de red	43
3.3.	Arquitectura general de la red IoT	45
3.4.	Configuración del nodo final.....	45
3.5.	Configuración del Gateway	47
3.6.	Configuración del Servidor LoraWAN	50
3.6.1.	Crear aplicación.....	51
3.6.2.	Registrar nodo final	51
3.6.3.	Registrar Gateway.....	53
3.7.	Integración con Node-RED	55
3.8.	Manejo de datos y calibración del sensor	57
3.8.1.	Recolectando parámetros.....	58
3.8.2.	Calibración empírica del sensor	59
3.8.3.	Cálculo tarifario.....	63
3.9.	Integración con la Base de Datos	68
3.10.	Integración con la plataforma de visualización	71
3.11.	Nodos adicionales.....	74
3.12.	Dimensionamiento de cobertura de la red	76
3.13.	Simulación de un sistema	80
CAPITULO IV: PRUEBAS Y RESULTADOS		83
4.1.	Pruebas de potencia de la señal cuando el nodo sensor está bajo tierra ...	83
4.2.	Pruebas de cobertura.....	85
4.3.	Pruebas de medición de volumen consumido y cálculo de errores.....	88
4.3.1.	Cálculo del error absoluto.....	90
4.3.2.	Cálculo del error relativo.....	92
4.3.3.	Comparación con medidores tradicionales.....	94
Conclusiones.....		97
Recomendaciones.....		99

BIBLIOGRAFIA	100
Anexos	104

INDICE DE TABLAS

Tabla 1: Características de las tecnologías LPWAN.	18
Tabla 2: Comparación de características de los nodos propuestos.	38
Tabla 3: Comparación de características de sensores propuestos.	40
Tabla 4: Comparación de características de Gateways propuestos.	42
Tabla 5: Comparación de características de servidores propuestos.	44
Tabla 6: Toma de pulsos proporcionado por los sensores.	60
Tabla 7: Muestras tomadas en campo.....	87
Tabla 8: Toma de medidas hechas por los sensores.	89
Tabla 9: Error relativo y absoluto de las medidas hechas por los sensores	91
Tabla 10: Comparación de medidores de agua aprobados por INACAL con el nodo sensor Dragino.....	95

INDICE DE FIGURAS

Figura 1: Ubicación geográfica donde se implementa el prototipo.	2
Figura 2: Porcentaje de agua no facturada – ANF.	4
Figura 3: Encuesta de satisfacción del cliente SedaCusco.	5
Figura 4: Comparación de tecnologías inalámbricas.	17
Figura 5: Espectrograma de una transmisión Lora.	20
Figura 6: Capas del protocolo LoraWAN.	22
Figura 7: Estructura de una red LoraWAN.	23
Figura 8: Clases de nodos finales del protocolo LoraWAN.	25
Figura 9: Protocolos de comunicación de un Gateway.	26
Figura 10: Seguridad en una red LoraWAN.	27
Figura 11: Vista interior de un medidor mecánico.	32
Figura 12: Medidor electrónico.	33
Figura 13: Medidor ultrasónico.	34
Figura 14: Nodo final Dragino SW3L.	39
Figura 15: Nodo final con su sensor de flujo integrado DW-004.	41
Figura 16: WisGate RAK7268.	43
Figura 17: Arquitectura general de la red IoT propuesta.	45
Figura 18: Claves únicas de cada nodo sensor.	46
Figura 19: Jumper para energizar el nodo sensor.	47
Figura 20: Acceso a internet del Gateway.	48
Figura 21: Configuración del Gateway como Packet Forwarder.	49
Figura 22: Parámetros regionales definidos por Lora Alliance.	49
Figura 23: Configuración de la banda AU915-928 del Gateway.	50
Figura 24: Creación de una aplicación para el proyecto.	51
Figura 25: Registro del nodo final en la plataforma The Things Network.	52
Figura 26: Decoder usado para decodificar el payload enviado por el nodo sensor.	53
Figura 27: Registro del Gateway en la plataforma The Things Network.	54
Figura 28: APIkey creada.	55
Figura 29: Nodo final se une exitosamente al servidor LoraWAN.	55
Figura 30: Credenciales del broker MQTT de TTN.	56
Figura 31: Suscripción al broker de TTN desde NodeRed.	57

Figura 32: Conexión exitosa al broker de TTN.....	57
Figura 33: Flujo para la recolección de datos del bróker TTN.....	58
Figura 34: Algoritmo para recolectar parámetros útiles enviados desde TTN.....	58
Figura 35: Sistema implementado para la calibración del sensor.	55
Figura 36: Algoritmo para calcular el volumen que pasa a través del sensor YF-S201.....	61
Figura 37: Calibración de las medidas tomadas por el nodo sensor.....	62
Figura 38: Algoritmo para calcular el volumen aproximado real que pasa a través del sensor DW-004.....	63
Figura 39: Pliego tarifario actual de la empresa SedaCusco.....	64
Figura 40: Recibo digital emitido por la empresa SedaCusco.....	65
Figura 41: Diagrama de flujo para el algoritmo de cálculo de tarifa.	66
Figura 42: Flujo para tarifar el volumen consumido.....	67
Figura 43: Algoritmo para el cálculo de tarifa de acuerdo al volumen consumido...	67
Figura 44: Datos procesados por Node-Red.....	68
Figura 45: Registramos un Bucket.	69
Figura 46: Credenciales para enlazar NodeRed e InfluxDB.....	70
Figura 47: Flujo para enviar datos a InfluxDB	70
Figura 48: Datos almacenados en la plataforma InfluxDB.	71
Figura 49: Integración de InfluxDB con Grafana.	72
Figura 50: Ventana de visualización del sistema IoT.	73
Figura 51: Integración de NodeRed con Telegram.	74
Figura 52: Consulta de medidas desde el bot de Telegram.....	75
Figura 53: Flujo para reiniciar el conteo del nodo sensor.....	76
Figura 54: Presupuesto de enlace - Link Budget.	77
Figura 55: Mapa de cobertura teórica de la red LoraWAN.....	80
Figura 56: Distribución de un Gateway y 30 nodos sensores.	81
Figura 57: Simulación de una red LoraWAN usando el software CupCarbon.....	82
Figura 58: Simulación de una caja de registro de agua.	83
Figura 59: Pruebas para la toma de RSSI recibido.....	84
Figura 60: Comparación del RSSI entre nodo a nivel del piso y bajo el piso.	85
Figura 61: Rutas recorridas para la toma de medidas de RSSI reales.....	86
Figura 62: Valores reales de RSSI en función de la distancia del nodo.....	87
Figura 63: Comparación de medidas hechas por los sensores VS medida patrón. 90	

Figura 64: Error relativo de las mediciones hechas por los sensores. 93

RESUMEN

Esta tesis implementó un prototipo para la lectura automatizada del consumo de agua domiciliario usando una red IoT basada en el protocolo LoRaWAN. El sistema busca contribuir a una gestión más eficiente del agua en entornos urbanos. La red consta de un nodo sensor (Dragino SW3L) con un sensor de caudal DW-004 y comunicación LoRaWAN, un Gateway (RAK7268) que recolecta los datos, el servidor en la nube The Things Network, una base de datos InfluxDB y la plataforma de visualización Grafana Cloud. Los resultados demostraron la viabilidad del prototipo, logrando una lectura remota con un radio de cobertura de 900 metros. El nodo sensor tiene una autonomía de 10 años enviando datos cada 20 minutos y una precisión del 2% en las mediciones. La plataforma permite a usuarios y la empresa proveedora monitorear en tiempo real el consumo y el costo asociado. Esta solución facilita mejorar el monitoreo, alertar sobre consumos anómalos, analizar patrones de uso y concienciar sobre el recurso hídrico. En conclusión, la implementación de una red IoT se presenta como una solución viable y escalable para una gestión más inteligente y sostenible del agua potable.

Palabras clave: IoT, LoRa, LoRaWAN, medición inteligente.

ABSTRACT

This thesis implemented a prototype for the automated reading of household water consumption using an IoT network based on the LoRaWAN protocol. The system aims to contribute to more efficient water management in urban environments. The network consists of a sensor node (Dragino SW3L) with a water flow sensor DW-004 and LoRaWAN communication, a Gateway (RAK7268) that collects the data, The Things Network cloud server, an InfluxDB database, and the Grafana Cloud visualization platform. The results demonstrated the prototype's viability, achieving remote reading with a coverage radius of 900 meters. The sensor node has a 10-year battery life when sending data every 20 minutes and a measurement accuracy of 2%. The platform allows both users and the utility company to monitor consumption and the associated cost in real time. This solution facilitates improved monitoring, alerts for anomalous usage, analysis of consumption patterns, and awareness of water usage. In conclusion, the implementation of an IoT network is a viable and scalable solution for smarter and more sustainable management of potable water.

Keywords: IoT, LoRa, LoRaWAN, smart metering.

CAPITULO I: GENERALIDADES

1.1. **Ámbito geográfico**

Esta investigación se desarrollará en la ciudad de Cusco, Perú. El diseño, implementación y las pruebas del prototipo se realizaron en el Laboratorio de Investigación e Innovación en Sistemas de Telecomunicaciones y Tecnologías de la Información (LIISTI). Cusco es una ciudad en desarrollo que enfrenta desafíos significativos en la gestión eficiente del agua, con elevadas tasas de agua no facturada y problemas de satisfacción del cliente. La implementación de un prototipo para la lectura automatizada de medidores de agua potable domiciliaria mediante una red IoT permitirá evaluar cómo esta tecnología puede mejorar el servicio de agua y su potencial para ser escalada en una zona más amplia.

Datos:

Departamento: Cusco, Provincia: Cusco, Distrito: Cusco, Altitud: 3399 msnm

Ubicación: 13°15'36"S, 72°6'36"W

Figura 1

Ubicación geográfica donde se implementa el prototipo.



Fuente: Wikipedia.

1.2. Descripción del problema

En el mundo, la administración eficaz del agua es un desafío crítico, especialmente en ciudades en desarrollo donde el incremento de la población urbana, el cambio climático, la gestión deficiente y el riesgo de escasez amenazan el correcto abastecimiento de agua potable a los ciudadanos, esto se ve agravado con la presencia de fugas en las tuberías de distribución, así como la falta de un sistema de monitoreo preciso. Esta situación resalta la necesidad de implementar soluciones tecnológicas innovadoras que permitan un monitoreo más efectivo de los recursos hídricos.

Organismos internacionales como la Organización de Naciones Unidas (ONU) en 2015 dio a conocer un plan de acción denominado Agenda 2030, que tiene como objetivo crear un mundo más próspero e inclusivo para las personas y el planeta, entre sus Objetivos de Desarrollo Sostenible (ODS) el número 6 hace mención: “Garantizar la disponibilidad, la gestión sostenible del agua y el saneamiento para todos” (CEPAL, 2021), lo que muestra una seria preocupación sobre el uso y gestión del agua. A nivel regional, América Latina ha experimentado un crecimiento poblacional acelerado y una rápida urbanización, lo que ha generado una presión

considerable sobre los recursos hídricos y la infraestructura de agua existente. Según Chesini (2022) actualmente más de la mitad de la población mundial vive en entornos urbanos, para el 2050 se estima que entre 7 y 8 de cada 10 personas vivirá en ciudades.

En nuestro país muchas ciudades enfrentan altos niveles de agua no facturada, que pueden alcanzar hasta el 50% en algunos casos, indicando ineficiencias notables en los sistemas de agua. Además, la satisfacción del usuario con la calidad del servicio de agua es frecuentemente baja, con quejas recurrentes sobre problemas como la intermitencia del suministro, la baja presión, la mala calidad del agua y desacuerdos con la tarifa cobrada.

1.3. Formulación del problema

En el Perú, la situación no es diferente. Según datos de la SUNASS, la empresa de servicios de agua local, SEDACUSCO S.A., reporta un porcentaje de agua no facturada que alcanza hasta el 41.71%, lo que sugiere deficiencias significativas en la monitorización y gestión de la red de agua (Gutiérrez, 2016).

Figura 2

Porcentaje de agua no facturada – ANF.

EMPRESA PRESTADORA DE SERVICIOS DE SANEAMIENTO	AGUA NO FACTURADA (ANF) (%)
EMUSAP AMAZONAS	36.02
SEDACAJ S.A.	25.7
EMAPAT S.R.LTDA.	37.47
EMAPA MOYOBAMBA S.R.LTDA.	20.25
EMAPA HUANCVELICA S.A.C.	49.97
SEDAPAL S.A.	30.78
SEDAPAR S.A.	30.18
SEDACUSCO S.A.	41.71
EPS CHAVIN S.A.	41.95
EMPSSAPAL S.A.	44.78
EPS MARAÑON S.R.L.	51.69

Fuente: Adaptado de la Superintendencia Nacional de Servicios de Saneamiento SUNASS.

Adicionalmente, según una encuesta de satisfacción del cliente realizado por EPS SEDACUSCO S.A. (2021), reveló áreas de mejora en la calidad del servicio, obteniendo puntuaciones promedio que apenas superan el nivel satisfactorio.

Figura 3

Encuesta de satisfacción del cliente SedaCUSCO.

Anexo N° 02

EPS:	SEDACUSCO S.A.
N° DE PERSONAS QUE RESPONDIÓ LA ENCUESTA:	344
ENCUESTA DE SATISFACCIÓN DEL CLIENTE 2021	

Ayúdenos a mejorar

Por favor dedique 10 minutos para completar la presente encuesta. La información obtenida servirá para conocer el nivel de satisfacción de nuestros clientes

Sus respuestas serán tratadas en forma CONFIDENCIAL y ANÓNIMA y serán analizadas de forma agregada.

En esta encuesta le haremos una serie de preguntas sobre distintos aspectos de nuestra empresa. P. favor utilice la siguiente escala de valoración

1 = NO

2 = NO, pero con algún matiz

3 = SI, depende

4 = SI, pero con algún matiz

5 = SI

N/A = NO APLICABLE

N°	PREGUNTAS		ESCALA DE VALORACIÓN						N° de Encuestas
			1	2	3	4	5	N/A	
I Respetto al servicio recibido									
1.1	¿Cómo calificaría en general su satisfacción con respecto a los servicios de agua potable y/o desagüe que brinda la EPS? (Calificar del 1 al 5, 1 como mínima y 5 como máxima)	3.34	8	59	120	119	36	2	344
1.2	¿El agua que recibe cubre sus necesidades adecuadamente? (Calificar del 1 al 5, 1 como mínima y 5 como máxima)	3.34	21	57	95	125	46	0	344
1.3	¿El agua que recibe tiene olor, color o sabor anormales? (Calificar del 1 al 5, 1 como con olor, color y sabor anormales y 5 como ningún olor, color o sabor anormales)	3.80	5	43	84	95	115	2	344
1.4	¿La continuidad (horas de servicio) en su zona es adecuada? (Calificar del 1 al 5, 1 como mínima y 5 como máxima)	3.36	25	54	99	101	64	1	344
1.5	¿La presión en su zona es adecuada? (Calificar del 1 al 5, 1 como mínima y 5 como máxima)	3.29	30	54	99	108	52	1	344
1.6	¿Se entera de los avisos de la EPS de cortes del servicio? (Calificar del 1 al 5, 1 como No y 5 como Si)	3.16	80	34	55	97	76	2	344
1.7	¿Las redes de agua se rompen con frecuencia? (Para este caso, si es No calificar como 5, si es Si calificar como 1)	3.20	99	17	42	82	101	3	344
1.8	¿Las redes de desagüe se atoran con frecuencia? (Para este caso, si es No calificar como 5, si es Si calificar como 1)	3.18	102	19	39	72	105	7	344
II Ergonomia y condiciones ambientales									
2.1	¿Como calificaría usted en general la labor o desempeño de la EPS?	3.38	11	52	104	148	29	0	344
2.2	¿Cómo calificaría su satisfacción respecto a la atención al cliente que brinda la EPS?	3.50	8	35	119	139	42	1	344
2.3	¿Como calificaría la información disponible por parte de la EPS?	3.35	14	34	129	147	18	2	344
2.4	¿Considera que el tiempo de espera en la atención en las oficinas de la EPS es el adecuado?	3.23	29	55	107	112	39	2	344
2.5	¿Como calificaría la capacidad técnica del personal de la EPS sobre los servicios que brindan?	3.44	6	48	109	146	33	2	344
2.6	¿Si es que ha presentado algún reclamo, ¿La atención ha sido buena?	3.56	13	48	79	138	64	2	344
2.7	Si es que ha presentado algún reclamo, ¿La solución fue rápida? (Calificar del 1 al 5, 1 como muy lenta y 5 como rápida)	3.33	55	29	69	123	65	3	344
PROMEDIO TOTAL		3.36							
Resultado de la encuesta de satisfacción al cliente en % (Considerando que el valor máximo de 5 sería 100%)		67							

Fuente:

https://www.sedacusco.com/transparencia/gobernanza/encuestas/Encuesta_satisfaccion_cliente_2021.pdf

Estas ineficiencias en la gestión del agua no solo implican pérdidas económicas considerables para las empresas de servicios y la economía local, sino que también perjudican el bienestar de los habitantes. Las altas tasas de agua no facturada pueden ser indicativas de imprecisiones en la medición del consumo, manipulación del medidor y problemas como fugas en la red. Por otro lado, la insatisfacción del cliente con el servicio sugiere deficiencias en la gestión del recurso

que podrían abordarse mediante una mejor monitorización del consumo de agua domiciliaria.

La EPS SEDACUSCO S.A., para determinar los consumos de los clientes para su respectiva facturación, realiza la toma de lectura de los medidores de agua de forma manual, lo que puede generar ineficiencias en el registro de las lecturas y generar cobros indebidos al usuario, además de ser un proceso lento y poco eficiente.

Frente a este panorama, la implementación de un prototipo para la lectura automatizada de medidores de agua potable domiciliaria mediante una red IoT emerge como una solución tecnológica avanzada que podría mejorar significativamente la eficiencia en la gestión del agua. Un sistema de sensores interconectados permitiría la medición precisa del consumo, la detección temprana de fugas y la supervisión continua del servicio de agua, proporcionando datos en tiempo real para una respuesta rápida por parte de los operadores del sistema.

Sin embargo, para lograr una implementación exitosa del prototipo, es necesario abordar los desafíos relacionados con la limitada infraestructura tecnológica, la falta de personal especializado en tecnologías IoT y la necesidad de integrar eficazmente estos sistemas con procesos existentes de gestión del agua. Superar estas barreras es fundamental para aprovechar plenamente el potencial de las tecnologías IoT en la mejora de la eficiencia del servicio.

En este contexto, esta investigación propone evaluar la viabilidad y el impacto de integrar un prototipo de nodo sensor IoT en la medición del consumo de agua en un ambiente controlado, con el objetivo de optimizar el uso de este recurso y acrecentar el nivel de satisfacción del cliente. Al realizar la implementación experimental de este prototipo, se busca generar evidencia sobre los beneficios potenciales de esta tecnología y sentar las bases para su escalabilidad.

Los resultados de este estudio tendrán un impacto considerable para la mejora del servicio de agua, al demostrar cómo la adopción de tecnologías IoT puede conducir a mejoras tangibles como una gestión operativa óptima, reducción de pérdidas y la calidad del servicio. Además, la investigación busca contribuir al desarrollo de capacidades locales en tecnologías IoT aplicadas a la gestión del agua, fomentando la innovación y la transformación digital de este sector.

1.4. Justificación

La implementación de un prototipo IoT para la lectura automatizada de medidores de agua en el ámbito domiciliario tiene el potencial de generar beneficios tangibles para la sociedad. La medición precisa del consumo y la detección de escapes pueden llevar a una disminución notable de las pérdidas no facturadas, lo que se traduce en ahorros económicos tanto para el usuario final como para la EPS, además de una mayor eficiencia operativa. El seguimiento constante del volumen consumido y la identificación de problemas pueden mejorar el bienestar de los usuarios y fortalecer la confianza en la provisión de este servicio esencial, lo que podría motivar una mayor disposición a pagar por los servicios de este recurso.

Desde una perspectiva social, la optimización del manejo del sistema de agua tiene implicaciones directas para el bienestar general de los residentes. La disminución de los desperdicios de agua y el perfeccionamiento del servicio contribuyen a un suministro más confiable y seguro, lo cual es esencial para el bienestar comunitario, el entorno natural y una mejor calidad de vida para la sociedad.

Tecnológicamente, este proyecto aprovecha los avances recientes en IoT, sensores y se evalúa la información para desarrollar una propuesta novedosa para el seguimiento del servicio del recurso hídrico. La investigación demostrará cómo estas tecnologías pueden adaptarse y aplicarse en el contexto de una ciudad en desarrollo, teniendo en cuenta las limitaciones de infraestructura disponible y recursos.

Con esta investigación se demostrará cómo la incorporación de dispositivos sensoriales, comunicaciones inalámbricas y plataformas de evaluación de la información puede transformar la forma en que se mide el consumo del recurso hídrico. Los resultados de este estudio contribuirán al creciente cuerpo de conocimiento sobre la aplicación de tecnologías inteligentes para abordar desafíos de sostenibilidad en ciudades en desarrollo y la conservación de los recursos naturales.

1.5. Objetivos

1.5.1. Objetivo general

Implementar un prototipo para la lectura automatizada de medidores de agua potable domiciliaria mediante una red IoT.

1.5.2. Objetivo específico

- Definir el sensor que formará parte del nodo del prototipo, los módulos para la transmisión y recepción de datos.
- Configurar el nodo sensor y el Gateway para la automatización de la lectura del volumen de agua consumido.
- Integrar el prototipo en un entorno de prueba, asegurando su funcionamiento y optimización dentro de la infraestructura de red.
- Desarrollar un sistema para el análisis de datos y una interfaz de usuario.
- Validar el prototipo del sistema integral.

1.6. Alcances

- El estudio está centrado en la implementación de un prototipo para la lectura automatizada del volumen de agua consumido en viviendas mediante una red IoT.
- Se buscará la tecnología adecuada para el sistema, tomando en cuenta variables como integración de componentes, precios, accesibilidad y fiabilidad.
- La implementación del prototipo se realizará en un ambiente propicio, donde se cuenten con los permisos necesarios y buscar que se asemejen a las condiciones reales.
- Se recolectarán y analizarán los datos generados por el prototipo en el ambiente controlado. Se evaluarán métricas clave como la reducción de pérdidas de agua, cobertura de la red y la precisión en la medición del consumo.

- Se desarrollará una interfaz de usuario para facilitar el acceso y la interpretación de los datos recolectados.
- Se evaluarán los resultados obtenidos con la implementación del prototipo, considerando las métricas expuestas anteriormente.

1.7. Limitaciones

- El estudio se limitará a la implementación de un solo prototipo en un ambiente controlado, donde se tengan los accesos necesarios. Si bien esto permitirá una evaluación detallada del impacto del prototipo IoT, los resultados pueden no ser directamente generalizables a otras áreas o contextos urbanos sin consideraciones y adaptaciones adicionales.
- El prototipo solo contempla la lectura del parámetro de flujo de agua; no se contemplan otras variables como calidad del agua y presión.
- La implementación del prototipo IoT estará sujeta a las limitaciones de infraestructura y recursos tecnológicos disponibles.

CAPITULO II: MARCO TEÓRICO

2.1. Antecedentes

Castillo, Z y Marticonera, C. 2024. En su proyecto de tesis: “**Red LPWAN que mide el consumo de agua potable en el distrito Daniel Hernández**” se desarrolla el estudio e implementación de un sistema destinado a la medición del consumo de agua potable en el distrito Daniel Hernández - Huancavelica. En esta área específica, las viviendas no cuentan con medidores de agua y la facturación del consumo de agua se lleva a cabo según la categoría asignada a cada vivienda, van desde usuarios domésticos, comerciales, industriales o estatales. Este método de tarificación conlleva pérdidas para la municipalidad encargada de la gestión del agua en el distrito, ya que no se cuenta con la capacidad de medir el consumo individual de cada vivienda. Para enfrentar esta problemática, los investigadores desarrollaron un prototipo de red LPWAN punto a punto, se emplearon dos módulos TTGO Lora32 uno como transmisor y el otro como receptor, para sensar el flujo de agua se utilizó el sensor YF-S201 y la interfaz de usuario se desarrolló en el programa LabView. En las conclusiones los investigadores obtienen mediciones del volumen consumidos con errores menores al 2%, el prototipo se estima una autonomía de 3 años y una cobertura de comunicación inferior a 1km. (Marticonera & Castillo, 2024).

Calero, M. 2023. En su trabajo final de máster titulado: “**Red de sensores multiservicio LPWAN, LoRa, LoraWAN, TTN y MQTT**” se estudió una solución de sensores de área extensa aplicados a potenciales ciudades inteligentes. Se inició con el estudio de las distintas tecnologías LPWAN y la infraestructura necesaria para su despliegue, se desarrolló un prototipo de red basado en un sensor y se realizaron pruebas en redes públicas TTN, así como con Gateways privados. También se realizaron diseños teóricos de una red de sensores simulando entornos reales de la ciudad de Zaragoza, se incluyeron distintos tipos de sensores útiles para aplicaciones comunes en una ciudad como: contadores de agua, gestión de alumbrado público, gestión de la basura. En las conclusiones del trabajo indica que para una red de sensores es recomendable el uso de tecnologías LPWAN y dependiendo de los requisitos de la aplicación el más equilibrado en lo que se refiere a alcance, eficiencia energética, tasa de datos, escalamiento y sencillez es LoRa/LoraWAN, también tiene

la ventaja de ser Open Source. En las pruebas reales se lograron alcances máximos en entornos urbanos alrededor de 1.44 Km, RSSI medio de -116.385 dBm, además se realizan recomendaciones de tener cuidado con el relieve del terreno, e indica que LoRa/LoraWAN tiene poca capacidad de atravesar entornos con edificios, sin embargo, en exteriores funciona bastante bien. En la parte de diseño indica que no es sencillo distribuir un alto número de dispositivos, se recomienda estudiar cada caso para la aplicación requerida, pero que la tecnología LoRa/LoraWAN es aplicable para este tipo de proyectos. (Calero, 2023).

Syrmos, E., Sidiropoulos, V., Bechtsis, D. 2023 en su artículo titulado: **“An Intelligent Modular Water Monitoring IoT System for Real-Time Quantitative and Qualitative”** propone un Sistema IoT para monitorear el consumo y calidad del agua en tiempo real, integran sensores avanzados como un flujómetro, sensor de PH, temperatura y conductividad eléctrica, tecnología LoraWAN y aprendizaje automático (ML). Su enfoque busca transformar la infraestructura hídrica tradicional en una red inteligente, sostenible y escalable en el tiempo, aplicable a entornos urbanos y rurales. Usa la plataforma TTN para procesar los datos recolectados, realiza modelos predictivos usando aprendizaje automático para aprender patrones de consumos y realizar predicciones de la calidad del agua. Entre las conclusiones destaca que el sistema demostró ser robusto y preciso, LoraWAN es ideal para aplicaciones que requieren transmisiones de larga distancias y bajos consumos, el sistema es aplicable a ciudades inteligentes. (Syrmos et al., 2023).

2.2. Internet de las cosas – IoT

El internet, un producto de la agencia estadounidense Advanced Research Projects Agency (ARPA) creada durante la década de los 50's en la guerra fría, con la visión de crear una red de comunicación directa entre computadoras de los centros de investigación situados en distintos lugares. A partir de esa fecha, el alcance y complejidad de la red solo fue aumentando, adoptando protocolos como TCP/IP, implementando nuevos recursos como HTML, HTTP y un programa llamado Web Browser. El uso de la red estaba restringido para una audiencia de científicos, ingenieros y trabajadores del gobierno, pero en los años 90's se diseñó el primer módem telefónico lo que facilitó el acceso a internet al público general.

El concepto de “Internet of Things” fue creado en 1999 por el ingeniero Kevin Ashton, durante los años 2008-2009 el concepto creció con más fuerza; se estaba desarrollando y creciendo mucho más allá de pequeñas comunidades de expertos en informática. Hasta hoy donde grandes empresas tecnológicas como Google, Amazon, Cisco, Huawei entre otras adoptaron esta tecnología ofreciéndola en productos y servicios.

El internet de las cosas IoT es una arquitectura de información global basada en internet, donde objetos dotados de capacidad de cómputo y conectividad generan, intercambian y consumen datos con una mínima intervención humana. Así IoT está revolucionando el concepto de internet tal como lo conocíamos, sumando una interconectividad mayor, una mejor percepción de la información y servicios inteligentes más completos.

Actualmente se pueden conectar objetos cotidianos, electrodomésticos, automóviles, termostatos y casi cualquier objeto que se pueda imaginar a internet, el mundo físico y el digital se complementan y cooperan entre ellos. Los datos recolectados por sistemas IoT tienen un gran valor para las personas y empresas, conducen a tomar decisiones más informadas, optimizar procesos, reducir costos, mejorar la experiencia del usuario y personalizar servicios.

En 2018 se estimó que había 22 mil millones de aparatos IoT conectados a internet y recientes pronósticos afirman que hasta el año 2030 serán alrededor de 50 mil millones. Los datos generados deben ser almacenados, procesados y analizados; esto se relaciona directamente con el paradigma de “Cloud computing” o computación en la nube, que representa la oferta de servicios de computación, procesamiento y almacenamiento a través de internet. (Tarazona, 2020).

Conceptos como “Ciudad Inteligente – Smart City”, agricultura de precisión, big data no se podrían desarrollar si no van de la mano con sistemas IoT. El internet de las cosas IoT aporta hiperconectividad y medición de las cosas en tiempo real al concepto de Smart City. Se pueden encontrar cientos de casos de uso y aplicaciones que proporcionan inteligencia a una ciudad, ejemplos como gestión inteligente de basura (Smart Waste Managment), medición inteligente (Smart metering), gestión del tráfico, gestión inteligente de la iluminación entre otros más.

