



**UNIVERSIDAD NACIONAL DEL ALTIPLANO DE PUNO**  
**FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA,**  
**ELECTRÓNICA Y SISTEMAS**  
**ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA**



**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE**  
**COMUNICACIÓN LORAWAN PARA LA LOCALIZACIÓN DE**  
**VICUÑAS EN LA COMUNIDAD PRIMER CHIMPA JILAHUATA –**  
**AZÁNGARO -AZÁNGARO - PUNO**

**TESIS**

**PRESENTADA POR:**

**Bach. KENNY JHUVENAL QUISPE ROBLES**

**PARA OPTAR EL TÍTULO PROFESIONAL DE:**

**INGENIERO ELECTRÓNICO**

**PUNO – PERÚ**

**2021**



## DEDICATORIA

*Dedico ésta tesis a mi familia, a mi padre Pedro Quispe Laura a quien admiro mucho por haber sacado adelante a mi familia, a mis hermanos Raul Alvin y Robin David con quienes compartimos una vida alegre y divertida y en especial para la persona que más extraño en este mundo, a mi madre Flora Basilia Robles Castillo, no sabes cuanta falta nos haces. Los amo.*

***Kenny Jhuvnal Quispe Robles***



## AGRADECIMIENTOS

*Estoy profundamente agradecido con las personas que en el transcurso hicieron esto posible, en especial al Dr. Jose Enmanuel Cruz de la Cruz que me apoyó durante el proceso. Así también agradecer a los docentes EPIE que me formaron académicamente y a todos mis amigos de la escuela profesional de Ingeniería Electrónica que han permitido el desarrollo de ésta tesis y a mi casa de estudios (UNA-PUNO).*

***Kenny Jhuvnal Quispe Robles***



## ÍNDICE GENERAL

**DEDICATORIA**

**AGRADECIMIENTOS**

**ÍNDICE GENERAL**

**ÍNDICE DE FIGURAS**

**ÍNDICE DE TABLAS**

**ÍNDICE DE ACRÓNIMOS**

**RESUMEN ..... 15**

**ABSTRACT..... 16**

### **CAPÍTULO I**

#### **INTRODUCCIÓN**

**1.1. PLANTEAMIENTO DEL PROBLEMA..... 18**

1.1.1. Formulación del problema ..... 18

**1.2. HIPÓTESIS DE LA INVESTIGACIÓN ..... 19**

1.2.1. Hipótesis general..... 19

**1.3. JUSTIFICACIÓN DEL PROBLEMA..... 19**

**1.4. OBJETIVOS DE LA INVESTIGACIÓN ..... 20**

1.4.1. Objetivo general..... 20

1.4.2. Objetivos específicos ..... 20



## CAPÍTULO II

### REVISIÓN DE LITERATURA

<b>2.1. ANTECEDENTES .....</b>	<b>21</b>
<b>2.2. LPWAN.....</b>	<b>24</b>
2.2.1. Acceso inalámbrico.....	27
2.2.2. Características de LPWAN.....	28
2.2.3. Tecnologías LPWAN.....	33
<b>2.3. LORA y LORAWAN.....</b>	<b>38</b>
2.3.1. Capa de enlace de LoRaWAN.....	41
2.3.2. Escalabilidad en las redes de LoRaWAN.....	45
2.3.3. Parámetros regionales de LoRaWAN.....	46
2.3.4. Activación y Roaming en LoRaWAN.....	47
2.3.5. Geolocalización con LoRaWAN.....	49
2.3.6. Seguridad en LoRaWAN.....	50
<b>2.3.7. Activación del dispositivo final .....</b>	<b>52</b>
<b>2.4. THE THINGS NETWORK .....</b>	<b>59</b>
2.4.1. Funcionalidad del núcleo.....	61
2.4.2. Procesamiento del flujo de mensajes.....	63



## CAPÍTULO III

### MATERIALES Y MÉTODOS

<b>3.1. MATERIALES</b> .....	<b>72</b>
3.1.1. Hardware .....	72
3.1.2. Software .....	76
<b>3.2. MÉTODO</b> .....	<b>79</b>
3.2.1. Diseño de investigación. ....	79
3.2.2. Nivel de la investigación.....	79
3.2.3. Población de la investigación .....	80
3.2.4. Muestra de la investigación .....	80
3.2.5. Ubicación de la investigación. ....	80
3.2.6. Recolección de datos. ....	80
3.2.7. Técnicas de procesamiento y análisis .....	81

## CAPÍTULO IV

### RESULTADOS Y DISCUSIÓN

<b>4.1. COMPORTAMIENTO DE LA COMUNICACIÓN</b> .....	<b>98</b>
4.1.1. Comportamiento en la activación. ....	98
4.1.2. Comportamiento en el envío de información de localización. ....	102
<b>4.2. EXACTITUD DE LA LOCALIZACIÓN</b> .....	<b>108</b>
<b>4.3. COBERTURA</b> .....	<b>109</b>
<b>4.4. DISTANCIA</b> .....	<b>113</b>
<b>V. CONCLUSIONES</b> .....	<b>117</b>



<b>VI. RECOMENDACIONES .....</b>	<b>118</b>
<b>VII. REFERENCIAS.....</b>	<b>119</b>
<b>ANEXOS.....</b>	<b>124</b>
<b>Anexo 1:</b> Línea de comandos para la interpretación de los datos hexadecimales....	124
<b>Anexo 2:</b> Forma de cálculo de tiempo de transmisión. ....	126
<b>Anexo 3:</b> Mapa de Cobertura de la Comunicación LoRa .....	128

**Área : Telecomunicaciones**

**Tema : Comunicación LoRaWAN en zonas rurales**

**FECHA DE SUSTENTACIÓN: 12 DE MARZO DEL 2021**



## ÍNDICE DE FIGURAS

<b>Figura 1.</b> Aplicaciones de las tecnologías LPWAN en diferentes sectores. ....	26
<b>Figura 2.</b> Acceso inalámbrico en la cobertura geográfica. ....	27
<b>Figura 3.</b> Tecnologías LPWAN .....	33
<b>Figura 4.</b> Arquitectura SIGFOX .....	36
<b>Figura 5.</b> Arquitectura y Protocolos de LoRaWAN .....	41
<b>Figura 6.</b> Red LoRaWAN .....	42
<b>Figura 7.</b> Las velocidades de datos en función del alcance de la comunicación y la duración de la transmisión de paquetes .....	43
<b>Figura 8.</b> Impacto del despliegue de multicelular, en la capacidad. ....	46
<b>Figura 9.</b> Separación de JS de los servidores de la red para el procedimiento de activación. ....	48
<b>Figura 10.</b> Uso de energía por la combinación de tecnologías. ....	49
<b>Figura 11.</b> Autenticación mutua y encriptación de extremo a extremo proporcionada por la seguridad de LoRaWAN.....	50
<b>Figura 12.</b> Mecanismo de seguridad LoRaWAN basado en algoritmos criptográficos de encriptación avanzadas. ....	51
<b>Figura 13.</b> Diagrama de activación por OTAA. ....	58
<b>Figura 14.</b> Diagrama de activación por ABP.....	59
<b>Figura 15.</b> Arquitectura de The Things Network.....	60
<b>Figura 16.</b> Arquitectura de funcionalidades de The Things Network. ....	63
<b>Figura 17.</b> Visión general del flujo de The Things Network.....	63
<b>Figura 18.</b> Raspberry Pi. ....	73
<b>Figura 19.</b> Modulo LoRa para Gateway. ....	74
<b>Figura 20.</b> Convertidor para la conexión del Raspberry Pi y el módulo LoRa .....	75





<b>Figura 21.</b> Dispositivo final – End Node.....	76
<b>Figura 22.</b> Arquitectura empleada en la investigación.....	81
<b>Figura 23.</b> Montaje del Gateway .....	82
<b>Figura 24.</b> Detección de la dirección IP del Gateway .....	84
<b>Figura 25.</b> Acceso remoto mediante SSH.....	85
<b>Figura 26.</b> Ventana de la herramienta de configuración de Raspberry Pi.....	86
<b>Figura 27.</b> Diagrama de conexión del Gateway.....	86
<b>Figura 28.</b> Pagina de The Things Network.....	88
<b>Figura 29.</b> Registro del Gateway - parte 1.....	89
<b>Figura 30.</b> Registro del Gateway - parte 2.....	90
<b>Figura 31.</b> Resumen del Gateway registrado.....	91
<b>Figura 32.</b> Agregar una aplicación en The Things Network.....	92
<b>Figura 33.</b> Datos del registro de la Aplicación.....	93
<b>Figura 34.</b> Registro de dispositivo final - End Node.....	94
<b>Figura 35.</b> Resumen de los datos para el dispositivo final.....	95
<b>Figura 36.</b> Acceso al servidor mediante OTAA.....	96
<b>Figura 37.</b> Trama enviada por el End Node para la activación.....	99
<b>Figura 38.</b> Trama de datos en el Gateway, obtenidos en la solicitud de activación del End Node.....	100
<b>Figura 39.</b> Trama de datos en el Gateway, obtenidos en la respuesta de activación del End Node.....	102
<b>Figura 40.</b> Trama de información enviada por el End Node.....	103
<b>Figura 41.</b> Trama de información que pasa por el Gateway, a partir del End Node hacia TTN, parte 1.....	104



<b>Figura 42.</b> Trama de información que pasa por el Gateway, a partir del End Node hacia TTN, parte 2.....	105
<b>Figura 43.</b> Trama de respuesta que pasa por el Gateway, a partir del TTN hacia el End Node, parte 1.....	106
<b>Figura 44.</b> Trama de respuesta que pasa por el Gateway, a partir del TTN hacia el End Node, parte 2.....	107
<b>Figura 45.</b> Ubicación del Gateway. ....	110
<b>Figura 46.</b> Cobertura de la Comunicación LoRa. ....	111
<b>Figura 47.</b> Ubicación de los End Node.....	112
<b>Figura 48.</b> Diagrama de RSSI.....	115
<b>Figura 49.</b> Diagrama de SNR.....	116



## ÍNDICE DE TABLAS

<b>Tabla 1.</b> Especificaciones de algunas tecnologías LPWAN .....	34
<b>Tabla 2.</b> Formato de dirección del dispositivo final. ....	52
<b>Tabla 3.</b> Formato de mensaje de solicitud. ....	55
<b>Tabla 4.</b> Formato de mensaje de aceptación. ....	57
<b>Tabla 5.</b> Resultados de pruebas de coordenadas tomadas por el End Node. ....	108
<b>Tabla 6.</b> Resumen de la cobertura en relación a los End Node.....	113
<b>Tabla 7.</b> Resultados de pruebas de distancia de LoRa. ....	114



## ÍNDICE DE ACRÓNIMOS

### **ABP**

Authentication By Personalisation..... 54, 61, 77, 123

### **ADR**

Adaptive Data Rate ..... 44

### **AES**

Advanced Encryption Standard ..... 52, 53, 56, 59

### **BPSK**

Binari Phase Shift Keying..... 36

### **CDMA**

Code Division Multiple Access ..... 38, 39

### **CMAC**

Cipher-based Message Authentication Code ..... 53

### **CSMA/CA**

Carrier Sense Multiple Access with Collision Avoidance..... 33

### **CTR**

Counter mode..... 53

### **ETSI**

European Telecommunications Standards Institute..... 39, 125

### **EUI**

Identificador Unico Extendido..... 53, 91, 95, 96, 101, 102, 103

### **FSK**

Frequency Shift Keying ..... 44, 75

### **GPRS**

General Packet Radio Service..... 23, 69

### **GPS**

Global Positioning System..... 23, 24, 52, 66, 70, 76, 77, 99, 102, 105

### **IEEE**

Institute of Electrical and Electronic Engineers 39, 53, 55, 56, 74, 122, 123, 124, 125,  
126

### **IoT**



Internet of Things.....	25, 26, 27, 29, 30, 32, 33, 36, 40, 41, 42, 122, 123, 124, 125
<b>ISM</b>	
Industrial, Scientific and Medical .....	36, 38, 39, 41, 42, 43, 48, 49, 50
<b>JS</b>	
Join Server .....	46, 83
<b>LoRa</b>	
Long Range,Una de las tecnologías LPWAN	24, 25, 26, 40, 41, 43, 44, 49, 51, 53, 54, 55, 56, 57, 59, 60, 62, 66, 75, 76, 77, 81, 89, 102, 104, 105, 122, 123, 124, 125, 126
<b>LPWAN</b>	
Low-Power Wide-Area Network .	25, 26, 27, 28, 30, 31, 32, 33, 34, 35, 36, 38, 39, 40, 43, 48, 122, 123, 124, 125
<b>M2M</b>	
Machine to Machine .....	26, 27, 30, 32, 122, 124, 125
<b>MAC</b>	
Media Access Control Address.....	31, 33, 52, 55, 57, 71, 72, 91
<b>NS</b>	
Network Server .....	43, 46, 47, 48, 50, 99
<b>ONG</b>	
Organización no Gubernamental .....	21
<b>ONU</b>	
Organizacion de las Naciones Unidas .....	48, 49
<b>OTAA</b>	
Device Activation Over The Air.....	54, 56, 60, 77, 99, 101, 102, 123
<b>QoS</b>	
Quality of Service .....	34
<b>RAM</b>	
Random Access Memory .....	74
<b>RF</b>	
Radio Frecuencia .....	44, 52, 89
<b>RPMA</b>	
Random Phase Multiple Access .....	31, 38, 39
<b>RTS/CTS</b>	
Request to Send y Clear to Send.....	33



<b>SMS</b>	
Short Message Service.....	23
<b>SNR</b>	
Signal-to-Noise Ratio .....	66, 75
<b>SPI</b>	
Serial Peripheral Interface .....	75, 76, 84, 87
<b>TDMA</b>	
Time Division Multiple Access .....	33
<b>TDoA</b>	
Time Difference of Arrival .....	51
<b>TDOA</b>	
Time Difference Of Arrival .....	24
<b>UDP</b>	
User Datagram Protocol.....	89
<b>UNB</b>	
Ultra Narrow Band.....	39, 40, 125
<b>WLAN</b>	
Wireless Local Area Network.....	30
<b>WNAN</b>	
Wireless Neighborhood Area Network.....	30
<b>WPAN</b>	
Wireless Personal Area Network .....	29, 30
<b>WSN</b>	
Wireless Sensor Networks .....	24
<b>XBEE</b>	
Soluciones integradas que brindan un medio inalámbrico para la interconexión y comunicación entre dispositivos .....	24



## RESUMEN

La vicuña es un camélido sudamericano silvestre, el cual es el símbolo nacional del Perú representando la riqueza animal de este país. Sin embargo, los actos delincuenciales ponen en peligro la población de las vicuñas. En esta tesis se propone diseñar e implementar un sistema de comunicación LoRaWAN para la localización de estos camélidos, esto para asegurar a estos animales, monitoreando su traslado de esta manera prevenir el acceso a zonas peligrosas; también mediante este sistema de comunicación se determina si se encuentran en zonas de pastoreo y acceso al agua, para garantizar su alimentación. La implementación lleva a cabo dos tipos de equipos, un Gateway y otro es el End Node. El Gateway es de una ubicación estática, con la función principal de trasladar la información obtenida por la comunicación LoRa hacia Internet. El End Node es el que se encuentra en movimiento con las vicuñas, proporcionando la localización del mismo. Se obtuvieron resultados en donde se demostró su amplia cobertura obteniéndose un radio de cobertura de 3 kilómetros de manera continua y un alcance máximo de 6 kilómetros según el estado de línea de vista; también una exactitud de localización promedio de 9.11 metros, estos resultados demuestran que es aplicable el sistema en la comunidad Primer Chimpa Jilahuta, además tiene la posibilidad de abarcar a las comunidades aledañas.

**Palabras Clave:** LoRaWAN, localización, vicuñas, rural.



## ABSTRACT

The vicuña is a wild South American camelid, which is the national symbol of Peru representing the animal wealth of this country. However, criminal acts endanger the vicuña population. In this thesis it is proposed to design and implement a LoRaWAN communication system for the location of these camelids, this to secure these animals, monitoring their movement in this way preventing access to dangerous areas; Also through this communication system it is determined if they are in grazing areas and access to water, to guarantee their food. The implementation carries out two types of equipment, a Gateway and another is the End Node. The Gateway is of a static location, with the main function of transferring the information obtained by LoRa communication to the Internet. The End Node is the one that is in motion with the vicuñas, providing its location. Results were obtained in which its wide coverage was demonstrated, obtaining a coverage radius of 3 kilometers continuously and a maximum range of 6 kilometers according to line of sight status; Also an average location accuracy of 9.11 meters, these results show that the system is applicable in the Primer Chimpa Jilahunta community, it also has the possibility of covering the surrounding communities.

**Keywords:** LoraWAN, location, vicuñas, countryside.





# CAPÍTULO I

## INTRODUCCIÓN

LoRaWAN es una arquitectura de sistema inalámbrico de extremo a extremo que proporciona una solución de conectividad de bajo consumo, larga duración, bajo costo, segura y escalable para operadores públicos y redes privadas con una amplia gama de casos de uso de IoT. LoRaWAN permite que los dispositivos finales funcionen con baterías pequeñas por varios años, para lo cual utiliza puertas de enlace de radio con un alcance amplio en áreas rurales. Se basa en el Estándar de cifrado avanzado de 128 bits (AES128) para garantizar la seguridad total de la red, incluida la autenticación mutua de punto final, la autenticación del origen de datos, la reproducción y la protección de la integridad y la privacidad. Su uso de bandas de radio industriales, científicas y médicas (ISM) permite una alta capacidad, operación de bajo costo y diseño específico en los requisitos de IoT.

La presente tesis titulada “DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE COMUNICACIÓN LORAWAN PARA LA LOCALIZACIÓN DE VICUÑAS EN LA COMUNIDAD PRIMER CHIMPA JILAHUATA – AZÁNGARO -AZÁNGARO – PUNO”, es para desarrollar una forma de monitoreo de localización fiable para las vicuñas.

En el capítulo I, se menciona el problema que afecta al hábitat natural de los animales salvajes ya sean mamíferos, reptiles o aves, lo que ocasiona la migración de estos y el riesgo de la extinción; por tal sentido se centra en la localización de las vicuñas. Este sistema emite la posición del animal, de esta manera visualizar la localización de las vicuñas.



En el capítulo II, menciona los antecedentes, y el marco teórico de referente al proyecto de investigación. El marco teórico inicia con la definición de LPWAN ya que es la tecnología donde pertenece la tecnología LoRa y por ende LoRaWAN; posteriormente continúa con la definición de esta tecnología, donde se describe aspectos importantes en la comunicación. De igual manera se describe la plataforma The Things Network, el cual es usado en esta investigación como el servidor de red y servidor de activación.

En el capítulo III, indica los materiales y métodos utilizados, incluyendo el procedimiento de la configuración de los equipos utilizados.

En el capítulo IV, se muestra los resultados del comportamiento de la activación y envío de información del End Node, que es visualizado en la plataforma de The Things Network. Al igual se observa el comportamiento del tráfico que procesa el Gateway, también puede visualizarse en la plataforma anteriormente mencionado. Otros resultados son la precisión de la localización que brinda el End Node, la cobertura y el alcance que puede llegar la comunicación LoRaWAN.

## **1.1. PLANTEAMIENTO DEL PROBLEMA**

### **1.1.1. Formulación del problema**

Se pretende realizar una investigación del nivel exploratoria y el diseño de investigación es cuasi-experimental. Implicando aspectos como la población que va creciendo año tras año, esto viene afectando al hábitat natural de los animales salvajes ya sean mamíferos, reptiles o aves, lo que ocasiona la migración de estos y el riesgo de la extinción. Por tal sentido esta investigación se centra en la localización de las vicuñas, colocándose un dispositivo para transmitir la ubicación de estos animales, lo cual será recepcionado por concentradores ubicadas en zonas estratégicas, la implementación real



de este sistema no demandara elevados gastos lo que permitirá que sean ejecutadas por gobiernos locales u ONG`s. La localización pretende proteger a estos mamíferos, conociendo si su traslado se está realizando por zonas de pastoreo y agua o si acceden en zonas peligrosas.

## **1.2. HIPÓTESIS DE LA INVESTIGACIÓN**

### **1.2.1. Hipótesis general**

El diseño e implementación de un sistema de comunicación LoraWAN permitirá localizar vicuñas dentro de la comunidad Primer Chimpa Jilahuata del distrito de Azángaro

### **1.2.2. Hipótesis específicas**

- El diseño del sistema de comunicación realizado con la tecnología LoRaWAN, proporciona una comunicación de largo alcance y alta duración de baterías.
- La implementación del sistema de localización para el monitoreo de vicuñas, permite un monitoreo en tiempo real.

## **1.3. JUSTIFICACIÓN DEL PROBLEMA**

La vicuña representa la riqueza animal en el escudo nacional, y no es para menos. La fibra de este camélido sudamericano es una de las más finas del mundo y uno de los materiales de exportación no tradicionales que lleva la marca Perú a los mercados internacionales. Actos delincuenciales ponen en peligro la población de las vicuñas aprovechando la topografía de las zonas, y la búsqueda de agua y pastos por estos animales, para reunir ejemplares, matarlos y quitarles la fibra. Hasta el último censo poblacional de la especie, realizado en 2012, se contabilizaban 202 mil unidades vivas,



lo que implica un crecimiento de solo 5% anual con respecto al censo realizado en 2000. Obteniendo la localización de las vicuñas se puede asegurar a estos animales, monitoreando su traslado de esta manera prevenir el acceso a zonas peligrosas; también mediante este sistema de comunicación se determina si se encuentran en zonas de pastoreo y acceso al agua, para garantizar su alimentación. Este sistema consta con End Node colocados en cada vicuña el cual emite la posición del animal, esta información será recepcionado mediante un Gateway el cual enviará la información obtenida por LoRa, hacia un servidor, de esta manera visualizar la localización de las vicuñas.

## **1.4. OBJETIVOS DE LA INVESTIGACIÓN**

### **1.4.1. Objetivo general**

Diseñar e implementar un sistema de comunicación LoraWAN para localizar vicuñas dentro de la comunidad de Chimpa Jilahuata del distrito de Azángaro – Azángaro – Puno.

### **1.4.2. Objetivos específicos**

- Diseñar un sistema de comunicación aplicando el protocolo LoraWAN.
- Implementar un sistema de localización para el monitoreo de vicuñas.



## CAPÍTULO II

### REVISIÓN DE LITERATURA

#### 2.1. ANTECEDENTES

(Basantes, 2016) en la investigación titulada “ANÁLISIS DE FACTIBILIDAD TÉCNICA Y DE VIABILIDAD COMERCIAL DE DISPOSITIVOS PARA LOCALIZACIÓN DE MASCOTAS CANINAS MEDIANTE EL USO DE TECNOLOGÍA GPS EN DISTRITO METROPOLITANO DE QUITO”, analiza y desarrolla un conjunto de aplicaciones de índole informática para localización GPS en base a redes móviles de índole celular GPRS basados en un geo localizador enfocado a animales domésticos, implementando un servidor GPS de localización de dispositivos móviles (GPS tracker), para lo cual será programado y codificado en base a software libre se localicen los dispositivos de este tipo colocados en las mascotas respectivas mediante una interacción con dispositivos móviles o fijos cumpliendo con la disposición de la efectiva búsqueda y localización del can que use dicho dispositivo. Los sensores GPS de uso de la mascota poseen una interacción con la red GSM/GPRS por lo que pueden interactuar con mensajes de texto SMS de la operadora celular hacia el GPS tracking y la plataforma web en general capta las señales de coordenadas y aspectos básicos del dispositivo. De esta manera, tanto dueños, como guardianes, cuidadores y personas interesadas en el cuidado de la mascota se pueden beneficiar de la ubicación del animal en caso de pérdida o extravió, así como una expansión para personas adultas mayores o vehículos en particular que necesiten ser localizadas pues en base a este método se logra generar un fin común.

(Sosa et al., 2015) con la investigación titulada como “LOCALIZACIÓN GEOGRÁFICA DE GANADO UTILIZANDO MODELOS DE PROPAGACIÓN DE



SEÑAL Y XBEE”, presentan los avances en una línea de investigación y desarrollo enfocado en la experimentación con aplicaciones de Redes de Sensores Inalámbricos para soportar servicios en un contexto de Ambientes Inteligentes. El objetivo, radica en el diseño e implantación de un prototipo hardware/software basado en Redes de Sensores Inalámbricos (Wireless Sensor Networks WSN) para capturar la intensidad de las señales de los enrutadores móviles, y mediante el método de Trilateración, ayudar a la localización de ganado en un área determinada. El prototipo permite analizar distintos modelos de propagación y técnicas de estimación de distancia y visualizar los resultados en una interfaz web amigable. El prototipo está en su fase de prueba en laboratorio y de obtención de los primeros resultados en campo.

(Acevedo, Coduri, & Perera, 2018) en la investigación titulado “GEOLOCALIZACIÓN CON LORA MEDIANTE MULTILATERACION”, presentan un proyecto para el cual se construyó un prototipo de geolocalización con el uso de LoRa, una tecnología emergente de radiofrecuencia que promete ser de bajo consumo energético y gran alcance. El mismo se desarrolló con la motivación de sustituir dispositivos que utilizan GPS para la geolocalización animal en estudios de comportamiento de los mismos, debido a que varias de estas investigaciones mencionan que deben cambiar las baterías de los dispositivos al menos dos veces por día afectando completamente el comportamiento de los animales. Este proyecto analiza la duración de las baterías de los dispositivos que cargan los animales. Las pruebas realizadas sobre la duración de las mismas dieron extremadamente satisfactorias obteniendo estimaciones en el orden de meses. Además, el proyecto analiza la precisión de la geolocalización del sistema construido. El prototipo utiliza el método TDOA para realizar los cálculos de ubicación. Las pruebas realizadas para analizar la precisión del sistema dieron resultados con errores de cientos de metros. Se identifican en el proyecto las razones de estos errores y se



presentan no solo las causantes de los mismos, sino también la manera de mitigarlos. Finalmente, concluye que LoRa tiene un gran potencial como medio alternativo de geolocalización para casos de uso donde la batería no puede ser reemplazada fácilmente ya que es una tecnología de bajo consumo energético. Dejando como trabajo a futuro minimizar los errores de precisión.

(Liy et al., 2018) en la investigación de título “TOWARDS LOCATION ENHANCED IOT: CHARACTERIZATION OF LORA SIGNAL FOR WIDE AREA LOCALIZATION”, caracteriza completamente las señales LoRa en CN470 MHz, y evalúa los servicios de ubicación reales a través de la red de área amplia de baja potencia (LPWAN). Mientras tanto, este artículo ha considerado múltiples factores, como la cobertura de la señal, la distribución de la señal a través del espacio y a lo largo del tiempo, la diversidad de los dispositivos, y el movimiento de ellos. Los resultados de este documento pueden proporcionar información suficiente para los servicios de IoT mejorados en ubicación.

(Tanaka et al., 2017) en la investigación de título “A STUDY OF BUS LOCATION SYSTEM USING LORA: BUS LOCATION SYSTEM FOR COMMUNITY BUS “NOTTY””, proponen un método de ubicación de autobuses utilizando LoRa. Este sistema propuesto fue capaz de reducir el número de repetidores y paradas de autobús a 1/4 o menos en comparación con el sistema que utiliza Wi-SUN. El sistema propuesto es superior al sistema que utiliza Wi-SUN en términos de costo de instalación.

(Lin, Ying, & Zheng, 2019) en la investigación “DESIGN AND IMPLEMENTATION OF LOCATION AND ACTIVITY MONITORING SYSTEM BASED ON LORA”, proponen un nuevo mecanismo para recopilar y transmitir información de monitoreo basada en la tecnología LoRa. El dispositivo de monitoreo con sensores puede recopilar



la información de actividad y ubicación en tiempo real y transmitirla al servidor en la nube a través de la puerta de enlace LoRa. El usuario puede consultar todo su historial e información actual a través de las aplicaciones móviles diseñadas específicas. Llevan a cabo un experimento para verificar la comunicación, el consumo de energía y el rendimiento de supervisión de todo el sistema. Los resultados experimentales demuestran que este sistema puede recopilar información de monitoreo y actividad con precisión y proporcionar la cobertura de largo alcance con bajo consumo de energía.

