



UNIVERSIDAD DE CHILE  
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS  
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

## COMPARACIÓN Y VALIDACIÓN EXPERIMENTAL DE LA GAMA LORA EDGE SOBRE REDES LORAWAN

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL ELÉCTRICO

IGNACIO PEÑA ESPINOZA

PROFESORA GUÍA:  
Sandra Céspedes Umaña

MIEMBROS DE LA COMISIÓN:  
José González García  
Francisco Rivera Serrano

Este trabajo ha sido parcialmente financiado por:  
Proyecto Fondecyt 1201893 y el AC3E Proyecto ANID Basal FB0008

SANTIAGO DE CHILE  
2022

RESUMEN DE LA MEMORIA PARA OPTAR  
AL TÍTULO DE INGENIERO CIVIL ELÉCTRICO  
POR: IGNACIO PEÑA ESPINOZA  
FECHA: 2022  
PROF. GUÍA: SANDRA CÉSPEDES U.

## **COMPARACIÓN Y VALIDACIÓN EXPERIMENTAL DE LA GAMA LORA EDGE SOBRE REDES LORAWAN**

El presente documento presenta una comparación, cuantitativa, entre la última versión de la tecnología LoRa, de Semtech, llamada LoRa Edge, y su versión previa. La tecnología LoRa es apta para ser utilizada en las llamadas redes de área amplia y de bajo consumo energético, que es un tipo de red muy común para el despliegue de proyectos basados en el Internet de las Cosas. La gama LoRa Edge, bajo estudio en esta memoria, ofrece servicios de localización, tanto en interiores como en exteriores, a dispositivos pequeños y restringidos. En esta memoria se realiza la comparación de la precisión del LoRa Edge frente a la precisión de la versión anterior de LoRa, para la cual se activa la localización mediante GPS. El estudio presentado es realizado de forma experimental, con la implementación de una red LoRaWAN de pocos nodos y en la cual se utilizaron transmisores con LoRa y LoRa Edge, usando los hardwares disponibles comercialmente.

Luego de un análisis de resultados, se logró comprobar que el desempeño de la gama LoRa Edge, dentro de los escenarios escogidos, tiene una precisión similar a la que tiene una herramienta de localización que utiliza la búsqueda GPS. Además, presenta la gran ventaja de ser una solución versátil que puede ser utilizada en ambientes exteriores e interiores, mostrando una clara ventaja en ambientes de interior. En ambientes de exterior, el GPS mostró una tener una mejor precisión, pero este detalle se ve compensado por el ahorro energético que promete la gama LoRa Edge.

*A mi Madre y mi Padre,  
mis pilares.*

***Amor eterno***

# Agradecimientos

Agradecimientos especiales a mis padres, mis hermanos, mis amigos y amigas, y a mis familiares que me han acompañado durante todos estos años. Esta demás decir que el amor que les tengo es inmenso y les debo mucho por todo el apoyo y cariño que me han entregado.

Gracias a la profesora Sandra por ayudarme con el tema para la memoria y brindarme la oportunidad de trabajar en este proyecto y guiarme a lo largo de todo el proceso. También debo agradecer a los profesores José y Francisco por sus comentarios y correcciones, que me sirvieron durante el proceso de pruebas y durante la confección de este documento.

A mis personas más cercanas, gracias por estar conmigo en las buenas y en las malas. No sé qué sería de mi sin ustedes.

# Tabla de Contenido

<b>1. Introducción</b>	<b>1</b>
1.1. Motivación y Antecedentes . . . . .	1
1.2. Descripción del problema . . . . .	3
1.3. Objetivos . . . . .	3
1.3.1. Objetivo General . . . . .	3
1.3.2. Objetivos Específicos . . . . .	3
1.4. Estructura de la memoria . . . . .	4
<b>2. Marco Teórico y Estado del Arte</b>	<b>5</b>
2.1. Marco Teórico. . . . .	5
2.1.1. LoRaWAN . . . . .	5
2.1.1.1. Arquitectura y topología . . . . .	5
2.1.1.2. Características del espectro . . . . .	8
2.1.1.3. Long Range - Frequency Hopping Spread Spectrum LR-FHSS . . . . .	8
2.1.2. Visión general del chip LR1110 . . . . .	9
2.1.3. Fórmula del Haversine . . . . .	10
2.2. Estado del Arte. . . . .	10
2.2.1. Trabajos relacionados: LoRaWAN . . . . .	10
2.2.1.1. Trabajos para armar el setup, pruebas y tipo de métricas . . . . .	12
2.2.2. Trabajos con el chip LR1110 . . . . .	12
<b>3. Revisión general de Hardware y la Arquitectura de red</b>	<b>13</b>
3.1. Hardware . . . . .	13
3.1.1. Gateway . . . . .	13
3.1.2. Development Kit chip LR1110 . . . . .	15
3.1.3. Wisblock . . . . .	16
3.2. Arquitectura de la red LoRaWAN . . . . .	17
<b>4. Pruebas de comportamiento</b>	<b>19</b>
4.1. Metodología . . . . .	20
4.1.1. Parámetros de la modulación LoRa para el transmisor . . . . .	20
4.1.2. Experimentos . . . . .	20
4.2. Resultados y análisis . . . . .	25
4.2.1. Experimento 1 . . . . .	25
4.2.2. Resultados generales . . . . .	28
4.2.2.1. Escenario 1 . . . . .	29
4.2.2.2. Escenario 2 . . . . .	31
4.2.2.3. Escenario 3 . . . . .	33

<b>5. Pruebas Comparativas</b>	<b>37</b>
5.1. Metodología . . . . .	37
5.1.1. Experimentos . . . . .	38
5.2. Resultados y análisis . . . . .	42
5.2.1. Experimento 13 . . . . .	43
5.2.2. Resultados generales . . . . .	46
5.2.2.1. Escenario 1 . . . . .	46
5.2.2.2. Escenario 2 . . . . .	48
5.2.2.3. Escenario 3 . . . . .	50
5.2.2.4. Escenario 4 . . . . .	53
5.3. Análisis de consumo . . . . .	55
<b>6. Conclusiones</b>	<b>56</b>
<b>Bibliografía</b>	<b>59</b>
<b>Anexos</b>	
<b>Anexo A. Configuraciones de la red LoRaWAN</b>	<b>62</b>
A.1. Montar las partes del Gateway . . . . .	63
A.2. Configurar el Gateway . . . . .	63
A.3. Registrar y asegurar el Gateway en el <i>Semtech Network Server</i> . . . . .	71
A.4. Armar el transmisor <i>LoRa Edge™ LR1110 Tracking Device</i> . . . . .	75
A.5. Instalar el software del núcleo y del chip LR1110 . . . . .	76
A.6. Reclamar el dispositivo en el portal de <i>LoRa Cloud™</i> . . . . .	79
A.7. Crear un propietario del dispositivo y obtener el Token de acceso . . . . .	82
A.8. Actualizar el almacenamiento de GNSS en el chip LR1110 . . . . .	85
A.9. Agregar el transmisor del <i>Development Kit</i> en el <i>Semtech Network Server</i> y conexión con los servicios de <i>LoRa Cloud™</i> . . . . .	88
A.10. Pruebas de localización del <i>Development Kit</i> . . . . .	94
A.11. Visualización mediante <i>ThingsBoard</i> . . . . .	102

# Índice de Tablas

2.1.	Especificaciones de LoRaWAN para Chile. . . . .	8
4.1.	Parámetros LoRa para pruebas de comportamiento. . . . .	20
4.2.	Caracterización de cada experimento en las pruebas de comportamiento. . . . .	24
4.3.	Tabla de la correlación entre el error (distancia) de posición y la cantidad de redes escaneadas, para todos los experimentos. . . . .	36
5.1.	Caracterización de cada experimento en las pruebas comparativas. . . . .	42

# Índice de Ilustraciones

2.1.	Arquitectura de red LoRaWAN, extraída de [9]	6
2.2.	Tabla de caracterización del Sread Factor de LoRaWAN®, extraída desde [10].	7
2.3.	Tabla de caracterización del Data Rate de LoRa® y la configuraciones para alcanzar dicho DT, extraída desde [10].	7
2.4.	Capas físicas de LoRa, extraída de [9]	8
3.1.	Partes para armar el gateway. Visto desde la izquierda hacia la derecha, las partes más importantes son: RAK 2247, RAK 2247 PI HAT, Raspberry pi 3 B+.	14
3.2.	Partes necesarias para armar el transmisor con la antena activa de GNSS.	15
3.3.	Transmisor y localizador empleado para las comparaciones con el chip LR1110, basado en la herramienta Wisblock.	17
3.4.	Diagrama de los dispositivos que conforman la red LoRaWAN diseñada para el trabajo.	18
4.1.	Diagrama del Setup 1.	21
4.2.	Diagrama del Setup 2.	21
4.3.	Diagrama del Setup 3.	22
4.4.	Diagrama del Setup 4.	23
4.5.	Diagrama del Setup 5.	24
4.6.	Gráfico de la cantidad de redes encontradas por el escaneo de Wi-Fi en el experimento 1.	25
4.7.	Gráfico de las longitudes medidas por el escaneo de Wi-Fi en el experimento 1.	26
4.8.	Gráfico de las latitudes medidas por el escaneo de Wi-Fi en el experimento 1.	26
4.9.	Gráfico del error medido entre el punto real y el punto obtenido por el escaneo de Wi-Fi en el experimento 1.	27
4.10.	Gráfico de los RSSI medidos por el escaneo de Wi-Fi en el experimento 1.	27
4.11.	Gráfico de los SNR medidos por el escaneo de Wi-Fi en el experimento 1.	28
4.12.	Gráfico del error medido entre el punto real y el punto obtenido por el escaneo de Wi-Fi en los experimentos realizados en el escenario 1, mostrando en azul el promedio y rango de la desviación estándar.	29
4.13.	Gráfico de los RSSI medidos por el escaneo de Wi-Fi en los experimentos realizados en el escenario 1.	30
4.14.	Gráfico de los SNR medidos por el escaneo de Wi-Fi en los experimentos realizados en el escenario 1.	30
4.15.	Gráfico del error medido entre el punto obtenido por los experimentos realizados en el escenario 2.	31
4.16.	Gráfico del RSSI promedio obtenido en los experimentos realizados en el escenario 2.	32
4.17.	Gráfico de los SNR promedio obtenido en los experimentos realizados en el escenario 2.	33

4.18.	Gráfico del error medido entre el punto real y el punto obtenido por los experimentos realizados en el escenario 3. . . . .	34
4.19.	Gráfico de los RSSI promedio obtenido en los experimentos realizados en el escenario 3. . . . .	35
4.20.	Gráfico de los SNR promedio obtenido en los realizados en el escenario 3. . . .	35
5.1.	Diagrama del Setup 6. . . . .	38
5.2.	Diagrama del Setup 7. . . . .	39
5.3.	Diagrama del Setup 8. . . . .	39
5.4.	Diagrama del Setup 9. . . . .	40
5.5.	Diagrama del Setup 10. . . . .	41
5.6.	Diagrama del Setup 11. . . . .	41
5.7.	Gráfico de la Longitud encontrada por el chip LR1110 (rojo) con el escaneo de Wi-Fi y para GPS del Wisblock (azul), en las repeticiones del experimento 13.	43
5.8.	Gráfico de la Latitud encontrada por el chip LR1110 (rojo) con el escaneo de Wi-Fi y para GPS del Wisblock (azul), en las repeticiones del experimento 13.	43
5.9.	Gráfico del error calculado entre la posición de referencia y la posición encontrada por el chip LR1110 (naranja) con sus escaneos RF y el error de la posición para GPS del Wisblock (azul), en las repeticiones del experimento 13. . . . .	44
5.10.	Gráfico del RSSI medido para el transmisor del chip LR1110 (rojo) con el escaneo de Wi-Fi y para el transmisor del Wisblock (azul), en las repeticiones del experimento 13. . . . .	45
5.11.	Gráfico del SNR medido para el transmisor del chip LR1110 (rojo) con el escaneo de Wi-Fi y para el transmisor del Wisblock (azul), en las repeticiones del experimento 13. . . . .	45
5.12.	Gráfico del error calculado entre la posición de referencia y la posición encontrada por el chip LR1110 (naranja) con sus escaneos RF y el error de la distancia con la posición para el GPS del Wisblock (azul), en los experimentos para el Escenario 1. . . . .	46
5.13.	Gráfico del RSSI medido para el transmisor del chip LR1110 (naranja) con el escaneo de Wi-Fi y para el transmisor del Wisblock (azul), en los experimentos para el Escenario 1. . . . .	47
5.14.	Gráfico del SNR medido para el transmisor del chip LR1110 (naranja) con el escaneo de Wi-Fi y para el transmisor del Wisblock (azul), en los experimentos para el Escenario 1. . . . .	47
5.15.	Gráfico del error calculado entre la posición de referencia y la posición encontrada por el chip LR1110 (naranja) con sus escaneos RF y el error de la posición para el GPS del Wisblock (azul), en los experimentos para el Escenario 2. . .	48
5.16.	Gráfico del RSSI medido para el transmisor del chip LR1110 (naranja) con el escaneo de Wi-Fi y para el transmisor del Wisblock (azul), en los experimentos para el Escenario 2. . . . .	49
5.17.	Gráfico del SNR medido para el transmisor del chip LR1110 (naranja) con el escaneo de Wi-Fi y para el transmisor del Wisblock (azul), en los experimentos para el Escenario 2. . . . .	50
5.18.	Gráfico del error calculado entre la posición de referencia y la posición encontrada por el chip LR1110 (naranja) con sus escaneos RF y el error de la posición para el GPS del Wisblock (azul), en los experimentos para el Escenario 3. . .	50

5.19.	Gráfico del RSSI medido para el transmisor del chip LR1110 (naranja) con sus escaneos RF y para el transmisor del Wisblock (azul), en los experimentos para el Escenario 3. . . . .	51
5.20.	Gráfico del SNR medido para el transmisor del chip LR1110 (naranja) con sus escaneos RF y para el transmisor del Wisblock (azul), en los experimentos para el Escenario 3. . . . .	52
5.21.	Gráfico del error calculado entre la posición de referencia y la posición encontrada por el chip LR1110 (naranja) con sus escaneos RF y el error de la posición para el GPS del Wisblock (azul), en los experimentos para el Escenario 4. . .	53
5.22.	Gráfico del RSSI medido para el transmisor del chip LR1110 (naranja) con sus escaneos RF y para el transmisor del Wisblock (azul), en los experimentos para el Escenario 4. . . . .	54
5.23.	Gráfico del SNR medido para el transmisor del chip LR1110 (naranja) con sus escaneos RF y para el transmisor del Wisblock (azul), en los experimentos para el Escenario 4. . . . .	54
A.1.	Gateway montado. . . . .	63
A.2.	Partes para utilizar la micro SD. . . . .	64
A.3.	Flasheo de la imagen para la Raspberry en la tarjeta de 16GB, mediante balenaEtcher. . . . .	64
A.4.	SD insertada en la parte posterior de la Raspberry. . . . .	65
A.5.	Ingresar al gateway utilizando el AP de este mismo y por medio de SSH. . . .	66
A.6.	Configuración del Wi-Fi a utilizar para conexión a internet. . . . .	66
A.7.	Ingresar al gateway utilizando el AP de una red con conexión a internet y por medio de SSH. . . . .	67
A.8.	Configuraciones de concentrador a utilizar. . . . .	68
A.9.	Ingresando a las configuraciones del concentrador en ChirpStack. . . . .	68
A.10.	Cambio del pin de reseteo en el archivo <i>global.toml</i> . . . . .	69
A.11.	ID del gateway desplegado en el menú de opciones. . . . .	70
A.12.	Configuraciones iniciales realizadas correctamente. . . . .	71
A.13.	Agregar un nuevo Gateway en el <i>Network Server</i> de Semtech. . . . .	72
A.14.	Mapa para localizar el gateway en el <i>Network Server</i> de Semtech y finalmente registrar el gateway. . . . .	72
A.15.	Pestaña de certificados para el gateway y la opción de generarlos. . . . .	73
A.16.	Opción para ingresar en las configuraciones del puente con el <i>Network Server</i> , para el gateway. . . . .	73
A.17.	Opción para configurar el enlace del MQTT y editar los certificados necesarios. . .	74
A.18.	Configuración de <i>CA certificate</i> . . . . .	74
A.19.	Configuración de <i>TLS certificate</i> . . . . .	74
A.20.	Configuración de <i>TLS KEY</i> . . . . .	75
A.21.	Pestaña de gateway donde se muestra el gateway previamente registrado. . . .	75
A.22.	Transmisor armado, pero, sin el cable USB. . . . .	76
A.23.	Copia del <i>updater tool</i> . . . . .	77
A.24.	Estados del transmisor al actualizarlo con el <i>updater tool</i> . . . . .	77
A.25.	Estados del transmisor al actualizarlo con el <i>updater tool</i> . . . . .	78
A.26.	Pestaña de Lora Cloud Device Join, desde el portal <i>LoRa Cloud™</i> . . . . .	79
A.27.	Pestaña de <i>application owners</i> y donde crear un nuevo perfil . . . . .	80
A.28.	Asignación de <i>free allowance</i> para el perfil creado. . . . .	80

A.29.	Agregando un dispositivo con la opción <i>claim devices</i> , dentro de la pestaña de <i>devices</i> . . . . .	81
A.30.	Introduciendo las <b>fetch keys</b> y agregando finalmente el dispositivo . . . . .	81
A.31.	Dispositivo agregado exitosamente. . . . .	82
A.32.	Pestaña LoRa Cloud Device & Application Services, desde el portal de <i>LoRa Cloud™</i> . . . . .	83
A.33.	Sección de <i>device owners</i> y opción donde crear el perfil nuevo. . . . .	83
A.34.	Campos a rellenar y la opción <i>submit</i> para crear el perfil. . . . .	84
A.35.	Asignación del <i>free access</i> para el perfil creado. . . . .	84
A.36.	Copia del <i>API Access Token</i> que corresponde al dispositivo agregado. . . . .	85
A.37.	Verificación de las versiones instaladas de <i>python</i> y <i>pip</i> . . . . .	86
A.38.	Instalación del <i>AllmanacUpdate</i> exitosa. . . . .	87
A.39.	Actualización exitosa. . . . .	88
A.40.	Pestaña de <i>device-profiles</i> y la opción para crear un nuevo perfil. . . . .	89
A.41.	Configuración del <i>join</i> para el perfil que se estaba creando y la creación de este. . . . .	89
A.42.	Creación exitosa del perfil de para el dispositivo. . . . .	90
A.43.	Pestaña de <i>applications</i> y opción para crear una nueva aplicación. . . . .	90
A.44.	Campos a rellenar para crear un aplicación. . . . .	91
A.45.	<i>Application</i> creada con éxito. . . . .	91
A.46.	Pestaña de <i>devices</i> , dentro de la aplicación creada anteriormente. . . . .	92
A.47.	Campos a rellenar para agregar un nuevo dispositivo en la aplicación. . . . .	92
A.48.	Dispositivo agregado exitosamente. . . . .	93
A.49.	Pestaña de <i>integrations</i> , dentro de la aplicación creada anteriormente, y la opción para integrar <i>LoRa Cloud™</i> . . . . .	93
A.50.	Campos a rellenar para la integración de <i>LoRa Cloud™</i> y opción para agregar la integración. . . . .	94
A.51.	Permiso necesario para poder utilizar los servicios de LoRa Cloud™. . . . .	95
A.52.	Pestaña de <b>DEVICE DATA</b> , dentro del dispositivo enlazado previamente. . . . .	95
A.53.	Opción <b>CONNECTIVITY</b> dentro del transmisor del Development Kit . . . . .	96
A.54.	Conectando el transmisor a la red LoRaWAN. . . . .	96
A.55.	Transmisor conectado a la red LoRaWAN. . . . .	97
A.56.	Solicitud de unirse a la red LoRaWAN, hecha por el transmisor. . . . .	97
A.57.	Seleccionando el modo de escaneo de Wi-Fi dentro del transmisor del Development Kit. . . . .	98
A.58.	Escaneo de redes Wi-Fi y envío de resultados a la red LoRaWAN. . . . .	98
A.59.	Paquete recibido en el <i>network server</i> , con la posición obtenida por el escaneo de Wi-Fi. . . . .	99
A.60.	Selección de escaneo de GNSS asistido. . . . .	100
A.61.	“Posición” por defecto del transmisor, dentro de las opciones del escaneo asistido de GNSS. . . . .	100
A.62.	Configuración el escaneo del GNSS asistido para realzar el mejor esfuerzo en vez de ahorrar energía. . . . .	101
A.63.	Realización del escaneo asistido de GNSS y envío por medio de la red LoRaWAN. . . . .	101
A.64.	Paquete recibido en el <i>network server</i> , con la posición obtenida por el escaneo asistido de GNSS. . . . .	102
A.65.	Opción de agregar un nuevo dispositivo dentro de la pestaña de Dispositivos en la plataforma de <i>ThingsBoard Demo</i> . . . . .	103

A.66.	Datos necesarios para agregar un nuevo dispositivo en la plataforma de <i>ThingsBoard Demo</i> . . . . .	103
A.67.	Gestión de las credenciales del dispositivo agregado a la plataforma de <i>ThingsBoard Demo</i> . . . . .	104
A.68.	Agregando el <b>Access token</b> de <i>ThingsBoard Demo</i> en el <i>network server</i> de Semtech, dentro de la aplicación creada previamente. . . . .	104
A.69.	Seleccionando la opción para integrar <i>ThingsBoard</i> en la aplicación del <i>network server</i> de Semtech. . . . .	105
A.70.	Completando los campos necesarios para integrar <i>ThingsBoard</i> en la aplicación del <i>network server</i> de Semtech. . . . .	105
A.71.	Panel creado en <i>ThingsBoard Demo</i> para poder visualizar en tiempo real las variables previamente seleccionadas. . . . .	106

# Capítulo 1

## Introducción

### 1.1. Motivación y Antecedentes

Uno de los objetivos de la Industria 4.0 es realizar cambios en los procesos y operaciones dentro de la industria, con el objetivo de incorporar técnicas y herramientas avanzadas, las cuales se enfoquen en mejorar la producción mediante el uso de las llamadas tecnologías inteligentes. Algunas de las nuevas tecnologías que se busca utilizar son: nanotecnología, avances en robótica, Internet de las cosas (IoT), inteligencia artificial, Big Data, entre otras.

La incorporación de dispositivos IoT a escalas industriales ha provocado que las áreas de investigación, las cuales se enfocan en el uso de redes inalámbricas de bajo consumo energético, hayan experimentado un gran crecimiento durante el último tiempo. La principal causa del interés que se ha generado sobre estas redes es debido a que, para muchos tipos de aplicaciones de IoT, el despliegue de este tipo de redes mejora el desempeño de los dispositivos inalámbricos que se utilicen y, también, mejora la calidad de la comunicación en la red misma, logrando diseñar sistemas que pueden asegurar una gran duración en las baterías y una comunicación que sea bidireccional, confiable e inalámbrica.

Las habilidades y conocimientos que posee un profesional que se desempeñe en el área de las Telecomunicaciones, como puede ser el caso de un Ingeniero Civil Eléctrico, o Ingeniera Civil Eléctrica, son cualidades perfectas para estar dentro de los cambios que está impulsando la llamada *cuarta revolución industrial*, permitiéndole realizar proyectos que ayuden a actualizar los procesos que ya existen o, incluso, realizar las investigaciones y pruebas para descubrir nuevos modelos que permitan avances en las tecnologías emergentes.

Dentro de las redes de bajo consumo, existen aquellas que se adecuan para ser capaces de entregar una cobertura amplia para la conectividad de dispositivos, son llamadas redes de área amplia y de bajo consumo energético, o *Low Power Wide Area Network (LPWAN)*. Una red que cumple con las características de las LPWAN y, que también es de las que presentan mejor desempeño en una gran cantidad de escenarios, como se menciona en [1], son las redes LoRaWAN, siendo estas las que se caracterizan por utilizar la modulación LoRa, la cual fue desarrollada por *Semtech*. En la actualidad, las redes LoRaWAN<sup>1</sup> tienen un buen desempeño y se caracterizan por ser redes que operan dentro de la banda para uso industrial, científico y médico (ISM)<sup>2</sup>, lo que permite una transparencia en cuanto al acceso que tiene

<sup>1</sup> Es una especificación para las redes de baja potencia y área amplia (LPWAN), en donde, principalmente, se utilizan dispositivos LoRa.

<sup>2</sup> ISM: industrial, scientific and medical

esta tecnología. Los estudios que se han realizado permiten crear documentos que expliquen las características principales y los alcances que puede llegar a tener las redes LoRaWAN, siendo [1] un buen ejemplo, lo que permite a los siguientes trabajos estar enfocados en realizar mejoras dentro de este tipo de redes. Las mejoras pueden tomar un sin fin de caminos distintos, generando una búsqueda por especializar las redes para casos puntuales, como se muestra en [2] en donde la mejora apunta a adecuar una red LoRa para un uso estudiantil o, como se puede observar en [3] donde se busca mejorar el desempeño del protocolo Aloha, en LoRa, mediante algoritmos especializados.

El uso de la modulación LoRa en, redes LoRaWAN, permite que, en una misma red, se conecten muchos dispositivos inalámbricos, teniendo un buen desempeño hasta con 100 dispositivos interconectados, como se menciona en [4], sin mencionar la buena cobertura que se alcanza al permitir transmisiones de kilómetros de distancia. Sin embargo, al buscar tener procesos cada vez más inteligentes e independiente, es necesario poder integrar una cantidad muy alta de dispositivos y que deben, además, estar dentro de la misma red para que estos se puedan comunicar entre ellos y se logre tener una visión informada de todo lo que sucede dentro del proceso que se está asistiendo. Debido a esta necesidad, es que las mejoras que se logran al agregar algún software o hardware externo a las redes LoRaWAN, permiten avanzar en la dirección para ampliar la escalabilidad de estas redes. Algunos de los desafíos que se deben superar son problemas de interferencia y baja inmunidad al ruido que se presenta en ambientes poco favorables o con muchos obstáculos, lo cual depende del *Spread Factor* que se esté utilizando y la cantidad de dispositivos en este [5]. Con lo anterior en mente es que la *LoRa Alliance* se ha dedicado a ir realizando mejoras a las redes LoRaWAN y, de esta forma, ir expandiendo los usos que se les puede dar, además de mejorar enormemente el desempeño que ya se tiene.

Una de las ultimas actualizaciones que realizó *Semtech* se presenta en [6], donde se muestra la nueva gama de dispositivos llamada *LoRa Edge™*, la cual esta basada en el nuevo chip *LR1110*. Como se menciona en [6], *LoRa Edge™* es una plataforma con un consumo energético ultra bajo y que integra transmisores *LoRa®* a una multi constelación de escaneo, permitiendo el escaneo pasivo de direcciones Wi-Fi AP MAC<sup>3</sup>, y el escaneo pasivo/activo del *Sistema Global de Navegación Satelital* (GNSS<sup>4</sup>, por su sigla en ingles). Dentro de los beneficios claves, tal como se describe en [6], se puede mencionar que: provee una solución que permite balancear el buen desempeño de la localización y un bajo consumo energético, siendo también una solución con la que se puede reducir los costos de un proyecto, al tener un chip que logra integrar GNSS y Wi-Fi en un solo dispositivo y que puede ser utilizado tanto en interiores como en exteriores, y, permite al usuario utilizar soluciones que utilicen 10 veces menos energía que los dispositivos existentes.

Al ser el chip LR1110 una tecnología tan reciente, introducido el año 2020, es normal que solo existan unos pocos estudios al respecto, los cuales se enfocan en realizar pruebas de desempeño, alcances, escalabilidad. Sin embargo, el interés por esta nueva modulación es alto, ya que, como se expone en [6], se espera poder proporcionar una herramienta que utilice muy poca energía y que se pueda aplicar en escenarios de interiores y exteriores. Además, está modulación cuenta con la cualidad de poder ser utilizada para el *tracking*, lo que permite

<sup>3</sup> Punto de acceso para redes Wi-Fi, definido por su sigla: Media Access Control (MAC)

<sup>4</sup> Global Navigation Satellite System

tener otra solución aparte del GPS. Debido a esto, se ha generado interés sobre la gama LoRa Edge™, como es el caso de distintas compañías, como *Skyhook*, que buscan integrar servicios de tracking. También se espera que el chip LR1110 aumente la versatilidad y alcances de una de las redes LPWAN que ya presenta uno de los mejores desempeños que se pueden encontrar en el mercado, tal como se expone en [7].

## 1.2. Descripción del problema

En la actualidad, las investigaciones realizadas con el chip LR1110 son pocas y solo se enfocan en un análisis de algunas de las características del chip LR1110 en despliegues para IoT [8]. Sin embargo, el escenario que se evalúa en [8] es un caso general para redes LoRaWAN y, específicamente, es un escenario de exteriores, en donde se verifican las cualidades que presenta el LR1110 frente a un caso genérico, y en el cual un dispositivo basado en *on-board RTC*, como es el RAK4630, ya tiene buenos resultados. La idea principal es comparar los desempeños de ambas herramientas, al realizar una búsqueda de la ubicación y, posteriormente, ser evaluados frente a una de las métricas más importantes para las redes LPWAN: el uso energético de los *end-devices*.

El problema que se aborda en esta memoria es que no hay una cantidad suficiente de antecedentes experimentales en un escenario de aplicación para las redes LoRaWAN y que utilice el chip LR1110, lo que genera una incertidumbre de como trabaja en realidad la nueva tecnología. Bajo este antecedente, no se puede tener una idea clara de cuáles son las mejoras o diferencias que presenta la gama LoRa Edge™ frente a otra versión de chips para transmisores en redes LoRaWAN, como lo es el chip SX1262 integrado en el RAK4630, y que se puede utilizar junto con el módulo GPS RAK1910. Debido al interés que existe en el uso de IoT como, por ejemplo, en el tracking *indoor* u *outdoor*, es necesario poder resolver las dudas frente al desempeño de las herramientas de la gama LoRa Edge™ y que también se logre analizar su desempeño frente a las métricas importantes para las redes LoRaWAN.

## 1.3. Objetivos

### 1.3.1. Objetivo General

Evaluar experimentalmente la funcionalidad de localización presente en la gama LoRa Edge™, frente a un escenario genérico que utilice versiones previas de la tecnología LoRa, más la localización mediante GPS, para establecer, cuantitativamente, las mejoras que presenta la nueva gama de módulos LoRa en despliegues de redes bajo el estándar LoRaWAN y, finalmente, evaluar comparativamente el desempeño de ambas tecnologías.

### 1.3.2. Objetivos Específicos

Los objetivos específicos que buscan alcanzar con este trabajo son los siguientes:

- Diseñar e implementar un montaje experimental de una red LoRaWAN, la cual sea

simple y de pocos dispositivos, que permita realizar pruebas experimentales, utilizando el chip LR1110, para escenarios de alcance no mayor a  $30[m]$  y con pruebas en escenarios de interior y exteriores.

- Diseñar experimentos para evaluar el desempeño de la localización, dentro de la red LoRaWAN, utilizando la gama LoRa Edge integrada en el chip LR1110 y la versión anterior de LoRa integrada en el chip SX1262 y desplegada en el módulo comercial RAK 4630. La evaluación comparativa se realiza usando métricas estándar para las redes LPWAN.
- Analizar el desempeño presentado por las dos tecnologías, sobre la red LoRaWAN, y comparar las ventajas y/o desventajas de la gama LoRa Edge™.

## 1.4. Estructura de la memoria

Este trabajo inicia con la Introducción pertinente a toda investigación, la cual está presente en el Capítulo 1 de a la memoria. En la Introducción se aborda la Motivación y Avances del tema que se va a tratar, además de mencionar el Problema que se va a intentar atacar con los objetivos del trabajo de título. Luego se presenta el Marco Teórico y Estado del Arte en el Capítulo 2, donde se busca dejar a disposición del lector toda la información necesaria para poder entender el contenido central de esta obra en su totalidad, así como también, se pretende dejar una pequeña recopilación de los trabajos relacionados con el tema y la actualidad de este.

Para comenzar con el trabajo realizado, era necesario llevar acabo las configuraciones de los dispositivos dentro de la red LoRaWAN y la configuraciones de la red misma, lo cual está presente en el Anexo, dejando en esta sección todos los pasos que se siguieron para poder utilizar las herramientas de LoRa Cloud™ y el *Network Server* de Semtech. En adición a lo anterior, en el Capítulo 3, se dejó un apartado que esta enfocado en realizar una Revisión general del Hardware y la Arquitectura de la red utilizada.

A continuación, en el Capítulo 4, se presentan las pruebas realizadas para poder evaluar el comportamiento individual del dispositivo basado en la gama LoRa Edge™, dejando explícita la metodología y los resultados obtenidos, con para, finalmente, realizar el análisis respectivo. Posteriormente, en el Capítulo 5 se deja la metodología y los resultados de las Pruebas Comparativas, donde se evaluaron ambos dispositivos al mismo tiempo, con la finalidad de a comparar y evaluar los resultados. De la misma forma que se realizó anteriormente, el Capítulo 5 finaliza con el análisis de los resultados obtenidos en las pruebas de comparación.

Finalmente, se tiene las Conclusiones, las cuales se dejan en el Capítulo 7 y sirven para dejar explícitamente: los objetivos logrados, los resultados finales más importantes, los aprendizajes obtenidos, y, los aportes del trabajo de memoria.

# Capítulo 2

## Marco Teórico y Estado del Arte

Este capítulo se centra en recopilar toda la información, ecuaciones y definiciones necesarias para poder entender el contenido de este trabajo de memoria. Se ha separado en dos secciones; el marco teórico y el estado del arte, siendo el último la investigación preliminar de la actualidad científica en este tema.

### 2.1. Marco Teórico.

Dentro de este apartado se busca recopilar la información necesaria para que se entienda la relación existente en todos los conceptos, formulas y definiciones, junto con el trabajo realizado en esta memoria .

#### 2.1.1. LoRaWAN

Este tipo de redes se encuentran dentro de las redes de largo alcance y de bajo consumo (LPWAN) y, como se describe en [9], es un protocolo que está diseñado para dispositivos inalámbricos que requieran uso de baterías, siendo útil para aplicaciones de IoT a escala regional, nacional o global. Además, cuenta con las características principales para estas funciones: comunicación bidireccional, seguridad de fin a fin, movilidad y servicios de localización.

##### 2.1.1.1. Arquitectura y topología

La arquitectura se basa en una topología de red tipo *star of stars*, donde los dispositivos que conforman la red se clasifican de 3 tipos: Servidor, Gateway (GW) y End-device (ED).

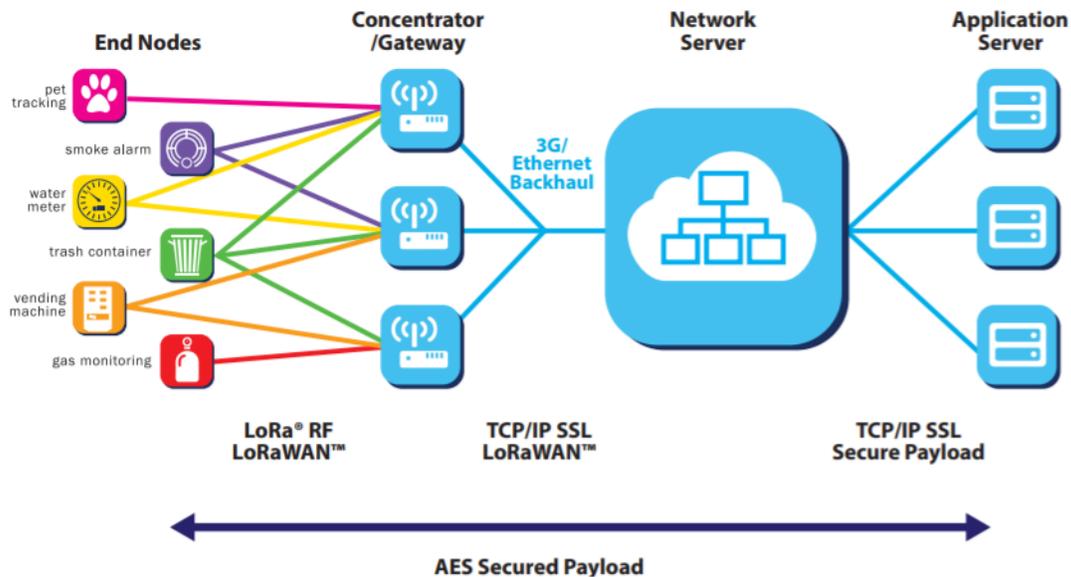


Figura 2.1: Arquitectura de red LoRaWAN, extraída de [9]

La red presentada en la Figura 2.1 funciona básicamente de la siguiente forma:

- Los ED corresponden a los dispositivos que recopilan la información o son los actuadores. Bajo su comunicación inalámbrica está aplicada toda la capa física de la modulación LoRA y estos ED se encargan de enviar la información hacia los módulos y microcontroladores capacitados con comunicación LoRA, lo cuales son los encargados de manejar los paquetes y la información para luego ser entregada al GW si es necesario salir de la red local.
- El GW se utiliza para conectar los mensajes que se envían entre los End-Devices y el centro de la red, hacia el exterior de esta e incorporar el uso de aplicaciones con Internet para el manejo de los datos. Los GW se conectan con el servidor de la red por medio de IP estándar y realizan el trabajo de convertir los paquetes de radio frecuencia a paquetes IP y viceversa.
- El servidor es el encargado de recopilar y almacenar los datos que pretenden ser rescatados. Cabe mencionar, que, dependiendo de las características de la red, se puede hacer una conexión con servidores online y luego llevar la información a plataformas virtuales para su posterior visualización o uso de aplicaciones externas.

Existen distintas clases de ED que se pueden utilizar, los cuales se describen a continuación:

- **Clase A: consumo mínimo, bidireccional**

Con estos dispositivos la comunicación es siempre iniciada por los ED y es asíncrona. Luego de cada transmisión de subida, el dispositivo da un tiempo para las transmisiones de baja o transmisiones de control por parte de la red. Esta clase de dispositivo tiene la capacidad de entrar en un estado *sleep* para el ahorro de energía y el momento en que entra en este estado o la duración de este depende enteramente de cada red. Es la clase que usa menos energía mientras permite las transmisiones de subida en cualquier momento.