2.2.1. Medición inteligente - Smart metering

El Smart metering es uno de los sistemas clave de las Smart Cities, se basan en el uso de Smart Meters (medidores o contadores inteligentes). Este tipo de contador eléctrico digital captura datos sobre el uso de un recurso (agua, electricidad, gas, etc) en tiempo real y la envía de forma segura a un centro de control de operaciones.

El tema abordado en la presente tesis se centra en la captura de datos detallados sobre el consumo de agua potable domiciliario a través de un contador inteligente tele-gestionable. Los nuevos desafíos que requiere una Smart City hacen imprescindible mejorar o reemplazar los viejos contadores mecánicos que van quedando obsoletos, debido a la digitalización y aparición de nuevas tecnologías. Estos nuevos medidores inteligentes o Smart Meters consisten en contadores electrónicos de alta eficiencia capaces de hacer lectura del consumo del recurso y almacenar datos para posteriormente enviarlos a una central. Otra característica importante es que son automatizados y de control remoto, de manera que no requiere de un personal que haga la lectura del medidor de manera manual a diferencia de los contadores convencionales.

El Smart Metering está revolucionando el sector hídrico, entrega herramientas potentes para la gestión eficiente y sostenible del agua. Estos dispositivos van más allá de la simple medición, otorgan beneficios como: (IDRICA, s. f.).

- **Eficiencia en la gestión:** Tener información del consumo preciso de agua permite a las empresas de servicios públicos la toma de decisiones informadas, para una gestión eficiente del recurso.
- **Detección temprana de fugas:** Monitorizar en tiempo real el flujo de agua permite la detección rápida de anomalías en los consumos. Con esto no solo se reduce el desperdicio de agua, también reduce el impacto ambiental y costos relacionados a reparaciones.
- **Facturación precisa y transparente:** Smart Metering permite una tarificación transparente basada en el consumo real, lo que conduce a una relación

más ecuánime entre las empresas prestadoras del servicio y los clientes finales.

- Cambios en el comportamiento del consumidor: Los usuarios al tener conocimiento detallado sobre su consumo pueden tomar conciencia de sus hábitos y ajustarlos para tener un consumo responsable y contribuir a la conservación del agua.
- Automatización y optimización: Una medición inteligente (Smart Metering) permite automatizar procesos, como la lectura de medidores y administración de datos, con esto se optimizan los recursos y se mejora el servicio.

2.3. Sistemas de comunicación

Una red IoT no se puede desplegar si no existe una infraestructura IT (o TIC) de telecomunicaciones, los centros urbanos tienen la ventaja de contar con una infraestructura de telecomunicaciones previa, lo que nos indica que existe un nivel de penetración de internet y conectividad alto o que va en aumento.

Los “Smart Meters” se distribuyen en una superficie grande y toda la información recolectada se tienen que transmitir de alguna forma a una central que gestione los datos. Actualmente se dispone de distintos medios para transmitir la información a través de distintos medios, como cableado o medios no físicos, como ondas de radiofrecuencia, wifi, bluetooth, comunicación celular.

Los dispositivos conectados mediante medios cableados son poco frecuentes, por lo que nos enfocaremos en los dispositivos de comunicación inalámbrica.

2.4. Tecnologías de comunicación inalámbrica disponibles para redes IoT

Los dispositivos IoT mantienen una conexión continua con las plataformas de almacenamiento, procesamiento y visualización, estos centros de recolección suelen ser servidores y se accede a ellos a través de internet.

Los dispositivos IoT pueden transmitir la información recolectada a través de diferentes protocolos, no existe un único o mejor forma, la elección correcta depende

de las necesidades particulares de la aplicación, lo importante es que los paquetes enviados por los dispositivos IoT lleguen a los servidores. Los principales métodos usados para conectarse se desarrollan a continuación.

2.4.1. Wi-Fi

Es una de las más utilizadas al ser una de las tecnologías más extendidas en la actualidad, es propiedad de la Wi-Fi Alliance y se fundamenta en el estándar IEEE 802.11. Es utilizado en redes de áreas locales LANs y permite que dispositivos se conecten a internet a través de un router inalámbrico, utiliza las bandas de frecuencia 2.4 GHz y 5 GHz para transmitir información. (IEEE, 2016).

2.4.2. Bluetooth

Es una tecnología de radio de corto alcance, diseñada para el intercambio de datos sin necesidades de medios cableados y utiliza la banda ISM sin licencia de 2.4 GHz. Una variante es Bluetooth Low Energy (BLE), que fue diseñado para aplicaciones IoT que requieren consumos energéticos muy bajos y aplicaciones que no requieren conexiones continuas. Admite múltiples topologías de comunicación: punto a punto y malla. (Bluetooth, s.f.)

2.4.3. Redes de baja potencia área amplia (LPWAN)

Las LPWAN proyectan una alternativa sólida para los millones de aparatos previstos para el internet de las cosas. Las redes LPWAN están diseñadas para aplicaciones que requieren conectividad de largo alcance, bajos consumos de energía y manejan pequeñas tasas de transferencia de datos, lo que presenta un desafío para las tecnologías inalámbricas tradicionales. Algunas usan el espectro no licenciado como Lora y Sigfox, a diferencia de 2/3/4G, NB-IoT que si usan bandas licenciadas.

- **LoraWAN:** LoraWAN es una especificación de red de baja potencia y área amplia (LPWAN) diseñada para conectar inalámbricamente cosas provistas de una batería a internet en redes regionales, nacionales o globales. (LoRa Alliance, s.f.). Permiten conexiones de largo alcance y comunicación bidireccional, baja frecuencia de transmisión, alta inmunidad al ruido. Usa las bandas de radiofrecuencia sin licencia, debido a esto usar el espectro

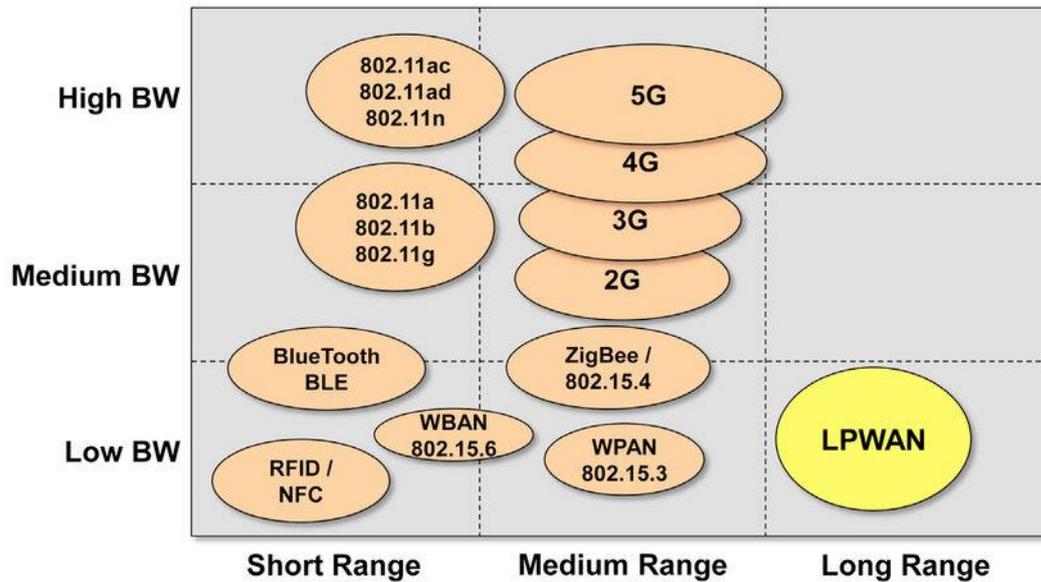
asignado a LoraWAN no tiene costos asociados y no hay necesidad de pedir permisos para su uso ante ningún ente regulador.

- **Sigfox:** Tecnología pensada para dispositivos de bajo consumo y su despliegue sea independiente de operadores de telefonía móvil. Sigfox es un operador de red global que implementa redes inalámbricas (despliega antenas y repetidores) para conectar dispositivos de bajo consumo y está diseñado para comunicaciones de baja velocidad. Entre las desventajas está la falta de despliegue de la red en algunas zonas y el costo asociados por la suscripción al uso de su red privada.
- **NB-IoT:** Es una tecnología LPWAN que utiliza la red de telefonía móvil existente para la comunicación inalámbrica, maneja pequeñas cantidades de transmisión de datos bidireccionales de manera segura, eficiente y confiable. NB-IoT mejora significativamente el consumo de energía al operar con potencias de transmisión bajas además de tener una cobertura profunda al usar la infraestructura de las compañías de telefonía móvil. La principal desventaja son los costos asociados que se tienen que pagar al operador dueño de la infraestructura.

2.4.4. Comparación de tecnologías inalámbricas

Figura 4

Comparación de tecnologías inalámbricas.



Fuente: Estudio en detalle de LoraWAN. Comparación con otras tecnologías LPWAN considerando diferentes patrones de tráfico (Hernández, 2020).

Cada tecnología tiene sus puntos fuertes y sus limitaciones, en la Figura 4 se aprecia claramente que una sola tecnología no puede cubrir todos los requerimientos para las variadas aplicaciones IoT que se puedan presentar.

En la siguiente tabla se hace una comparativa de las tecnologías inalámbricas LPWAN disponibles para IoT.

Tabla 1*Características de las tecnologías LPWAN.*

Tecnología	LoraWAN	SigFox	NB-IoT
Frecuencia	Banda ISM sin licencia (915 MHz para Perú)	Banda ISM sin licencia (915 MHz para Perú)	Bandas de frecuencia LTE con licencia.
Modulación	CSS	BPSK	QPSK
Ancho de Banda	250 y 125 KHz	100Hz	200 KHz
Velocidad máxima de datos	50 kbps	100 bps	200 kbps
Alcance	5 km (urbano), 20 km (rural)	10 km (urbano), 40 km (rural)	Misma de la telefonía celular.
Consumo Energético	Muy bajo	Muy bajo	Bajo
Inmunidad a la interferencia	Muy alto	Muy alto	Bajo
Ventajas	Bajo consumo, largo alcance, comunicación bidireccional semidúplex.	Red global desplegada, muy bajo consumo.	Alta capacidad de datos, alta cobertura urbana.
Desventajas	Baja velocidad, inversión inicial alta.	Limitado tamaño de mensaje y poca capacidad de datos, costos asociados al suscribirse a la red SigFox.	Costos operativos altos, requiere de una operadora telefónica.

Fuente: Elaboración propia.

LPWAN cubre necesidades de transmisión de largas distancias, con un uso de ancho de banda bajo y consumos bajos de potencia, así se decidió por conveniente

escoger la tecnología LPWAN para el desarrollo de esta tesis, en específico usar LoraWAN.

2.5. Lora

Lora fue desarrollada por Cycleo en Francia y posteriormente adquirida por Semtech, un fabricante de chips de radio. La capa de acceso al medio (LoraWAN) está administrada por la Lora Alliance quien dedica sus esfuerzos a la estandarización del protocolo LoraWAN que usa Lora como tecnología de comunicaciones subyacente, en cambio Lora se ocupa únicamente de la capa física.

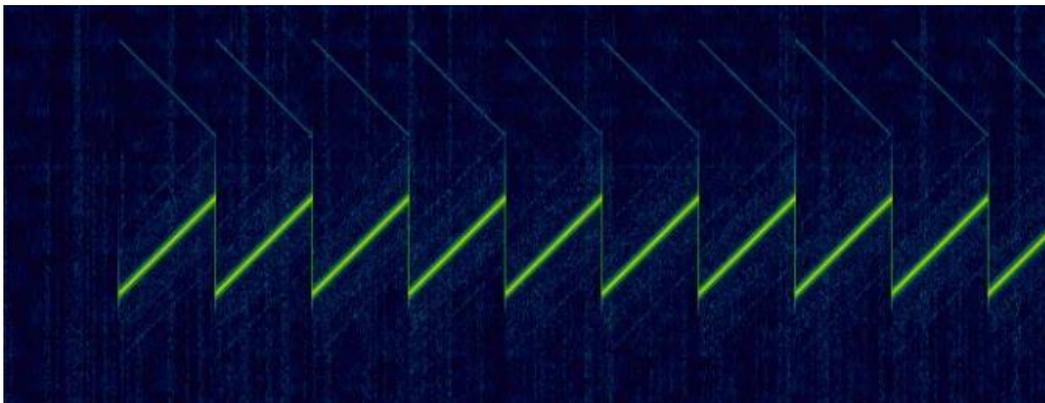
LoRa (Long Range): Es la capa física (PHY), que constituye la técnica de modulación utilizada para la comunicación de largo alcance y baja potencia. LoRa es la señal de radio que transmite los datos a través del aire utilizando la tecnología de espectro extendido. (Hernández, 2020).

Lora utiliza una modulación de espectro ensanchado (Spread Spectrum) el cual propaga una señal de banda estrecha sobre un ancho de banda BW de canal más amplio, al aumentar el BW de la señal se compensa la degradación de la relación de señal/ruido de un canal de radio. Lora usa la técnica de modulación de capa física CSS (Chirp Spread Spectrum), CSS trabaja al tomar una señal sinusoidal y variar su frecuencia linealmente con el tiempo formando así una señal de chirrido (Chirp), esta señal posteriormente se modula en una frecuencia portadora.

Lora varía de manera continua y lineal una frecuencia portadora, en el rango de una $F_{inicial}$ y F_{final} . Cuando llega a la frecuencia final se resetea regresando nuevamente el valor de la frecuencia de inicio y repite el ciclo. Cada tiempo de símbolo T_s la frecuencia salta a un valor fijo y comienza a crecer, el valor al que salta representa el símbolo del mensaje enviado. La cantidad de bits que se pueden codificar por símbolo está dada por el Factor de dispersión (SF- Spreading Factor)

Figura 5

Espectrograma de una transmisión Lora.



Fuente: Modulación LoRa (Solera, 2018).

Esta técnica de modulación proporciona las siguientes ventajas en una red LPWAN:

- **Robustez:** Las señales Chirp son resistentes frente a interferencias y al ruido, lo que es imprescindible en un ambiente tan diverso como es una ciudad.
- **Alta sensibilidad de recepción:** CSS hace posible extraer la señal incluso cuando está degradada por la distancia, obstáculos o interferencia de otras señales de radio. Lora permite alcanzar altos niveles de sensibilidad de hasta -137dBm.

Lora modula las señales en las bandas ISM a frecuencias menores a 1 GHz; para Europa utiliza la banda de 868 MHz, 915 MHz para Norte América, 433 MHz en Asia, etc. La asignación completa está publicada en la página web de la Lora Alliance, donde describe los parámetros específicos “Regional Parameters” para diferentes regiones regulatorias en todo el mundo. (LoRa Alliance, s.f.).

La modulación Lora está caracterizado por parámetros que determinan su rendimiento y capacidad en una red LoraWAN:

2.5.1. Ancho de Banda – BW (Bandwith)

El ancho de banda BW determina la velocidad de transmisión de datos y el alcance que tendrá. Los valores comunes de ancho de banda son 125KHz, 250 KHz y 500 KHz, un ancho de banda mayor es sinónimo de una transferencia mayor de tasa de datos sin embargo se reduce la sensibilidad de recepción y la distancia de transmisión.

2.5.2. Factor de dispersión – SF (Spreading Factor)

El factor de dispersión SF determina la cantidad de símbolos usados para representar un solo bit de información, cuanto mayor sea el SF mejor será la transmisión ya que se aumenta el tiempo de transmisión por símbolo, lo que incrementa la sensibilidad y el alcance, sin embargo, se reduce la tasa de datos y requiere una mayor cantidad de energía para realizar la transmisión. Típicamente los valores de Spreading Factor van de SF7 – SF12.

2.5.3. Tasa de códigos - CR (Coding Rate)

La tasa de códigos (CR) es el grado de redundancia en los datos transmitidos, aumenta el número de bits transmitidos para realizar la corrección y detección de errores. Un mayor CR significa más redundancia lo que aumenta la robustez de la señal frente a interferencias, pero reduce la eficiencia espectral al ocupar mayor tiempo el espectro y además de consumir mayor energía. Los valores típicos de CR son 4/5, 4/6, 4/7 y 4/8.

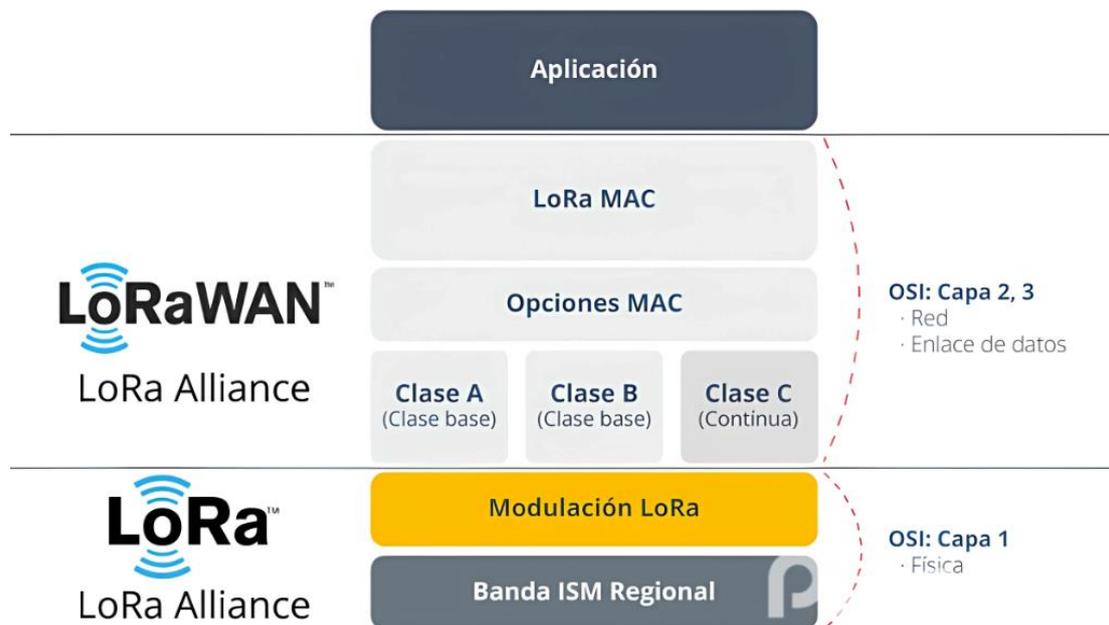
2.6. LoraWAN

Es un protocolo de red de área amplia de baja potencia (LPWAN) diseñada con el objetivo de gestionar la comunicación de dispositivos basados en modulación Lora, LoraWAN es el protocolo que define la arquitectura de la red y las reglas de acceso al medio MAC (Media Access Control) definiendo como los dispositivos finales se conectan a la red, se encarga de gestionar los mensajes y la seguridad de la red.

LoraWAN proporciona un protocolo de nivel superior que va más allá de la capa física, por lo tanto, se encarga de gestionar la comunicación mientras que Lora es la tecnología base para que esto ocurra.

Figura 6

Capas del protocolo LoraWAN



Fuente: LoraWAN y sus aportaciones a las tecnologías IIoT (Baltuille, 2023).

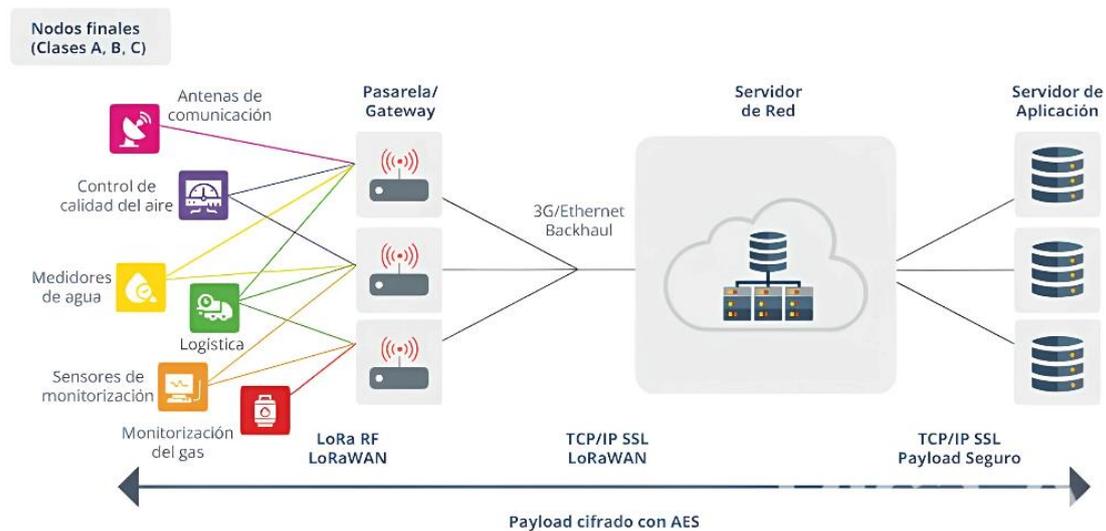
Entre sus puntos fuertes sobresalen: la comunicación bidireccional, seguridad a todos los niveles, movilidad y servicios de localización. Con LoraWAN se llega a alcanzar enlaces de hasta 20 Km en condiciones ideales de visibilidad entre dispositivos. Así en ciudades se reduce a 2 Km y en zonas rurales de 10 a 15 Km.

2.7. Estructura de una red LoraWAN

Una red LoraWAN se despliega en una topología estrella de estrellas, donde la puerta de enlace llamado también "Gateway" es el puente entre los dispositivos finales y un servidor de red centralizado. El Gateway actúa como un puente donde los paquetes RF que recibe de los nodos finales los convierte a paquetes IP y viceversa.

Figura 7

Estructura de una red LoRaWAN.



Fuente: LoRaWAN y sus aportaciones a las tecnologías IIoT (Baltuille, 2023).

Las especificaciones técnicas para LoRaWAN dada por la Lora Alliance proporcionan una interoperabilidad perfecta entre todos los fabricantes, lo que hace que se tenga una libertad de innovación y diferenciación en los productos ofrecidos. Esto se traduce en la aparición de múltiples fabricantes y más productos IoT que usan el protocolo LoRaWAN.

La arquitectura de una red LoRaWAN se compone de varios elementos interconectados que trabajan juntos, aunque cada uno con una función específica dentro de la red, a continuación, detallaremos cada componente.

2.7.1. Dispositivos Finales (End Devices)

También llamados Nodos Finales, son los dispositivos que son el origen o el destino de las transmisiones dentro de una red LoRaWAN, estos nodos tienen integrados un módulo de radio Lora usado para establecer el enlace de comunicación, vienen integrados con capacidades de detección o sensorización de un fenómeno y tienen capacidades limitadas de potencia de cálculo. Un dispositivo final LoRaWAN no se comunica específicamente con un único Gateway: todos los Gateways presentes en el área de cobertura reciben los mensajes del dispositivo y los procesan. (Montagny, 2020).

Los dispositivos finales se clasifican en 3 clases diferentes, dependiendo del consumo de energía y de la forma en la que maneja la comunicación, cada uno tiene características únicas que optimizan esta tecnología de acuerdo a los requerimientos que se necesiten, a continuación, se presentan estos tres tipos.

- **Clase A:** En principio todos los dispositivos finales son de clase A, es la clase por defecto. Los nodos inician la comunicación con una transmisión de enlace ascendente (Uplink) de forma asíncrona ósea que no verifica la disponibilidad del Servidor, esta transmisión va seguida por dos ventanas de recepción Rx1 y Rx2 donde el Gateway puede enviar una transmisión de enlace descendente (Downlink), así los dispositivos de clase A tienen comunicación bidireccional, pero siempre lo inicia el Nodo Final nunca el Servidor. Esta clase permite las autonomías más altas, el nodo entra en reposo hasta que se tenga que transmitir un nuevo mensaje Uplink.
- **Clase B:** Tienen un comportamiento parecido a la clase A, pero la diferencia es que el dispositivo se sincroniza con el Gateway; además de las ventanas Rx1 y Rx2 se habilitan ventanas de recepción periódicas, esto le da la capacidad al servidor que sea consciente del momento en que el nodo final está listo para recibir una transmisión Downlink. Esta clase es la menos común entre dispositivos finales debido a la dificultad de implementación, además consume más energía que los dispositivos de clase A.
- **Clase C:** Tiene un funcionamiento contrario a los nodos de clase A, las ventanas de recepción siempre están abiertas entre dos enlaces ascendentes (Uplink) y solo se cierran cuando el nodo transmite. Al estar escuchando continuamente estos dispositivos son los que menos autonomía consiguen.

Figura 8

Clases de nodos finales del protocolo LoraWAN.



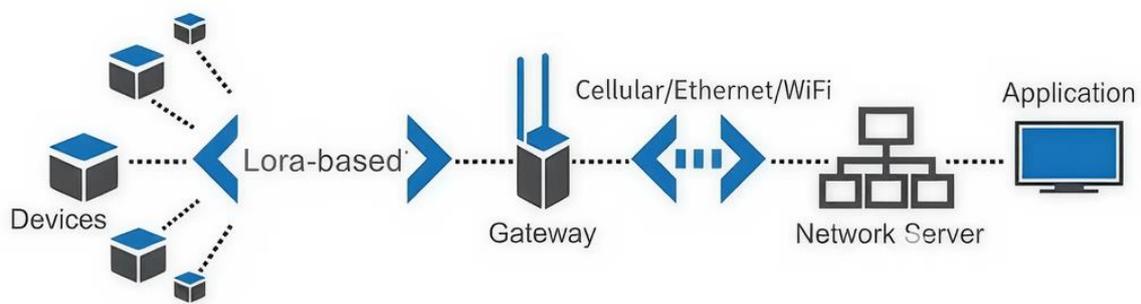
Fuente: LoraWAN y sus aportaciones a las tecnologías IIoT (Baltuille, 2023).

2.7.2. Puerta de Enlace (Gateway)

Una puerta de enlace actúa como punto de acceso para los dispositivos finales, por un lado, recibe una modulación Lora en su antena y por otro lado lo transmiten a un servidor de red a través de conectividad basada en IP como Wi-Fi, 3G, 4G, 5G, Ethernet, LTE, etc. Los Gateways también reciben mensajes del servidor y los envían a los dispositivos finales.

Figura 9

Protocolos de comunicación de un Gateway.



Fuente: ¿Cómo envía y recibe datos el sensor LoRa? (MOKOSmart, 2019).

2.7.3. Servidor de Red (Network Server)

El servidor de red tiene la función de gestionar la comunicación entre los dispositivos finales y las aplicaciones, se encarga de la autenticación de dispositivos, gestiona claves de seguridad y se encarga de la optimización de la red. El servidor de red recibe los mensajes transmitidos por el Gateway, los pasa por un proceso de autenticación gracias a la clave NwkSKey (Network Session Key - verifica que el mensaje sea de un nodo registrado), el servidor de red no decodifica el mensaje (en ningún momento conoce el contenido del mensaje). Si el proceso de autenticación falla, se desecha el mensaje; por el contrario, si el proceso de autenticación tiene éxito, se transfiere el mensaje codificado al servidor de aplicaciones.

2.7.4. Servidor de Aplicaciones (Application Server)

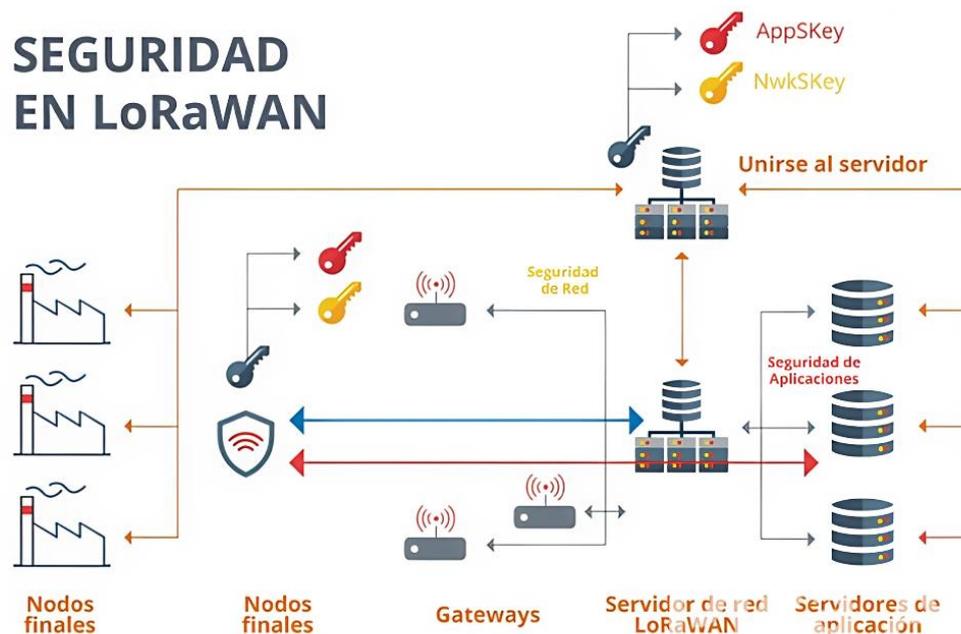
El servidor de aplicaciones es la capa última en una arquitectura de red LoraWAN, dentro del servidor se aloja el software encargado de procesar y almacenar los mensajes recibidos de forma segura, el servidor de red decodifica los mensajes recibidos gracias a la clave AppSKey (Application Session Key). El servidor de aplicaciones procura una interfaz web y APIs para la administración de usuarios, organizaciones, aplicaciones y dispositivos.