## 2.2. LPWAN

Con el surgimiento de las comunicaciones de Internet de las Cosas (IoT) y de la comunicación Máquina a Máquina (M2M), se espera un crecimiento masivo en el despliegue del nodo sensor pronto. Según el pronóstico de Ericsson, alrededor de 29 mil millones de dispositivos se conectarán a Internet para 2022. Estos dispositivos IoT conectados incluyen automóviles, máquinas, medidores, sensores, terminales de punto de venta, productos electrónicos de consumo, dispositivos portátiles conectados, y otros. La encuesta de IoT informada en el sitio web de Forbes pronostica más de 75 mil millones de conexiones de dispositivos IoT para 2025. HIS Markit pronosticó que el número de dispositivos IoT conectados crecería a 125 mil millones en 2030. El crecimiento exponencial en IoT está impactando virtualmente todas las etapas de la industria y casi todas las áreas del mercado. Está redefiniendo las formas de diseñar, administrar y mantener las redes, datos, nubes y conexiones (Chaudhari & Zennaro, 2020).

Las aplicaciones de IoT tienen requisitos específicos como largo alcance, baja velocidad de datos, bajo consumo de energía y rentabilidad. Las tecnologías de radio de corto alcance ampliamente utilizadas (por ejemplo, ZigBee, Bluetooth) no están





adaptadas para escenarios que requieren una transmisión de largo alcance. Las soluciones basadas en comunicaciones celulares (por ejemplo, 2G, 3G y 4G) pueden proporcionar una mayor cobertura, pero consumen una energía excesiva del dispositivo. Por lo tanto, los requisitos de las aplicaciones IoT han impulsado el surgimiento de una nueva tecnología de comunicación inalámbrica: la Red de Área Amplia de Baja Potencia (LPWAN - Low Power Wide Area Network)

Las Redes de Área Amplia de Baja Potencia (LPWAN) están atrayendo mucha atención principalmente debido a su capacidad de ofrecer conectividad asequible a los dispositivos de baja potencia distribuidos en áreas geográficas muy grandes. Al realizar la visión de Internet de las cosas (IoT), las tecnologías LPWAN complementan y algunas veces reemplazan a las tecnologías inalámbricas celulares y de corto alcance convencionales en rendimiento para varias aplicaciones emergentes de ciudad inteligente y máquina a máquina (M2M) (Raza, Kulkarni, & Sooriyabandara, 2017).

Las redes de área amplia de baja potencia (LPWAN) representan un paradigma de comunicación novedosa, que complementará las tecnologías inalámbricas celulares y de corto alcance tradicionales para abordar los diversos requisitos de las aplicaciones IoT. Las tecnologías LPWAN ofrecen conjuntos únicos de características que incluyen conectividad de área amplia para dispositivos de baja potencia y baja velocidad de datos. Se espera que su mercado sea enorme. Aproximadamente una cuarta parte del total de 30 mil millones de dispositivos IoT/M2M deben conectarse a Internet mediante redes LPWAN utilizando tecnologías patentadas o celulares (Raza et al., 2017). La Figura 1 destaca la variedad de aplicaciones en varios sectores comerciales que pueden explotar las tecnologías LPWAN para conectar sus dispositivos finales. Estos sectores comerciales incluyen, entre otros, ciudad inteligente, aplicaciones personales de IoT, red inteligente, medición inteligente, logística, monitoreo industrial, agricultura, etc.



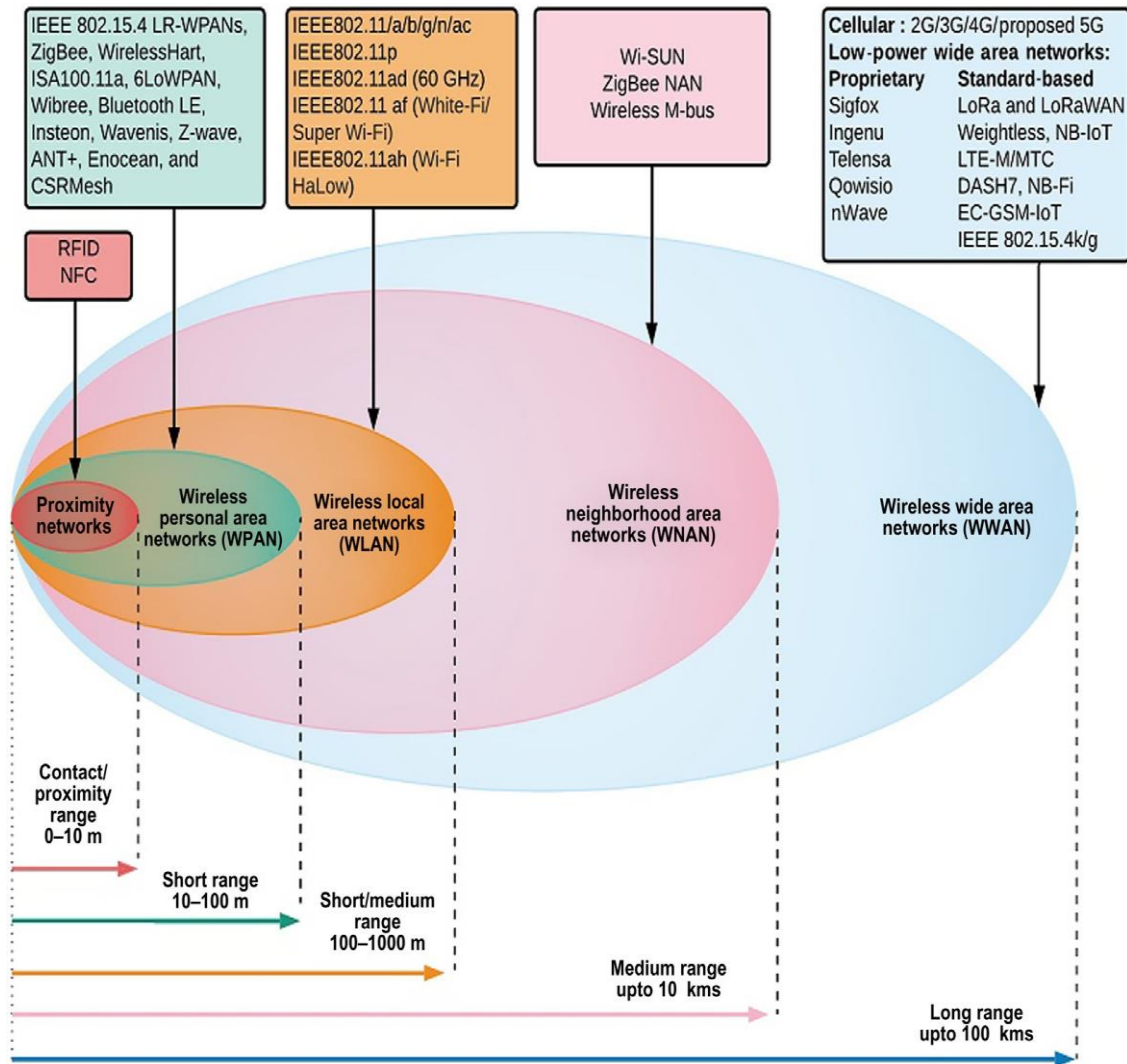
**Figura 1.** Aplicaciones de las tecnologías LPWAN en diferentes sectores.

Fuente: (Raza et al., 2017)

La operación de baja potencia está impulsada principalmente por la disponibilidad de suministro de energía eléctrica, por ejemplo, aplicaciones agrícolas. En tales situaciones, varios sensores están muy lejos y a veces son difíciles de alcanzar, por lo que se necesitan baterías que duren 101 años sin recargarse. La operación de baja potencia se considera de gran importancia en tales aplicaciones. En otros, por ejemplo, el comercio minorista, la energía eléctrica puede estar fácilmente disponible y la operación de baja potencia puede considerarse de baja prioridad. En muchos casos, una aplicación con una gran cantidad de dispositivos requiere dispositivos de muy bajo costo, por ejemplo, medición inteligente, mientras que otros, como los hogares inteligentes, pueden absorber costos razonables. Por lo tanto, esto se captura como de baja relevancia para tales aplicaciones.(Chaudhari & Zennaro, 2020).

### 2.2.1. Acceso inalámbrico

Para las comunicaciones e interconexiones de las aplicaciones de IoT, hay disponible una gama de soluciones patentadas y basadas en estándares. Estas comunicaciones abarcan diferentes rangos geográficos, como se muestra en la figura 2.



**Figura 2.** Acceso inalámbrico en la cobertura geográfica.

Fuente: (Chaudhari & Zennaro, 2020)

Las redes de proximidad inalámbricas basadas en la identificación por radiofrecuencia y la comunicación de campo cercano son las redes de comunicación de tipo red de área cercana para los dispositivos que se encuentran en las proximidades. Los WPAN se



utilizan para transmitir información a distancias cortas entre el grupo de dispositivos participantes con poca o ninguna infraestructura; la mayoría de los WPAN están diseñados para soluciones de baja velocidad de datos, eficiencia energética, corta distancia y de bajo costo. Las WLAN están diseñadas principalmente para el intercambio de datos a alta velocidad entre los dispositivos con una cobertura en todo el campus limitada a unos pocos cientos de metros. La red de área de vecindario inalámbrico (WNAN) ha evolucionado en un nuevo elemento de sistema arquitectónico para aplicaciones de distribución local inalámbrica de banda ancha, que comprende un área de servicio más pequeña que las redes de área metropolitana pero más grande que las de área local (Chaudhari & Zennaro, 2020).

Las WWAN se pueden clasificar en términos generales en celulares y LPWAN. Las redes celulares como 3G y 4G están diseñadas principalmente para transferir datos a alta velocidad durante unos pocos a decenas de kilómetros. Estas redes admiten movilidad y, por lo tanto, proporcionan una cobertura extendida más allá del alcance de una sola célula a través de mecanismos de transferencia. Las LPWAN son las tecnologías de comunicación inalámbrica diseñadas para permitir comunicaciones de largo alcance con bajo consumo de energía, interfaz de bajo costo y una tasa de bits relativamente baja para aplicaciones IoT y M2M (Chaudhari & Zennaro, 2020).

## **2.2.2. Características de LPWAN**

### **2.2.2.1. Cobertura**

LPWAN necesita proporcionar una comunicación de largo alcance de hasta 1040 km en zonas rurales/desérticas y 15 km en zonas urbanas. Las tecnologías LPWAN están diseñadas para una cobertura de área amplia y una excelente propagación de señal a lugares interiores de difícil acceso, como sótanos. Cuantitativamente, se apunta a una ganancia de +20 dB sobre los sistemas celulares heredados (Chaudhari & Zennaro, 2020).



El uso de la banda Sub-GHz, con la excepción de algunas tecnologías LPWAN (por ejemplo, WEIGHTLESS-W e INGENU), que ofrece una comunicación robusta y confiable con presupuestos de baja potencia. Se debe a que, las señales de frecuencia más baja experimentan menos atenuación y desvanecimiento por trayectos múltiples causados por obstáculos y superficies densas como paredes de concreto. Además, la banda sub-GHz está menos congestionado que 2.4 GHz, una banda utilizada por la mayoría de las tecnologías inalámbricas populares. Sin embargo, la tecnología RPMA de INGENU es una excepción que aún explota la banda de 2.4 GHz debido a regulaciones de espectro más relajadas en el ciclo de trabajo de radio y la potencia de transmisión máxima en esta banda en múltiples regiones (Raza et al., 2017).

Las tecnologías LPWAN están diseñadas para lograr un presupuesto de enlace de  $150 \pm 10$  dB que permite un rango de unos pocos kilómetros y decenas de kilómetros en áreas urbanas y rurales, respectivamente. La capa física compromete la alta velocidad de datos y reduce la velocidad de modulación para poner más energía en cada bit (o símbolo) transmitido (Raza et al., 2017)

#### **2.2.2.2. Bajo Costo**

El éxito comercial de las redes LPWAN está vinculado a la conexión de una gran cantidad de dispositivos finales, mientras se mantiene el costo del hardware por debajo de \$ 5. El uso de conectividad de tipo estrella (en lugar de malla), protocolos MAC simples y técnicas para descarga simples permite a los fabricantes diseñar dispositivos finales simples y, por lo tanto, de bajo costo (Raza et al., 2017).

Para manejar la gran cantidad, el bajo costo y la cobertura de largo alcance, el diseño de dispositivos pequeños y de bajo complejo se convierte en un requisito esencial. La estructura de complejidad reducida de las tecnologías LPWAN de hardware para



aplicaciones IoT y M2M permite la reducción del consumo de energía en dispositivos alimentados por batería, sin sacrificar demasiado rendimiento. En general, se espera que los dispositivos posean bajas capacidades de procesamiento. La arquitectura de red simple y los protocolos deben ser compatibles con el hardware. Desde el punto de vista tecnológico, para lograr la adaptabilidad requerida de los dispositivos LPWAN, los transceptores de radio deben ser dispositivos flexibles y reconfigurables por software (Chaudhari & Zennaro, 2020).

### **2.2.2.3. Eficiencia energética**

La operación de bajo consumo de energía es un requisito clave para aprovechar la gran oportunidad de negocio que brindan los dispositivos IoT/M2M que funcionan con baterías. Es deseable una vida útil de la batería de 10 años o más con pilas AA o pilas de moneda para reducir el costo de mantenimiento. Una gama muy larga de tecnologías LPWAN supera estas limitaciones conectando dispositivos finales directamente a las estaciones base. La topología resultante es una estrella que se usa ampliamente en redes celulares y ofrece enormes ventajas de ahorro de energía (Raza et al., 2017).

El ciclo de trabajo permite que los dispositivos finales LPWAN apaguen sus transceptores, cuando no es necesario. Solo cuando los datos se van a transmitir o recibir, el transceptor se enciende. Si una aplicación necesita transferir los datos solo a través del enlace ascendente, los dispositivos finales pueden reactivarse solo cuando los datos estén listos para ser transmitidos. Por el contrario, si también se requieren transmisiones de enlace descendente, los dispositivos finales se aseguran de escuchar cuando la estación base realmente transmite. Los dispositivos finales logran esto acordando un horario de escucha. En el ámbito de las tecnologías LPWAN, el ciclo de trabajo del transceptor de



datos no solo es un mecanismo de ahorro de energía sino también un requisito legislativo (Chaudhari & Zennaro, 2020).

en las redes celulares, sincronizan las estaciones base y el equipo de usuario (UE) con precisión para beneficiarse de esquemas MAC complejos que explotan la diversidad de frecuencia y tiempo; este control de estos protocolos MAC puede ser incluso más costoso que la comunicación de los dispositivos LPWAN. Además, una sincronización muy ajustada que necesitan estos esquemas es difícil de cumplir con dispositivos finales de muy bajo costo (\$ 1- \$ 5) que tienen osciladores baratos de baja calidad. El acceso múltiple de detección de portadora con prevención de colisión (CSMA/CA) es uno de los protocolos MAC más populares implementados en redes inalámbricas de corto alcance. El número de dispositivos por estación base está limitado para tales redes. Sin embargo, a medida que aumenta el número de estos dispositivos en las redes LPWAN, la detección del operador se vuelve menos efectiva y costosa (Xie, Xu, & Lei, 2014). Si bien la detección virtual del operador utilizando el mecanismo RTS/CTS se utiliza para superar este problema, introduce una sobrecarga de comunicación adicional sobre el enlace ascendente y el enlace descendente. Con un número masivo de dispositivos, las tecnologías LPWAN generalmente no pueden permitirse esta sobrecarga de señalización excesiva (Raza et al., 2017).

Debido a estas razones, múltiples tecnologías LPWAN como SIGFOX y LORAWAN recurren al uso de ALOHA, un protocolo MAC de acceso aleatorio en el que los dispositivos finales transmiten sin detectar ningún operador. Esto mantiene el diseño del transceptor simple y de bajo costo. No obstante, INGENU y NB-IoT también consideran los protocolos MAC basados en TDMA para asignar recursos de radio de manera más eficiente, aunque a expensas de una mayor complejidad y costo para los dispositivos finales.





#### **2.2.2.4. Capacidad y escalabilidad.**

Uno de los requisitos esenciales para LPWAN es admitir una gran cantidad de dispositivos conectados simultáneamente con la baja velocidad de datos. Muchas aplicaciones requieren soporte para dispositivos 100,0001 de manera escalable. La escalabilidad se refiere a la capacidad de crecer sin problemas desde una red de una pequeña cantidad de dispositivos heterogéneos a una gran cantidad de dispositivos, nuevos dispositivos, aplicaciones y funciones sin comprometer la calidad y la provisión de servicios existentes. Como los dispositivos finales LPWAN tienen capacidades informáticas y de potencia bajas, los dispositivos de red como las puertas de enlace y las estaciones de acceso también pueden desempeñar un papel vital en la mejora de la escalabilidad. El empleo de multicanal y multiantena basado en diferentes técnicas de diversidad también puede mejorar significativamente la escalabilidad de las redes LPWAN. Una cantidad tan grande de dispositivos también da como resultado una alta densificación. En tal caso, siempre existe la posibilidad de un cuello de botella en el acceso a los medios, grandes interferencias y, por lo tanto, una degradación sustancial del rendimiento de la red (Chaudhari & Zennaro, 2020).

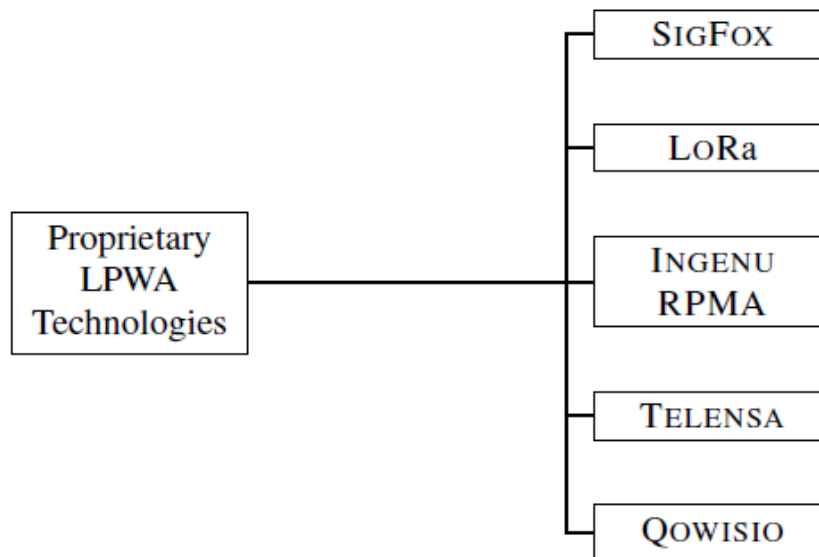
#### **2.2.2.5. Calidad de servicio (QoS)**

Las tecnologías LPWAN se dirigen a un conjunto diverso de aplicaciones con requisitos variables. En un extremo, satisface retrasar las aplicaciones de medición inteligente tolerantes, mientras que en el otro extremo debe entregar las alarmas generadas por las aplicaciones de seguridad doméstica en un tiempo mínimo. Hasta donde se investigó, las tecnologías LPWAN actuales proporcionan QoS nula o limitada (Raza et al., 2017).



### 2.2.3. Tecnologías LPWAN

En esta sección, destacamos las tecnologías en la Figura 3 y sus aspectos, posteriormente en la tabla 1 se resumen las especificaciones técnicas de las tecnologías SIGFOX, LORAWAN, INGENU y TELENDA.



**Figura 3.** Tecnologías LPWAN

Fuente: (Raza et al., 2017)

**Tabla 1:** Especificaciones de algunas tecnologías LPWAN

	<b>SigFox</b>	<b>LoRaWan</b>	<b>INGENU</b>	<b>TELENSA</b>
<b>Modulation</b>	UNB DBPSK(UL)	CSS	RPMA-DSS(UL), CDMA(DL)	UNB 2-FSK
<b>Band</b>	Sub-GHz ISM:EU (868MHz), US(902)MHz	Sub-GHz ISM: EU (433MHz, 868MHz). US(915MHz), ASIA (430MHz)	ISM 2.4GHz	Sub-GHz bands including ISM: EU(858MHz), US(915MHz), ASIA (430 MHz)
<b>Data rate</b>	100 bps(UL), 600 bps(DL)	0.3-37.5 kbps (LoRa), 50kbps (FSK)	78 kbps (DL), 19.5 kbps (DL) (39)	62.5 bps(UL), 500 bps (DL)
<b>Range</b>	10 km (urban), 50 km (rural)	5 km (urban), 15 km (rural)	15 km urban	1 km urban
<b>Num. of channels/orthogonal signals</b>	360 channels	10 in EU, 64+8(UL) and 8(DL) in US plus SFs	40 1MHz channels, up to 1200 signals per channel	multiple channels
<b>Link symmetry</b>	x	√	x	x
<b>Forward error correction</b>	x	√	√	√
<b>MAC</b>	Unslotted-ALOHA	Unslotted-ALOHA	CDMA-like	?
<b>Topology</b>	star	Star of stars	Star, tree	star
<b>Adaptive Data Rate</b>	x	√	√	x
<b>Payload length</b>	12B (UL), 8B(DL)	Up to 250B (depends on SF & region)	10KB	?
<b>Handover</b>	End devices do not join a single base station	End devices do not join a single base station	√	?
<b>Authentication &amp; encryption</b>	Encryption not supported	AES 128b	16B hash, AES 256b	?
<b>Over the air updates</b>	x	√	√	√
<b>SIA support</b>	x	x	x	x
<b>Location</b>	x	√	x	x

Fuente: (Raza et al., 2017)

### 2.2.3.1. SIGFOX

Sigfox es un operador de red LPWAN que ofrece una solución de conectividad IoT de extremo a extremo basada en sus tecnologías patentadas como se muestra en la figura 4. Sigfox implementa sus estaciones base patentadas equipadas con radios definidas por software y las conecta a los servidores de back-end utilizando una red basada en IP. Los dispositivos finales conectados a estas estaciones base utilizan modulación de desplazamiento de fase binaria (BPSK) en una portadora de banda subGHZ ISM de banda ultra estrecha (100 Hz). Sigfox utiliza bandas ISM sin licencia, por ejemplo, 868 MHz en Europa, 915 MHz en América del Norte y 433 MHz en Asia. Al emplear la banda ultra estrecha, Sigfox utiliza el ancho de banda de frecuencia de manera eficiente y experimenta niveles de ruido muy bajos, lo que lleva a un consumo de energía muy bajo,



una alta sensibilidad del receptor y un diseño de antena de bajo costo a expensas del rendimiento máximo de solo 100 bps (Mekki, Bajic, Chaxel, & Meyer, 2019).

Inicialmente, Sigfox solo admitía la comunicación de enlace ascendente, pero luego evolucionó a tecnología bidireccional con una asimetría de enlace significativa. La comunicación de enlace descendente, es decir, los datos de las estaciones base a los dispositivos finales solo pueden ocurrir después de una comunicación de enlace ascendente. El número de mensajes a través del enlace ascendente está limitado a 140 mensajes por día. La longitud máxima de carga útil para cada mensaje de enlace ascendente es de 12 bytes. Sin embargo, el número de mensajes a través del enlace descendente está limitado a cuatro mensajes por día, lo que significa que no se admite el reconocimiento de cada mensaje de enlace ascendente. La longitud máxima de carga útil para cada mensaje de enlace descendente es de ocho bytes. Cada mensaje del dispositivo final se transmite varias veces (tres por defecto) a través de diferentes canales de frecuencia. Para este propósito, en Europa, por ejemplo, la banda entre 868.180 MHz y 868.220 MHz se divide en 400 canales ortogonales de 100 Hz (entre ellos, 40 canales están reservados y no se usan). Como las estaciones base pueden recibir mensajes simultáneamente en todos los canales, el dispositivo final puede elegir aleatoriamente un canal de frecuencia para transmitir sus mensajes. Esto simplifica el diseño del dispositivo final y reduce su costo (Muteba, Djouani, & Olwal, 2019).



**Figura 4.** Arquitectura SIGFOX

FUENTE: (<https://www.sigfox.com/en/what-sigfox/technology>, 2020)

#### **2.2.3.2. LORA**

Esta tecnología se desarrollará en la sección 2.3, ya que es utilizada para esta investigación.

#### **2.2.3.3. INGENU RPMA**

INGENU (anteriormente conocido como On-Ramp Wireless) propuso una tecnología LPWAN patentada, que a diferencia de la mayoría de las otras tecnologías no se basa en mejores propiedades de propagación de la banda subGHZ. En cambio, opera en la banda ISM de 2.4 GHz y aprovecha las regulaciones libres sobre el uso del espectro en diferentes regiones. Para ofrecer un ejemplo, las regulaciones en EE. UU. Y Europa no imponen un límite máximo en el ciclo de trabajo para la banda de 2.4 GHz, lo que permite un mayor rendimiento y más capacidad que otras tecnologías que operan en la banda subGHZ (Centenaro, Vangelista, Zanella, & Zorzi, 2016).

Lo que es más importante, INGENU utiliza un esquema de acceso físico patentado denominado Random Phase Multiple Access (RPMA), que emplea solo para la comunicación de enlace ascendente. Como una variación del Acceso Múltiple por División de Código (CDMA), RPMA permite que múltiples transmisores compartan un



solo intervalo de tiempo. Sin embargo, RPMA primero aumenta la duración del intervalo de tiempo del CDMA tradicional y luego dispersa el acceso al canal dentro de este intervalo al agregar un retraso de desplazamiento aleatorio para cada transmisor. Al no otorgar acceso de canal a los transmisores exactamente a la vez (es decir, al comienzo de un intervalo), RPMA reduce la superposición entre las señales transmitidas y, por lo tanto, aumenta la relación señal/interferencia para cada enlace individual. En el lado receptor, las estaciones base emplean demoduladores múltiples para decodificar las señales que llegan en diferentes momentos dentro de una ranura. INGENU proporciona comunicación bidireccional, aunque con una ligera asimetría de enlace. Para la comunicación de enlace descendente, las estaciones base difunden las señales para dispositivos finales individuales y luego las transmiten usando CDMA. Se informa que RPMA alcanza hasta -142 dBm de sensibilidad del receptor y 168 dB de presupuesto de enlace. Además, los dispositivos finales pueden ajustar su potencia de transmisión para llegar a la estación base más cercana y limitar la interferencia a los dispositivos cercanos (Raza et al., 2017).

INGENU lidera los esfuerzos para estandarizar las especificaciones de la capa física bajo el estándar IEEE 802.15.4k.

#### **2.2.3.4.       TELENSA**

TELENSA proporciona soluciones de extremo a extremo para aplicaciones LPWAN que incorporan pilas de red verticales totalmente diseñadas con soporte para la integración con software de terceros. Para una conectividad inalámbrica entre sus dispositivos finales y las estaciones base, TELENSA diseñó una técnica patentada de modulación UNB, que opera en la banda subGHZ ISM sin licencia a bajas velocidades de datos. Si bien se sabe menos acerca de la implementación de su tecnología inalámbrica, TELENSA tiene como objetivo estandarizar su tecnología utilizando las especificaciones de redes de bajo rendimiento ETSI para una fácil integración dentro de las aplicaciones (Telensa, 2020).



TELENSA actualmente se enfoca en algunas aplicaciones de ciudades inteligentes como iluminación inteligente, estacionamiento inteligente, etc. Para fortalecer sus ofertas de LPWAN en el negocio de iluminación inteligente, TELENSA está involucrada con el consorcio TALQ en la definición de estándares para monitorear y controlar los sistemas de iluminación exterior (Raza et al., 2017).

#### **2.2.3.5. QOWISIO**

QOWISIO despliega redes LPWAN de modo dual que combinan su propia tecnología patentada UNB con LORA. Proporciona conectividad LPWAN como un servicio para los usuarios finales: ofrece dispositivos finales, implementa infraestructura de red, desarrolla aplicaciones personalizadas y las aloja en una nube de back-end. Sin embargo, se sabe menos acerca de las especificaciones técnicas de su tecnología UNB subyacente y otros componentes del sistema (Raza et al., 2017).