- **Clase B: comunicación bidireccional con determinación de la latencia en la subida**

Son similares a los clase A, pero en este caso se tiene que están sincronizados con la red, por lo que se permite la transmisión de baja por medio de tiempos programados. Al usar más energía se permite que las trasmisiones de bajada cuenten con latencia determinada.

- **Clase C: latencia mínima, bidireccional**

A diferencia de las otras dos clases, en este caso se tiene que la transmisión de bajada está habilitada en cualquier momento que el dispositivo este transmitiendo. Con esta característica el consumo de energía sube considerablemente, por lo que estos dispositivos deben ser energizados continuamente, como ser conectados a la red eléctrica.

Existe una característica de las redes LoRaWAN que se conoce como *Data Rate* (DR), la cual sirve para poder ajustar el rango y la duración de los mensajes a enviar. Con esto es posible configurar las comunicaciones con cada ED y de esta forma optimizar la energía y cobertura que se necesita para entregar los paquetes. Existe otro factor llamado *Spread Factor* (SF) el cual sirve para alcanzar un DR en específico, sin embargo no son equivalentes, pero si están relacionados. Para poder dejar más claro estos conceptos, se deja a disposición las tablas presentes en la Figura 2.2 y la Figura 2.3, las cuales se han rescatado de [10].

Spreading Factor (For UL at 125 KHz)	Bit Rate	Range (Depends on Terrain)	Time on Air for an 11-byte payload
SF10	980 bps	8 km	371 ms
SF9	1760 bps	6 km	185 ms
SF8	3125 bps	4 km	103 ms
SF7	5470 bps	2 km	61 ms

Figura 2.2: Tabla de caracterización del Sread Factor de LoRaWAN®, extraída desde [10].

Data Rate (DR)	Spreading Factor (SF)	Channel Frequency	Uplink or Downlink	Bitrate (Bits/Sec)	Maximum User Payload Size (Bytes)
0	SF10	125 kHz	Uplink	980	11
1	SF9	125 kHz	Uplink	1,760	53
2	SF8	125 kHz	Uplink	3,125	125
3	SF7	125 kHz	Uplink	5,470	242
4	SF8	500 kHz	Uplink	12,500	242
5-7					
8	SF12	500 kHz	Downlink	980	53
9	SF11	500 kHz	Downlink	1,760	129
10	SF10	500 kHz	Downlink	3,125	242
11	SF9	500 kHz	Downlink	5,470	242
12	SF8	500 kHz	Downlink	12,500	242
13	SF8	500 kHz	Downlink	21,900	242

Figura 2.3: Tabla de caracterización del Data Rate de LoRa® y la configuraciones para alcanzar dicho DT, extraída desde [10].

### 2.1.1.2. Características del espectro

Las redes LoRaWAN hacen referencia a los protocolos de comunicación y los sistemas para la arquitectura a utilizar. Por otra parte, LoRa hace referencia a la capa física que permite las comunicaciones de largo alcance. De esta forma, se deja la Figura 2.4, en donde se muestran las capas físicas de las redes LoRa.

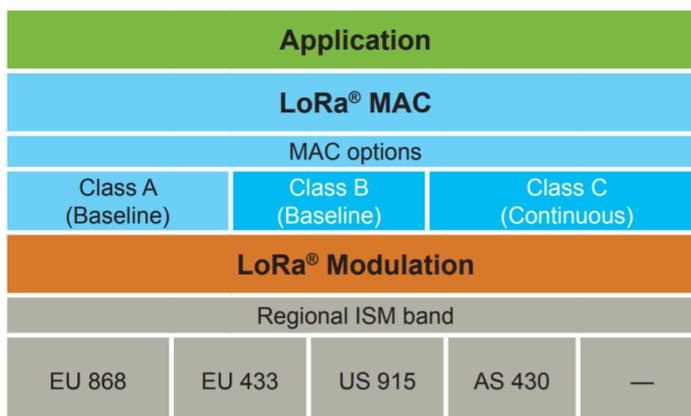


Figura 2.4: Capas físicas de LoRa, extraída de [9]

Como se puede ver en el documento *Informe de la situación actual de los requerimientos de transmisión de datos y la estimación de la demanda prospectiva de consumo de datos para zonas agrícolas. Licitación ID 6606-25-LQ16* presente en [11], las características del espectro que se utilizan en Chile muestran en la Tabla 2.1:

Tabla 2.1: Especificaciones de LoRaWAN para Chile.

Especificación/Características	Soporte LoRa
Alcance	3-5 Km en áreas urbanas densas. 10-15 Km en áreas sub urbanas
Banda de Frecuencia	Banda ISM 868 MHz y 915 .
Estándar	Abierto
Modulación	Utiliza tipo de modulación de espectro extendido. La frecuencia se incrementa o disminuye sobre ciertos períodos donde se codifica la información a ser transmitida. Provee 30dB de mejoramiento respecto de FSK
Capacidad	Un Gateway LoRa soporta aprox. 60.000 dispositivos
Duración Batería	Mayor a 10 años
Característica	Comunicación bidireccional y capacidad de encriptación
Capa física LoRa	Se encarga de la frecuencia, energía, modulación y señalización entre los nodos y el Gateway.

### 2.1.1.3. Long Range - Frequency Hopping Spread Spectrum LR-FHSS

El LR-FHSS se presenta en [12] y la descripción de sus características se encuentran en [13], dejando claro que esta modulación solo se usa en el uplink y donde básicamente se define

el LR-FHSS como una modulación FHSS rápida con bit rate de 162 [bits/s] y 325 [bits/s]. Al transmitir un paquete mediante LR-FHSS en un canal, se divide el paquete en frecuencias que siguen la siguiente fórmula:

$$F_{interval} = centr\ freq \pm BW/2 \quad (2.1)$$

Dependiendo del estándar que se esté utilizando es la cantidad de saltos que se va a realizar, sin embargo, los canales físicos se espera que sean usados de forma pareja. La transmisión comienza en una frecuencia aleatoria y luego los saltos también son aleatorios y con una periodicidad de 50 [ms]. Sin embargo, es importante anunciar al receptor sobre estos saltos mediante el encabezado del paquete.

Es importante mencionar que, actualmente, esta modulación no se encuentra disponible comercialmente.

### 2.1.2. Visión general del chip LR1110

El chip LR1110 es una solución que utiliza la geolocalización de LoRa Cloud™, de Semtech, y permite reducir significativamente el consumo energético de los dispositivos. A continuación se presenta una revisión de las características generales con las que cuenta el chip a utilizar, las cuales están presentes en [6].

- **Transmisor RF *half-duplex* de bajo consumo y alta sensibilidad, operable con LoRa y (G)FSK:** Transmisor operable en la banda ISM dentro del rango de 150 – 960[MHz], proporcionando una potencia de transmisión que llega hasta los +22[dBm] para altos consumos energéticos y una potencia máxima de +15[dBm] para eficiencia energética. También presenta la particularidad de ser totalmente compatible con la familia de chips SX1261/2/8 y con el estándar LoRaWAN® definido por la LoRa Alliance®.
- **Utilización de RF para multipropósitos y enfocados en geoposicionamiento:** Cuenta con la capacidad de escanear redes del tipo GNSS (GPS/BeiDou) y con un bajo consumo energético, así como también escanear redes Wi-Fi (802.11b/b/n) con un consumo energético ultra bajo. También, puede realizar una búsqueda continua dentro del rango de los 150 – 2700[MHz].
- **Motor criptográfico:** Presenta un hardware capaz de cifrar/descifrar algoritmos basados en los estándares AES-128. El manejo de credenciales propias para cada dispositivo, o perfil a vincular, permite proteger la confidencialidad dentro de la red, utilizando herramientas tales como: DevEUI, JoinEUI, llaves encriptadas, NwkKey, AppKey, entre otros.

Es importante destacar que, tal como se menciona en [12], la gama de transmisores que se basen en el chip LR1110 (LoRa Edge™), van a estar capacitados para poder utilizar la modulación LR-FHSS cuando esta esté disponible, permitiendo que las redes LoRaWAN tengan una notable mejora frente a la resistencia al ruido y también se espera una gran escalabilidad en cuanto a la cantidad de dispositivos que se pueden conectar a una misma red. La capacidad de utilizar, en un futuro, la modulación LR-FHSS, da cuenta de que, en el momento que se libere esta nueva modulación, también se generará un gran interés por los transmisores/chips que sean capaces de utilizarla, tal como va a ser el caso del LR1110 en un futuro cercano.

### 2.1.3. Fórmula del Haversine

La fórmula del Haversine, o del semiverseno, es una fórmula que permite calcular la distancia entre dos puntos geográficos, los cuales estén definidos por las coordenadas de latitud y longitud clásicas. Esa herramienta sera muy útil al momento de analizar la precisión de la localización de los dispositivos a utilizar. La fórmula viene dada por

$$\text{semiversin}\left(\frac{d}{R}\right) = \text{semiversin}(\theta_1 + \theta_2) + \cos(\theta_1) \cos(\theta_2) \text{semiversin}(\Delta\lambda) \quad (2.2)$$

Con lo que, posteriormente, se puede calcular la distancia con la siguiente fórmula

$$d = R \text{semiversin}^{-1}\left(\text{semiversin}\left(\frac{d}{R}\right)\right) = 2R \arcsin\left(\sqrt{\text{semiversin}\left(\frac{d}{R}\right)}\right) \quad (2.3)$$

Donde

- *semiversin* corresponde a la función semiverseno.
- *d* es la distancia entre los dos puntos geográficos.
- *R* es el radio de la Tierra.
- $\theta_1$  corresponde a la latitud del punto 1.
- $\theta_2$  corresponde a la latitud del punto 2.
- $\Delta\lambda$  es la diferencia de longitudes entre ambos puntos.

## 2.2. Estado del Arte.

A continuación se hará una discusión del estado del arte para evaluar el desempeño de LoRaWAN y por otro lado del chip LR1110. El enfoque para las redes LoRaWAN será para comprobar las investigaciones que se han realizado y que se están realizando, además de mostrar que en la actualidad se siguen haciendo investigaciones más profundas para mejorar su desempeño, ya sea: cambiando la arquitectura, agregando nuevos dispositivos, implementando un nuevo software, etc. En adición a lo anterior, se entregarán ejemplos de trabajos relacionados con la evaluación del desempeño de las redes LoRaWAN en un contexto de aplicaciones *indoor* de las tecnologías IoT.

Por otra parte, con respecto al chip LR1110, se mostrará que se trata de una herramienta emergente y que tiene un gran potencial para aplicaciones que requieran un sistema más adecuado para usos IoT y que también permita utilizar la geolocalización, pero que, sin embargo, no existen muchos trabajos en los cuales se evalúe esta tecnología.

### 2.2.1. Trabajos relacionados: LoRaWAN

Las redes LoRaWAN ya se encuentran muy estudiadas bajo la modulación que se utiliza normalmente. Esto se puede ver en [1], donde, básicamente, se exponen las limitaciones que presentan las redes LoRaWAN, además de hacer una revisión de las características principales, la escalabilidad, la confianza de estas y los distintos usos que se les puede dar. También,

se deja un apartado para poder hablar de algunas mejoras que se les puede dar al hacer algunos cambios dentro de la arquitectura, la capa física, etc., con los cuales se dejan abiertos los desafíos para investigar en el futuro.

Como es de esperar, al tener una herramienta completamente evaluada, se abre el camino para comenzar con las investigaciones para mejorarla, e iniciar las investigaciones para aplicar distintos cambios y evaluar el desempeño que se logra obtener. Un caso particular es el mostrado en [3], donde se analiza la propuesta de utilizar *Slotted ALOHA* en la capa MAC para mejorar ciertas características como: retrasos, el *throughput* y el *packet drop rate*, concluyendo que este cambio puede ser muy útil pero se debe tener cuidado del tipo de algoritmo que se usa, ya que la elección de este depende del objetivo al cual se quiera llegar, como, por ejemplo, el estudio mediante simulaciones y en condiciones de carga baja, media y alta. También, se encuentran investigaciones que se enfocan en realizar mejoras para escenarios específicos, como se puede apreciar en [2], donde se diseña e implementa un Gateway que utiliza multicanales, llegando a mejorar la calidad de las redes que se podían diseñar y probar dentro de un ambiente educacional, para que los estudiantes hagan sus pruebas con redes basadas en IoT.

Tal como se mencionó anteriormente, el interés sobre el desempeño de las redes LoRaWAN sigue siendo un tema de investigación, teniendo casos como [14] [15] [16]. Los trabajos que se mencionaron se caracterizan por plantear desafíos específicos, por lo que se está tratando de llegar a diseños especializados para distintas áreas y proyectos. Sin embargo, para efectos de este trabajo se necesita revisar evaluaciones que sean, básicamente, pruebas de desempeño de las redes. Uno de los enfoques es realizar pruebas para usos *indoor* de las redes LoRa, por lo que se hace un filtrado de los artículos a presentar. Un trabajo actual es [17], donde se realiza un análisis del RSSI que presenta en una red LoRaWAN al ser aplicada en un escenario real e *indoor*, donde el Gateway se dispuso en un lugar estático y se establecieron distintos puntos de interés para las mediciones que van a realizarse. Es importante destacar que, en el ejemplo anterior, se cuenta con una situación que es muy similar a lo que sería un escenario real, lo que quiere decir que la red coexiste con distintos dispositivos electrónicos que provocan interferencia, además de contar con la presencia de distintos obstáculos, ya sea una cantidad considerable de personas, murallas, muebles, etc. Se concluye diciendo que el tener una clara idea de cómo se comporta la intensidad de la señal dentro de un espacio cerrado es fundamental para comprender las variables ambientales que afectan este factor y así considerar ideas para realizar pruebas e investigaciones dentro de ciudades concurridas, permitiendo diseñar redes que ayuden a la aparición de ciudades inteligentes. Otro ejemplo, es el caso de [18], donde se presenta un caso de estudio de cómo se comporta una red con un Gateway dual en un espacio reducido. Incluso, el interés dentro de aplicaciones en interiores es tal que se llega a comparar las tecnologías actuales y de esta forma poder proporcionar una guía de cuál es más útil, dependiendo del caso que se presente o de las cualidades de red que se busquen. En [19] se compara el NB-IoT con LoRaWAN para aplicaciones de *Deep-Indoor*. Otras investigaciones, [20] [21] [22], presentan trabajos de evaluación para escenarios de interior y exterior, pero enfocados en lugares específicos, como Bangkok, Duisburg y Lebanon, respectivamente.

Gracias a que se tiene una gran cantidad de trabajos previos sobre evaluación de LoRaWAN, en escenarios *indoor*, estos se usarán como apoyo para poder decidir las métricas y el

*setup* a utilizar para este trabajo, además de tener una guía de como se deberían comportar las redes LoRaWAN.

#### 2.2.1.1. Trabajos para armar el setup, pruebas y tipo de métricas

Como se quiere tener una base, antes probada, para diseñar el *setup* a utilizar, es necesario tener un conjunto de trabajos anteriores y seguir o basarse en los diseños utilizados para las pruebas. Además, se debe tener en cuenta que no se puede emplear cualquier tipo de métrica para evaluar el desempeño de este tipo de redes, ya que en las redes LPWAN no se consideran tantas variables como en las redes de telecomunicaciones normales. Debido a al detalle previamente mencionado, es que basarse en trabajos anteriores también resulta útil para guiar bien los experimentos y como evaluarlos.

Cabe mencionar que se tiene un listado de algunos trabajos los cuales sirven como base y son usados para decidir: los escenarios, el *setup*, los tipos de pruebas y las métricas a utilizar. Los trabajos corresponden a [23] [24] [25] [26] [27] [28].

#### 2.2.2. Trabajos con el chip LR1110

Al momento de realizar el trabajo de esta memoria, se lograron encontrar solo dos trabajos experimentales en los cuales se utilizaba el chip LR1110. Sin embargo, solo uno era público y solo se hará referencia a este dentro de esta sección.

En [8] la investigación se enfoca en realizar una comparación entre la búsqueda de ubicación geográfica por medio de GNSS versus la posición que se obtiene desde un Ublox ZOE-M8G<sup>5</sup>. Luego de realizar la comparación, también se evalúa el uso energético que presentan ambas herramientas y las diferencias de rendimiento que se lograron ver. Finalmente, se realiza una revisión de lo que podría ser un caso de uso, en donde se evalúa que tan buena sería la solución basada en el LR1110 para diseñar collares del tipo *opencollar* y usarlos para rastrear rinocerontes dentro de un escenario de *Smart Park*.

<sup>5</sup> Modulo ZOE-M8G con capacidad de escaneo para localización la mediante GNSS y GPS, de la compañía Ublox

# Capítulo 3

## Revisión general de Hardware y la Arquitectura de red

Para plasmar de una forma clara qué dispositivos se utilizarán, además de las funcionalidades de estos y la posición de cada uno dentro de la red, se proveen los detalles más importantes de cada tecnología.

### 3.1. Hardware

En cuanto al hardware empleado, principalmente se pueden reconocer tres dispositivos: el gateway, el development kit que incluye el chip LR1110 con la versión LoRa Edge, y el módulo Wisblock, que incluye el chip SX1262 con la versión anterior de LoRa.

#### 3.1.1. Gateway

Las partes que utilizaron para armar el gateway se muestran en la Figura 3.1:



Figura 3.1: Partes para armar el gateway. Visto desde la izquierda hacia la derecha, las partes más importantes son: RAK 2247, RAK 2247 PI HAT, Raspberry pi 3 B+.

Para entender la funcionalidad de cada parte presente en la Figura 3.1, a continuación se provee una breve descripción de cada una de ellas

- **RAK 2247**

Es un módulo concentrador LoRa, el cual está basado en el chip SX1301 y utiliza el estándar de comunicación SPI para las conexiones entre hardware. Tiene hasta 10 rutas de demodulación paralelas y permite la integración de hasta 500 nodos por kilómetro cuadrado y en un entorno de interferencia moderada. Para este trabajo se utilizó la versión que opera dentro de la banda US915.

- **RAK 2247 Pi HAT**

Corresponde a una placa convertidora, que permite al RAK 2247 entablar una comunicación con una Raspberry Pi SBC. Tiene el beneficio de proporcionar un GPS que habilita al RAK 2247 con localización e información horaria

- **Raspberry Pi 3 B+**

Perteneciendo a los ordenadores de placa reducida, es un modelo que incorpora una CPU Cortex-A53 de cuatro núcleos, 64 bit y 1.4 Ghz, la cual incluye una conexión Wi-Fi con estándares 802.11 en 2,4 GHz y 5 GHz. Cuenta también con conexión Bluetooth.

Es importante mencionar que para las configuraciones del gateway, se empleó un software para que asistiera el proceso y que ayudara simplificando la interfaz desde donde se realizarán las configuraciones. Para este caso se utilizó el software proporcionado por Chirpstack, el cual está pensado para interactuar con el gateway a través de una conexión que utilice el protocolo *Secure SHell* (SSH). Es importante destacar que solo se tuvo que mantener el gateway energizado y conectado a una red Wi-Fi para manipularlo con este software, por lo que, cualquier interacción extra, ya sea directamente con alguno de las partes del gateway o mediante alguna otra conexión, se excluyó del trabajo.

### 3.1.2. Development Kit chip LR1110

Para poder probar el chip LR1110, que corresponde a la médula de la nueva gama LoRa Edge™, se optó por utilizar el *Development Kit* de esta tecnología. Para este trabajo se usó la versión *LR1110DVK1TCKS* ya que es la utilizada dentro de la banda de 915[MHz].

En la Figura 3.2, se muestran las partes utilizadas del *Development Kit* y con las cuales se armó el transmisor a utilizar en las pruebas de localización y envío de paquetes.

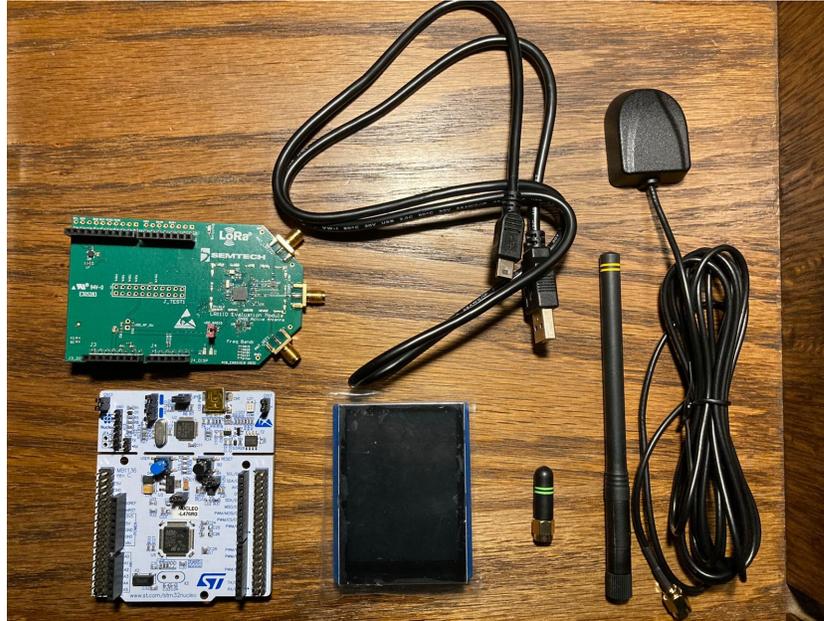


Figura 3.2: Partes necesarias para armar el transmisor con la antena activa de GNSS.

Es importante destacar que es posible programar el microcontrolador del *Development Kit* y realizar pruebas más especializadas e independientes. Sin embargo, para este trabajo se consideró pertinente solo utilizar los firmwares ya disponibles para controlar el transmisor con el chip LR1110 de manera manual y con esto probar las funciones de geolocalización y LoRaWAN.

Los firmwares disponibles son dos. El primero sirve para emplear el *Development Kit* únicamente como transmisor (*Transceiver Firmware*) y que todas las configuraciones extras, tales como el proceso de host para la red LoRaWAN, la lógica de comportamiento del dispositivo, interacciones con el *network server* y el procesamiento para conectarse a la red LoRaWAN, se tengan que realizar de manera externa y por medio de alguna otra herramienta o método. Por otra parte, la otra opción disponible es el *LoRa Basics™ Modem-E*, la cual es una versión más completa y permite que la manipulación del transmisor sea simple. La gran ventaja del *Modem-E* es que se logra la automatización de la mayoría de las interacciones con el *network server* junto con proporcionar un menú interactivo con el cual se pueden realizar los escaneos RF de manera directa.

Las cualidades más importantes de la funcionalidad *Modem-E*, rescatadas de [29], se muestran a continuación:

- Soporta el protocolo LoRaWAN version 1.0.3, clase A.
- Comandos únicos para unión del transmisor a una red LoRaWAN, envío de paquetes hacia el servidor, transferencia de subida para archivo largo de prueba, uso de *Reliable Octet Stream Encoding*, escaneo GNSS y escaneo pasivo de redes Wi-Fi.
- Automatización de los servicios que maneja el dispositivo, lo cual es entregado por medio del *LoRa Cloud Device & Application Services* de Semtech.
- Manejo automático de las configuraciones del dispositivo, dentro de la red LoRaWAN, y envío automático de información importante dentro los paquetes enviados, tal como nivel de batería, temperatura, SNR, RSSI, duración de la ultima bajada de datos, CRC, estado del almacenamiento de GNSS y las respuestas de *debug* de este.
- Asistencia automática del escaneo GNSS por medio del registro de satélites disponible.
- Es un instrumento que está sometido a cambios y actualizaciones para mantenerlo al día, tanto para las nuevas especificaciones como para las nuevas características que se vayan cambiando, e incluso, incorporando en los estándares LoRaWAN y en la misma capa física de la modulación LoRa.

Cabe destacar que la localización del chip LR1110 se realiza mediante el uso de los servicios de LoRa Cloud™. Básicamente, se envían los identificadores únicos de cada red escaneada, ya sea Wi-Fi o GNSS, con lo que posteriormente se calcula la distancia entre el transmisor y dichas redes. Finalmente, se triangula la posición del transmisor con la información antes descrita, donde una localización por medio de redes Wi-Fi no permite obtener el dato de altitud mientras que los escaneo GNSS si lo entregan.

### 3.1.3. Wisblock

La herramienta conocida como *Wisblock* corresponde a un módulo empleado en soluciones para redes LPWAN, el cual presenta la característica de ser una herramienta muy fácil de montar y desmontar sensores, actuadores, antenas, etc. Esta versatilidad permite tener un instrumento muy flexible y que da una gran facilidad al momento de cambiar el diseño de la solución que se está creando, ya que no es necesario incorporar o cambiar algún aspecto del hardware y solo se necesita cambiar partes que se sujetan a la “placa controladora”, que corresponde al microcontrolador programable que automatiza este instrumento. A continuación, se muestra, en la Figura 3.3, una imagen del transmisor armado para la localización por medio de GPS y el envío de paquetes por medio de la red LoRaWAN.



Figura 3.3: Transmisor y localizador empleado para las comparaciones con el chip LR1110, basado en la herramienta Wisblock.

Para el dispositivo mostrado en la Figura 3.3, se emplearon las siguientes herramientas

- **RAK 4630:** corresponde al módulo LoRa necesario para poder conectar el Wisblock con una red LoRaWAN y realizar las configuraciones pertinentes.
- **RAK 1910:** es un sensor capacitado para realizar escaneos de redes GPS y, posteriormente, proporcionar una ubicación geográfica del dispositivo en cuestión.
- **Antenas:** una antena LoRa, para poder utilizar la red LoRAWAN, y una antena GPS, para realizar la búsqueda de posición.

## 3.2. Arquitectura de la red LoRaWAN

Para la red LoRaWAN, se ocuparon los dispositivos mencionados anteriormente, además de tener que decidir qué servicio se utilizaría para el *network server* y, ya que era necesario utilizar los servicios de LoRa Cloud™, se empleó esta herramienta como *application server*. Para simplificar las cosas, se utilizó el **Semtech network server**.

A continuación, se presenta la Figura 3.4, la cual es una imagen ilustrativa de todos los componentes de la red LoRaWAN que se diseñó.

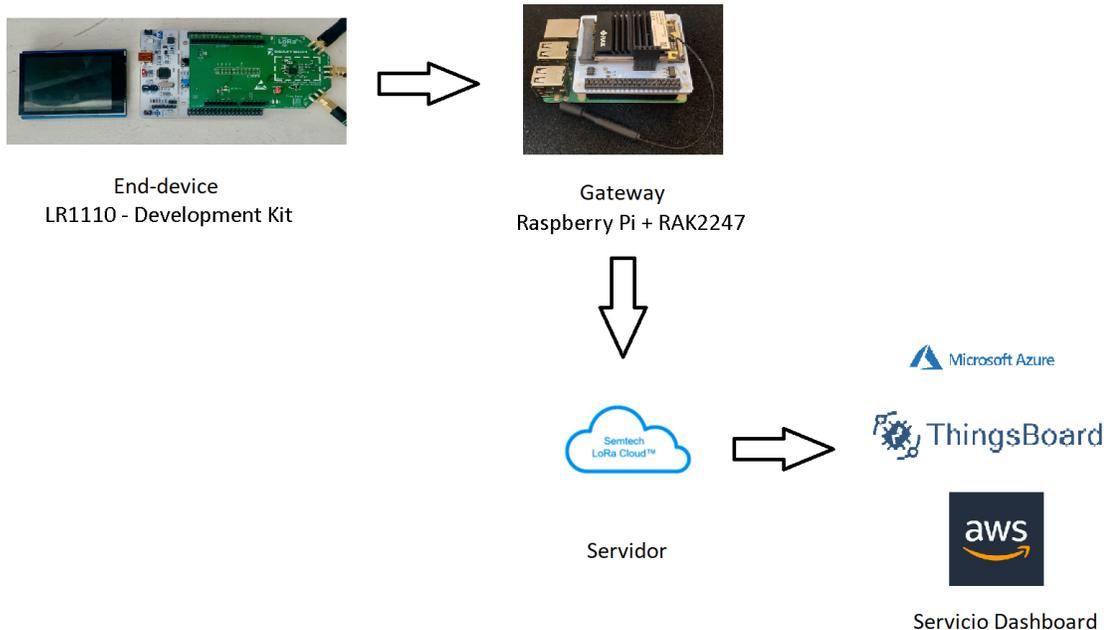


Figura 3.4: Diagrama de los dispositivos que conforman la red LoRaWAN diseñada para el trabajo.

Para el caso de la parte denominada *Servicios de Dashboard*, se incorporó esta herramienta para realizar el enlace completo que se espera de una solución que utiliza una red LoRaWAN, donde se busca tener un monitoreo continuo y actualizado de las variables, o paquetes, que se envían hacia los servidores. De esta forma se diseñó un panel dentro de ThingsBoard, sin embargo, este panel no se utilizó para nada más que para visualizaciones de la red y para monitorear ciertos resultados inmediatos.

En el Anexo A, se dejan todas las configuraciones para montar la red LoRaWAN. Es importante mencionar que está detallado el paso a paso que se siguió para realizar las configuraciones, registros y autenticaciones necesarias para utilizar los servicios de *network server* y *LoRa Cloud™*, que proporciona Semtech. En adición a lo anterior, se muestran las configuraciones, específicas, que se realizaron a los dispositivos físicos de la red.

# Capítulo 4

## Pruebas de comportamiento

La idea principal de este capítulo es recopilar los resultados de las pruebas realizadas para entender el comportamiento del transmisor del *Development Kit*, el cual contiene el chip **LR1110**, y de sus principales características. Para las pruebas se consideró analizar tres escenarios, los cuales eran los principales candidatos para realizar las posteriores pruebas de desempeño y comparar los resultados con el transmisor de Wisblock, el cual está basado en el SX1262.

Los escenarios escogidos fueron los siguientes

- **Escenario 1: Laboratorio de Comunicaciones Avanzadas y 5to piso**

Laboratorio en el 5to piso del Departamento de Ingeniería Eléctrica (DIE) de la Universidad de Chile. Este escenario fue escogido ya que cuenta con artículos eléctricos tales como computadores, microondas, sistema de ventilación y otros, los cuales se pueden utilizar para semejar un escenario de interior tal como una oficina o también una casa con electrodomésticos. También existe la similitud con el escenario utilizado en [27].

- **Escenario 2: Pisos del Departamento de Ingeniería Eléctrica**

A diferencia del escenario anterior, este escenario se pensó para tener una idea de los límites que puede presentar el transmisor dentro de un escenario de interior pero de más de un piso. De esta forma se espera poder asemejar un edificio que necesite servicio de IoT dentro de sus actualizaciones, como puede ser un complejo de oficinas o un complejo de departamentos. También da una idea del comportamiento de la señal dentro de un edificio de nivel académico. La idea principal es entender la influencia que tiene la presencia de murallas, escaleras, pisos “enteros” de distancia, mamparas, puertas, etc., y como afectan estas características dentro de la comunicación entre el transmisor y el gateway. Además, posteriormente se busca comparar dichas características con las que presente el transmisor de Wisblock. Este escenario tiene cierta similitud con un escenario utilizado en [23].

- **Escenario 3: Patio Facultad de Ciencias Físicas y Matemáticas**

Este escenario se escogió para evaluar el desempeño de una de las principales mejoras de LoRa Edge™, el escaneo de GNSS, y que este no se vea influenciado por la presencia de algún techo u otro obstáculo que perjudique la búsqueda de satélites. También fue necesario elegir un escenario de exterior para realizar las comparaciones con la búsqueda de GPS que tiene el transmisor de Wisblock.

## 4.1. Metodología

La metodología consistió en probar los escaneos RF que tiene el chip LR1110 y que se realizan manualmente con el modo Modem-E de del kit. Con ese objetivo en mente, se realizaron escaneos en los distintos escenarios que se mencionaron y, de esta forma, se analizó el comportamiento individual del transmisor del *Development Kit* en cuanto a calidad del escaneo y calidad de la comunicación con el gateway, la cual se espera analizar mediante el SNR y RSSI.

### 4.1.1. Parámetros de la modulación LoRa para el transmisor

Es importante mencionar que para los envíos de paquetes, se utilizó siempre el DR3. Esta decisión fue tomada ya que las pruebas se realizaron dentro de un rango cercano al gateway y, también, porque es una ayuda para dos cosas importantes: primero, para poder visualizar el efecto de cambiar los pisos o posiciones y, segundo, para despreocuparse de las variables del transmisor y dejar esa configuración “automática” al solo establecer un *data rate* en específico. Para obtener el DR antes mencionado, se utilizaron los parámetros presentes en la Tabla 4.1:

Tabla 4.1: Parámetros LoRa para pruebas de comportamiento.

Parámetro	Valor / Rango
Ancho de Banda	125
Spread Factor	7
Code Rate	4/5
Frecuencia de operacion	US915 = 902-928[MHz]

### 4.1.2. Experimentos

Los experimentos se basaron en realizar escaneos de RF y buscar la posición del transmisor. El objetivo principal, como se mencionó antes, es analizar las características de estos escaneos y evaluar su desempeño, lo cual, se realiza mediante una comparación de las distintas configuraciones de cada tipo de escaneo y, también, entre los distintos tipo de escaneos realizados en un mismo conjunto de escenario y *setup*. Cada par escenario/*setup* es identificado como un experimento distinto, tanto para el escaneo de redes Wi-Fi como para los escaneos de GNSS, asistido y autónomo.

El *setup* utilizado se describe mediante imágenes y está presente en las Figuras 4.1, 4.2, 4.3, 4.4 y 4.5, ilustrando la cantidad de obstáculos, distancias, niveles, etc. En adición a esto, se muestran las características de cada experimento mediante la especificación del escenario y también el *setup* utilizado.

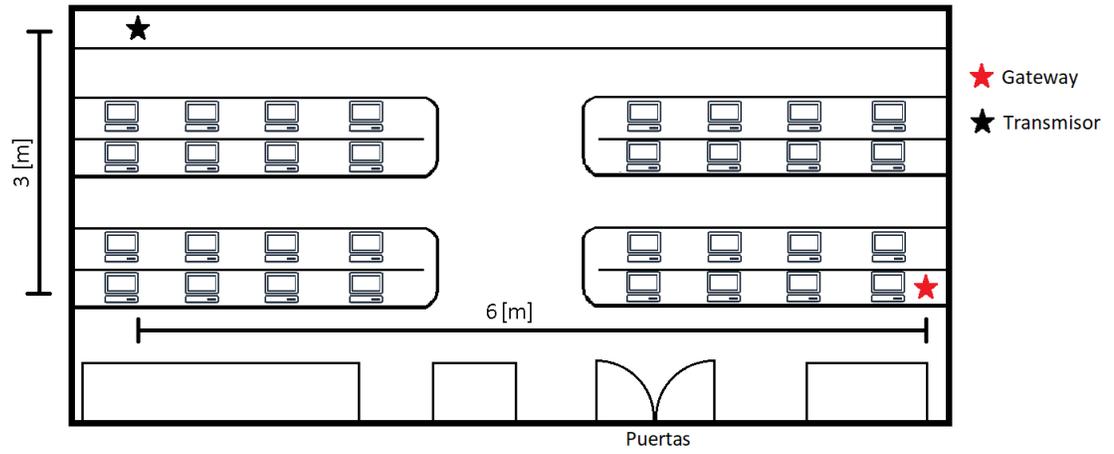


Figura 4.1: Diagrama del Setup 1.

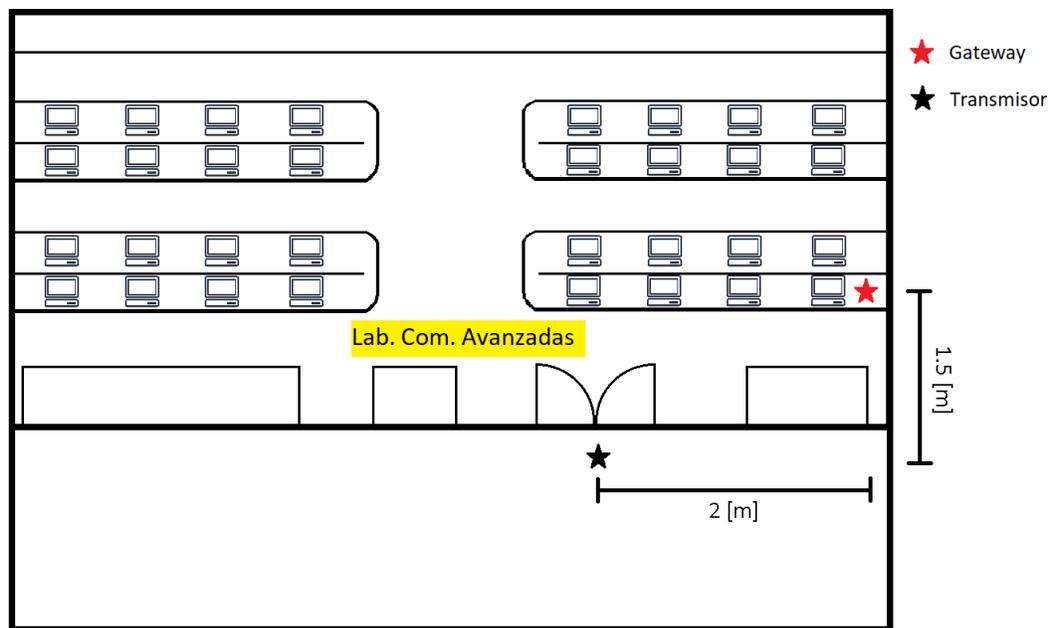


Figura 4.2: Diagrama del Setup 2.