2.8. Seguridad en redes LoRaWAN

La seguridad se vuelve un aspecto fundamental en el despliegue de una red LoRaWAN, se debe garantizar la confidencialidad e integridad de los datos transmitidos, se usa algoritmos AES-128 para el cifrado de los datos.

Figura 10

Seguridad en una red LoRaWAN.



Fuente: LoRaWAN y sus aportaciones a las tecnologías IIoT (Baltuille, 2023).

Entre los mecanismos de seguridad implementados en LoRaWAN podemos destacar los siguientes:

- **Cifrado de extremo a extremo:** Los datos se encuentran encriptados desde el dispositivo final hasta el servidor de red, mediante la clave NwkSKey.
- **Autenticación de dispositivos:** Es obligatorio que un dispositivo final se autentique antes de enviar o recibir datos, mediante un proceso de unión (Join Procedure) donde se le asignan claves únicas al nodo.

- **Integridad de datos:** Se asegura que los datos no sean modificados durante el proceso de transmisión gracias a códigos de integridad. Los datos serán únicamente accesibles para la aplicación autorizada.

2.9. Activación de dispositivos finales

En LoraWAN existen dos métodos de activación de los nodos finales: Activación por personalización (ABP - Activation by Personalization) y Activación por aire (OTAA - Over The Air Activation), activación está referido al proceso por el cual un dispositivo final se une a la red y obtiene claves de seguridad que le permitan enviar y recibir datos. Los tres elementos principales para que se establezca la comunicación son:

- **DevAddr:** Es la dirección del dispositivo, usado para identificar al nodo.
- **NwkSKey:** Es una clave de seguridad para el proceso de autenticación.
- **AppSKey:** Es una clave de seguridad para el proceso de cifrado.

2.9.1. Activación por personalización – ABP

Es el método más sencillo, usado para realizar prototipos y pruebas rápidas de comunicación con la red LoraWAN, las claves de activación: DevAddr, NwkSKey y AppSKey se configuran manualmente en el nodo sensor, así mismo estas mismas claves se almacenan en el servidor. Una vez configurados, los nodos empiezan a comunicarse con la red sin pasar por el proceso de unión.

Tiene la ventaja de permitir despliegues sencillos y rápidos, pero en contra tiene que la seguridad es estática; las claves no se renuevan a menos que se configure manualmente. Si un atacante tiene acceso a las claves preconfiguradas en los nodos finales la seguridad de la red se ve comprometida.

2.9.2. Activación por aire - OTAA

Con este método las claves de activación: DevAddr, NwkSKey y AppSKey se generan dinámicamente durante un proceso de unión, durante este proceso el nodo y el servidor de red intercambian mensajes para generar las claves de activación ya mencionadas.

Los dispositivos finales se deben configurar con elementos identificadores denominados: Identificador Único Extendido (EUI – Extended Unique Identifier) los cuales son:

- **DevEUI:** Device EUI, es un identificador único del dispositivo final.
- **AppKey:** Application Key, es la clave maestra para generar las claves de sesión. Con esta clave se cifra la aceptación de unión.
- **AppEUI:** Application EUI, es el identificador de la aplicación en la red.

El dispositivo final envía un mensaje de solicitud de unión al servidor que incluye los identificadores DevEUI y AppEUI, el servidor que también tiene registrado el mismo DevEUI y AppEUI responde con un mensaje de aceptación, que incluye las claves: DevAddr, NwkSKey y AppSKey generados dinámicamente. Para este punto el dispositivo final ya se encuentra dentro de la red y es capaz de enviar y recibir datos.

Las ventajas de OTAA sobre ABP son claras, tiene mayor seguridad ya que las claves de sesión se generan dinámicamente, si el nodo pierde conectividad y se sale de la red tiene que realizar nuevamente un proceso de unión donde se generen nuevas claves y recién se le permite ingresar nuevamente a la red.

LoRaWAN opera bajo un esquema de acceso al medio ALOHA puro, donde los nodos transmiten sin coordinación previa. Esto puede generar riesgos de:

- **Colisión de paquetes:** Cuando transmisiones simultáneas ocurren en el mismo canal y SF, lo que puede conducir a pérdidas de paquetes, retransmisiones y aumento en los consumos. Para mitigar se recomienda limitar el ciclo de trabajo, no sobrecargar un área con demasiados nodos y planificar las ventanas de transmisión de los nodos.
- **Saturación del canal:** Cuando demasiados dispositivos comparten un mismo canal, puede conducir a la degradación de la capacidad de la red y congestión en el Gateway. Para mitigar se recomienda desplegar

Gateways adicionales, habilitar ADR y usar los múltiples canales disponibles.

2.10. Seguridad de la red completa

Los datos recolectados viajarán a través de internet hasta llegar al usuario final, es por esta razón que se debe garantizar la confidencialidad, integridad y disponibilidad (CIA) de la información.

- **Confidencialidad:** Asegura que la información solo sea accesible por personal autorizado, sistemas o procesos autorizados. Así se asegura la privacidad de los datos.

Se usan protocolos seguros como TLS (Transport Layer Secure), además MQTT usa certificados TLS/SSL para autenticación de clientes.

- **Integridad:** Se asegura que la información no sea alterada o modificada. Así se asegura que los datos lleguen sin cambios desde el emisor al receptor.

TLS/SSL incluye mecanismos de integridad basados en MAC (Message Authentication Codes) para verificar que los mensajes no fueron alterados.

- **Disponibilidad:** Que la información y los sistemas estén accesibles para usuarios o aplicaciones autorizadas. Así se evita interrupciones en el acceso a datos o servicios.

El protocolo MQTT soporta QoS (Quality of Service) que le entrega garantías para la entrega de mensajes.

2.11. Sistema de medición de agua

El agua es un recurso imprescindible para la sociedad, la gestión y optimización eficiente de este recurso hídrico es esencial en ciudades modernas que enfrentan retos de escasez y aumento de la demanda. Con una correcta instrumentación y monitoreo se permite optimizar el uso del recurso al realizar detección temprana de fugas y anomalías en los consumos, además de contribuir a la sostenibilidad del recurso. Una medición precisa conduce a una facturación justa, basadas en el consumo real y para las empresas que prestan el servicio le proporciona datos valiosos para planificar y desarrollar estrategias a largo plazo.

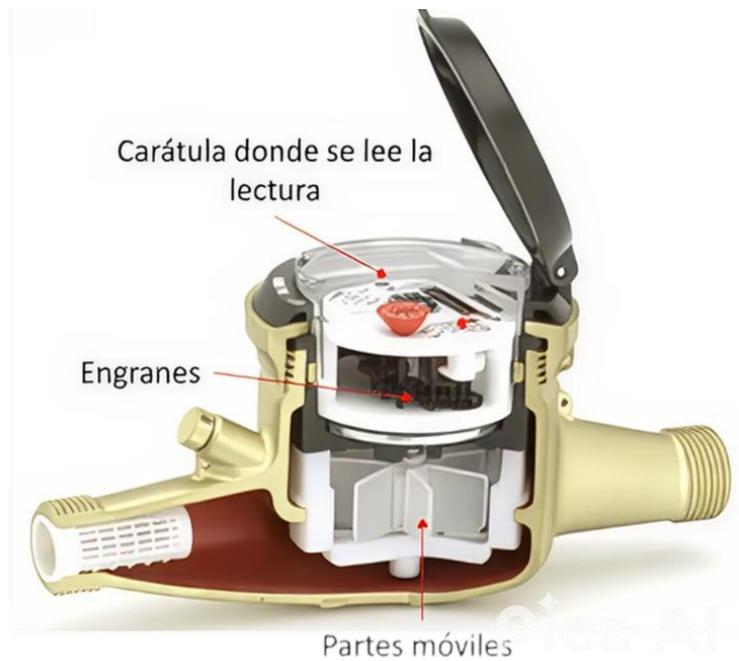
2.11.1. Métodos de medición de agua potable

Tradicionalmente la medición del agua consumida es realizada mediante medidores o también llamados contadores, su función es calcular el volumen de agua que pasa a través de él, con el propósito de monitorear el gasto del recurso y facturar su consumo. Se usan comúnmente en tuberías de la red de suministro de agua en instalaciones residenciales e industriales. A continuación, mencionaremos algunos tipos de medidores:

- **Medidores mecánicos:** Son los tradicionales y que llevan más tiempo en funcionamiento, son dispositivos simples y más económicos, además que no requieren energía eléctrica para su funcionamiento. Sus principales desventajas radican en que son menos precisos y requieren lectura manual. Estos medidores funcionan mediante unas paletas que se mueven según el caudal, el giro de la turbina se transmite a un eje y engranajes hasta un cabezal que acumula en su contador el volumen de agua que ha circulado.

Figura 11

Vista interior de un medidor mecánico.



Fuente: <https://cicasa.com/como-elegir-un-medidor/>

- **Medidores electrónicos:** Son dispositivos más modernos que los medidores mecánicos, utilizan sensores para medir el flujo de agua y registran los datos de manera digital. Realizan lecturas más precisas y permiten la lectura remota, tiene el inconveniente de ser más costosas y requerir de una fuente de energía para su funcionamiento.

Figura 12

Medidor electrónico.



Fuente: <https://es.younio.com/news-show-1001417.html>

- **Medidores ultrasónicos:** Miden el flujo de agua utilizando ondas ultrasónicas, son medidores muy precisos, confiables al no tener partes móviles y son de larga vida útil. La principal desventaja es su elevado costo y usado en su mayoría en entornos industriales.

Figura 13

Medidor ultrasónico.



Fuente: <https://flujometros-caudalímetros.com/flujometros-ultrasonicos/caudalímetro-doppler/>

2.11.2. Integración en el internet de las cosas – IoT

Teniendo claro la importancia del recurso hídrico y conociendo el concepto de IoT con sus potenciales beneficios, la integración de ambos temas da como resultado conceptos como medición inteligente “Smart Metering”. Donde la integración de medidores de agua equipados con sensores y módulos de comunicación desplegados en un área geográfica amplia, se conectan a una infraestructura red de para la recolección, procesamiento y análisis de datos.

Una gestión inteligente del agua representa un avance significativo al conectar los medidores a internet, esta integración no solo beneficia a los usuarios y a los proveedores de agua, además suma al esfuerzo de sostenibilidad del medio ambiente y al desarrollo de ciudades modernas.

2.11.3. Smart Metering y la transformación de datos en información útil

Con Smart Metering se genera una cantidad masiva de datos (Datasets) que posteriormente deben ser procesados y analizados, esto constituye una revolución en la gestión del agua transformando los datos crudos en inteligencia operativa. Esta transformación digital conduce a la gestión eficiente del recurso y la gobernanza basada en datos, proveyendo potenciales beneficios para las empresas prestadoras del servicio y los clientes finales.

Con la aparición de herramientas disruptivas como es la Inteligencia Artificial IA se abren nuevas posibilidades para el manejo de datos, como el uso de algoritmos de IA que son capaces de analizar vastos volúmenes de información y devolver predicciones futuras o tendencias. Como, por ejemplo:

- Automatización de tareas y procesos complejos.
- Predicción de la demanda: Analizando los patrones de consumo diario de los clientes finales, la empresa puede prever picos y valles de demanda.
- Optimización de inversiones en infraestructura mediante simulaciones basadas en datos.
- Detección de anomalías: Identificando consumos inusuales que pueden ser por causa de fugas, rotura de tuberías.
- Eficiencia operativa: La empresa puede monitorear continuamente la red de agua sin intervención continua de un personal.

CAPITULO III: DISEÑO E IMPLEMENTACIÓN DEL PROTOTIPO

3.1. Introducción

El capítulo se centra en el diseño e implementación de una red IoT basada en LoraWAN, se iniciará con la búsqueda y comparación detallada de los componentes apropiados de la red para lograr alcanzar los objetivos planteados. Se describen los componentes claves que hacen que una red IoT trabaje de manera adecuada, esto incluirá el nodo sensor, el Gateway, la infraestructura de comunicación basada en LoraWAN y la parte de interacción humana con la red, que es el componente de visualización y monitoreo de la red.

Seguidamente, se abordarán las pruebas preliminares, como la calibración del sensor de flujo, las pruebas de la infraestructura de red, además de pruebas con la plataforma de visualización y monitoreo para asegurar su correcta integración con el prototipo final.

Finalmente, teniendo superada las pruebas preliminares se procede con la integración final de todos los componentes de la red, teniendo como resultado un sistema IoT que garantice funcionalidad y eficiencia en el proceso de lectura automatizada de los medidores de agua domiciliaria.

3.2. Elección de componentes del sistema IoT

3.2.1. Nodo final

El nodo sensor está compuesto por un microcontrolador, con la función de captura y procesamiento de los datos recolectados por el sensor, además de un módulo de comunicación a cargo de la transmisión de los datos. Entre las principales características buscadas en un nodo son las siguientes:

- **Bajo consumo energético:** Es una característica clave, lograr operar con consumos sumamente bajos se traduce en autonomías altas. Los nodos sensores además deben ser portables lo que implica el uso de baterías integradas.

- Transmisiones de largo alcance: La modulación Lora permite cubrir grandes distancias, en el orden de kilómetros usando bandas de frecuencias específicas. Esto permite usarlo en aplicaciones donde los nodos están distribuidos en áreas extensas.
- Captura de datos en tiempo real: En una red inteligente es necesario tener información actualizada del fenómeno a sensar.
- Diseño robusto y con protección exterior: Debe estar diseñado para operar en condiciones ambientales adversas, mediremos el flujo de agua por tanto se debe esperar protección contra el agua, humedad, etc.

Luego de realizar una búsqueda detallada en línea se compararon los siguientes nodos finales:

Tabla 2

Comparación de características de los nodos propuestos.

NODOS FINALES			
CARACTERISTICAS	DRAGINO SW3L	TTGO-TBEAM	LXSY 15 – LXSY 300
Chip Lora	SX1262 LoRa Core	SX1276 4MB	-
Frecuencia de operación	915 – 923 MHz	915 – 923 MHz	433 – 923 MHz
Clase de protección	IP68	Sin protección	IP68
Clase	A	A y C	B
Duración de batería	Aproximadamente 10 años	Alimentación externa	Aproximadamente 6 años
Potencia de transmisión	22 dBm	20dBm	19.5 dBm
Precio	S/. 280	S/. 150	S/. 270

Fuente: Elaboración propia.

Se eligió el módulo Dragino SW3L, teniendo gran relevancia las siguientes características: el módulo tiene integrado un chip compatible con LoraWAN, el módulo viene precargado con unas claves únicas para el registro en el servidor donde, después de su encendido el módulo se conecta automáticamente. Además, se tomó en cuenta que el módulo tiene una gran potencia de transmisión y una alta sensibilidad, mayor a los otros dos módulos comparados y posee una batería de grado industrial que le otorga una gran autonomía ideal para aplicaciones IoT. Además, el fabricante Dragino ofrece una amplia bibliografía de sus productos lo que facilita la implementación del prototipo.

Figura 14

Nodo final Dragino SW3L.



Fuente: Sensor de flujo exterior SW3L (Dragino, s.f.).

3.2.2. Sensor

El sensor proporciona los datos que posteriormente serán analizados, visualizados o monitorizados. Las características buscadas en un sensor de flujo de agua son las siguientes:

- Precisión del sensor: Se busca un sensor que mida el volumen de agua que pasa a través de una tubería de 1/2", el sensor debe contar con una precisión aceptable que genere errores aceptables.
- Bajo consumo energético: El sensor será alimentado con la misma batería del nodo final. Por tanto, se busca un sensor que tenga consumos bajos adecuados para dispositivos IOT.
- El sensor debe soportar una presión de agua adecuada para aplicaciones de medición de agua en entornos residenciales.

Luego de realizar una búsqueda detallada se compararon los siguientes sensores:

Tabla 3

Comparación de características de sensores propuestos.

SENSORES		
CARACTERISTICAS	DW-004	YF-S201
Tipo de sensor	De flujo	De flujo
Diámetro	1/2"	1/2"
Rango de trabajo	1 – 30 L/min	1 – 30 L/min
Pulsos por litro	450	450
Presión de trabajo	1.75MPa	1.75MPa
Precisión	+5%	+10%
Rango de temperatura	< 80°C	-25 a 80 °C
Consumo de energía	1uA, 3.6V	10mA, 5V

Fuente: Elaboración propia.

El sensor YF-S201 es un sensor de flujo bastante conocido y de fácil acceso, pero entre sus limitaciones está que funciona a 5V, está pensado para usarlo con placas como Arduino y su precisión es inferior.

El sensor DW-004 viene integrado con el módulo Dragino SW3L, se observa que tiene características superiores respecto al sensor YF-S201, en cuanto a la precisión y al consumo energético que es de suma importancia en un módulo IoT alimentado por batería, es por esta razón que se decidió usar este sensor para el proyecto.

Figura 15

Nodo final con su sensor de flujo integrado DW-004.



Fuente: Sensor de flujo exterior SW3L (Dragino, s.f.).

3.2.3. Gateway LoraWAN

Un Gateway es el puente de comunicación entre el nodo final y el servidor de red. Su tarea fundamental es la recepción y demodulación de las señales de radio transmitida por el nodo final y retransmitirlos hacia el servidor de red mediante protocolos estándar como es Ethernet, Wi-Fi o redes móviles (3G, 4G, 5G).

Tabla 4

Comparación de características de los Gateways propuestos.

GATEWAYS LORAWAN			
CARACTERISTICAS	RAK7268	RAK831 Pilot Gateway	Dragino LG01
Chip usado	Semtech SX1302 (soporte completo de LoraWAN V1.0.3)	Semtech SX1301(soporte para LoraWAN V1.0.2)	Semtech SX1262
Numero de canales soportados	8 canales	8 canales	8 canales
Bandas de operación	US915 – AU915	US915 – AU915	US915 – AU915
Tipos de conexiones soportadas	Wi-Fi (2.4 GHz), Ethernet, LTE	Ethernet	Wi-Fi (2.4 GHz), Ethernet, LTE
Fuente de alimentación	Fuente DC 12V – 1A, admite PoE	Fuente DC 5V – 2.5A	Fuente DC 5V – 2A
Antena	LoRa: Conector hembra RP-SMA	Conector macho SMA	LoRa: Conector hembra RP-SMA
Potencia de transmisión	27 dBm (Max)	27 dBm (Max)	20 dBm
Sensibilidad	-139 dBm (Min)	-142.5 dBm (Min)	-148 dBm
Clases	A y C	A y C	A y C
Protección	IP30	-	-
Temperatura de funcionamiento	-10 a 55 °C	0 a 70°C	-20 a 65°C
Consumo de energía	12W	12.5W	10W

Fuente: Elaboración propia.

El modelo escogido fue el RAK7268 WisGate Edge Lite, ya que el laboratorio LISTTI cuenta con este Gateway, las características del Gateway son suficientes para los objetivos buscados en la tesis.

Figura 16

WisGate RAK7268.



Fuente: <https://www.rakwireless.com/en-us/products/lpwan-gateways-and-concentrators/rak7268-wisgate-edge-lite-2>

3.2.4. Servidor de red

El servidor de red tiene como función la gestión, procesamiento y enrutamiento de la información proveniente del nodo final en ambas direcciones: Uplink y Downlink. El servidor de red se encarga de filtrar datos, autenticar nodos previamente registrados, asegura la integridad de la red, administra los dispositivos y seguridad de las comunicaciones al codificar y decodificar los mensajes a un formato legible para la aplicación final.

Tabla 5

Comparación de características de servidores propuestos.

SERVIDORES DE RED LORAWAN		
CARACTERISTICAS	The Things Network	ChirpStack
Tipo de plataforma	Servicio basado en la nube	Plataforma local
Tipo de red	Red pública, red privada	Red privada
Clases de dispositivos soportados	A, B y C	A, B y C
Versiones soportadas de LoraWAN	V1.0 y V1.1	V1.0 y V1.1
Soporte para integraciones	MQTT, gRPC y HTTP	MQTT y HTTP
Seguridad	Cifrado de extremo a extremo.	Cifrado de extremo a extremo.
Licencia	Software libre	Software libre
Numero de Gateways y nodos	Limitado para la versión gratuita	Ilimitado

Fuente: Elaboración propia.

Después de detallar las características principales de los servidores de red propuestos en la tabla 5, se eligió la plataforma The Things Networks (TTN) ya que es ideal para proyectos pequeños o medianos, su infraestructura en la nube simplifica el trabajo al no requerir un control completo de la red, como se hace en ChirpStack donde es una plataforma local y totalmente privada. TTN cumple ambas funciones: es servidor de red y también servidor de aplicaciones.

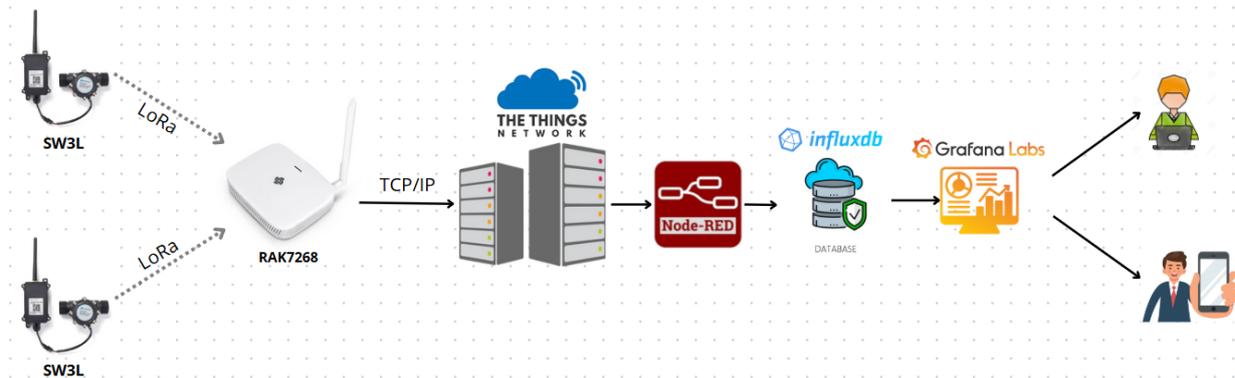
Además, permite la integración con plataformas externas por medio de API Keys para almacenar los datos recolectados en un base de datos y la integración con plataformas de visualización de datos.

3.3. Arquitectura general de la red IoT

La red IoT será de topología estrella, con los elementos de red elegidos se procede con el diagrama general de la red propuesta, mostrada a continuación:

Figura 17

Arquitectura general de la red IoT propuesta.



Fuente: Elaboración propia.

3.4. Configuración del nodo final

El nodo final Dragino SW3L viene precargado de fábrica con un conjunto de claves únicas para el registro en el servidor de red, estas claves se deben mantener seguras y accesibles únicamente por los propietarios.

Figura 18

Claves únicas de cada nodo sensor.

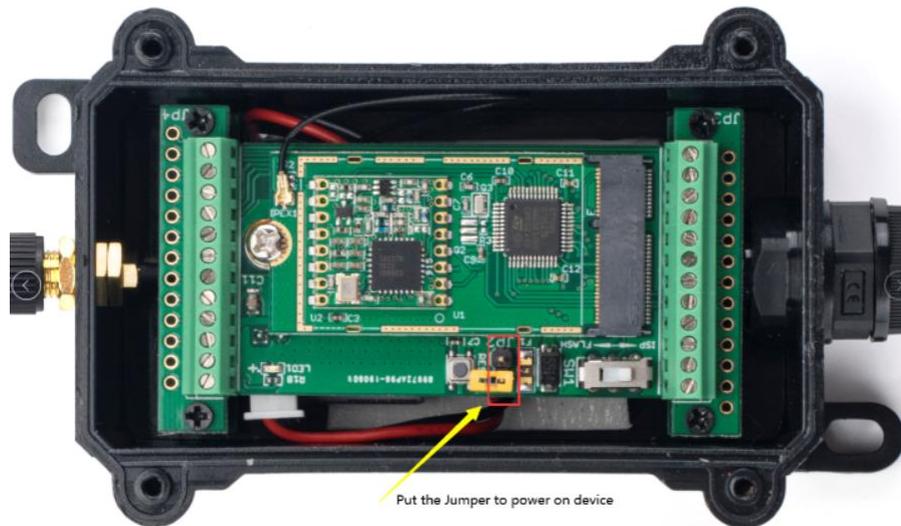


Fuente: Elaboración propia.

El nodo viene desconectado de su batería, por lo que es necesario sacar los tornillos que aseguran la tapa exterior y conectar el jumper de color amarillo para alimentar la placa LSN50V2 que es el corazón del nodo sensor, asegurándose de haber conectado la antena al nodo final antes de energizar el equipo. Ahora el nodo final estará listo para unirse a la red.

Figura 19

Jumper para energizar el nodo sensor.



Fuente: Fuente: Sensor de flujo exterior SW3L (Dragino, s.f.).

3.5. Configuración del Gateway

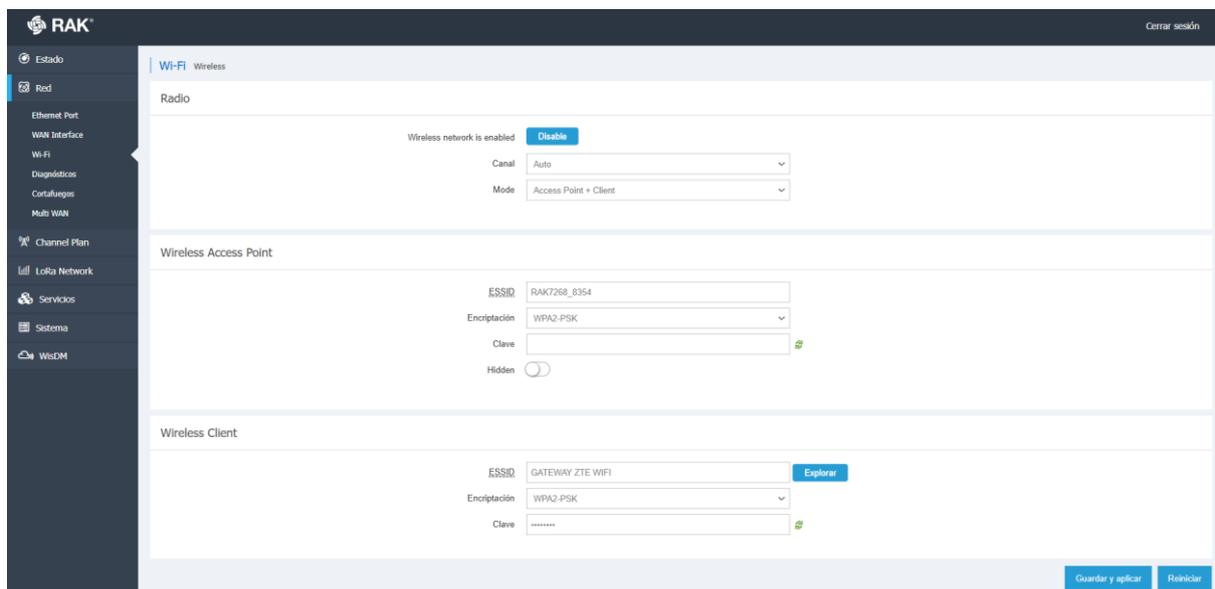
Para la configuración del Gateway RAK7268 primero se debe conectarse a la interfaz web proporcionada por el fabricante, asegurarse de conectar primero la antena al conector RP-SMA del Gateway antes de energizar el equipo, esto por recomendación del fabricante. Para acceder al Gateway existen dos alternativas: Modo AP-WiFi y Puerto WAN Ethernet.

Se escogió el método de AP-WiFi, lo que significa que el Gateway funcionará en modo Punto de acceso (Acces Point-AP Wi-Fi), se puede encontrar un SSID con nombre RAK7268_XXXX en la lista de redes Wi-Fi de su PC. XXXX son los dos últimos bytes de la dirección MAC del Gateway. Mediante un navegador web se dirige a la dirección IP: 192.168.230.1 y las credenciales por defecto son "root" tanto para Username y Password.

Una vez dentro, ingresamos al campo "Red" > "WiFi", configuramos el Gateway en modo "Access Point + Client" y llenamos las credenciales para que el Gateway tenga acceso a internet.

Figura 20

Acceso a internet del Gateway.

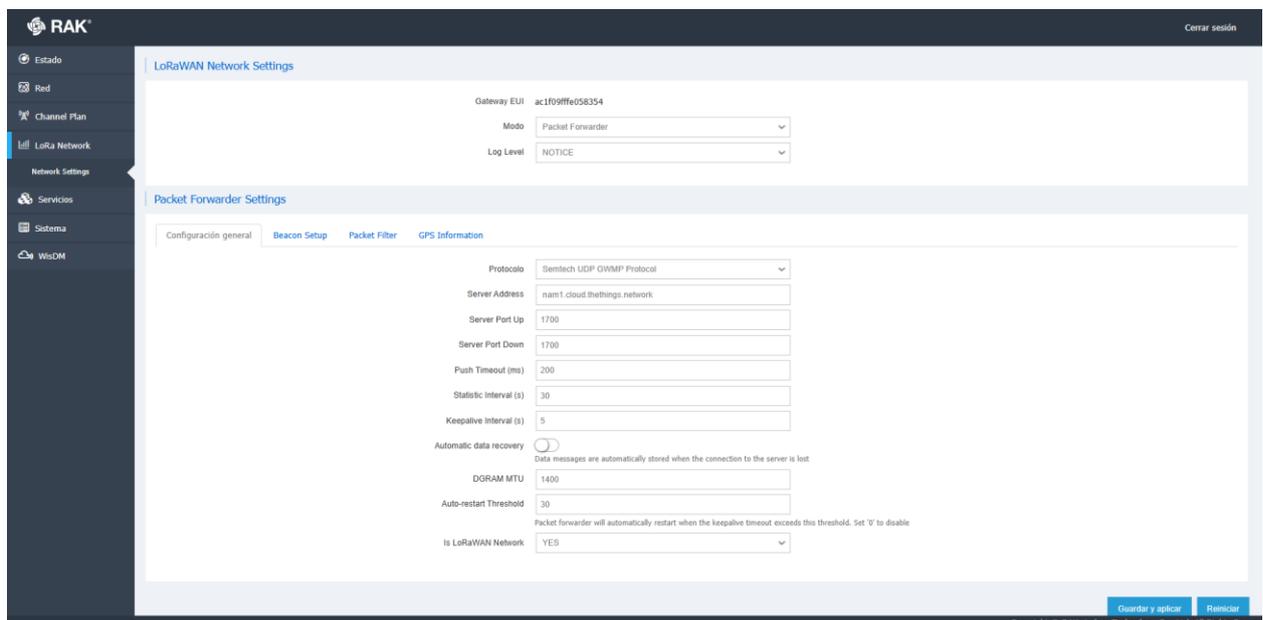


Fuente: Elaboración propia.