### **2.3. LORA y LORAWAN**

LoRaWAN es una arquitectura de sistema inalámbrico de extremo a extremo que proporciona una solución de conectividad de bajo consumo, larga duración, bajo costo, segura y escalable para operadores públicos y redes privadas con una amplia gama de casos de uso de IoT. La arquitectura LoRaWAN utiliza la capa física LoRa basada en la modulación de espectro extendido (Pham, Bounceur, Clavier, Noreen, & Ehsan, 2020).

LoRaWAN permite que los dispositivos finales funcionen con baterías pequeñas por hasta 10 años, para lo cual utiliza puertas de enlace de radio con un alcance de hasta 30 millas (más de 48 Km) en áreas rurales. Es capaz de penetrar ambientes urbanos densos e interiores profundas. LoRaWAN se basa en el Estándar de cifrado avanzado de 128 bits (AES128) para garantizar la seguridad total de la red, incluida la autenticación mutua de punto final, la autenticación del origen de datos, la reproducción y la protección de la



integridad y la privacidad. Su uso de bandas de radio industriales, científicas y médicas (ISM) permite una alta capacidad, operación de bajo costo y diseño específico en los requisitos de IoT. También se basa en la disponibilidad de estándares abiertos y un ecosistema abierto (Pham et al., 2020).

La arquitectura del sistema y los protocolos están siendo desarrollados por LoRa Alliance (LoRaAlliance.org), una asociación abierta y sin fines de lucro con un ecosistema grande y en constante crecimiento que abarca una amplia gama de entidades, desde fabricantes de chips hasta proveedores de la nube. LoRa Alliance facilita la producción de especificaciones de interoperabilidad que están disponibles públicamente y de forma gratuita, un programa de certificación para respaldar la proliferación de dispositivos de manera eficiente (Yegin et al., 2020).

La primera especificación desarrollada por LoRa Alliance, es la especificación de capa de enlace LoRaWAN, que describe la capa que reside sobre la capa física de LoRa y debajo de la capa de aplicación entre el dispositivo final y la red. Esta capa de enlace, que actúa como un transporte por aire, garantiza que los dispositivos finales puedan enviar y recibir cargas útiles de la capa de aplicación (Zhou, Xing, Hou, Xu, & Zheng, 2019).

La especificación de la capa de enlace LoRaWAN anterior a la Versión 1.0.2 incluía los parámetros de la capa física que varían según las regiones reguladoras, como las frecuencias de canal, la potencia de transmisión y las velocidades de datos. Más tarde, LoRa Alliance separó estas especificaciones en un documento dedicado de parámetros regionales que puede evolucionar en respuesta a cambios regulatorios y la adición de nuevas regiones (Croce, Gucciardo, Santaromita, Mangione, & Tinnirello, 2020).

A medida que la arquitectura LoRaWAN evolucionó, las especificaciones de la interfaz de back-end LoRaWAN se introdujeron posteriormente. A medida que las redes

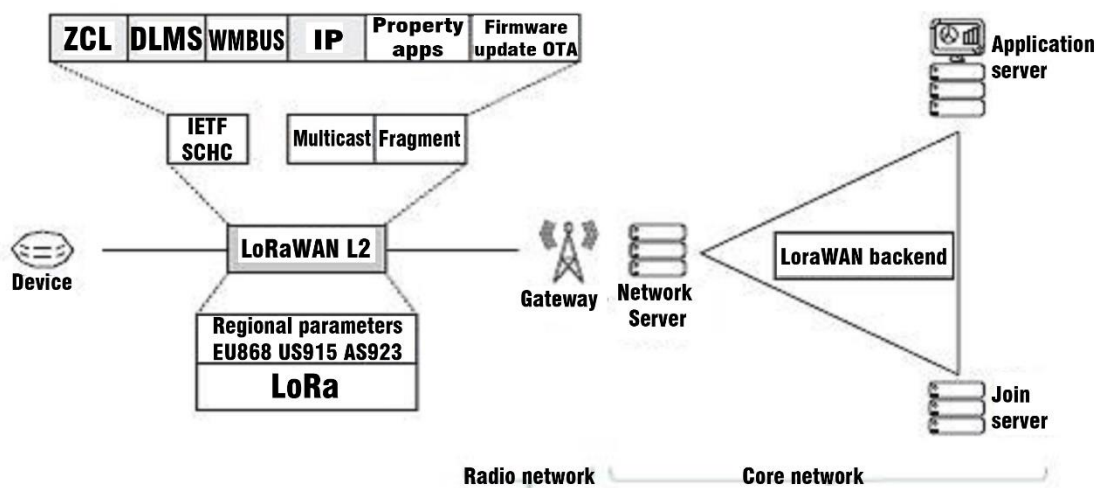


comenzaron a estar disponibles en todo el mundo, el siguiente paso lógico fue hacer que estas redes colaboren mediante el establecimiento de capacidades de itinerancia mutua. La integración de redes de acceso de radio a través de múltiples redes LoRaWAN requería una interfaz interoperable entre sus redes centrales. Además, para facilitar el aprovisionamiento de dispositivos, las redes centrales LoRaWAN también evolucionaron para externalizar la entidad que puede almacenar las credenciales a largo plazo de los dispositivos al tiempo que les permite activarse en cualquier red del mundo. Las especificaciones de la interfaz de back-end describen los protocolos necesarios para que la itinerancia y la activación se realicen en dominios administrativos separados que utilizan plataformas de varios proveedores independientes (Yegin et al., 2020).

Uno de los habilitadores claves para lograr dispositivos de bajo costo es la capacidad de implementarlos una vez y luego no tener que lidiar con ellos durante 510 años. El software instalado en estos dispositivos, ya sea en la pila de aplicaciones o en el firmware, requiere actualizaciones por las razones obvias de la entrega de nuevas funciones y correcciones de errores. Por esa razón, es crítico que los dispositivos IoT tengan un mecanismo de actualización de firmware por aire. La función de actualización inalámbrica de firmware de LoRaWAN (FUOTA) se basa en dos especificaciones, una para la fragmentación del enlace descendente y la otra para la gestión de multidifusión. Estas especificaciones definen un mecanismo seguro, confiable y eficiente para transportar archivos grandes simultáneamente a un conjunto de dispositivos. La confianza en la corrección simultánea de errores hacia adelante y la transmisión física ha sido un elemento clave para lograr estas características bajo las restricciones de las regulaciones de banda ISM. Los protocolos de fragmentación y multidifusión se han definido en la capa de aplicación (es decir, por encima de la capa de enlace) para que sean aplicables a las versiones heredadas de las capas de enlace LoRaWAN (Croce et al., 2020).



La figura 5, muestra los diversos elementos de red que conforman la arquitectura de red LoRaWAN, así como los protocolos y especificaciones para soportarla. LoRa Alliance presta especial atención a mantener la complejidad lo más baja posible, al tiempo que puede abordar las necesidades cambiantes del mercado. Un diseño básico que siga una filosofía de diseño sólida es imprescindible para el éxito de cualquier arquitectura de sistema LPWAN que operen en bandas de ISM. Los estándares abiertos respaldados por implementaciones de código abierto y hardware de bajo costo también son herramientas esenciales para este propósito. LoRa Alliance se ha destacado por aprovechar estos elementos para construir una tecnología a prueba de futuro y al mismo tiempo nutrir un ecosistema cada vez mayor a su alrededor (Yegin et al., 2020).



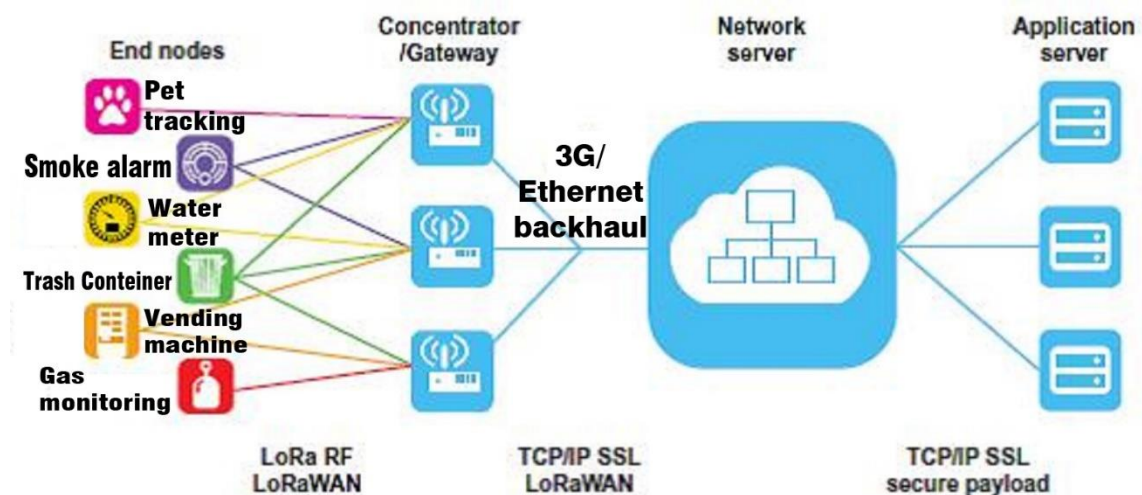
**Figura 5.** Arquitectura y protocolos de LoRaWAN

Fuente: (Yegin et al., 2020)

### 2.3.1. Capa de enlace de LoRaWAN

Las redes LoRaWAN se presentan en una topología de star-of-stars (estrella de estrellas) en la que las puertas de enlace (Gateway) retransmiten paquetes entre los dispositivos y un servidor de red central (NS – Network Server). El NS, a su vez, enruta los paquetes recibidos por las puertas de enlace a un servidor de aplicaciones asociado y

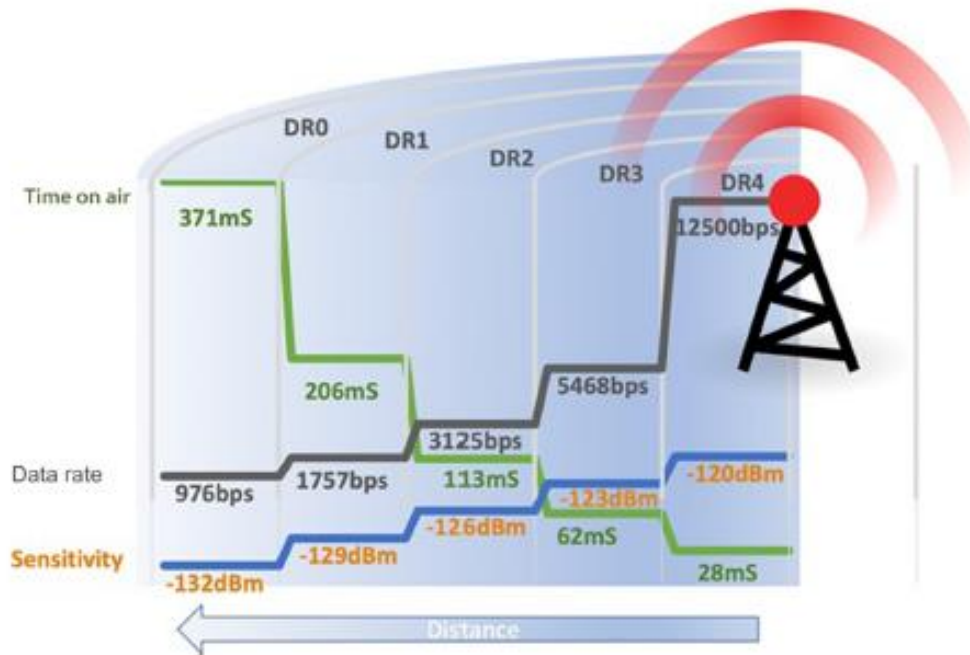
viceversa. La comunicación es generalmente bidireccional, aunque se espera que la comunicación de enlace ascendente desde un dispositivo a la red y los servidores de aplicaciones sea el tráfico predominante. Además, los enlaces ascendentes pueden ser recibidos por múltiples puertas de enlace, es decir, no existe una asociación fija entre dispositivos y puertas de enlace, como se muestra en la figura 6 (Yegin et al., 2020).



**Figura 6.** Red LoRaWAN

Fuente: (Yegin et al., 2020)

La comunicación entre dispositivos y puertas de enlace utiliza transmisiones de radio LoRa y modulación FSK, mientras que las transmisiones LoRa se distribuyen en varios canales de frecuencia y velocidades de datos. Seleccionar la velocidad de datos, que varía de 0.3 a 50 kbps, es una compensación entre el rango de comunicación y la duración de la transmisión de paquetes. Las tasas de datos más bajas tienen un mayor alcance, pero consumen más tiempo de emisión. Para maximizar tanto la duración de la batería de los dispositivos como la capacidad general de la red, la infraestructura de red LoRaWAN puede administrar la velocidad de datos y la potencia de salida de radiofrecuencia (RF) para cada dispositivo individualmente mediante un esquema de velocidad de datos adaptativa (ADR) como se ilustra en figura 7 (Yegin et al., 2020).



**Figura 7.** Las velocidades de datos en función del alcance de la comunicación y la duración de la transmisión de paquetes

Fuente: (Yegin et al., 2020)

Los dispositivos pueden transmitir por cualquier canal disponible en cualquier momento utilizando cualquier velocidad de datos disponible, siempre que se observen las siguientes reglas:

1. El dispositivo cambia de canal de forma pseudoaleatoria para cada transmisión de radio. La diversidad de frecuencias resultante hace que el sistema sea más resistente a las interferencias.
2. El dispositivo respeta el ciclo de trabajo máximo de transmisión relativo a la sub-banda en la que está operando y cumple con las regulaciones locales.
3. El dispositivo respeta la duración máxima de transmisión (o tiempo de permanencia) relativa a la sub-banda en la que está operando y cumple con las regulaciones locales.



Para las transmisiones de radio seguras, el protocolo LoRaWAN se basa en una criptografía simétrica que utiliza claves de sesión derivadas de claves de raíz específicas del dispositivo. En el back-end, las claves de raíz del dispositivo y las operaciones de derivación de claves asociadas se almacenan en un servidor de unión (JS – Join Server) durante un procedimiento de activación por aire. Alternativamente, las claves de sesión específicas del dispositivo pueden fabricarse directamente en el dispositivo, lo que se conoce como activación por personalización (Pham et al., 2020).

Los dispositivos LoRaWAN suelen seguir un patrón de comunicación de tipo ALOHA, en el que los dispositivos pueden funcionar en una de las tres clases siguientes:

- **Clase A:** Dispositivos bidireccionales. Los dispositivos de clase A permiten la comunicación bidireccional. El funcionamiento de clase A es el sistema de dispositivo de menor potencia para aplicaciones que sólo requieren una comunicación de enlace descendente desde el NS poco después de que el dispositivo haya enviado una transmisión de enlace ascendente. Las comunicaciones de enlace descendente desde el NS en cualquier otro momento deben esperar hasta el siguiente enlace ascendente programado.
- **Clase B:** Dispositivos bidireccionales con ranuras de recepción programadas. Además, de las ventanas de recepción aleatorias de clase A, los dispositivos de clase B abren más ventanas de recepción a horas programadas. Para que el dispositivo abra su ventana de recepción a la hora programada, recibe una baliza sincronizada con la hora de una puerta de enlace.
- **Clase C:** Dispositivos bidireccionales con ranuras de recepción máximas. Los dispositivos de clase C tienen ventanas de recepción casi continuamente abiertas que se cierran sólo cuando transmiten. Utilizan más energía que los



dispositivos de clase A o clase B, pero tienen la latencia más baja para la comunicación entre el servidor y el dispositivo final.

### **2.3.2. Escalabilidad en las redes de LoRaWAN**

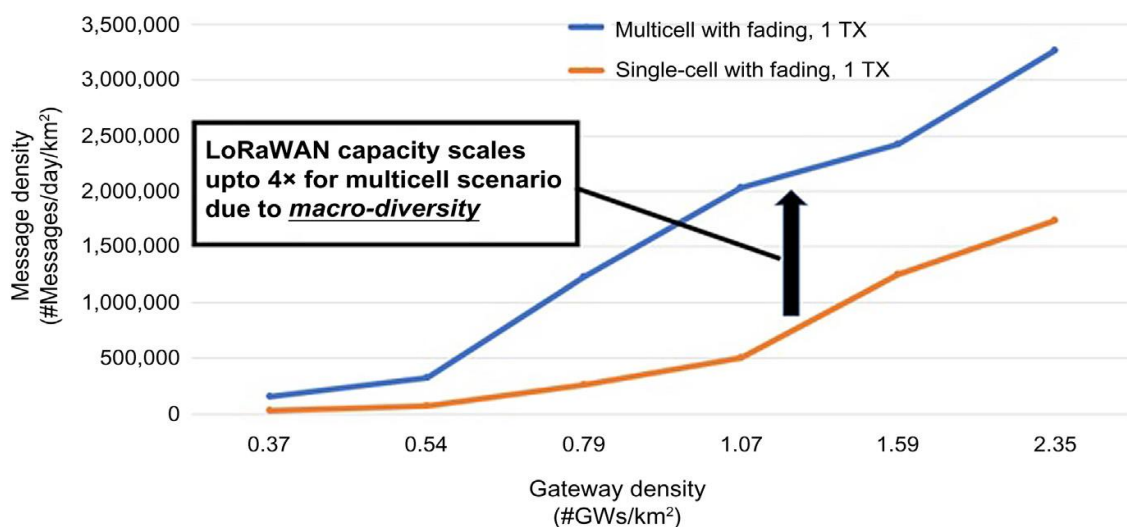
Para dar cabida a esta tendencia, la capacidad y la escala de los despliegues de LoRaWAN han sido un tema de investigación tanto en el ámbito académico como en el industrial. Varias publicaciones recientes han presentado los resultados iniciales con respecto a la capacidad de la LoRaWAN (Barro, Zennaro, & Pietrosevoli, 2019).

la característica de diseño más importante de LoRaWAN es su diversidad de recepción. A medida que crece el uso de espectros sin licencia, el ruido de radio de fondo está aumentando. Algunos expertos predicen que las redes sin licencia se enfrentarán inevitablemente a una creciente pérdida de paquetes y, por lo tanto, no pueden garantizar la calidad de servicio (QoS) a largo plazo. Pero esto no es de hecho inevitable. Las redes LoRaWAN pueden adaptarse al ruido aprovechando las múltiples pasarelas de recepción que funcionan simultáneamente para cada dispositivo final. Los mensajes de enlace ascendente de las redes LoRaWAN pueden ser recibidos por cualquier pasarela (macrodiversidad de RX). Esta macrodiversidad de enlace ascendente mejora significativamente la capacidad y la calidad de servicio de la red porque es muy poco probable que se produzcan interferencias destructivas en todas las antenas simultáneamente (Barro et al., 2019).

LoRaWAN también soporta ADR, que permite a NS cambiar dinámicamente los parámetros de los dispositivos finales como la potencia de transmisión, la lista de frecuencias, el factor de dispersión y la tasa de repetición del enlace ascendente. El ajuste cuidadoso de la potencia de transmisión es necesario no sólo para asegurar que los dispositivos utilizarán la menor potencia necesaria para comunicarse con la pasarela, sino

también para minimizar el ruido innecesario para las células de radio adyacentes, y así evitar que los dispositivos más cercanos tengan un efecto de sombra en los dispositivos situados en el borde de la célula (Yuan, Wen, Lu, & Pan, 2018).

La Figura 8, muestra los resultados de la capacidad máxima de la red y de una sola célula. Confirma que la capacidad se escala fácilmente con la densidad de las pasarelas de la red.



**Figura 8.** Impacto del despliegue de multicelular, en la capacidad.

Fuente: (Yegin et al., 2020)

LoRaWAN proporciona una solución de conectividad horizontal para atender las amplias necesidades de las aplicaciones de IoT para despliegues de LPWAN. Sin embargo, estos beneficios sólo son posibles con algoritmos inteligentes de NS.

### 2.3.3. Parámetros regionales de LoRaWAN.

Un dispositivo o red LoRaWAN usa bandas ISM sin licencia. Por lo tanto, para que un dispositivo funcione en cualquier parte del mundo, debe cumplir con los requisitos reglamentarios de esa región o territorio. En enero de 2019 había 195 países en el mundo, además de varias docenas de otras entidades, 193 de las cuales son miembros de las Naciones Unidas (ONU) (Yegin et al., 2020).



La Unión Internacional de Telecomunicaciones gestiona el uso espectral común para la ONU, pero la autoridad de asignación de frecuencias definitiva la otorgan los gobiernos de cada país o región. Sin embargo, muchos países pequeños o en desarrollo no disponen de una autoridad de asignación de frecuencias. Esto crea un escenario complicado para los despliegues de LoRaWAN en todo el mundo. Se necesita un entendimiento en todos los países y regiones, para desplegar LoRaWAN de manera que se pueda obtener el cumplimiento de la normativa y se puedan crear planes de canales LoRaWAN (Yegin et al., 2020).

Para definir los planes de canales LoRaWAN en cada país o región, la LoRa Alliance ha formado un grupo de trabajo que crea, gestiona y mantiene las especificaciones de los parámetros regionales de LoRaWAN. Debido a estas diferentes entidades reguladoras regionales, no es posible proporcionar un plan de canales único para todo el mundo. No obstante, el grupo de trabajo sobre parámetros regionales de la LoRaWAN ha combinado varios países en diversos planes regionales para reducir el número de regiones de la LoRaWAN. Si un país o región no está aún cubierto por una de las 12 regiones actuales de la LoRaWAN, puede ser posible incluirlo en una región existente, dependiendo de su uso reglamentario de las bandas ISM (Croce et al., 2020).

#### **2.3.4. Activación y Roaming en LoRaWAN**

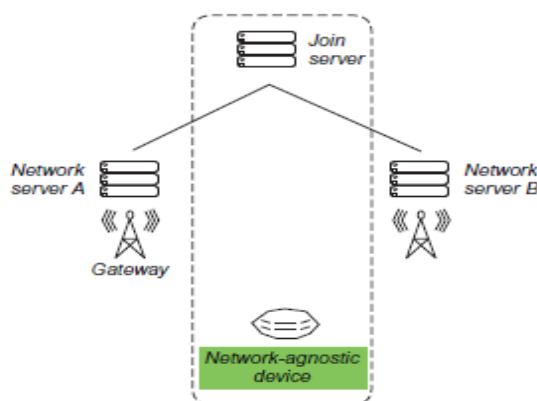
En los primeros días de LoRa Alliance, sólo había una especificación de interoperabilidad llamada capa de enlace, y trataba sólo dos puntos finales: el dispositivo final y la red. A medida que la tecnología, las implementaciones y los despliegues comenzaron a madurar, se hizo evidente la necesidad de descomponer la "red", lo que dio lugar a la definición de tres elementos de red distintos que forman el núcleo de la red LoRaWAN: servidores de unión, servidores de red y servidores de aplicación (Croce et al., 2020).



Los servidores de unión almacenan las credenciales de los dispositivos finales para realizar una autenticación mutua antes de que el dispositivo final sea admitido en la red. El procedimiento que implica un apretón de manos criptográfico y la configuración del dispositivo final con parámetros de red se denomina procedimiento de activación o unión (Song, Lin, Tang, & Dong, 2017).

Los servidores de red, terminan la capa de enlace LoRaWAN en el lado de la red y actúan como puentes que transportan el tráfico de datos a los servidores de aplicaciones y señalan el tráfico al servidor de unión.

Debido a las limitaciones del uso de la banda ISM, que especifican que las tramas sean muy pequeñas en tamaño y número, LoRaWAN utiliza criptografía simétrica. Esto requiere que una clave de raíz simétrica específica del dispositivo que esté disponible en la red antes de que el dispositivo intente activarse. La especificación de la interfaz del backend LoRaWAN hizo esto posible al definir e implementar una interfaz estándar entre NS y JS; proporcionando credenciales del dispositivo en una JS centralizada y permitir que el dispositivo se activara en cualquier red o NS en cualquier lugar del mundo (al igual que en la Figura 9) (Yegin et al., 2020).



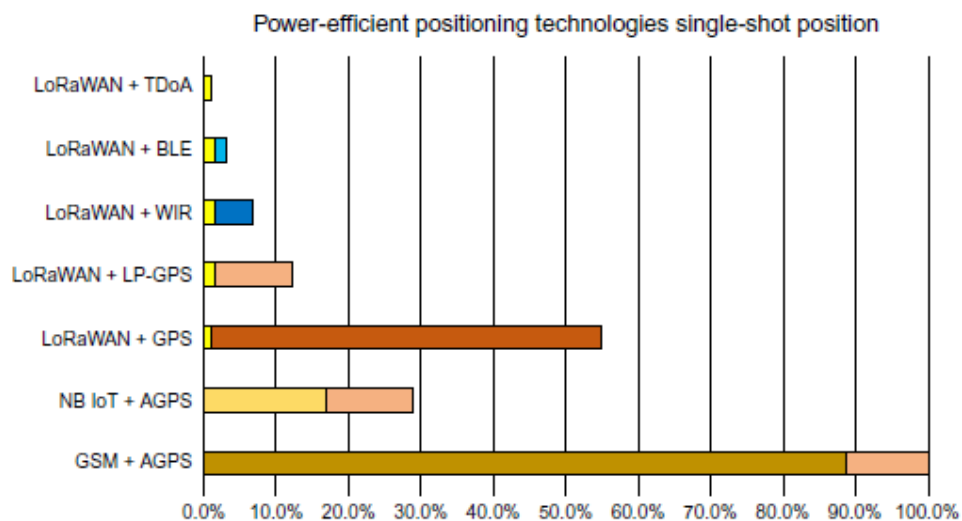
**Figura 9.** Separación de JS de los servidores de la red para el procedimiento de activación.

Fuente: (Yegin et al., 2020)



### 2.3.5. Geolocalización con LoRaWAN.

Como se ilustra en la Figura 10, existen numerosas combinaciones tecnológicas potenciales que servirán para satisfacer la necesidad de geolocalización de activos. Las barras horizontales indican cuánta energía es necesaria, traducida en el tamaño de la batería o en la frecuencia de sustitución de la misma, que son ambos impulsores clave del coste total de propiedad de la geolocalización (Yegin et al., 2020).



**Figura 10.** Uso de energía por la combinación de tecnologías.

Fuente: (Yegin et al., 2020)

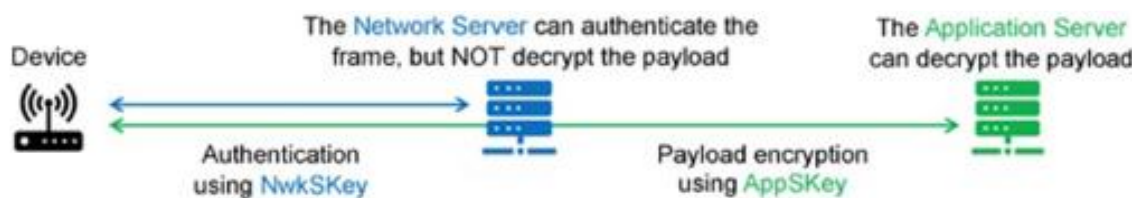
Resulta inmediatamente sorprendente que el uso de LoRaWAN como tecnología de retroceso de área amplia para transferir datos de localización a una aplicación es una mejora significativa con respecto a las redes celulares tradicionales, y sigue siendo significativamente superior a las tecnologías celulares de baja potencia más recientes.

La triangulación basada en la diferencia de tiempo de llegada (TDoA) es una característica atractiva de las redes LoRaWAN, gracias al ancho de banda relativamente amplio de los canales de LoRa (125 kHz o más), que permite marcar la hora de los paquetes con una precisión de nanosegundos. Como no se requiere ningún tipo de procesamiento en el dispositivo, se trata con mucho de la tecnología de mayor eficiencia

energética que presenta el menor costo total de propiedad de todas las opciones. Sin embargo, debido a los multi-trayectos (es decir, las señales de RF pueden rebotar, por lo que la señal recibida no se desplaza necesariamente en línea recta), la precisión de la geolocalización es inferior a la del GPS (Yegin et al., 2020).

### 2.3.6. Seguridad en LoRaWAN

Las propiedades fundamentales que respalda la seguridad de la LoRaWAN son la autenticación mutua, la protección de la integridad y la confidencialidad, véase la Figura 11.