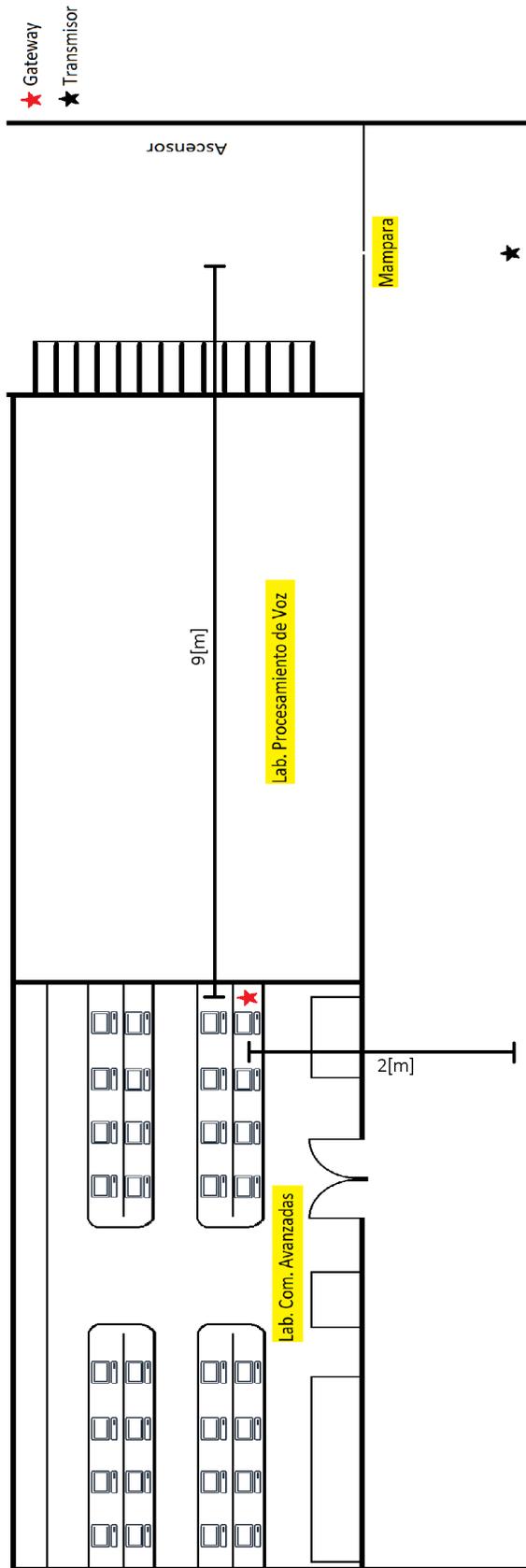


Figura 4.3: Diagrama del Setup 3.

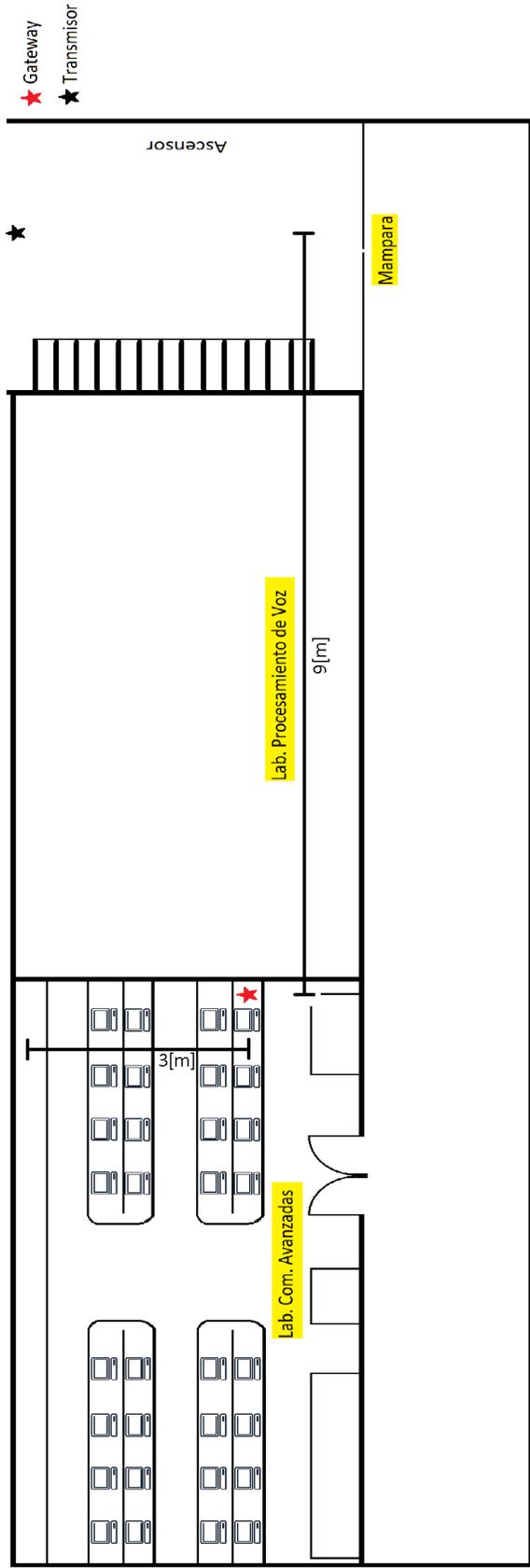


Figura 4.4: Diagrama del Setup 4.

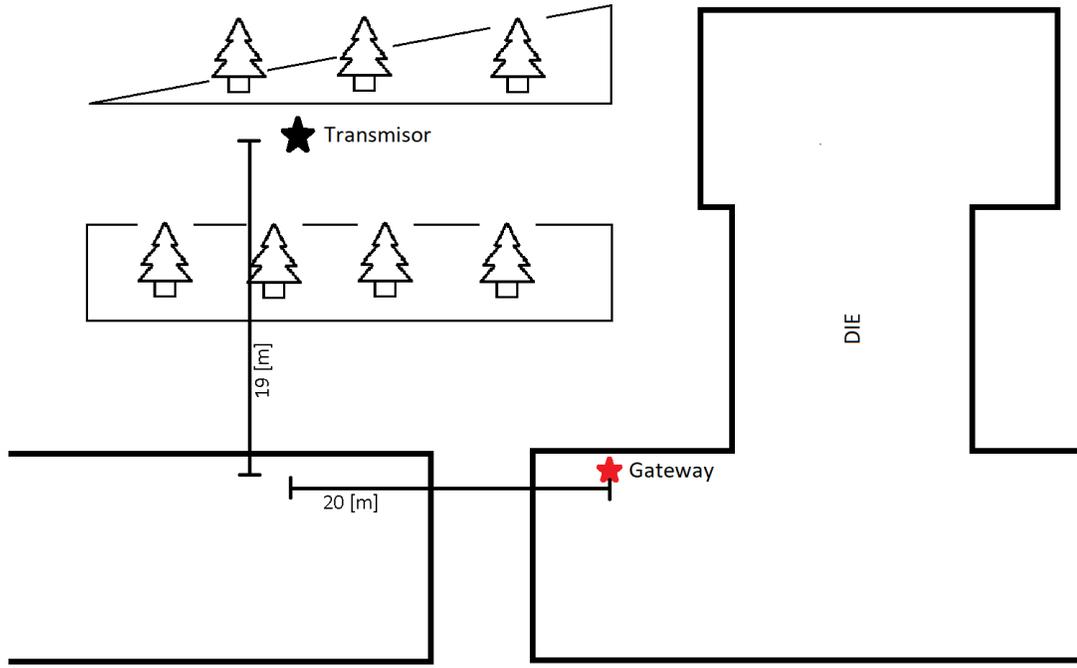


Figura 4.5: Diagrama del Setup 5.

Para tener claro qué combinación de escenario, y setup, corresponde para cada experimento, y que escaneo RF se realizó en cada uno, a continuación se presenta la Tabla 4.2.

Tabla 4.2: Caracterización de cada experimento en las pruebas de comportamiento.

N° de Experimento	Tipo de Escaneo	Escenario	Setup
1	Wi-Fi	1	1
2	Wi-Fi	1	2
3	Wi-Fi	1	3
4	Wi-Fi	2	4
5	Wi-Fi	2	4
6	Wi-Fi	2	4
7	Wi-Fi	2	4
8	Wi-Fi	2	4
9	GNSS asistido bajo consumo	3	5
10	GNSS asistido mejor esfuerzo	3	5
11	GNSS autónomo	3	5
12	Wi-Fi	3	5

## 4.2. Resultados y análisis

Para visualizar los resultados, se muestra inicialmente en forma detallada, el experimento 1 y se entregan los resultados en gráficos de amplitud versus muestra, donde la unidad de la amplitud dependerá de la variable, mientras que la muestra es el número de medición dentro de un total de 30 repeticiones, siendo esta la cantidad de repeticiones utilizadas para cada experimento. El objetivo de esto es poder analizar el comportamiento de cada variable, lo cual fue similar para cada experimento y se caracterizó por la naturaleza no constante de los resultados de cada repetición realizada.

Se deja un análisis de los detalles más importantes que se lograron rescatar.

### 4.2.1. Experimento 1

A continuación se dejan todos los gráficos, presentes en las Figuras 4.6, 4.7, 4.8, 4.9, 4.10 y 4.11, para mostrar los resultados de las pruebas de comportamiento del experimento 1.

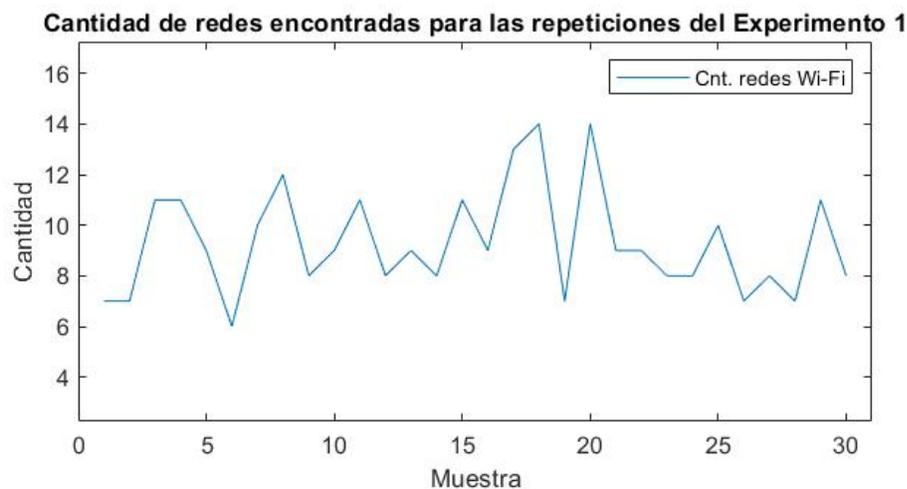


Figura 4.6: Gráfico de la cantidad de redes encontradas por el escaneo de Wi-Fi en el experimento 1.

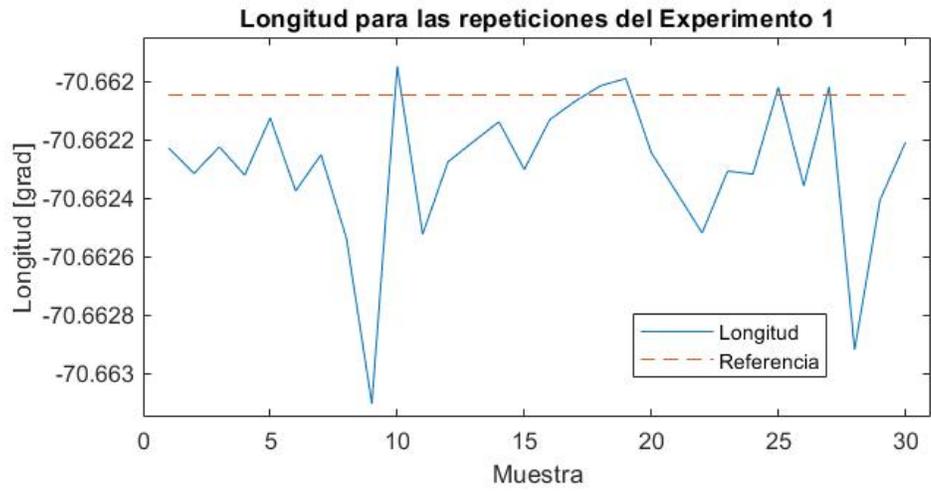


Figura 4.7: Gráfico de las longitudes medidas por el escaneo de Wi-Fi en el experimento 1.

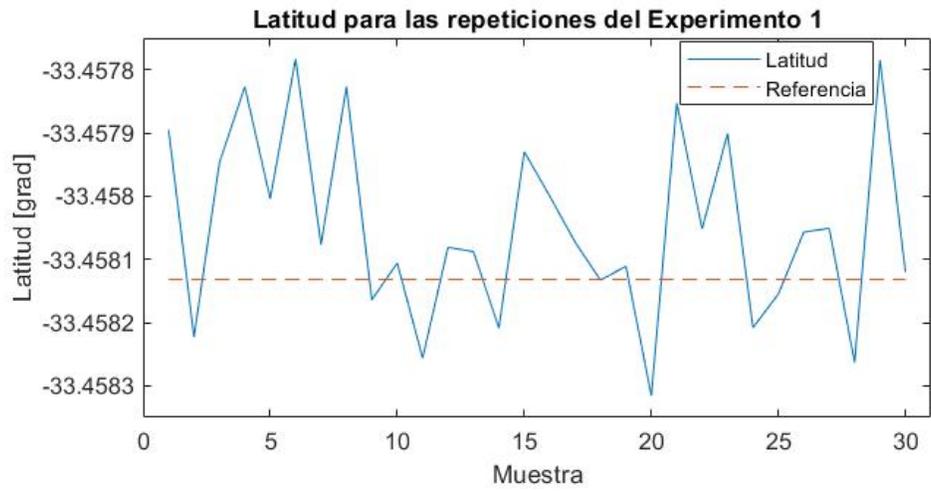


Figura 4.8: Gráfico de las latitudes medidas por el escaneo de Wi-Fi en el experimento 1.



Figura 4.9: Gráfico del error medido entre el punto real y el punto obtenido por el escaneo de Wi-Fi en el experimento 1.

Un detalle importante, es que la distancia medida, desde el punto obtenido por el escaneo hasta el punto de referencia, no permanece constante y tiene casos malos en que, incluso, llega a ser un valor cercano a los  $98.174[m]$ , sin embargo, también se presentan casos bastante buenos; con valores cercanos a los  $2.952[m]$ , lo cual se puede observar en la Figura 4.9. Esto se explica directamente por el comportamiento de la Longitud y Latitud que se observan en la Figura 4.7 y en la Figura 4.8, respectivamente, donde se tienen resultados que varían en unos  $0.0005[grad]$  lo cual es poco, pero, por otra parte, es suficiente como para tener un error de unos  $100[m]$  en la posición brindada por el escaneo Wi-Fi. Sin embargo, se cree que el comportamiento general del error de medición está dentro de un rango aceptable de error, ya que la mayoría de las mediciones se mantienen con un error inferior a los  $49.31[m]$ , siendo este uno de los errores más altos presentados en la Figura 4.9.

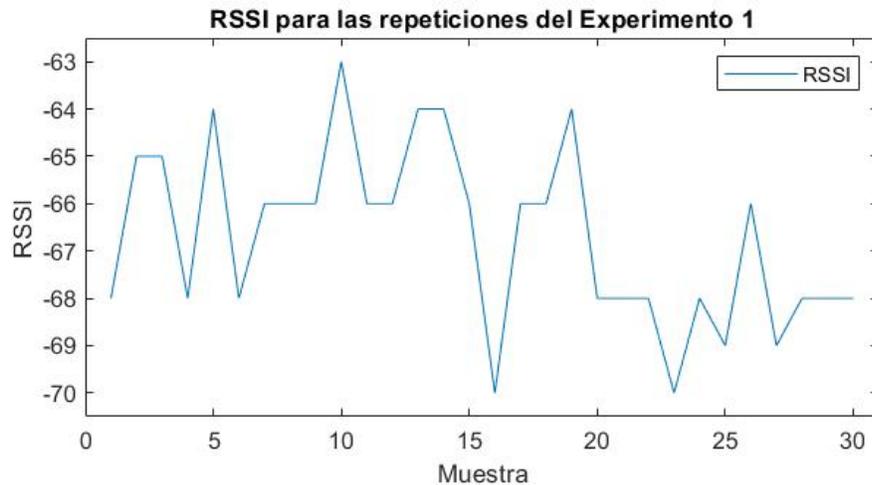


Figura 4.10: Gráfico de los RSSI medidos por el escaneo de Wi-Fi en el experimento 1.

Observando el comportamiento del RSSI, el cual se mantiene dentro del rango entre los  $[-63, -70]$  de RSSI, se presenta una intensidad de señal lo suficientemente buena como para

asegurar una comunicación entre los dispositivos y también se muestra que no tiene variaciones bruscas, como se observa en la Figura 4.10. Sin embargo, esta variable no depende solo de las características del transmisor y hay ciertos parámetros que no se pueden controlar, haciendo que los únicos cambios posibles sean un aumento en la potencia de la antena del transmisor, un aumento en la sensibilidad de wateway o cambiar el setup y colocar el gateway/transmisor en un lugar más favorable.



Figura 4.11: Gráfico de los SNR medidos por el escaneo de Wi-Fi en el experimento 1.

Por otra parte, se encuentra el SNR, la cual es una variable que esta muy ligada al comportamiento del ambiente y, también, en el mismo comportamiento de la señal dentro del escenario utilizado. De esta forma, es que se logra ver, dentro de la Figura 4.11, un gráfico con muchos cambios bruscos, llegando al punto de no ser constante en ningún intervalo y se observa que esta variable esta muy influenciada por las características ambientales del momento exacto en que se realizó en envío del paquete. También, se puede observar que se presentan sectores, dentro de la Figura 4.11, donde la señal se comporta de manera similar pero no se logra identificar ningún patrón predominante. Cabe señalar que el SNR se mantiene dentro del rango de  $[6.75, 11]$ , lo cual es un valor bastante bueno, siendo el SNR típico de LoRA de unos  $-20[dB]$  a  $+10[dB]$ , sin mencionar que nunca se bajó de los 6.75 de SNR, lo cual es siempre un valor sobre 0 y concede la característica de que la señal enviada siempre está “sobre” el nivel de intensidad del ruido presente en el ambiente.

Con respecto al experimento 1, cuyos resultados están mostrados desde la Figura 4.6 hasta la Figura 4.11, es importante destacar la naturaleza variable que se presenta en los gráficos antes mencionados, lo cual evidencia que no existe una tendencia clara en sus comportamientos.

## 4.2.2. Resultados generales

A continuación, se muestran los resultados de todos los experimentos pero con gráficos de amplitud/desviación estándar versus experimento, presentes en las Figuras 4.12, 4.13, 4.14, 4.15, 4.16, 4.17, 4.18, 4.19 y 4.20, donde la desviación estándar se muestra en forma de bloque

y la amplitud es el promedio de todas las repeticiones realizadas para cada experimento, mientras que el eje de las ordenadas corresponde al número del experimento realizado. En adición a lo anterior, se utilizan también gráficos de barras para ciertos resultados que no intentan mostrar ninguna tendencia. Es importante agregar la Tabla 4.3, la cual permite divisar ciertos resultados no graficables. El objetivo es realizar comparaciones de todos los experimentos mediante las métricas estadísticas, tales como promedio, desviación estándar y correlación.

#### 4.2.2.1. Escenario 1

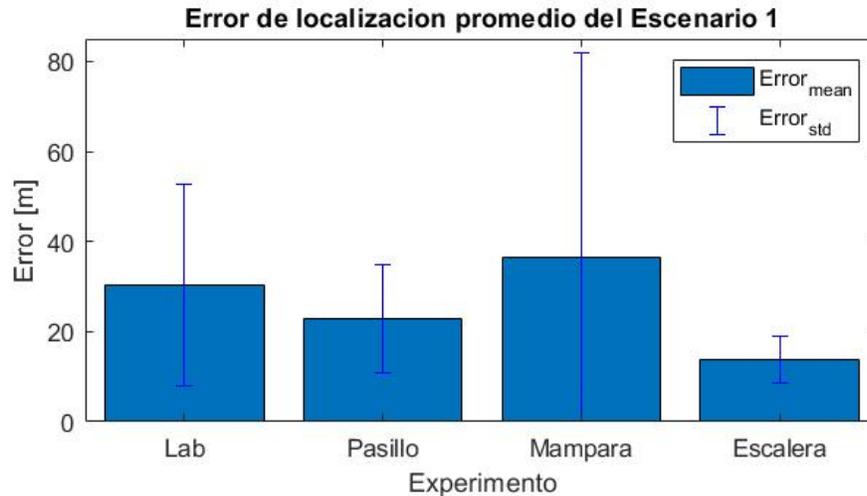


Figura 4.12: Gráfico del error medido entre el punto real y el punto obtenido por el escaneo de Wi-Fi en los experimentos realizados en el escenario 1, mostrando en azul el promedio y rango de la desviación estándar.

Dentro del Escenario 1, se realizaron pruebas enfocadas en analizar como se comporta el escaneo de redes Wi-Fi, dentro de un espacio indoor como lo es el 5to piso del Departamento de Ingeniería Eléctrica (DIE). Como se puede observar en la Figura 4.12, la distancia medida, desde el punto obtenido por el escaneo hasta el punto de referencia, es decir, el error de localización, varía entre los  $30.2[m]$  y los  $13.72[m]$  aproximadamente, lo cual permite entender el rango de “precisión” que tiene la posición obtenida por el escaneo de redes Wi-Fi dentro de las pruebas realizadas en este escenario. Es importante destacar que el comportamiento más variable fue el de la medición realizada cerca de la mampara del piso 5, el cual corresponde al Experimento 3, y presentó una desviación estándar de  $45.599[m]$ . Una de las razones por las que se cree que este experimento presentó una desviación estándar tan grande es debido a que el transmisor se encontraba cerca de una ventana que estaba dirigida hacia el Parque O’Higgins, por lo que se tenía una zona muy grande que no tenía presencia de redes Wi-Fi. Se cree que este detalle dificultaba la tarea de triangular la posición del transmisor por medio de las direcciones MAC escaneadas, particularmente, debido a las redes que se encontraban muy lejos y que “movian” la posición del escaneo. Algo similar sucede con el experimento 1, en donde el transmisor también se encontraba cerca de una ventana, la cual apuntaba hacia el centro de la facultad y podría ser un punto “influenciable” por las redes Wi-Fi del resto de la Universidad.

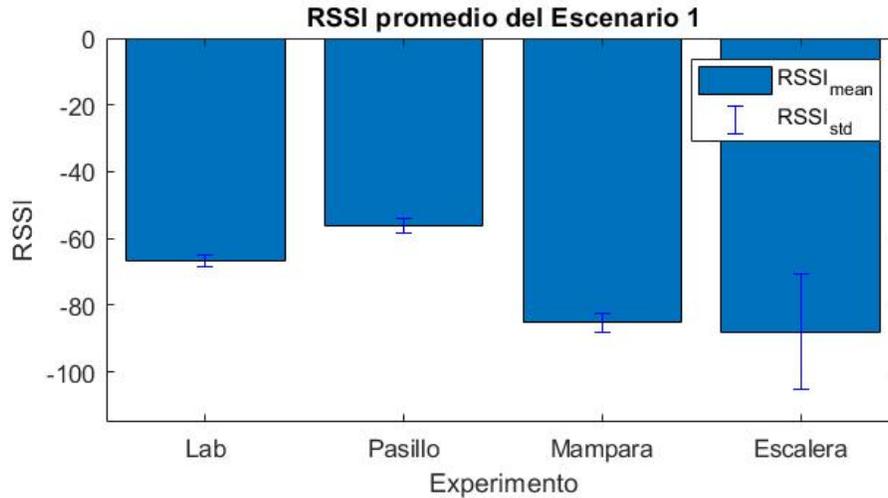


Figura 4.13: Gráfico de los RSSI medidos por el escaneo de Wi-Fi en los experimentos realizados en el escenario 1.

Al analizar la métrica de RSSI, se puede observar en la Figura 4.13 que el valor de esta variable está dentro del rango  $[-88.03, -66.7]$  de RSSI, siendo valores buenos para establecer una comunicación inalámbrica. Un detalle importante es que se logra ver una baja del RSSI a medida que se va alejando del gateway, lo cual se logra al ir cambiando de experimento/setup, teniendo casos como en que el RSSI máximo es en el pasillo y corresponde al experimento más cercano, y también el caso del RSSI mínimo cerca de la escalera y corresponde al experimento más lejano. Cabe resaltar que el cambio de “ambiente” también significa agregar paredes, puertas, ventanas, etc., por lo que no se puede comparar directamente cada experimento. Por otra parte, si se puede decir que el comportamiento es el esperado y no muestra variaciones muy significativas al momento de ir aumentando la distancia y obstáculos entre el transmisor y el gateway. Por lo anterior, se cree que el comportamiento del *Development Kit* en este escenario fue bueno.

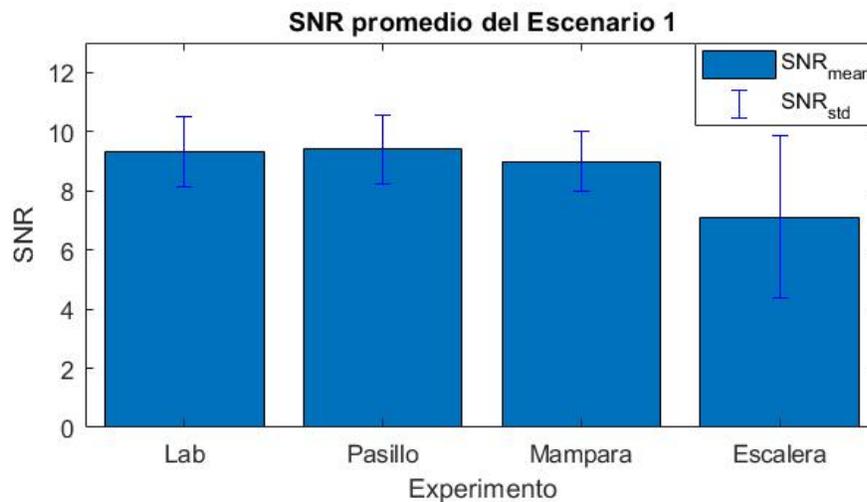


Figura 4.14: Gráfico de los SNR medidos por el escaneo de Wi-Fi en los experimentos realizados en el escenario 1.

En cuanto al SNR, en la Figura 4.14 se ve una tendencia similar al del RSSI y es parecido

al comportamiento que se espera, teniendo valores dentro del rango [47.108] de SNR. Sin embargo, se puede observar que, en la Figura 4.14, en el experimento realizado en la escalera (Experimento 4), se observó una desviación estándar mayor que en los otros casos, con un valor de 2.728 de SNR, y también un menor índice de ruido/señal, con un valor de 7.108 de SNR. Se cree que la razón de este aumento en la varianza del SNR es debido a que se puede estar produciendo un efecto de *multipath*. Esto se adjudica al hecho de que el Laboratorio de Comunicaciones Avanzadas, lugar donde se encontraba el gateway, contaba con una ventana que daba hacia el interior del DIE, y que, además, esta estuviera “cercana” a las escaleras. Esta característica podría generar un camino extra para las señales, provocando que tanto el RSSI como el SNR tuvieran cambios en sus valores y que dependía del camino recorrido para llegar al gateway.

#### 4.2.2.2. Escenario 2

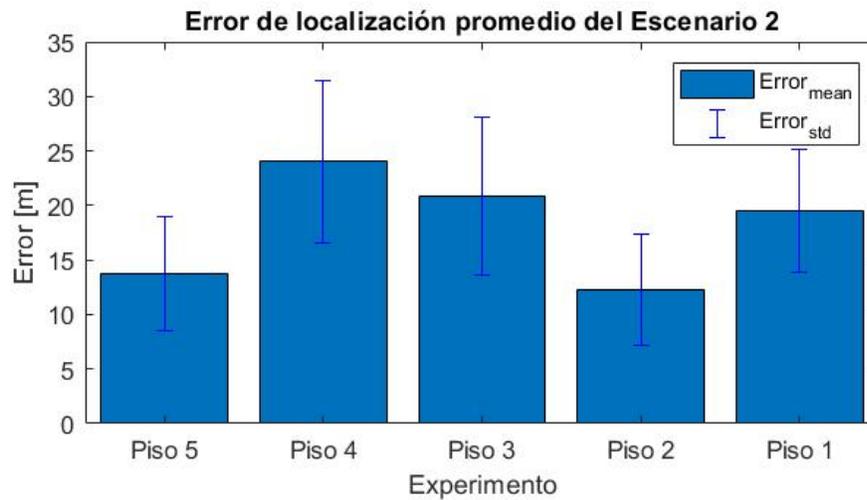


Figura 4.15: Gráfico del error medido entre el punto obtenido por los experimentos realizados en el escenario 2.

Para analizar el Escenario 2, primero se debe tener en cuenta que la idea fue evaluar el comportamiento de la geolocalización, y del transmisor, al ir cambiando la altura e ir cambiando el piso desde donde se realizaba el escaneo de redes Wi-Fi y el envío de datos a la red LoRaWAN. De esta forma, se realizó un experimento similar para todos los pisos del DIE. Primero, al observar el error promedio dentro de la Figura 4.15, se ve que la precisión varía entre los 12.252[m] y los 24.036[m], aproximadamente, siendo un rango aceptable y estando por debajo de los 30[m] esperables. También, se observa una desviación estándar algo similar para los experimentos y que ronda entre los 5.122[m] y los 7.458[m], aproximadamente. Con esto se puede decir que el cambio de altura no parece afectar dentro del posicionamiento geográfico por medio del escaneo de redes Wi-Fi, donde, inicialmente, el valor de *altura* no es posible ser localizado por medio de este escaneo RF.

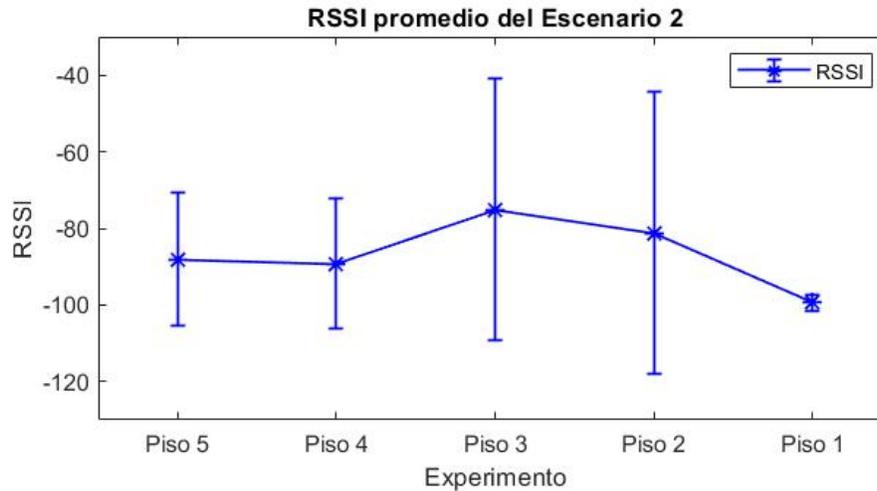


Figura 4.16: Gráfico del RSSI promedio obtenido en los experimentos realizados en el escenario 2.

Al ver la Figura 4.16 y la Figura 4.17, se puede ver una tendencia que va acorde a lo que se espera si es que se aleja el transmisor del gateway; una disminución en el RSSI y el SNR. Por otra parte, es importante mencionar que, en la Figura 4.16, se ve un comportamiento algo extraño del RSSI dentro de los pisos 3 y 2, lo cual se cree que se debe a un *multipath*. Además, se cree que el *multipath* tuvo un efecto aún mayor en los pisos centrales, donde los caminos que toma la señal se van “pareciendo” cada vez más. Por lo que, la disposición del gateway provoca que el camino “extra”, en alguna ocasiones, sea una mejor opción que el camino “convencional” que se toma en otros pisos. Lo antes descrito también explicaría las grandes desviaciones estándar de 34.196 y 36.964, presentes en los pisos 3 y 2, respectivamente, y que sería la variación de caminos que se cree que se está provocando. Dejando fuera el detalle antes mencionado, los valores de RSSI varían dentro del rango  $[-99.433, -75.133]$ , siendo este un rango aceptable para establecer una comunicación y demuestra un buen desempeño. Todo esto, considerando que es un prueba con envío de paquetes dentro de un edificio y cambiando los pisos, lo cual influye en aumentar la distancia y los obstáculos que debe atravesar la señal que proviene desde el transmisor.

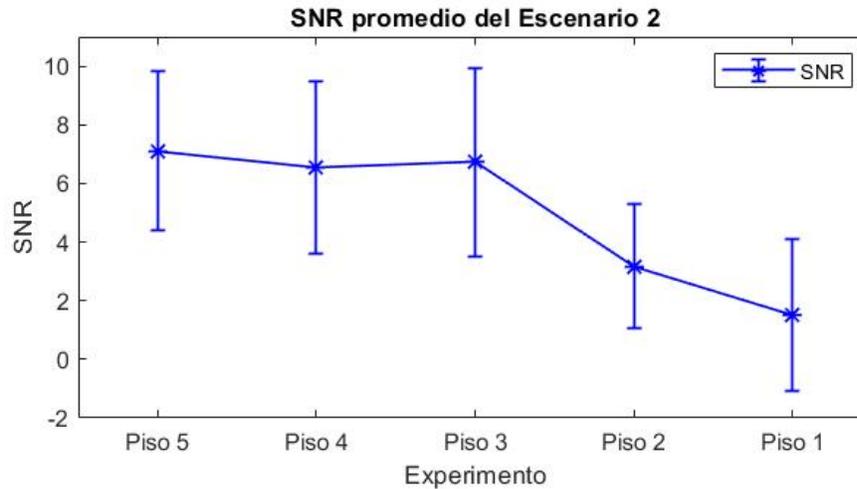


Figura 4.17: Gráfico de los SNR promedio obtenido en los experimentos realizados en el escenario 2.

Viendo la Figura 4.17, el SNR se comporta como se esperaba y muestra una disminución en su valor a medida que el transmisor avanza un piso más, lo cual significa que a la señal le cuesta, cada vez más, superar el ruido del ambiente a medida que este se va alejando del gateway, inclusive si la distancia varía solo en términos de altura. Los valores de SNR varían entre 7.108 y 1.5, teniendo el máximo en el piso 5, el mismo piso del gateway, y el valor mínimo en el piso 1. Sin embargo, aunque se esté en el piso más lejano, de todas formas se tiene un valor de ruido/señal mayor a 0 y se logra superar el piso de ruido presente dentro del Escenario 2.

#### 4.2.2.3. Escenario 3

Se creyó pertinente el tener una mirada del desempeño que presenta el escaneo Wi-Fi en outdoor, y que pusiera a prueba todos los escaneos de RF disponibles, por lo que se utilizó el Escenario 3 y que de esta forma fuera posible incluir el uso de los escaneos de GNSS, logrando obtener un escenario claro para las comparaciones que se realizaron posteriormente con el RAK4630 y su herramienta para rastreo GPS. Es importante mencionar que los escaneos de GNSS se realizaron solo en escenarios outdoor, ya que en escenarios de indoor no fue posible localizar los satélites necesarios para poder encontrar la ubicación, lo cual se realizaba por medio de los servicios de LoRa Cloud™.

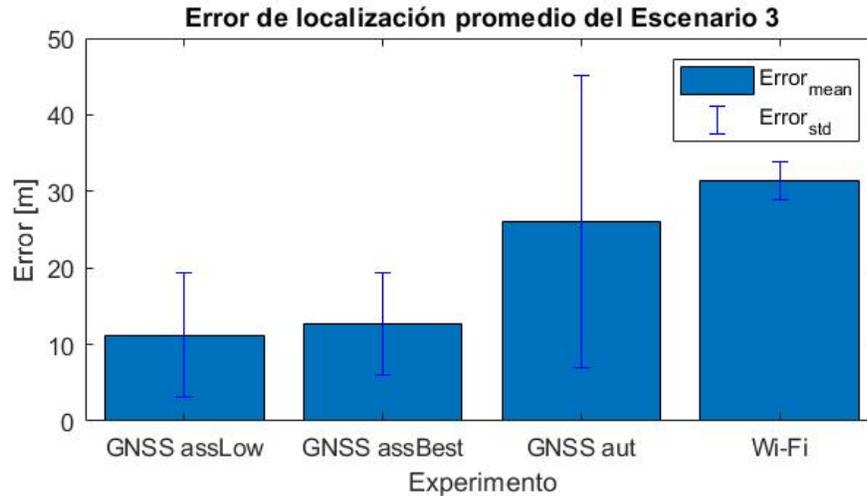


Figura 4.18: Gráfico del error medido entre el punto real y el punto obtenido por los experimentos realizados en el escenario 3.

El Escenario 3 fue escogido pensando en algún lugar dentro de la ciudad, pero con un mínimo de visión del cielo y que permitiera realizar los escaneos de GNSS, siendo este un caso similar de algún recinto con áreas sin techo o el caso de un parque cualquiera. De esta forma, al analizar el gráfico de error, presente en la Figura 4.18, se puede ver que para los dos casos de escaneo GNSS asistido se tiene un resultado muy similar, presentando un valor de  $11.239[m]$  para el escaneo de bajo consumo y  $12.749[m]$  para el escaneo con el mejor esfuerzo. Estos resultados dan la sensación de que la única diferencia, entre ambos escaneos GNSS asistidos, es que: en situaciones donde la “asistencia” no es suficiente y se necesita un extra de potencia (energía), existe la posibilidad de realizar un mayor esfuerzo para sobrellevar algún factor adverso, el cual dificulte el escaneo (outdoor) de GNSS. También, se observa que, a diferencia del escaneo GNSS autónomo, la presencia de una “base de datos” para los escaneos GNSS asistidos es una herramienta que permite aumentar el desempeño del escaneo y que efectivamente se logra encontrar una mayor cantidad de satélites. Se cree que, esta particularidad, influye directamente en la precisión de la ubicación encontrada por el escaneo GNSS. Por lo que, la precisión decae cuando la cantidad de satélites encontrados es cercano al mínimo necesario para triangular la posición del transmisor.

Para el caso del escaneo GNSS autónomo, se pudo evidenciar que este modo depende mucho de las condiciones de visibilidad del cielo que se presenten al momento de realizar el escaneo. Para el caso particular del Experimento 11, el cual corresponde a un sector con árboles que no tapan completamente la vista al cielo, se tuvieron que realizar más de 80 repeticiones para poder obtener los resultados de geolocalización necesarios, sin mencionar que siempre se tuvieron que realizar con la cantidad (mínima) de 6 redes GNSS. Además, se pudo ver que la disponibilidad de satélites va variando con respecto a la hora y el día, teniendo casos donde el escaneo GNSS autónomo no logró encontrar ubicación durante 20 repeticiones, seguidas, del experimento. También, existieron otros casos donde se lograba encontrar la ubicación; una de cada dos veces que se repetía el experimento. Sin embargo, al desconocer los horarios de los satélites, no fue posible contrastar los resultados con ese dato. En adición a lo anterior, la precisión de la ubicación para el escaneo GNSS autónomo no es mala, con un error de  $26.073[m]$ , pero presenta una desviación estándar alta, con respecto a los otros tipos de escaneo, de  $19.15[m]$ . Se piensa que, este detalle, se debió a la mala calidad del escaneo dentro del escenario escogido, provocando una gran variación en los resultados y también que no se

lograra triangular la posición de la mejor manera.