En el campo “LoRa Network” > “Network Settings” se configura el Gateway en modo “Packet Forwarder”, dentro de “Packet Forwarder Settings” > “Protocol” > Semtech UDP GWMP Protocol y el “Server Address” es la dirección del Servidor de Red a donde se reenviarán los paquetes Lora, se realizará la configuración más adelante en el apartado de Registro del Gateway.

Figura 21

Configuración del Gateway como Packet Forwarder.



Fuente: Elaboración propia.

Finalmente configuramos el plan de frecuencias a utilizar, según el Ministerio de Transportes y Comunicaciones (MTC) indica a la banda 915-928MHz como frecuencia no licenciada, Lora Alliance en sus parámetros regionales recomienda que Perú use las bandas AU915-928 y AS923. En la presente tesis se hará uso de la banda AU915-928.

Figura 22

Parámetros regionales definidos por Lora Alliance.



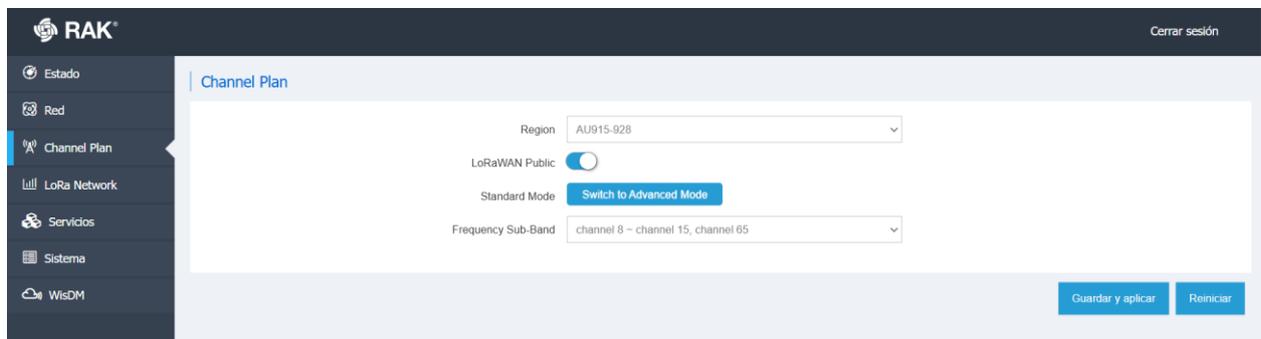
LoRaWAN 1.0.3 Regional Parameters

Country name	Band / channels	Channel Plan
Paraguay	433.05 - 434.79 MHz	EU433
	915 - 928 MHz	AU915-928, AS923
Peru	915 - 928 MHz	AU915-928, AS923
Papua New Guinea	915 - 925 MHz	AU915-928

Fuente: LoRaWAN regional parameters (Lora Alliance, s.f.)

Figura 23

Configuración de la banda AU915-928 del Gateway.



Fuente: Elaboración propia.

Ahora el Gateway está listo para el registro dentro de la plataforma del servidor LoraWAN.

3.6. Configuración del Servidor LoraWAN

Iniciaremos con la configuración del Servidor LoraWAN, como mencionamos antes se eligió la plataforma The Things Networks (TTN) que usa la plataforma The Things Stack (TTS) como plataforma central, dentro de TTS existen diferentes versiones; para la presente tesis se escogió The Things Stack Community Edition (TTN Sandbox) ya que es gratuita y abierta a la comunidad. Se escoge el “Cluster Nam1” recomendado para América Latina, que es la ubicación de los servidores donde se procesarán los datos.

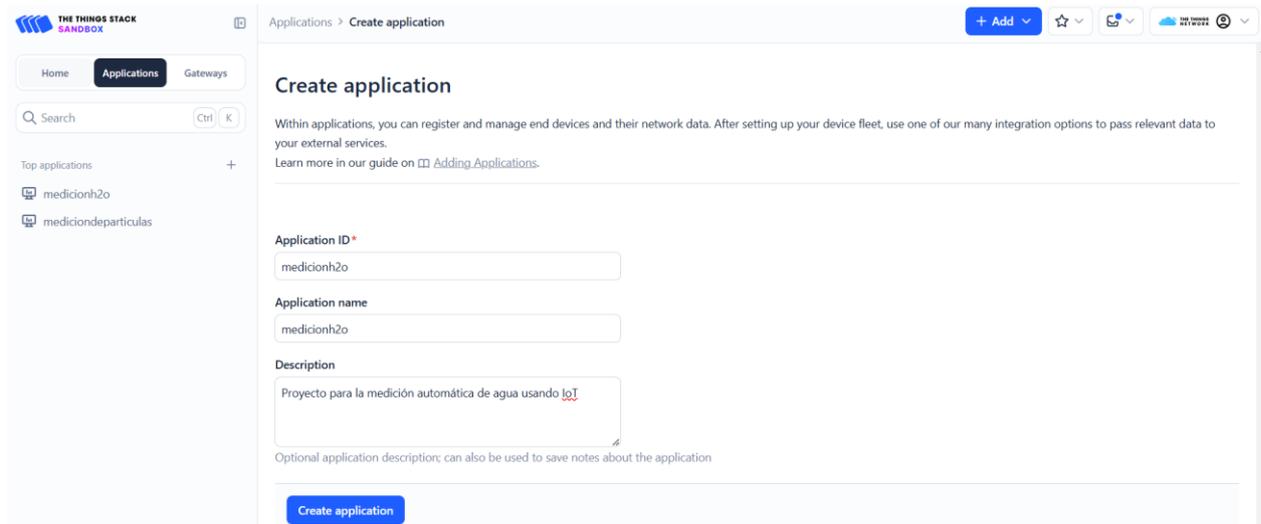
Iniciamos con la creación de una Aplicación, una aplicación es un contenedor que organiza, administra los dispositivos finales registrados y sus datos, es recomendable que cada proyecto tenga su propia aplicación ya que al utilizar servicios externos (como base de datos, dashboards, etc.) solo se envíen los datos relacionados al proyecto en específico. Luego procedemos a registrar el dispositivo final que pertenecerá a la aplicación creada y finalmente registramos el Gateway que es de uso general así que no se registra dentro de ninguna aplicación.

3.6.1. Crear aplicación

En la pestaña “Home” > “Create Application” y rellenamos los campos requeridos

Figura 24

Creación de una aplicación para el proyecto.



The screenshot shows the 'Create application' interface in The Things Stack Sandbox. The left sidebar contains navigation tabs for 'Home', 'Applications', and 'Gateways', with 'Applications' selected. Below the tabs is a search bar and a list of 'Top applications' including 'medicionh2o' and 'mediciondeparticulas'. The main content area is titled 'Create application' and contains the following fields:

- Application ID***: A text input field containing 'medicionh2o'.
- Application name**: A text input field containing 'medicionh2o'.
- Description**: A text area containing 'Proyecto para la medición automática de agua usando IoT'.

Below the description field, there is a note: 'Optional application description; can also be used to save notes about the application'. At the bottom of the form is a blue 'Create application' button.

Fuente: Elaboración propia.

3.6.2. Registrar nodo final

En la pestaña “Home” > “Add End Device” > Elegimos la aplicación (medicionh2o) y rellenamos los campos requeridos. Se escoge un plan de frecuencia a utilizar, la versión de LoraWAN usada por el dispositivo final y el modo de activación OTAA. El campo JoinEUI es equivalente a AppEUI.

Figura 25

Registro del nodo final en la plataforma The Things Network.

Register end device [Show advanced activation, LoRaWAN class and cluster settings](#)

Does your end device have a LoRaWAN® Device Identification QR Code? Scan it to speed up onboarding.

[Scan end device QR code](#) [Device registration help](#)

End device type

Input method Select the end device in the LoRaWAN Device Repository Enter end device specifics manually

Frequency plan

LoRaWAN version

Regional Parameters version

Provisioning information

JoinEUI

This end device can be registered on the network.

DevEUI 2/50 used

AppKey

End device ID

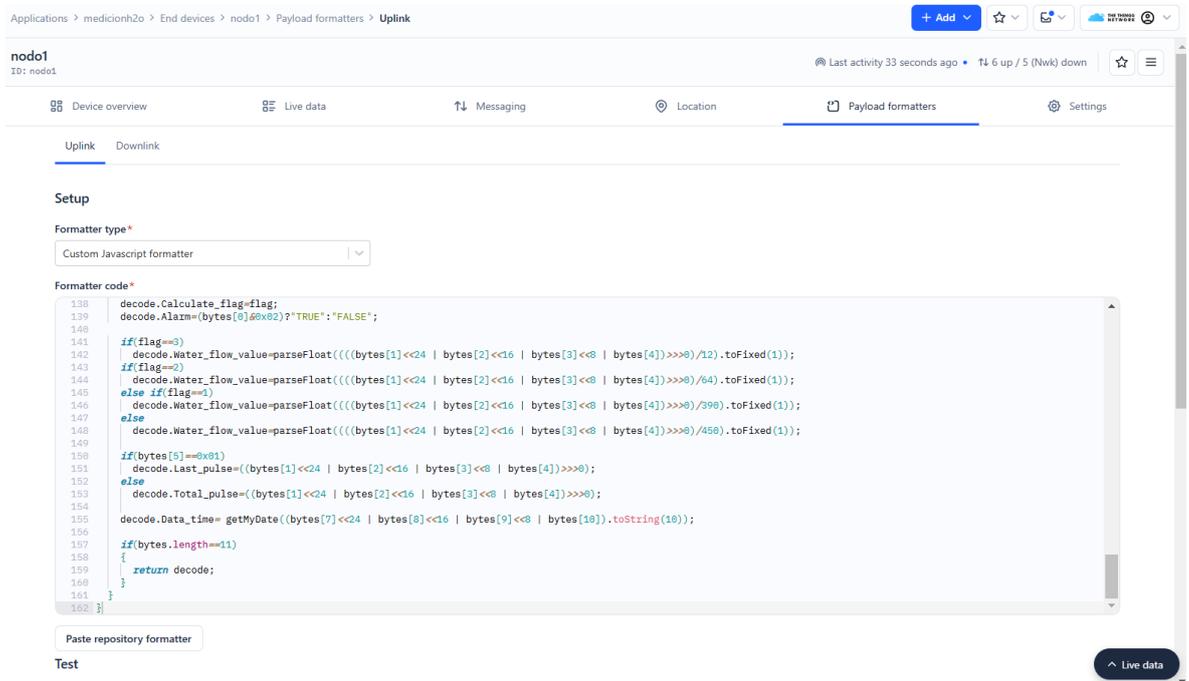
After registration View registered end device Register another end device of this type

Fuente: Elaboración propia.

El Dragino SW3L requiere un decodificador para interpretar los datos crudos (payload) enviados y convertirlo a un formato entendible por el usuario. El decoder está escrito en lenguaje Javascript, para implementarlo nos dirigimos a “End devices” > “nodo1” > “Payload formatters” > “Uplink” > “Formatter type” > “Custom Javascript Formatter” y copiamos el código decodificador modificado, el código está disponible en el **Anexo 1**.

Figura 26

Decoder usado para decodificar el payload enviado por el nodo sensor.



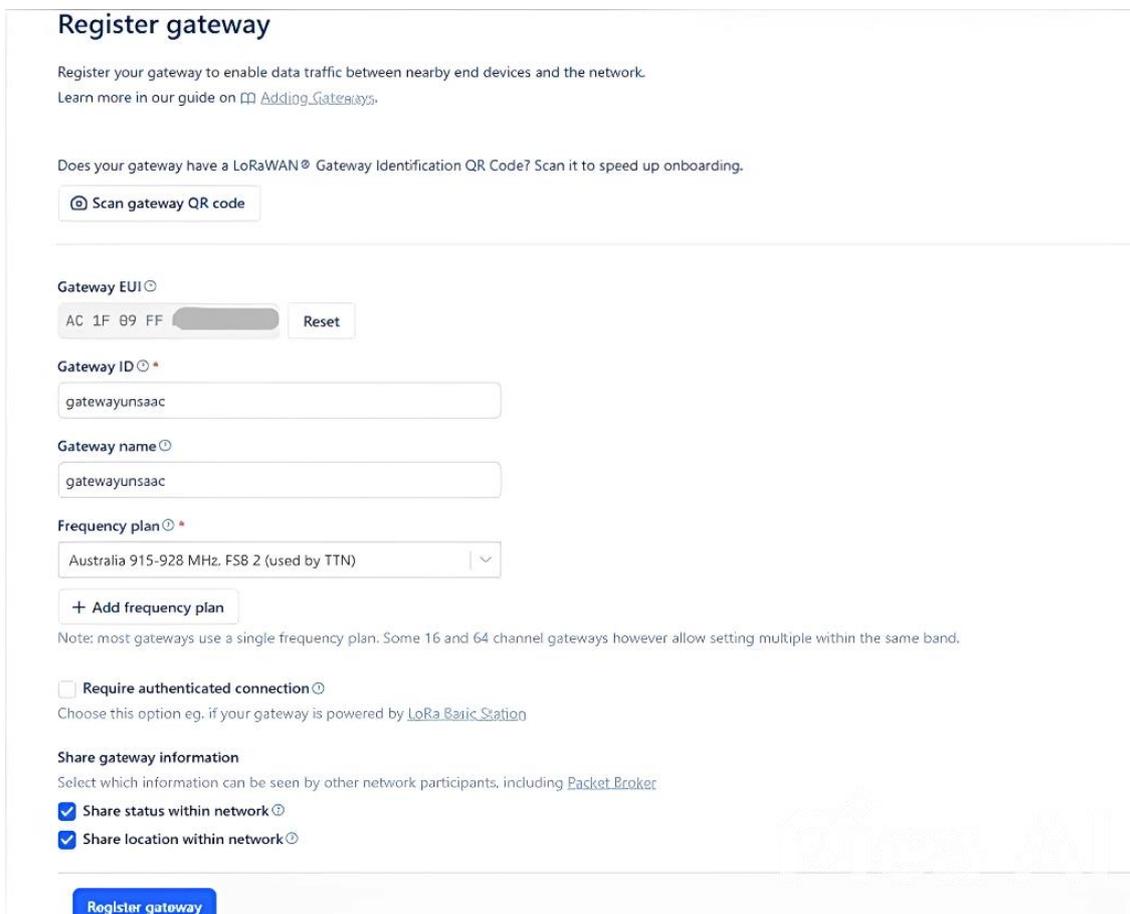
Fuente: Elaboración propia.

3.6.3. Registrar Gateway

En la pestaña "Home" > "Register gateway" y rellenamos los campos requeridos.

Figura 27

Registro del Gateway en la plataforma The Things Network.



The screenshot shows the 'Register gateway' form in The Things Network. The form includes the following fields and options:

- Gateway EUI:** A text input field containing 'AC 1F 89 FF' and a 'Reset' button.
- Gateway ID:** A text input field containing 'gatewayunsaac'.
- Gateway name:** A text input field containing 'gatewayunsaac'.
- Frequency plan:** A dropdown menu set to 'Australia 915-928 MHz, FS8 2 (used by TTN)' with a '+ Add frequency plan' button below it.
- Require authenticated connection:** A checkbox that is currently unchecked.
- Share gateway information:** Two checked checkboxes: 'Share status within network' and 'Share location within network'.

A blue 'Register gateway' button is located at the bottom of the form.

Fuente: Elaboración propia.

Ya registrado el Gateway debemos crear una APIKey que permitirá a dispositivos externos (en este caso el Gateway RAK7268) unirse e interactuar con la plataforma The Things Network. Nos dirigimos a nuestro Gateway registrado (gatewayunsaac) > "API keys" > "Add API key" y concedemos los permisos necesarios. Obtenida la API key, registramos la clave en la plataforma web del Gateway RAK7268 mostrado anteriormente.

Figura 28

APIkey creada.

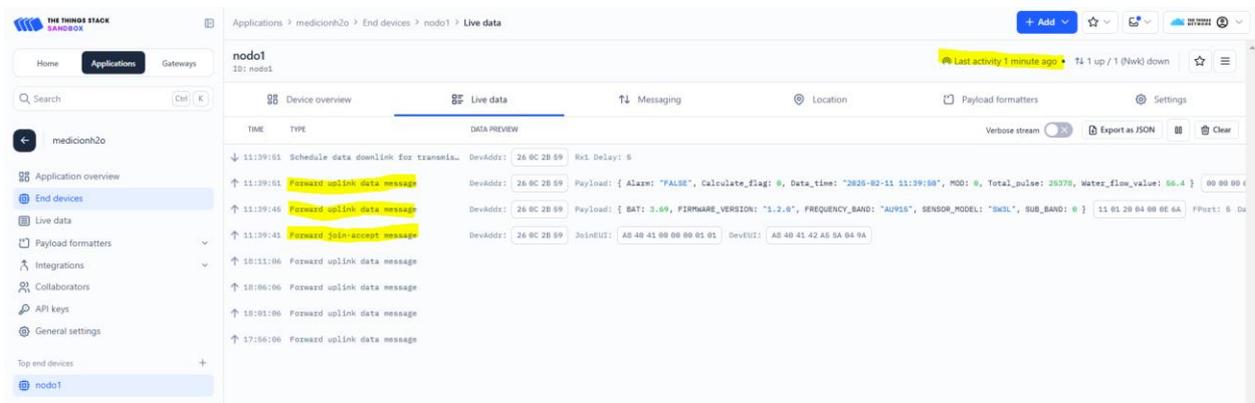


Fuente: Elaboración propia.

Una vez registrados el Gateway y el nodo final, observamos en la Figura 29 que se realiza el procedimiento de unión (Join Procedure) satisfactoriamente (Join Accept), ahora el nodo final puede enviar y recibir mensajes al Servidor LoraWAN, se observa que el servidor empieza a recibir los primeros mensajes provenientes del nodo sensor.

Figura 29

Nodo final se une exitosamente al servidor LoraWAN.



Fuente: Elaboración propia.

3.7. Integración con Node-RED

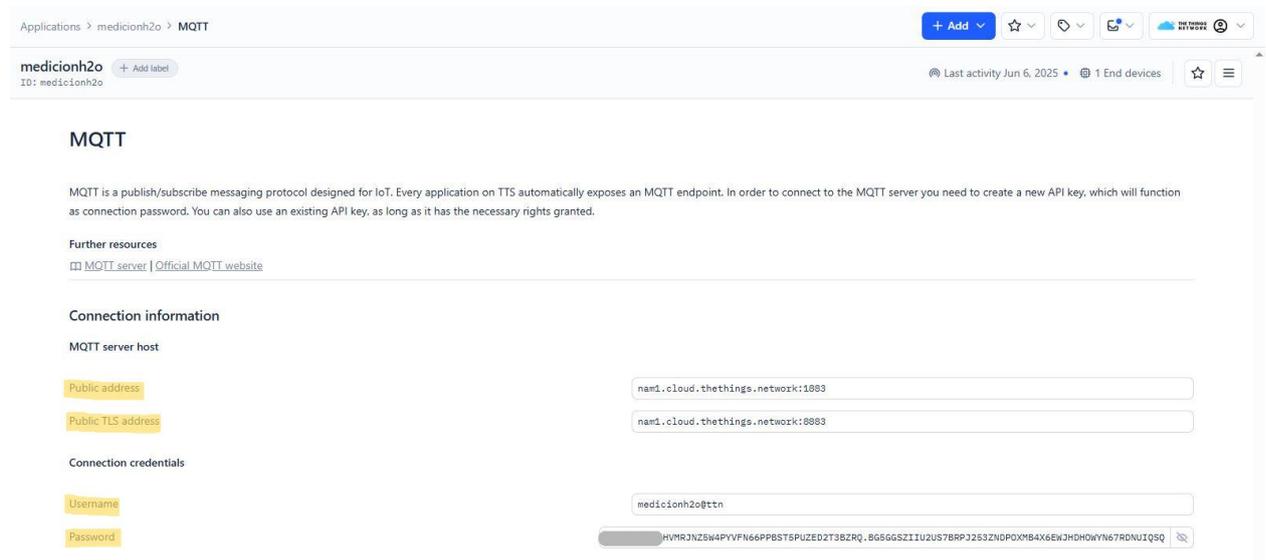
Node-RED es una herramienta de programación para conectar dispositivos de hardware, APIs y servicios en línea. Es útil en aplicaciones basadas en eventos, ofrece un editor en el navegador basado en flujos y posee una gran variedad de nodos útiles en el Internet de las cosas. Permite el procesamiento y visualización de datos, integración con servicios externos mediante APIs y automatización de tareas.

Los datos obtenidos de los nodos finales por el Servidor LoraWAN se redireccionan a Node-RED mediante una conexión MQTT. The Things Network ofrece un broker MQTT que permite suscribirse a los datos de los dispositivos en tiempo real.

En TTN nos dirigimos a nuestra aplicación y en el campo “Applications” > “Integrations” > “MQTT” obtenemos las credenciales el MQTT server host, estos datos son útiles para conectarse a Node-RED. En Node-RED usando el nodo “mqtt in” permite suscribirse a un bróker.

Figura 30

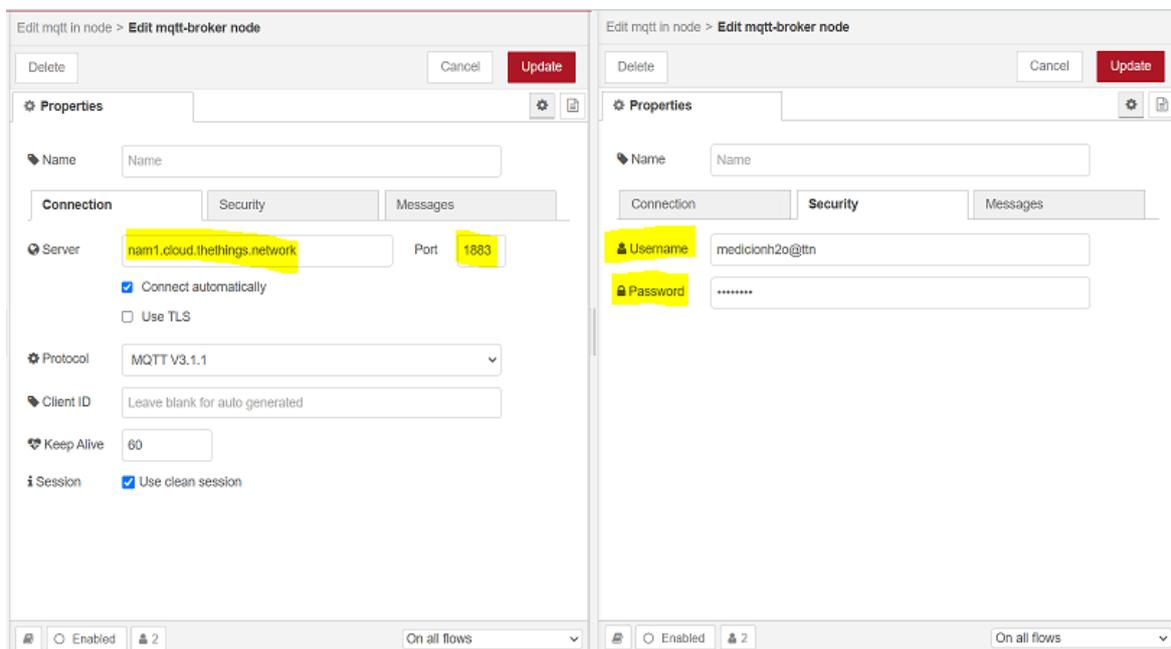
Credenciales del broker MQTT de TTN.



Fuente: Elaboración propia.

Figura 31

Suscripción al broker de TTN desde NodeRed.



Fuente: Elaboración propia.

Figura 32

Conexión exitosa al broker de TTN.

```
11 Feb 13:37:58 - [info] Server now running at http://127.0.0.1:1880/  
11 Feb 13:37:58 - [info] Starting flows  
11 Feb 13:37:58 - [info] Started flows  
11 Feb 13:37:59 - [info] [mqtt-broker:15f97f18bdb50a46] Connected to broker: mqtt://nam1.cloud.thethings.network:1883
```

Fuente: Elaboración propia.

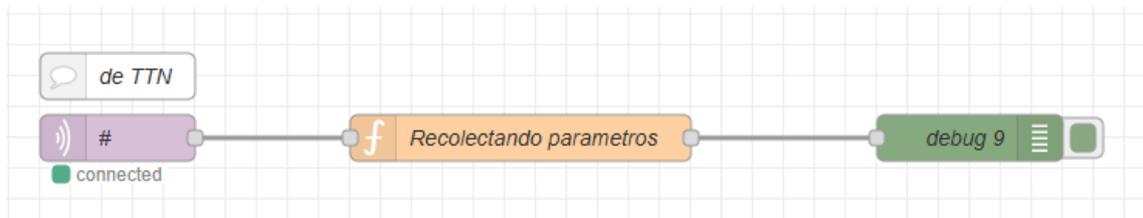
3.8. Manejo de datos y calibración del sensor

Una vez suscrito al broker de TTN, se procede con el procesamiento de datos con las herramientas que proporciona Node-RED. Utilizaremos el nodo “Function” que permite escribir y ejecutar código Javascript dentro de un flujo, es útil para decodificar datos recibidos, aplicar cálculos o filtros.

3.8.1. Recolectando parámetros

Figura 33

Flujo para la recolección de datos del bróker TTN.



Fuente: Elaboración propia.

No todos los datos enviados por el broker The Things Network son útiles para propósitos de la tesis, entonces se recolectan únicamente los parámetros que se utilizarán, recolectamos los siguientes parámetros mostrados en la Figura 34. El código en JavaScript utilizado está disponible en el **Anexo 2**.

Figura 34

Algoritmo para recolectar parámetros útiles enviados desde TTN.

```
1 // Recolectamos parametros utiles del sensor
2 // Parametros Rx
3 let device_id = msg.payload['end_device_ids']['device_id'];
4 let rssi = msg.payload['uplink_message']['rx_metadata']['0']['rssi'];
5 let snr = msg.payload['uplink_message']['rx_metadata']['0']['snr'];
6
7 let parametros = {
8   "device_id" : device_id,
9   "rssi" : rssi,
10  "snr" : snr
11 };
12
13 // Parametros del sensor de flujo
14 let dataSensor = msg.payload['uplink_message']['decoded_payload'];
15
16 // Unimos todos los parametros
17 let combined = { ...parametros, ...dataSensor};
18
19 msg.payload = combined;
20
21 return msg;
```

debug

11/2/2025, 14:39:44 node: debug 9
v3/medicionh2o@ttn/devices/nodo1/up : msg.payload :
Object

object

- device_id: "nodo1"
- rssi: -36
- snr: 13.8
- Alarm: "FALSE"
- Calculate_flag: 0
- Data_time: "2025-02-11 14:39:45"
- MOD: 0
- Total_pulse: 25378
- Water_flow_value: 56.4

Fuente: Elaboración propia.

3.8.2. Calibración empírica del sensor

Para garantizar una facturación precisa es imprescindible calibrar el sensor de flujo para conseguir mediciones precisas, para esto se realizó una prueba empírica.

En la Figura 35 se muestra el sistema implementado, donde el sensor DW-004 y un sensor de flujo YF-S201 adicional se conectaron en serie y el agua pasa a través de ellos, esto para comprobar las medidas realizadas de los sensores. Como patrón de medición se utilizó una jarra graduada para que pase un volumen de agua exacto.

Figura 35

Sistema implementado para la calibración empírica del sensor.



Fuente: Elaboración propia.

Se inicio con la toma de muestras de la Tabla 6, el módulo Dragino SW3L entrega el parámetro llamado “Total_pulse” que es la cantidad de pulsos que se genera cuando pasa un determinado volumen de agua. El sensor YF-S201 se conecta a un módulo ESP32 para la toma de datos de pulsos, el programa fue desarrollado en Arduino IDE y está disponible en el **Anexo 3**. El factor de conversión “k” se puede hallar experimentalmente, conociendo la cantidad de pulsos que marcan los sensores cuando un volumen de agua (jarra graduada con 1L) pasa a través de los sensores.

$$k = \frac{n^{\circ} \text{ pulsos}}{\text{Volumen} * 60}$$

En la tabla 6 se visualiza la toma de datos de ambos sensores, la medida patrón fue siempre de 1L.

Tabla 6

Toma de pulsos proporcionado por los sensores.