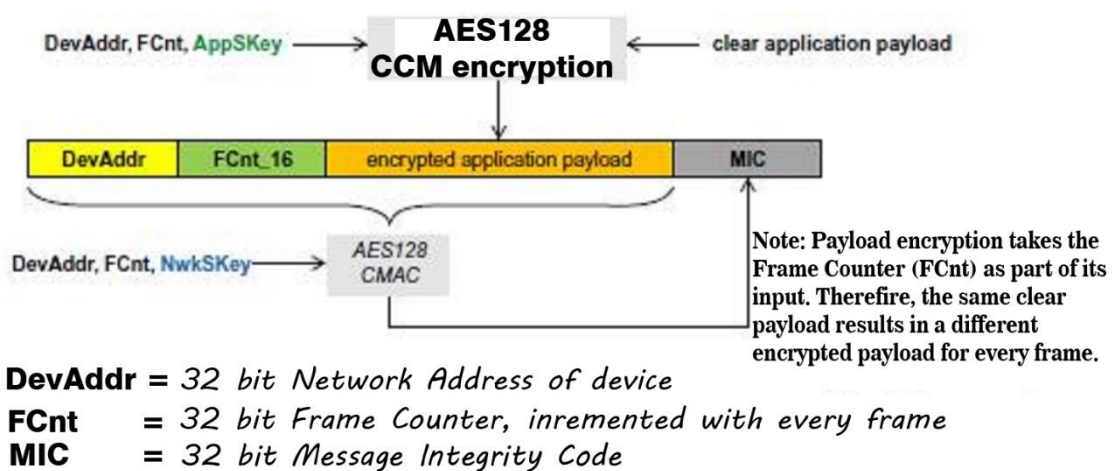
**Figura 11.** Autenticación mutua y encriptación de extremo a extremo proporcionada por la seguridad de LoRaWAN.

Fuente: (Pham et al., 2020)

La autenticación mutua se establece entre un dispositivo LoRaWAN y la red LoRaWAN como parte del procedimiento de unión de redes. Esto asegura que sólo los dispositivos genuinos y autorizados se unan a las redes genuinas y auténticas. Los paquetes MAC y de aplicación de LoRaWAN están autenticados en origen, protegidos por integridad, protegidos contra la reproducción y encriptados. Esta protección, combinada con la autenticación mutua, asegura que el tráfico de la red no se altere, se origine en un dispositivo legítimo, no sea comprensible para los fisgones y no sea capturado y reproducido por actores deshonestos (Yegin et al., 2020).

Los mecanismos de seguridad mencionados anteriormente se basan en los algoritmos criptográficos AES bien probados y normalizados. En particular, la seguridad de

LoRaWAN utiliza la primitiva criptográfica AES combinada con varios modos de operación: código de autenticación de mensajes basado en el cifrado (CMAC) para la protección de la integridad y el cifrado en counter-mode (CTR). En otras palabras, todo el tráfico de LoRaWAN está protegido mediante dos claves de sesión, como se ilustra en la Figura 12. Cada carga útil está cifrada por AES-CTR y lleva un contador de tramas para evitar la repetición de paquetes y un código de integridad del mensaje calculado con AES-CMAC para evitar la manipulación de paquetes (Yegin et al., 2020).



**Figura 12.** Mecanismo de seguridad LoRaWAN basado en algoritmos criptográficos de encriptación avanzadas.

Fuente: (Yegin et al., 2020)

Cada dispositivo LoRaWAN se personaliza con una clave AES única de 128 bits llamada AppKey y un identificador único global (DevEUI basado en EUI-64), ambos utilizados durante el proceso de autenticación del dispositivo. La asignación de los identificadores EUI-64 requiere que el cedente tenga un identificador único para la organización emitido por la Autoridad de Registro del IEEE. Análogamente, las redes LoRaWAN se identifican mediante un identificador único global de 24 bits asignado por la LoRa Alliance (LoRa Alliance<sup>TM</sup>, 2018).

### 2.3.7. Activación del dispositivo final

Para participar en una red LoRaWAN, cada dispositivo final tiene que ser personalizado y activado. La activación de un dispositivo final puede lograrse de dos maneras, ya sea mediante la activación por aire (OTAA) cuando se despliega o reinicia un dispositivo final, o mediante la activación por personalización (ABP) en la que los dos pasos de personalización y activación del dispositivo final se realizan en un solo paso. Después de la activación, la siguiente información se almacena en el dispositivo final: una dirección de dispositivo (DevAddr), un identificador de aplicación (AppEUI), una clave de sesión de red (NwkSKey) y una clave de sesión de aplicación (AppSKey) (LoRa Alliance™, 2018).

#### 2.3.7.1. Dirección del dispositivo final (DevAddr)

El DevAddr consiste en 32 bits que identifican el dispositivo final dentro de la red actual. Su formato es el siguiente:

**Tabla 2:** Formato de dirección del dispositivo final.

Bit	[31..25]	[24..0]
DevAddr bits	NwkID	NwkAddr

Fuente: (LoRa Alliance™, 2018)

Los 7 bits más importantes se utilizan como identificador de red (NwkID) para separar las direcciones de las redes que se superponen territorialmente de los diferentes operadores de red y para remediar los problemas de itinerancia. Los 25 bits menos significativos, la dirección de red (NwkAddr) del dispositivo final, pueden ser asignados arbitrariamente por el administrador de la red (LoRa Alliance™, 2018).



### **2.3.7.2. Identificador de la aplicación (AppEUI)**

El AppEUI es un ID de aplicación global en el espacio de direcciones IEEE EUI64 que identifica de forma exclusiva la entidad capaz de procesar la trama JoinReq. El AppEUI se almacena en el dispositivo final antes de que se ejecute el procedimiento de activación (LoRa Alliance<sup>TM</sup>, 2018).

### **2.3.7.3. Clave de sesión de red (NwkSKey)**

La NwkSKey es una clave de sesión de red específica para el dispositivo final. Es utilizada tanto por el servidor de la red como por el dispositivo final para calcular y verificar el MIC (código de integridad de los mensajes) de todos los mensajes de datos para asegurar la integridad de los datos. Se utiliza además para cifrar y descifrar el campo de carga de los mensajes de datos de un MAC solamente (LoRa Alliance<sup>TM</sup>, 2018).

### **2.3.7.4. Clave de sesión de aplicación (AppSKey)**

La AppSKey es una clave de sesión de aplicación específica para el dispositivo final. Es utilizada tanto por el servidor de aplicaciones como por el dispositivo final para cifrar y descifrar el campo de carga de los mensajes de datos específicos de la aplicación. Las cargas útiles de las aplicaciones se cifran de extremo a extremo entre el dispositivo final y el servidor de aplicaciones, pero no están protegidas por la integridad. Esto significa que un servidor de red puede alterar el contenido de los mensajes de datos en tránsito. Los servidores de red se consideran de confianza, pero se recomienda a las aplicaciones que deseen aplicar la confidencialidad y la protección de la integridad de extremo a extremo que utilicen soluciones de seguridad adicionales de extremo a extremo, que están fuera del alcance de esta especificación (LoRa<sup>TM</sup> Alliance, 2016).



### 2.3.7.5. OTAA, Activación por aire

Para la activación por aire, los dispositivos finales deben seguir un procedimiento de unión antes de participar en los intercambios de datos con el servidor de la red. Un dispositivo final tiene que pasar por un nuevo procedimiento de unión cada vez que ha perdido la información de contexto de la sesión.

El procedimiento de unión requiere que el dispositivo final se personalice con la siguiente información antes de iniciar el procedimiento de unión: un identificador único global del dispositivo final (DevEUI), el identificador de la aplicación (AppEUI) y una clave AES-128 (AppKey).

Para la activación por aire, los dispositivos finales no están personalizados con ningún tipo de clave de red. En cambio, cada vez que un dispositivo final se une a una red, se obtiene una clave de sesión de red específica para ese dispositivo final para cifrar y verificar las transmisiones a nivel de red. De esta forma se facilita la itinerancia de los dispositivos finales entre redes de distintos proveedores. La utilización tanto de una clave de sesión de red como de una clave de sesión de aplicación permite además federar servidores de red en los que los datos de aplicación no pueden ser leídos ni manipulados por el proveedor de la red (LoRa Alliance™, 2018).

**Identificador del dispositivo terminal (DevEUI):** El DevEUI es una identificación global de dispositivo final en el espacio de direcciones IEEE EUI64 que identifica de forma exclusiva el dispositivo final.

**Clave de aplicación (AppKey):** La AppKey es una clave de raíz AES-128 específica para el dispositivo final. Cada vez que un dispositivo final se une a una red a través de una activación por aire, la AppKey se utiliza para derivar las claves de sesión NwkSKey

y AppSKey específicas para ese dispositivo final para encriptar y verificar la comunicación de la red y los datos de las aplicaciones.

**Procedimiento de unión:** Desde el punto de vista de un dispositivo final, el procedimiento de unión consiste en dos mensajes MAC intercambiados con el servidor, a saber, una solicitud de unión y una aceptación de unión.

**Mensaje de solicitud:** El procedimiento de unión siempre se inicia desde el dispositivo final enviando un mensaje de solicitud de unión.

**Tabla 3:** Formato de mensaje de solicitud.

<b>Size (bytes)</b>	8	8	2
<b>Join Request</b>	AppEUI	DevEUI	DevNonce

Fuente: (LoRa Alliance<sup>TM</sup>, 2018)

El mensaje de solicitud de adhesión contiene el AppEUI y el DevEUI del dispositivo final seguido de un Nonce de 2 octetos (DevNonce).

DevNonce es un valor aleatorio. Para cada dispositivo final, el servidor de red mantiene un registro de un cierto número de valores DevNonce usados por el dispositivo final en el pasado, e ignora las peticiones de unión con cualquiera de estos valores DevNonce de ese dispositivo final.

Este mecanismo evita los ataques de repetición mediante el envío de mensajes de solicitud conjunta previamente grabados con la intención de desconectar el dispositivo final respectivo de la red. Cada vez que el servidor de la red procesa una solicitud de unión y genere una trama de aceptación de unión, mantendrá tanto el antiguo contexto de seguridad (claves y contadores, si los hay) como el nuevo hasta que reciba la primera trama de enlace ascendente satisfactoria utilizando el nuevo contexto, después de lo cual el antiguo contexto podrá ser eliminado con seguridad. Esto proporciona una defensa



contra un adversario que repite una Joinrequest anterior usando un DevNonce que se sale de la lista finita de valores rastreados por el servidor de la red.

El mensaje de solicitud de unión puede ser transmitido usando cualquier tasa de datos y siguiendo una secuencia de salto de frecuencia aleatoria a través de los canales de unión especificados. Se recomienda utilizar una pluralidad de velocidades de datos.

**Mensaje de aceptación:** El servidor de la red responderá al mensaje de solicitud de unión con un mensaje de join-accept si se permite al dispositivo final unirse a una red. El mensaje join-accept se envía como un enlace descendente normal, pero utiliza retardos JOIN\_ACCEPT\_DELAY1 o JOIN\_ACCEPT\_DELAY2 (en lugar de RECEIVE\_DELAY1 y RECEIVE\_DELAY2, respectivamente). La frecuencia de canal y la velocidad de datos utilizada para estas dos ventanas de recepción son idénticas a la utilizada para las ventanas de recepción de RX1 y RX2.

No se da ninguna respuesta al dispositivo final si no se acepta la solicitud de adhesión. El mensaje join-accept contiene una solicitud nonce (AppNonce) de 3 octetos, un identificador de red (NetID), una dirección del dispositivo final (DevAddr), un retardo entre TX y RX (RxDelay) y una lista opcional de frecuencias de canal (CFList) para la red a la que se está uniendo el dispositivo final. La opción CFList es específica de la región y está definida en el documento de Parámetros Regionales de LoRaWAN [PARAMS].



**Tabla 4:** Formato de mensaje de aceptación.

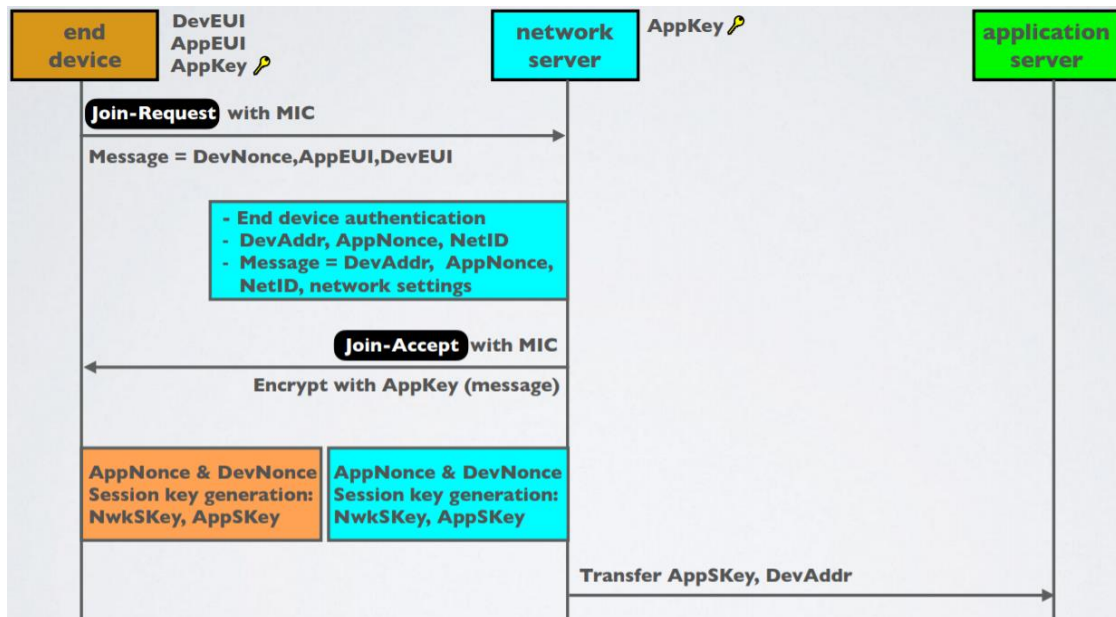
<b>Size (bytes)</b>	3	3	4	1	1	(16) Optional
<b>Join Accept</b>	AppNonce	NetID	DevAddr	DLSettings	RxDelay	CFList

Fuente: (LoRa Alliance<sup>TM</sup>, 2018)

El servidor de la red utiliza una operación de descryptación AES en el modo ECB para encriptar el mensaje de join-accept para que el dispositivo final pueda utilizar una operación de encriptación AES para descryptar el mensaje. De esta manera un dispositivo final sólo tiene que implementar encriptación AES pero no descryptación AES.

El establecimiento de estas dos claves de sesión permite una infraestructura de servidor de red federada en la que los operadores de red no pueden espiar los datos de las aplicaciones. En tal situación, el proveedor de aplicaciones debe apoyar al operador de la red en el proceso de que un dispositivo final se una a la red y establezca la NwkSKey para el dispositivo final. Al mismo tiempo, el proveedor de aplicaciones se compromete con el operador de red a asumir los gastos de cualquier tráfico en que incurra el dispositivo final y a mantener un control total sobre la AppSKey utilizada para proteger sus datos de aplicación.

El formato del NetID es el siguiente: Los siete LSB del NetID se llaman NwkID y coinciden con los siete MSB de la dirección corta de un dispositivo final como se ha descrito anteriormente. Las redes vecinas o superpuestas deben tener diferentes NwkIDs. Los 17 MSB restantes pueden ser elegidos libremente por el operador de la red (LoRa Alliance<sup>TM</sup>, 2018).



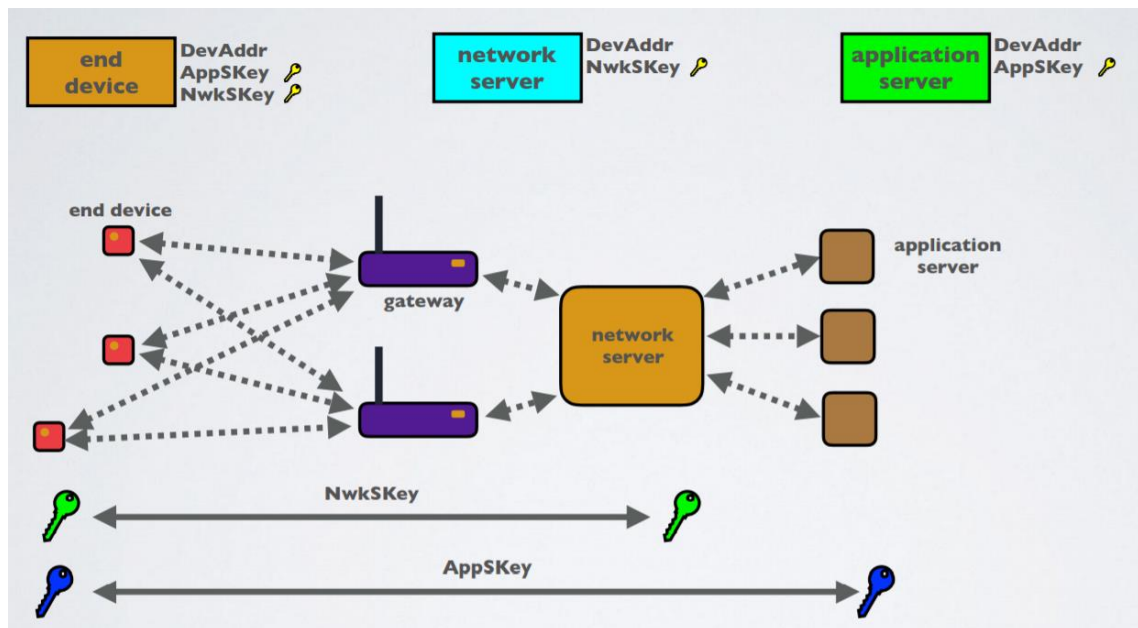
**Figura 13.** Diagrama de activación por OTAA.

Fuente: (Lie, 2018)

#### 2.3.7.6. Activación por personalización

En determinadas circunstancias, los dispositivos finales pueden activarse mediante la personalización. La activación por personalización vincula directamente un dispositivo final a una red específica pasando por el procedimiento de solicitud de adhesión y aceptación de adhesión. La activación de un dispositivo final mediante personalización significa que el DevAddr y las dos claves de sesión NwkSKey y AppSKey se almacenan directamente en el dispositivo final en lugar del DevEUI, AppEUI y AppKey. El dispositivo final está equipado con la información necesaria para participar en una red LoRa específica cuando se inicia (LoRa Alliance<sup>TM</sup>, 2018).

Cada dispositivo debe tener un conjunto único de NwkSKey y AppSKey. Comprometer las claves de un dispositivo no debería comprometer la seguridad de las comunicaciones de otros dispositivos. El proceso para construir esas claves debe ser tal que las claves no puedan derivarse de ninguna manera de la información disponible públicamente (como la dirección del nodo, por ejemplo).

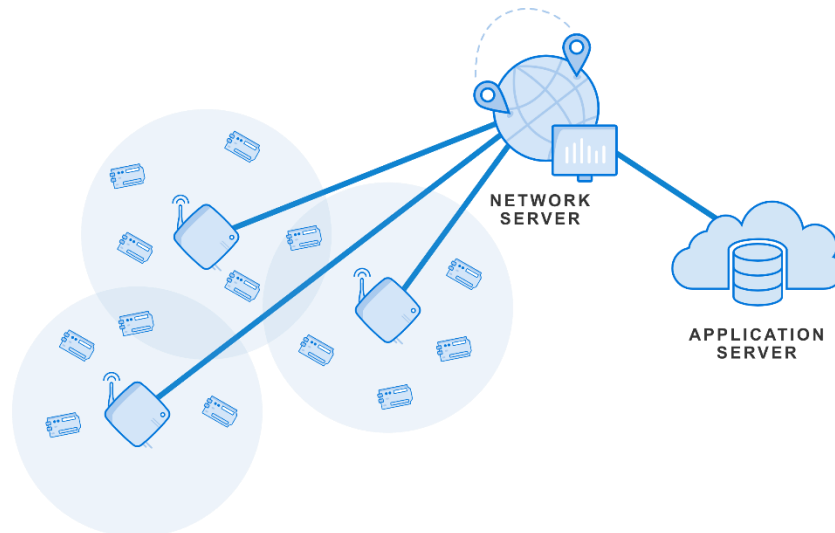


**Figura 14.** Diagrama de activación por ABP.

Fuente: (Lie, 2018)

## 2.4. THE THINGS NETWORK

Los sistemas de backend de The Things Network son responsables de enrutar los datos de Internet de las Cosas entre los dispositivos y las aplicaciones. Una red típica de Internet de las Cosas requiere pasarelas como puente entre protocolos de radio específicos e Internet. En los casos en que los propios dispositivos soportan la pila de IP, estas pasarelas sólo tienen que reenviar paquetes a Internet. Los protocolos no IP, como LoRaWAN, requieren algún tipo de enrutamiento y procesamiento antes de que los mensajes puedan ser entregados a una aplicación. La red se sitúa entre las pasarelas y las aplicaciones (véase la figura a continuación) y se encarga de estos pasos de enrutamiento y procesamiento (Barro et al., 2019).



**Figura 15.** Arquitectura de The Things Network

Fuente: (The Things Network, 2020)

Los nodos transmiten mensajes LoRaWAN a través del protocolo de radio LoRa. Estos mensajes son recibidos por varios Gateway. El Gateway es una pieza de hardware que envía las transmisiones de radio al backend y está conectado a un router. El router es responsable de gestionar el estado de la puerta de enlace y de programar las transmisiones. Cada Router está conectado a uno o más Brokers. Los Brokers son la parte central de la Red de Cosas. Su responsabilidad es asignar un dispositivo a una aplicación, reenviar los mensajes de enlace ascendente a la aplicación correcta y reenviar los mensajes de enlace descendente al router correcto (que los reenvía a un gateway). El Network Server, es responsable de la funcionalidad que es específica para LoRaWAN. Un Handler es responsable de manejar los datos de una o más Aplicaciones. Para ello, se conecta a un Broker donde registra las aplicaciones y los dispositivos. El Handler es también el punto donde los datos son encriptados o desencriptados (The Things Network, 2020).

El objetivo de The Things Network es ser muy flexible en cuanto a las opciones de despliegue. La opción preferida es conectarse a la red comunitaria pública alojada por



The Things Network Foundation o sus asociados. En este caso la Aplicación se conecta a un Handler de Red Comunitaria Pública, usualmente usando el API MQTT.

También es posible desplegar redes privadas, ejecutando todos estos componentes en un entorno privado. De esta manera, todos los datos permanecerán dentro del entorno privado, pero aún se puede utilizar el Servidor de cuentas alojado de TTN para la autenticación y la autorización.

Los despliegues híbridos serán posibles en el futuro. La opción más simple para esto, es que alguien corra su propio Handler, permitiéndole manejar la encriptación y desencriptación de los mensajes. Una opción más complicada es una red privada que intercambie datos con la red pública. Para que esto funcione, los Routers privados tendrán que conectarse a los Brokers públicos y viceversa. En este caso, la red privada puede descargar el tráfico público a la red comunitaria y utilizar la red comunitaria pública como respaldo. Esto último todavía no es posible con la actual implementación del backend (The Things Network, 2020).

#### **2.4.1. Funcionalidad del núcleo**

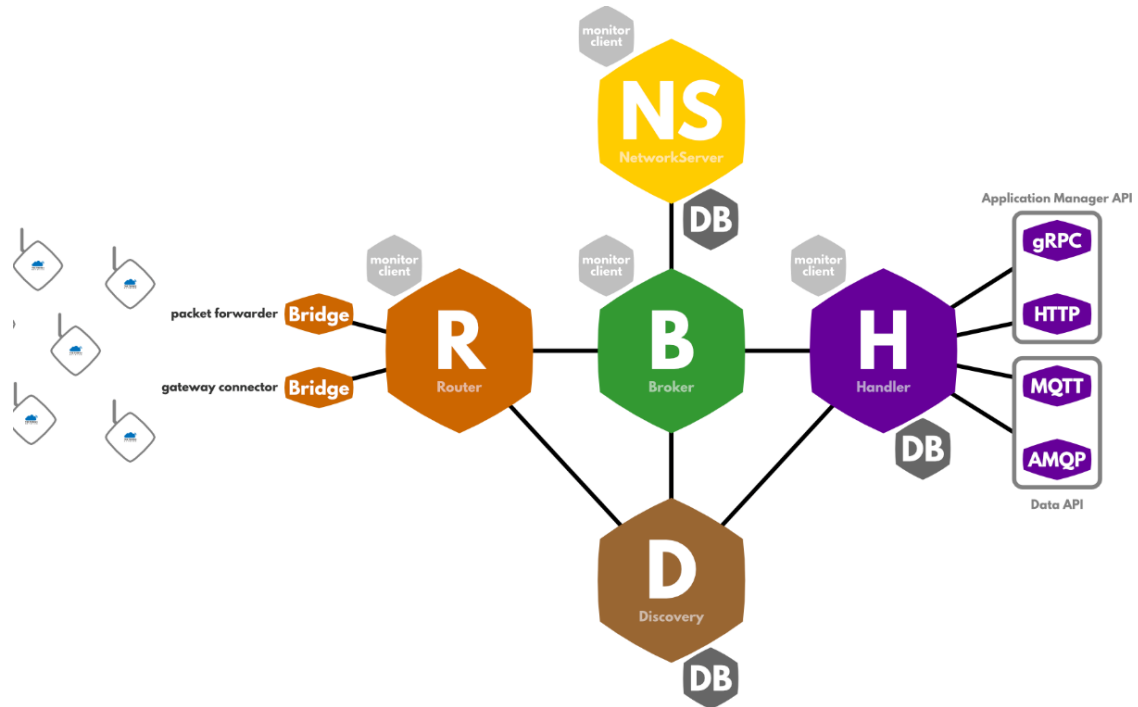
La Red de Cosas provee un Servidor de Red LoRaWAN. LoRaWAN fue diseñado para la arquitectura centralizada de los operadores de telecomunicaciones, por lo que para poder funcionar en una infraestructura distribuida como The Things Network, se tuvieron que añadir algunos pasos. En el backend de The Things Network ahora distinguimos una serie de funciones básicas diferentes. En primer lugar, hay funciones relacionadas con los Gateway, como la programación y la gestión de la utilización de los Gateway. La programación es necesaria porque un portal sólo puede hacer una transmisión al mismo tiempo. La información se usa para distribuir uniformemente la carga en las diferentes pasarelas y para cumplir con los ciclos de trabajo europeos. Otra característica importante



es la vigilancia del estado de cada puerta de enlace. En segundo lugar, necesitamos funciones relacionadas con los dispositivos que gestionen el estado de los dispositivos en la red. Como las direcciones de los dispositivos no son únicas, la red tiene que hacer un seguimiento de las direcciones que utilizan cada uno de los dispositivos para poder asignar un mensaje al dispositivo y aplicación correctos. Otras cosas de las que la red debe hacer un seguimiento son las claves de seguridad y los contadores de tramas. En el futuro también comenzaremos a llevar un registro de la utilización de la red de cada nodo (The Things Network, 2020).

En tercer lugar, hay algunas funcionalidades relacionadas con las aplicaciones. Por ejemplo, los Brokers y Handlers necesitan saber a qué servidor de tráfico debe ser enviado, para una aplicación específica. Los Handlers necesitan saber cómo interpretar los datos binarios, y hacer un puente con los protocolos de capa superior, como AMQP y MQTT (The Things Network, 2020).

Finalmente, y lo más importante, como la Red de Cosas es una red distribuida, tiene que haber una funcionalidad que soporte esta distribución. La funcionalidad de descubrimiento de servicios ayuda a los componentes a determinar hacia dónde debe dirigirse el tráfico. Actualmente, esto se implementa como un servidor de Discovery centralizado, dando a The Things Network Foundation el control sobre qué componentes están autorizados a anunciar servicios específicos. La arquitectura de funcionalidades se muestra en la figura 16.