Finalmente, para el caso del escaneo Wi-Fi, se puede observar que es el escaneo que obtuvo un mayor error promedio, de  $31.423[m]$  con respecto al punto de referencia. Sin embargo, este es el experimento que presenta la menor desviación estándar dentro del Escenario 3, lo cual se puede explicar al ser un escenario “protegido” por edificios y otros obstáculos que dificultan el escaneo de redes muy lejanas o que están muy opacadas por los obstáculos presentes. De esta forma se tiene un resultado con poca varianza pero que pareciera estar influenciado por las redes que se escanean (distancia) y no por la cantidad.

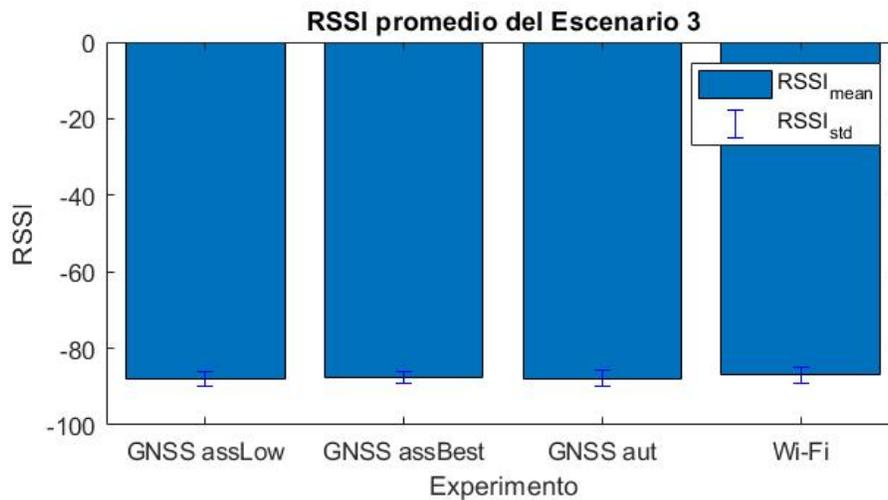


Figura 4.19: Gráfico de los RSSI promedio obtenido en los experimentos realizados en el escenario 3.

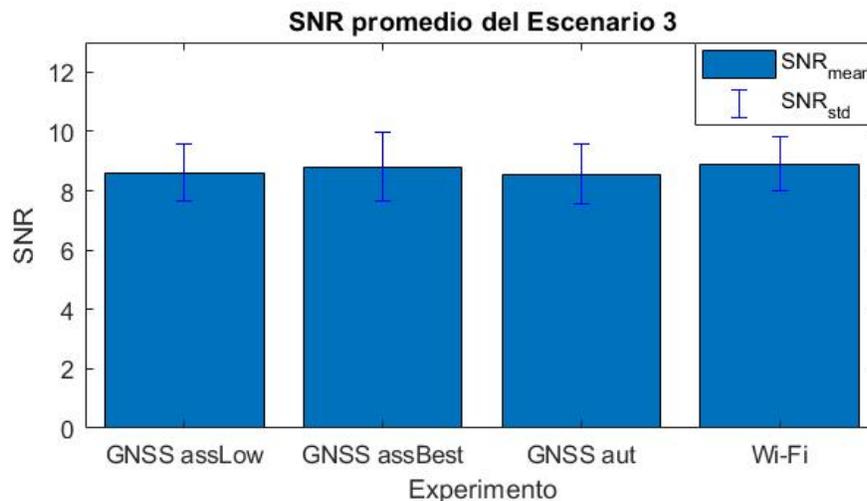


Figura 4.20: Gráfico de los SNR promedio obtenido en los realizados en el escenario 3.

Con respecto al RSSI y el SNR, los resultados presentes en la Figura 4.19 y en la Figura 4.20, muestran resultados parejos y que evidencian el hecho de que todos los experimentos fueron realizados desde el mismo punto y con condiciones similares dentro de un mismo escenario. Es importante mencionar que el RSSI cercano a  $[-88, -87]$  es un valor bastante

bueno al considerar que se esta realizando una prueba de transmisión de paquetes desde un transmisor que se encuentra en el patio, y hacia un gateway que se encuentra al interior de un 5to piso de un edificio. Para el SNR, se cree que el resultado es el esperado ya que al momento de realizar las pruebas no se presencié ningún fenómeno que provocara algún aumento de ruido y se permitió que todas las pruebas fueran sometidas a las mismas fuentes/cantidades de ruido. También, se cree que un valor de SNR que varía entre [8.5, 8.9] es muy bueno y se da que el transmisor logró superar el piso de ruido presente en este escenario, inclusive acercándose a los mejores casos con un valor de  $10[dB]$ .

Tabla 4.3: Tabla de la correlación entre el error (distancia) de posición y la cantidad de redes escaneadas, para todos los experimentos.

<b>Experimento</b>	<b>Correlación</b>
Ex 1	-0.1565
Ex2	0.2542
Ex3	0.1259
Ex4	0.1280
Ex5	-0.1291
Ex6	0.0411
Ex7	0.0437
Ex8	-0.0683
Ex9	0.0018
Ex10	-0.1286
Ex11	-0.1021
Ex12	0.2034

Cabe destacar que la correlación entre la cantidad de redes y la distancia medida, desde el punto obtenido por el escaneo hasta el punto de referencia, es un valor cercano a cero, como se puede apreciar en la Tabla 4.3, y significa que la precisión del posicionamiento por medio de escaneo de redes Wi-Fi no mejora, o empeora, de una manera proporcional si es que la cantidad de redes escaneadas varía; comportamiento que se ve como una tendencia general dentro de todos los experimentos y escaneos RF realizados en el Capítulo 4 y que se puede observar dentro de la Tabla 4.3.

# Capítulo 5

## Pruebas Comparativas

Luego de tener los resultados necesarios para entender el comportamiento de los escaneos RF con que cuenta el chip LR1110, fue necesario realizar experimentos enfocados en comparar el desempeño de la gama LoRa Edge™, frente a alguna otra solución con la versión previa de LoRa, como el transmisor LoRa SX1262 integrado en la herramienta *Wisblock* (RAK4630). De esta forma, se estudian las mejoras, cambios, escenarios favorables, etc, que se observan luego de realizar los experimentos y obtener los resultados mostrados en este capítulo. Es importante mencionar que, además de utilizar los 3 escenarios mencionados en el capítulo anterior, se agregó un escenario que estaba enfocado en utilizar el escaneo GNSS autónomo de una buena forma, para lo cual se tenía la hipótesis de que sería en un escenario con visión bien despejada del cielo y que no tuviera obstáculos como árboles y edificios.

- **Escenario 4: Cancha de la Facultad de Ciencias Físicas y Matemáticas**

Este escenario cuenta con una visibilidad parcialmente despejada, ya que por temas de alcance del gateway hacia un *access point*, fue necesario estar no muy lejos de algún edificio. Sin embargo, se contaba con una mejor visibilidad que la utilizada para realizar los experimentos en el Escenario 3. En este escenario se provo el escaneo de GNSS autónomo en un lugar favorable para este y que su desempeño no este influenciado por esta variable, para, también, intentar igualar las condiciones y luego realizar las comparaciones con la solución basada en Wisblock.

### 5.1. Metodología

Para las pruebas realizadas con el chip LR1110, se siguió la misma metodología que se utilizó en las pruebas para evaluar su comportamiento. La diferencia es que se utilizaron distintos setup para el Escenario 1, además de tener la incorporación del Escenario 4 y los setup utilizados en este.

En cuanto al RAK4630, los parámetros para la modulación LoRa son los mismos que se usaron para las pruebas del Capítulo 4, presentes en la Tabla 4.1, lo cual se realizó pensando en que las condiciones de las pruebas fueran lo más similares entre ambos chips LoRa y que, posteriormente, sus resultados fueran comparables. Por la misma razón, se utilizó el mismo setup para ambos dispositivos y se realizaron las pruebas simultáneamente, además de tener los transmisores uno al lado del otro, siendo esto un solo experimento y que las diferencias en los resultados no se provocara por un cambio de posición, distancia o ambiente.

Es importante mencionar que, a diferencia de la funcionalidad manual del *Development Kit* del chip L1110, la solución basada en Wisblock no se puede manipular manualmente y se debe escribir un código para ser subido al microcontrolador. Para el código del RAK4630, se utilizó el IDE de Arduino y también se utilizaron los códigos de ejemplo para una red LoRaWAN y la utilización del RAK1910 como rastreador GPS, los cuales estaban proporcionados por el mismo IDE de Arduino, luego de descargar las librerías necesarias para programar este dispositivo. Basándose en los códigos de ejemplo, se tuvieron que configurar todos los parámetros necesarios para que el dispositivo lograra conectarse a la red LoRaWAN previamente configurada y montada, tales como el *device eui*, banda de operación **US915**, etc., y que se lograra realizar la OTAA<sup>6</sup> de manera automática. También, se dejó el código configurado para que el Wisblock enviara un mensaje autónomo, cada 20 segundos, con la posición encontrada luego de un rastreo GPS.

### 5.1.1. Experimentos

Los experimentos se basaron en realizar un escaneo RF con el chip LR1110 y buscar la posición del transmisor. Mientras tanto, la solución basada en Wisblock se encontraba enviando la última posición encontrada por el rastreo GPS. Se realizaron experimentos que permitieran comparar los desempeños de ambas opciones, teniendo como métricas el RSSI, SNR y el error de la posición en metros, calculado de acuerdo a la ecuación 2.3.

A continuación se proveen los diagramas de los setup utilizados para realizar las pruebas comparativas. Sin embargo, se continuará con la enumeración seguida anteriormente, por lo que el primer setup es el **Setup 6**. Es importante destacar que todos los setup utilizados, principalmente desde el 6 al 10, se muestran en las Figuras 5.1, 5.2, 5.3, 5.4 y 5.5 y que estos se parecen a los setup utilizados en el Capítulo 4, solo que existen algunas variaciones con respecto a la posición del gateway y/o del transmisor, sin mencionar que, también, se agrega el setup 11, que se muestra en la Figura 5.6 .

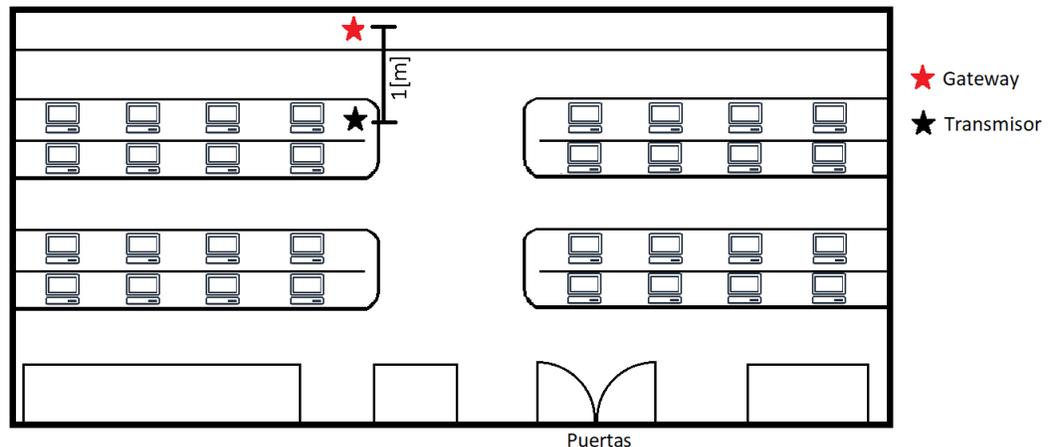


Figura 5.1: Diagrama del Setup 6.

<sup>6</sup> Over-the-Air Activation

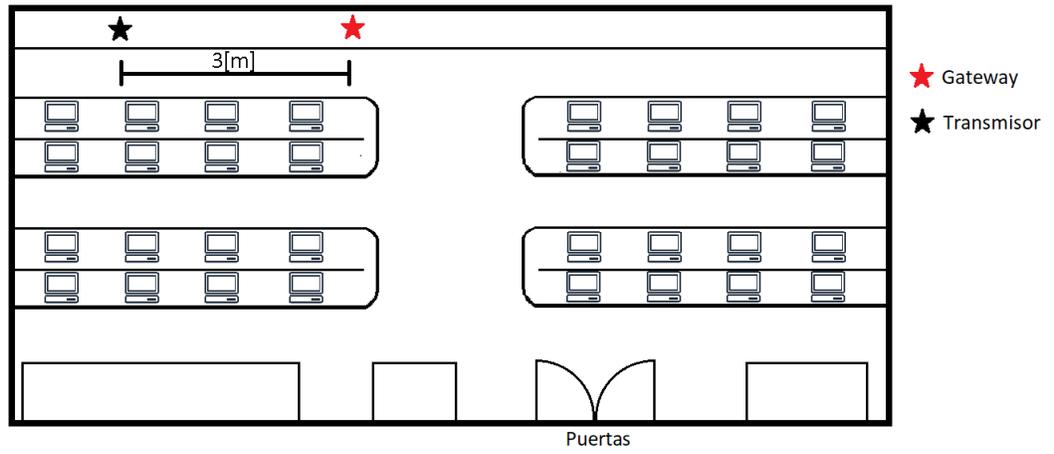


Figura 5.2: Diagrama del Setup 7.

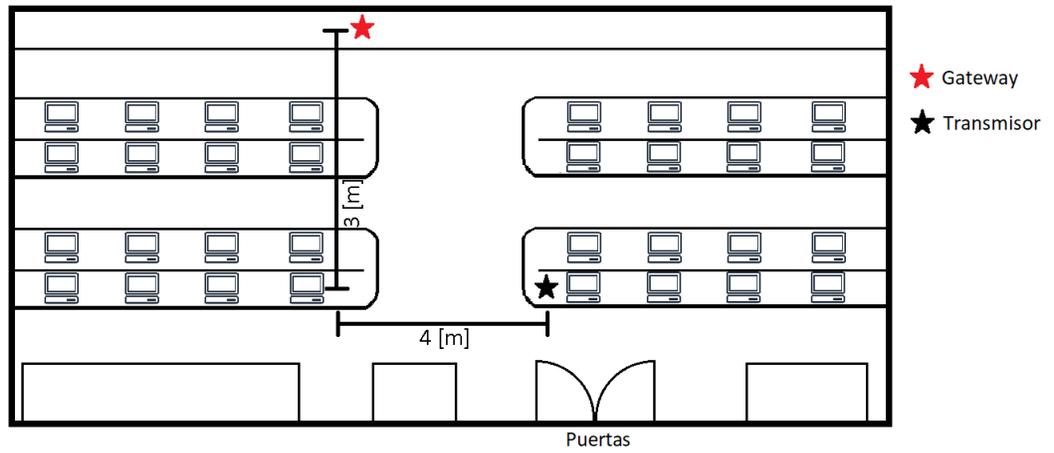


Figura 5.3: Diagrama del Setup 8.

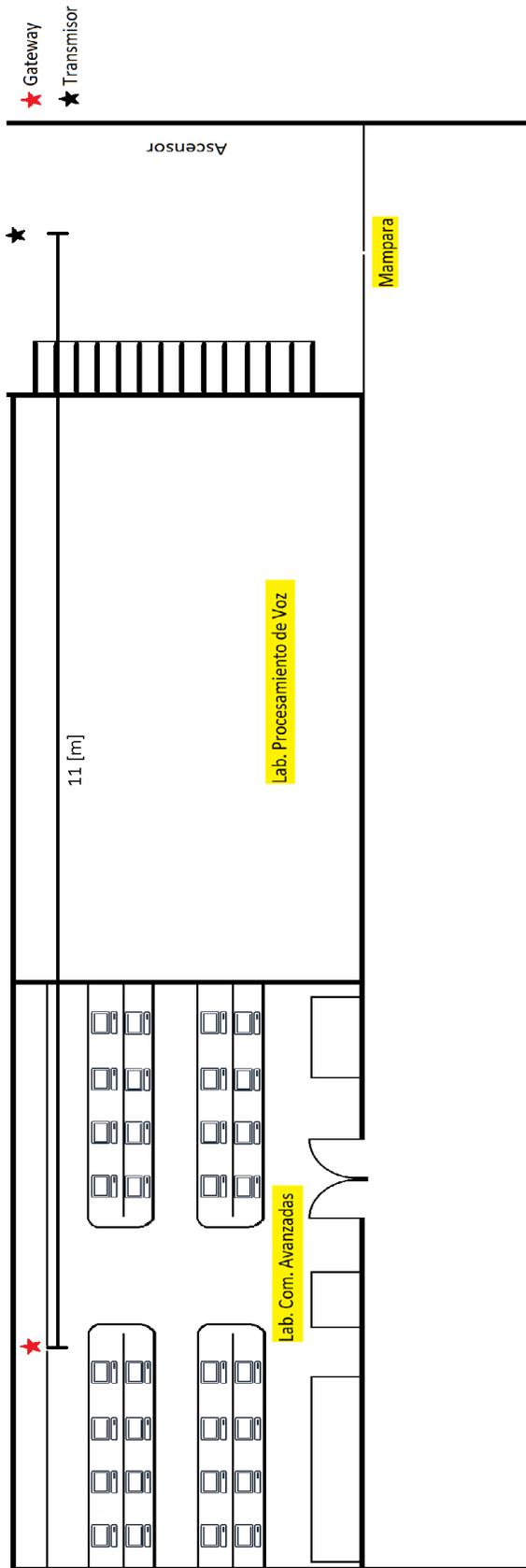


Figura 5.4: Diagrama del Setup 9.

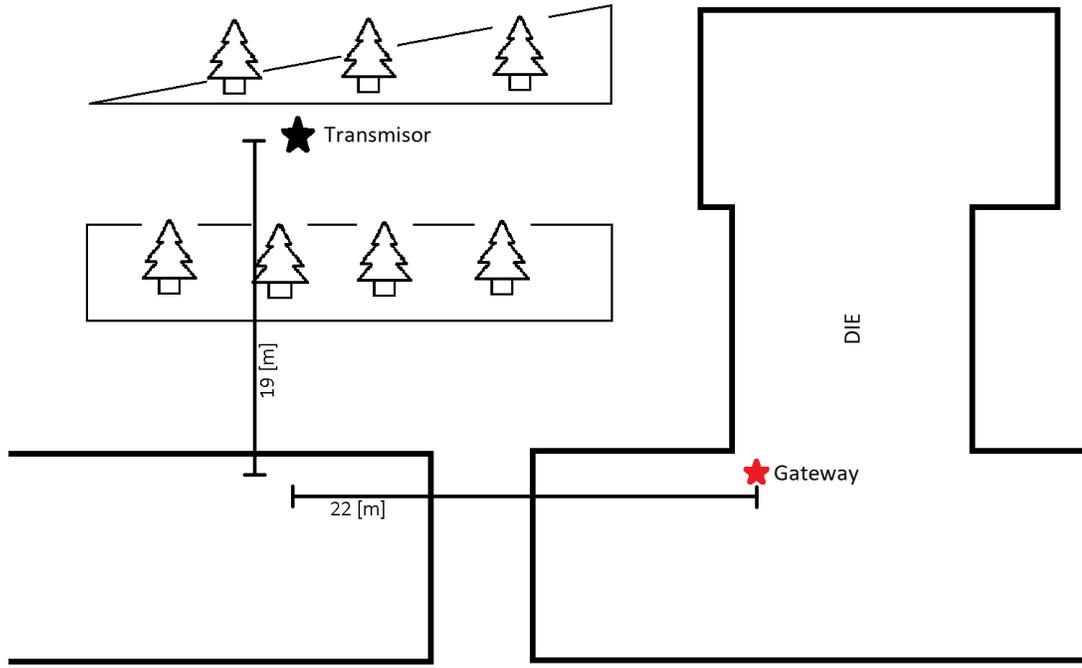


Figura 5.5: Diagrama del Setup 10.

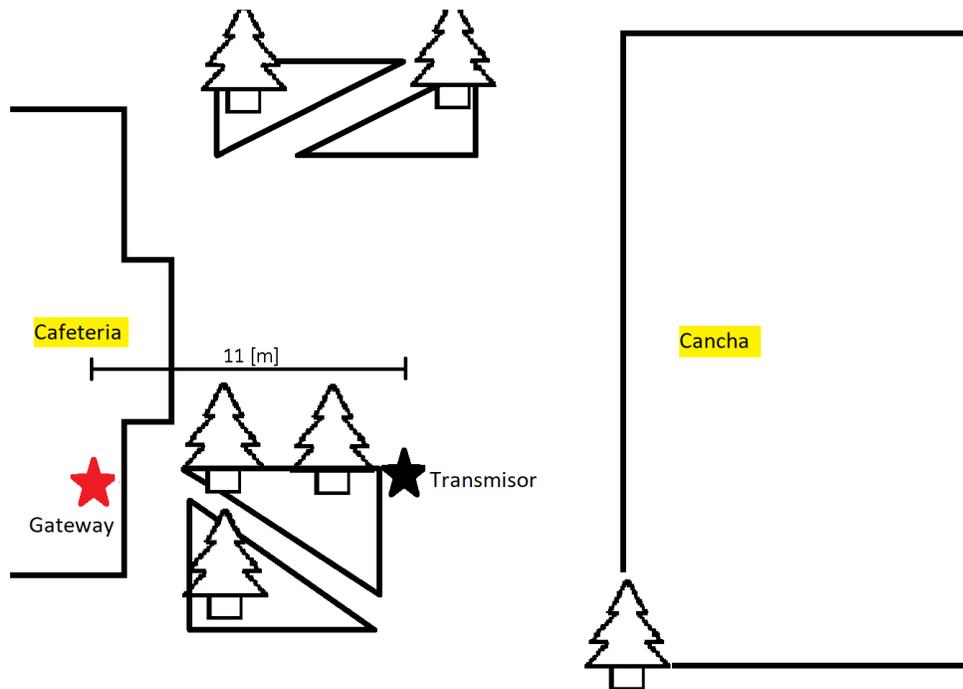


Figura 5.6: Diagrama del Setup 11.

Para tener claro qué combinación de escenario y setup corresponde a cada experimento, y qué escaneo RF se realizó en cada uno, a continuación se describe de cada experimento realizado en las pruebas desempeño.

Tabla 5.1: Caracterización de cada experimento en las pruebas comparativas.

N° de Experimento	Tipo de Escaneo	Escenario	Setup
13	Wi-Fi	1	6
14	Wi-Fi	1	7
15	Wi-Fi	1	8
16	Wi-Fi	2	9
17	Wi-Fi	2	9
18	Wi-Fi	2	9
19	Wi-Fi	2	9
20	Wi-Fi	2	9
21	GNSS autónomo	3	10
22	GNSS asistido bajo consumo	3	10
23	GNSS asistido mejor esfuerzo	3	10
24	Wi-Fi	3	10
25	GNSS autónomo	4	11
26	GNSS asistido bajo consumo	4	11
27	GNSS asistido mejor esfuerzo	4	11
28	Wi-Fi	4	11

## 5.2. Resultados y análisis

Al igual como se realizó en el capítulo anterior, primero se muestra, detalladamente, los resultados del experimento 13. Los resultados serán entregados en forma de gráficos de amplitud versus muestra, donde la unidad de la amplitud dependerá de la variable que se este mostrando; mientras que la muestra es el número de medición dentro de un total de 30 repeticiones, siendo esta la cantidad de repeticiones utilizadas para cada experimento. El objetivo es poder analizar el comportamiento de cada variable y visualizar cómo se se comportan ambos dispositivos.

### 5.2.1. Experimento 13

A continuación se dejan todos los gráficos, presentes en las Figuras 5.7,5.8,5.9,5.10 y 5.11, para mostrar los resultados de las pruebas comparativas del experimento 13, dejando los resultados bien detallados.

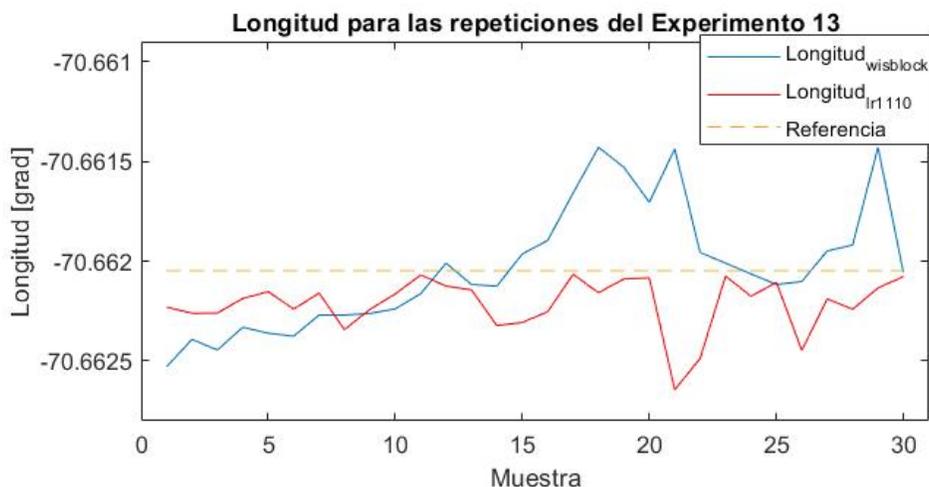


Figura 5.7: Gráfico de la Longitud encontrada por el chip LR1110 (rojo) con el escaneo de Wi-Fi y para GPS del Wisblock (azul), en las repeticiones del experimento 13.

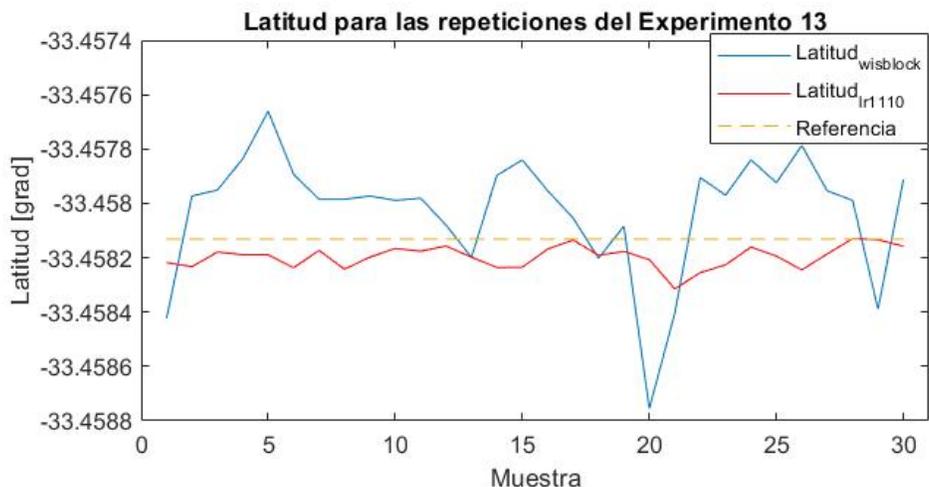


Figura 5.8: Gráfico de la Latitud encontrada por el chip LR1110 (rojo) con el escaneo de Wi-Fi y para GPS del Wisblock (azul), en las repeticiones del experimento 13.

El comportamiento de la Longitud y la Latitud, presentes en la Figura 5.7 y la Figura 5.8 respectivamente, es similar para ambos dispositivos, teniendo en cuenta que se comportan como un valor que no se mantiene constante a lo largo del tiempo. Es importante destacar que las posiciones de longitud y latitud del chip LR1110 se mantienen dentro de un rango más acotado que las encontradas por el RAK4630, lo cual se cree que se debió a la característica de indoor del escenario, provocando que las búsquedas de GPS no sean muy exactas, incluso llegando a casos como el de los escaneos GNSS del chip

LR1110 que no lograron detectar ningún satélite dentro de las repeticiones del Experimento 13. Las variaciones de Longitud para el chip LR1110 están entre los  $-70.6626[grad]$  y los  $-70.6621[grad]$  ( $0.0005[grad]$ ), mientras que para el Wisblock varía entre los  $-70.6625[grad]$  y los  $-70.6614[grad]$  ( $0.0011[grad]$ ). Las variaciones de Latitud para el chip LR1110 están entre los  $-33.4581[grad]$  y los  $-33.4583[grad]$  ( $0.0002[grad]$ ), mientras que para el Wisblock varía entre los  $-33.4588[grad]$  y los  $-33.4577[grad]$  ( $0.0011[grad]$ ).

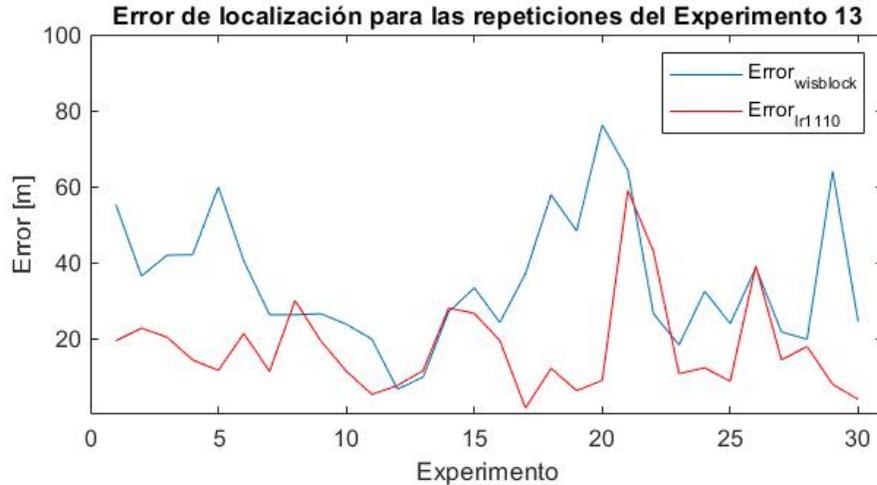


Figura 5.9: Gráfico del error calculado entre la posición de referencia y la posición encontrada por el chip LR1110 (naranja) con sus escaneos RF y el error de la posición para GPS del Wisblock (azul), en las repeticiones del experimento 13.

Las diferencias entre las variables antes analizadas, provocan que el comportamiento de la distancia/error para ambos dispositivos sea un valor no constante, como se puede observar en la Figura 5.9, y permite ver de manera preliminar que la solución basada en Wisblock presenta un mayor error con respecto a la posición de referencia, teniendo la curva del chip LR1110 siempre por debajo de la curva perteneciente al RAK4630. El chip LR1110 presenta un error máximo de  $58.9648[m]$  y un mínimo de  $1.7329[m]$ , mientras que la solución basada en Wisblock tuvo un error máximo de  $76.2859[m]$  y un mínimo de  $7.7021[m]$ , lo cual muestra una mejor precisión por parte del chip LR1110, con un escaneo Wi-Fi, para un escenario indoor y cercano a una ventana.

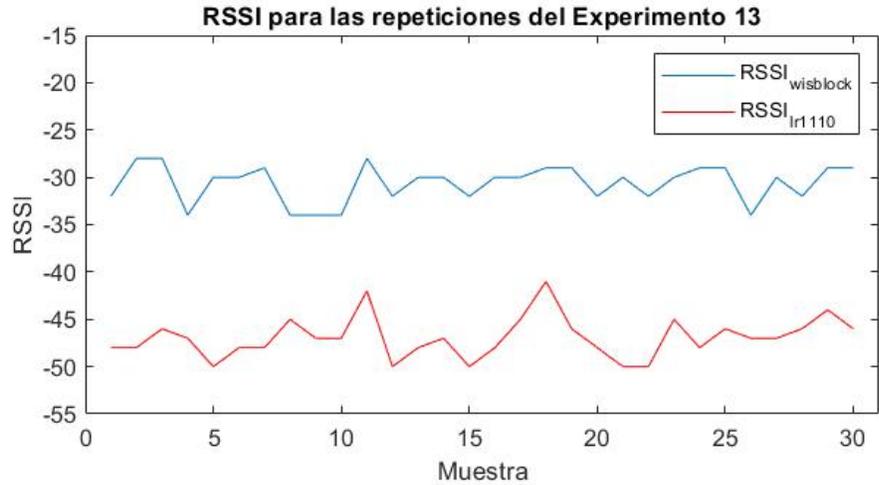


Figura 5.10: Gráfico del RSSI medido para el transmisor del chip LR1110 (rojo) con el escaneo de Wi-Fi y para el transmisor del Wisblock (azul), en las repeticiones del experimento 13.

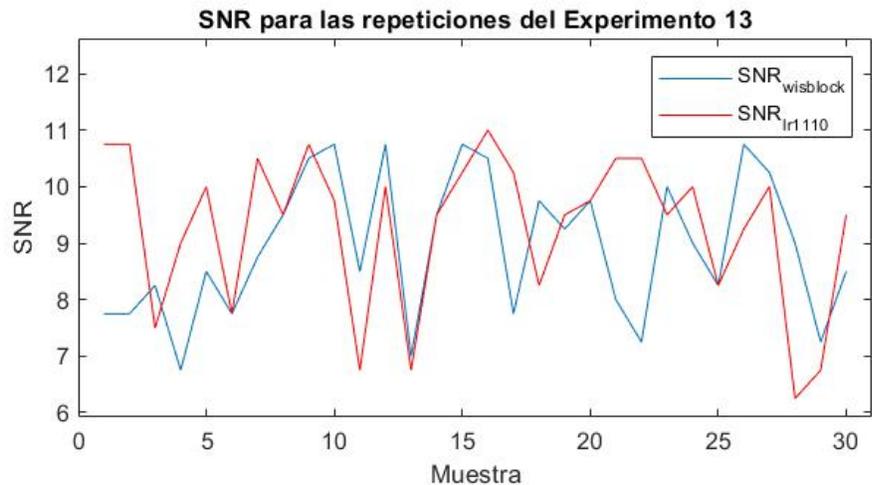


Figura 5.11: Gráfico del SNR medido para el transmisor del chip LR1110 (rojo) con el escaneo de Wi-Fi y para el transmisor del Wisblock (azul), en las repeticiones del experimento 13.

Al observar la Figuras 5.10 y la Figura 5.11, se puede observar que las variables de RSSI y SNR se comportan de manera similar para ambos dispositivos, pero se nota una clara diferencia en magnitud en el gráfico del RSSI. Esta diferencia de aproximadamente 7, donde los valores para el LR1110 están dentro del rango  $[-50, -41]$  y los valores del Wisblock están dentro del rango  $[-34, -28]$ , se le adjudica a una característica de los transmisores utilizados y no solo del chip en cuestión. Al ver la gráfica de SNR, los valores máximos son de 11 y 10.75 para el chip LR1110 y el Wisblock respectivamente, así como también presentan valores mínimos de  $-6.25$  y  $6.75$  respectivamente. Se concluye que las pruebas se realizaron en un ambiente similar ya que presentan valores similares de SNR y también muestran resultados superiores a 0, siendo el caso en que se logró superar el piso de ruido presente.

## 5.2.2. Resultados generales

A continuación, se muestran los resultados de todos los experimentos realizados en este Capítulo, mostrando gráficos de amplitud/desviación estándar versus experimento, presentes en las Figuras 5.12, 5.13, 5.14, 5.15, 5.16, 5.17, 5.18, 5.19, 5.20, 5.21, 5.22 y 5.23, y, donde la desviación estándar se muestra en forma de bloque y la amplitud es el promedio de todas las repeticiones realizadas para cada experimento, mientras que el eje de las abscisas corresponde a un identificador claro del escaneo RF realizado o del experimento en sí. En adición a lo anterior, se utilizan también gráficos de barras para ciertos resultados que no intentan mostrar ninguna tendencia.

### 5.2.2.1. Escenario 1

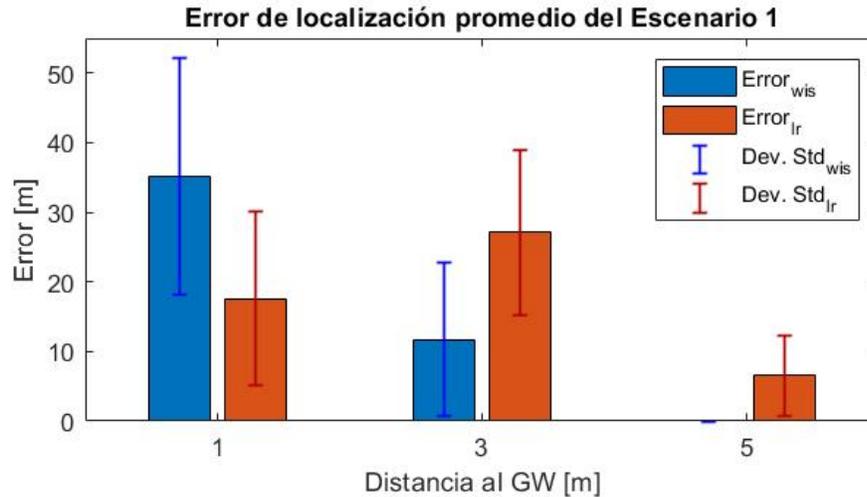


Figura 5.12: Gráfico del error calculado entre la posición de referencia y la posición encontrada por el chip LR1110 (naranja) con sus escaneos RF y el error de la distancia con la posición para el GPS del Wisblock (azul), en los experimentos para el Escenario 1.