TOMA DE PULSOS DE SENSORES				
SENSOR	YF-S201		SW3L – DW004	
LITROS PATRON	PULSOS	K	PULSOS	K
1	433	7.2166	452	7.5333
1	485	8.0833	480	8.0000
1	481	8.0166	488	8.1333
1	490	8.1666	517	8.6166
1	500	8.3333	520	8.6666
1	467	7.7833	496	8.2666
1	478	7.9666	478	7.9666
1	472	7.8666	482	8.0333
1	457	7.6166	444	7.4000
1	447	7.4500	446	7.4333
1	464	7.7333	491	8.1833

Fuente: Elaboración propia.

- **Calibración del sensor YF-S201:** Para la calibración se saca una medida promedio $K_{promedio} = 7.839394$ que es el factor de conversión, teniendo este valor calibrado lo insertamos en el algoritmo para calcular el volumen de agua que pasa a través del sensor en base al número de pulsos del sensor, esto en la placa ESP32. El código está disponible en el **Anexo 4**.

Figura 36

Algoritmo para calcular el volumen que pasa a través del sensor YF-S201.

The screenshot shows the Arduino IDE interface. The main window displays the code for 'medicion_volumen2.ino'. The code includes comments in Spanish and defines variables for pulse counting and volume calculation. The serial monitor at the bottom shows the output of the program, which consists of two columns of data: 'Caudal: 0.000 L/min' and 'Volumen: 49.470 L', repeated multiple times.

```

1 // Mide el volumen de agua que pasa a travez del sensor YF-s201
2
3 volatile int NumPulsos = 0; // Variable para la cantidad de pulsos recibidos
4 const int PinSensor = 12; // Pin del sensor YF-S201 conectado al GPIO12 del ESP32
5 const float factor_conversion = 7.839394; // Factor hallado experimentalmente
6 float volumen = 0;
7 unsigned long t0 = 0; // Tiempo anterior en milisegundos
8
9 // Función que se ejecuta en interrupción
10 void ContarPulsos() {
11     NumPulsos++; // Incrementamos la variable de pulsos
12 }
13
14 // Función para obtener la frecuencia de los pulsos
15 float ObtenerFrecuencia() {
16     int pulsos;
17     unsigned long tiempoInicio = millis();
18     NumPulsos = 0; // Reiniciamos el contador de pulsos
19     interrupts(); // Habilitamos las interrupciones
20     delay(1000); // Esperamos 1 segundo
21     noInterrupts(); // Deshabilitamos las interrupciones
22     pulsos = NumPulsos;
23 }

```

Serial Monitor Output:

```

-----
Caudal: 0.000 L/min      Volumen: 49.470 L
-----

```

Fuente: Elaboración propia.

- **Calibración del módulo Dragino SW3L:** Según la documentación del nodo Dragino SW3L, indica que 1L es equivalente a 450 pulsos (Dragino,

s.f.), pero como se observa en la Tabla 6 esto no ocurre en la práctica, es por esta razón que se realiza la calibración de las mediciones. Los pulsos promedio por litro es: $P_{promedio} = 481.272727$ [pulsos/litro]

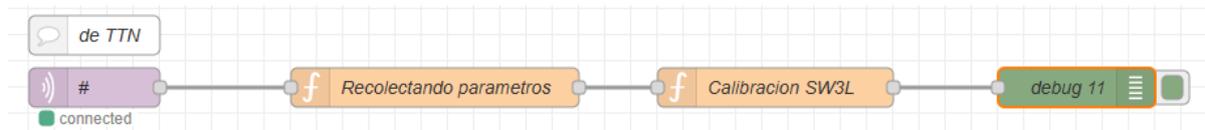
Para hallar el volumen consumido se usa la fórmula:

$$Volumen (L) = \frac{Pulsos\ Totales}{P_{promedio}(pulsos/L)}$$

Sabiendo cómo calcular el volumen consumido real, se crea la variable "Water_flow_value_calibrado" que es la medición en Litros del volumen de agua que pasa a través del sensor DW-004 después de la calibración, se implementó en un nodo "Function" en Node-RED, el código está disponible en el **Anexo 5**.

Figura 37

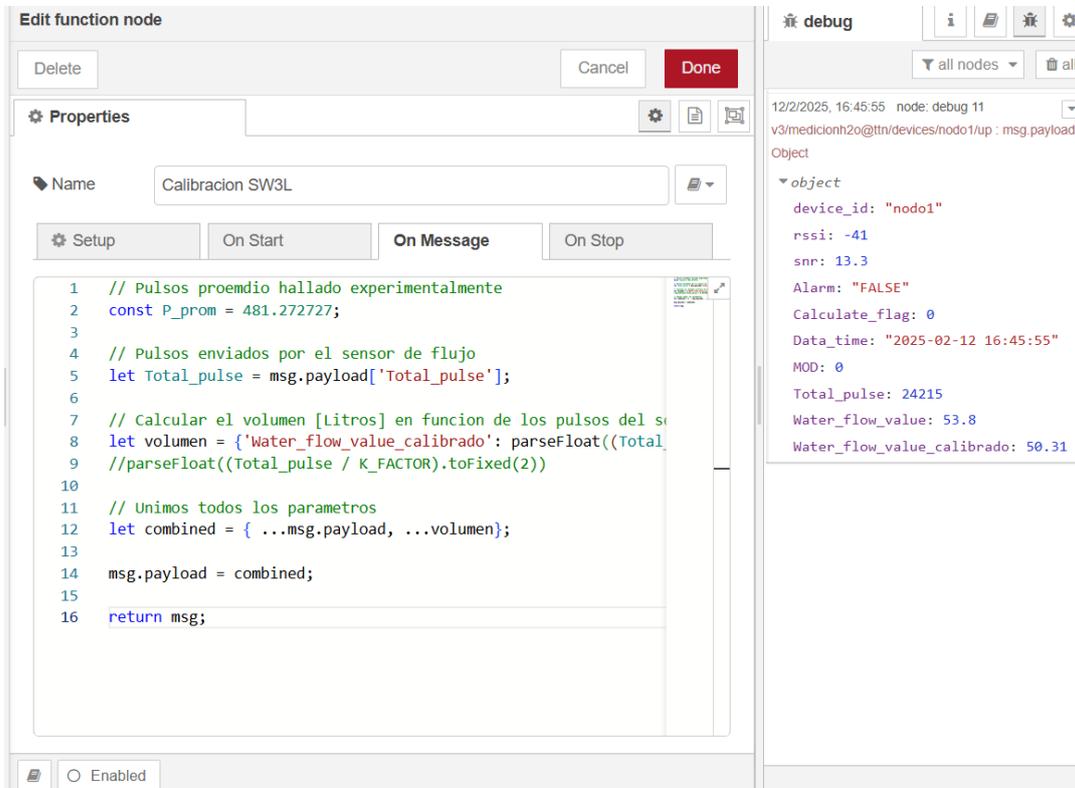
Calibración de las medidas tomadas por el nodo sensor.



Fuente: Elaboración propia.

Figura 38

Algoritmo para calcular el volumen aproximado real que pasa a través del sensor DW-004.



Fuente: Elaboración propia.

3.8.3. Cálculo tarifario

Teniendo calibrado el nodo sensor y teniendo medidas de volumen de agua consumida precisas, se continúa con la tarifación de ese volumen medido. El pliego tarifario actual establecido por SedaCusco S.A que es la empresa prestadora del servicio de agua, mediante resolución N°148-2024-GG-EPSSDACUSCO S.A. aplica la siguiente estructura tarifaria. (Sedacusco, s.f.). La presente tesis se enfoca en usuarios de clase Residencial y categoría Doméstico.

Figura 39

Pliego tarifario actual de la empresa SedaCUSCO.

Pliego tarifario Mayo 2024

PLIEGO TARIFARIO EPS SEDACUSCO S.A. (CUSCO)

Clase	Categoría	Rango de consumo m ³	Tarifa Agua Potable (S//m ³)	Tarifa Alcantarillado (S//m ³)	Cargo Fijo(S/ por mes)
Residencial	Social	0 a 8	1.0674	0.9103	5.11
		9 a más	1.8349	1.5686	5.11
	Doméstico	0 a 8	1.0674	0.9103	5.11
		9 a 22	1.8349	1.5686	5.11
		23 a más	4.5263	3.8681	5.11
	No Residencial	Comercial I	0 a 30	4.5263	3.8681
31 a 50			6.0061	5.1341	5.11
51 a más			8.8804	7.5902	5.11
Comercial II		0 a 30	3.0452	2.6019	5.11
		31 a 50	6.0061	5.1341	5.11
		51 a más	8.8804	7.5902	5.11
Industrial		0 a 100	5.9221	5.0598	5.11
		101 a más	11.2680	9.6330	5.11
Estatal		0 a 70	3.0452	2.6019	5.11
		71 a más	4.5263	3.8681	5.11

Fuente: <https://www.sedacusco.com/estructura-tarifaria-de-los-servicios/>

En el recibo por el servicio de agua se cobran por tres conceptos:

- Volumen de agua consumido.
- Servicio de desagüe.
- Un cargo fijo mensual, actualmente es de S/. 5.11.

La suma de estos tres conceptos da un Subtotal:

$$\text{Subtotal} = \text{Agua} + \text{Desague} + \text{Cargo Fijo}$$

Al valor Subtotal obtenido se le aplica un IGV (actualmente del 18%), finalmente el total facturado del mes será:

$$Total = Agua + Desague + Cargo Fijo + IGV$$

Figura 40

Recibo digital emitido por la empresa SedaCUSCO.



COPIA DE RECIBO
11/02/2025
www.sedacusco.com

Nombre:

Dirección:

DNI/RUC: /

Ruta: **1-1-7-1-63**

Tarifa: **DOM1-012(1) - Beneficiario (NO)**

Modo Fact: **LECTURA**

Abastecimiento en su zona es: 18:0 hrs.

MEDIDOR	Periodo	Lecturas	Cod
Nro. DA19176945	18/12/2024 Anterior	154	
Consumo 4 m3	17/01/2025 Actual	158	

Código	1-1-7-18-70-0
Inscripción	01184478
Periodo C-1	NÚMERO DE RECIBO
FEBRERO - 2025	S101-5680831-62

EVOLUCIÓN CONSUMO	
ENE	4
DIC	4
NOV	4
OCT	3
SEP	5
AGO	6

DESCRIPCIÓN DE CONCEPTOS	NO IMPONIBLE	IMPORTE
201 SERVICIO DE AGUA POTABLE		4.27
301 SERVICIO DESAGUE		3.64
200 CARGO FIJO		5.11
	SUBTOTAL	13.02
	I.G.V. 18%	2.34
	Redondeo	-0.06
	TOTAL DEL MES	15.30
	TOTAL A PAGAR S/.	15.30

Prom Hora Abast ZONA X: 04:00 a 12:00 y 12:00 a 22:00, Horas: Min18:00 y max18:00 <https://www.sedacusco.com/servicios/horario-de-abastecimiento/>

FECHA DE EMISIÓN	06/02/2025	ÚLTIMO DÍA DE PAGO	18/02/2025
------------------	-------------------	--------------------	-------------------

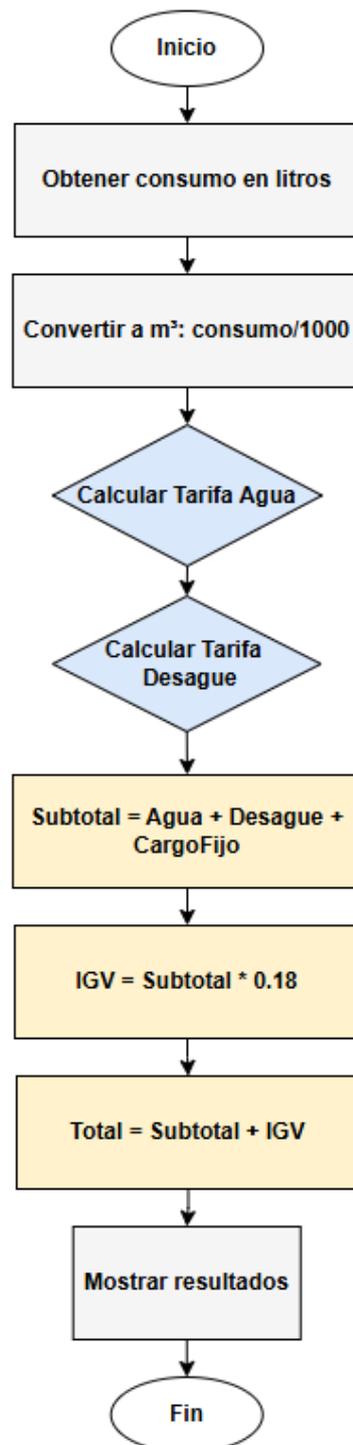
Fuente: <https://comercial.sedacusco.com/cliente/recibo/inscrinro>

Según el pliego tarifario, la tarifa cobrada por agua consumida y el alcantarillado varía de acuerdo a los rangos de consumo (en m^3) que tiene una vivienda. Sabiendo la estructura tarifaria lo siguiente es implementarlo en un algoritmo de acuerdo al diagrama de flujo mostrado en la Figura 41. Para la implementación de

este algoritmo se usará el nodo "Function" de Node-Red, el código está disponible en el **Anexo 6**.

Figura 41

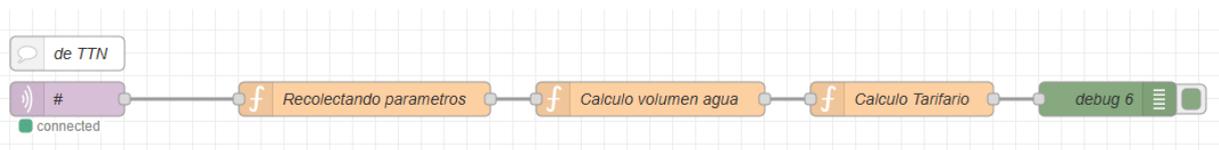
Diagrama de flujo para el algoritmo de cálculo de tarifa.



Fuente: Elaboración propia.

Figura 42

Flujo para tarifar el volumen consumido.



Fuente: Elaboración propia.

Figura 43

Algoritmo para el cálculo de tarifa de acuerdo al volumen consumido.

```
1 // Obtener el valor del consumo de agua[Litros] desde el mensaje
2 let consumo = msg.payload["water_flow_value_calibrado"];
3 // Convertimos de Litros a m3
4 let consumo_m3 = {"Consumo_m3" : consumo / 1000};
5
6 // Cargo fijo mensual definido por la EPS de Agua
7 let Cargo_fijo = 5.11;
8
9 // Función para calcular la tarifa por el consumo de agua
10 function calcularTarifaAgua(consumo) {
11     let tarifa_total = 0;
12
13     // Primer tramo (0 a menos de 9 m³)
14     if (consumo > 0) {
15         let tramo_1 = Math.min(consumo, 9); // Consumo entre 0
16         tarifa_total += tramo_1 * 1.0674;
17         consumo -= tramo_1;
18     }
19
20     // Segundo tramo (9 a menos de 23 m³)
21     if (consumo > 0) {
```

Fuente: Elaboración propia.

Finalmente, el resultado del manejo de datos nos proporciona los parámetros mostrados en la Figura 44. Estos parámetros finales ahora deberán ser guardados en una base de datos.

Figura 44

Datos procesados por Node-Red.

```
18/2/2025, 10:00:09 node: debug 6
v3/medicionh2o@ttn/devices/nodo1/up : msg.payload :
Object
  ▼ object
    device_id: "nodo1"
    rssi: -36
    snr: 13.8
    Alarm: "FALSE"
    Calculate_flag: 0
    Data_time: "2025-02-18 10:00:15"
    MOD: 0
    Total_pulse: 0
    Water_flow_value: 0
    Water_flow_value_calibrado: 0
    Tarifa_Agua: 0
    Tarifa_Alcantarillado: 0
    Consumo_m3: 0
    Total_a_Pagar: 6.03
```

Fuente: Elaboración propia.

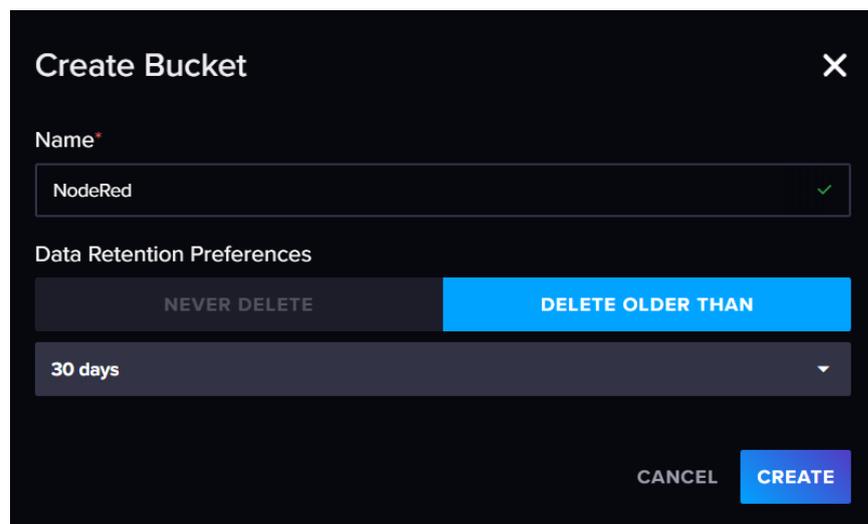
3.9. Integración con la Base de Datos

Una Base de Datos es un software que permite recopilar y organizar datos, se escogió la plataforma InfluxDB Cloud que es una base de datos de series temporales lo que significa que puede registrar datos y almacenarlos como eventos basados en tiempos, esto es útil para la monitorización y seguimiento de cualquier dato del mundo real.

El primer paso es registrarse en la plataforma InfluxDB Cloud, la versión gratuita permite retener los datos durante un periodo de 30 días. Una vez dentro creamos una Organización que es el encargado del manejo general de la plataforma, seguidamente en la pestaña “Load Data” > “Create Bucket”. Un “Bucket” es el nombre de la localización donde los datos serán guardados.

Figura 45

Registramos un Bucket.



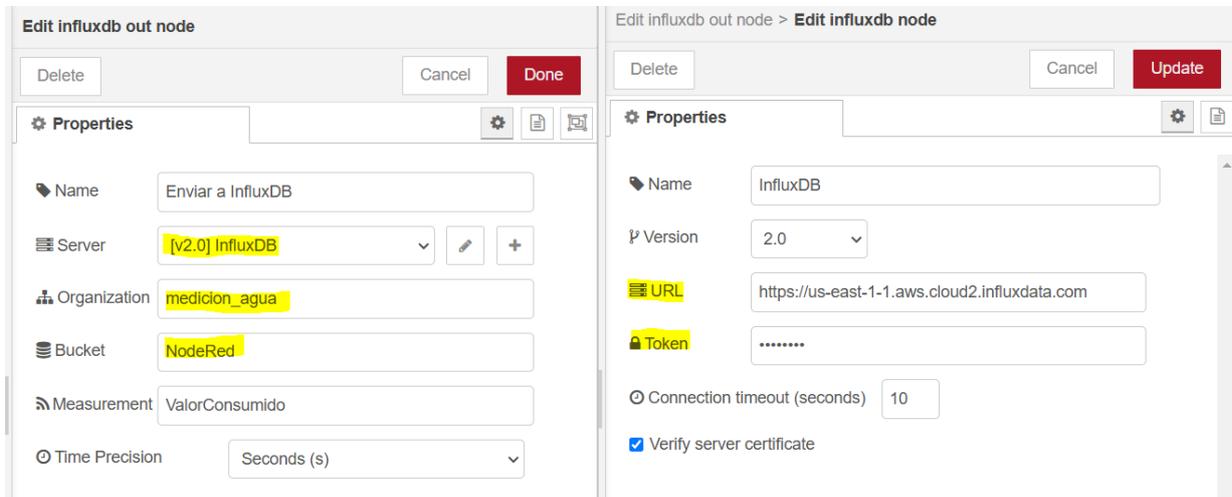
The image shows a 'Create Bucket' dialog box. The title is 'Create Bucket' with a close button (X) in the top right corner. Below the title is a 'Name' field with a red asterisk, containing the text 'NodeRed' and a green checkmark on the right. Underneath is the 'Data Retention Preferences' section, which has two buttons: 'NEVER DELETE' (disabled) and 'DELETE OLDER THAN' (active). Below these buttons is a dropdown menu showing '30 days'. At the bottom right are two buttons: 'CANCEL' and 'CREATE'.

Fuente: Elaboración propia.

Seguidamente procedemos a crear una “API Key” para enlazar NodeRed con la plataforma de InfluxDB mediante el protocolo MQTT, en la pestaña “API Tokens” > “Generate API Token” > “All Access API Token” se otorgan todos los permisos. En la plataforma de NodeRed usando el nodo “InfluxDB out” rellenamos los campos necesarios.

Figura 46

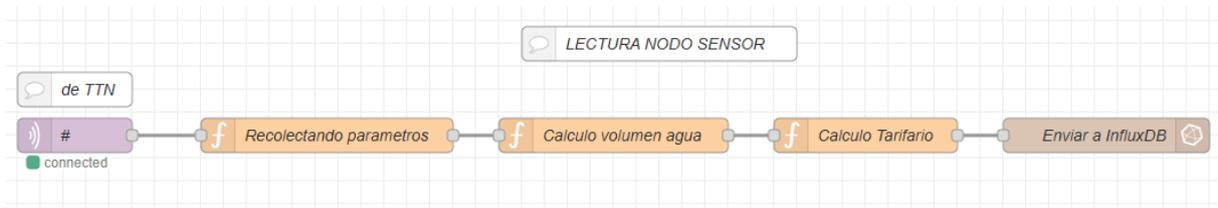
Credenciales para enlazar NodeRed e InfluxDB.



Fuente: Elaboración propia.

Figura 47

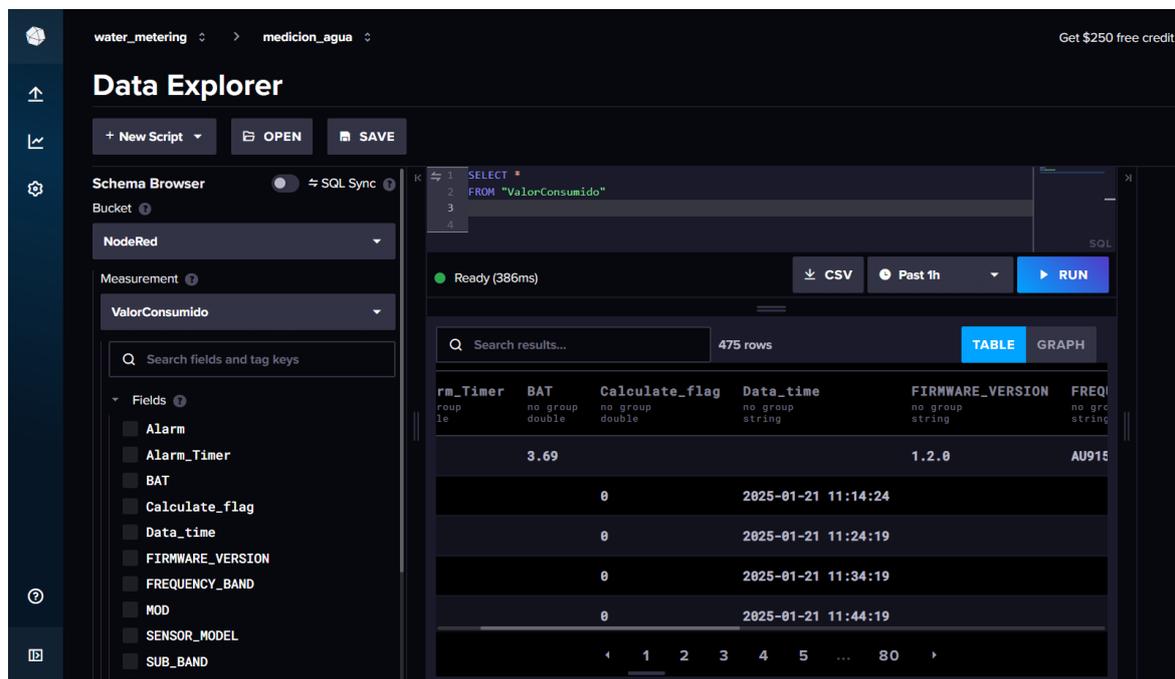
Flujo para enviar datos a InfluxDB



Fuente: Elaboración propia.

Figura 48

Datos almacenados en la plataforma InfluxDB.



Fuente: Elaboración propia.

En la Figura 48, se corrobora que los datos empiezan a guardarse correctamente y en tiempo real dentro de la plataforma de InfluxDB.

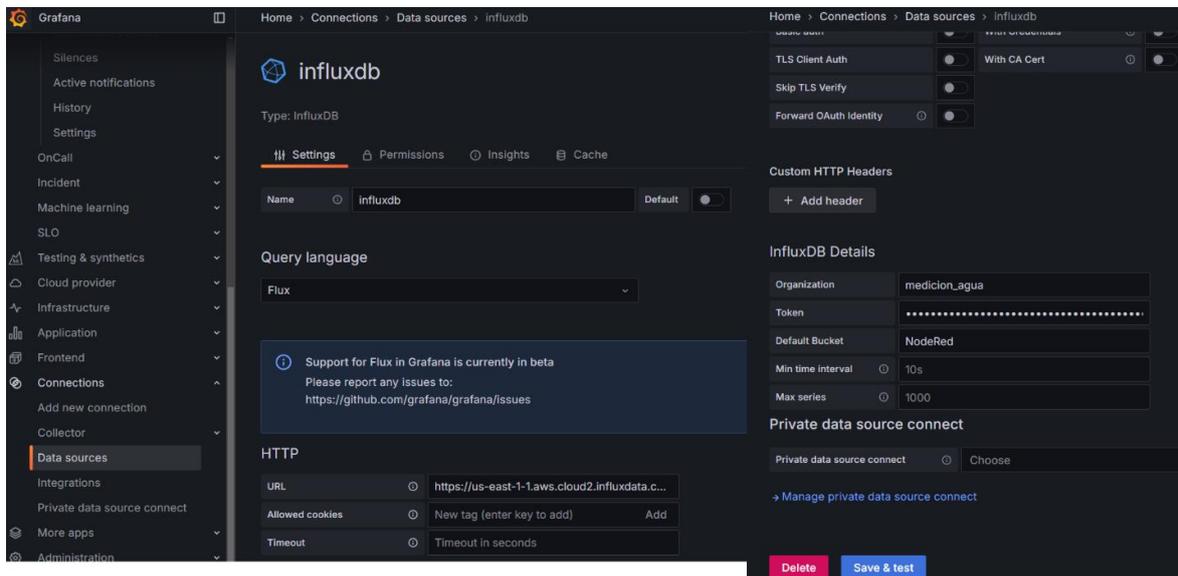
3.10. Integración con la plataforma de visualización

Para la plataforma de visualización se utilizó Grafana Cloud, que es un servicio de código abierto que facilita el análisis, así como la visualización de datos, sus interfaces son atractivos y muy visuales.

Para integrar Grafana con Influx DB primeramente se tiene que indicar de donde provienen los datos, en Grafana en la pestaña "Connections" > "Data Sources" > "InfluxDB" > "Settings" rellenamos los campos requeridos.

Figura 49

Integración de InfluxDB con Grafana.



Fuente: Elaboración propia.

Teniendo acceso a los datos almacenados en InfluxDB desde Grafana, nos queda crear ventanas de monitorización también llamadas “Dashboards”. En la pestaña “Dashboards” > “New” > “New Dashboard” > “Add visualization” > “InfluxDB” nos despliega una nueva ventana, donde mediante lenguaje “Flux” que es un lenguaje de programación parecido a SQL diseñado especialmente para InfluxDB; nos permite interactuar con los datos seleccionados y plasmarlos en gráficos.

Figura 50

Ventana de visualización del sistema IoT.



Fuente: Elaboración propia.

El Dashboard mostrado en la Figura 50 integra múltiples visualizaciones para el monitoreo en tiempo real:

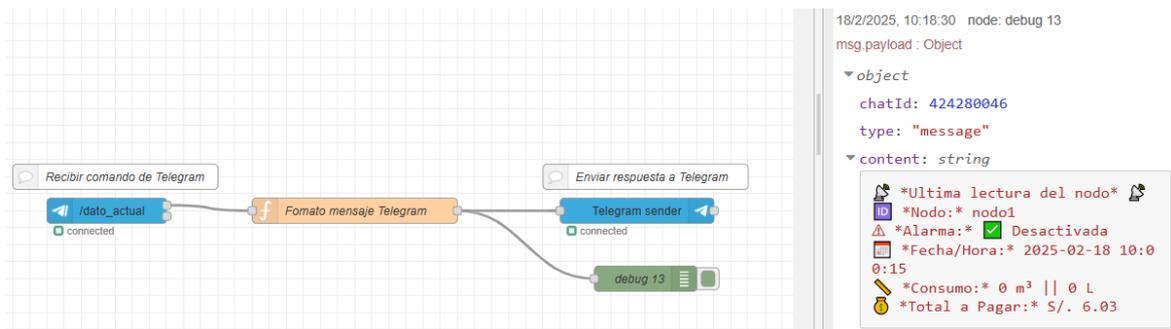
- El consumo de agua en *Litros* y m^3 además de su respectiva tarificación.
- Nivel del voltaje de la batería del nodo sensor en *Voltios*.
- Intensidad de la señal recibida RSSI en *dBm*.
- Alarma para flujo anómalo: Se activa cuando se detecta un inicio de flujo de agua y no detecta un evento de parada de 15 segundos dentro de un temporizador de 60 minutos de flujo continuo. Grafana envía un correo electrónico a una dirección email previamente registrada cuando Alarma pasa a estado TRUE.