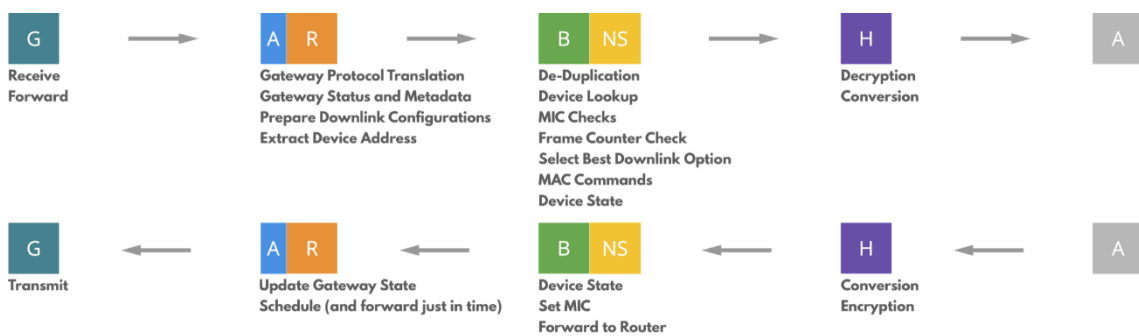


**Figura 16.** Arquitectura de funcionalidades de The Things Network.

Fuente: (The Things Network, 2020)

#### 2.4.2. Procesamiento del flujo de mensajes.

Basándonos en la separación de preocupaciones se implementa el backend de The Things Network. Como cada componente tiene un número de responsabilidades de alto nivel, tiene que ejecutar una serie de tareas al procesar los mensajes de enlace ascendente y descendente. En la figura a continuación se muestra una visión general de este flujo, que se analiza en detalle en el resto de esta sección (The Things Network, 2020).



**Figura 17.** Visión general del flujo de The Things Network.

Fuente: (The Things Network, 2020)

#### **2.4.2.1. Traslado del protocolo de Gateway (Router/Bridge)**

Cuando una pasarela recibe un mensaje que fue transmitido por la LoRa, se encapsula y se reenvía a la Red de Cosas por Internet. Muchas pasarelas utilizan el mismo protocolo de referencia, pero se han desarrollado protocolos alternativos para determinar backends. The Things Network también está desarrollando su propio protocolo de pasarela, que es más adecuado para The Things Network que el protocolo de referencia.

La mayoría de los protocolos de pasarela tienen la misma estructura. Cuando se reciben uno o más mensajes, su carga útil binaria se remite al backend, junto con metadatos como la intensidad de la señal (RSSI) y la relación señal-ruido (SNR). Periódicamente, la pasarela también envía alguna información de estado sobre la propia pasarela, como las coordenadas GPS, el número de paquetes recibidos y transmitidos y otras métricas.

Esperamos conectar los Gateway de diferentes proveedores, ejecutando diferentes protocolos. Con el fin de mantener el backend de The Things Network lo más genérico posible, implementamos una serie de puentes que se traducen de cada protocolo de Gateway específico de cada vendedor, al protocolo utilizado internamente en el backend de The Things Network (Barro et al., 2019).

#### **2.4.2.2. Estado del Gateway y Metadata (router)**

Las coordenadas GPS de una pasarela pueden ser especialmente relevantes para la aplicación. Por lo tanto, el backend almacena el último mensaje de estado enviado por la pasarela e inyecta la información GPS en los metadatos de cada mensaje de enlace ascendente (The Things Network, 2020).

#### **2.4.2.3. Configuración del enlace descendente (Router)**

En LoRaWAN la respuesta del enlace descendente a un mensaje de enlace ascendente depende en gran medida de la región geográfica de la puerta de enlace, y se describen en





la especificación de los "Parámetros regionales de la LoRaWAN". Dado que el Router es responsable de todos los detalles relacionados con la pasarela y específicos de la región, el Router tiene que determinar cómo se puede enviar una respuesta de enlace descendente a un dispositivo. Después de cada mensaje de enlace ascendente, hay dos ventanas de recepción, una a exactamente 1 segundo después del enlace ascendente, la otra a los 2 segundos. Por lo tanto, para cada puerta de enlace que recibió el mensaje de enlace ascendente, un Router construye dos configuraciones de enlace descendente.

Para poder seleccionar la mejor opción, el Router tiene que calcular adicionalmente una puntuación para cada opción. Esta puntuación está influenciada por una serie de factores. Por el momento consideramos el tiempo de emisión, la intensidad de la señal, la utilización de la puerta de enlace y las transmisiones ya programadas. Esto último es bastante obvio, ya que una pasarela no puede hacer dos transmisiones al mismo tiempo. La programación de un mensaje de enlace descendente en una pasarela que tuviera una mejor intensidad de señal (relación señal-ruido) también hace más probable que un nodo reciba el enlace descendente correctamente.

La combinación del tiempo de emisión de un mensaje y la utilización de una pasarela se utiliza para optimizar la red en su conjunto. Como cada transmisión bloquea los receptores de una pasarela durante algún tiempo, es mejor enviar los mensajes en un tiempo más corto. Por lo tanto, se prefieren los mensajes de enlace descendente a una velocidad de datos mayor que los mensajes con una velocidad de datos menor. La utilización de una pasarela indica el porcentaje de tiempo que una pasarela está recibiendo mensajes. Por consiguiente, las pasarelas con una mayor utilización (por estar situadas en un buen lugar) son más valiosas para la red que las pasarelas con una utilización menor. Por lo tanto, se debe dar preferencia a las segundas para los mensajes de enlace



descendente, de modo que las primeras puedan seguir recibiendo los mensajes de enlace ascendente (The Things Network, 2020).

#### **2.4.2.4. Extracción de la dirección del dispositivo – DevAddr (Router)**

El primer paso en el enrutamiento de un paquete se basa en la dirección del dispositivo. Se trata de una dirección no única de 32 bits, de la cual el operador de red puede asignar 25 bits. La Red de Cosas ha elegido distribuir el tráfico basado en el prefijo de la dirección del dispositivo. Cada Broker anuncia un número de prefijos de direcciones de dispositivos con un servicio de descubrimiento. Estos prefijos son similares a la forma en que los rangos de direcciones IP son anunciados en BGP. Los componentes de la red pueden recuperar periódicamente la lista de Brokers y sus prefijos anunciados. Los mensajes se reenvían a todos los Brokers que anuncian un prefijo que coincide con la dirección de dispositivo del mensaje (The Things Network, 2020).

#### **2.4.2.5. Desduplicación (Broker)**

LoRaWAN es un protocolo de radio de largo alcance, lo que hace probable que el mensaje sea recibido por más de un Gateway. Esto significa que el backend tiene que realizar algún tipo de des-duplicación para entregar un mensaje sólo una vez a la aplicación. Esto no significa que los duplicados no sean importantes. Los metadatos de estos mensajes también pueden ser valiosos. Por ejemplo, al combinar las ubicaciones de las puertas de enlace que recibieron el mensaje con la hora de recepción y la intensidad de la señal, podría ser posible determinar la ubicación del dispositivo que envió el mensaje.

La actual implementación del backend a nivel local des-duplica los mensajes de enlace ascendente basados en la suma de los md5 de la carga útil. La carga útil del mensaje será la misma para todos los duplicados, y la posibilidad de que durante el período de des-



duplicación (normalmente un par de segundos) llegue un mensaje diferente con el mismo hash es extremadamente baja (The Things Network, 2020).

Las puertas de enlace pueden conectarse a cualquier red de acceso. Algunas están conectadas a ethernet con cable, otras utilizan conexiones WiFi o incluso GPRS a Internet. Así que aunque las ondas de radio viajan a la velocidad de la luz, el retardo de la red hace que los mensajes duplicados no lleguen al Broker al mismo tiempo. Para recoger los metadatos añadidos al mensaje por cada puerta de enlace, el Broker tiene que almacenar los duplicados en un búfer durante algún tiempo. Ese tiempo debe ser lo suficientemente largo para reunir el mayor número posible de duplicados, pero lo suficientemente corto para dar a la aplicación tiempo suficiente para responder a un mensaje en la ventana de recepción que se abrirá 1 segundo después de la transmisión.

Nuestras mediciones en el despliegue actual de La Red de las Cosas han mostrado que el retraso medio entre el primer y el último duplicado es de poco menos de 100 ms, con el retraso máximo de unos 300 ms. Nos damos cuenta de que estas mediciones se hacen en una red donde las puertas de enlace no están aun densamente desplegadas, por lo que necesitamos más datos para poder sacar conclusiones. Sin embargo, estos valores dan una indicación decente del tiempo de des-duplicación requerido, que actualmente está fijado en 200 ms. Cuando se reúnan más datos, este tiempo puede ser optimizado aún más (The Things Network, 2020).

#### **2.4.2.6. Búsqueda de dispositivos y aplicaciones (Broker/Network Server)**

Debido a que las direcciones de los dispositivos no son únicas, es necesario determinar el dispositivo exacto que envió el mensaje, y la aplicación a la que pertenece. Para ello, el backend tiene que realizar una serie de comprobaciones del código de integridad de mensajes criptográficos (MIC), una para cada dispositivo que utiliza la misma dirección



de dispositivo. Para ello, el Broker solicita una lista de dispositivos con la dirección de dispositivo dada al Servidor de Red y comprueba si el MIC puede ser validado utilizando la clave de sesión de red. Si no se encuentra ninguna coincidencia, el mensaje se elimina (The Things Network, 2020).

#### **2.4.2.7. Comprobación del contador de tramas (Broker)**

El contador de tramas en los mensajes LoRaWAN es una medida de seguridad que se utiliza para detectar los ataques de repetición. Después de validar el MIC, el Broker comprueba si el contador de tramas es válido. Además, el Broker tiene que verificar que la brecha entre el último contador de tramas conocido y el contador del mensaje no sea demasiado grande. De acuerdo con la especificación de LoRaWAN, la brecha máxima es de 16384. LoRaWAN soporta tanto contadores de tramas de 16 como de 32 bits. Sin embargo, sólo los 16 bits menos significativos del contador se incluyen en el encabezamiento del mensaje. Por lo tanto, el backend tiene que hacer un seguimiento del contador de tramas de 32 bits completo y utilizarlo en lugar del contador de 16 bits que se incluye en el mensaje (The Things Network, 2020).

#### **2.4.2.8. Colección de metadatos (Broker)**

Cuando todas las comprobaciones hayan tenido éxito, el Broker puede continuar procesando el mensaje. Primero, fusiona los duplicados recibidos de todos los diferentes Routers y Gateway. En este paso es importante diferenciar entre los metadatos que son iguales para cada puerta de enlace que recibió el mensaje y los metadatos que son específicos de cada recepción. Por ejemplo, la frecuencia, la modulación y la velocidad de los datos serán las mismas para todas las pasarelas, por lo que sólo es necesario reenviarlos una vez. Por otra parte, la intensidad de la señal, la hora de recepción y las coordenadas GPS de cada puerta de enlace deben incluirse al reenviar el mensaje. En este



paso también se recogen las diferentes configuraciones de los enlaces descendentes para seleccionar la mejor opción en el siguiente paso (The Things Network, 2020).

#### **2.4.2.9. Selección de la mejor opción de enlace descendente (Broker)**

El Broker tiene que seleccionar la mejor opción para una respuesta de enlace descendente a un mensaje. Como el Broker no tiene ninguna información sobre la puerta de enlace que recibió un mensaje, es muy difícil hacer esto. Por lo tanto, el Router ya ha calculado una puntuación para cada configuración de enlace descendente. Si este cálculo de puntuación se hace de forma estándar, el Broker ahora sólo tiene que ordenar todas las opciones posibles de enlace descendente y utilizar la mejor opción (The Things Network, 2020).

#### **2.4.2.10. Estado del dispositivo y comandos MAC (Network Server)**

Antes de reenviar el mensaje de subida al Handler, primero se envía al Network Server para que el estado del dispositivo pueda ser actualizado. El Network Server también agrega una plantilla de enlace descendente al mensaje. Esta plantilla puede ser utilizada por el Handler para enviar un mensaje de enlace descendente al dispositivo. Contiene todos los valores necesarios (como el contador de tramas, el tipo de mensaje y los indicadores de opción) para que el Handler sólo tenga que agregar la carga útil de la aplicación al mensaje. Además, esto le da al Network Server la oportunidad de agregar comandos MAC al mensaje. Por ejemplo, basándose en el número de puertas de enlace que recibieron un mensaje y en la intensidad de su señal, el Network Server puede agregar comandos MAC que le indiquen al dispositivo que transmita a una mayor velocidad de datos (The Things Network, 2020).

#### **2.4.2.11. Descifrado de mensajes (Handler)**

Como los mensajes están cifrados de extremo a extremo, el backend también es responsable de descifrar los mensajes. Sin embargo, no en todos los casos el dueño de la aplicación puede querer que The Things Network sea responsable de eso. Por lo tanto, el descifrado de los mensajes se coloca en un componente separado (el Handler), lo que permite al propietario de la aplicación ejecutar este Handler en su propio entorno privado (The Things Network, 2020).

#### **2.4.2.12. Conversión de la carga útil (Handler)**

Después de la descriptación, el Handler puede decodificar y convertir la carga útil en un formato fácilmente accesible por la aplicación. Por consiguiente, la aplicación del Handler predeterminado incluye las llamadas funciones de carga útil. Estas funciones son simples funciones de JavaScript que pueden utilizarse para decodificar, convertir y validar datos (The Things Network, 2020).

#### **2.4.2.13. Downlink (Handler)**

Después de publicar el mensaje de enlace ascendente a MQTT, el Handler determinará si es necesario responder al dispositivo con un mensaje de enlace descendente. Hay tres situaciones en las que es necesario enviar un mensaje de enlace descendente. La primera y más obvia es cuando la aplicación tiene una carga útil disponible para enviar al dispositivo. En este caso, la carga útil se añade a la plantilla de respuesta generada por el servidor de red. El segundo caso es cuando el mensaje de enlace ascendente requiere confirmación. Independientemente de que la carga útil del enlace descendente esté disponible o no, se debe enviar un acuse de recibo. El tercer caso es cuando el Network Server necesita enviar comandos MAC al dispositivo. Si esto se detecta, el Handler podría



decidir si obedece o no al Network Server, aunque la implementación actual siempre sigue la petición del Network Server.

De manera similar a los mensajes de enlace ascendente, el Handler es responsable de encriptar la carga útil del mensaje. Si no se dispone de una carga útil de enlace descendente, el Handler puede optar por esperar un breve período de tiempo para que la aplicación prepare un mensaje de enlace descendente basado en el mensaje de enlace ascendente que acaba de recibir. Una vez vencido este plazo, el Handler debe enviar el mensaje de enlace descendente al Broker (The Things Network, 2020).

#### **2.4.2.14. Estado del dispositivo (Network Server)**

Después de que el Broker recibe un mensaje de enlace descendente de un Handler, envía el mensaje al Network Server, que actualizará el estado del dispositivo (específicamente, los contadores de tramas) en la base de datos y generará el MIC del mensaje. Después de esto, el Broker reenvía el mensaje de enlace descendente al Router que es responsable de la pasarela que tiene que transmitir el mensaje de enlace descendente (The Things Network, 2020).

#### **2.4.2.15. Programación del enlace descendente (Router)**

Como se mencionó al principio de este capítulo, el Router es responsable de gestionar el horario del portal. Como la mayoría de las pasarelas sólo tienen un búfer de 1 mensaje de enlace descendente, el Router tiene que almacenar en un búfer los mensajes programados hasta el último momento, y luego enviar cada mensaje justo a tiempo a la pasarela (The Things Network, 2020).



## CAPÍTULO III

### MATERIALES Y MÉTODOS

#### 3.1. MATERIALES

##### 3.1.1. Hardware

###### Ordenador portátil

- Modelo: LENOVO IdeaPad S340
- Procesador: Intel(R) Core(TM) i5-7500 2.70GHz
- RAM: 8GB
- Tipo de sistema: Sistema Operativo de 64 bits Windows 10

###### Ordenador Raspberry Pi

- Modelo: Raspberry Pi 3 Model B+
- Procesador: Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC

@ 1.4GHz

- RAM: 1GB LPDDR2 SDRAM
- Adaptador de red inalámbrica: 5GHz IEEE 802.11.b/g/n/ac wireless LAN,

Bluetooth 4.2, BLE

- Adaptador de red: Gigabit Ethernet
- Puertos USB: 4 USB 2.0
- Entrada de energía: 5V/2.5A DC





**Figura 18.** Raspberry Pi.

Fuente: (Raspberry Pi Foundation, 2016)

### **Modulo LoRa**

- Voltaje de funcionamiento: 5V DC
- La temperatura de funcionamiento: -40 °C a +85 °C
- Chipset inalámbrico: SX1301 half-duplex
- Canales: 10 canales, 8 de bajada, 1 de subida y 1 para FSK
- Potencia de salida: encima de 25.2dBm
- Sensibilidad: debajo de -139dBm
- Procesamiento de SNR: 9dB
- Interfaz: SPI
- Dimensión: 80mmx50mmx5mm
- Rango: encima de 15km (con línea de vista)
- Banda de frecuencia: 915MHz
- Ganancia de antena: 6dBi
- Tipo de antena: lineal-omnidireccional

- Impedancia de antena: 50 Ohm

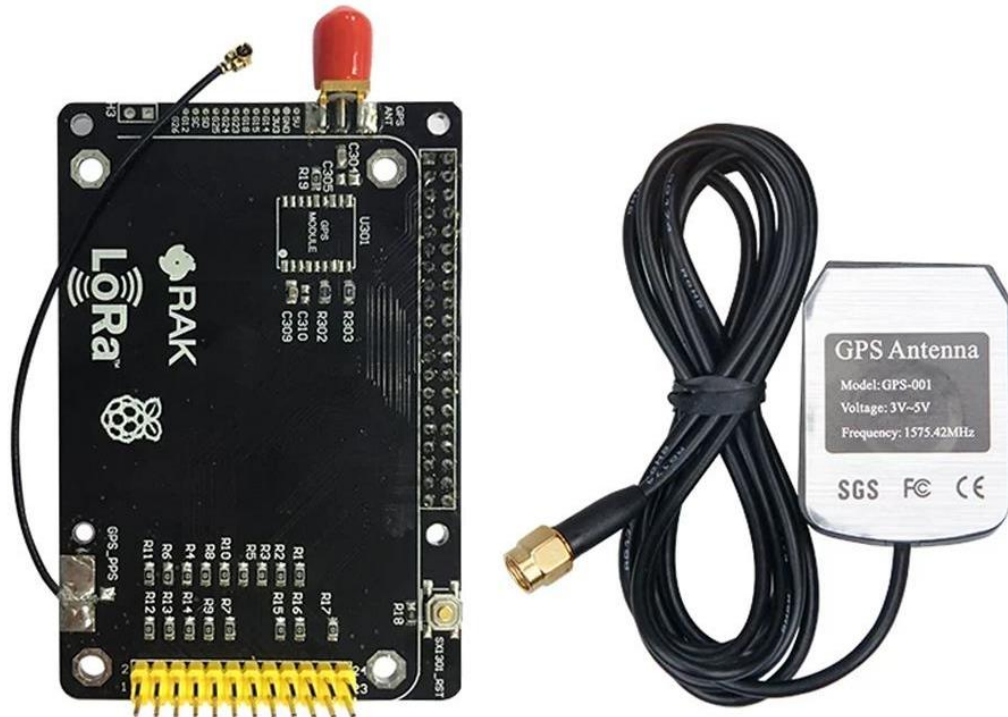


**Figura 19.** Modulo LoRa para Gateway.

Fuente: (SHENZHEN RAKWIRELESS TECHNOLOGY, 2017b)

### **Placa convertora de Raspberry Pi con Modulo LoRa**

- Voltaje de funcionamiento: 3.6V DC
- La temperatura de funcionamiento: -40 °C a +85 °C
- Chipset GPS: u-blox MAX-7Q
- Canales: 56 canales
- Sensibilidad: 161dBm
- Límite de operación de velocidad: 500m/s
- Límite de operación de altitud: 5000m
- Interfaz: SPI
- Dimensión: 80mmx50mmx5mm
- Ganancia de antena: 27dB



**Figura 20.** Convertidor para la conexión del Raspberry Pi y el módulo LoRa

Fuente: (SHENZHEN RAKWIRELESS TECHNOLOGY, 2017b)

### Dispositivo final (End Node)

- Chipset inalámbrico: Semtech SX1276 LoRaWAN™ Version V1.0.2
- GPS: U-Blox GPS
- Soporte de activación: ABP y OTAA
- Temperatura de operacion: -20°C ~ 60°C
- Dimensiones: 74mm x 43mm x17mm
- Potencia de salida: 20dBm
- VSWR de la antena LoRa: 1.5:1
- Ganancia de antena LoRa: 2.0 dBm
- VSWR de la antena GPS:  $\leq 2.0 \pm 0.5$
- Ganancia de antena LoRa: 6dBi

- Polarización: Circular



**Figura 21.** Dispositivo final – End Node.

Fuente: (SHENZHEN RAKWIRELESS TECHNOLOGY, 2017a)

### 3.1.2. Software

#### ArcMap

Versión: 10.5

Dirección web: <https://desktop.arcgis.com/es/desktop/>

ArcMap es el lugar donde visualiza y explora los dataset SIG de su área de estudio, donde asigna símbolos y donde crea los diseños de mapa para imprimir o publicar. Es también la aplicación que utiliza para crear y editar los dataset. ArcMap representa la información geográfica como una colección de capas y otros elementos en un mapa. Los elementos de mapa comunes son el marco de datos, que contiene las capas de mapa para una extensión determinada, más la barra de escala, la flecha de norte, el título, texto descriptivo, una leyenda de símbolos, etc. Las tareas habituales que se puede realizar en ArcMap son las siguientes:



- Trabajar con mapas.
- Imprimir mapas.
- Compilar y editar datasets SIG.
- Utilizar geoprocésamiento para automatizar el trabajo y realizar análisis
- Organizar y administrar geodatabases y documentos de ArcGIS
- Documentar la información geográfica
- Personalizar la experiencia incluyendo herramientas de personalización, incluyendo la capacidad para escribir add-ins de software para agregar nueva funcionalidad.

### **Google Earth Pro**

Versión: 7.3.3

Dirección web: <https://www.google.com/intl/es/earth/download/gep/agree.html>

El mapa de Google Earth está compuesto por una superposición de imágenes obtenidas por imágenes satelitales, fotografías aéreas, información geográfica proveniente de modelos de datos SIG de todo el mundo y modelos creados por computadora. En el cual se puede crear polígonos, líneas y puntos geográficos, a la vez se puede hacer cálculos de distancia y áreas, en la misma muestra perfiles del suelo. Es una herramienta útil al visualizar los archivos creados en ArcMap.

### **PuTTY**

Lanzamiento: 0.72

Dirección web: <https://www.putty.org/>

PuTTY es un cliente SSH y telnet, desarrollado originalmente por Simon Tatham para la plataforma Windows. PuTTY es un software de código abierto que está disponible con código fuente y es desarrollado y apoyado por un grupo de voluntarios.



## **Radio Mobile**

Versión: 11.6.6

Página web: <https://www.ve2dbe.com/english1.html>

Radio Mobile está dedicado a la radioaficionado y el uso humanitario. Aunque el uso comercial no está prohibido, el autor no se hace responsable de su uso. Usado para el cálculo de radio enlaces de larga distancia en terreno irregular y uso de perfiles geográficos y los datos de los equipos que quieren simularse. Permite la introducción de parámetros como; Potencia, Sensibilidad, Características de las antenas, Pérdidas, Etc. Permite cargar el perfil topográfico de la zona de trabajo, esto es, datos de la elevación del terreno. Los tipos de mapas disponibles son: SRTM, GTOPO30, DTED, GLOBE, BIL y existen en internet datos de elevación de terreno casi que de cualquier zona del mundo

## **Advanced IP Scanner**

Versión: 2.5

Dirección web: <https://www.advanced-ip-scanner.com/es/>

Escáner de red fiable y gratuita para analizar LAN. El programa escanea todos los dispositivos de red, le da acceso a las carpetas compartidas y a los servidores FTP, le proporciona control remoto de las computadoras (mediante RDP y Radmin) e incluso puede apagar las computadoras de manera remota. Es fácil de usar y se ejecuta como una edición portable. Debe ser la primera opción para cada administrador de red.

## **balenaEtcher**

Versión: 1.5

Dirección web: <https://www.balena.io/etcher/>



balenaEtcher es una aplicación gráfica que te permite grabar imágenes de tu sistema operativo favorito en tarjetas SD y memorias USB, de forma segura, y sobretodo, fácilmente.

### **RAK SERIAL\_PORT TOOL**

Versión: 1.2.1

Dirección web: <https://downloads.rakwireless.com/LoRa/Tools/>

Es una herramienta de RAK Wireless, proporcionada para ingresar líneas de comando y visualizar el comportamiento del End Node.

### **The Things Network**

Dirección web: <https://www.thethingsnetwork.org>

Se detalló en la sección 2.4

## **3.2. MÉTODO**

### **3.2.1. Diseño de investigación.**

El Diseño de investigación es cuasi-experimental, ya que es un estudio de campo con experimento llevado a cabo fuera del laboratorio. A diferencia de los estudios de caso y los estudios observacionales, un experimento de campo repite todos los pasos del proceso científico, abordando los problemas de investigación y generando hipótesis. La ventaja evidente de un estudio de campo es que es práctico y permite la experimentación. Todos estos aspectos están relacionados con esta investigación

### **3.2.2. Nivel de la investigación**

El nivel de investigación se refiere a la profundidad del conocimiento que se busca lograr con la investigación, por tanto, el nivel de la presente investigación en cierto modo es exploratoria, donde señalan que las investigaciones exploratorias buscan abrir nuevos caminos en el desarrollo del conocimiento humano. La presente investigación siendo un





prototipo busca abrir un camino para un nuevo método de monitoreo de animales en peligro de extinción mediante una comunicación LoraWAN dentro de la comunidad Primer Chimpa Jilahuata del distrito de Azángaro.

### **3.2.3. Población de la investigación**

La población de la investigación es la cantidad vicuñas que habitan dentro de la comunidad de Primer Chimpa Jilahuata del distrito de Azángaro, y alrededor de la laguna Quequerana. Debido a la escasez de información sobre la cantidad exacta de vicuñas que habitan en el área descrita no se cuenta con una cifra definida.

### **3.2.4. Muestra de la investigación**

La presente investigación realizará un muestreo no probabilísticas, las muestras que se eligen o los elementos no se hacen en base a la probabilidad, sino más bien se realizan en base a las características de la propia investigación o lo que estime conveniente el investigador. En tal sentido se aplicará en un caso específico de localización, para determinar cobertura y duración de energía.

### **3.2.5. Ubicación de la investigación.**

La investigación se desarrollará en la comunidad Primer Chimpa Jilahuata del distrito de Azángaro, provincia de Azángaro – Puno, a 8 kilómetro del distrito de Azángaro y alrededores de la laguna Quequerana.

### **3.2.6. Recolección de datos.**

#### **3.1.1.1. Técnicas**

Las técnicas de recolección de datos se basarán con la observación experimental hacia el objeto en estudio, analizando y documentando los resultados obtenidos.



### 3.1.1.2. Instrumentos

Los instrumentos para la recolección son las hojas o fichas de registro de datos, ya sean digitales o físicas, también se usará estrategias que el investigador adopte.

### 3.2.7. Técnicas de procesamiento y análisis

#### 3.1.1.1. Plan de recolección de datos

Para la recolección de datos, se describirá etapas de la investigación, detallándose las configuraciones, materiales (hardware y software) usados y datos obtenidos. Estas etapas se describen a continuación:

#### Etapa 1: Configuración del concentrador (Gateway)

Esta investigación, de acuerdo a la revisión literaria realizada, se dispuso a utilizar la arquitectura de la tecnología LoRaWAN, la cual está comprendida con los End Node (puntos finales), el Gateway (Concentrador) y el servidor, que es para unión (Join Server) y de red (Network Server); conforme a los que se muestra en la figura 22.