Al observar los resultados del Escenario 1, se puede destacar el error de localización, el cual se muestra en la Figura 5.12, y que deja un detalle claro, el rastreo de GPS no funciona cuando se está dentro de un escenario indoor y que el rastreador este a una distancia mayor a 2 metros de una ventana, como es el caso del experimento realizado a 5 metros del gateway, donde no se lograron obtener registros de localización por parte de la solución basada en Wisblock. Por otra parte, el experimento que fue realizado a 1 metro del gateway permite observar un detalle: el chip LR1110 obtiene una localización más precisa que el rastreo GPS conectado al RAK4630, donde el escaneo Wi-Fi presenta un error de  $17.5945[m]$ , siendo este casi la mitad de los  $35.1396[m]$  de error que presentó el escaneo GPS del Wisblock. Incluso, se tiene una precisión aún mejor para el caso donde se el transmisor está a 5 metros del gateway, bajando a un error de  $6.5737[m]$  en promedio. Sin embargo, para el experimento que se realizó con el transmisor a tres metros del gateway, se dio la casualidad que los transmisores estaban en el punto más cercano a una ventana que diera al exterior, por lo que la precisión del GPS mejoró, mientras que la precisión del escaneo Wi-Fi empeoró, teniendo errores promedio de  $11.6865[m]$  y  $27.1258[m]$  respectivamente, siendo mejor el desempeño del transmisor con el

RAK4630. Lo anterior refuerza la idea de que el escaneo Wi-Fi es una herramienta muy buena para usos de indoor, incluso teniendo la hipótesis de que el rendimiento mejora cada vez más al ir “aislando” la cantidad de redes escaneadas y la distancia de estas.

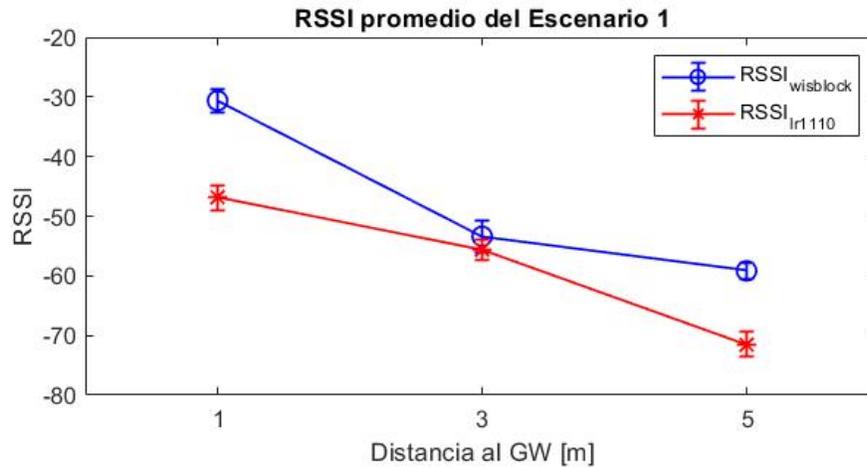


Figura 5.13: Gráfico del RSSI medido para el transmisor del chip LR1110 (naranja) con el escaneo de Wi-Fi y para el transmisor del Wisblock (azul), en los experimentos para el Escenario 1.

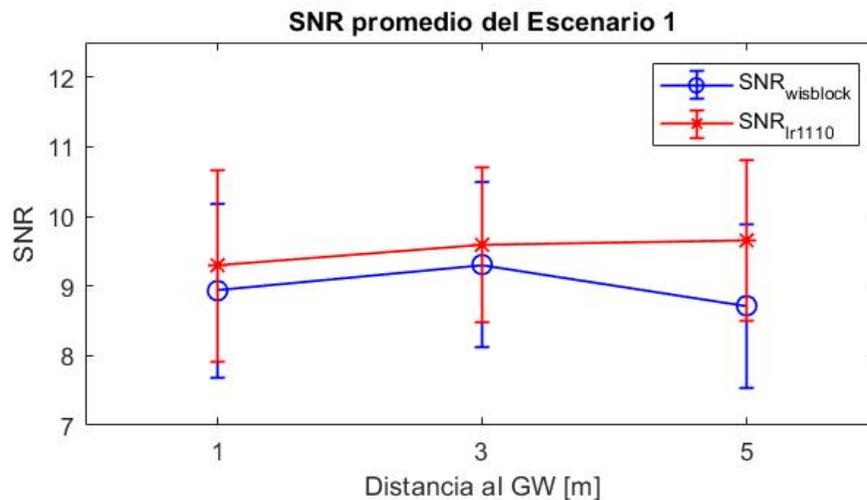


Figura 5.14: Gráfico del SNR medido para el transmisor del chip LR1110 (naranja) con el escaneo de Wi-Fi y para el transmisor del Wisblock (azul), en los experimentos para el Escenario 1.

Al analizar el desempeño en la red LoRaWAN, se puede observar la Figura 5.13 en donde se ve la clara tendencia de que al ir alejando el transmisor también sucede que el RSSI va disminuyendo, pero siempre con la curva del RAK4630 por encima de la curva del chip LR1110. Los valores obtenidos cuando el transmisor está lo más cerca del gateway son de  $-46.9333$  para el LR1110 y  $-30.6333$  para el RAK4630, valores que decaen hasta  $-71.4333$  y  $-59.1667$  respectivamente, donde el mejor resultado es un desempeño excelente y el peor caso presenta los valores normales para establecer una buena comunicación inalámbrica. En cuanto al SNR, presente en la Figura 5.14, se tiene un comportamiento “estable” entre los 8

y 10 de SNR, donde siempre la curva del LR1110 está sobre la curva del RAK4630, lo cual habla de una mejor capacidad para superar las barreras de ruido y tener un mejor índice ruido/señal.

### 5.2.2.2. Escenario 2

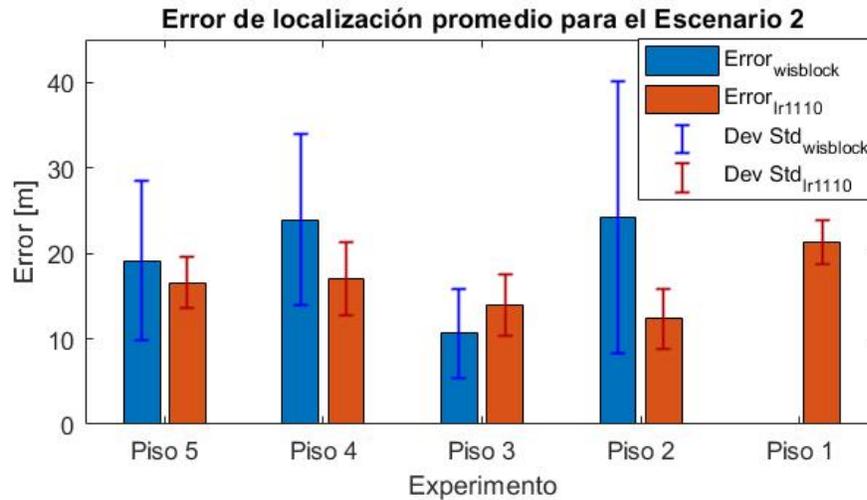


Figura 5.15: Gráfico del error calculado entre la posición de referencia y la posición encontrada por el chip LR1110 (naranja) con sus escaneos RF y el error de la posición para el GPS del Wisblock (azul), en los experimentos para el Escenario 2.

Para el caso del Escenario 2, primero que todo se debe mencionar la clara superioridad del chip LR1110 con respecto a la precisión de la localización, lo cual es muestra en la Figura 5.15. Una vez más se logra ver que, dentro de un escenario indoor, el escaneo Wi-Fi del chip LR1110 presenta un error menor al que presenta la localización por GSP del RAK4630, patrón que se repite en los 5 experimentos realizados en Escenario 2. Sin embargo, en el experimento realizado en el piso 2, sucedió que el escaneo Wi-Fi (LR1110) presentó un error de 13.941[m], mientras que el rastreo GPS solo presentó un error de 10.61[m], siendo este el único caso donde la solución basada en Wisblock tiene una mejor precisión que el chip LR1110 dentro de este escenario. Cabe resaltar que los experimentos realizados con el RAK4630 presentan, además, una desviación estándar mayor que los escaneos Wi-Fi, por lo que se cree que este detalle es una desventaja del Wisblock frente al chip LR1110, el cual presenta una mejor precisión y no tiene tanta dispersión en la localización encontrada. También se debe mencionar que dentro de este escenario no fue posible utilizar los escaneos GNSS del chip LR1110, donde se tuvo el peor caso al realizar el experimento del Piso 1, donde el GPS del RAK4630 tampoco fue capaz de encontrar localización. El Piso 1 no presenta ventanas cerca, por lo que era esperable que no se pudiera encontrar localización por parte del GPS, sin embargo sí logró encontrar una localización dentro de los otros pisos, lo cuales contaban con ventanas y se podrían considerar los límites del GPS ya que al alejarse 1 o 2 metros más, la localización se perdía completamente. Por otra parte, el escaneo Wi-Fi presentó un error de 21.2339[m], lo cual no es de los valores más malos, pero si es un valor aceptable dentro de un experimento en uno de los peores casos para ambos dispositivos en este experimento, un

escenario completamente de interior y que es susceptible a distintas redes Wi-Fi.

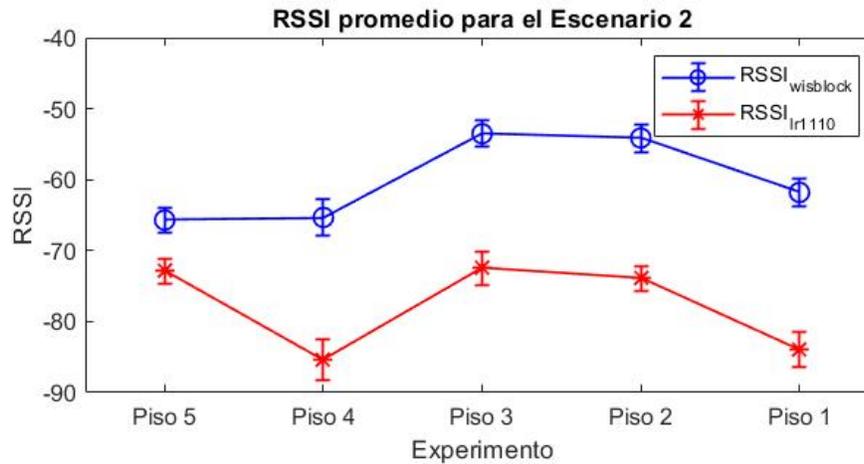


Figura 5.16: Gráfico del RSSI medido para el transmisor del chip LR1110 (naranja) con el escaneo de Wi-Fi y para el transmisor del Wisblock (azul), en los experimentos para el Escenario 2.

Tal como sucedió con los resultados del Capítulo 4, el RSSI, presente en la Figura 5.16, tiene un comportamiento algo extraño y que se le adjudica al *multipath* que se produce al tener varios caminos que puede recorrer la señal para alcanzar al gateway. Se observa un claro aumento de RSSI cuando se pasa desde el piso 4 al piso 3, resultado inesperado ya que se esperaría que esta métrica disminuya si es que se aleja el gateway. Es importante mencionar que este comportamiento sucede solo en ese cambio de piso, luego, y antes, el RSSI se comporta como se esperaría; desciende a medida que se va alejando el transmisor del gateway. También existe la característica de que en el gráfico del RSSI, la curva del Wisblock se mantiene siempre por encima de la curva del chip LR1110, con una diferencia de 20 aproximadamente, pero ambas curvas tienen una forma similar y logra dar cuenta de que el comportamiento mencionado anteriormente no es de un transmisor en específico y que es una característica ligada al escenario en donde se realizaron las pruebas.

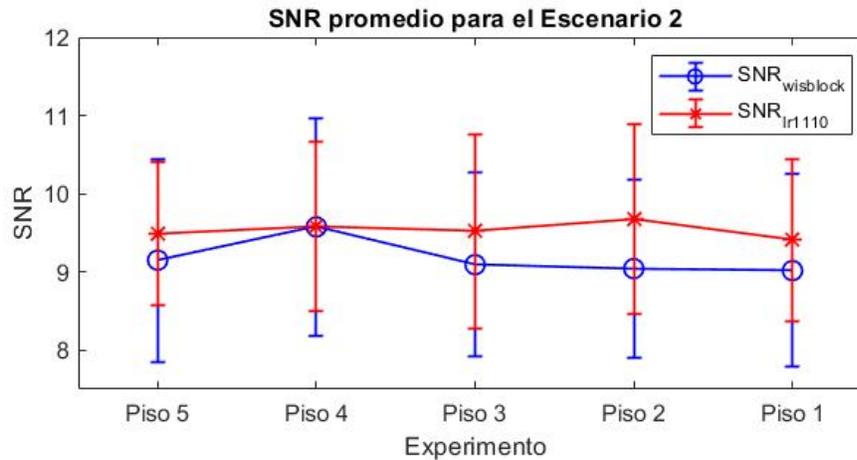


Figura 5.17: Gráfico del SNR medido para el transmisor del chip LR1110 (naranja) con el escaneo de Wi-Fi y para el transmisor del Wisblock (azul), en los experimentos para el Escenario 2.

En cuanto al SNR, al analizar la Figura 5.17 se logra verificar que el comportamiento de esta variable es cercano a ser constante, manteniéndose entre los valores de 9 y 10 para ambos dispositivos, permitiendo decir de que las proporciones de ruido/señal son similares para todos los experimentos realizados en este escenario y facilita la tarea de realizar experimentos comparables dentro de este escenario.

### 5.2.2.3. Escenario 3

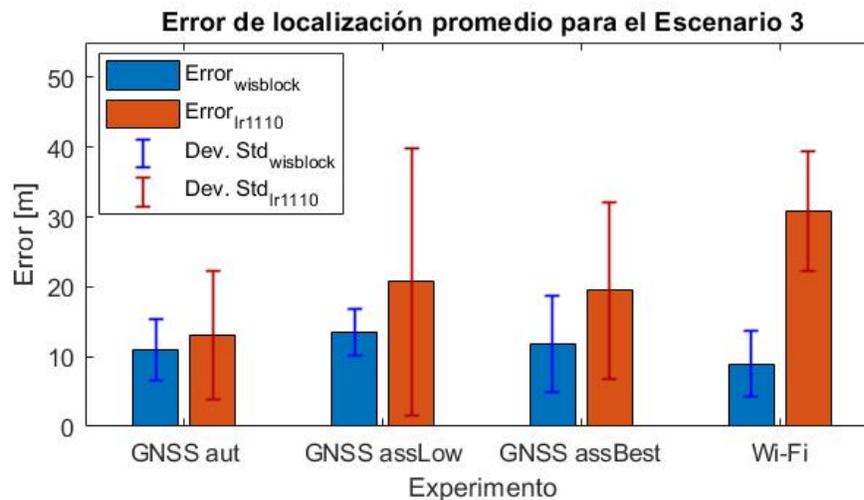


Figura 5.18: Gráfico del error calculado entre la posición de referencia y la posición encontrada por el chip LR1110 (naranja) con sus escaneos RF y el error de la posición para el GPS del Wisblock (azul), en los experimentos para el Escenario 3.

Al comenzar con los experimentos realizados en exterior, se tiene el caso del Escenario 3, donde la Figura 5.18 permite observar como la precisión del GPS, así como la de los escaneos GNSS, logra superar la precisión del escaneo Wi-Fi, por lo que el desempeño de aquellos

basados en uso de satélites esta influenciado por el porcentaje de visibilidad del cielo, donde en un caso de exterior se ve un desempeño muy superior a los escenarios anteriores. Es importante señalar que el escaneo GNSS autónomo presenta el menor error de los escaneos RF realizados en este escenario, con un valor de  $13.02[m]$ , siendo el escaneo GNSS con una implementación directa y sin la necesidad de realizar una actualización de datos o configuración previa, como es el caso para los escaneos GNSS asistidos. Para el caso del Wi-Fi, se tiene que es el experimento con mayor porcentaje de error y presenta un valor de  $30.7617[m]$ , lo cual se cree que se debe a la característica de exterior del Escenario 3 y que genera que el transmisor del chip LR1110 esté influenciado por una mayor cantidad de redes Wi-Fi, lo cual se puede observar al tener una diferencia de  $21.8114[m]$  entre los errores presentados por ambos dispositivos en ese mismo experimento. Por otra parte se tiene los casos de los escaneos GNSS asistidos de bajo consumo y con el mejor esfuerzo, los cuales tienen un error de  $20.7227[m]$  y  $19.482[m]$  respectivamente, los cuales presentan un desempeño similar al rastreo GPS. Sin embargo, ambos presentan un porcentaje de error más grande que la solución de Wisblock, en  $7.1646[m]$  y  $7.7148[m]$  respectivamente, teniendo un resultado similar en las comparaciones para ambos tipos de escaneos GNSS asistidos.

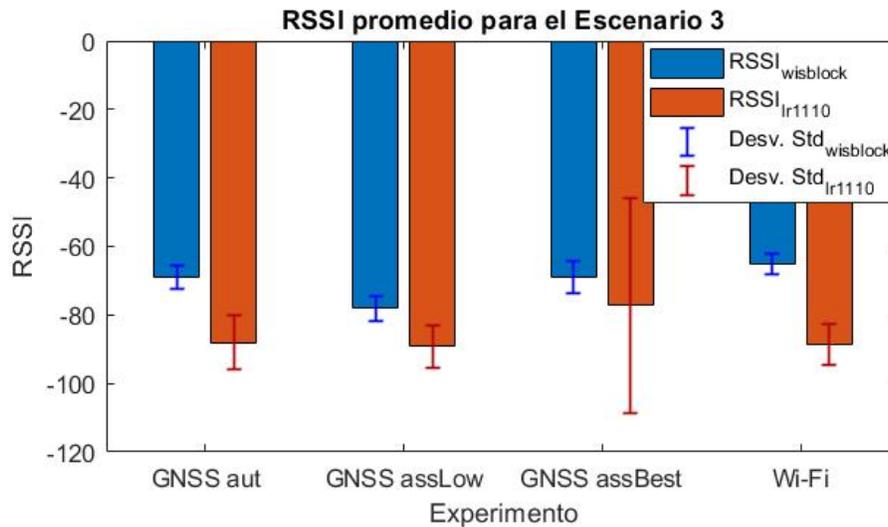


Figura 5.19: Gráfico del RSSI medido para el transmisor del chip LR1110 (naranja) con sus escaneos RF y para el transmisor del Wisblock (azul), en los experimentos para el Escenario 3.

Al analizar la Figura 5.19 se puede observar que los valores de RSSI presentados por el transmisor del chip LR1110 tienen una menor magnitud que los valores presentados por el transmisor del Wisblock, donde este patrón se repite para los 4 experimentos y presentan un resultado similar entre los experimentos. Es importante señalar que la prueba con el escaneo GNSS asistido, forzando el mejor esfuerzo, presenta una desviación estándar mayor que el resto de experimentos, sin embargo se cree que esto es un comportamiento que está influenciado por el ambiente más que por el dispositivo en sí, generando que el esfuerzo extra tenga mejores resultados y resultados similares, por lo que es esperable que la dispersión de los datos aumente a la vez que el promedio “mejora”. Para el caso del LR1110, el RSSI se mantiene dentro del rango de  $[-77.2333, -89.1667]$ , mientras que el RAK4630 tiene su RSSI dentro del rango  $[-65.2, -78.1333]$ , donde las diferencias no superan los  $24[dB]$  y ambos dispositivos se

mantienen dentro de los niveles buenos de RSSI para establecer una comunicación efectiva, además de tener en cuenta de que se están realizando las pruebas en exterior y con el gateway dentro el laboratorio.

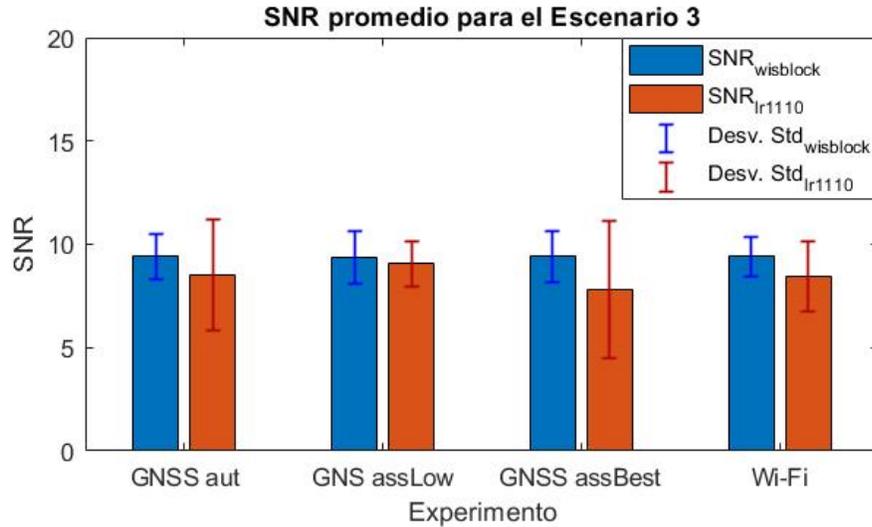


Figura 5.20: Gráfico del SNR medido para el transmisor del chip LR1110 (naranja) con sus escaneos RF y para el transmisor del Wisblock (azul), en los experimentos para el Escenario 3.

En cuanto al SNR, se puede decir que, según lo mostrado en la Figura 5.20, esta variable tiene un comportamiento similar para todos los experimentos y también para ambos transmisores utilizados, manteniéndose en todos los casos dentro del rango  $[7.825, 9.425]$ , donde la diferencia entre los dispositivos no supera los 1.58333 de SNR. Se corrobora la idea de que los experimentos tienen una relación ruido/señal similar y que permite que las pruebas sean comparables al establecer un escenario con las mismas características, sin mencionar que en todos los casos se logra superar la barrera de ruido presente en el ambiente.

#### 5.2.2.4. Escenario 4

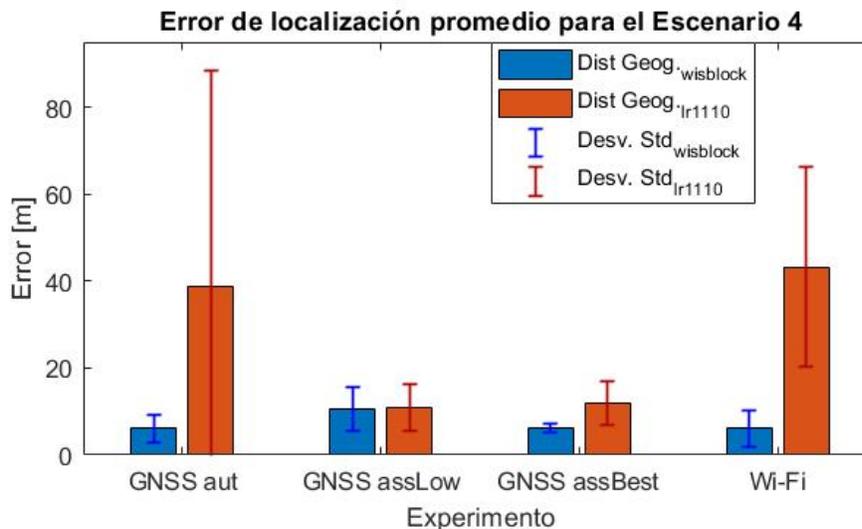


Figura 5.21: Gráfico del error calculado entre la posición de referencia y la posición encontrada por el chip LR1110 (naranja) con sus escaneos RF y el error de la posición para el GPS del Wisblock (azul), en los experimentos para el Escenario 4.

Finalmente, al analizar el escenario que se agregó para este capítulo, se tiene la Figura 5.21, en la cual se muestra la distancia (el error) con respecto a la posición de referencia para cada experimento. Lo primero que se observa es el mal desempeño, comparado con el resto, que tiene el escaneo de GNSS autónomo y el escaneo Wi-Fi, presentando un error de  $38.64855[m]$  y de  $43.2924[m]$  respectivamente. Se cree que para el caso del Wi-Fi, el escenario utilizado es desfavorable y se debe a la baja “protección” que se ofrece frente a redes muy lejanas, lo cual se cree que es el principal factor que afecta a la precisión de la localización de este tipo de escaneo. Sin embargo, para el escaneo GNSS autónomo se tiene un desempeño muy distinto al presentado por los otros escaneos GNSS o incluso por el escaneo GPS, el cual presenta un error de solo  $6.15862[m]$  dentro de la misma prueba. Este comportamiento del escaneo GNSS autónomo da la idea de que su precisión no depende únicamente de la visibilidad y se debe escoger alguno de los otros escaneos GNSS o incluso es mejor utilizar el rastreo GPS del RAK4630 para ciertos casos. Sin embargo es importante destacar que en la prueba del GNSS automático dentro del escenario 4, se logró captar una cantidad de satélites más variada y de una manera mucho más fácil, permitiendo que la utilización de esta herramienta fuera mucho más sencilla.

En cuanto a los escaneos GNSS asistidos, su desempeño es similar al presentado por el rastreo GPS del Wisblock, teniendo una diferencia de precisión, con respecto al GPS, de  $0.3099[m]$  para el GNSS asistido de bajo consumo y  $5.64424[m]$  para el GNSS asistido con el mejor esfuerzo, permitiendo tener una herramienta de localización en exterior que presenta buenos resultados y logra competir con otro tipo de solución presente en el mercado.

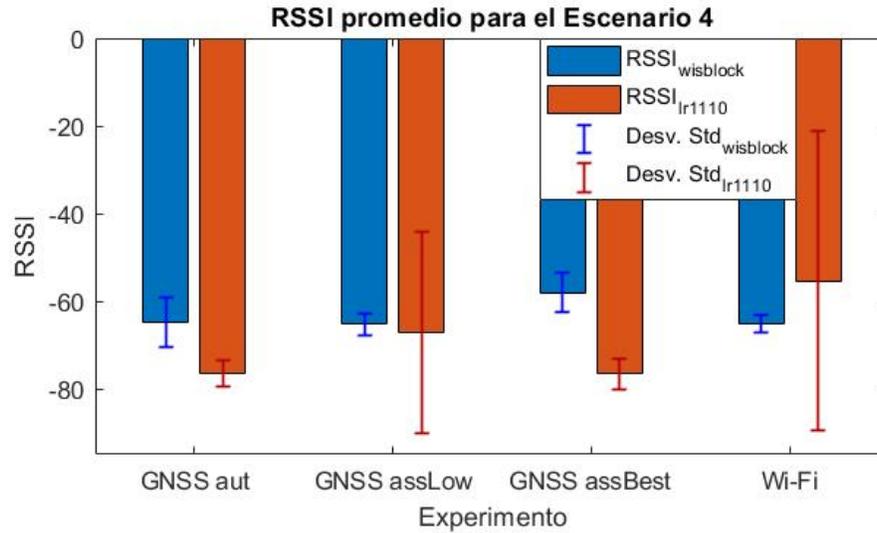


Figura 5.22: Gráfico del RSSI medido para el transmisor del chip LR1110 (naranja) con sus escaneos RF y para el transmisor del Wisblock (azul), en los experimentos para el Escenario 4.

Al igual que los experimentos anteriores, el RSSI, presentado en la Figura 5.22, tiene un comportamiento similar para todas las pruebas realizadas y se mantiene la tendencia de que la intensidad de señal del LR1110 es menor a la intensidad presentada por el transmisor basado en el Wisblock. Los rangos de RSSI son  $[-76.6, 755.3333]$  para el LoRa Edge™ y  $[-65.3, -58.0667]$  para el RAK4630, teniendo ambos dispositivos dentro de un rango aceptable de intensidad de señal para poder establecer una comunicación inalámbrica.

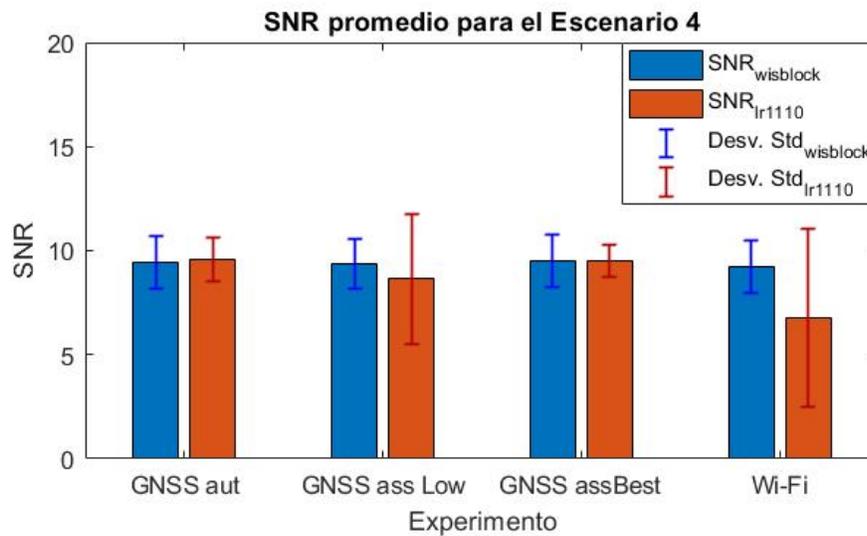


Figura 5.23: Gráfico del SNR medido para el transmisor del chip LR1110 (naranja) con sus escaneos RF y para el transmisor del Wisblock (azul), en los experimentos para el Escenario 4.

En cuanto al SNR, se puede observar que los resultados mostrados en la Figura 5.23 dan cuenta de que esta variable no cambia mucho su valor para los distintos experimentos y escaneos realizados, teniendo valores que se mantienen dentro del rango  $[6.78333, 9.59167]$ , para

ambos dispositivos, y no presentan una diferencia mayor a 2.43334, teniendo un escenario con condiciones similares en cuanto a la relación de ruido/señal que presentan las pruebas realizadas y que permite su comparación, done en todos los casos se tiene un valor de ruido/señal que supera la barrera de ruido presente en este ambiente.

### 5.3. Análisis de consumo

Al analizar el *datasheet* del chip LR1110, se puede observar que el costo energético al realizar transmisiones, es de  $28[mA]$ , siendo este un valor bajo frente los  $50[mA]$  promedio que presenta, por ejemplo, el módulo LoRa **DRF1278F**, el cual está basado en el chip SX1278.

Por otro lado, es importante mencionar que

- Al realizar un escaneo Wi-Fi se utiliza entre  $0.5[\mu Wh]$  y  $0.7[\mu Wh]$ .
- Al realizar un escaneo GNSS autónomo se utiliza entre  $4.4[\mu Wh]$  y  $6.9[\mu Wh]$ .
- Al realizar un escaneo GNSS asistido se utiliza entre  $6.8[\mu Wh]$  y  $10.2[\mu Wh]$ .

Con lo que se tiene un consumo bajo por parte de los 3 tipos de escaneos RF, ol cual es una mejora al consumo general de otras opciones de localización, las cuales tienen un consumo convencional de unos  $20[mA]$  a  $26[mA]$ ; valores mostrados en [30].

Queda claro que el consumo es menor, tanto para transmitir como para realizar una búsqueda de localización. Por este es que, dependiendo del uso, una solución basada en el chip LR1110 puede ser la mejor opción en cuanto a uso energético, teniendo casos en los cuales se puede llegar a ahorrar un 50% de energía.

# Capítulo 6

## Conclusiones

En esta memoria se realizó un trabajo de evaluación y comparación de la gama Lora Edge, junto con una solución basada en la gama de chips SX1262. El objetivo fue identificar las mejoras/beneficios que tiene la nueva versión de transmisores LoRa, basados en el chip LR1110. Todo esto para proporcionar un trabajo experimental, el cual considere una evaluación en escenarios de interiores y exteriores. Para cumplir con la tarea, fue necesario realizar las configuraciones necesarias para armar la red LoRaWAN, tanto de manera física como de manera virtual, realizando las configuraciones, registros y autenticaciones pertinentes para utilizar los servicios y dispositivos proporcionados por Semtech. Una vez armada la red se continuo con el diseño de los experimentos realizados, con un total de 28 experimentos y 4 escenarios distintos. Finalmente se analizaron los resultados obtenidos y se lograron rescatar los detalles más importantes de cada escaneo RF evaluado, junto con su respectivo desempeño frente a una solución que utiliza GPS.

Dependiendo del caso en que se esté considerando utilizar una solución basada en módulos LoRa, el chip LR1110 es una muy buena opción si es que también se pretende utilizar una herramienta de localización. Cabe mencionar que se debe conocer muy bien el ambiente o las características de este, ya que este detalle pareciera ser fundamental para decidir qué tipo de escaneo RF se va a utilizar. Esto se debe a la variedad de escaneos disponibles; pudiendo escoger entre escaneos para escenarios indoor y outdoor, donde, además, se tienen tres variantes de escaneo GNSS para los casos outdoor. Esto conlleva que la elección sea en base a: el porcentaje de visión disponible, el consumo energético que se quiera establecer o a la simpleza que se le quiera dar a la localización, donde la cualidad del GNSS autónomo de no necesita un almacenaje de satélites cercanos permite simplificar las cosas, considerando que, también, no se necesita una actualización periódica de este. Esta característica versátil de la gama LoRa Edge™, permite tener un localizador que es utilizable en una gran cantidad de casos, tales como: smart farming, smart city, smart house, localización de trasportes, rastreo de productos, drones, usos industriales, etc.

Es importante destacar que, la precisión de la localización que se obtuvo por medio de escaneos RF, obtenida en los experimentos de esta memoria, tiene un desempeño muy bueno en escenarios de interior y permite utilizar una solución que no depende en ningún grado de el acceso al exterior; solo se necesita un mínimo de redes Wi-Fi para encontrar posición y un *access point* para enviar los resultados. También, se observo que la precisión es similar, el algunos casos, a la que se logra obtener mediante otras soluciones para exteriores, las cuales no tienen la capacidad de funcionar dentro de espacios cerrados. Por otra parte, el uso de

escaneos RF en el exterior, principalmente los escenarios con vista del cielo despejada, no logra superar la precisión que presenta una solución normal basada en GSP, como es el caso del Wisblock. Sin embargo, se puede observar que en los resultados de las comparaciones, si bien es cierto se muestra que la precisión puede ser algo peor, el chip LR1110 es una buena solución de localización para el exterior y que, además, presenta las características de ser una tecnología que logra disminuir el consumo energético con respecto los métodos de rastreo GPS que se utilizan normalmente. Se entiende que, el chip LR1110 es una tecnología especializada en redes LPWAN y cuenta con una versatilidad para distintos escenarios, demostrando una precisión más que aceptable.

Se cree que es la mejor forma de utilizar una tecnología basada en LoRa Edge™ es realizando un trabajo similar al realizado en este proyecto, donde, primero, se realizó una serie de pruebas para verificar el comportamiento de cada escaneo RF en distintos escenarios, lo que permitió entender como se comporta la localización y ayuda para decidir qué escaneo emplear. Se cree que con esta herramienta se puede cubrir cualquier caso que este ambientado en la ciudad, o en sus cercanías, ya que se cuenta con un escaneo Wi-Fi para casos en que haya mala visibilidad o también se cuenta con las soluciones de GNSS si se está en un escenario con mucha presencia de “ruido” para el escaneo Wi-Fi. Es posible, incluso, que esta herramienta pueda ser utilizada para rastrear objetivos en movimiento, pero se debe tener siempre en cuenta los alcances de la red LoRaWAN montada y la visibilidad del cielo, ya que se cree que para este tipo de rastreo es mejor utilizar los escaneos GNSS y no depender de la presencia de redes Wi-Fi o de que sea un ambiente aislado de redes muy lejanas. Cabe mencionar que, una vez que esté liberada la nueva modulación LR-FHSS, los usos de la gama LoRa Edge™ van a poder escalar en una gran medida, dejando que esta herramienta sea aplicable a distancias más lejanas y utilizar una mayor cantidad de dispositivos conectados, de manera simultanea, a la misma red.

El trabajo realizado se basó en evaluar y comparar los dispositivos empleados, lo cual se logró realizar de buena manera al diseñar experimentos enfocados en distintos ambientes. La metodología y los escenarios escogidos favorecieron el poder entender como funcionan los escaneos RF del chip LR1110, para, posteriormente, comparar sus resultados con otra solución, como el módulo RAK4630. Al final del trabajo, se lograron los objetivos establecidos en un principio, además de lograr anteponerse ante las dificultades que se presentaron durante el proceso.

Es importante mencionar que el aporte del trabajo de memoria se basa en entregar una evaluación y comparación detallada de una gama específica de transmisores, los cuales pueden llegar a formar parte de proyectos muy ambiciosos dentro del mundo industrial, por lo que, es necesario disponer con alguna fuente de información que permita tomar la mejor decisión al momento de escoger la tecnología a emplear. Para el caso específico del LoRa Edge™, los trabajos son escasos y no resaltan un detalle que se considera importante: el uso de localización en indoor y con una buena precisión. Esto provoca un vacío en la información disponible sobre una tecnología que, probablemente, sea un foco de interés luego de que sea liberado el LR-FHSS, ya que se espera que el chip LR1110 sea uno de los módulos capaces de utilizar dicha modulación.

Se cree pertinente que, para trabajos futuros, se debe considerar el utilizar un escenario lo más controlado posible, dejando completamente aislado el transmisor de señales externas y,

posteriormente, controlar la cantidad de redes, o, incluso, las señales cercanas. Las variaciones pueden ser en: la distancia, el tipo de red Wi-Fi, el tipo de escaneo, etc., logrando estudiar bien como funciona la precisión del escaneo Wi-Fi y los demás escaneos GNSS. Dentro de este trabajo, se llegó a plantear la hipótesis de que la precisión varía con respecto la cantidad de redes Wi-Fi “cercanas”, donde una mayor cantidad mejora la precisión. Sin embargo, aunque se encuentre una gran cantidad de redes, si son redes que están lejos, se provoca un aumento del error cuando se realiza la triangulación de la localización. La única forma de poder comprobar dicha hipótesis es con un experimento que considere lo descrito anteriormente: controlar el experimento en, por ejemplo, un laboratorio especializado. También, se debe hacer un estudio exhaustivo del consumo energético del chip LR1110 ya que hay muchas variables a considerar y es importante tener un estudio especializado para poder dar un veredicto o evaluación confiable, el cual no puede estar influenciado por agentes externos o malas tecnologías de medición.