3.11. Nodos adicionales

Para facilitar el acceso rápido a la información de consumo de agua que ocurre en una vivienda se decidió integrar NodeRed con la plataforma de mensajería Telegram, el acceso a los valores será en tiempo real y además de ser una plataforma de fácil uso. NodeRed tiene los nodos “command” y “sender” que integran la plataforma Node-Red con un “bot” creado desde la plataforma Telegram, el bot se encuentra bajo el alias de “@MakerH2O_bot”. A continuación, se usa un nodo “Function” que acondicionará los parámetros del nodo sensor que deseamos enviar, cuando ocurra una petición desde Telegram. El código implementado en el nodo “Function” se encuentra disponible en el **Anexo 7**.

Figura 51

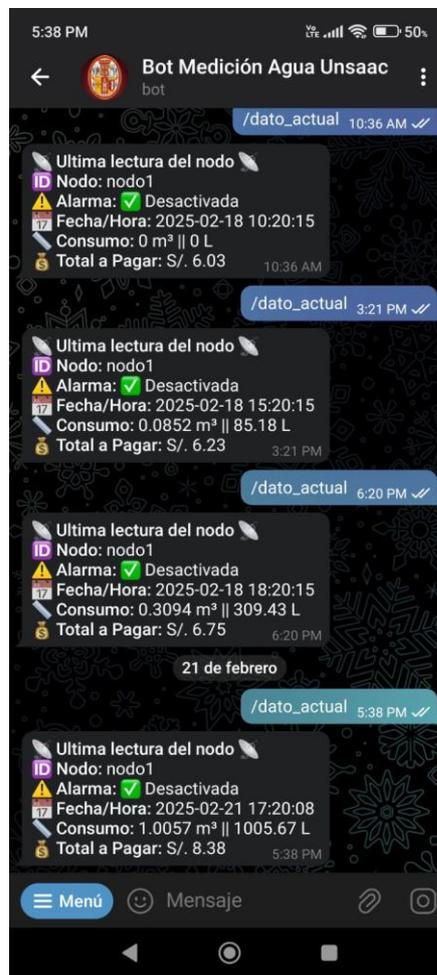
Integración de NodeRed con Telegram.



Fuente: Elaboración propia.

Figura 52

Consulta de medidas desde el bot de Telegram.



Fuente: Elaboración propia.

El nodo Dragino SW3L tiene la funcionalidad de reestablecer los valores medidos y que están almacenados dentro de su tarjeta con un simple mensaje “Downlink”, aprovechando esta interesante característica se implementó un flujo para que cada inicio de mes se reinicie el conteo a cero y las medidas que haga el nodo sensor sea de un mes exacto. Entre los beneficios para reiniciar cada inicio de mes se encuentran:

- El código de cálculo del consumo del mes se simplifica, al no tener la necesidad de calcular el ultimo conteo del mes actual y restarlo con el ultimo

conteo del mes anterior, como se hace actualmente con los medidores de agua mecánicos.

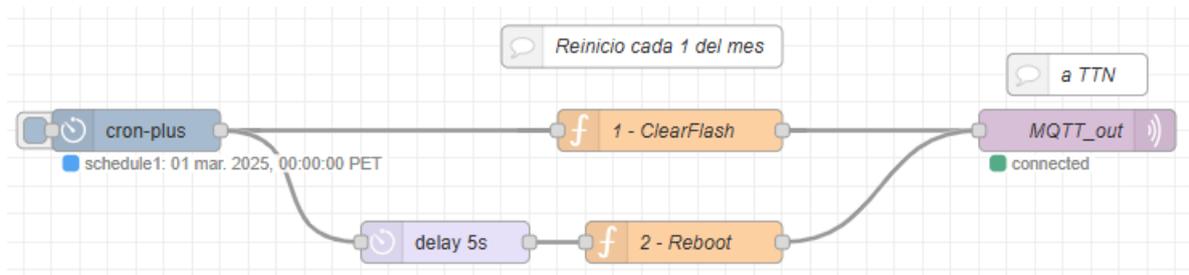
- Reiniciar el conteo a cero cada mes, hace que el error generado durante las medidas no se acumule mes a mes, el máximo error que se tendrá es de un mes de facturación.

De acuerdo al manual otorgado por el fabricante para resetear el conteo del módulo sensor Dragino SW3L se debe enviar un mensaje Downlink en formato hexadecimal “A301” seguido de un “04FF” que reinicia el nodo. Se implemento un diagrama de flujo mostrado en la figura 53. Los códigos de los nodos “Function” se encuentran disponibles en el **Anexo 8**.

El nodo “cron-plus” permite inyectar un mensaje para que empiece un evento en una fecha programada, se configuro para que lance dicho evento cada primero de cada mes a las 00:00 horas, este evento se lanza al servidor LoraWAN mediante MQTT, el servidor envía como un mensaje Downlink al nodo final.

Figura 53

Flujo para reiniciar el conteo del nodo sensor.



Fuente: Elaboración propia.

3.12. Dimensionamiento de cobertura de la red

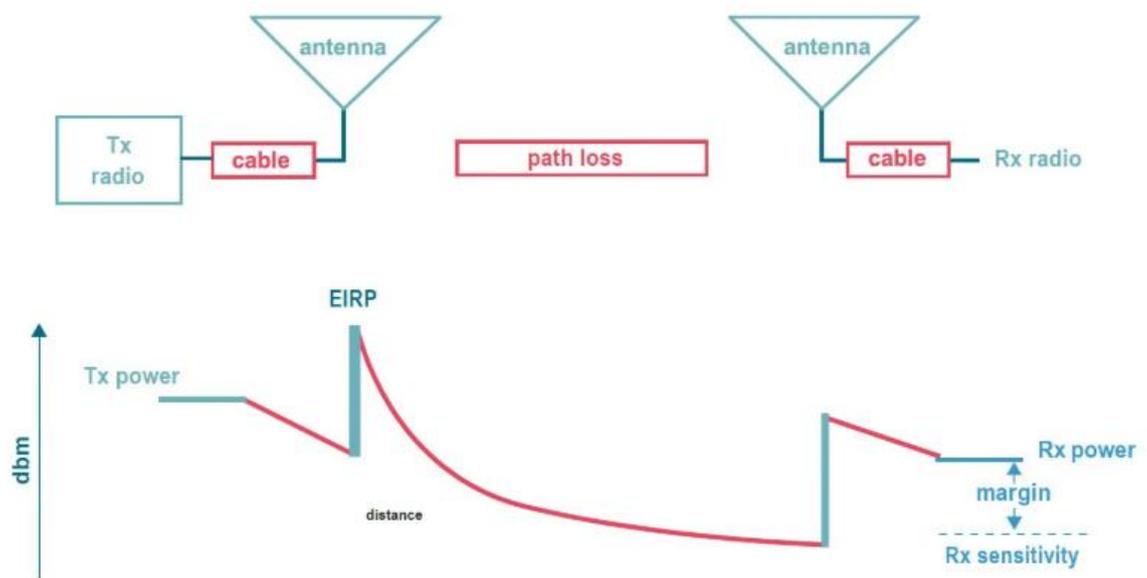
La cobertura de una red LoRaWAN en entornos urbanos presenta desafíos, como la presencia de edificios, la complejidad del terreno y la presencia de otras tecnologías inalámbricas que pueden llegar a afectar la propagación y el alcance de la red. Una cobertura fiable se traduce la capacidad del nodo se pueda comunicar de manera confiable con un Gateway y viceversa dentro de un área determinada, una

cobertura adecuada es fundamental para el éxito en una implementación, ya que afecta directamente en la escalabilidad y rentabilidad de una red LoRaWAN.

Para dimensionar el alcance aproximado de la red propuesta, iniciamos calculando el Presupuesto de enlace (Link Budget) que es un valor de las pérdidas y ganancias desde el emisor, pasando por el canal de propagación y llegando al receptor.

Figura 54

Presupuesto de enlace - Link Budget.



Fuente: Implementación de soluciones basadas en Lora en Alemania. (Semtech, 2019).

Para hallar el presupuesto de enlace, se necesita hallar la pérdida por trayecto (Path Loss), se debe considerar las pérdidas que ocurren en una ciudad y las pérdidas cuando el nodo se encuentra debajo del nivel del piso.

La red se desplegará en una ciudad por tanto hay que considerar los obstáculos presentes en especial los edificios y otras construcciones. Se adiciona una pérdida provocada por los edificios, donde un valor típico es considerar 20 dB como pérdida. (Moya, 2018).

El nodo sensor se instalará dentro una caja de registro que generalmente se encuentra en un pozo a una cierta profundidad bajo el nivel del piso. Según el artículo: Estudio del rendimiento de un enlace inalámbrico LoRa en minas subterráneas, se encontró que existe una reducción del rendimiento aproximado entre 2.5dB a 6dB cuando un nodo sensor se encuentra bajo tierra. (Lloyed, 2019). Se toma como perdida provocada por la caja de registro un valor de 6dB.

$$PathLoss[dB] = EiRP_{Tx}[dBm] - Sensibilidad_{Rx} - 20 - 6$$

$EiRP$ es la Potencia Isotrópica Irradiada que está definida por:

$$EiRP_{Tx}[dBm] = P_{tx}[dBm] + G[dB] - L[dB]$$

La Sensibilidad está definida por:

$$Sensibilidad_{Rx}[dBm] = -174 + 10 \log(B) + SNR + NF$$

Donde:

- B - Ancho de banda del canal: 125KHz.
- SNR – Relación señal a ruido: -20dB para SF12.
- NF - Figura de ruido: 6dB valor típico de Lora.

$$Sensibilidad_{Rx}[dBm] = -174 + 10 \log(125) - 20 + 6 = \mathbf{-137 \text{ dBm}}$$

Con los parámetros de los equipos a emplearse, para un enlace ascendente (Uplink), donde:

- Banda de frecuencia: 915 MHz.
- Sensibilidad de Rx: -137 dBm para un SF12.
- Potencia de transmisión de Tx EiRP: +14 dBm limite recomendado por Semtech Corporation.

$$PathLoss[dB] = 14 - (-137) - 20 - 6 = \mathbf{125 \text{ dB}}$$

Este valor resultante de PathLoss nos indica que podría perderse 125dB en la potencia de señal transmitida y el receptor aún sería capaz de demodular la señal.

Conociendo el valor de pérdida en el trayecto, se puede aplicar un modelo de predicción de propagación para encontrar la distancia desde el Gateway hasta el nodo final. El modelo Okumura-Hata Cost 231, es un modelo de propagación de radio, válido dentro del rango de frecuencias de 800 MHz a 2000 MHz. (Moya, 2018).

$$L[dB] = 46.3 + 33.9 \log(f) - 13.82 \log(h_b) - a(h_m) + (44.9 - 6.55 \log(h_b)) \log(d)$$

Donde:

- f – frecuencia de transmisión en MHz: 915.
- h_b – altura del Gateway en m: 20.
- h_m – altura del nodo sensor en m: 1.
- $a(h_m)$ – factor de corrección.
- d – distancia del Gateway al nodo sensor.

El modelo Okumura-Hata indica que la altura de la antena receptora h_m debe ubicarse entre 1m y 10m sobre el nivel del suelo. Aunque este modelo no fue diseñado para nodos sensores subterráneos, en este estudio se adaptó mediante la incorporación de una pérdida adicional de 6dB mencionado en la página anterior.

El factor de corrección para una ciudad de densidad media, se define como:

$$a(h_m) = (1.1 \log(f) - 0.7)h_m - 1.56 \log(f) + 0.8$$

$$a(h_m) = (1.1 \log(915) - 0.7) * 1 - 1.56 \log(915) + 0.8 = -1.26$$

Reemplazando en la ecuación de Okumura-Hata Cost 231:

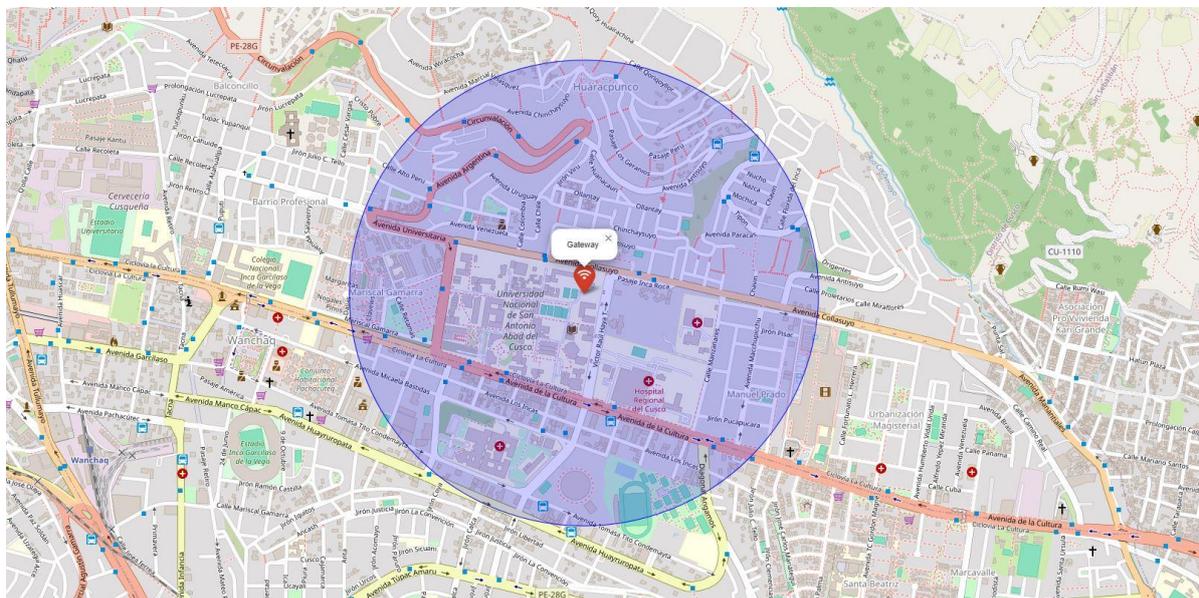
$$125 = 46.3 + 33.9 \log(915) - 13.82 \log(20) - (-1.26) + (44.9 - 6.55 \log(20)) \log(d)$$

$$d = 0.73 \text{ km}$$

Por tanto, se obtiene un radio de cobertura aproximado de 0.73Km, este resultado es esperado para un Gateway diseñado para interiores y una ciudad medianamente densa donde las edificaciones limitan la propagación y alcance de la señal.

Figura 55

Mapa de cobertura teórica de la red LoraWAN.



Fuente: Elaboración propia.

3.13. Simulación de un sistema

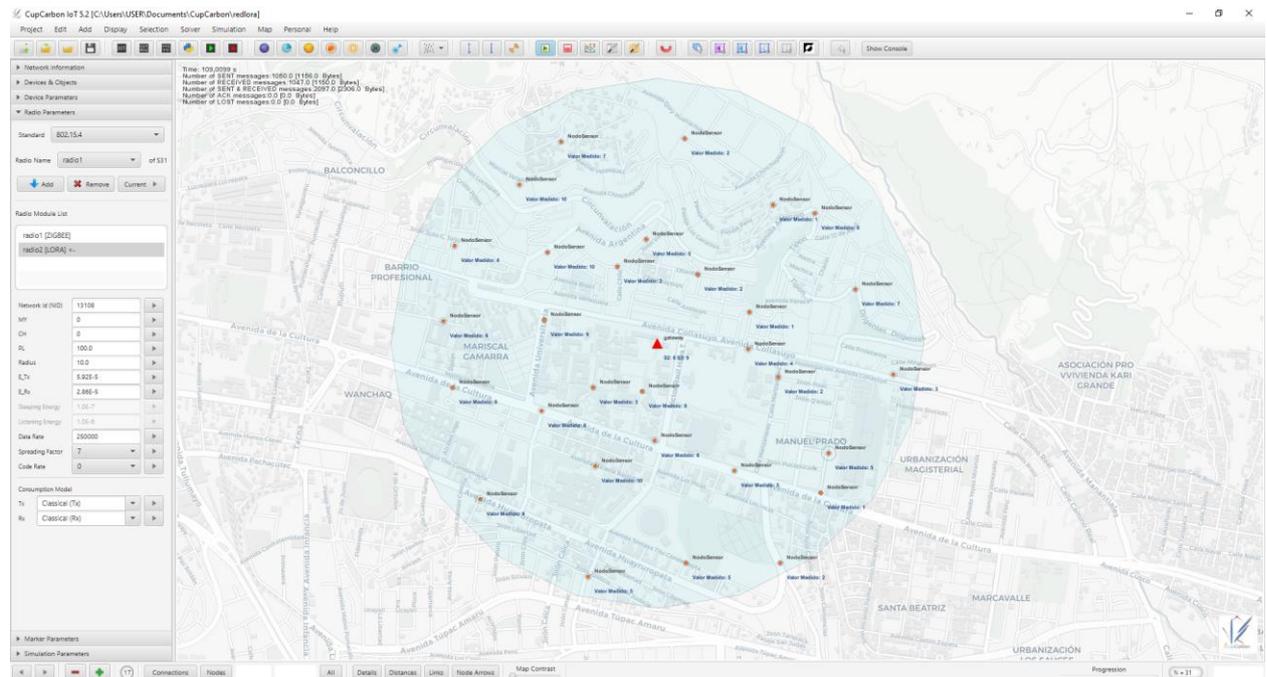
Para simular un sistema con múltiples nodos se usó el software de simulación CupCarbon IoT 5.2, CupCarbon es un software de simulación de Red de Sensores Inalámbricos (SCI-WSN) aplicado a Ciudades Inteligentes e IoT. El software permite diseñar, visualizar, depurar y validar algoritmos distribuidos para la monitorización, la recopilación de datos, etc., y crear escenarios ambientales como incendios, gases, dispositivos móviles y, en general, en proyectos educativos y científicos. No solo ayuda a explicar visualmente los conceptos básicos de las redes de sensores y su funcionamiento, sino que también puede ayudar a los científicos a probar sus topologías inalámbricas, protocolos, etc. (CupCarbon, s.f.).

El software permite la simulación de escenarios usando mapas reales basados en OpenStreetMaps, además se pueden añadir múltiples nodos sensores y estaciones base (Gateways). Se permite programar y configurar los nodos mediante un script basado en Senscript que es lenguaje usado por CupCarbon.

Para la simulación del sistema IoT basado en LoraWAN propuesto, se simuló un Gateway situado en la escuela profesional de Ingeniería Electrónica Unsaac y 30 nodos sensores distribuidos dentro del rango de cobertura del Gateway.

Figura 56

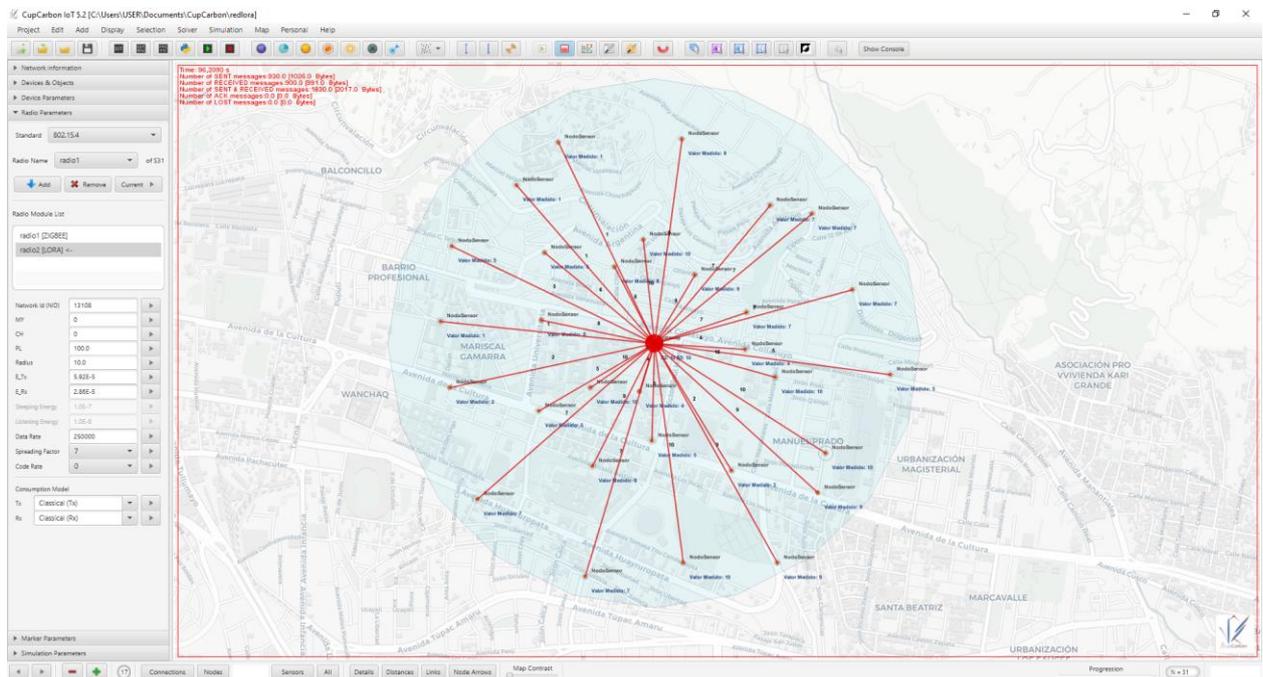
Distribución de un Gateway y 30 nodos sensores.



Fuente: Elaboración propia.

Figura 57

Simulación de una red LoraWAN usando el software CupCarbon.



Fuente: Elaboración propia.

CAPITULO IV: PRUEBAS Y RESULTADOS

4.1. Pruebas de potencia de la señal cuando el nodo sensor está bajo tierra

En la ciudad de Cusco, los medidores de agua se encuentran instalados debajo del nivel del suelo, protegidos por una tapa que cubre la caja de registro. Por tanto, en una instalación real, el nodo sensor se instalará dentro de una caja de registro que se encuentra bajo el nivel del piso.

Para simular estas condiciones, se excavó un pozo al lado del pabellón de la escuela profesional de Ingeniería Electrónica que replica las dimensiones de una caja de registro de agua real, sobre el pozo se colocó tapa de acero galvanizado que es comúnmente usado.

Figura 58

Simulación de una caja de registro de agua.



Fuente: Elaboración propia.

Se tomaron muestras de la potencia recibida RSSI que recibe el Gateway, las 10 primeras tomas se realizaron colocando el nodo sensor encima de la tapa, a nivel del piso. Las siguientes 10 muestras se tomaron colocando el nodo sensor dentro de la caja de registro con la tapa cerrada.

Figura 59

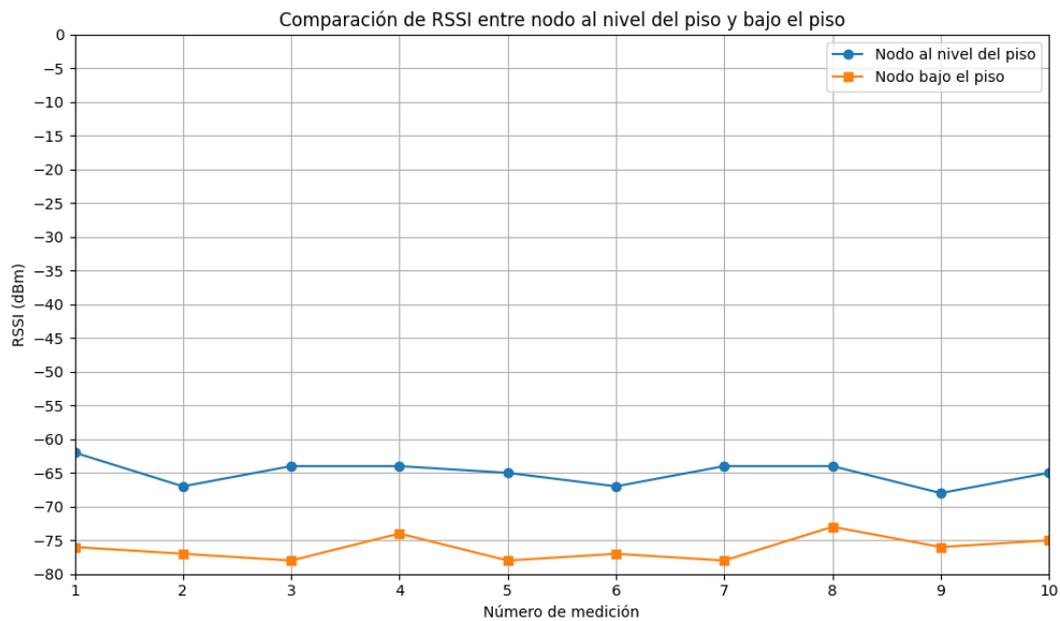
Pruebas para la toma de RSSI recibido.



Fuente: Elaboración propia

Figura 60

Comparación del RSSI entre nodo a nivel del piso y bajo el piso.



Fuente: Elaboración propia.

- El valor de RSSI promedio cuando el nodo sensor se encuentra al nivel del piso es de -65 dBm.
- El valor de RSSI promedio cuando el nodo sensor se encuentra bajo el piso y con la tapa cerrada es de -76.2 dBm.

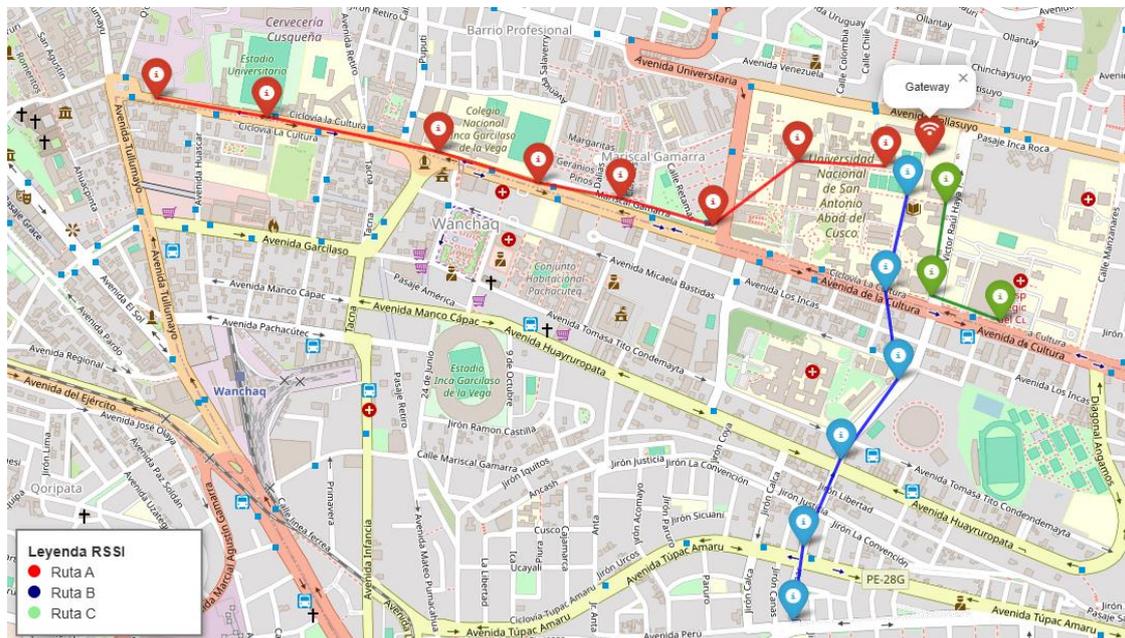
Por tanto, los resultados muestran que el RSSI promedio disminuye de -65 dBm a -76.2 dBm. Lo que indica que una caja de registro de agua añade una atenuación aproximada de 11.2 dB.

4.2. Pruebas de cobertura

Para realizar las pruebas de cobertura real de la red propuesta se tomaron muestras del Indicador de Intensidad de Señal Recibida RSSI aproximadamente cada 200 metros en distintos puntos de la ciudad, el nodo sensor se colocó al nivel del piso durante la toma de muestras. Se siguieron tres rutas diferentes, mostrado en la figura 61.

Figura 61

Rutas recorridas para la toma de medidas de RSSI reales.



Fuente: Elaboración propia.