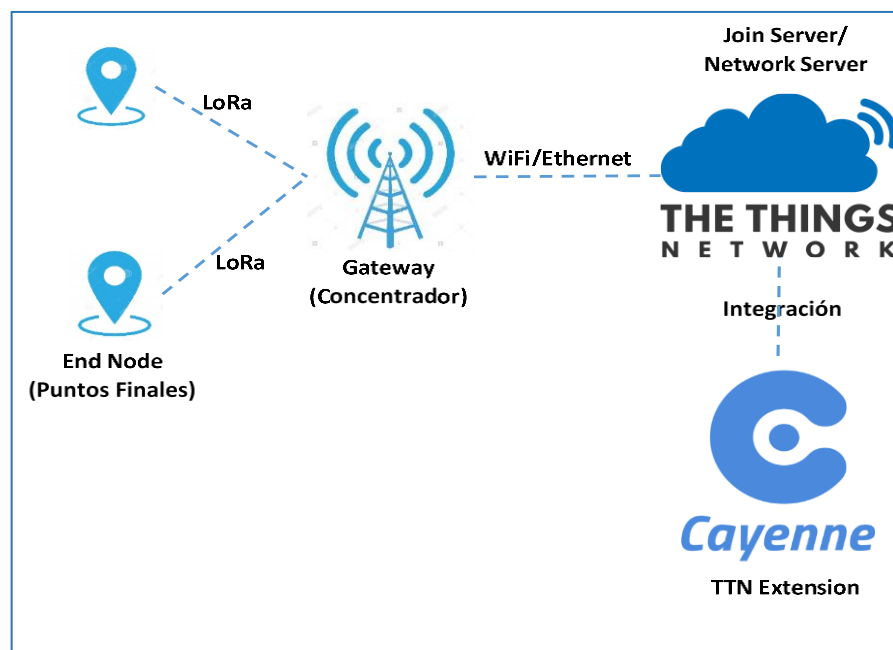


Figura 22. Arquitectura empleada en la investigación.

Elaboración propia

En esta primera etapa, se detallará las configuraciones realizadas y los materiales utilizados para el funcionamiento del Gateway. Los materiales usados son, el Raspberry Pi, el módulo RAK831, placa de conversión, las especificaciones técnicas fueron resumidas en el punto 3.1.1.

Para establecer el funcionamiento del Gateway, el módulo RAK831 debe comunicarse con el procesador host (placa Raspberry Pi3) a través de SPI. Como ambas placas tienen conectores de cabecera, las conexiones se pueden hacer con cables. Sin embargo, es mejor utilizar la placa adaptadora, que se ajusta a ambas placas; como se observa en la figura 23.



**Figura 23.** Montaje del Gateway

Elaboración propia

Para la instalación del Software en el Raspberry Pi, es necesario utilizar una tarjeta de memoria, en esta tarjeta se almacenará el sistema operativo y sus archivos. Es recomendable una memoria de 8 GB de clase 4.

El sistema operativo utilizado es el Raspbian Pi OS (32 bits) Lite, el cual está disponible en la siguiente dirección web;

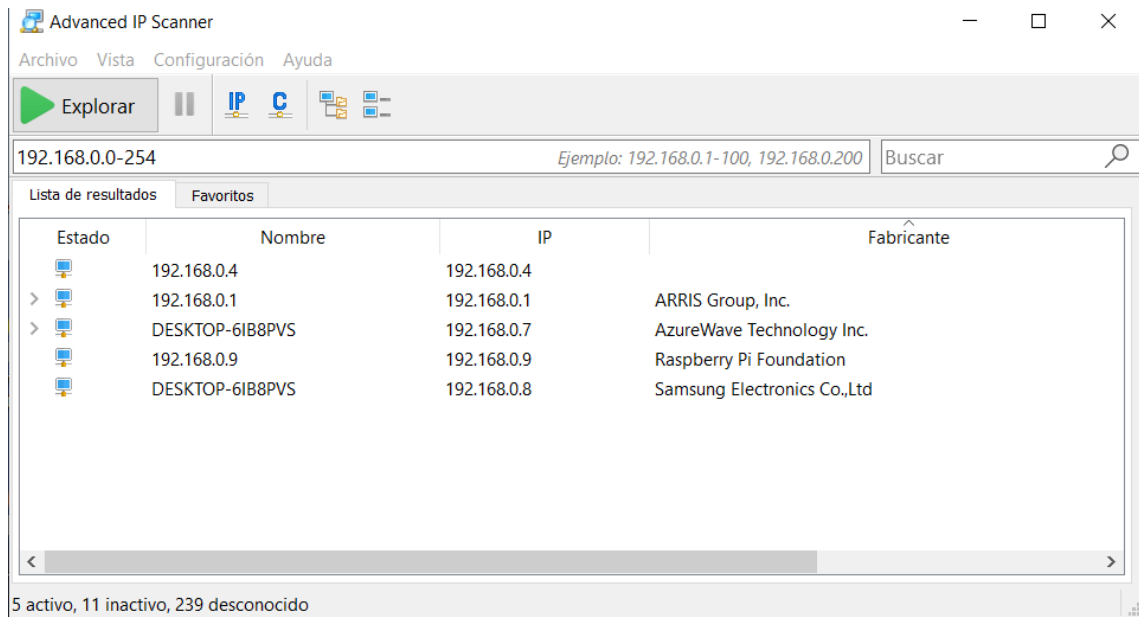


<https://www.raspberrypi.org/downloads/raspberry-pi-os/>. Si bien hay la posibilidad de utilizar las demás variedades del sistema operativo de Raspberry Pi, que brindan el entorno de escritorio y programas recomendados; para la investigación estas funciones no son necesarias, ya que la configuración puede realizarse de manera remota por SSH.

Para realizar la transferencia de imagen del sistema operativo hacia la tarjeta de memoria, se utilizó el programa balenaEtcher, el cual está disponible para descargar en la dirección web; <https://www.balena.io/etcher/>, para su instalación en Windows. Es necesario insertar la tarjeta de memoria al ordenador con el programa instalado (balenaEtcher), posteriormente iniciar el programa, en donde se debe seleccionar la imagen de sistema operativo descargado (extensión .img) y la tarjeta de memoria.

Para el encendido, se inserta la tarjeta de memoria en la ranura correspondiente del Raspberry Pi. Además, la conexión mediante el puerto Ethernet a una red local que otorga direcciones IP por DHCP. Por último, se conecta a una fuente de alimentación con al menos de 5V y 2.5A – de acuerdo a lo que establece el fabricante.

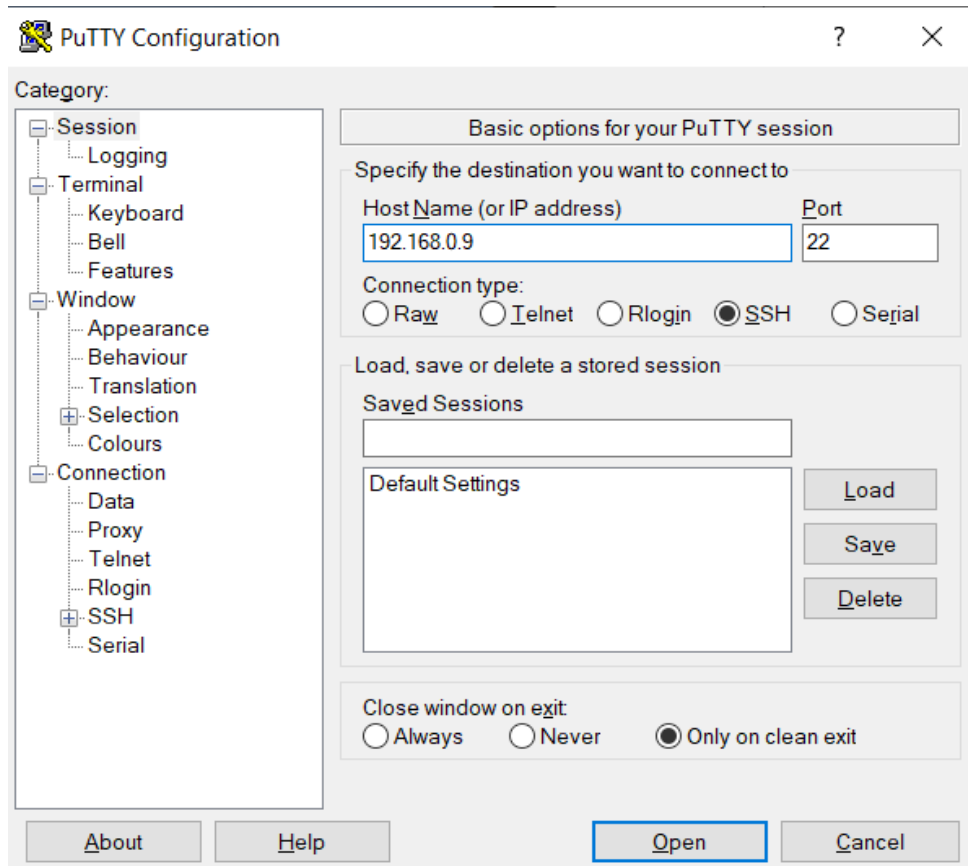
Ya encendido el Raspberry Pi, el Router de la red local asignará una dirección IP mediante DHCP, para conocer esta dirección e iniciar la configuración, se utiliza un ordenador conectado en la misma red local. El ordenador con la ayuda de algunos comandos en el terminal o con un programa puede dar a obtener la dirección IP – en la investigación se utilizó el programa Advanced IP Scanner en Windows. En la figura 24, se observa que la dirección IP asignada.



**Figura 34.** Detección de la dirección IP del Gateway

Elaboración propia

La configuración del Raspberry Pi fue realizada mediante comandos, los que fueron realizados de manera remota. Conocida la dirección IP y con la ayuda de un programa de acceso remoto (en esta investigación se usó PUTTY) se estableció la conexión mediante SSH. En la figura 25, se muestra la configuración para acceder de manera remota. Realizada la configuración de acceso, el Raspberry Pi solicitara el usuario y contraseña, ya autenticado el acceso, se inicia con el ingreso de los comandos de configuración.

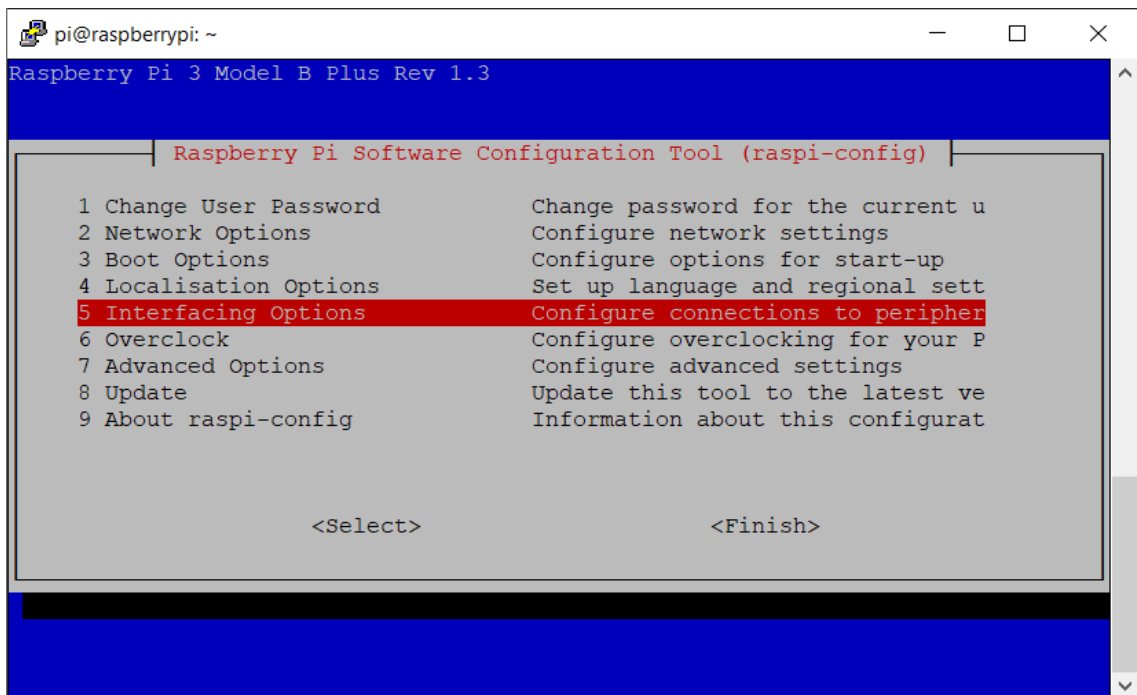


**Figura 25.** Acceso remoto mediante SSH

Elaboración propia

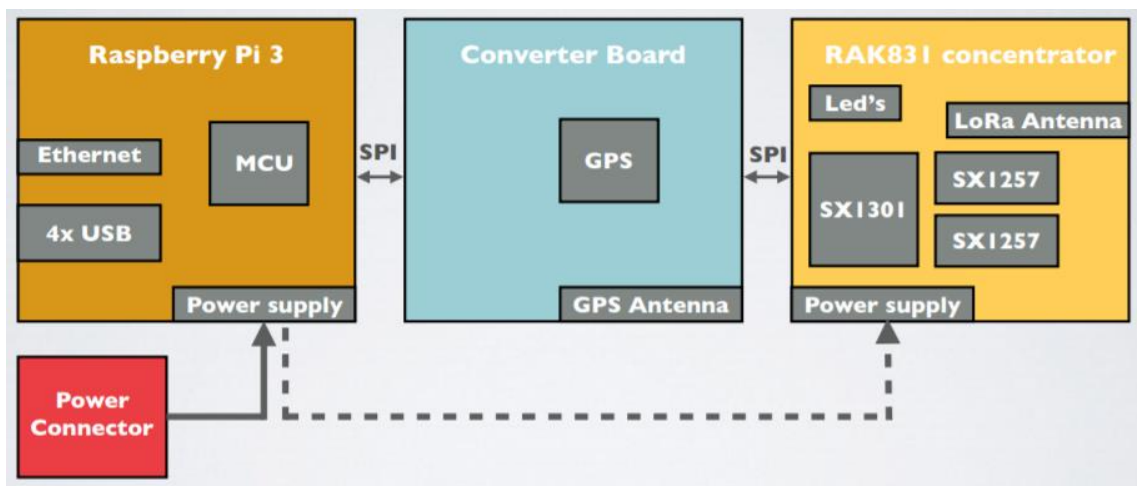
Primero se realiza la actualización del sistema operativo, con el comando *sudo apt update* y *sudo apt upgrade*, esta actualización demorara entre 10 a 20 minutos. El comando siguiente es abrir la herramienta de configuración de Raspberry Pi, con el comando *sudo raspi-config*, lo cual mostrara una ventana (figura 26) donde se puede realizar las configuraciones en el sistema operativo. Las configuraciones realizadas fueron la zona horaria y habilitar el interfaz SPI. Para cambiar la zona horaria, fue realizada ingresando a, *Localisation Options - Change Timezone*, seleccionado el área geografía *América* y zona horaria *Lima*. Al habilitar el SPI, fue realizado ingresando a *Interfacing Options – SPI*, esto habilitará el SPI (Serial Peripheral Interface, o en español, Interfaz Periférica Serial) en el Raspberry Pi. El bus SPI se utiliza para enviar datos entre el(los) microcontrolador(es) y los periféricos o módulos; En este caso, el envío

de datos hacia y desde el Raspberry Pi y el concentrador RAK831 a través de la placa convertidora, así como se detalla en el diagrama de la figura 27.



**Figura 26.** Ventana de la herramienta de configuración de Raspberry Pi.

Elaboración propia



**Figura 27.** Diagrama de conexión del Gateway

Fuente: (Lie, 2018)



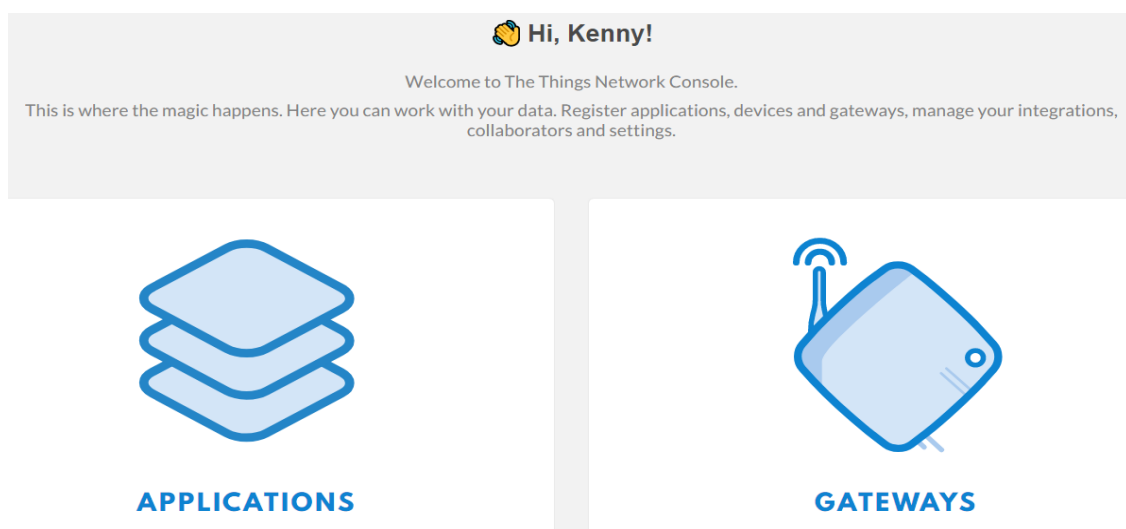
Antes de dar la funcionalidad como Gateway, debe realizarse la creación e instalación de las carpetas:

- Semtech LoRa library (V5.0.1), ubicado en la dirección `/opt/ttn-gateway/lora_gateway`. Este directorio contiene las fuentes de la biblioteca para construir una puerta de enlace basada en un receptor RF multicanal Semtech LoRa (también conocido como Gateway). La biblioteca también viene con un montón de programas de pruebas básicas que se utilizan para probar los diferentes subprogramas de la biblioteca.
- Semtech UDP Packet Forwarder (V4.0.1), ubicado en la dirección `/opt/ttn-gateway/packet_forwarder`. Es un programa que se ejecuta en el host de un Gateway de Lora que reenvía los paquetes RF recibidos por el concentrador a un servidor a través de un link IP/UDP, y emite paquetes RF que fueron enviados por el servidor. Los paquetes que son enviados y recibidos, se detallan a continuación
  - Enlace ascendente: paquetes de radio recibidos por el Gateway, con los metadatos agregados por el Gateway, remitidos al servidor. También puede incluir el estado de la puerta de enlace.
  - Enlace descendente: paquetes generados por el servidor, con metadatos adicionales, que serán transmitidos por el Gateway en el canal de radio. También puede incluir datos de configuración para la puerta de enlace.

Ambas carpetas pueden ser descargadas al Raspberry Pi (junto a todo los archivos), desde el repositorio GitHub, las direcciones web de ambas carpetas son; [https://github.com/Lora-net/lora\\_gateway](https://github.com/Lora-net/lora_gateway) y [https://github.com/Lora-net/packet\\_forwarder](https://github.com/Lora-net/packet_forwarder), respectivamente.

Por último, para otorgar la funcionalidad de Gateway, en la carpeta **lora\_pkt\_fwd** (ubicado en la dirección; /opt/ttn-gateway/packet\_forwarder), se descarga la configuración para establecer la comunicación desde el Gateway hacia el servidor de The Things Network. Esta configuración a descargar debe ser en relación a la banda en que está establecido en el módulo RAK831 del Gateway, en este caso viene a ser la banda US915, debido que el diseño de hardware. Por tal motivo, la configuración a descargar está en la dirección web; [https://github.com/TheThingsNetwork/gateway-conf/blob/master/US-global\\_conf.json](https://github.com/TheThingsNetwork/gateway-conf/blob/master/US-global_conf.json), desde el repositorio GitHub, el cual es proporcionado por The Things Network. La configuración descargada de nombre *US-global\_conf.json* debe reemplazarse con la configuración predeterminada *global\_conf.json*. Reemplazado la configuración predeterminada y para finalizar debe reiniciarse el Raspberry Pi.

Ya culminada con la instalación en el Raspberry Pi, el siguiente paso es registrar en la página de The Things Network, para lo cual es necesario crear una cuenta, es de manera gratuita. Creada la cuenta ya es posible registrar un Gateway o una aplicación – ver figura 28.



**Figura 28.** Página de The Things Network.  
Elaboración propia



En las figuras 29 y 30, se muestra los datos ingresados para el registro. Lo ingresado en el espacio **Gateway EUI** (Extended Unique Identifier), es el identificador que es proporcionado por el Gateway. Para obtener el identificador se puede realizar obteniendo la dirección MAC del puerto **eth0** del Raspberry Pi (el puerto **wlan0**, es deshabilitado automáticamente para el **Gateway EUI**), lo cual es **b8:27:eb:05:1c:54**; luego de obtener la dirección MAC, se agrega **fffe** después del tercer byte, para lo que queda de la siguiente manera **b8:27:eb:fffe05:1c:54**, finalmente, quitar los “:” y cambiarlos a mayúscula, resultando es el Gateway EUI, **B827EBFFFE051C54**. En el espacio **Description**, es una descripción con lo que será visualizado el Gateway.

**REGISTER GATEWAY**

**Gateway EUI**  
The EUI of the gateway as read from the LoRa module

B8 27 EB FF FE 05 1C 54 8 bytes

**I'm using the legacy packet forwarder**  
Select this if you are using the legacy [Semtech packet forwarder](#).

**Description**  
A human-readable description of the gateway

LoraWAN-Gateway

**Frequency Plan**  
The [frequency plan](#) this gateway will use

United States 915MHz

**Router**  
The router this gateway will connect to. To reduce latency, pick a router that is in a region which is close to the location of the gateway.

ttn-router-us-west

**Figura 29.** Registro del Gateway - parte 1.

Elaboración propia

En **Frequency Plan**, es en la banda de frecuencia que trabaja el Gateway, el diseño del hardware del módulo y la configuración realizada son para la regulación regional de US915, por lo tanto, se selecciona la opción de **United States 915MHz**. En el espacio de

**Router**, la opción seleccionada va en relación de la configuración *global\_conf.json* del Raspberry Pi y el plan de frecuencia seleccionada, esto para enviar los paquetes hacia el enrutador designado. En la figura 30, se designa la ubicación del Gateway y la colocación de la misma (dentro o fuera). Configurado todos los espacios, se registra el Gateway.

**Location**  
The exact location of you gateway. This will be used if your gateway cannot determine its location by itself. Set a location by clicking on the map.

lat -14.95112318  
lng -70.26696136

Google

Datos de mapas ©2020 Términos de uso Notificar un problema de Maps

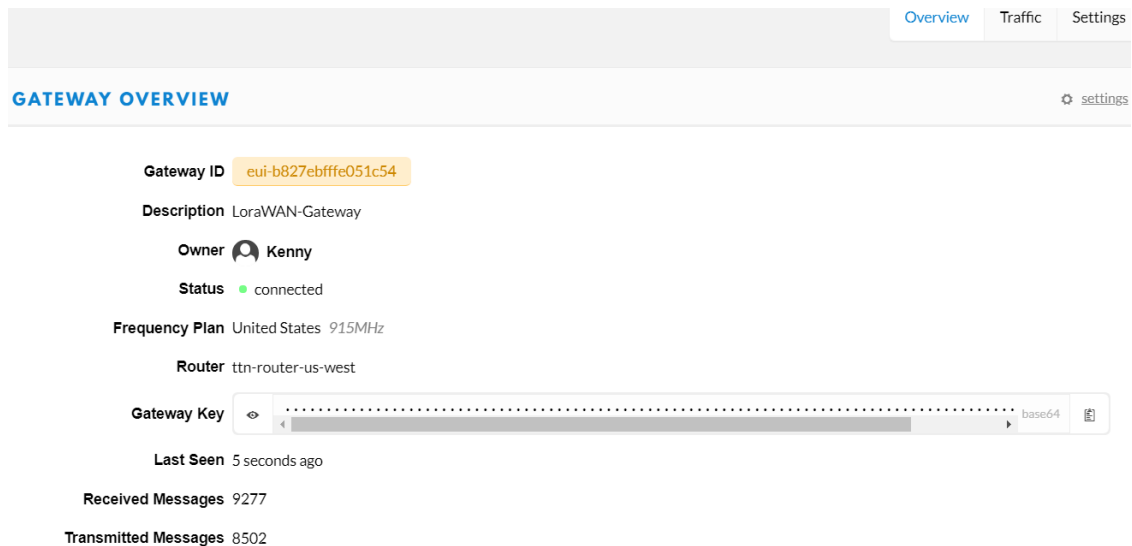
**Antenna Placement**  
The placement of the gateway antenna

indoor outdoor

Cancel Registrar Gateway

**Figura 30.** Registro del Gateway - parte 2.

Elaboración propia



**Figura 31.** Resumen del Gateway registrado.

Elaboración propia

Culminado con el registro, el Gateway debe presentar el estado de conectado, como se muestra en la figura 31. Eso es todo en cuanto a la relación de la etapa 1, el Gateway está dispuesto a ser usado, la visualización de datos enviados y recibidos se puede observar ingresando a la pestaña de **Traffic**, sin embargo, aún no se puede visualizarse, porque no hay ningún End Node conectado.

## **Etapas 2: Configuración del nodo final localizador (end node)**

En la segunda Etapa se usará el End Node, el cual es detallado en la parte 3.1.1, la configuración realizada dentro del End Node fue realizada de acuerdo a la siguiente dirección; <https://github.com/RAKWireless/RAK813-BreakBoard>, proporcionada por el fabricante. Culminada con toda la actualización del software interno del End Node, el siguiente paso fue la creación de una aplicación en The Thing Network. Este proceso fue accediendo a la cuenta creada anteriormente, los pasos realizados se describen a continuación.

Antes de que puedas comunicarte con los dispositivos, se debe añadir la aplicación a la red y registrar los dispositivos en ella. Los usuarios pueden crear aplicaciones y pueden autorizar el acceso de otros usuarios a las aplicaciones. Las aplicaciones se identifican mediante una ID de aplicación única. Cada aplicación tiene una o más llaves de acceso para acceder a los datos de la aplicación y / o administrar dispositivos. Existen múltiples opciones para integrar aplicaciones con The Things Network, que van desde trabajar directamente con APIs, pasando por SDKs más amigables o integraciones de plataformas con un solo clic.

**ADD APPLICATION**

**Application ID**  
The unique identifier of your application on the network

rak811\_arduino\_wisnode

**Description**  
A human readable description of your new app

End Node Location

**Application EUI**  
An application EUI will be issued for The Things Network block for convenience, you can add your own in the application settings page.

EUI issued by The Things Network

**Handler registration**  
Select the handler you want to register this application to

ttn-handler-us-west

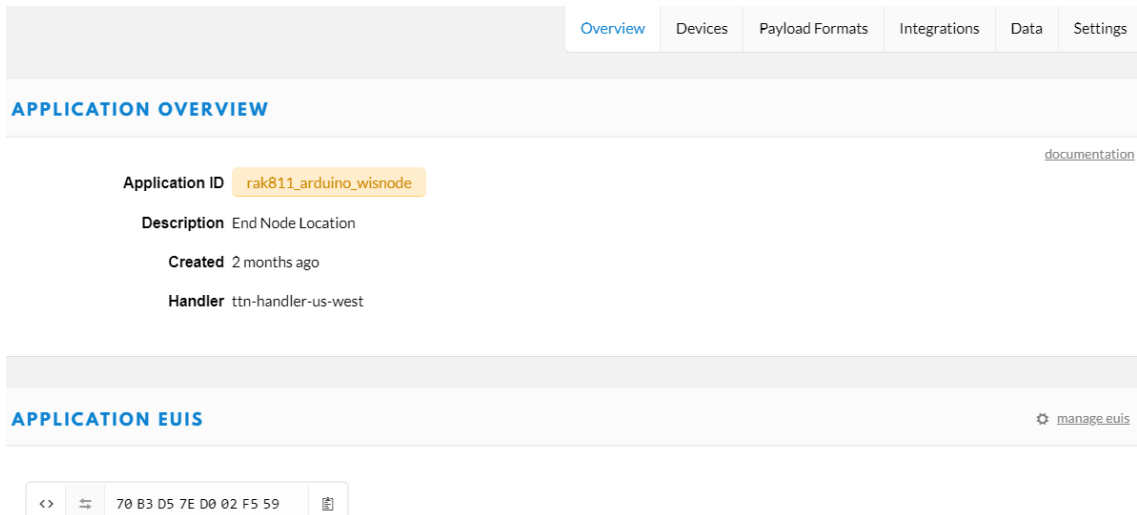
Cancel Add application

**Figura 32.** Agregar una aplicación en The Things Network.

Elaboración propia

Lo necesario para agregar una aplicación asignar un nombre, el cual es único para la identificación dentro de la red LoraWAN y el servidor de The Things Network, esta es la **Application ID** (identificador de la aplicación). Después de esto se introduce la descripción de la aplicación, esta puede ser cambiada posteriormente, a diferencia que el identificador de la aplicación. **Application EUI**, al igual que el EUI del Gateway, es un

identificador hexadecimal de 8 byte, el cual será determinado por el servidor The Things Network; por último, seleccionar el controlador en el que se registrará esta aplicación, al igual que en la elección del enrutador del Gateway, el controlador debe ser de la misma dirección. Al terminar con el registro, se mostrará los datos ingresados y el **Application EUI** generado, ver la figura 33.