# Bibliografía

- [1] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne, “Understanding the limits of lorawan,” *IEEE Communications Magazine*, vol. 55, no. 9, pp. 34–40, 2017.
- [2] N. P. Manchev, K. K. Angelov, P. G. Kogias, and S. M. Sadinov, “Development of multichannel lorawan gateway for educational applications in low-power wireless communications,” in *2019 IEEE XXVIII International Scientific Conference Electronics (ET)*, pp. 1–4, 2019.
- [3] A. Xanthopoulos, A. Valkanis, G. Beletsioti, G. I. Papadimitriou, and P. Nicopolitidis, “On the use of backoff algorithms in slotted aloha lorawan networks,” in *2020 International Conference on Computer, Information and Telecommunication Systems (CITS)*, pp. 1–4, 2020.
- [4] G. Boquet, P. Tuset-Peiró, F. Adelantado, T. Watteyne, and X. Vilajosana, “Lr-fhss: Overview and performance analysis,” *IEEE Communications Magazine*, vol. 59, no. 3, pp. 30–36, 2021.
- [5] A. Farhad, D.-H. Kim, P. Sthapit, and J.-Y. Pyun, “Interference-aware spreading factor assignment scheme for the massive lorawan network,” in *2019 International Conference on Electronics, Information, and Communication (ICEIC)*, pp. 1–2, 2019.
- [6] Semtech, “Lora edge™.” <https://www.semtech.com/products/wireless-rf/lora-edge>, 2021. Información general sobre LoRa Edge™.
- [7] M. Ballerini, T. Polonelli, D. Brunelli, M. Magno, and L. Benini, “Experimental evaluation on nb-iot and lorawan for industrial and iot applications,” in *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, vol. 1, pp. 1729–1732, 2019.
- [8] M. S. Luka Mustafa, “Lr1110 chip: one solution for lora and gnss tracking.” <https://www.irnas.eu/lr1110-chip-one-solution-for-lora-and-gnss-tracking/>, 2020. Blog Irnas.
- [9] L. Alliance®, “What is lorawan®.” [https://lora-alliance.org/resource\\_hub/what-is-lorawan/](https://lora-alliance.org/resource_hub/what-is-lorawan/), Nov 2015. “A technical overview of LoRa® and LoRaWAN™”.
- [10] Semtech, “What are lora® and lorawan®?” <https://lora-developers.semtech.com/documentation/tech-papers-and-guides/lora-and-lorawan/>, Accedido en noviembre 2021. Especificaciones LoRa® y LoRaWAN®.
- [11] R. S. C. B. Constanza Hojas, Alejandra Svriz, “Informe de la situación actual de los requerimientos de transmisión de datos y la estimación de la demanda prospectiva de consumo de datos para zonas agrícolas. licitación id 6606-25-lq16.” [https://www.subtel.gob.cl/wp-content/uploads/2017/10/02\\_InformeII\\_Demanda\\_Agro\\_Industrias.pdf](https://www.subtel.gob.cl/wp-content/uploads/2017/10/02_InformeII_Demanda_Agro_Industrias.pdf), Marzo 2017. Informe II.

- [12] T. Lu, “Lorawan protocol expands network capacity with new long range – frequency hopping spread spectrum technology.” <https://blog.semtech.com/lorawan-protocol-expands-network-capacity-with-new-long-range-frequency-hopping-spread-spectrum-technology>, Noviembre 2020. Presentacion LR-FHSS.
- [13] L. Alliance®, “Rp2-1.0.2 lorawan® regional parameters.” [https://lora-alliance.org/resource\\_hub/rp2-102-lorawan-regional-parameters/](https://lora-alliance.org/resource_hub/rp2-102-lorawan-regional-parameters/), Octubre 2020. Authored by the LoRa Alliance Technical Committee Regional Parameters Workgroup.
- [14] Q. Liu, Y. Mu, J. Zhao, J. Feng, and B. Wang, “Characterizing packet loss in city-scale lorawan deployment: Analysis and implications,” in *2020 IFIP Networking Conference (Networking)*, pp. 704–712, 2020.
- [15] S.-Y. Wang, J.-J. Zou, Y.-R. Chen, C.-C. Hsu, Y.-H. Cheng, and C.-H. Chang, “Long-term performance studies of a lorawan-based pm2.5 application on campus,” in *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, pp. 1–5, 2018.
- [16] L. Maziero, T. B. Marchesan, C. H. Barriquello, D. P. Bernardon, F. G. Carlotto, F. G. Reck, W. D. Vizzotto, and F. V. Garcia, “Monitoring of electric parameters in the federal university of santa maria using lorawan technology,” in *2019 IEEE PES Innovative Smart Grid Technologies Conference - Latin America (ISGT Latin America)*, pp. 1–6, 2019.
- [17] R. Muppala, A. Navnit, S. Poondla, and A. M. Hussain, “Investigation of indoor lorawan signal propagation for real-world applications,” in *2021 6th International Conference for Convergence in Technology (I2CT)*, pp. 1–5, 2021.
- [18] K. Mikhaylov and A. Pouttu, “On spatial diversity for lorawan: Experimental evaluation of performance of a dual-gateway network with and without downlink,” in *2019 11th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, pp. 1–5, 2019.
- [19] S. Ruepp, A. C. Mateo, K. Mateusz Malarski, J. Thrane, and M. N. Petersen, “Internet of things connectivity in deep-indoor environments,” in *2018 9th International Conference on the Network of the Future (NOF)*, pp. 96–100, 2018.
- [20] N. Vatcharatiansakul, P. Tuwanut, and C. Pornavalai, “Experimental performance evaluation of lorawan: A case study in bangkok,” in *2017 14th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, pp. 1–4, 2017.
- [21] M. M. Erbati, G. Schiele, and G. Batke, “Analysis of lorawan technology in an outdoor and an indoor scenario in duisburg-germany,” in *2018 3rd International Conference on Computer and Communication Systems (ICCCS)*, pp. 273–277, 2018.
- [22] R. El Chall, S. Lahoud, and M. El Helou, “Lorawan network: Radio propagation models and performance evaluation in various environments in lebanon,” *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2366–2378, 2019.
- [23] G. DoĀan, G. Yildirim, and Y. Tatar, “Empirical observations on lora performance for different environments,” in *2019 International Conference on Applied Automation and Industrial Diagnostics (ICAAID)*, vol. 1, pp. 1–6, 2019.
- [24] M. I. Muzammir, H. Z. Abidin, S. A. C. Abdullah, and F. H. K. Zaman, “Performance analysis of lorawan for indoor application,” in *2019 IEEE 9th Symposium on Computer Applications Industrial Electronics (ISCAIE)*, pp. 156–159, 2019.

- [25] P. Neumann, J. Montavont, and T. Noël, “Indoor deployment of low-power wide area networks (lpwan): A lorawan case study,” in *2016 IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pp. 1–8, 2016.
- [26] E. D. Ayele, C. Hakkenberg, J. P. Meijers, K. Zhang, N. Meratnia, and P. J. M. Havinga, “Performance analysis of lora radio for an indoor iot applications,” in *2017 International Conference on Internet of Things for the Global Community (IoTGC)*, pp. 1–8, 2017.
- [27] F. U. Khan, M. Awais, M. B. Rasheed, B. Masood, and Y. Ghadi, “A comparison of wireless standards in iot for indoor localization using lopy,” *IEEE Access*, vol. 9, pp. 65925–65933, 2021.
- [28] A. M. Yousuf, E. M. Rochester, and M. Ghaderi, “A low-cost lorawan testbed for iot: Implementation and measurements,” in *2018 IEEE 4th World Forum on Internet of Things (WF-IoT)*, pp. 361–366, 2018.
- [29] Semtech, “Understanding lora basics™ modem-e.” <https://lora-developers.semtech.com/documentation/tech-papers-and-guides/understanding-lora-basics-modem-e>, Accedido en noviembre 2021. Especificaciones LoRa Basics™ Modem-E.
- [30] Semtech, “Lora edge™: A revolution in lpwan technology.” <https://tech-journal.semtech.com/lora-edge-a-revolution-in-lpwan-technology>, Mar 23, 2020. Revisión general LoRa Edge.

# Anexos

## Anexo A

### Configuraciones de la red LoRaWAN

En esta sección se dejará detallado el procedimiento que se siguió para la configuración de los dispositivos y luego las configuraciones necesarias para utilizarlos en una red LoRaWAN, la cual estaba basada en el Development Kit, un gateway y los servicios proporcionados por el *Network Server* de Semtech y el portal de *LoRa Cloud™*.

Para las configuraciones realizadas, se utilizó el laboratorio “*LR1110 Development Kit Lab*” como base para realizar las instalaciones de software y actualizaciones necesarias para utilizar los dispositivos. También, se logró realizar las configuraciones para poder comunicar los dispositivos entre sí y, el gateway con la nube y servidor de LoRaCLOUD/Semtech. Para todo el trabajo realizado se siguieron los siguientes pasos:

1. Montar las partes del Gateway.
2. Configurar el Gateway.
3. Registrar y asegurar el Gateway con el *Semtech Network Server*.
4. Armar el transmisor *LoRa Edge™ LR1110 Tracking Device*.
5. Instalar el software del núcleo y del chip LR1110.
6. Reclamar el dispositivo en el portal de *LoRa Cloud™*.
7. Crear un propietario de dispositivo y obtener el Token de acceso.
8. Actualizar el almacenamiento de GNSS en el chip LR1110.
9. Agregar el *Development Kit* en el *Semtech Network Server* y conexión con los servicios del *LoRa Cloud™*.
10. Pruebas de localización del *Development Kit*.
11. Visualización mediante *ThingsBoard*.

## A.1. Montar las partes del Gateway

Como se tuvo que adquirir el gateway por partes, fue necesario esperar a que llegaran todos los pedidos antes de comenzar con las configuraciones. Es importante mencionar que es necesario armar bien el gateway, es decir asegurar bien las piezas ya que ante cualquier movimiento fuerte se podría desarmar parcialmente o completamente el gateway. El gateway se muestra en la Figura A.1.



Figura A.1: Gateway montado.

## A.2. Configurar el Gateway

Al momento de configurar el gateway, fue necesario instalar un software en la Raspberry que se concentrara en hacer más amigable las configuraciones al proporcionar un menú interactivo con el cual “ingresar” al gateway y realizar las integraciones de *APIs* de manera más directa. Para este trabajo se utilizó *ChirpStack Gateway OS*, cuya *imagen* para descargar esta disponible en su página oficial. En este trabajo se utilizó la imagen *chirpstack-gateway-os-base-raspberrypi3-20210510105535.rootfs.wic.gz*. Para grabar la imagen era necesario contar con una micro SD de 16GB, además de necesitar el adaptador de esta y un lector de tarjetas con conexión USB, para poder leer dicha micro SD en el computador. Es importante mencionar que antes de utilizar la tarjeta micro SD fue necesario realizar un formateo de esta. Para este trabajo se utilizó los dispositivos que se muestran en la Figura A.2



Figura A.2: Partes para utilizar la micro SD.

Luego, debido a que se requería grabar una imagen, se tuvo que instalar un programa que pudiera realizar esta tarea, que para el caso de este trabajo se utilizó *balenaEtcher*, que corresponde a un grabador de imágenes OS y esta disponible en su página oficial. El grabado de la imagen en la micro SD fue bastante directo con dicho programa y se muestra en la Figura A.3.

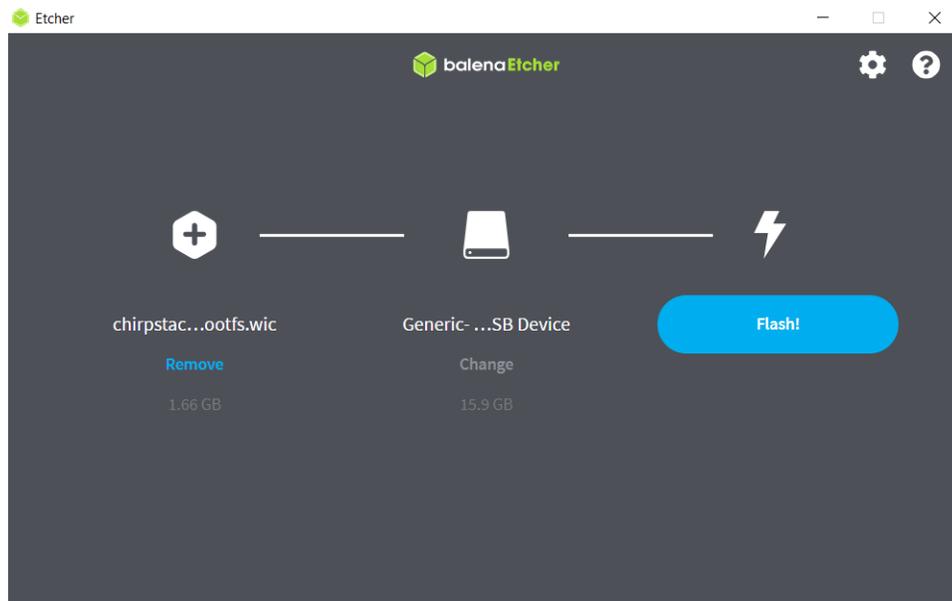


Figura A.3: Flasheo de la imagen para la Raspberry en la tarjeta de 16GB, mediante baletaEtcher.

Una vez grabada la imagen en la micro SD solo faltaba configurar el gateway, para lo que era necesario primero insertar la tarjeta en la Raspberry. El espacio para insertar la tarjeta micro SD se muestra en la Figura A.4



Figura A.4: SD insertada en la parte posterior de la Raspberry.

Al energizar el gateway, la configuración predeterminada que tenía el *software* previamente instalado era la de un Access Point con el nombre de *ChirpStackAP*. Debido a que las configuraciones iban a ser realizadas mediante el protocolo SSH, no era necesario contar con más accesorios para interactuar con la Raspberry, tales como mouse, teclado y pantalla. Sin embargo, fue necesario verificar que la computadora contaba con la instalación del SSH. Una vez energizado el gateway, se debió acceder al access point de Chirp Stack y utilizar la contraseña *ChirpStackAP*, con lo que se logró establecer una conexión mediante Wi-Fi para posteriormente poder realizar una comunicación mediante SSH. En esta parte se utilizó el *Windows PowerShell* y se ejecutó como administrador. La ip a la cual se debía conectar era la que tenía el gateway por defecto y el comando completo era `ssh admin@192.168.0.1`, luego, tras aceptar la conexión con el host desconocido, se ingresó la contraseña por defecto *admin*. Una vez conectado con el gateway, se debió el siguiente comando `sudo gateway-config` y utilizar la contraseña por defecto, la cual es siempre la misma, *admin*. El ingreso se muestra en la Figura A.5.

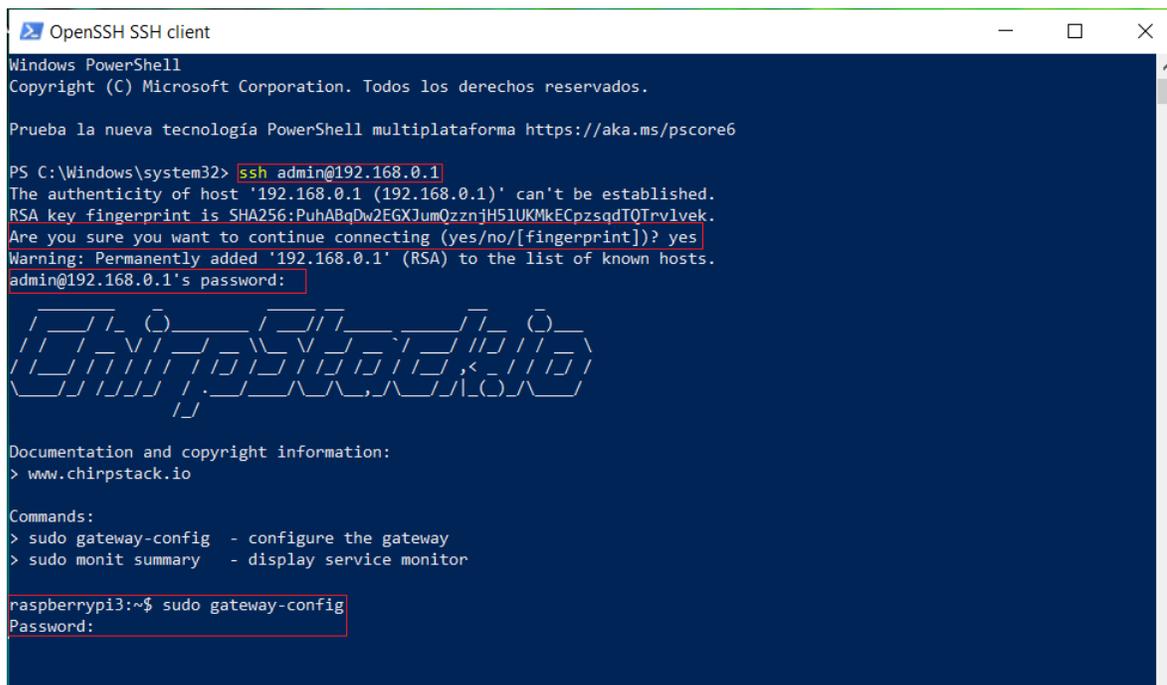


Figura A.5: Ingresar al gateway utilizando el AP de este mismo y por medio de SSH.

Al ingresar el comando anterior, se desplegó el menú de configuraciones del gateway. Primero se debe configurar la red Wi-Fi que se utilizará para conectarse a internet, ya que el Access Point del gateway solo sirve para conectarse a este mismo. Se seleccionó la opción **Configure WIFI** y posteriormente se ingresaron los parámetros de la red a utilizar, tales como el nombre del AP y la contraseña de este. Es importante señalar que de debe tener mucho cuidado al momento de ingresar los parámetros solicitados ya que si hay un error, se debe formatear la tarjeta SD y grabar nuevamente la imagen. La opción para configurar el Wi-Fi se muestra en la Figura A.6.

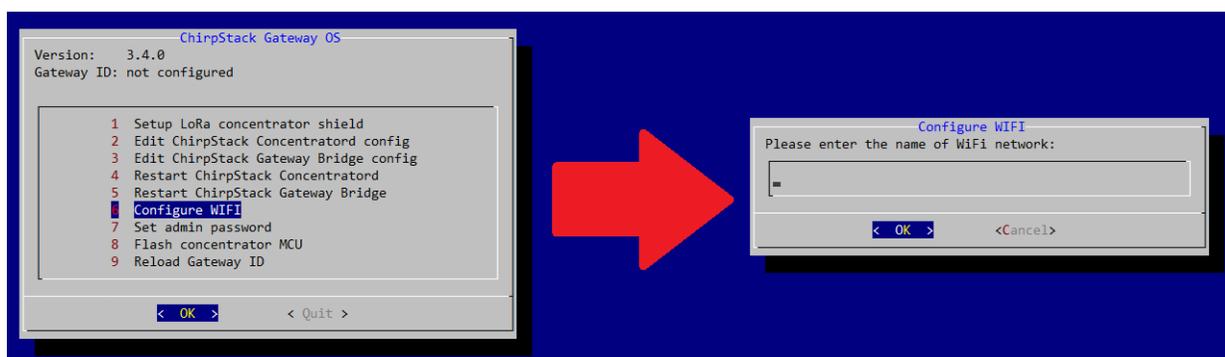
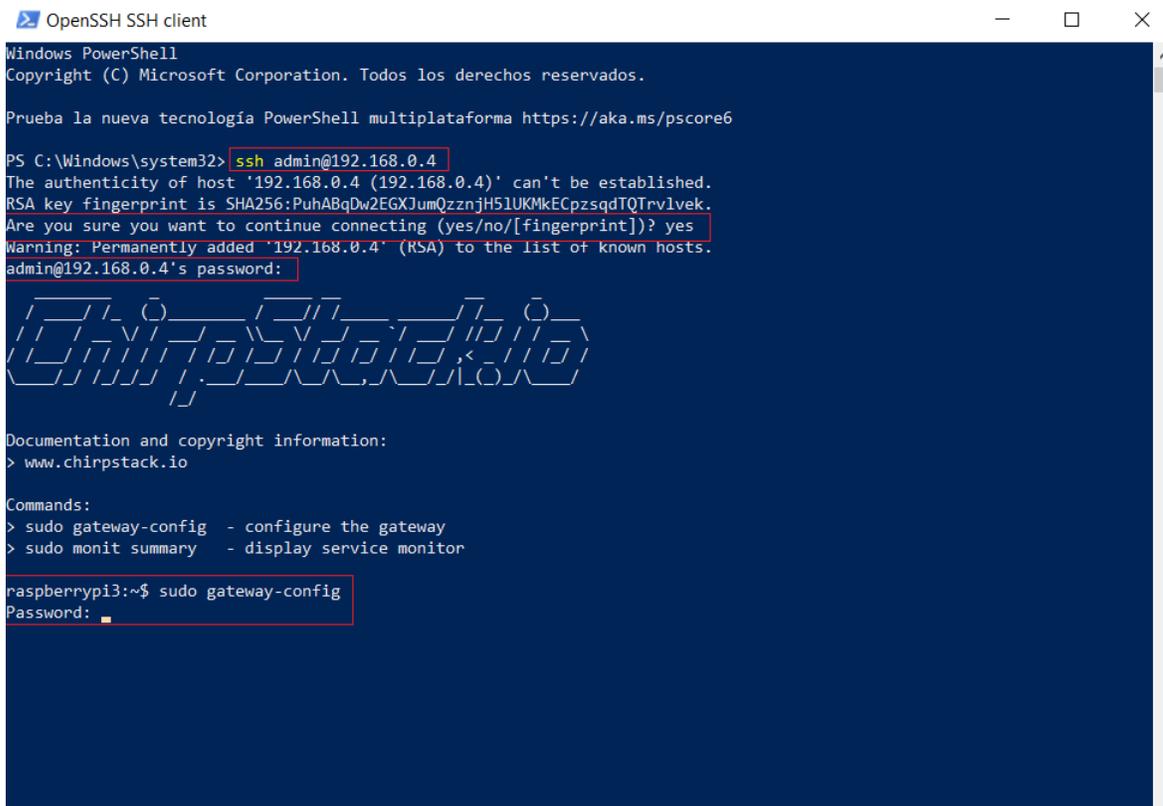


Figura A.6: Configuración del Wi-Fi a utilizar para conexión a internet.

Una vez configurada la red Wi-Fi a utilizar, fue necesario reiniciar el gateway manualmente para poder aplicar los cambios. Desde este punto en adelante fue necesario que el gateway estuviera conectado a la misma red que el computador que se utilizaría para realizar las configuraciones mediante SSH. Como se cambió de red, también se modificó la ip del gateway, por lo que se utilizó un programa para poder identificar la nueva ip, para este trabajo se utilizó

la aplicación *Fing Desktop* la cual esta disponible en su página oficial. Una vez identificada la ip, fue necesario ingresar al gateway por medio de SSH, de la misma forma que se realizó para configurar el Wifi, solo que se debía utilizar la nueva ip. Posteriormente se ingresó al menú de configuraciones con el comando `sudo gateway-config`. El ingreso del comando se muestra en la Figura A.7.



```
OpenSSH SSH client
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Windows\system32> ssh admin@192.168.0.4
The authenticity of host '192.168.0.4 (192.168.0.4)' can't be established.
RSA key fingerprint is SHA256:PuhABqDw2EGXJumQzznjH51UKMkECpzsqdTQTrvlvek.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.0.4' (RSA) to the list of known hosts.
admin@192.168.0.4's password:

Documentation and copyright information:
> www.chirpstack.io

Commands:
> sudo gateway-config - configure the gateway
> sudo monit summary - display service monitor

raspberrypi3:~$ sudo gateway-config
Password: 
```

Figura A.7: Ingresar al gateway utilizando el AP de una red con conexión a internet y por medio de SSH.

El siguiente paso era seleccionar los dispositivos que se utilizaron el gateway, que para este trabajo se obtuvieron por separado y se utiliza el concentrador *RAK 2247*. En el menú de configuraciones se tuvo que seleccionar la opción **Setup LoRa concentrator shield**. Luego, se desplegaban todas las opciones para concentrador que se podían escoger, se seleccionó la opción **Pi Supply - LoRa Gateway HAT**, que corresponde al gateway armado para este trabajo. Finalmente, se debía seleccionar la frecuencia de operación, que en este caso fue **US915**, y, posteriormente, seleccionar los canales a utilizar, que para este trabajo se seleccionó la opción **Channels 8 - 15 +65**. Al terminar las configuraciones del concentrador, se desplegó el mensaje de que la configuración de canales fue exitosa y que se va a reiniciar el gateway. Las opciones seleccionadas para la configuración anterior se muestran en la Figura A.8.

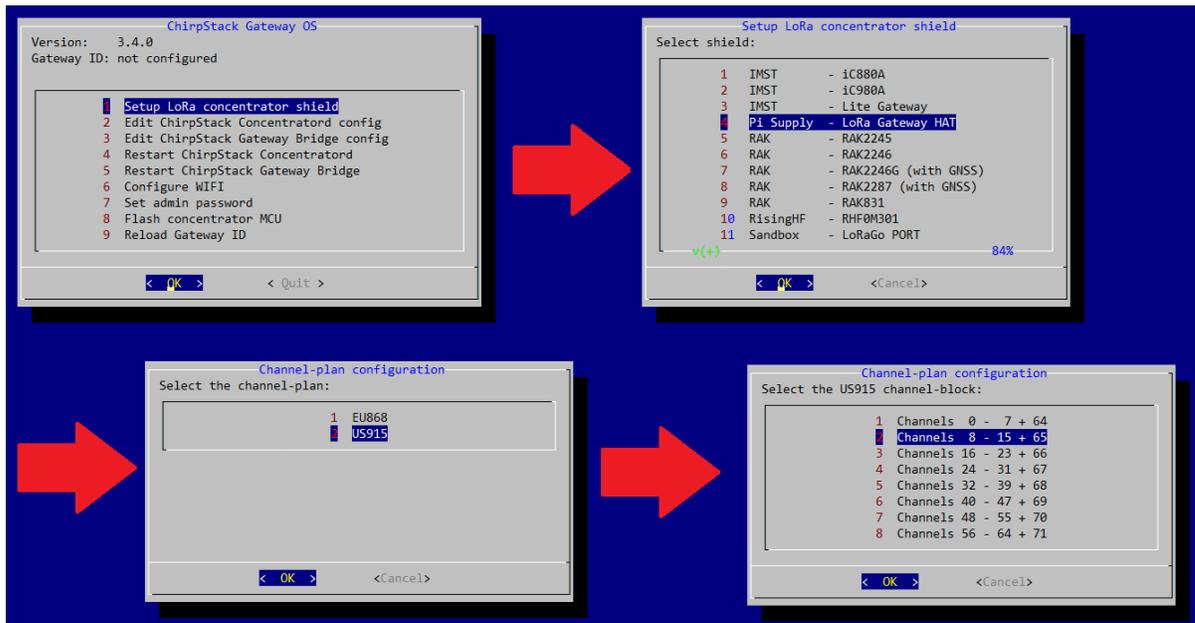


Figura A.8: Configuraciones de concentrador a utilizar.

Con lo anterior ya debería haber estado lista la configuración inicial del gateway, sin embargo se encontró el problema de que no se estaba reconociendo el concentrador *RAK2247*. Luego de investigar y realizar algunas pruebas, se logró descubrir que el problema se debía a la configuración del *reset\_pin*, el cual era la conexión entre la Raspberry y el concentrador *RAK 2247* y se utilizaba para poder reiniciar este último. Es importante mencionar que se pudo reconocer este problema ya que luego de “terminar” las configuraciones, el gateway no lograba adquirir una ID, la cual dependía de la ID misma del concentrador que se estaba utilizando. Para solucionar esto se debía seleccionar, desde el menú principal, la opción **Edit ChirpStack Concentrator config**, luego seleccionar la opción **General configuration**. Para editar las configuraciones del concentrador, se escogieron las opciones que se muestran en la Figura A.9.

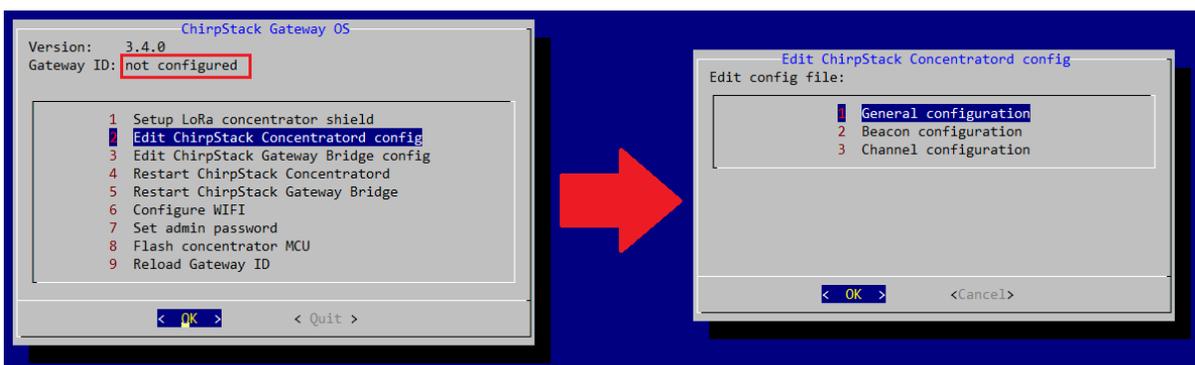
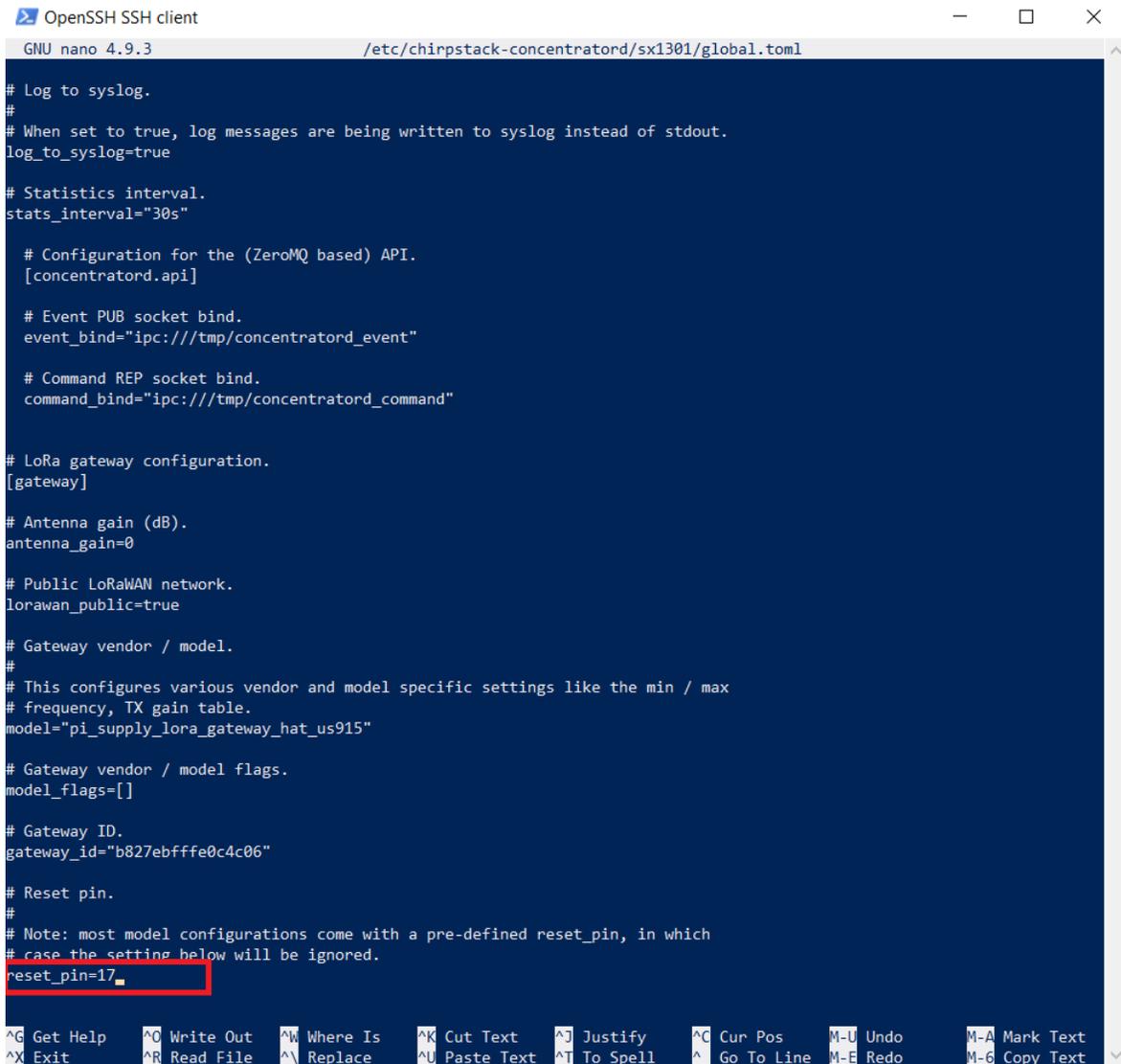


Figura A.9: Ingresando a las configuraciones del concentrador en ChirpStack.

Al seleccionar la opción anterior, se abrió un archivo llamado *global.toml*. En el archivo anterior, se tuvo que cambiar únicamente la última línea, la cual decía *reset\_pin=0* por defecto. Para este trabajo, el pin que reiniciaba el concentrador era el pin 17, por lo que se dejó

ese valor en vez de 0.

A continuación se deja una imagen detallada del archivo *global.toml* y el cambio específico que se realizó, esperando que de esta manera se logre ilustrar la modificación necesaria para que funcionara bien el gateway. Cabe resaltar que el solucionar este problema fue fundamental, ya que, al momento de buscar una solución, se llegó incluso a utilizar otro *software* distinto al proporcionado por ChirpStack, sin embargo tampoco se logró configurar el gateway ya que no se lograba reconocer el concentrador, lo cual confirmó que se tenía que configurar un detalle más y que permitiera al gateway adquirir una ID. El cambio en específico se muestra en la Figura A.10.



```
OpenSSH SSH client
GNU nano 4.9.3 /etc/chirpstack-concentrator/sx1301/global.toml
# Log to syslog.
#
# When set to true, log messages are being written to syslog instead of stdout.
log_to_syslog=true
# Statistics interval.
stats_interval="30s"
# Configuration for the (ZeroMQ based) API.
[concentrator.api]
# Event PUB socket bind.
event_bind="ipc:///tmp/concentrator_event"
# Command REP socket bind.
command_bind="ipc:///tmp/concentrator_command"
# LoRa gateway configuration.
[gateway]
# Antenna gain (dB).
antenna_gain=0
# Public LoRaWAN network.
lorawan_public=true
# Gateway vendor / model.
#
# This configures various vendor and model specific settings like the min / max
# frequency, TX gain table.
model="pi_supply_lora_gateway_hat_us915"
# Gateway vendor / model flags.
model_flags=[]
# Gateway ID.
gateway_id="b827ebfffe0c4c06"
# Reset pin.
#
# Note: most model configurations come with a pre-defined reset_pin, in which
# case the setting below will be ignored.
reset_pin=17
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text   ^J Justify   ^C Cur Pos   M-U Undo     M-A Mark Text
^X Exit      ^R Read File  ^\ Replace   ^U Paste Text ^T To Spell  ^_ Go To Line M-E Redo     M-G Copy Text
```

Figura A.10: Cambio del pin de reseteo en el archivo *global.toml*.

Una vez realizado este cambio, se guardó el archivo modificado para que posteriormente se reiniciara automáticamente el gateway. Luego de esto, se debía desplegar automáticamente el ID del gateway, y de no ser así, se podía seleccionar la opción **Reload Gateway ID** para intentar que se cargara nuevamente.

A continuación, se muestra el resultado luego de que las configuraciones, del pin de reinicio, fueron realizadas correctamente, logrando adquirir una ID para el gateway como resultado. En la Figura A.11 se muestra la ID del gateway armado, junto con la opción que se puede seleccionar para intentar recargar la ID, en caso de que no se muestre.

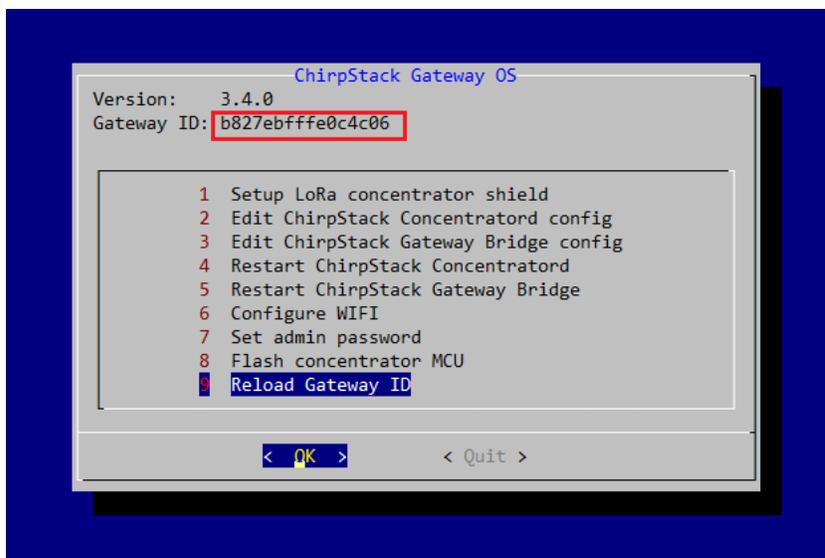


Figura A.11: ID del gateway desplegado en el menú de opciones.

Finalmente, y para comprobar que las configuraciones iniciales estuvieran bien realizadas, se ingresó nuevamente al gateway por medio de SSH. Una vez dentro del gateway, se utilizó el comando **sudo monit summary**, lo cual despliega el estado del gateway y sus partes. Para que las configuraciones iniciales estén correctas, era necesario que todos los campos bajo la columna de **Status** se marcaran como **OK**. Como se puede observar en la Figura A.12, todas las configuraciones realizadas estaban correctas y se lograron reconocer todos las partes del gateway.

```

OpenSSH SSH client
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Windows\system32> ssh admin@192.168.0.4
admin@192.168.0.4's password:

```

```

raspberrypi3:~$ sudo monit summary
Password:
Monit 5.26.0 uptime: 59m

```

Service Name	Status	Type
raspberrypi3	OK	System
chirpstack-gateway-bridge	OK	Process
chirpstack-concentrator	OK	Process

```

raspberrypi3:~$

```

Figura A.12: Configuraciones iniciales realizadas correctamente.

### A.3. Registrar y asegurar el Gateway en el *Semtech Network Server*

Para lograr una conexión segura, y que contara con las herramientas necesarias de los servicios que se requieren para una red LoRaWAN , fue necesario registrar el gateway en el *Network Server* de Semtech. La página que se utilizó, para ingresar al *Network Server*, es <https://ns.loracloud.com/> y es donde se registrara el gateway y los dispositivos de la red.