Se tomaron muestras del valor de RSSI recibido por el Gateway que está ubicado en el quinto piso de la escuela profesional de Ingeniería Electrónica, donde los valores obtenidos se muestran en la siguiente tabla:

Tabla 7

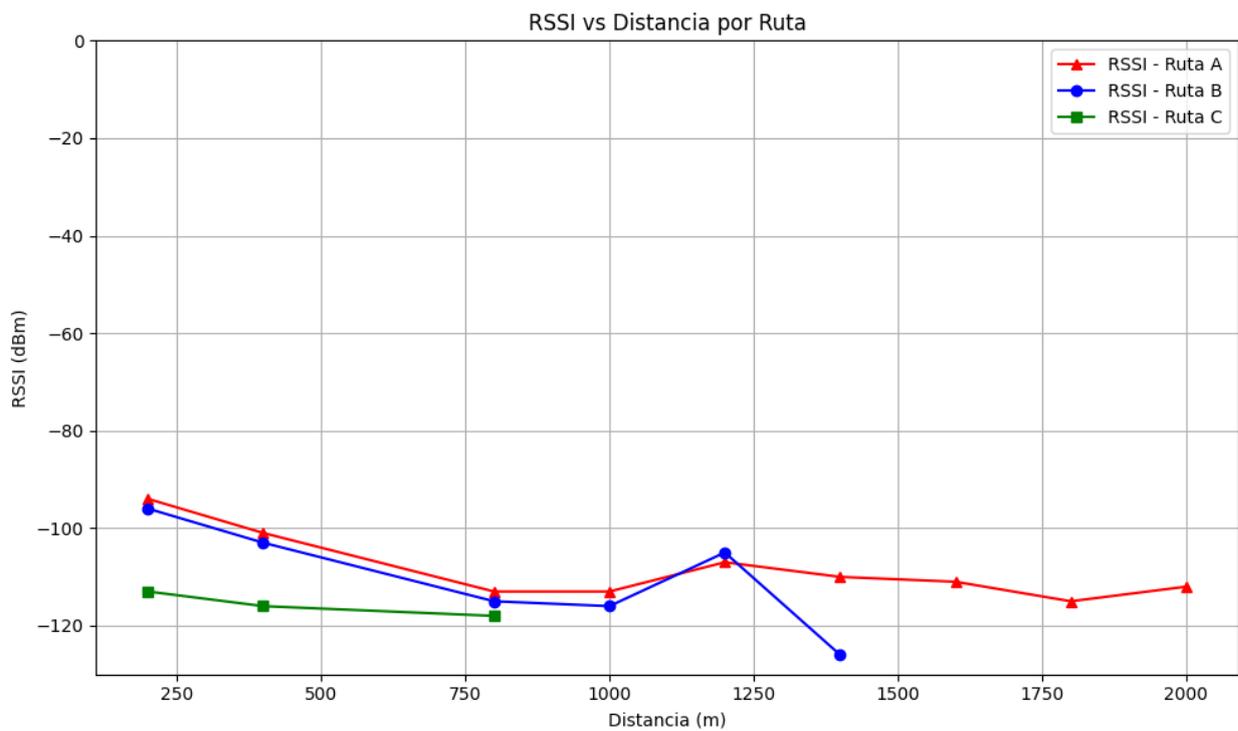
Muestras tomadas en campo

Distancia (m)	RSSI (dBm)		
	Ruta A	Ruta B	Ruta C
200	-94	-96	-113
400	-101	-103	-116
800	-113	-115	-118
1000	-113	-116	-
1200	-107	-105	-
1400	-110	-126	-
1600	-111	-	-
1800	-115	-	-
2000	-112	-	-

Fuente: Elaboración propia.

Figura 62

Valores reales de RSSI en función de la distancia del nodo.



Fuente: Elaboración propia.

En la Figura 62 está plasmado la variación del indicador de intensidad de señal recibida RSSI a medida que se incrementa la distancia entre el nodo sensor y el Gateway. Se observa una disminución progresiva a medida que el nodo sensor se aleja, esto es esperable por la atenuación de la señal en entornos complicados para una señal inalámbrica como es una ciudad. La ruta A se alcanzó la mayor cobertura llegando hasta los 2 Km, esta ruta presenta línea de vista directa lo que facilita la propagación de la señal, la ruta B con una línea de vista indirecta logro una cobertura de 1.2 Km, en contraste con la ruta C sin línea de vista y con múltiples obstrucciones como son muros de concreto donde se alcanzó solamente 800m. Estas diferencias de cobertura se atribuyen principalmente a la ubicación del Gateway, ya que es un Gateway diseñado para interiores y está ubicado en un ambiente cerrado sin línea de vista directa en algunas direcciones.

A los valores de RSSI obtenidos se le debe añadir una pérdida de 11.2dB atribuido a la caja de registro de agua, por tanto, los valores de cobertura cambiarán. En las pruebas de cobertura el RSSI mínimo obtenido fue de -126 dBm, por tanto, tomamos como referencia este valor como límite mínimo de potencia que es capaz de reconocer el Gateway. Entonces realizamos la resta del RSSI mínimo menos la pérdida de 11.2dBm, lo que resulta en -114.8dBm como potencia mínima que le debe llegar al Gateway.

De la tabla 7, en la ruta A con línea de vista directa la potencia de la señal del nodo sensor es capaz de alcanzar al Gateway hasta los 1600m, en la ruta B con línea de vista indirecta hasta los 800m y en la ruta C sin línea de vista hasta los 200m. Entonces, aproximadamente les resta una distancia en promedio de 500m a los valores medidos cuando el nodo sensor se encontraba al nivel del piso. Por tanto, se estima un radio de cobertura real del Gateway de 900m.

4.3. Pruebas de medición de volumen consumido y cálculo de errores

A continuación, se muestra las pruebas realizadas para comprobar las mediciones hechas por el sensor de flujo. Se tomaron muestras con el patrón de la jarra graduada a 1 litro y se tomaron muestras a distintos valores hasta llegar a los 50 litros como se muestra en la tabla 8.

Tabla 8

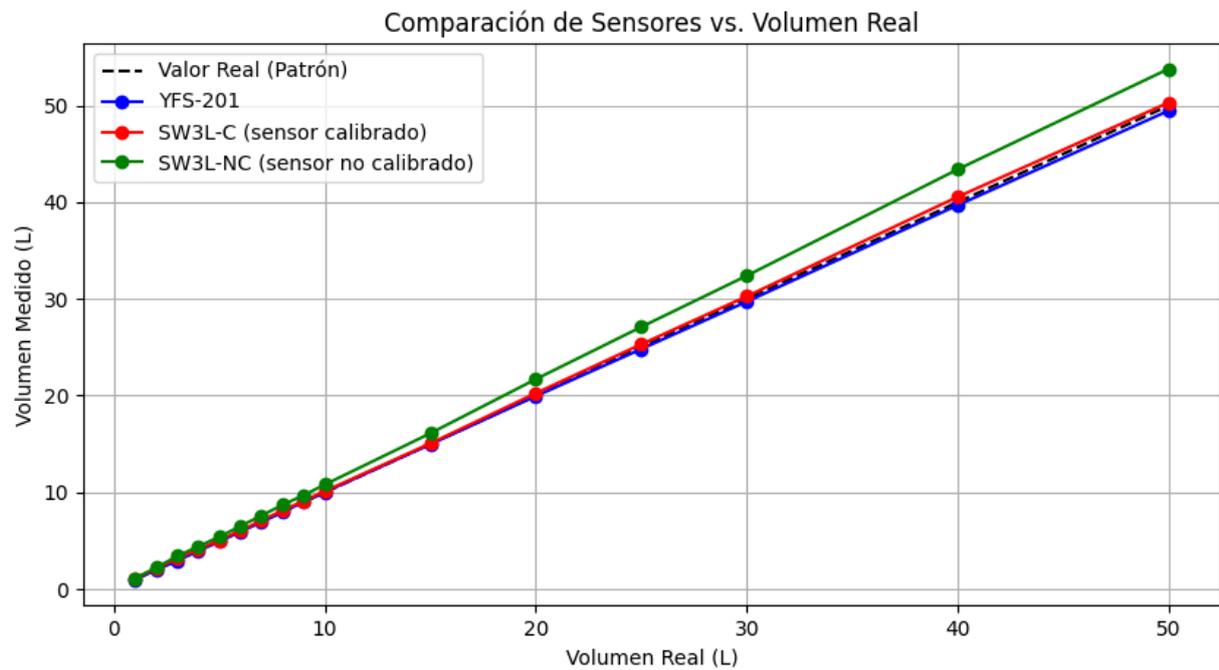
Toma de medidas hechas por los sensores.

PATRON [L]	YFS-201	SW3L - C	SW3L - NC
1	0.982	1.1	1.1
2	1.954	2.08	2.2
3	2.93	3.13	3.4
4	3.935	4.09	4.4
5	4.915	5.03	5.4
6	5.921	6.04	6.5
7	6.933	7.07	7.6
8	7.949	8.13	8.7
9	9.027	9.1	9.7
10	10.028	10.12	10.8
15	14.961	15.05	16.1
20	19.946	20.25	21.7
25	24.815	25.31	27.1
30	29.724	30.27	32.4
40	39.695	40.56	43.4
50	49.47	50.31	53.8

Fuente: Elaboración propia.

Figura 63

Comparación de medidas hechas por los sensores VS medida patrón.



Fuente: Elaboración propia.

En el gráfico 63 muestra que el sensor calibrado tiene medidas muy cercanas al valor patrón, lo que nos confirma la importancia de calibrar de un sensor para garantizar mediciones precisas, en especial en rangos mayores de consumo.

4.3.1. Cálculo del error absoluto

El error absoluto E_a es la diferencia entre el valor de la medida patrón y el valor medido por el sensor, es una forma de expresar el error y se utiliza para indicar la calidad de una medida.

$$E_a = | Valor_{Medido} - Valor_{Patrón} |$$

Tabla 9

Error relativo y absoluto de las medidas hechas por los sensores

PATRO N [L]	YFS-201			SW3L - C			SW3L - NC		
	Medida [L]	Ea	Er	Medid a [L]	Ea	Er	Medid a [L]	Ea	Er
1	0.982	0.018	1.8	1.1	0.1	10	1.1	0.1	10
2	1.954	0.046	2.3	2.08	0.08	4	2.2	0.2	10
3	2.93	0.07	2.333	3.13	0.13	4.333	3.4	0.4	13.333
4	3.935	0.065	1.625	4.09	0.09	2.25	4.4	0.4	10
5	4.915	0.085	1.7	5.03	0.03	0.6	5.4	0.4	8
6	5.921	0.079	1.316	6.04	0.04	0.666	6.5	0.5	8.333
7	6.933	0.067	0.957	7.07	0.07	1	7.6	0.6	8.571
8	7.949	0.051	0.637	8.13	0.13	1.625	8.7	0.7	8.75
9	9.027	0.027	0.3	9.1	0.1	1.111	9.7	0.7	7.777
10	10.028	0.028	0.28	10.12	0.12	1.2	10.8	0.8	8
15	14.961	0.039	0.26	15.05	0.05	0.333	16.1	1.1	7.333
20	19.946	0.054	0.27	20.25	0.25	1.25	21.7	1.7	8.5
25	24.815	0.185	0.74	25.31	0.31	1.24	27.1	2.1	8.4

30	29.724	0.27 6	0.92	30.27	0.27	0.9	32.4	2.4	8
40	39.695	0.30 5	0.76 2	40.56	0.56	1.4	43.4	3.4	8.5
50	49.47	0.53	1.06	50.31	0.31	0.62	53.8	3.8	7.6

Fuente: Elaboración propia.

El módulo Dragino SW3L después de la calibración presenta un error absoluto promedio de 0.165L, sin calibración presenta un error absoluto de 1.21L y el sensor YFS-201 presenta un error absoluto de 0.12L.

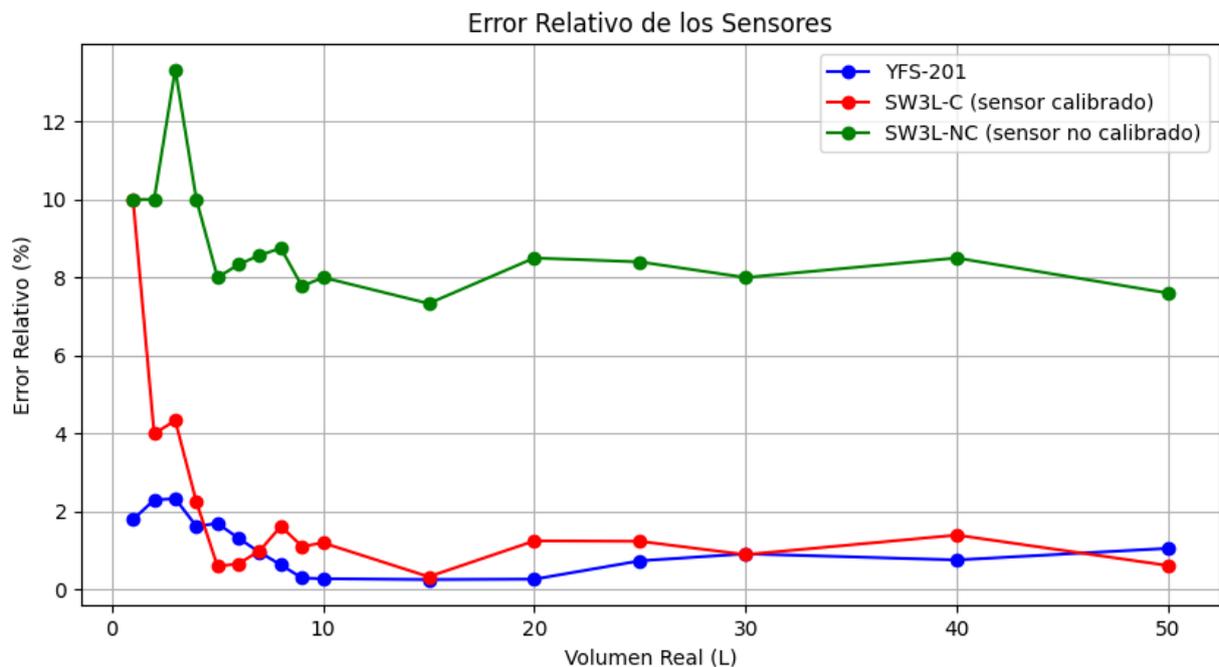
4.3.2. Cálculo del error relativo

El error relativo E_r , se define como el cociente del error absoluto E_a y el valor de la medida patrón, esta medida posibilita evaluar la magnitud del error de una medición en relación con el valor de la medida patrón y permite comparar la precisión entre las diferentes mediciones.

$$E_r = \frac{E_a}{Valor_{patrón}} * 100\%$$

Figura 64

Error relativo de las mediciones hechas por los sensores.



Fuente: Elaboración propia.

Con el análisis de los errores podemos decir que:

- El sensor YFS-201 muestra un error relativo promedio de 1.01%, con un error máximo de 2.3% que ocurre en los primeros rangos de medición.
- El sensor del Dragino SW3L después de calibrar tiene un error relativo promedio de 2.03%, con un error máximo del 10% en los primeros rangos de medición, después de los 5 litros se estabiliza en un error promedio del 0.99%.
- El sensor Dragino SW3L sin calibrar tiene un error relativo promedio de 8.82%, con un error máximo de 13.33%.

4.3.3. Comparación con medidores tradicionales

En la actualidad, en la ciudad de Cusco la mayor parte de medidores de agua utilizados son de tipo mecánico. Como ejemplo, la empresa sedaCusco mediante nota de prensa N° 06-2025-SEDACUSCO-OII (Sedacusco, 2025) indica que renovará 33500 medidores de agua que hayan superado una vida útil mayor a 5 años por unos nuevos medidores de agua mecánicos o también llamados analógicos de la marca DH Galacer.

A continuación, se presenta una tabla comparativa entre medidores mecánicos comúnmente usados en zonas residenciales y el medidor de agua propuesto en la presente tesis.

Tabla 10

Comparación de medidores de agua aprobados por INACAL con el nodo sensor Dragino.

FABRICANTE	DH Galacer	WENLING YOUNIO	MADDALEN A	DRAGINO
Modelo	 LXSG-E DN15	 LXSG- 13D3e/RLN1	 SJ PLUS	 SW3L
Tipo	Mecánico	Mecánico	Mecánico	Electrónico o
Principio de medición	Chorro único	Chorro único	Chorro único	Chorro único
Clase de exactitud	2	2	2	-
Error máximo permitido en la zona inferior del caudal (zonas residenciales)	5%	5%	5%	2%
Alcance de indicación (m3)	99999	99999	99999	Ilimitado
Presión soportada	PMA 16	PMA 16	PMA 16	PMA 17.5
Tipo de conexión (rosca de tornillo)	1/2"	1/2"	1/2"	1/2"
Precio (Soles)	110	165	95	280

Fuente: Elaboración propia.

La Dirección de Metrología del Instituto Nacional de Calidad (INACAL) aprueba una serie de documentos de carácter obligatorio llamado Normas Metrológicas Peruanas (NMP) que son reglamentos técnicos para instrumentos de medición entre ellos los medidores de agua.

En la Tabla 10 se muestra varios medidores de agua mecánicos aprobados por la NMP-005-1:2018 indica que para zonas con caudales bajos (zonas residenciales) el error máximo permitido de un medidor de agua debe ser de +/- 5%. Por tanto, el medidor de agua propuesto que presenta un error del 2% cumple el requerimiento de esta norma, lo que demuestra la viabilidad para su implementación.

Conclusiones

- Se implementó un prototipo de red IoT para la lectura automatizada en tiempo real del consumo de agua cuantificado en *litros* y m^3 ; además del cálculo del valor de facturación correspondiente en *soles*.
- La elección de los dispositivos que conforman la red IoT se realizó en base a los requerimientos del proyecto, la disponibilidad de los componentes y su eficacia para cumplir los objetivos de la presente tesis. El sistema implementado utiliza el protocolo de comunicación LoRaWAN para la transmisión de datos entre el nodo sensor y el Gateway.
- Se configuró exitosamente el nodo sensor y el Gateway para la recolección y transmisión automática de datos hasta el Servidor LoRaWAN.
- Se integraron todos los dispositivos en una red IoT funcional. Posteriormente, se realizaron pruebas empíricas de calibración para garantizar mediciones precisas en las lecturas del volumen de agua consumido, obteniéndose un error de 2.03% en las mediciones respecto al patrón de referencia. Este resultado cumple con la norma NMP-005-1:2018, que establece un error máximo admisible de +/-5% para medidores de agua.
- Se realizaron pruebas de cobertura LoRaWAN en un entorno urbano real, comprobando que la señal es atenuada significativamente en presencia de obstáculos, además la caja de registro añade una pérdida de 11.2dB. El radio de cobertura promedio de la red fue de 900m.
- Para la interacción con la red IoT implementada, se desarrolló una interfaz de visualización en *Grafana* y un bot en *Telegram*. Por tanto, el usuario final tiene acceso en tiempo real a las mediciones realizadas por el nodo sensor tanto desde una PC como desde un celular, con actualizaciones cada 20 minutos.
- Se validó el funcionamiento integral del sistema, confirmando su viabilidad técnica y operativa para implementarse en ciudades en desarrollo, ofreciendo una

solución eficiente para la lectura automatizada del consumo que registran medidores de agua.

Recomendaciones

- Para implementaciones reales se recomienda usar Gateways diseñados para exteriores, ubicándolos en puntos estratégicos para mejorar la cobertura de la red.
- En base a los experimentos realizados y resultados obtenidos, los sensores de flujo deben someterse a un proceso de calibración que mejore la precisión en las mediciones.
- Para la calibración del sensor se realizaron pruebas empíricas, es recomendable realizar la calibración con empresas que tienen personal y equipamiento especializado para realizar esta labor.
- Para despliegues masivos en redes LoRaWAN se recomienda profundizar en la literatura acerca de la cantidad de nodos que puede dar servicio un Gateway, ya que dependerá de algunos parámetros como la frecuencia de envío, longitud del paquete, el factor de dispersión SF, etc. Esto puede afectar directamente para que una red LoRaWAN escale en una ciudad entera con los miles de nodos que se esperaría integrar. Según algunos ejemplos prácticos, como el de *The Things Network (TTN)* que indica que un Gateway puede dar servicio a unos 1000 nodos.
- Durante las pruebas de calibración del sensor, se encontró que no tiene suficiente resolución para detectar caudales de agua muy bajos, atribuidos a fugas de tipo gota a gota. Se recomienda tener en cuenta este detalle para futuras tesis.
- Se recomienda usar una tapa de plástico en la caja de registro de agua, ya que las metálicas atenúan la señal transmitida por el nodo sensor. En su defecto, se puede instalar el transmisor fuera de la caja, con la debida protección contra robo o daños consiguiendo así mejoras en la cobertura de la red.

BIBLIOGRAFIA

- CEPAL. (2021) *Reflexiones sobre la gestión del agua en América Latina y el Caribe*.
Obtenido de. <https://patagonia3mil.com.ar/wp-content/uploads/2021/05/CEPAL-Reflexiones-sobre-la-gestion-del-agua-en-America-Latina-y-el-Caribe.pdf>
- Chesini, F. (2022). *Informe técnico: Cambio climático y salud: una visión iberoamericana*.
Obtenido de <https://www.researchgate.net/.../Informe-tecnico-Cambio-climatico-y-salud-Una-vision-iberoamericana.pdf>
- Gutiérrez, E. (2016). *Análisis y determinación de agua no facturada (ANF) en el sistema de abastecimiento de agua potable en la subzona Larapa en la E.P.S. SEDACUSCO S.A.* (Tesis de licenciatura). Universidad Andina del Cusco.
- EPS SEDACUSCO S.A. (2021). *Encuesta de satisfacción del cliente 2021*. Recuperado de https://www.sedacusco.com/transparencia/gobernanza/encuestas/Encuesta_satisfaccion_cliente_2021.pdf
- Marticorena, P., & Castillo, R. (2024). *Red LPWAN que mide el consumo de agua potable en el distrito de Daniel Hernández* (Tesis de maestría). Universidad Nacional de Huancavelica.
- Calero, M. (2023). *Red de sensores multiservicio LPWAN* (Tesis de maestría). Universitat Oberta de Catalunya.
- Syrmos, E., Sidiropoulos, V., & Bechtsis, D. (2023). *An intelligent modular water monitoring IoT system for real-time quantitative and qualitative analysis* (Tesis de maestría). International Hellenic University.
- Tarazona, C. (2020). *Análisis de las necesidades de una Smart City en el marco de un desarrollo sostenible* (Tesis de máster). Universitat Oberta de Catalunya. Recuperado de <http://hdl.handle.net/10609/118547>
- IDRICA. (s. f.). *Smart metering: Transformando la gestión del agua. Beneficios y aplicaciones* [. <https://www.idrica.com/es/blog/smart-metering/>

IEEE. (2016). *IEEE Standard for Information technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications* (IEEE Std 802.11-2016). Recuperado de https://standards.ieee.org/standard/802_11-2016.html

Bluetooth SIG. (s. f.). *Bluetooth® wireless technology*. Recuperado de <https://www.bluetooth.com/learn-about-bluetooth/tech-overview/>

LoRa Alliance. (s. f.). *What is LoRaWAN?* Recuperado de <https://loro-alliance.org/about-lorawan/>

Hernández, S. (2020). Estudio en detalle de LoRaWAN. Comparación con otras tecnologías LPWAN considerando diferentes patrones de tráfico. Obtenido de <https://openaccess.uoc.edu/bitstream/10609/106369/7/shernandezc6TFM012Omemoria.pdf>

Solera, E. (2018). *Modulación LoRa: Long range modulation—exploración de aplicaciones usando tecnología LoRa*, Medium. Recuperado de <https://medium.com/pruebas-de-laboratorio-de-la-modulaci3n-lora/modulaci3n-lora-4ad74cabd59e>

LoRa Alliance. (s. f.). *Regional parameters LoRaWAN v1.0.3* (Documento técnico). Recuperado de https://loro-alliance.org/wp-content/uploads/2020/11/lorawan_regional_parameters_v1.0.3reva_0.pdf

Baltuille, P. (2023). *LoRaWAN y su aportación a las tecnologías IIoT*. Recuperado de <https://www.incibe.es/incibe-cert/blog/lorawan-y-su-aportacion-las-tecnologias-iiot>

Montagny, S. (2020). *LoRaWAN-LoRa and Internet of Things* (Libro). Université Savoie Mont Blanc.

MOKOSmart. (2019). *How does LoRa sensor send and receive data*. Recuperado de <https://www.mokosmart.com/es/how-does-lora-sensor-send-and-receive-data/>

Cicasa. (2021). *¿Cómo elegir un medidor de agua?* - CICASA: Gestión eficiente del agua.

Recuperado de <https://cicasa.com/como-elegir-un-medidor/>

Younio. (s.f.). *Análisis de las ventajas del medidor de agua inteligente*. Recuperado de

<https://es.younio.com/news-show-1001417.html>

Viva Industrial (s.f.). *Caudalímetro doppler Flujoímetro ultrasónico efecto Doppler*.

Recuperado de <https://flujometros-caudalimetros.com/flujometros-ultrasonicos/caudalimetro-doppler/>

Dragino (s.f.). *SW3L LORAWAN Outdoor Flow Sensor*.

<https://www.dragino.com/products/water-meter-flow-sensor/item/222-sw3l.html>

WisGate Edge Lite 2 RAK7268\RAK7268C - 8 channel indoor LoRaWAN gateway - IoT

Made Easy. (s. f.). <https://www.rakwireless.com/en-us/products/lpwan-gateways-and-concentrators/rak7268-wisgate-edge-lite-2>

Qiu, M. (s. f.). *SW3L -- LORAWAN Outdoor Flow Sensor - DRAGINO*.

<https://wiki.dragino.com/xwiki/bin/view/Main/User%20Manual%20for%20LoRaWAN%20End%20Nodes/SW3L%20LoRaWAN%20Outdoor%20Flow%20Sensor/>

Lora Alliance (s.f.). *LoraWAN regional parameters V1.0.3*, [https://lora-alliance.org/wp-](https://lora-alliance.org/wp-content/uploads/2020/11/lorawan_regional_parameters_v1.0.3reva_0.pdf)

[content/uploads/2020/11/lorawan_regional_parameters_v1.0.3reva_0.pdf](https://lora-alliance.org/wp-content/uploads/2020/11/lorawan_regional_parameters_v1.0.3reva_0.pdf)

Naylamp Mechatronics (s. f.). *Tutorial sensor de flujo de agua*.

https://naylampmechatronics.com/blog/47_tutorial-sensor-de-flujo-de-agua.html

Sedacusco. (s.f.). *Pliogo Tarifario EPS SEDACUSCO S.A*. Recuperado de

<https://www.sedacusco.com/estructura-tarifaria-de-los-servicios/>

Sedacusco. (s.f.). *Consulta y/o paga tu recibo*. Recuperado de

<https://comercial.sedacusco.com/cliente/recibo/inscrinro>

Semtech (2019). *Deploying LoRa®-based solutions in Germany*. Semtech.com. Recuperado

de <https://blog.semtech.com/deploying-lora-based-solutions-in-germany>

Moya, M. (2018). *Evaluación de pasarela Lora/LoraWan en entornos urbanos*.

https://www.researchgate.net/publication/357620788_Evaluacion_de_pasarela_LoRa_LoRaWAN_en_entornos_urbanos

Lloyed, E., Farjow, W., & Fernando, X. (2019). *LoRa wireless link performance in multipath underground mines*. Recuperado de

<https://ieeexplore.ieee.org/abstract/document/8910316>

CupCarbon. (s.f.). *CupCarbon user guide, versión U-One 5.1*. Recuperado de

https://freenwork.com/cupcarbon/cupcarbon_user_guide.pdf

Sedacusco. (2025). *NOTA DE PRENSA N° 06-2025-SEDACUSCO-OII*. Recuperado de

<https://www.sedacusco.com/comunicados/nota-de-prensa-n-06-2025-sedacusco-oii/>

Anexos

Anexo 1: DECODIFICADOR DE PAYLOAD DEL NODO SENSOR

```
1 // Decode decodes an array of bytes into an object.
2 // - fPort contains the LoRaWAN fPort number
3 // - bytes is an array of bytes, e.g. [225, 230, 255, 0]
4 // - variables contains the device variables e.g. {"calibration":
5 "3.5"} (both the key / value are of type string)
6 // The function must return an object, e.g. {"temperature": 22.5}
7 function datalog(i,bytes) {
8   var aa=(bytes[0+i]&0x02)?"TRUE":"FALSE";
9   var bb=(bytes[0+i]&0xFC)>>2;
10  var cc= bytes[1+i];
11  var dd=(bytes[3+i]<<24 | bytes[4+i]<<16 | bytes[5+i]<<8
12 |bytes[6+i])>>>0;
13  var ee;
14
15  if(bb==0x02)
16    ee= (dd/64).toFixed(1);
17  else if(bb==0x01)
18    ee= (dd/390).toFixed(1);
19  else
20    ee= (dd/450).toFixed(1);
21
22  var ff= getMyDate((bytes[7+i]<<24 | bytes[8+i]<<16 | bytes[9+i]<<8 |
23 bytes[10+i])).toString(10));
24  var string='['+aa+', '+bb+', '+cc+', '+dd+', '+ee+', '+ff+'],'+', ';
25
26  return string;
27 }
28
29 function getzf(c_num){
30   if(parseInt(c_num) < 10)
31     c_num = '0' + c_num;
32
33   return c_num;
34 }
35
36 function getMyDate(str) {
37   var c_Date;
38   if(str > 9999999999)
39     c_Date = new Date(parseInt(str));
40   else
41     c_Date = new Date(parseInt(str) * 1000);
42
43   var c_Year = c_Date.getFullYear(),
44   c_Month = c_Date.getMonth()+1,
45   c_Day = c_Date.getDate(),
46   c_Hour = c_Date.getHours()-5,
47   c_Min = c_Date.getMinutes(),
48   c_Sen = c_Date.getSeconds();
49   var c_Time = c_Year + '-' + getzf(c_Month) + '-' + getzf(c_Day) + ' ' +
50 getzf(c_Hour) + ':' + getzf(c_Min) + ':' + getzf(c_Sen);
51
```

```

52  return c_Time;
53 }
54
55 function Decoder(bytes, port) {
56  if(port==0x03)
57  {
58    for(var i=0;i<bytes.length;i=i+11)
59    {
60      var data= datalog(i,bytes);
61      if(i=='0')
62        data_sum=data;
63      else
64        data_sum+=data;
65    }
66    return{
67      DATALOG:data_sum
68    };
69  }
70  else if(port==0x04)
71  {
72    var tdc= bytes[0]<<16 | bytes[1]<<8 | bytes[2];
73    var stop_timer= bytes[4];
74    var alarm_timer= bytes[5]<<8 | bytes[6];
75
76    return {
77      TDC:tdc,
78      Stop_Timer:stop_timer,
79      Alarm_Timer:alarm_timer,
80    };
81  }
82  else if(port==0x05)
83  {
84    var sub_band;
85    var freq_band;
86    var sensor;
87
88    if(bytes[0]==0x11)
89      sensor= "SW3L";
90
91    if(bytes[4]==0xff)
92      sub_band="NULL";
93    else
94      sub_band=bytes[4];
95
96    if(bytes[3]==0x01)
97      freq_band="EU868";
98    else if(bytes[3]==0x02)
99      freq_band="US915";
100   else if(bytes[3]==0x03)
101     freq_band="IN865";
102   else if(bytes[3]==0x04)
103     freq_band="AU915";
104   else if(bytes[3]==0x05)
105     freq_band="KZ865";
106   else if(bytes[3]==0x06)
107     freq_band="RU864";