The screenshot displays the 'APPLICATION OVERVIEW' section of a web interface. It includes a navigation bar with tabs for 'Overview', 'Devices', 'Payload Formats', 'Integrations', 'Data', and 'Settings'. The main content area shows the following details for an application:

- Application ID:** rak811\_arduino\_wisnode
- Description:** End Node Location
- Created:** 2 months ago
- Handler:** ttn-handler-us-west

Below this, the 'APPLICATION EUIS' section is visible, showing a hex string: 70 B3 D5 7E D0 02 F5 59.

**Figura 33.** Datos del registro de la Aplicación.

Elaboración propia

Terminada con el registro de la aplicación, se agrega los dispositivos finales. En la pestaña de **Device**, dentro de la ventana de la aplicación creada, se realiza el registro del dispositivo final. En la figura 34, se observa los requisitos para registrar un dispositivo final o End Node, al igual que la aplicación, es necesario ingresar un identificador único dentro del servidor de The Things Network (**Device ID**), se ingresó la serie del End Node. En los espacios de **Device EUI**, **App Key** y **App EUI**, son generados automáticamente por el servidor, para luego ser configurada con estos datos el End Node, la definición de esos espacios se explica a continuación:



Overview **Devices** Payload Formats Integrations Data Settings

**REGISTER DEVICE** [bulk import devices](#)

**Device ID**  
This is the unique identifier for the device in this app. The device ID will be immutable.

201845

**Device EUI**  
The device EUI is the unique identifier for this device on the network. You can change the EUI later.

this field will be generated

**App Key**  
The App Key will be used to secure the communication between you device and the network.

this field will be generated

**App EUI**

70 B3 D5 7E D0 02 F5 59

Cancel Register

**Figura 34.** Registro de dispositivo final - End Node.

Elaboración propia

La figura 35, muestra las llaves e identificadores para ser configurados en el dispositivo final y de esta manera activar en el servidor.

### DEVICE OVERVIEW

**Application ID** rak811\_arduino\_wisnode

**Device ID** 201845

**Description** End\_Node\_Location

**Activation Method** OTAA

**Device EUI** <> ⇅ 00 1F 82 8B 98 AA 9F B2 [📄]

**Application EUI** <> ⇅ 70 B3 D5 7E D0 02 F5 59 [📄]

**App Key** <> ⇅ 👁 ..... [📄]

**Device Address** <> ⇅ 26 02 2A 90 [📄]

**Network Session Key** <> ⇅ 👁 ..... [📄]

**App Session Key** <> ⇅ 👁 ..... [📄]

**Status** ● 1 minute ago

**Frames up** 24 [reset frame counters](#)

**Frames down** 25

**Figura 35.** Resumen de los datos para el dispositivo final.

Elaboración propia

```
Selected LoRaWAN 1.0.2 Region: US915
UART1 work mode: RUI_UART_NORAMAL
BME680 init success.
LIS3DH init OK.
GPS Init OK.GPS timeout:30s
autosend_interval: 90s
Initialization OK,Current work_mode:LoRaWAN, join_mode:OTAA,
Class: A

OTAA Join Start...
[LoRa]:Join Success
OK
Start Search Satellite(about 30 seconds) ...

SENDING(With \r\n)
at+set_config=device:restart
```

SEND

**Figura 36.** Acceso al servidor mediante OTAA.

Elaboración propia

En la figura 36, se muestra la activación del dispositivo donde indica la versión de LoraWAN, que es la 1.0.2. También, indica la región siendo US915, siendo la banda donde transmitirá el dispositivo, en esta misma banda transmite el Gateway. Muestra la activación del GPS y el tiempo que tomará para esperar la información GPS para ser transmitida. Posteriormente indica el modo de sesión, lo que es OTAA, al igual que la configuración realizada en la plataforma de The Things Network. Finalmente indica el tipo de clase utilizado para la transmisión y recepción, en este caso es la clase A, el sistema de dispositivo de menor potencia para aplicaciones que sólo requieren una comunicación de enlace descendente desde el NS poco después de que el dispositivo haya enviado una transmisión de enlace ascendente. Las comunicaciones de enlace descendente desde el NS en cualquier otro momento deben esperar hasta el siguiente enlace ascendente





programado. En la segunda parte de la figura 36, indica el inicio del modo de activación y el mensaje desde el servidor, el que indica que se inició la sesión; y posteriormente preparar la información satelital para enviarla.



## CAPÍTULO IV

### RESULTADOS Y DISCUSIÓN

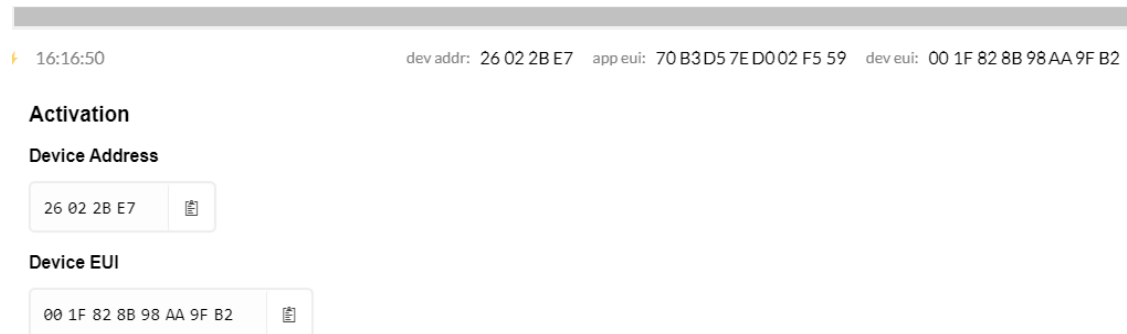
#### 4.1. COMPORTAMIENTO DE LA COMUNICACIÓN

Los resultados obtenidos en las pruebas de conexión entre The Things Network (en adelante TTN), Gateway y el End Node serán mostrados a continuación, esto mediante capturas de pantallas y copias de resultados obtenidos. Cabe precisar que TTN, muestra los datos obtenidos mediante el End Node, además, registra el tráfico que existe en el Gateway, ambos en tiempo real. Conforme se identifiquen las imágenes, estas serán detalladas a lo ocurrido.

##### 4.1.1. Comportamiento en la activación.

La arquitectura empleada es la descrita en la figura 22, esta primera prueba fue realizada con el fin de observar el comportamiento en la activación y en la transferencia de datos del End Node hacia el Gateway y a TTN. Los resultados se detallan a continuación:

La primera comunicación realizada por el End Node, es la solicitud de la activación en TTN, mediante la opción de OTAA. En la figura 37 se muestra la primera trama que se envía desde el End Node, para la activación del mismo en TTN, para esto envía los parámetros previamente establecidos en el End Node como, Device Address, Device EUI (Dev EUI) y App EUI.



**Figura 47.** Trama enviada por el End Node para la activación.

Elaboración propia

El resultado obtenido a partir del puerto consola del End Node fue la siguiente:

```
Selected LoRaWAN 1.0.2 Region: US915
UART1 work mode: RUI_UART_NORAMAL

GPS Init OK.
GPS timeout: 30s
autosend_interval: 40s
Initialization OK, Current work_mode: LoRaWAN, join_mode: OTAA, Class: A

OTAA Join Start...
[LoRa]: Join Success
OK
```

Lo primero fue la inicialización del dispositivo, indicando la versión de LoRaWAN y la región de banda seleccionada para la operación; en seguida, muestra la inicialización del GPS, *GPS timeout* describe el tiempo máximo que tomara el GPS para obtener las coordenadas y *autosend\_interval* indica el intervalo de tiempo en que el dispositivo enviará una nueva trama; *Current work\_mode: LoRaWAN*, *join\_mode: OTAA* y *Class: A*, indican el modo de trabajo del dispositivo, el modo de activación en TTN y la clase de comunicación que realizará el End Node, respectivamente. El espacio siguiente indica que se está realizando el intento de activación mediante OTAA y en el espacio siguiente se obtiene la respuesta, donde indica la activación exitosa.

La trama de activación enviada por el End Node pasa por el Gateway, en el mismo se puede observar el Dev EUI y el App EUI; también un Payload, esta carga útil es generada

por el Gateway y enviado hacia TTN. En la figura 38, se muestra esos datos, los cuales son para la solicitud de activación por parte del End Node, Join Request.

16:16:50      903.9      4/5      SF 10 BW 125      370.7      app eui: 70 B3 D5 7E D0 02 F5 59      dev eui: 00 1F 82 8B 98 AA 9F

**Join Request**

**Dev EUI**

00 1F 82 8B 98 AA 9F B2

**App EUI**

70 B3 D5 7E D0 02 F5 59

**Physical Payload**

00 59 F5 02 D0 7E D5 B3 70 B2 9F AA 98 8B 82 1F 00 E7 EC 43 74 81 16

**Event Data**

```
1 {
2   "gw_id": "eui-b827ebfffe051c54",
3   "payload": "AFn1AtB+1bNwsp+qmIuCHwDn7EN0gRY=",
4   "dev_eui": "001F828B98AA9FB2",
5   "lora": {
6     "spreading_factor": 10,
7     "bandwidth": 125,
```

**Figura 38.** Trama de datos en el Gateway, obtenidos en la solicitud de activación del End Node.

Elaboración propia

En la sección de Event Data, se puede observar la identidad del Gateway y el identificador del End Node(Dev EUI), toda esta información es obtenida del Physical Payload. En esta sección, el Payload es codificado en base64, debido a esto no es mostrado al igual que en la sección Physical Payload, esto puede ser comprobado usando un decodificador (<https://lorawan-packet-decoder-0ta6puiniaut.runkit.sh/>).

```
{
  "gw_id": "eui-b827ebfffe051c54",
  "payload": "AFn1AtB+1bNwsp+qmIuCHwDn7EN0gRY=",
  "dev_eui": "001F828B98AA9FB2",
  "lora": {
    "spreading_factor": 10,
    "bandwidth": 125,
    "air_time": 370688000
  },
  "coding_rate": "4/5",
  "timestamp": "2020-08-08T21:16:50.083Z",
  "rssi": -89,
```



```
"snr": 11.2,  
"app_eui": "70B3D57ED002F559",  
"frequency": 903900000  
}
```

Al interior de la carga útil se tiene la información respecto a la comunicación LoRa adicional; como factor de propagación, ancho de banda y tiempo de transmisión. En esta misma trama enviada por el Gateway hacia TTN, se puede observar las características de transmisión como la tasa de codificación, el tiempo en que se realizó la comunicación, el indicador de la fuerza de señal recibida, la relación señal ruido, el identificador de la aplicación relacionada con el End Node y por último la frecuencia en el que se realizó la comunicación con el End Node.

Después de que el End Node envía la solicitud de activación, TTN responde esto hacia el End Node, por medio del Gateway, con una aceptación de activación o inicio, Join Accept, la trama que se observa en el Gateway muestra una carga útil (figura 39 , donde presenta características como el identificador del Gateway, el factor de propagación en la comunicación LoRa, el ancho de banda hacia el End Node, tiempo de transmisión hacia el End Node, la tasa de codificación, el momento en que se realizó la comunicación y la frecuencia en que se remite la respuesta.

16:16:54 923.3 4/5 SF 10 BW 500 82.4

### Join Accept

#### Physical Payload

20 EB 10 B3 26 C2 4F B6 E4 91 5B 3F 48 91 26 49 A3

#### Event Data

```
1 {
2   "gw_id": "eui-b827ebfffe051c54",
3   "payload": "IOsQsybCT7bkkVs/SJEmSaM=",
4   "lora": {
5     "spreading_factor": 10,
6     "bandwidth": 500,
7     "air_time": 82432000
8   },
9   "coding_rate": "4/5",
10  "timestamp": "2020-08-08T21:16:54.081Z",
11  "frequency": 923300000
12 }
```

**Figura 39.** Trama de datos en el Gateway, obtenidos en la respuesta de activación del End Node.

Elaboración propia

La respuesta por TTN hacia el End Node, se muestra en la consola del mismo como a continuación:

```
[LoRa]:Join Success
OK
```

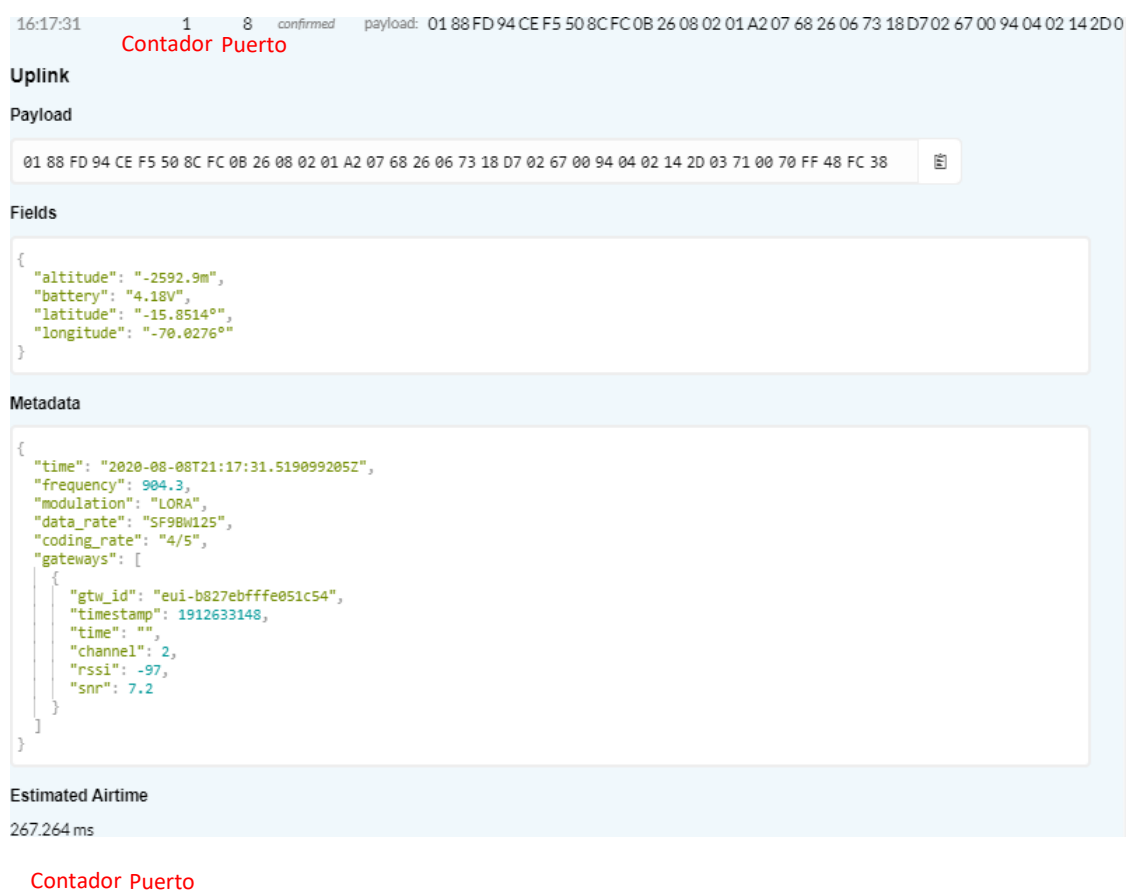
#### 4.1.2. Comportamiento en el envío de información de localización.

Ya activado el End Node, comenzará a enviar a la localización hacia TTN, a través del Gateway, para lo cual toma el tiempo configurado para obtener las coordenadas (30 segundos), como se muestra a continuación:

```
Wake up
Start Search Satellite(about 30 seconds) ...
GPS data:latitude: -15.851458, longitude: -70.027623, altitude: -2592.9m
```

El intervalo de envío es de 40 segundos, esto quiere decir que tomará este tiempo para quedar suspendido y cumplido con este tiempo, despertará (*Wake up*) y comenzará a obtener la localización y posteriormente enviarlo hacia TTN por medio del Gateway.

En la plataforma de TTN, analizando la trama del contador 1 del puerto 8, se observa la información enviada por el End Node, en la carga útil lleva toda la información obtenida de las coordenadas, el cual traduciéndolas mediante una línea de comandos (Anexo 1) se puede mostrar un resultado entendible, como se muestra en la sección *Fields* de la figura 40.



**Figura 40.** Trama de información enviada por el End Node.

Elaboración propia

En el espacio de *Metadata*, se muestra la información de la transmisión como la frecuencia enviada, la modulación, el factor de propagación y el ancho de banda

(data\_rate), la tasa de codificación, también, el canal de envío el indicador de la fuerza de la señal y la relación señal ruido, por último, se obtiene una estimación del tiempo de transmisión (para comprender la obtención del tiempo de transmisión ver Anexo 2).



**Figura 41.** Trama de información que pasa por el Gateway, a partir del End Node hacia TTN, parte 1.

Elaboración propia

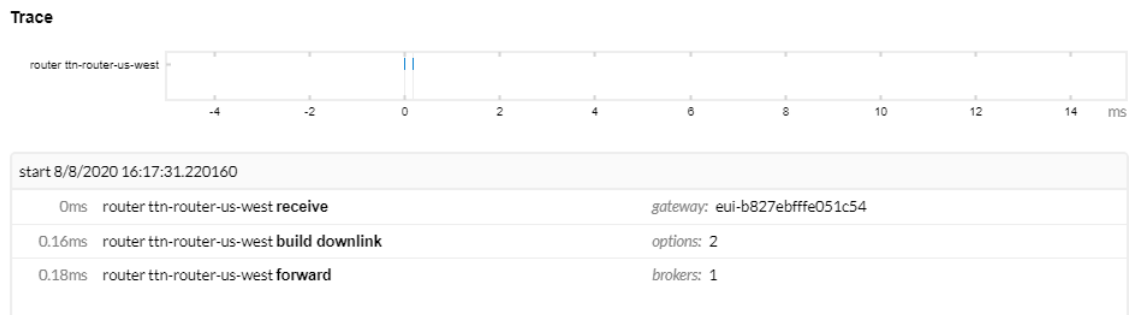
Al ver la trama enviada por el Gateway, se observa la identificación que asigno TTN al End Node (26022BE7) y la carga útil que contiene la información enviada por el End Node. Ya que toda información enviada por el End Node, pasa por el Gateway, la información mostrada en la sección de Metadata de la figura 40, también se observa en el Gateway, además, se agrega el contador correspondiente a la trama del End Node (“f\_cnt”), como se muestra a continuación:

```
{  
  "gw_id": "eui-b827ebfffe051c54",
```



```
"payload":  
"gOcrAiaEAQADBwMHCJqwYuy5SZWqbXVYZw/1l9iOYY4Tl6vFV5v8ICLnU1xIquqtf  
CxqQRk6qg==",  
"f_cnt": 1,  
"lora": {  
  "spreading_factor": 9,  
  "bandwidth": 125,  
  "air_time": 349184000  
},  
"coding_rate": "4/5",  
"timestamp": "2020-08-08T21:17:31.585Z",  
"rssi": -97,  
"snr": 7.2,  
"dev_addr": "26022BE7",  
"frequency": 904300000  
}
```

Por último, el Gateway muestra el trazo del paquete enviado hacia TTN, en donde por el momento solo indica la llegada hacia el Router de TTN (ver figura 42).



**Figura 42.** Trama de información que pasa por el Gateway, a partir del End Node hacia TTN, parte 2.

Elaboración propia

Con la llegada de la trama proveniente desde el End Node hacia TTN, este último responde con una trama que pasa por el Gateway, en el que detalla la identificación del End Node, en la misma trama va con una carga útil, en el cual responde al End Node.

16:17:32      923.3      lora   4/5      SF 12 BW 500      247.8      1 dev addr: 26 02 2B E7      payload size: 12 bytes

**Downlink**

**Dev Address**

26 02 2B E7

**Network:** The Things Network  
**Net ID:** 0x13  
**Region:** World

**Physical Payload**

60 E7 2B 02 26 20 01 00 B1 CA 05 E8

**Event Data**

```
2   "gw_id": "eui-b827ebfffe051c54",
3   "payload": "YOcrAiYgAQcXygXo",
4   "f_cnt": 1,
5   "lora": {
6     "spreading_factor": 12,
7     "bandwidth": 500,
8     "air_time": 247808000
9   },
10  "coding_rate": "4/5",
11  "timestamp": "2020-08-08T21:17:32.585Z",
12  "dev_addr": "26022BE7",
13  "frequency": 923300000
14 }
```

**Figura 43.** Trama de respuesta que pasa por el Gateway, a partir del TTN hacia el End Node, parte 1.

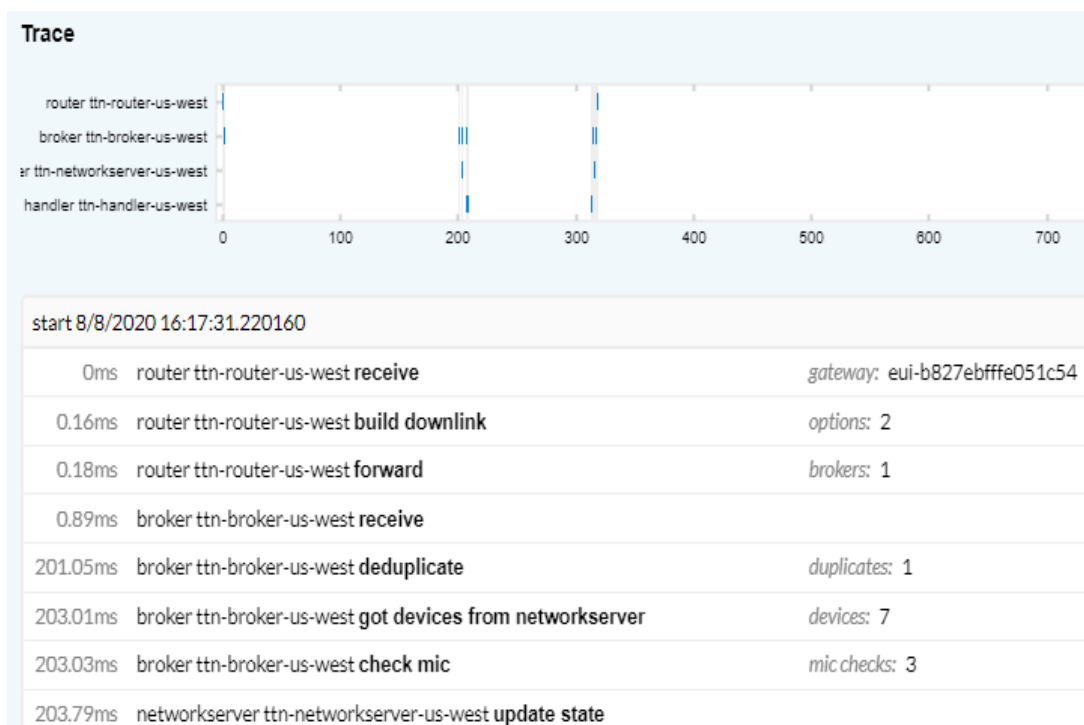
Elaboración propia

En esta respuesta reenviada por el Gateway, contiene al contador que se está respondiendo (“f\_cnt”), el factor de propagación, el ancho de banda, el tiempo de transmisión, la tasa de codificación, la hora de la comunicación, el identificador proporcionado por TTN hacia el End Node y por último la frecuencia de transmisión.

```
{
  "gw_id": "eui-b827ebfffe051c54",
  "payload": "YOcrAiYgAQcXygXo",
  "f_cnt": 1,
  "lora": {
    "spreading_factor": 12,
    "bandwidth": 500,
    "air_time": 247808000
  }
}
```

```
},  
  "coding_rate": "4/5",  
  "timestamp": "2020-08-08T21:17:32.585Z",  
  "dev_addr": "26022BE7",  
  "frequency": 923300000  
}
```

Por último, en la respuesta de TTN incluye la información del trazo de la ruta seguida por el paquete dentro de la red de TTN (figura 44). Conforme a lo detallado en la sección 2.4, de esta tesis, para explicar el trazo, se haría más amplio la investigación, por lo que se recomienda que esta etapa se pueda realizar en otras investigaciones y profundizar más sobre este aspecto.



**Figura 44.** Trama de respuesta que pasa por el Gateway, a partir del TTN hacia el End Node, parte 2.

Elaboración propia

## 4.2. EXACTITUD DE LA LOCALIZACIÓN

Para determinar la exactitud del GPS, se realizó 20 pruebas, en las que se tiene un aparato adicional para medir las coordenadas GPS de manera más exacta (coordenadas de referencia) que el End Node. El GPS y el End Node son portados por un sujeto en movimiento, el cual realiza el movimiento a la misma velocidad que se trasladan las vicuñas (este movimiento se refiere cuando las vicuñas no visualizan ningún riesgo). Los resultados se muestran en la siguiente tabla:

Número de prueba	Coordenadas de referencia		Coordenadas End Node		Diferencia (metros)
	Latitud	Longitud	Latitud	Longitud	
1	-15.849577	-70.024469	-15.8495	-70.0244	11.234853
2	-15.842536	-70.021202	-15.8425	-70.0211	11.64812
3	-15.841916	-70.016946	-15.8419	-70.0169	5.253693
4	-15.84	-70.017497	-15.8399	-70.0175	11.044855
5	-15.838973	-70.017305	-15.839	-70.0173	3.078153
6	-15.836028	-70.016936	-15.8361	-70.017	10.519545
7	-15.83564	-70.015912	-15.8357	-70.016	11.457966
8	-15.835373	-70.015368	-15.8354	-70.0154	4.508932
9	-15.834522	-70.015616	-15.8345	-70.0156	3.001351
10	-15.834201	-70.015808	-15.8341	-70.0157	16.120379
11	-15.833588	-70.015966	-15.8336	-70.0159	7.204449
12	-15.831284	-70.016009	-15.8312	-70.016	9.322861
13	-15.830937	-70.015768	-15.831	-70.0158	7.75593
14	-15.827444	-70.01478	-15.8274	-70.0147	9.829538
15	-15.827019	-70.013978	-15.827	-70.014	3.180268
16	-15.826144	-70.012552	-15.8261	-70.0125	7.451499
17	-15.82579	-70.011931	-15.8258	-70.0122	28.862973
18	-15.823176	-70.004021	-15.8231	-70.004	8.693205
19	-15.82331	-69.999923	-15.8233	-69.9999	2.710188
20	-15.823826	-69.995982	-15.8238	-69.9959	9.268514

**Tabla 5:** Resultados de pruebas de coordenadas tomadas por el End Node.

Elaboración propia

La tabla 5 muestra las coordenadas de referencia y las coordenadas tomadas por el End Node, las cuales son en grados decimales. Para obtener la distancia entre las coordenadas de referencia y del End Node, fue con la ayuda del software ArcMap. Las mediciones de distancia son más precisas cuando los datos de entrada estén en un sistema de coordenadas proyectadas en equidistancia (es decir UTM), sobre todo para reducir errores en el cálculo; por lo cual las coordenadas de la tabla fueron proyectadas a este sistema (UTM) en el mismo software. El proceso del cálculo de distancia se pudo llevar a cabo gracias a la herramienta *Point distance* (Distancia de punto) localizada en las herramientas de proximidad del ArcToolbox.