Antes de realizar el registro fue necesario crear un usuario dentro de esta página, lo cual es sencillamente ingresar los datos solicitados o simplemente utilizar una cuenta de un 3ro. Para este trabajo se utilizó una cuenta de *Google*. Una vez dentro del *Network Server* de Semtech, se seleccionó la pestaña de **Gateways** y luego **+Create** para poder crear un nuevo “perfil” para el dispositivo a registrar. La selecciones de opciones para crear un nuevo perfil se muestra en la Figura A.13.

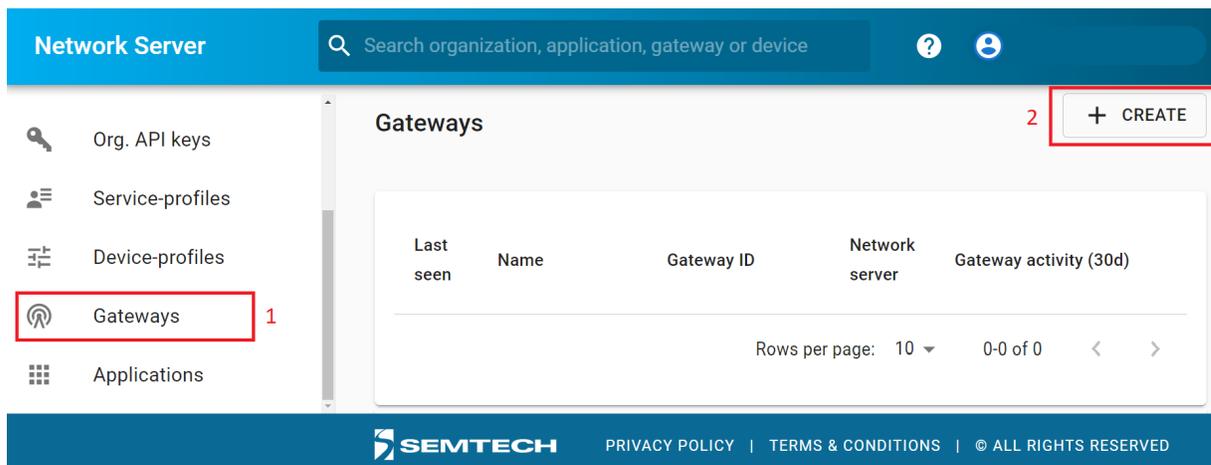


Figura A.13: Agregar un nuevo Gateway en el *Network Server* de Semtech.

Para crear el gateway dentro de la página, fue necesario completar los datos más importantes de todos los que se solicitan, por lo que principalmente se completo la información; *Nombre del Gateway*, *Descripción del Gateway*, *ID del Gateway* que es un dato de la parte anterior (Configurar el Gateway) y es importante que este seleccionada la opción **MSB** (*Most Significant Bit*). Además, se completaron los parámetros, *NetWork Server* que para este trabajo corresponde a US915, *Gateway-profile* que corresponde al mismo “valor” del Network Server, y finalmente la *Localización del Gateway* que se busco en un mapa dentro la misma pantalla donde se estaba creando el perfil. Los demás campos se dejaron como estaban y finalmente se escogió la opción **Create Gateway** que se encuentra en el fondo de la página y quedo registrado el gateway dentro del *Network Server* de Semtech. Una parte del mapa, además la opción para terminar de agregar el gateway, se muestra en la Figura A.14.

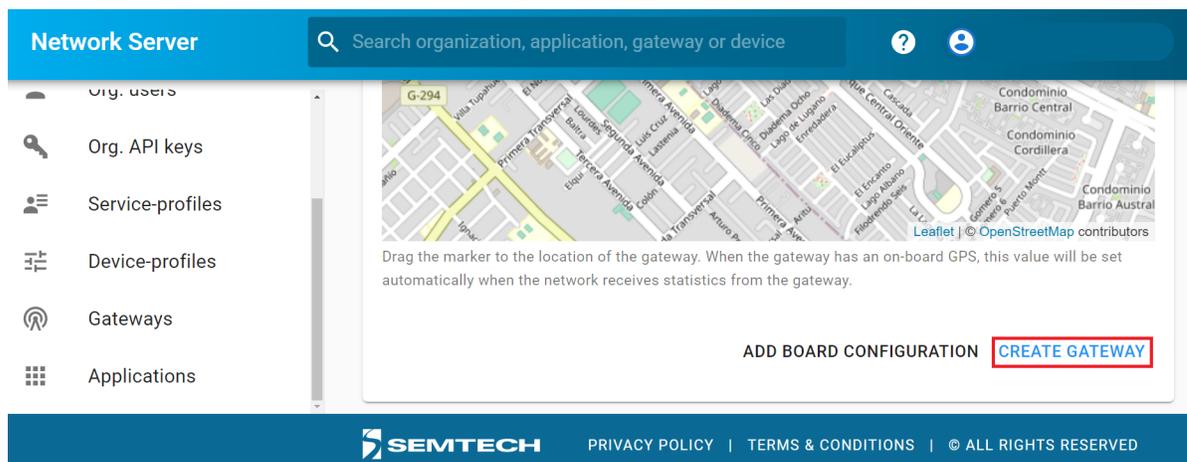


Figura A.14: Mapa para localizar el gateway en el *Network Server* de Semtech y finalmente registrar el gateway.

El siguiente paso fue asegurar el gateway con los certificados de cliente, los cuales se utilizan para establecer una comunicación segura entre el gateway y el *Network Server*, permitiendo tener un parámetro de identificación dentro de la red. Primero se debió seleccionar el gateway previamente registrado, lo cual permite abrir las configuraciones y detalles de

dicho gateway. Dentro de esta opción, se seleccionó la pestaña **Certificate** y luego la opción **Generate Certificate**, con lo que la página generó 3 certificados: el *CA certificate*, el *TLS certificate* y el *TLS key*. Con estos certificados fue posible hacer las autenticaciones necesarias para conectar el gateway con el *Network Server* de Semtech, por lo que fue necesario copiarlos y almacenarlos. En la Figura A.15 se muestra la pestaña donde se generan los certificados.



Figura A.15: Pestaña de certificados para el gateway y la opción de generarlos.

Una vez copiados los certificados era necesario ingresar nuevamente al gateway por medio de SSH, lo cual se realizó por medio del *PowerShell* de Windows. Dentro del menú de configuración del gateway, se seleccionó la opción **Edit ChirpStack Gateway Bridge config**, lo cual despliega el menú de configuraciones para la conexión del gateway hacia los servicios de Semtech. La opción seleccionada, en el menú del gateway, se muestra en la Figura A.16.

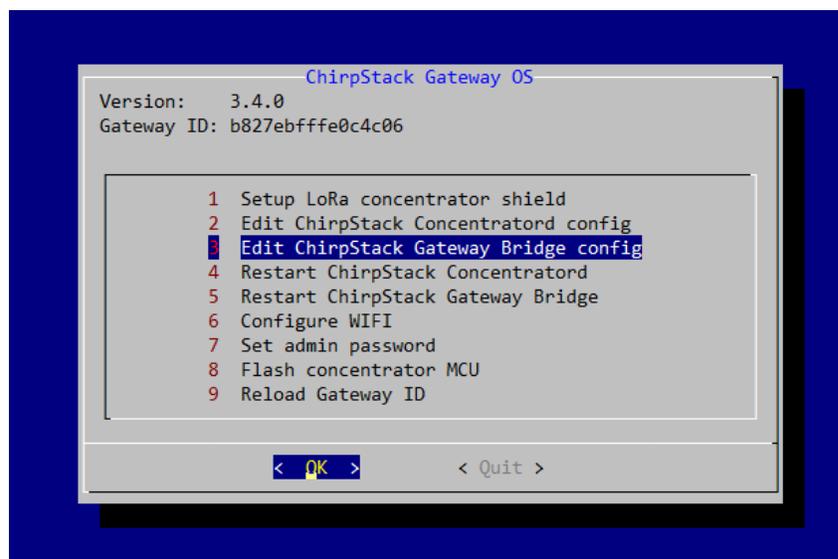


Figura A.16: Opción para ingresar en las configuraciones del puente con el *Network Server*, para el gateway.

Posteriormente, se seleccionó la opción **MQTT connection wizard**, que corresponde a la

dirección para utilizar el protocolo MQTT con el *Network Server* de Semtech. Para este trabajo se utilizó la dirección para US915, que corresponde a `ssl://us915.mqtt.loracloud.com:8883`. En la Figura A.17 se muestra dicha configuración.

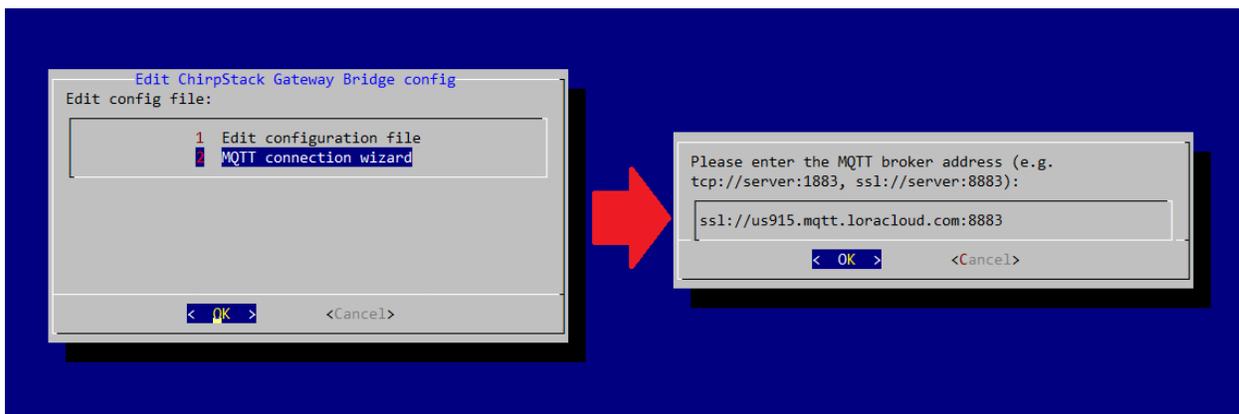


Figura A.17: Opción para configurar el enlace del MQTT y editar los certificados necesarios.

Al introducir la dirección especificada, luego se debió ingresar los certificados en el orden pedidos, que, para este caso, fue primero el *CA certificate*, luego el *TLS certificate* y finalmente el *TLS key*. Al momento de introducir los certificados, solo se debió copiarlos y guardarlos, en el archivo que se desplegó para cada uno. Luego de ser introducidos los certificados, y guardar los archivos del gateway, se desplegó un mensaje avisando que el *ChirpStack Gateway Bridge* ha sido reiniciado. Las instrucciones para ingresar los certificados se muestran en las Figuras A.18, A.19 y A.20.

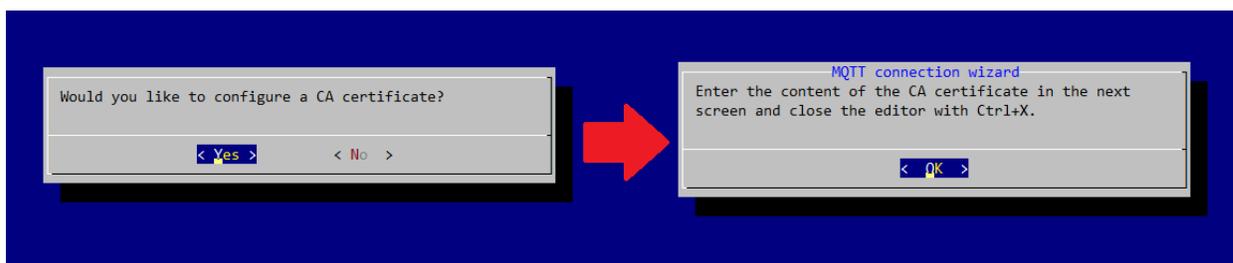


Figura A.18: Configuración de *CA certificate*.

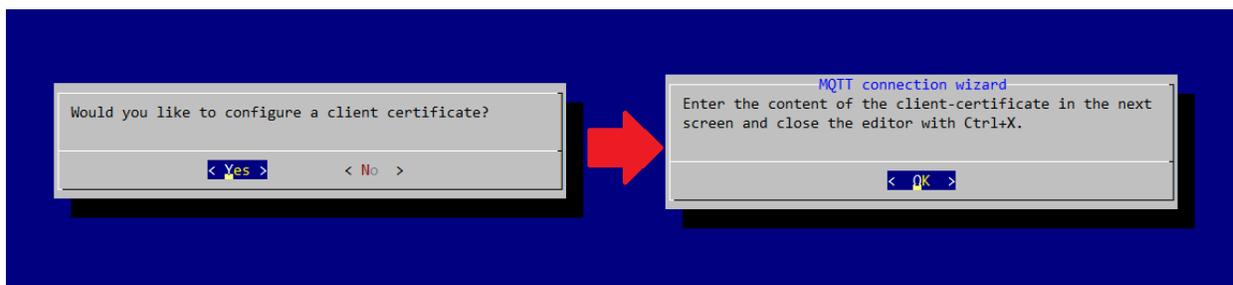


Figura A.19: Configuración de *TLS certificate*.

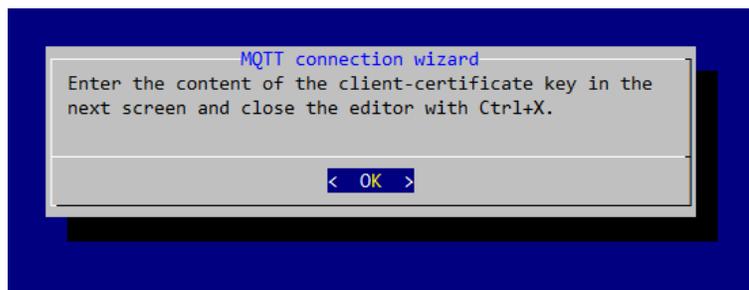


Figura A.20: Configuración de *TLS KEY*.

Con los certificados ya almacenados, es necesario revisar en el *Network Server* de Semtech que el gateway se a conectado y que esta funcionando bien. En la Figura A.21, se muestra que el gateway utilizado se conectó exitosamente.



Figura A.21: Pestaña de gateway donde se muestra el gateway previamente registrado.

## A.4. Armar el transmisor *LoRa Edge™ LR1110 Tracking Device*

Una vez armado el gateway, y configurado para poder ser utilizado con el *Network Server*, fue necesario comenzar con la manipulación del transmisor. Para este caso se utilizó el *Development Kit* del chip **LR1110** de Semtech. El *Development Kit* cuenta con las piezas necesarias para poder ser armar el transmisor con dos configuraciones distintas, las que se diferencian en el tipo de antena GNSS; ya sea la antena pasiva para un caso o antena activa para la otra configuración, además del *Modem Shield* correspondiente para cada antena. Para el trabajo realizado, se utilizó la antena GNSS activa y el módulo activo para esto. Cabe mencionar que es fue necesario contar con un cable USB con conexión tipo A por un lado y conexión tipo *mini B* por el otro, siendo este el tipo de conexión aceptada por el transmisor, para, posteriormente, ser energizado o conectado al computador para ser configurado.

Primero, se partió armando el módulo activo, al cual se le tuvo que conectar las 3 antenas necesarias; la antena para comunicación entre dispositivos de la red y que iba en la zona

demarcada `ANT_LR11XX`, la antena activa GNSS y que se conectaba en la zona de `ANT_GPS`, y la última antena; la antena para Wi-Fi y que iba conectada en el lugar demarcado por `ANT_WIFI`.

Posteriormente, se debía conectar la placa núcleo del transmisor, lo cual era simplemente acoplar el módulo previamente armado a las entradas ARDUINO® del núcleo.

Tras esto, se conectó la pantalla encima del módulo activo y en las entradas correspondientes. La pantalla era *touch* y era la encargada de desplegar el menú para manipular el transmisor, ya fuera para: configurar, observar, o realizar pruebas. El transmisor armado, hasta este punto, se muestra en la Figura A.22.

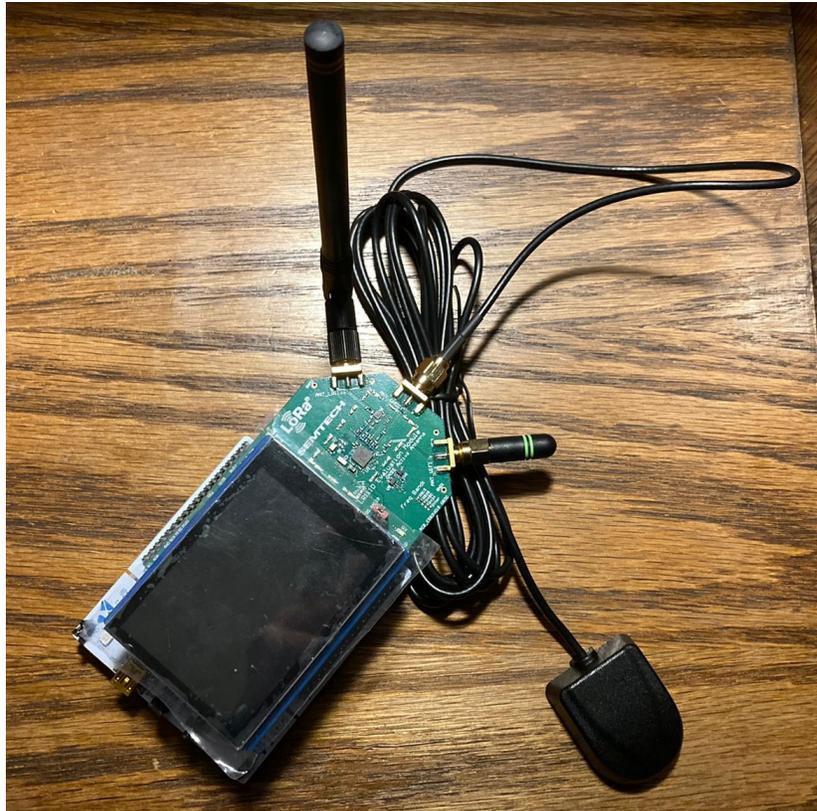


Figura A.22: Transmisor armado, pero, sin el cable USB.

Finalmente se conectó el cable USB en la placa núcleo, siendo conectado en la única entrada disponible para eso. Con esto se tenía ya armado el transmisor del *Development Kit*.

## A.5. Instalar el software del núcleo y del chip LR1110

Para poder utilizar el transmisor del *Development Kit*, fue necesario realizar la actualización al último *firmware* disponible, el cual se encontraba de manera online en la página [https://github.com/Lora-net/lr1110\\_updater\\_tool/wiki](https://github.com/Lora-net/lr1110_updater_tool/wiki). Para el este trabajo se utilizó la modalidad *Modem-E* del *Development Kit*, ya que es la que cuenta con las interfaz necesaria para, posteriormente, ser utilizado dentro de una red LoRaWAN y de forma directa. A fin de esto, fue necesario descargar el archivo `lr1110_updater_tool_v1.2.0_modem_v1.1.7.bin`,

siendo este el más actual en el momento de realizar este trabajo.

Para la realización de la actualización se tuvo que conectar el transmisor al computador y verificar que este último reconociera el dispositivo, el cual debía aparecer con el identificador de **NODE\_L476RG**. La actualización era tan simple como tomar el archivo previamente descargado y copiarlo en el **NODE\_L476RG**. La forma en que se copió el archivo se muestra en la Figura A.23.

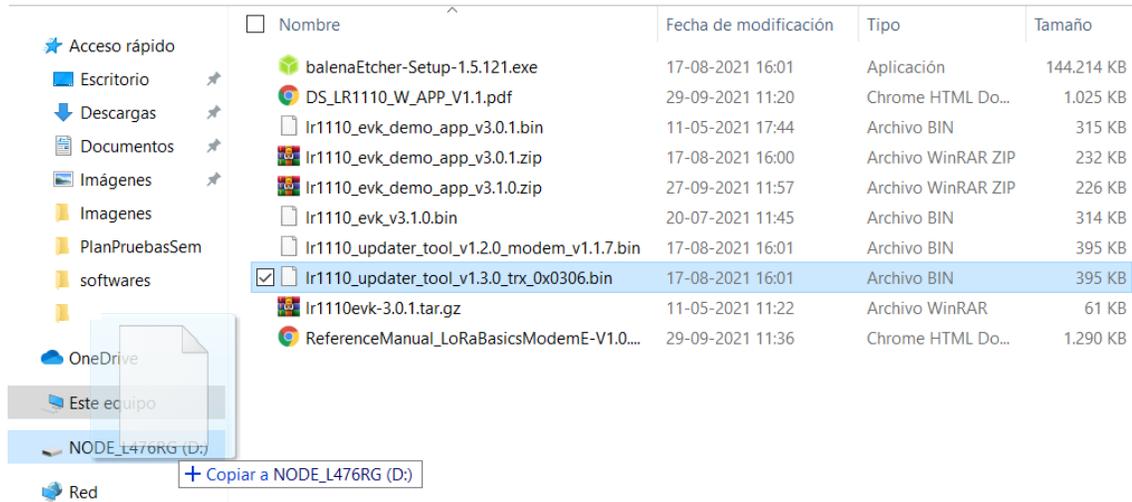


Figura A.23: Copia del *updater toll*

Una vez iniciada la copia del archivo, se logró ver inmediatamente como la pantalla del transmisor se mantenía en blanco mientras se realizaba la copia. Una vez finalizada la copia, se pudo divisar en la pantalla como el *firmware* realizaba su cometido y el dispositivo entraba en un estado de actualización, por lo cual se tuvo que esperar hasta que este proceso finalizara. Cuando finalizó la actualización, se desplegó un mensaje avisando que se había completado el proceso y que se debía instalar alguna aplicación para utilizar los menús disponibles para la modalidad *Modem-E*. El proceso de actualización, completo, se muestra en la Figura A.24.

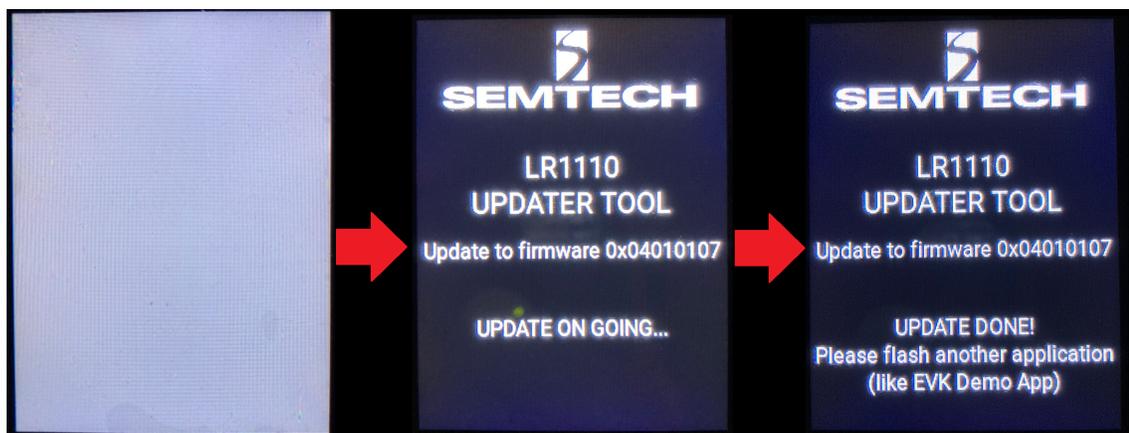


Figura A.24: Estados del transmisor al actualizarlo con el *updater tool*.

Para el *software* a instalar, fue necesario descargar un paquete que contenía distintos archivos y el cual esta disponible en la página [https://github.com/Lora-net/lr1110\\_evk\\_demo\\_app/wiki/Software-packages](https://github.com/Lora-net/lr1110_evk_demo_app/wiki/Software-packages). Al momento de realizar este trabajo, la versión más actual disponible corresponde a la V3.1.0. La descarga era un conjunto de archivos comprimidos, sin embargo, para esta sección solo se utilizó el archivo **lr1110\_evk\_v3.1.0.bin**. El archivo corresponde al aplicación demo para el *Evaluation Kit* y que proporciona el menú interactivo de este. El procedimiento para la instalación del *software* fue muy similar a la actualización realizada anteriormente, con la salvedad de que al momento de finalizar la copia del archivo *.bin*, el transmisor desplegaba el menú de inicio de la aplicación instalada, el cual se logra ver en la Figura A.25.



Figura A.25: Estados del transmisor al actualizarlo con el *updater tool*.

Con esto realizado, ya se contaba con lo necesario para poder utilizar el transmisor por medio de la pantalla táctil. Era importante tener en cuenta que se debía mantener siempre energizado el dispositivo.

## A.6. Reclamar el dispositivo en el portal de *LoRa Cloud™*

Para este trabajo se utilizó una activación en el aire (*Over the Air Activation*) para los dispositivos a enlazar con la *LoRa Cloud™*. Para esto, se tuvieron que utilizar las *AppKeys*, únicas, que provee el dispositivo a enlazar, y es por lo que este método corresponde a una de las formas más seguras para realizar la conexión con los servicios online de la nube.

Primero, fue necesario ingresar a la página <https://www.loracloud.com/> y crear una cuenta para poder acceder a los servicios proveídos por la *LoRa Cloud™*. Luego de iniciar sesión dentro de la página, se ingresó en la pestaña **LoRa Cloud Device Join**, lo que permitía crear un perfil como *Application Owner* y que se utilizaría más tarde para lograr conectar el transmisor con el gateway previamente configurado. La opción seleccionada en el *LoRa Cloud™* se muestra en la Figura A.26.

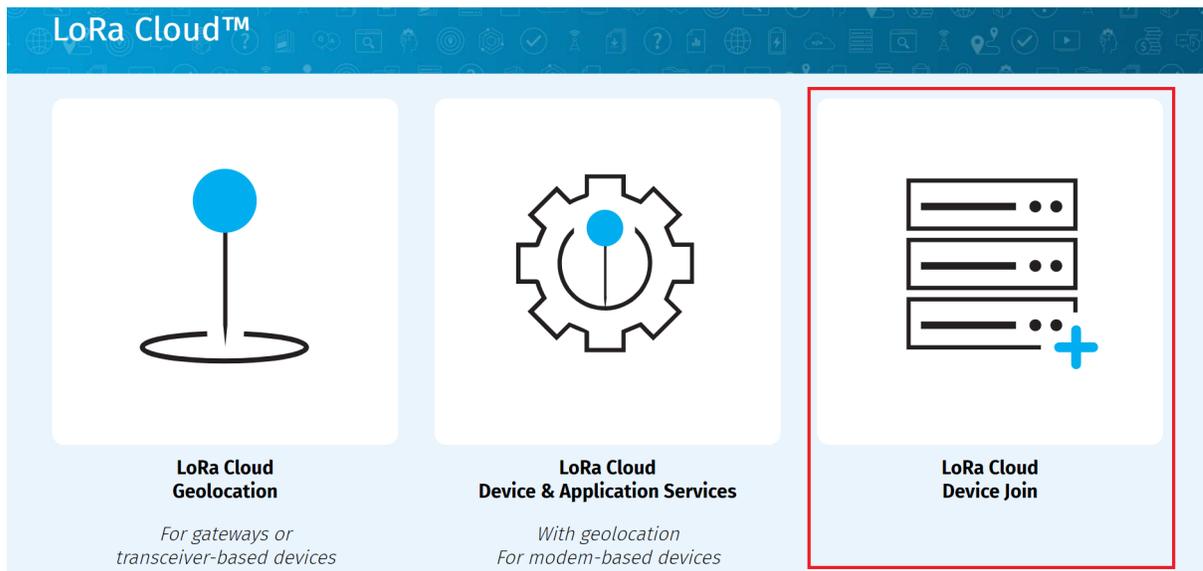


Figura A.26: Pestaña de Lora Cloud Device Join, desde el portal *LoRa Cloud™*.

En la sección que se ingresó previamente, se tuvo que seleccionar la pestaña **APPLICATION OWNERS**, luego se debía ingresar un nombre para el perfil a crear y posteriormente seleccionar la opción **CREATE A NEW OWNER**. Este proceso se muestra en la Figura A.27.

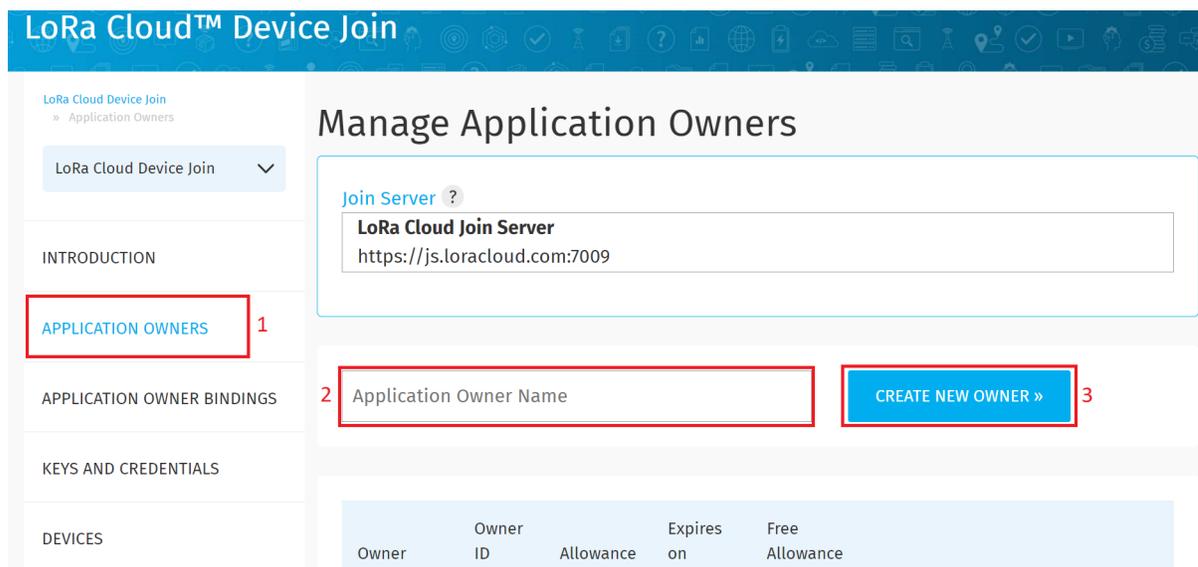


Figura A.27: Pestaña de *application owners* y donde crear un nuevo perfil

Con el perfil creado, lo siguiente fue asignar el *free allowance*, el cual es un “regalo” luego de crear la cuenta en la página. Sin embargo, es importante destacar que esto solo se puede realizar una única vez. El *free allowance* permite que el perfil creado obtenga 100 permisos para enlazar dispositivos en la página de LoRa Cloud™. Dentro de **APPLICATION OWNERS**, se tuvo que encontrar el perfil previamente creado y seleccionar la opción **Free Allowance**. En la Figura A.28 se muestra el perfil al cual se le asigno el **Free Allowance**.

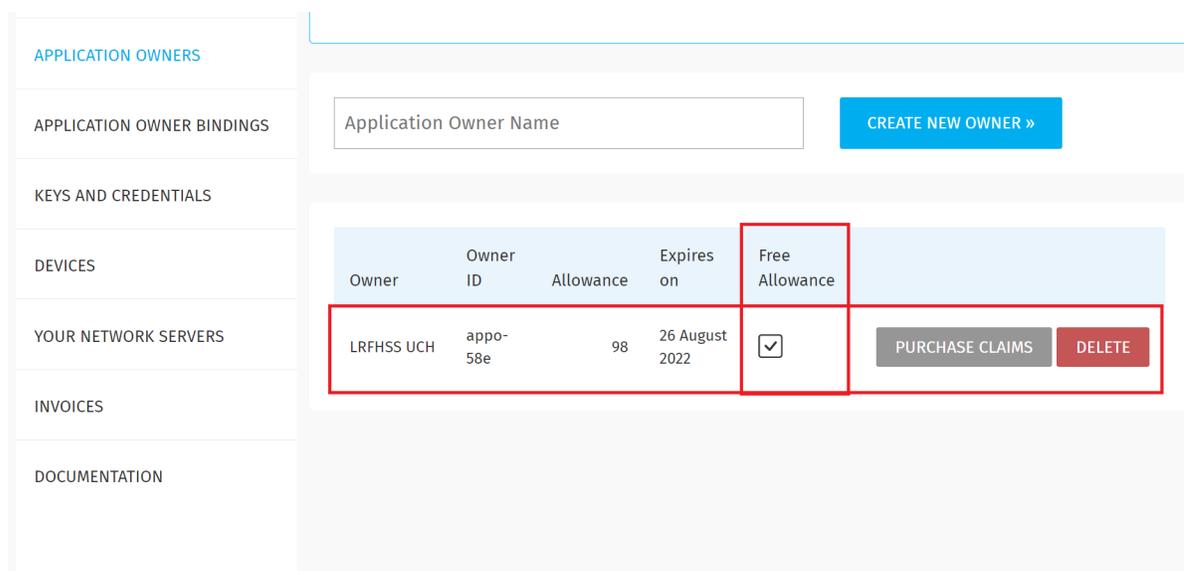


Figura A.28: Asignación de *free allowance* para el perfil creado.

Para enlazar el transmisor que se había configurado antes, fue necesario prenderlo y seleccionar la opción **START** dentro del menú de inicio del *demo application*. Luego se busco la opción **EUI**, que era la sección donde estaban las *fetch keys* propias del dispositivo y del chip LR1110. Las claves se divisaban luego de seleccionar **GO!**, el que se encontraba al lado de la opción **Provisioning (EUI, PIN)**.

Con los datos del transmisor, se debía ingresar nuevamente al LoRa Cloud™ y luego entrar en la sección **LoRa Cloud Device Join**. Dentro de la opción seleccionada, se debía ingresar en la pestaña **DEVICES** y luego escoger la opción **CLAIM INDIVIDUAL DEVICE**, proceso que se muestra en la Figura A.29.

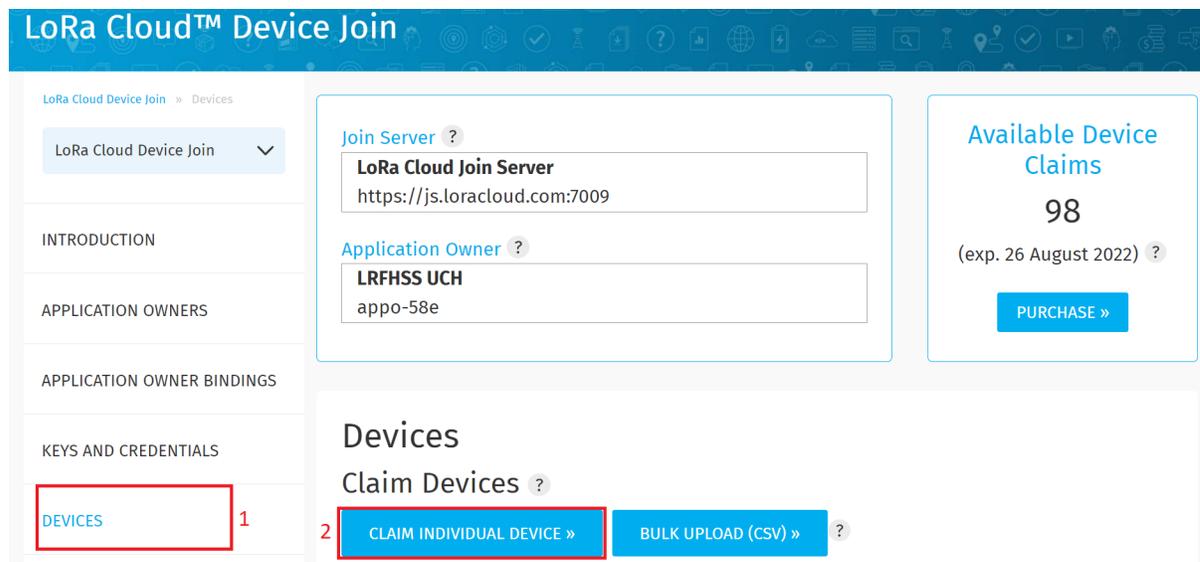


Figura A.29: Agregando un dispositivo con la opción *claim devices*, dentro de la pestaña de *devices*

Para agregar un dispositivo, fue necesario ingresar las *fetch keys* del transmisor que se estaba utilizando para posteriormente seleccionar la opción **CLAIM DEVICE**. El lugar donde se tuvo que agregar las *keys* se muestra en la Figura A.30.

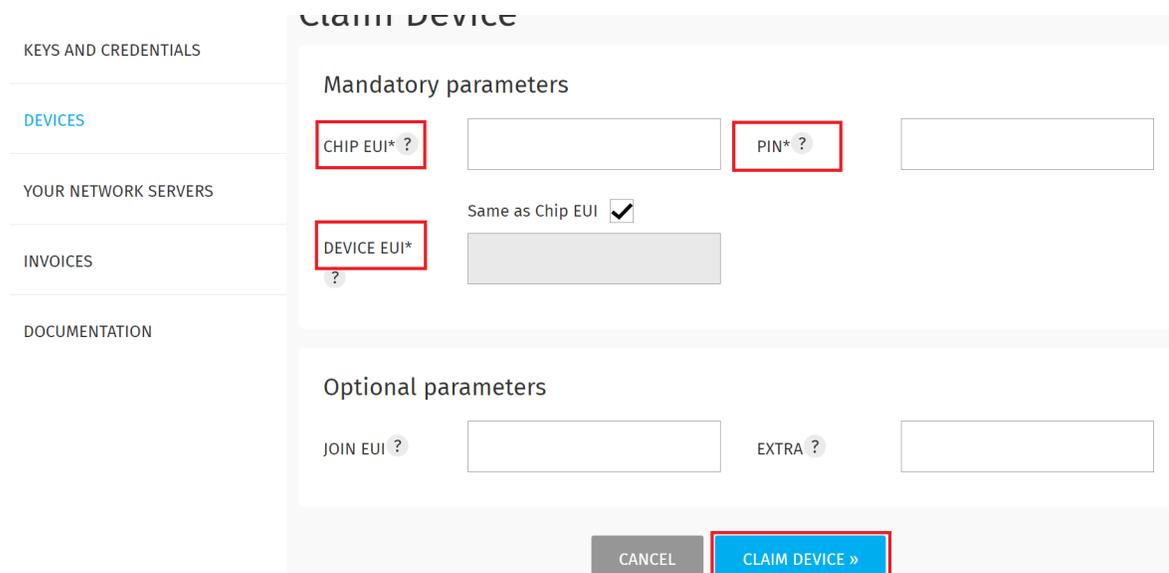


Figura A.30: Introduciendo las **fetch keys** y agregando finalmente el dispositivo

Para que el proceso anterior estuviera correcto, se debió revisar nuevamente la pestaña de **DEVICES** y verificar que apareciera el dispositivo agregado. El dispositivo agregado exitosamente se muestra en la Figura A.31.