```

```

108     else if (bytes[3]==0x07)
109         freq_band="AS923";
110     else if (bytes[3]==0x08)
111         freq_band="AS923_1";
112     else if (bytes[3]==0x09)
113         freq_band="AS923_2";
114     else if (bytes[3]==0x0A)
115         freq_band="AS923_3";
116     else if (bytes[3]==0x0B)
117         freq_band="CN470";
118     else if (bytes[3]==0x0C)
119         freq_band="EU433";
120     else if (bytes[3]==0x0D)
121         freq_band="KR920";
122     else if (bytes[3]==0x0E)
123         freq_band="MA869";
124
125     var firm_ver=
126 (bytes[1]&0x0f)+'.'+(bytes[2]>>4&0x0f)+'.'+(bytes[2]&0x0f);
127     var bat= (bytes[5]<<8 | bytes[6])/1000;
128
129     return {
130         SENSOR_MODEL:sensor,
131         FIRMWARE_VERSION:firm_ver,
132         FREQUENCY_BAND:freq_band,
133         SUB_BAND:sub_band,
134         BAT:bat,
135     };
136 }
137 else
138 {
139     var flag=(bytes[0]&0xFC)>>2;
140     var decode = {};
141
142     decode.MOD=bytes[5];
143     decode.Calculate_flag=flag;
144     decode.Alarm=(bytes[0]&0x02)?"TRUE":"FALSE";
145
146     if(flag==3)
147         decode.Water_flow_value=parseFloat((((bytes[1]<<24 |
148 bytes[2]<<16 | bytes[3]<<8 | bytes[4])>>>0)/12).toFixed(1));
149     if(flag==2)
150         decode.Water_flow_value=parseFloat((((bytes[1]<<24 |
151 bytes[2]<<16 | bytes[3]<<8 | bytes[4])>>>0)/64).toFixed(1));
152     else if(flag==1)
153         decode.Water_flow_value=parseFloat((((bytes[1]<<24 |
154 bytes[2]<<16 | bytes[3]<<8 | bytes[4])>>>0)/390).toFixed(1));
155     else
156         decode.Water_flow_value=parseFloat((((bytes[1]<<24 |
157 bytes[2]<<16 | bytes[3]<<8 | bytes[4])>>>0)/450).toFixed(1));
158
159     if(bytes[5]==0x01)
160         decode.Last_pulse=((bytes[1]<<24 | bytes[2]<<16 | bytes[3]<<8 |
161 bytes[4])>>>0);
162     else
163

```

```
164         decode.Total_pulse=((bytes[1]<<24 | bytes[2]<<16 | bytes[3]<<8 |
165 bytes[4]>>>0);
166
167         decode.Data_time= getMyDate((bytes[7]<<24 | bytes[8]<<16 |
168 bytes[9]<<8 | bytes[10])).toString(10));
169
170         if(bytes.length==11)
171         {
172             return decode;
173         }
174     }
175 }
```

Anexo 2: PARÁMETROS UTILES ENVIADOS POR EL NODO SENSOR

```
1 // Recolectamos parametros utiles del sensor
2 // Parametros Rx
3 let device_id = msg.payload['end_device_ids']['device_id'];
4 let rssi = msg.payload['uplink_message']['rx_metadata']['0']['rssi'];
5 let snr = msg.payload['uplink_message']['rx_metadata']['0']['snr'];
6
7 let parametros = {
8     "device_id" : device_id,
9     "rssi" : rssi,
10    "snr" : snr
11 };
12
13 // Parametros del sensor de flujo
14 let dataSensor = msg.payload['uplink_message']['decoded_payload'];
15
16
17 // Unimos todos los parametros
18 let combined = { ...parametros, ...dataSensor};
19
20 msg.payload = combined;
21
22 return msg;
```

Anexo 3: ALGORITMO PARA MEDIR EL NUMERO DE PULSOS DEL SENSOR YF-S201

```
1  volatile unsigned long pulsos = 0; // Cantidad de pulsos del sensor.
2  Volatile por el uso en la interrupción.
3  #define sensorDeFlujo 12 // Pin donde está conectado el sensor (INT0 en
4  Arduino UNO).
5
6  // Función de interrupción para contar los pulsos
7  void flujo() {
8      pulsos++; // Incrementa el número de pulsos.
9  }
10
11 void setup() {
12     pinMode(sensorDeFlujo, INPUT);           // Configuramos el pin del
13     sensor como entrada.
14     attachInterrupt(digitalPinToInterrupt(sensorDeFlujo), flujo, RISING);
15     // Asociamos la interrupción al pin 2.
16     Serial.begin(9600);                     // Inicializamos el puerto
17     serie.
18 }
19
20 void loop() {
21     // Mostramos el número de pulsos por el puerto serie.
22     Serial.print("Pulsos: ");
23     Serial.println(pulsos);
24     delay(500); // Retardo para evitar saturar el puerto serie.
25 }
```

Anexo 4: MEDICION DEL VOLUMEN DE AGUA QUE PASA A TRAVES DEL SENSOR YF-S201

```
1 // Mide el volumen de agua que pasa a traves del sensor YF-s201
2
3 volatile int NumPulsos = 0; // Variable para la cantidad de pulsos
4 recibidos
5 const int PinSensor = 12; // Pin del sensor YF-S201 conectado al
6 GPIO12 del ESP32
7 const float factor_conversion = 7.839394; // Factor hallado
8 experimentalmente
9 float volumen = 0;
10 unsigned long t0 = 0; // Tiempo anterior en milisegundos
11
12 // Función que se ejecuta en interrupción
13 void ContarPulsos() {
14   NumPulsos++; // Incrementamos la variable de pulsos
15 }
16
17 // Función para obtener la frecuencia de los pulsos
18 float ObtenerFrecuencia() {
19   int pulsos;
20   unsigned long tiempoInicio = millis();
21   NumPulsos = 0; // Reiniciamos el contador de pulsos
22   interrupts(); // Habilitamos las interrupciones
23   delay(1000); // Esperamos 1 segundo
24   noInterrupts(); // Deshabilitamos las interrupciones
25   pulsos = NumPulsos;
26   float tiempoMedido = (millis() - tiempoInicio) / 1000.0; // Tiempo en
27 segundos
28   return pulsos / tiempoMedido; // Hz (pulsos por segundo)
29 }
30
31 void setup() {
32   Serial.begin(115200); // Configuramos el puerto serie a 115200 baud
33   pinMode(PinSensor, INPUT);
34   attachInterrupt(digitalPinToInterrupt(PinSensor), ContarPulsos,
35   RISING); // Interrupción en flanco de subida
36   Serial.println("Envie 'r' para restablecer el volumen a 0 Litros");
37   t0 = millis();
38 }
39
40 void loop() {
41   if (Serial.available()) {
42     if (Serial.read() == 'r') volumen = 0; // Reiniciamos el volumen si
43 recibimos 'r'
44   }
45
46   float frecuencia = ObtenerFrecuencia(); // Obtenemos la frecuencia en
47 Hz
48   float caudal_L_m = frecuencia / factor_conversion; // Caudal en L/min
49   unsigned long t1 = millis();
50   float dt = (t1 - t0) / 1000.0; // Diferencia de tiempo en segundos
51   t0 = t1;
52   volumen += (caudal_L_m / 60.0) * dt; // Volumen acumulado en litros
53
```

```
54 // Enviar datos por el puerto serie
55 Serial.print("Caudal: ");
56 Serial.print(caudal_L_m, 3);
57 Serial.print(" L/min\tVolumen: ");
58 Serial.print(volumen, 3);
59 Serial.println(" L");
60 delay(10000);
}
```

Anexo 5: ALGORITMO PARA HALLAR EL VOLUMEN REAL QUE PASA A TRAVES DEL NODO SENSOR DRAGINO SW3L

```
1 // Factor de conversion hallado experimentalmente
2 const K_FACTOR = 481.272727;
3
4 // Pulsos enviados por el sensor de flujo
5 let Total_pulse = msg.payload['Total_pulse'];
6
7 // Calcular el volumen [Litros] en funcion de los pulsos del sensor
8 let volumen = {'Water_flow_value_calibrado': parseFloat((Total_pulse /
  K_FACTOR).toFixed(2))};
9 //parseFloat((Total_pulse / K_FACTOR).toFixed(2))
10
11 // Unimos todos los parametros
12 let combined = { ...msg.payload, ...volumen};
13
14 msg.payload = combined;
15
16 return msg;
```

Anexo 6: ALGORITMO PARA EL CALCULO TARIFARIO EN FUNCION DEL VOLUMEN DE AGUA CONSUMIDO

```
1 // Obtener el valor del consumo de agua[Litros] desde el mensaje de
2 entrada
3 let consumo = msg.payload["Water_flow_value_calibrado"];
4 // Convertimos de Litros a m3
5 let consumo_m3 = {"Consumo_m3" : parseFloat((consumo /
6 1000).toFixed(4))};
7
8 // Cargo fijo mensual definido por la EPS de Agua
9 let Cargo_fijo = 5.11;
10
11 // Función para calcular la tarifa por el consumo de agua
12 function calcularTarifaAgua(consumo) {
13     let tarifa_total = 0;
14
15     // Primer tramo (0 a menos de 9 m³)
16     if (consumo > 0) {
17         let tramo_1 = Math.min(consumo, 9); //Consumo entre 0 y menos
18 de 9
19         tarifa_total += tramo_1 * 1.0674;
20         consumo -= tramo_1;
21     }
22
23     // Segundo tramo (9 a menos de 23 m³)
24     if (consumo > 0) {
25         let tramo_2 = Math.min(consumo, 23 - 9); // Consumo entre 9 y
26 menos de 23
27         tarifa_total += tramo_2 * 1.8349;
28         consumo -= tramo_2;
29     }
30
31     // Tercer tramo (23 m³ en adelante)
32     if (consumo > 0) {
33         let tramo_3 = consumo; // Todo lo que quede después de 23 m³
34         tarifa_total += tramo_3 * 4.5263;
35     }
36
37     return { "Tarifa_Agua": parseFloat(tarifa_total.toFixed(2)) }; //
38 Redondear a dos decimales
39 }
40 // Función para calcular la tarifa del Alcantarillado/Desague
41 function calcularTarifaAlcantarillado(consumo) {
42     let tarifa_total = 0;
43
44     // Primer tramo (0 a menos de 9 m³)
45     if (consumo > 0) {
46         let tramo_1 = Math.min(consumo, 9); // Consumo entre 0 y menos
47 de 9
48         tarifa_total += tramo_1 * 0.9103;
49         consumo -= tramo_1;
50     }
51
52     // Segundo tramo (9 a menos de 23 m³)
53     if (consumo > 0) {
```

```

54     let tramo_2 = Math.min(consumo, 23 - 9); // Consumo entre 9 y
55 menos de 23
56     tarifa_total += tramo_2 * 1.5686;
57     consumo -= tramo_2;
58 }
59
60 // Tercer tramo (23 m³ en adelante)
61 if (consumo > 0) {
62     let tramo_3 = consumo; // Todo lo que quede después de 23 m³
63     tarifa_total += tramo_3 * 3.8681;
64 }
65
66 return { "Tarifa_Alcantarillado":
67 parseFloat(tarifa_total.toFixed(2)) }; // Redondear a dos decimales
68 }
69
70
71 // Calcular la tarifa de agua consumida
72 let tarifaAgua = calcularTarifaAgua(consumo_m3["Consumo_m3"]);
73
74 // Calcular la tarifa por Alcantarillado/Desague
75 let tarifaAlcantarillado =
76 calcularTarifaAlcantarillado(consumo_m3["Consumo_m3"]);
77
78
79
80 // Calculamos el SUBTOTAL: Agua + Desague + CargoFijo
81 //let Subtotal = {'Subtotal' : tarifaAlcantarillado["Tarifa
82 Alcantarillado"] + tarifaAgua["Tarifa Agua"] + Cargo_fijo};
83 let Subtotal = tarifaAlcantarillado["Tarifa_Alcantarillado"] +
84 tarifaAgua["Tarifa_Agua"] + Cargo_fijo;
85 //Calculamos el IGV = 18%
86 let IGV = Subtotal * 18 / 100;
87 //Calculamos Total a Pagar
88 let Total_Pagar = { "Total_a_Pagar": parseFloat((Subtotal +
89 IGV).toFixed(2)) };
90
91
92
93 // Combinamos los parametros en un solo mensaje
94 let combined = { ...msg.payload, ...tarifaAgua,
95 ...tarifaAlcantarillado, ...consumo_m3, ...Total_Pagar };
96
97 // Actualizamos el payload
98 msg.payload = combined;
   global.set("ultimoDato", msg.payload);
   return msg;

```

Anexo 7: FORMATO DE MENSAJE PARA ENVIAR A TELEGRAM

```
1 var ultimoDato = global.get("ultimoDato");
2
3 if (!ultimoDato) {
4     msg.payload = {
5         chatId: msg.payload.chatId,
6         type: "message",
7         content: "No hay datos aún."
8     };
9     return msg;
10 }
11
12 // Formatear mensaje
13 var mensaje = 📡 *Ultima lectura del nodo* 📡\n` +
14     📄 *Nodo:* ${ultimoDato.device_id}\n` +
15     ⚠️ *Alarma:* ${ultimoDato.Alarm === "TRUE" ? "⚠️ Activada"
16 : "✅ Desactivada"}\n` +
17     📅 *Fecha/Hora:* ${ultimoDato.Data_time}\n` +
18     📝 *Consumo:* ${ultimoDato.Consumo_m3} m3+ || ` +
19     💰 *Total a Pagar:* S/. ${ultimoDato.Total_a_Pagar}\n`
20
21 msg.payload = {
22     chatId: msg.payload.chatId,
23     type: "message",
24     content: mensaje,
25     options: { parse_mode: "Markdown" } // Para formato negritas y
26 emojis
27 };
28
29 return msg;
```

Anexo 8: MENSAJE DOWNLINK PARA REINICIAR VALORES DEL NODO SENSOR CADA INICIO DE MES

```
1 // Comandos Downlink para reinicializar la cuenta del SW3L
2 let ClearNode = 'A301';
3
4 // Convertir la cadena hexadecimal a un buffer
5 let hexBuffer = Buffer.from(ClearNode, "hex");
6
7 // Convertir el buffer a base64
8 let base64data = hexBuffer.toString("base64");
9
10 msg.payload = {
11   "downlinks": [{
12     "f_port": 4,
13     "frm_payload": base64data,
14     "priority": "NORMAL"
15   }]
16 }
17 return msg;
```

```
1 // Comandos Downlink para hacer reboot al SW3L
2 let RebootNode = '04FF';
3
4 // Convertir la cadena hexadecimal a un buffer
5 let hexBuffer = Buffer.from(RebootNode, "hex");
6
7 // Convertir el buffer a base64
8 let base64data = hexBuffer.toString("base64");
9
10 msg.payload = {
11   "downlinks": [{
12     "f_port": 4,
13     "frm_payload": base64data,
14     "priority": "NORMAL"
15   }]
16 }
17 return msg;
```



LoRaWAN Outdoor Flow Sensor

SW3L



OVERVIEW:

The Dragino SW3L is a LoRaWAN Flow Sensor. It detects water flow volume and uplink to IoT server via LoRaWAN network. User can use this to monitor the water usage for buildings.

SW3L is powered by 8500mAh Li-SOCl₂ battery. It is designed for long term use up to 10 years. (Actually Battery life depends on the use environment, update period.)

The SW3L will send water flow volume every 20 minutes. It can also detect the water flow status and send Alarm, to avoid the waste for water usage such as broken toilet case.

SW3L is designed for both indoor and outdoor use. It has a weatherproof enclosure and industrial level battery to work in low to high temperatures.

Each SW3L is pre-load with a set of unique keys for LoRaWAN registration, register these keys to LoRaWAN server and it will auto connect after power on.

Features:

- Outdoor Use
- Datalog feature
- Wall Mountable
- Monitor water waste
- Upload water flow volume
- LoRaWAN v1.0.3 Class A protocol
- Firmware upgradable via program port
- AT Commands to change parameters
- Uplink on periodically and open/close event
- Remote configure parameters via LoRa Downlink
- 8500mAh industrial battery (none-rechargeable)
- Bands: CN470/EU433/KR920/US915/EU868/AS923/AU915/IN865

Flow Sensor Spec:

- SW3L-004: DW-004 Flow Sensor, diameter: G1/2" / DN15, 450 pulse = 1 L
- SW3L-006: DW-006 Flow Sensor, diameter: G3/4" / DN20, 390 pulse = 1 L
- SW3L-010: DW-010 Flow Sensor, diameter: G 1" / DN25, 64 pulse = 1 L

Applications:

- Flow Sensor application
- Water Control
- Toilet Flow Sensor
- Monitor Waste water

Order Info: SW3L-XXX-YYY

XXX:

- XXX: Frequency Bands, options: EU433, CN470, EU868, IN865, KR920, AS923, AU915, US915

YYY:

- 004: DW-004 Flow Sensor
- 006: DW-006 Flow Sensor
- 010: DW-010 Flow Sensor



International size reference: C

ELECTRICAL CHARACTERISTICS

(Typical values for cells stored for one year or less, at +30 °C max.)

Nominal capacity 8.5Ah

(At 4mA,+20°C, 2.0V cut off. The capacity restored by the cell varies according to current drain, temperature and cut off voltage.)

Nominal voltage 3.6 V

Maximum continuous current 150mA

(Higher currents possible, consult EVE.)

Maximum pulse capability: Typical up to 200mA

Storage (recommended) +30°C max.
(for more severe condition consult EVE)

Operating temperature range -55 °C /+85 °C

(Operation at temperature different from ambient may lead to reduced capacity and lower voltage plateau readings.)

Typical weight 52 g

ER26500

Lithium-thionyl Chloride
(Li-SOCl₂) Battery

KEY FEATURES

- ✓ High and stable operating voltage
- ✓ High minimum voltage during pulsing
- ✓ Low self discharge rate (less than 1% after 1 year of storage at +25 °C)
- ✓ Stainless steel container
- ✓ Hermetic glass-to-metal sealing
- ✓ Non-flammable electrolyte
- ✓ Restricted for transport (class9)
- ✓ Compliant with IEC 60086-4 Safety standard and EN 60079 -11 intrinsic safety standard
- UL Underwriters Laboratories(UL) Component Recognition (File Number MH28717)

MAIN APPLICATIONS

- ✓ Utility metering
- ✓ Alarms and security devices
- ✓ Memory back-up
- ✓ Tracking systems
- ✓ Automotive electronics
- ✓ Professional electronics
- ✓ IoT devices
...etc.

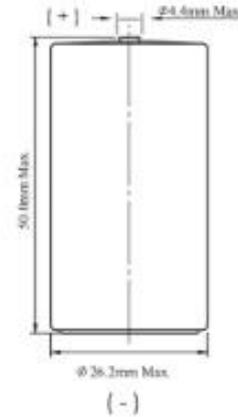
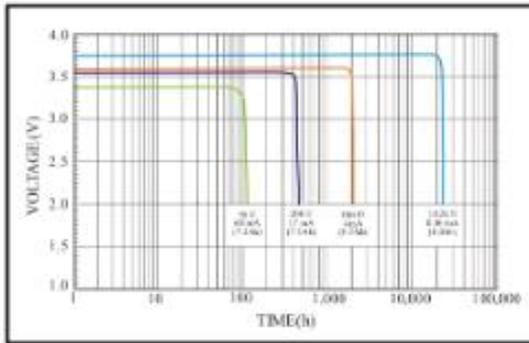
WARNING:

Fire, explosion and severe burn hazard. Do not recharge, crush, disassemble, heat above 100°C, incinerate, or expose contents to water.
Do not solder directly to the cell, use tabbed cell instead.

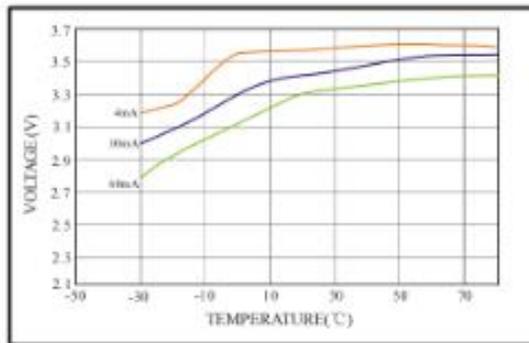
Jan.2020

ER26500

1. Typical discharge profile at +20°C (Typical value)



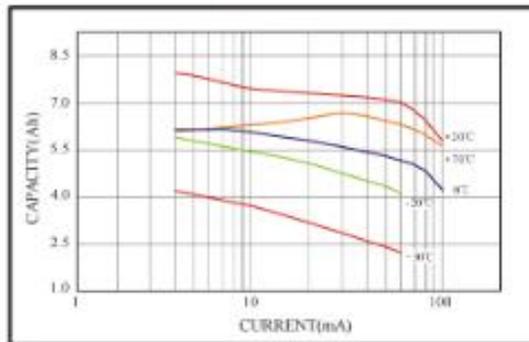
2. Voltage plateau versus Current and Temperature (at min-discharge)



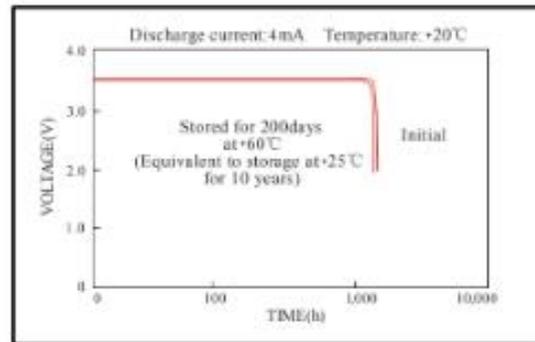
AVAILABLE TERMINATIONS

- Suffix-/S Standard
 - Suffix-/T Solder Tabs
 - Suffix-/W Flying Leads
- [View available terminations](#)

3. Capacity versus Current and Temperature (2.0V cut off) (Typical value)



4. STORAGE CHARACTERISTICS



* Any discharge data in this document are all vertical discharge. Other conditions, consult EVE.

Attention:

Information in this document is subject to change without notice. Any representations in this document concerning performance are for informational purpose only. This document becomes contractual only after written confirmation by EVE.

EVE ENERGY CO., LTD.

Address: EVE Industrial Park, XiKeng Industrial zone, Huihuan Town, Huizhou, Guangdong, China

Operator: (86-752)575 3325 Fax: (86-752)260 6033

<http://www.evebattery.com> Email: sales@evebattery.com

Anexo 11: HOJA TECNICA SENSOR YF-S201

MODEL: YF-S201

Description:

Water flow sensor consists of a plastic valve body, a water rotor, and a hall-effect sensor. When water flows through the rotor, rotor rolls. Its speed changes with different rate of flow. The hall-effect sensor outputs the corresponding pulse signal. This one is suitable to detect flow in water dispenser or coffee machine. We have a comprehensive line of water flow sensors in different diameters. Check them out to find the one that meets your need most.

Features:

- Compact, Easy to Install
- High Sealing Performance
- High Quality Hall Effect Sensor
- RoHS Compliant

Specifications:

- Working Voltage: DC 4.5V~24V
- Normal Voltage: DC 5V~18V
- Max. Working Current: 15mA (DC 5V)
- Load capacity: ≤ 10 mA (DC 5V)
- Flow Rate Range: 1~30L/min
- Load Capacity: ≤10mA (DC 5V)
- Operating Temperature: ≤80°C
- Liquid Temperature: ≤120°C
- Operating Humidity: 35%~90%RH
- Allowing Pressure: ≤1.75MPa
- Storage Temperature: -25~+ 80°C
- Storage Humidity: 25%~95%RH
- Electric strength 1250V/min
- Insulation resistance ≥ 100MΩ
- External threads: 1/2"
- Outer diameter: 20mm
- Intake diameter: 9mm
- Outlet diameter: 12mm



Application:

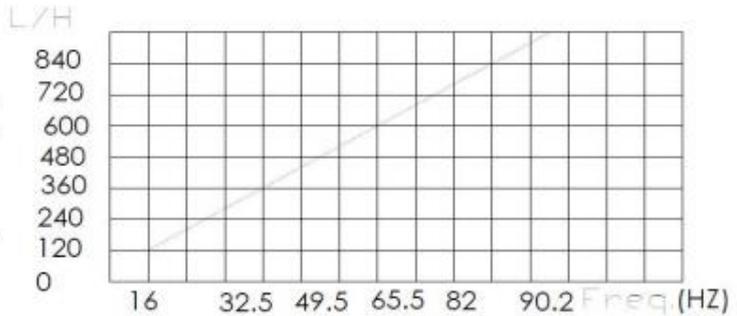
Water heaters, credit card machines, water vending machine, flow measurement device!

Cercuit:

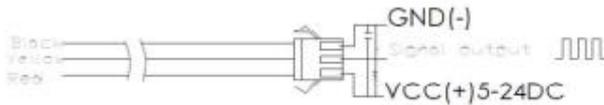
- Red: Positive
- Black: GND
- Yellow: Output signal

Flow Range: 100L/H-1800H-L/H

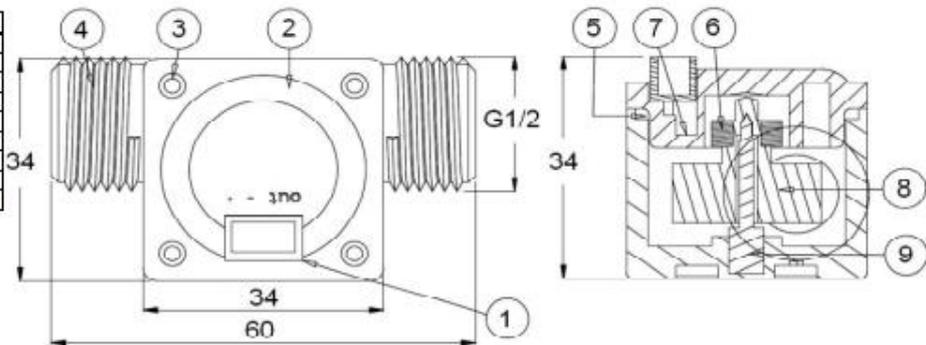
Flow (L/H)	Frezq.(Hz)	Erro range
120	16	±10 5%
240	32.5	
360	49.3	
480	65.5	
600	82	
720	90.2	



Connection method:



N°	Item	Material
1	Wire	PVC
2	Bonnet	PA
3	Screw	Zinc Plated
4	Valve Body	PA
5	Press Valve	
6	Magnet	
7	Hall	
8	Impeller	POM
9	Steel Shaft	SUS304



Closed

ANEXO 12: HOJA TECNICA DEL GATEWAY RAK7268

RAK7268 Quick Start Guide

Prerequisites

What Do You Need?

1. RAK7268/RAK7268C WisGate Edge Lite 2
2. Ethernet Cable (RJ-45 Port) – for Ethernet connection
3. A Windows/MacOS/Linux computer

Product Configuration

Power on the Gateway

1. Attach the LoRa Antenna.

First and foremost, screw on the antenna to the RP-SMA connector on the back panel of the RAK7268/C WisGate Edge Lite 2.

WARNING

Do not power the device if the LoRa Antenna port has been left open to avoid potential damage to the RAK7268/RAK7268C WisGate Edge Lite 2.

2. Power the gateway ON.

It is recommended to use the **12 V DC adapter** that comes with the RAK7268/RAK7268C WisGate Edge Lite 2. Optionally, you can use your own **PoE cable** and **injector** since the device supports PoE.

Casing and Ports



Figure 1: RAK7268/C WisGate Edge Lite 2 Top View

LEDs	Status Indication Description
PWR LED	Power indicator - The LED is on when the device powered.
Breathing LED	Breathing after system is up
ETH LED	ON - Link is up
	OFF - Link is down
LoRa LED	Flicker - Ongoing data transfer
	ON - LoRa is up
	OFF - LoRa is down
	Flicker - Ongoing data transfer
WLAN LED	AP Mode:
	- ON - The AP is up
	- OFF - The AP is down
	- Flicker - Ongoing data transfer
	STA Mode:
	- Slow flicker (1 Hz) - Disconnected
	- ON - Connected
- Flicker - Ongoing data transfer	
LTE LED (will light up only on RAK7268C)	Slow flicker (1800 ms High / 200 ms Low) - Network searching
	Slow flicker (200 ms High / 1800 ms Low) - Idle
	Fast flicker (125 ms High / 125 ms Low) - Ongoing data transfer

Reset Key Functions

The functions of the **Reset** key are as follows:

1. **Short press** - Restarts the gateway.
2. **Long press (5 seconds and above)** - Restores factory settings.

Access the Gateway

In this section, several ways of accessing the gateway are provided to have different alternatives for you to choose from depending on the availability of the requirements needed.

WARNING

Do not power the device if the LoRa Antenna port has been left open to avoid potential damage to the RAK7268/RAK7268C WisGate Edge Lite 2.

ANEXO 13: IMPLEMENTACION DE FLUJOS EN NODE-RED