Analizando la tabla se observa que la media aritmética de la distancia entre las coordenadas, es de **9.11 metros**, el cual fue calculado con la siguiente formula:

$$\bar{X} = \frac{\sum_{i=1}^n x_i}{n}$$

Dónde: x, es la diferencia entre las coordenadas y n, la cantidad de pruebas (20).

La media aritmética de la distancia entre las coordenadas del GPS de referencia con las tomadas por el End Node, se encuentra en un rango adecuado para determinar la ubicación del camélido. Además, esta distancia puede ser minimizada si se encuentra en reposo el animal. Cabe señalar que esta diferencia indica la precisión de las coordenadas tomadas por el End Node.

### 4.3. COBERTURA

Para determinar la ubicación del Gateway se ha realizado mediante el análisis de la superficie, mediante el software de ArcMap y Radio Mobile, en las cuales se han observado la altitud e interferencias mediante curvas de nivel y diagramas raster, como se observa en la figura 45. La ubicación del Gateway es en las coordenadas -14.94148, -

70.25664 (grados decimales) y a una altitud de 4116 msnm, a una distancia de 337 metros del centro poblado.

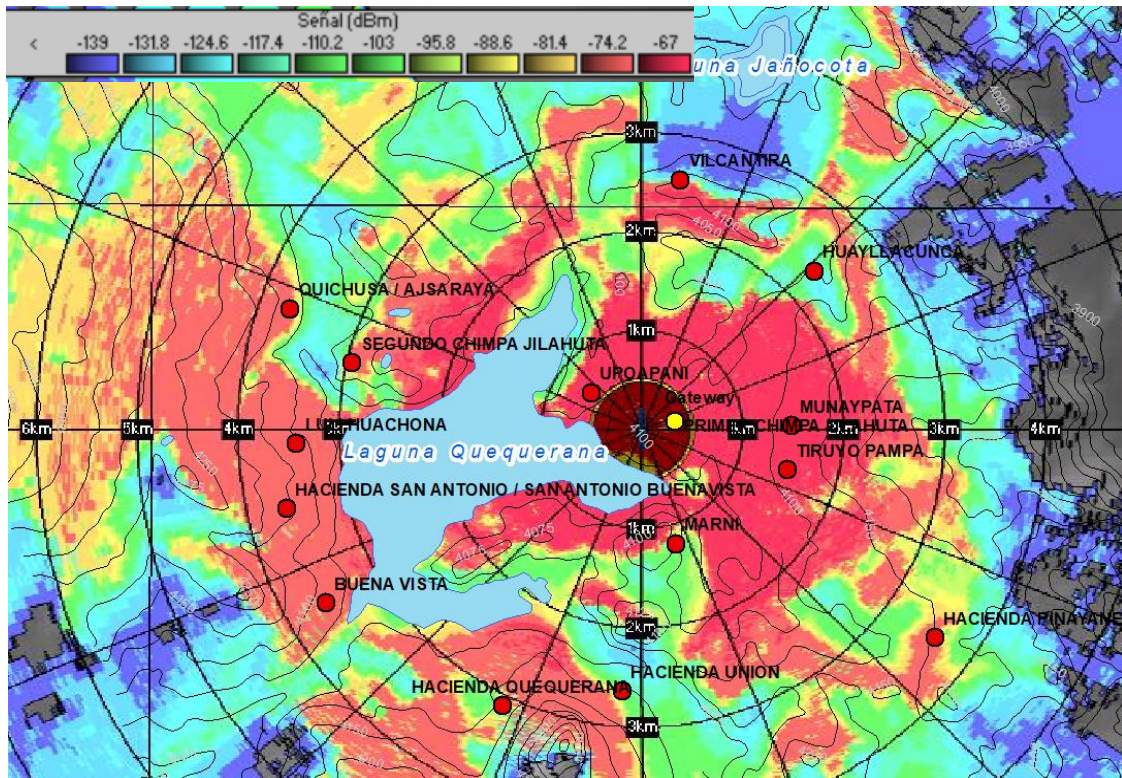


**Figura 45.** Ubicación del Gateway.

Elaboración propia

Con una simulación realizada en el software Radio Mobile, se ha obtenido la cobertura de la comunicación entre el Gateway y los End Node, estos datos pueden apreciarse en la figura 46, en donde la cobertura de la comunicación pasa alrededor de la Laguna Quequenara y llegando hacia otras comunidades, obtenido una cobertura continua por más de 3 Km (en el anexo 3 se mostrará con más amplitud el mapa de la cobertura).





**Figura 46.** Cobertura de comunicación LoRa.

Elaboración propia

En la simulación realizada, Radio Mobile proporciona más de 300mil datos generados sobre la sensibilidad de la señal con respecto al azimut de la antena y la distancia de ella, debido a que es una gran cantidad de datos, esto se resumirá con la localización de 10 puntos dispersos de acuerdo a la situación de la superficie del área, como se observa en la figura 47. Los End Node 9 y 10 se situaron en zonas profundas sin línea de vista con el Gateway, los End Node 2, 4 y 7, se ubican al extremo del lago, debido a la cobertura que muestra en la figura 46; los demás nodos se encuentran más cercanos hacia el Gateway, pero algunos no muestran una línea de vista directa.



**Figura 47.** Ubicación de los End Node.

Elaboración propia

En la tabla 6, resume el nivel de señal en cada una de los End Node figados como muestra en la figura 47, en este se observa que los End Node 9 y 10 son las que presentan menos nivel de señal, sin embargo, la comunicación es posible, ya que, la sensibilidad mínima es de -139dBm.



**Tabla 6:** Resumen de la cobertura en relación a los End Node.

End Node	Latitud	Longitud	Altura (msnm)	Nivel de señal (dBm)	Distancia (Km)	Obstrucción
End Node 1	-14.94943	-70.2686	4061	-63.3	1.56	NO
End Node 2	-14.9508	-70.29832	4222.1	-109.5	4.59	SI
End Node 3	-14.95228	-70.24809	4054.1	-62.4	1.51	NO
End Node 4	-14.96744	-70.27248	4097.1	-79.8	3.35	SI
End Node 5	-14.91782	-70.2798	4115	-102.7	3.62	SI
End Node 6	-14.93307	-70.24809	4051	-61.2	1.31	NO
End Node 7	-14.96036	-70.28423	4048	-96.1	3.63	SI
End Node 8	-14.96363	-70.2614	4077	-88.6	2.51	SI
End Node 9	-14.91816	-70.26058	4046	-114.8	2.63	SI
End Node 10	-14.93122	-70.28609	4078	-115.3	3.36	SI

Elaboración propia

#### 4.4. DISTANCIA

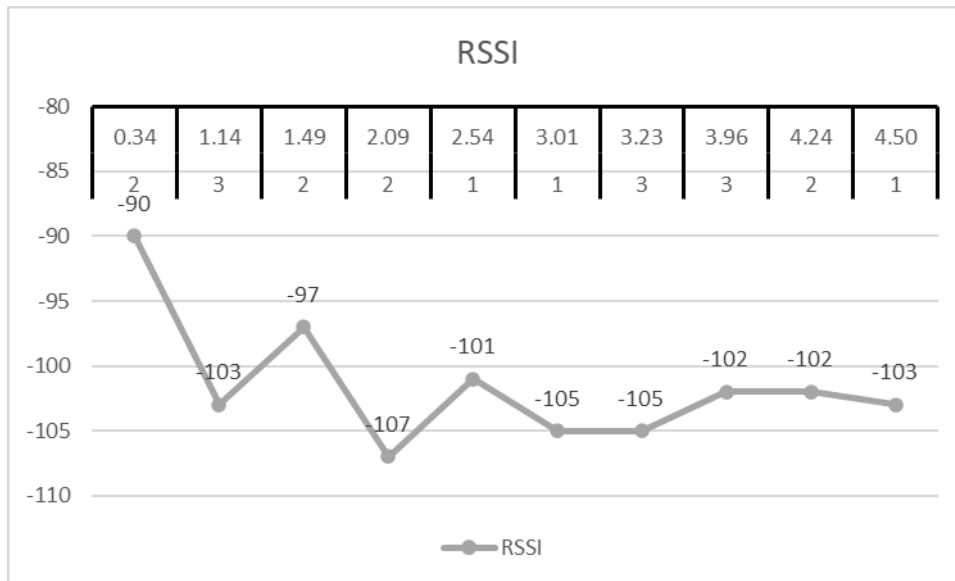
Para obtener estos resultados se realizaron 10 pruebas, el sujeto que portaba el End Node estaba en movimiento. Estos resultados son ordenados por la distancia (en kilómetros) del End Node hacia el Gateway, de manera ascendente. En los resultados se obtuvieron el RSSI (en dbm), SNR (en db), el factor de dispersión (SF), el ancho de banda, la frecuencia transmitida los datos. Además, en los resultados se agregó el estado de la línea de vista, donde, 1 indica que la línea de vista es buena, 2 que es regular y 3 que es mala. Los resultados obtenidos se muestran en la tabla 7.

**Tabla 7:** Resultados de la distancia (LoRa).

<b>N</b>	<b>Distancia (Km)</b>	<b>RSSI (dbm )</b>	<b>SNR (db)</b>	<b>SF</b>	<b>Ancho de banda (KHz)</b>	<b>Frecuencia (MHz)</b>	<b>Estado de línea de vista</b>
1	0.34	-90	8.8	7	125	904.9	2
2	1.14	-103	-6.8	8	125	904.9	3
3	1.49	-97	6.2	8	125	904.1	2
4	2.09	-107	-8.8	10	125	904.7	2
5	2.54	-101	-0.8	8	125	904.5	1
6	3.01	-105	-10.8	9	125	904.9	1
7	3.23	-105	-2.2	9	125	905.3	3
8	3.96	-102	-12.8	9	125	904.9	3
9	4.24	-102	-0.8	10	125	904.5	2
10	4.50	-103	-4	9	125	903.9	1

Elaboración propia

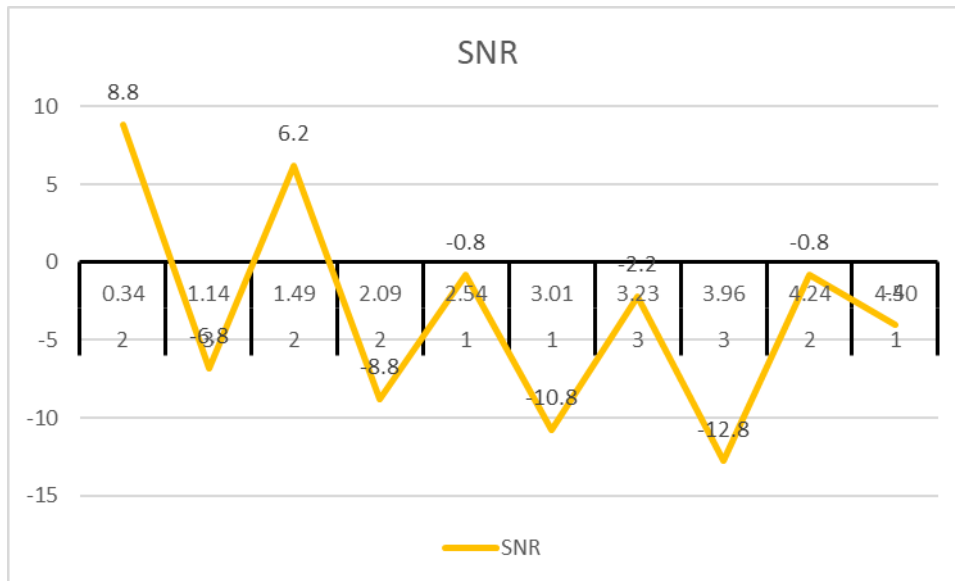
De la tabla anterior se obtiene la figura 48, donde se observa la relación que existe entre la distancia del End Node hacia el Gateway y el estado de línea de vista con respecto a RSSI. La indicación de la intensidad de la señal recibida (RSSI) es la potencia de la señal recibida, en este caso es medida en dBm. Este valor indica de lo bien que el End Node puede recibir una señal del Gateway. De la figura 48, se observa que cuanto más alejado este el End Node, disminuirá el RSSI, a la vez esto lleva relación con el estado de la línea de vista, ya que éste influye directamente con el resultado del RSSI. Por lo tanto, cuánto más despejado (bueno) esté la línea de vista, el alcance será a más distancia.



**Figura 48.** Diagrama de RSSI.

Elaboración propia

De igual manera que la figura anterior, se obtiene la relación de SNR respecto a la distancia y el estado de línea de vista. Normalmente, el nivel de señal de ruido es el límite físico de la sensibilidad, sin embargo, LoRa trabaja por debajo del nivel de ruido. Los valores típicos de SNR de LoRa, están entre: -20dB y +10dB. Un valor más cercano a +10dB significa que la señal recibida está menos corrompida. Y puede demodular señales que están entre -7,5 dB a -20 dB, por debajo del nivel de ruido. En la figura 49, el resultado más alto es de 8.8 dB y el más mínimo es de -12.8 dB. Al igual que en RSSI, SNR lleva relación con la distancia entre el End Node y Gateway y con el estado de línea de vista, esto indica que, a más distancia, la señal recibida estará por debajo de la señal de ruido, esto es más probable si el estado de línea de vista no es bueno.



**Figura 49.** Diagrama de SNR.

Elaboración propia

En las 10 pruebas realizadas, el End Node pudo enviar la información de la localización sin problemas, con datos entre 53 a 57 Bytes. Esta cantidad de Bytes, determina el factor de propagación (SF), cuanto más Bytes se envíen menos será el factor de propagación. Sin embargo, a diferencia del SF el ancho de banda es definitiva y no cambia durante la comunicación, para un enlace ascendente (del End Node hacia el Gateway) el ancho de banda es de 125 KHz. Por último, como se indica en la teoría, LoRa toma frecuencias de manera pseudoaleatoria para realizar la transmisión, por tal motivo en los resultados se observa 6 frecuencias utilizadas.



## V. CONCLUSIONES

El sistema de localización implementado para el monitoreo de vicuñas, es mediante un End Node que es portátil, pequeña y de larga duración, y otorgando un promedio de precisión de 9.11 metros, el cual es adecuado para determinar la ubicación del camélido, de igual manera la localización que es en tiempo real y de manera remota; por lo tanto, no incomoda el hábitat de la Vicuña.

Se obtuvo una cobertura continua de más de 3 kilómetros a la redonda, sin embargo, se obtiene un alcance de hasta 6 kilómetros en dirección al oeste, esto debido a las condiciones geográficas que nos proporcionan buena línea de vista, siendo éste apropiado para ser aplicado en la comunidad Primer Chimpa Jilahuata, pudiendo éste incluso cubrir otras comunidades aledañas como se observa en la Figura 46.

Se hicieron pruebas en 10 puntos diversos, siendo el más lejano a una distancia de 4.5 kilómetros del Gateway obteniendo un RSSI de -103dbm y SNR de - 4db obteniendo conectividad exitosa y con los parámetros dentro del rango establecido por el protocolo LoRaWAN, tal como se muestra en la Tabla 6 y Tabla 7.

Se tiene resultados óptimos en cuanto a cobertura, alcance, precisión, etc., los cuales determinan la viabilidad de la implementación del sistema de comunicación LoRaWAN para la localización de vicuñas en la comunidad Primer Chimpa Jilahuata – Azángaro – Azángaro – Puno.



## VI. RECOMENDACIONES

Si bien, el sistema ya cuenta con la localización de las vicuñas, se puede integrar más sensores al End Node, tales como el sensor de temperatura, humedad y pulso cardíaco; de tal manera poder tener un monitoreo más exhaustivo de estos camélidos. También puede agregarse End Node adicionales, que monitoreen el entorno del hábitat de la vicuña, como el PH de la laguna Quequerana, el clima alrededor de su hábitat (lluvias, nubosidad y vientos) y sensores de alerta (en caso de incendios). Estos puntos deben ser desarrollados de manera conjunta o tomando como antecedente esta investigación, con tesis de pregrado o postgrado de carreras relacionadas con la Medicina Veterinaria.

Para obtener un rango más amplio de la comunicación LoRaWAN, se puede adicionar otros Gateway, con el cual se formaría una red similar a la telefonía celular; y de esta manera conectar más End Node de diversos tipos y usos.

Para tener un seguimiento de localización más preciso, puede aplicarse la técnica de multilateración, que utiliza el tiempo de transmisión de los datos desde el Gateway y el End Node, y así lograr una triangulación del End Node, esto se lograría gracias a tres o más Gateway implementados.



## VII. REFERENCIAS

Acevedo, F., Coduri, G., & Perera, G. (2018). *Geolocalización con LoRa mediante multilateración* (Universidad de la República de Uruguay). Retrieved from <https://www.colibri.udelar.edu.uy/jspui/handle/20.500.12008/20293>

Barro, P. A., Zennaro, M., & Pietrosevoli, E. (2019). TLTN - The local things network: On the design of a LoRaWAN gateway with autonomous servers for disconnected communities. *IFIP Wireless Days, 2019-April*, 1–4. <https://doi.org/10.1109/WD.2019.8734239>

Basantes, J. (2016). Analisis de factibilidad tecnica y de viabilidad comercial de dispositivos para localizacion de mascotas caninas mediante el uso de tecnologia gps en distrito metropolitano de Quito. (Pontifica Universidad Católica del Ecuador). Retrieved from [http://www.puce.edu.ec/publicaciones/Centro\\_de\\_Publicaciones/Revistas/Publicaciones/Revista\\_70.pdf#page=115](http://www.puce.edu.ec/publicaciones/Centro_de_Publicaciones/Revistas/Publicaciones/Revista_70.pdf#page=115)

Centenaro, M., Vangelista, L., Zanella, A., & Zorzi, M. (2016). Long-range communications in unlicensed bands: the rising stars in the IoT and smart city scenarios. *IEEE Wireless Communications*, 23(5), 60–67. <https://doi.org/10.1109/MWC.2016.7721743>

Chaudhari, B. S., & Zennaro, M. (2020). Introduction to low-power wide-area networks. In *LPWAN Technologies for IoT and M2M Applications*. <https://doi.org/10.1016/b978-0-12-818880-4.00001-6>

Croce, D., Gucciardo, M., Santaromita, G., Mangione, S., & Tinnirello, I. (2020). Performance of LoRa technology: link-level and cell-level performance. In *LPWAN Technologies for IoT and M2M Applications*. <https://doi.org/10.1016/b978-0-12->



818880-4.00010-7

Lie, R. (2018). OTAA, ABP & LoRaWAN Security. Retrieved from Mobilefish.com website:  
[https://www.mobilefish.com/developer/lorawan/lorawan\\_quickguide\\_tutorial.html](https://www.mobilefish.com/developer/lorawan/lorawan_quickguide_tutorial.html)

Lin, S., Ying, Z., & Zheng, K. (2019). Design and Implementation of Location and Activity Monitoring System Based on LoRa. *Advanced Materials Research*, 1–15. Retrieved from <http://arxiv.org/abs/1902.01947>

Liy, Y., He, Z., Li, Y., Xu, H., Pei, L., & Zhang, Y. (2018). Towards location enhanced IoT: Characterization of LoRa signal for wide area localization. *Proceedings of 5th IEEE Conference on Ubiquitous Positioning, Indoor Navigation and Location-Based Services, UPINLBS 2018*.  
<https://doi.org/10.1109/UPINLBS.2018.8559844>

LoRa Alliance™. (2018). *LoRaWAN™ 1.0.3 Specification*. Retrieved from <https://lora-alliance.org/resource-hub/lorawanr-specification-v103>

LoRa™ Alliance. (2016). LoRaWAN™ Specification v1.0.2. *LoRaWAN™ 1.0 Specification, 1.0.2*, 1–91. Retrieved from [https://lora-alliance.org/sites/default/files/2018-05/lorawan1\\_0\\_2-20161012\\_1398\\_1.pdf](https://lora-alliance.org/sites/default/files/2018-05/lorawan1_0_2-20161012_1398_1.pdf)

Mekki, K., Bajic, E., Chaxel, F., & Meyer, F. (2019). A comparative study of LPWAN technologies for large-scale IoT deployment. *ICT Express*, 5(1), 1–7.  
<https://doi.org/10.1016/j.ict.2017.12.005>

Muteba, F., Djouani, K., & Olwal, T. (2019). A comparative Survey Study on LPWA IoT Technologies: Design, considerations, challenges and solutions. *Procedia Computer Science*, 155, 636–641.





<https://doi.org/10.1016/j.procs.2019.08.090>

Pham, C., Bounceur, A., Clavier, L., Noreen, U., & Ehsan, M. (2020). Radio channel access challenges in LoRa low-power wide-area networks. In *LPWAN Technologies for IoT and M2M Applications*. <https://doi.org/10.1016/b978-0-12-818880-4.00004-1>

Raspberry Pi Foundation. (2016). Raspberry Pi 3 Model B+ Datasheet. In *Datasheet*. Retrieved from <https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-Model-Bplus-Product-Brief.pdf>

Raza, U., Kulkarni, P., & Sooriyabandara, M. (2017). Low Power Wide Area Networks: An Overview. *IEEE Communications Surveys and Tutorials*, 19(2), 855–873. <https://doi.org/10.1109/COMST.2017.2652320>

SHENZHEN RAKWIRELESS TECHNOLOGY. (2017a). *RAK815 (RAK813 BreakBoard)*.

SHENZHEN RAKWIRELESS TECHNOLOGY. (2017b). *RAK831 Lora Gateway*. Retrieved from [www.rakwireless.com](http://www.rakwireless.com)

Song, Y., Lin, J., Tang, M., & Dong, S. (2017). An Internet of Energy Things Based on Wireless LPWAN. *Engineering*, 3(4), 460–466. <https://doi.org/10.1016/J.ENG.2017.04.011>

Sosa, E., Godoy, D., Lilli, R., Benitez, J. de D., Belloni, E., & Barreiro, H. (2015). Localización Geográfica de Ganado Utilizando Modelos de Propagación de Señal y XBee. *XVII Workshop de Investigadores En Ciencias de La Computación*, 5. Retrieved from [http://sedici.unlp.edu.ar/bitstream/handle/10915/45294/Documento\\_completo.pdf?](http://sedici.unlp.edu.ar/bitstream/handle/10915/45294/Documento_completo.pdf?)



sequence=1&isAllowed=y

Tanaka, M. S., Miyanishi, Y., Toyota, M., Murakami, T., Hirazakura, R., & Itou, T. (2017). A study of bus location system using LoRa: Bus location system for community bus “notty.” *2017 IEEE 6th Global Conference on Consumer Electronics, GCCE 2017, 2017-Janua(Gcce)*, 1–4. <https://doi.org/10.1109/GCCE.2017.8229279>

Telensa. (2020). UNB Pioneer Telensa joins Weightless SIG board; Weightless-N moves to ETSI LTN initiative. Retrieved from <https://www.telensa.com/news/weightless-and-etsi-partner-on-lpwan-iot-standards-development/>

The Things Network. (2020). Network Architecture.

Xie, Z., Xu, R., & Lei, L. (2014). A study of clear channel assessment performance for low powerwide area networks. *IET Seminar Digest, 2014(3)*, 311–315. <https://doi.org/10.1049/ic.2014.0119>

Yegin, A., Kramp, T., Dufour, P., Gupta, R., Soss, R., Hersent, O., ... Sornin, N. (2020). LoRaWAN protocol: specifications, security, and capabilities. In *LPWAN Technologies for IoT and M2M Applications*. <https://doi.org/10.1016/b978-0-12-818880-4.00003-x>

Yuan, P., Wen, X., Lu, Z., & Pan, Q. (2018). Dynamic backoff based access mechanism for LoRaWAN class A. *Proceedings - 2nd IEEE International Conference on Energy Internet, ICEI 2018*, 219–223. <https://doi.org/10.1109/ICEI.2018.00047>

Zhou, Q., Xing, J., Hou, L., Xu, R., & Zheng, K. (2019). A Novel Rate and



Channel Control Scheme Based on Data Extraction Rate for LoRa Networks. *IEEE*

*Wireless Communications and Networking Conference, WCNC, 2019-April, 1–6.*

<https://doi.org/10.1109/WCNC.2019.8885860>

## ANEXOS

### Anexo 1: Línea de comandos para la interpretación de los datos hexadecimales.

```
function Decoder(bytes, port) {
  var decoded = {};
  var hexString=bin2HexStr(bytes);
  return rakSensorDataDecode(hexString);
}

function bin2HexStr(bytesArr) {
  var str = "";
  for(var i=0; i<bytesArr.length; i++) {
    var tmp = (bytesArr[i] & 0xff).toString(16);
    if(tmp.length == 1) {
      tmp = "0" + tmp;
    }
    str += tmp;
  }
  return str;
}

function parseShort(str, base) {
  var n = parseInt(str, base);
  return (n << 16) >> 16;
}

function parseTriple(str, base) {
  var n = parseInt(str, base);
  return (n << 8) >> 8;
}

function rakSensorDataDecode(hexStr) {
  var str = hexStr;
  var myObj = {};

  while (str.length > 4) {
    var flag = parseInt(str.substring(0, 4), 16);
    switch (flag) {

      case 0x0188:// GPS
        myObj.latitude = parseFloat((parseTriple(str.substring(4, 10), 16) *
0.0001).toFixed(4)) + "°";//unit:°
        myObj.longitude = parseFloat((parseTriple(str.substring(10, 16), 16) *
0.0001).toFixed(4)) + "°";//unit:°
        myObj.altitude = parseFloat((parseTriple(str.substring(16, 22), 16) *
0.01).toFixed(1)) + "m";//unit:m
        str = str.substring(22);
        break;

      case 0x0371:// Triaxial acceleration
        myObj.acceleration_x = parseFloat((parseShort(str.substring(4, 8), 16) *
0.001).toFixed(3)) + "g";//unit:g
```



```
        myObj.acceleration_y = parseFloat((parseShort(str.substring(8, 12), 16) *  
0.001).toFixed(3)) + "g");//unit:g  
        myObj.acceleration_z = parseFloat((parseShort(str.substring(12, 16), 16) *  
0.001).toFixed(3)) + "g");//unit:g  
        str = str.substring(16);  
        break;  
  
    case 0x0802:// Battery Voltage  
        myObj.battery = parseFloat((parseShort(str.substring(4, 8), 16) *  
0.01).toFixed(2)) + "V");//unit:V  
        str = str.substring(8);  
        break;  
    default:  
        str = str.substring(7);  
        break;  
    }  
    }  
    return myObj;  
}
```



## Anexo 2: Forma de cálculo de tiempo de transmisión.

El tiempo en el aire (ToA) es la cantidad de tiempo que la antena transmisora está energizada y transmitiendo datos.

- Esto es calculado de la siguiente manera:

$$\mathbf{ToA = Tpaquete = Tpreámbulo + Tpayload}$$

Duración del preámbulo (Tpreámbulo) en segundos,

Duración del Payload (Tpayload) en segundos.

- La duración del preámbulo se calcula de la siguiente manera:

$$\mathbf{Tpreámbulo = (Npreámbulo + 4.25) Ts}$$

El número de preámbulo, por ejemplo, para EU868:  $Npreámbulo = 8$ .

La duración del símbolo (Ts) es en segundos.

- La duración del símbolo es calculada de la siguiente manera:

$$\mathbf{Ts(s) = 2^{SF}/BW}$$

El Ancho de Banda (BW) es en Hz.

El valor del factor de propagación (SF) varía entre 7-12.

- La duración del payload es calculado de la siguiente forma:

$$\mathbf{Tpayload = Ts(8 + \max(\text{ceil}((8PL - 4SF + 28 + 16CRC - 20H)/4(SF - 2DE)), (CR + 4), 0))}$$

Duración del símbolo (Ts) en segundos

Payload (PL) en bytes



Factor de dispersión (SF=7-12)

CRC (activado=1, desactivado=0. Para LoRaWAN por defecto CRC=1)

Cabecera (desactivada H=1 [implícita], activada H=0 [explícita]) Para LoRaWAN

la cabecera está activada: H=0

LowDataRateOptimize (activado DE=1, desactivado DE=0)

Tasa de codificación (CR=1, 2,3 o 4. Para LoRaWAN por defecto CR=1)



### Anexo 3: Mapa de cobertura de comunicación LoRa