EUIs	Claim Time	Last Join	Network Server	LoRaWAN® Version	Exported
<b>EUI:</b> 00-16-C0-01-F0-00-3A-D4	2021-08-26 14:03:55	2021-11-03 09:04:37	rens-106	1.0.3	<input type="checkbox"/>
<b>Chip EUI:</b> 00-16-C0-01-F0-00-3A-D4					
<b>Join EUI:</b> 00-16-C0-01-FF-FE-00-01					

Figura A.31: Dispositivo agregado exitosamente.

## A.7. Crear un propietario del dispositivo y obtener el Token de acceso

Antes de poder agregar un dispositivo al *Network Server* de Semtech, primero se tuvo que crear un propietario para el dispositivo, dentro del portal de *LoRa Cloud™*. Esto permitiría obtener los *token* de acceso y, además, habilitaría al dispositivo con los permisos necesarios para utilizar la plataforma de *LoRa Cloud™*.

Dentro de la página del portal de *LoRa Cloud™*, se ingresó en la opción **LoRa Cloud Device & Application Services**, lo cual se muestra en la Figura A.32.

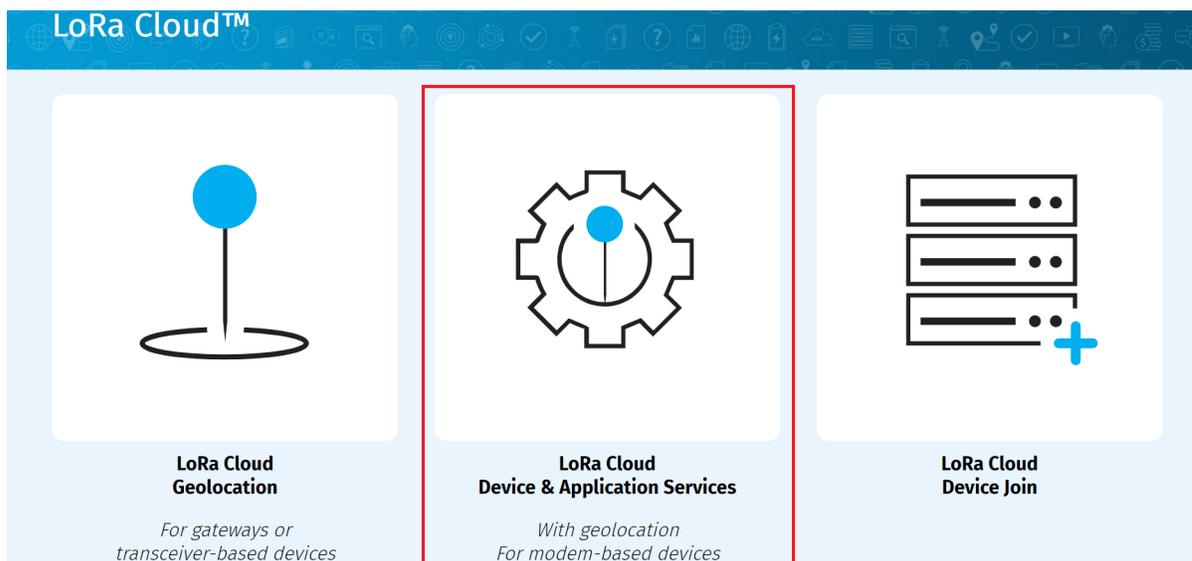


Figura A.32: Pestaña LoRa Cloud Device & Application Services, desde el portal de *LoRa Cloud™*.

En el menú disponible, se seleccionó la opción **DEVICE OWNERS** y luego se comenzó con la creación de un perfil, el cual era para manejar los dispositivos que se fueran a enlazar y poder asignar los *API access token* correspondientes. Para la configuración del perfil, se eligió la opción **CREATE NEW OWNER**, lo cual se ilustra en la Figura A.33.

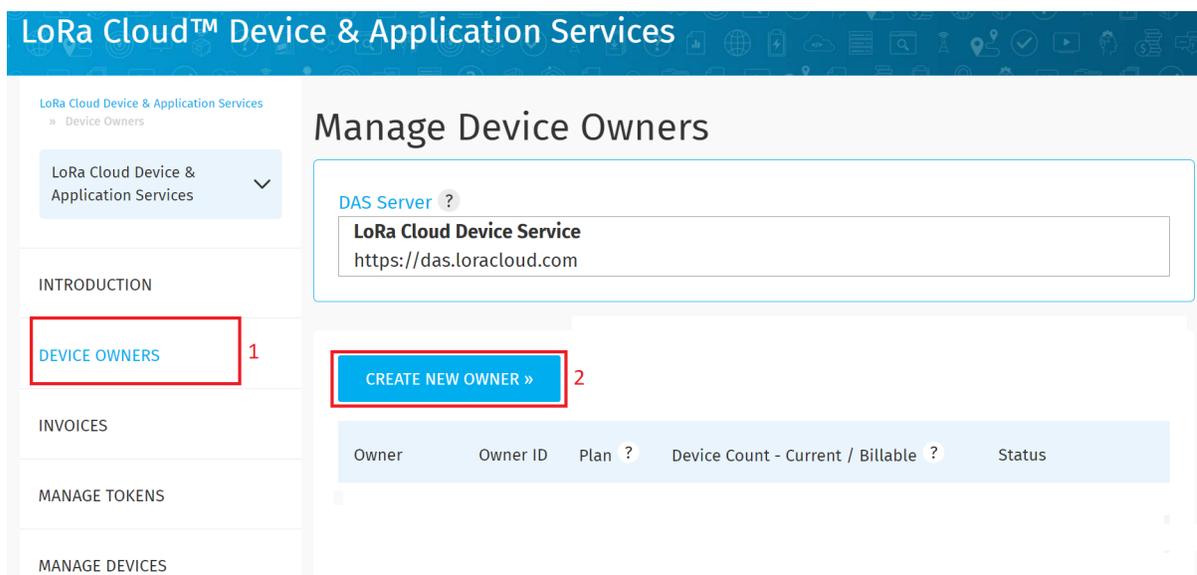


Figura A.33: Sección de *device owners* y opción donde crear el perfil nuevo.

Para agregar un *device owner* fue necesario completar dos campos, uno donde se debía ingresar un nombre para el perfil, y otro donde se tuvo que seleccionar el plan, que para este caso se utilizó la opción **One Stop Shop**. Finalmente, se debía guardar el perfil configurado al escoger la opción **SUBMIT**. Los campos a rellenar y la opción final, se muestran en la Figura A.34.

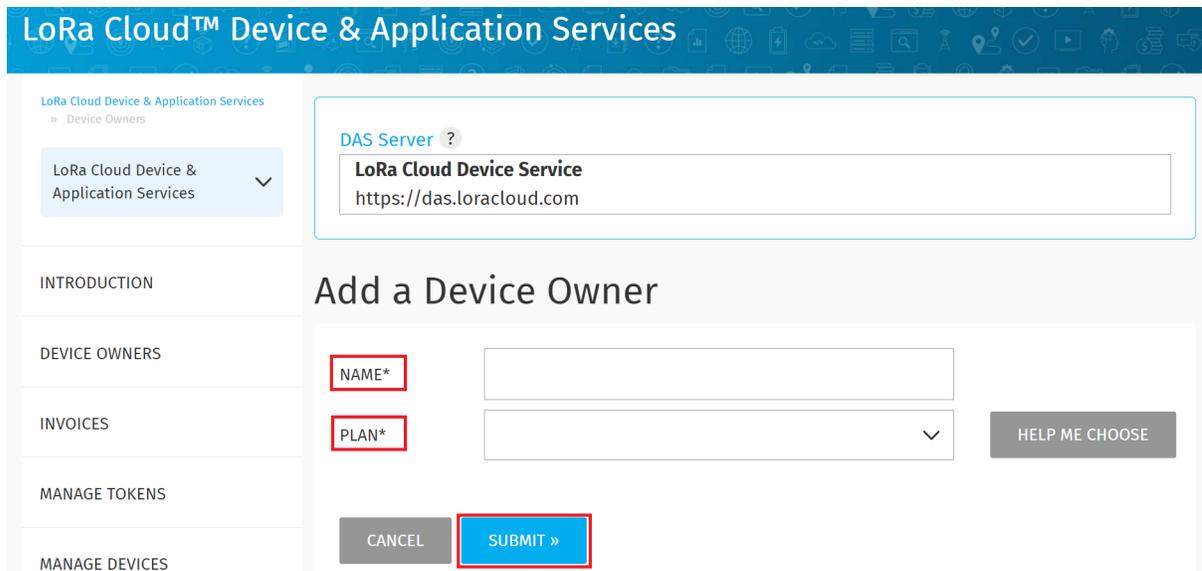


Figura A.34: Campos a rellenar y la opción *submit* para crear el perfil.

Al ingresar nuevamente en la pestaña **DEVICE OWNERS**, se observó el perfil previamente creado y con el estado *Free Access*, tal como se observa en la Figura A.35.

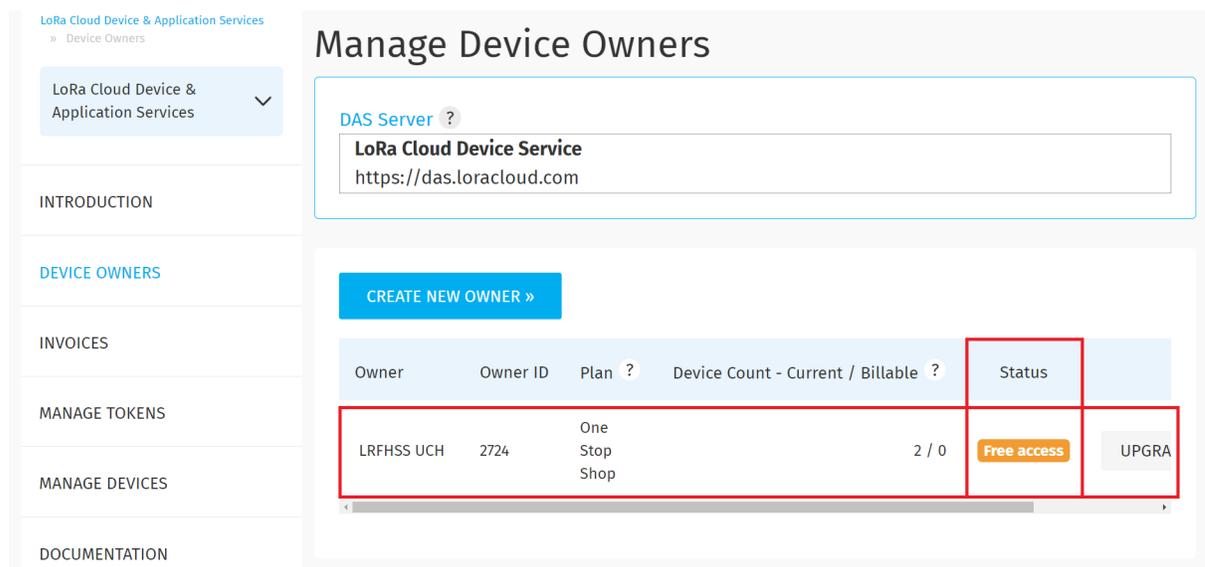
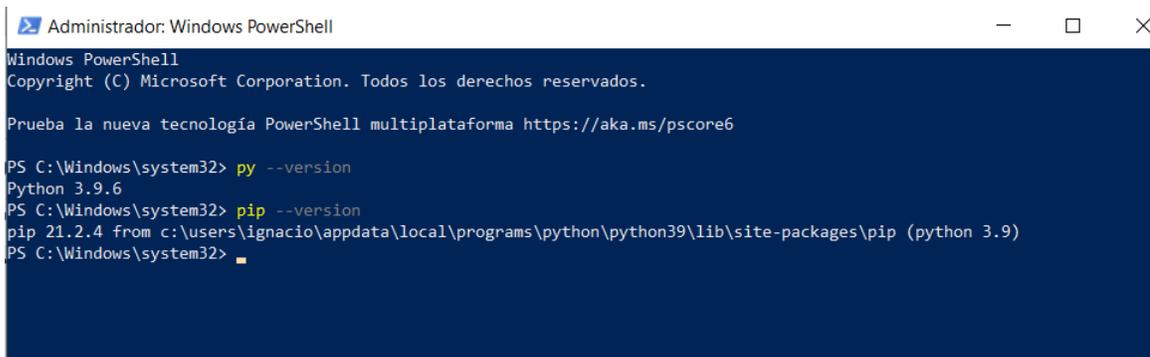


Figura A.35: Asignación del *free access* para el perfil creado.

Lo último fue encontrar el *API Access Token*, que serviría para conectar, posteriormente, los dispositivos al *Network Server* de Semtech. Para encontrar dicha información, fue necesario entrar en la pestaña **MANAGE TOKENS**, dentro del mismo menú del **LoRa Cloud Device & Application Services**. Una vez dentro de los *tokens* se pudo divisar dos “perfiles” donde se encontraban los token necesarios y se tuvo que seleccionar uno de los dos, siendo importante destacar que no repercutía en nada el cual escoger, solo se debía copiar uno de los dos y almacenarlo para ser utilizado más adelante. En la Figura A.36 se muestra



lo antes descrito, se debía instalar python nuevamente. En la Figura A.37 se muestran las versiones utilizadas para este trabajo.



```
Administrador: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Windows\system32> py --version
Python 3.9.6
PS C:\Windows\system32> pip --version
pip 21.2.4 from c:\users\ignacio\appdata\local\programs\python\python39\lib\site-packages\pip (python 3.9)
PS C:\Windows\system32>
```

Figura A.37: Verificación de las versiones instaladas de *python* y *pip*

Con las herramientas en sus versiones correctas, se debía instalar el *AlmanacUdate Toll* por medio de pip, que corresponde a la herramienta que serviría para actualizar el registro dentro del transmisor. Para la instalación, era necesario ir a los archivos comprimidos que se descargaron anteriormente desde [https://github.com/Lora-net/lr1110\\_evk\\_demo\\_app/wiki/Software-packages](https://github.com/Lora-net/lr1110_evk_demo_app/wiki/Software-packages), pero en esta sección se iba a utilizar el archivo con nombre **lr1110evk-3.1.0.tar.gz**. Una vez localizado el archivo a utilizar, se debía ingresar al *PowerShell* de Windows y usar el comando **pip install setuptools wheel** para verificar ciertas herramientas de pip. Luego, se procedió a instalar el archivo mediante el comando **pip install PATH\_OF\_TAR.GZ**, donde la parte del comando señalada como **PATH\_OF\_TAR.GZ** corresponde a la dirección donde se encontraba el archivo **lr1110evk-3.1.0.tar.gz** dentro del computador que se estaba utilizando. Una vez ejecutado este comando se debía comenzar con la instalación, que al finalizar, debía desplegar el mensaje **Successfully installed** para que todo estuviera correctamente instalado. A continuación, se muestra, en la Figura A.38, el proceso para que la instalación esté completa.

```

Administrador: Windows PowerShell

PS C:\Windows\system32> pip install setuptools wheel
Requirement already satisfied: setuptools in c:\users\ignacio\appdata\local\programs\python\python39\lib\site-packages (56.0.0)
Requirement already satisfied: wheel in c:\users\ignacio\appdata\local\programs\python\python39\lib\site-packages (0.37.0)
PS C:\Windows\system32> pip install C:\Users\Ignacio\Desktop\U\memoria\softwares\lr1110evk-3.1.0.tar.gz
Processing c:\users\ignacio\desktop\U\memoria\softwares\lr1110evk-3.1.0.tar.gz
Requirement already satisfied: pyserial in c:\users\ignacio\appdata\local\programs\python\python39\lib\site-packages (from lr1110evk==3.1.0) (3.5)
Requirement already satisfied: pyyaml in c:\users\ignacio\appdata\local\programs\python\python39\lib\site-packages (from lr1110evk==3.1.0) (5.4.1)
Requirement already satisfied: requests in c:\users\ignacio\appdata\local\programs\python\python39\lib\site-packages (from lr1110evk==3.1.0) (2.26.0)
Requirement already satisfied: js1 in c:\users\ignacio\appdata\local\programs\python\python39\lib\site-packages (from lr1110evk==3.1.0) (0.2.4)
Requirement already satisfied: jsonschema in c:\users\ignacio\appdata\local\programs\python\python39\lib\site-packages (from lr1110evk==3.1.0) (3.2.0)
Requirement already satisfied: pynmea2 in c:\users\ignacio\appdata\local\programs\python\python39\lib\site-packages (from lr1110evk==3.1.0) (1.18.0)
Requirement already satisfied: setuptools in c:\users\ignacio\appdata\local\programs\python\python39\lib\site-packages (from lr1110evk==3.1.0) (56.0.0)
Requirement already satisfied: attrs>=17.4.0 in c:\users\ignacio\appdata\local\programs\python\python39\lib\site-packages (from jsonschema->lr1110evk==3.1.0) (21.2.0)
Requirement already satisfied: pyrsistent>=0.14.0 in c:\users\ignacio\appdata\local\programs\python\python39\lib\site-packages (from jsonschema->lr1110evk==3.1.0) (0.18.0)
Requirement already satisfied: six>=1.11.0 in c:\users\ignacio\appdata\local\programs\python\python39\lib\site-packages (from jsonschema->lr1110evk==3.1.0) (1.16.0)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\ignacio\appdata\local\programs\python\python39\lib\site-packages (from requests->lr1110evk==3.1.0) (1.26.6)
Requirement already satisfied: idna<4,>=2.5 in c:\users\ignacio\appdata\local\programs\python\python39\lib\site-packages (from requests->lr1110evk==3.1.0) (3.2)
Requirement already satisfied: charset-normalizer<=2.0.0 in c:\users\ignacio\appdata\local\programs\python\python39\lib\site-packages (from requests->lr1110evk==3.1.0) (2.0.4)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\ignacio\appdata\local\programs\python\python39\lib\site-packages (from requests->lr1110evk==3.1.0) (2021.5.30)
Building wheels for collected packages: lr1110evk
  Building wheel for lr1110evk (setup.py) ... done
  Created wheel for lr1110evk: filename=lr1110evk-3.1.0-py3-none-any.whl size=130580 sha256=0b4a81ede143fd96de3f3611828d135322a85910138871870c89848252e2cd2
  Stored in directory: c:\users\ignacio\appdata\local\pip\cache\wheels\db\25\xf1\c79cdaa8dea3809a73593e5b21ced781b8d232b2831846a7
Successfully built lr1110evk
Installing collected packages: lr1110evk
  Attempting uninstall: lr1110evk
    Found existing installation: lr1110evk 3.0.1
    Uninstalling lr1110evk-3.0.1:
      Successfully uninstalled lr1110evk-3.0.1
Successfully installed lr1110evk-3.1.0
PS C:\Windows\system32>

```

Figura A.38: Instalación del *AllmanacUpdate* exitosa.

Para verificar que todo estuviera bien se utilizó el comando **AlmanacUpdate -version** y se mostró la versión que se había instalado. Con todo funcionando bien, ya se podía proceder con la actualización del registro de satélites del transmisor.

Para la actualización, primero se debía conectar el transmisor, por medio de USB, con el computador que se iba a utilizar. Una vez conectado, se debía verificar el puerto que se estaba utilizando para el cable USB, lo cual se podía revisar en el **Administrados de dispositivos** de Windows. Una vez dentro del administrador de dispositivos, se debía buscar la sección **Puertos (COM y LPT)** donde, al seleccionarla, se desplegaba los dispositivos conectados al computador y las interfaces de estas conexiones. La idea era encontrar el transmisor dentro de estos puertos, que si no se mostraba, era debido a que estaba oculto, y para poder verlo se debía seleccionar en la parte superior la opción **Ver** y luego marcar la opción **Mostrar dispositivos ocultos**. Luego de encontrar el transmisor y ver el puerto que estaba utilizando, se procedió a utilizar el *PowerShell* de Windows para ejecutar el siguiente comando **AlmanacUpdate BDEAGKqZWmIHpVyY+7TdS9TymWvRLIVG6tFdUzeMkgewdXxXXxxX**

`-d COM_TX`, donde el parámetro `COM_TX` corresponde al puerto que está utilizando el transmisor. Finalmente, luego de que la actualización terminara, se desplegaron los mensajes de finalización. Estos mensajes aseguraron que el procedimiento fue realizado correctamente, con lo cual ya se había terminado de actualizar el registro de GNSS dentro del transmisor y ya se podía utilizar la opción de búsqueda asistida sin problemas. La correcta actualización del *Almanac*, para el transmisor utilizado en este trabajo, se muestra en la Figura A.39.

```

PS C:\Users\Ignacio> AlmanacUpdate AQEAoJhDr5S1KD+XKDerHF0bG/u13rBbNQLVaYVURX0hpA97uWRR -d COM4
[2021-10-06 16:34:25.345289] Starting
[2021-10-06 16:34:25.439622] Fetching almanac data from url https://das.loracloud.com/api/v1/almanac/full...
[2021-10-06 16:34:27.822247] CRC is 788939604
[2021-10-06 16:34:27.822247] Start downloading to embedded...
[2021-10-06 16:34:27.822247] .
[2021-10-06 16:34:27.822247] .
[2021-10-06 16:34:27.822247] .
[2021-10-06 16:34:27.838030] .
[2021-10-06 16:34:27.838030] .
[2021-10-06 16:34:27.838030] .
[2021-10-06 16:34:27.838030] .
[2021-10-06 16:34:27.838030] .
[2021-10-06 16:34:27.838030] .
[2021-10-06 16:34:27.853744] .
[2021-10-06 16:34:27.853744] .
[2021-10-06 16:34:27.853744] .
[2021-10-06 16:34:27.853744] .
[2021-10-06 16:34:27.853744] .
[2021-10-06 16:34:27.853744] .
[2021-10-06 16:34:27.853744] .
[2021-10-06 16:34:27.853744] .
[2021-10-06 16:34:27.853744] .
[2021-10-06 16:34:27.869366] .
[2021-10-06 16:34:27.869366] .
[2021-10-06 16:34:27.884991] .
[2021-10-06 16:34:27.888001] .
[2021-10-06 16:34:28.107047] .
[2021-10-06 16:34:28.122623] .
[2021-10-06 16:34:28.122623] .
[2021-10-06 16:34:28.122623] .
[2021-10-06 16:34:28.122623] .
[2021-10-06 16:34:28.232449] .
[2021-10-06 16:34:28.232449] Downloading terminated
[2021-10-06 16:34:28.235506] Checking...
[2021-10-06 16:34:28.251262] Check terminated
[Serial Handler] Leaving runtime
[2021-10-06 16:34:29.266357] Bye

```

Figura A.39: Actualización exitosa.

## A.9. Agregar el transmisor del *Development Kit* en el *Semtech Network Server* y conexión con los servicios de *LoRa Cloud™*

Luego de haber realizado las configuraciones del transmisor, dentro del portal de *LoRa Cloud™*, se tuvo que agregar el dispositivo dentro del *Network Server* de Semtech. Para poder agregar un dispositivo al *Network Server*, primero se debía crear un perfil para el transmisor. Se comenzó ingresando a la página <https://ns.loracloud.com/> y, luego de iniciar sesión, se tuvo que ir sección **Device-profiles** y elegir la opción **+CREATE**. Las secciones para crear un nuevo perfil, se muestran en la Figura A.40.

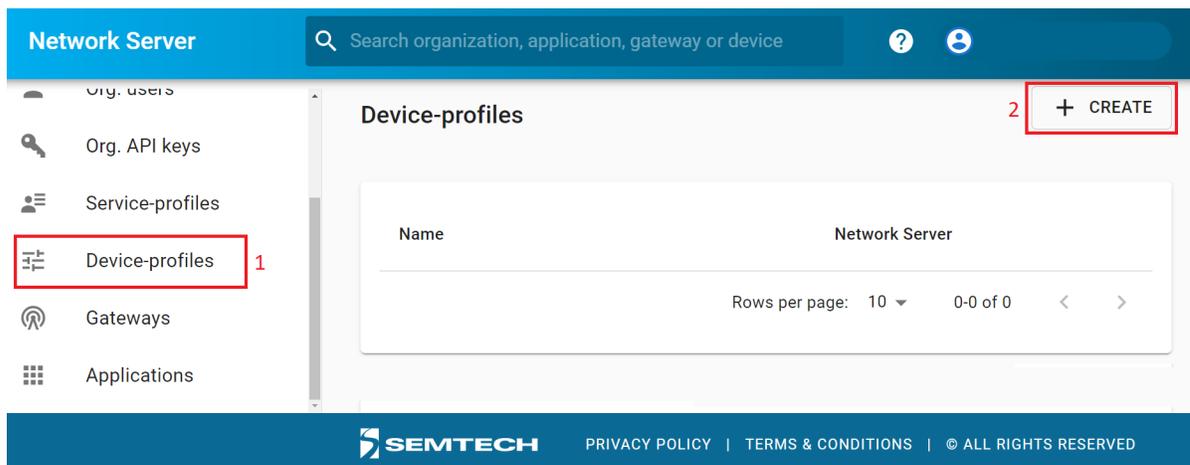


Figura A.40: Pestaña de *device-profiles* y la opción para crear un nuevo perfil.

Dentro de la creación del perfil, se tuvo que completar los siguientes campos: el **Device-profile name** con el nombre del perfil, el **Network Server** que corresponde a US915 para este trabajo, el **LoRaWAN MAC Version** que para el modo *Modem-e* del transmisor corresponde a la versión 1.0.3, el **LoRaWAN Regional Parameters revision** que se escogió la opción “A” y es la soportada por el modo utilizado, y, finalmente, el **Uplink interval (seconds)**, que se dejó con el valor de 300. Los demás campos se dejaron como estaban.

Antes de poder agregar el perfil, se tuvo que configurar otra sección, la cual era la pestaña **JOIN (OTAA/ABP)** y se marcó la casilla que decía **Device supports OTAA**. Con lo anterior realizado, ya se contaba con las configuraciones para agregar el perfil mediante la opción **CREATE DEVICE-PROFILE**, tal como se muestra en la Figura A.41.

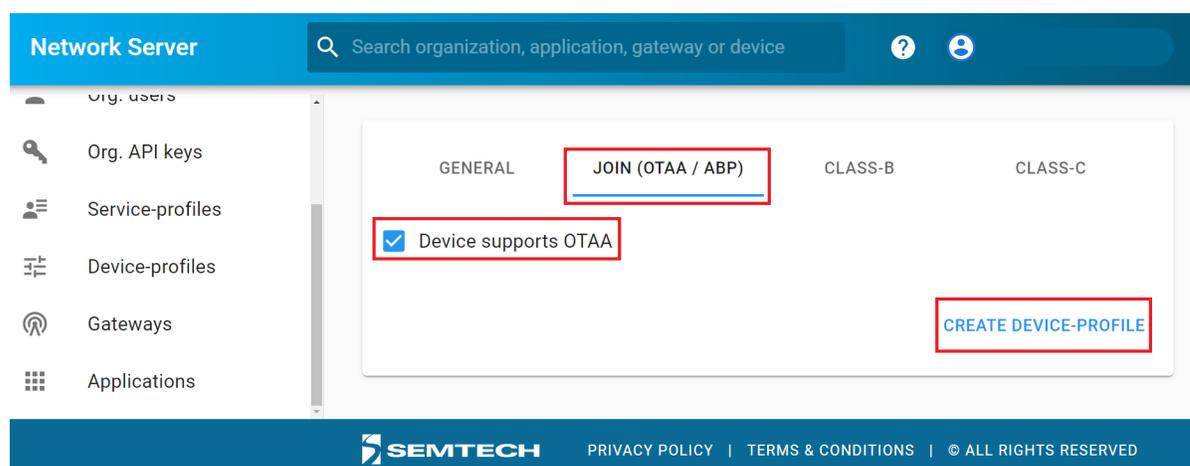


Figura A.41: Configuración del *join* para el perfil que se estaba creando y la creación de este.

Una vez agregado, el perfil creado debería aparecer dentro de los *Device profiles*, dentro del *Network Server*. El perfil creado, exitosamente, se muestra en la Figura A.42.

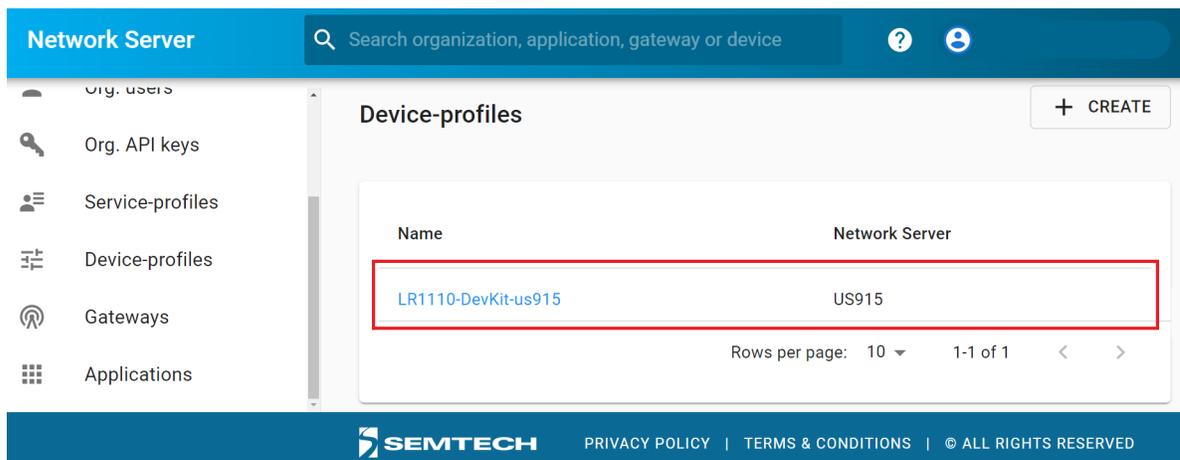


Figura A.42: Creación exitosa del perfil de para el dispositivo.

Antes de agregar algún dispositivo, se tuvo que crear una *Application* para poder tener un “lugar” donde juntar todos los dispositivos que se pretende que pertenezcan a la misma red o, también, al conjunto de dispositivos que se querían visualizar juntos. La idea de esta *application* es: poder enlazar este conjunto de dispositivos con alguna integración que sirviera para utilizar algún servicio externo al *Network Server*, tal como puede ser el visualizar los datos recopilados desde los dispositivos.

Para crear la *application*, se tuvo que ir a la pestaña de **Applications**, dentro de la página del *Network Server* de Semtech. Dentro de esta sección, se escogió la opción **+CREATE**, tal como se puede observar en la Figura A.43.

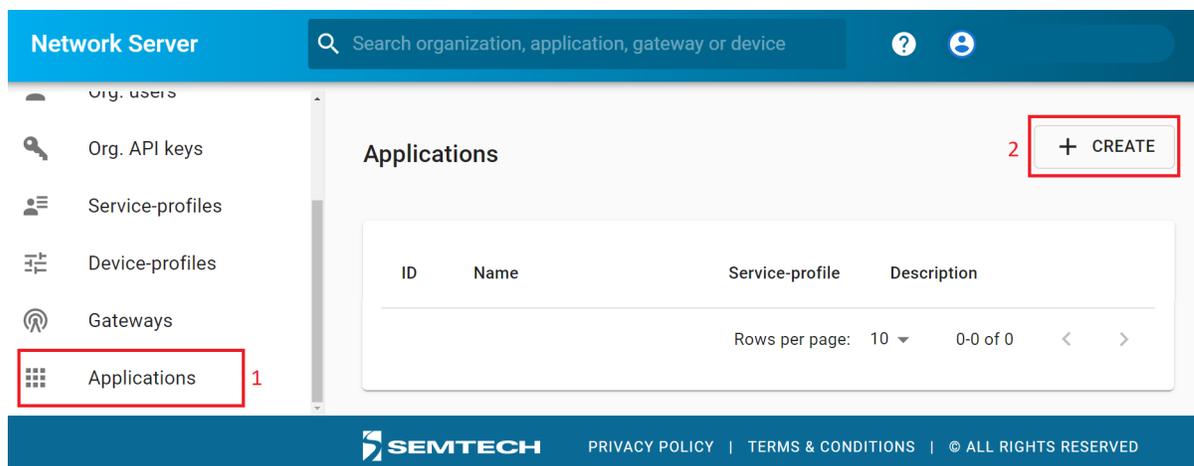


Figura A.43: Pestaña de *applications* y opción para crear una nueva aplicación.

Para poder crear la aplicación, fue necesario completar los siguientes campos: **Application name** con el nombre de la aplicación, el **Application description** con la descripción de la aplicación, y finalmente el **Service profile** que corresponde a US915 para este trabajo. Finalmente para agregar la *application* fue necesario elegir la opción **CREATE APPLICATION** y la aplicación creada se debía mostrar en la pestaña de **Applications**. Las secciones a rellenar, y el perfil creado exitosamente, se muestran en las Figuras A.44 y A.45, respecti-

vamente.

The screenshot shows the 'Network Server' interface with a search bar and a sidebar menu. The main content area contains a form with three input fields: 'Application name \*', 'Application description \*', and 'Service-profile \*'. The 'Service-profile \*' field is a dropdown menu with the text 'Select service-profile'. Below the form is a 'CREATE APPLICATION' button. The footer includes the SEMTECH logo and links for 'PRIVACY POLICY', 'TERMS & CONDITIONS', and '© ALL RIGHTS RESERVED'.

Figura A.44: Campos a rellenar para crear un aplicación.

The screenshot shows the 'Network Server' interface with the 'Applications' tab selected. A table lists the created application. The table has four columns: ID, Name, Service-profile, and Description. The first row is highlighted with a red border. Below the table, there is a 'Rows per page: 10' dropdown and '1-1 of 1' with navigation arrows. The footer includes the SEMTECH logo and links for 'PRIVACY POLICY', 'TERMS & CONDITIONS', and '© ALL RIGHTS RESERVED'.

ID	Name	Service-profile	Description
354	LR1110-application-us915	US915	LoRa Edge testing application

Figura A.45: *Application* creada con éxito.

Con todo lo anterior realizado, ya se podía agregar el transmisor del *Development kitt* al *Network Server* de Semtech.

En orden de agregar un dispositivo al *Network Server*, fue necesario ir a la pestaña de **Application** y seleccionar la aplicación previamente creada. Dentro de la aplicación, se tuvo que ir a la pestaña de **DEVICES** y seleccionar la opción **+CREATE**. La pestaña **DEVICES** se muestra en la Figura A.46.



Figura A.46: Pestaña de *devices*, dentro de la aplicación creada anteriormente.

Para crear un dispositivo dentro de la aplicación, fue necesario completar los siguientes campos; **Device name** con el nombre del dispositivo, el **Device description** con una descripción para el dispositivo, el **Device EUI** que corresponde a los datos utilizados anteriormente y que para este caso corresponde a las *keys* del transmisor y debe estar seleccionada la opción **MSB**, y finalmente el **Device-profile** y seleccionar el perfil que se creó anteriormente. Los demás campos se dejaron como estaban y para terminar con la creación se seleccionó la opción **CREATE DEVICE**. Las secciones completadas se muestran en la Figura A.47.

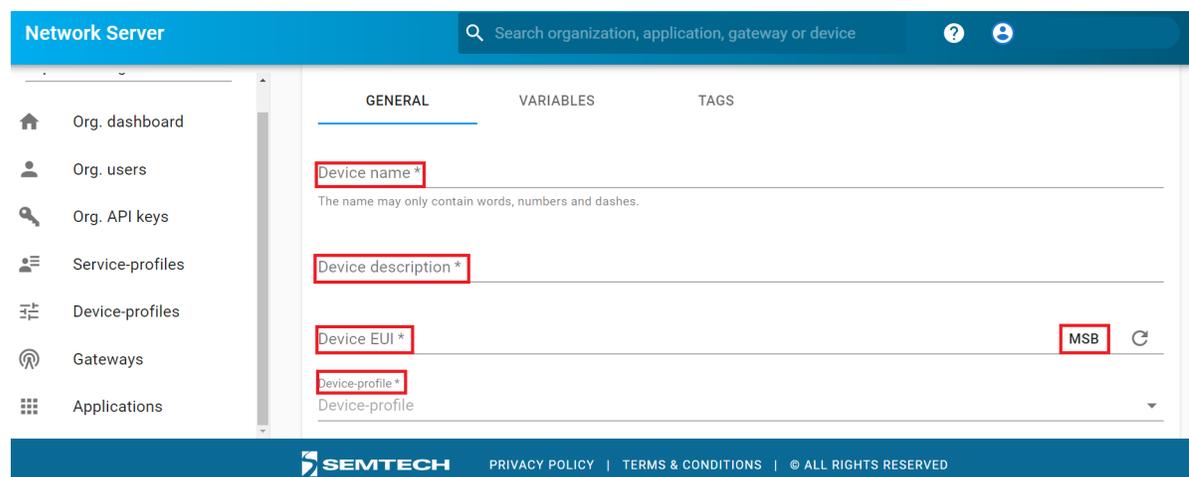


Figura A.47: Campos a rellenar para agregar un nuevo dispositivo en la aplicación.

Luego de crear el dispositivo, se desplegaba una página del mismo con información, sin embargo, no era necesario revisar esto, por lo que se tuvo que dirigir a la pestaña **Devices**, dentro de la aplicación creada, y verificar que el dispositivo fue agregado correctamente. El dispositivo agregado exitosamente se muestra en la Figura A.48.



Figura A.48: Dispositivo agregado exitosamente.

Con el dispositivo ya agregado en el **Network Server**, se procedió a realizar una integración con los servicios de *LoRa Cloud™*.

Para realizar la integración, se tuvo que ir a la aplicación creada, dentro de la sección de **Applications**, e ir a pestaña de **INTEGRATIONS**. En esta pestaña, se tuvo que buscar la integración que correspondía a *LoRa Cloud™* y seleccionar **ADD**, tal como se muestra en la Figura A.49.

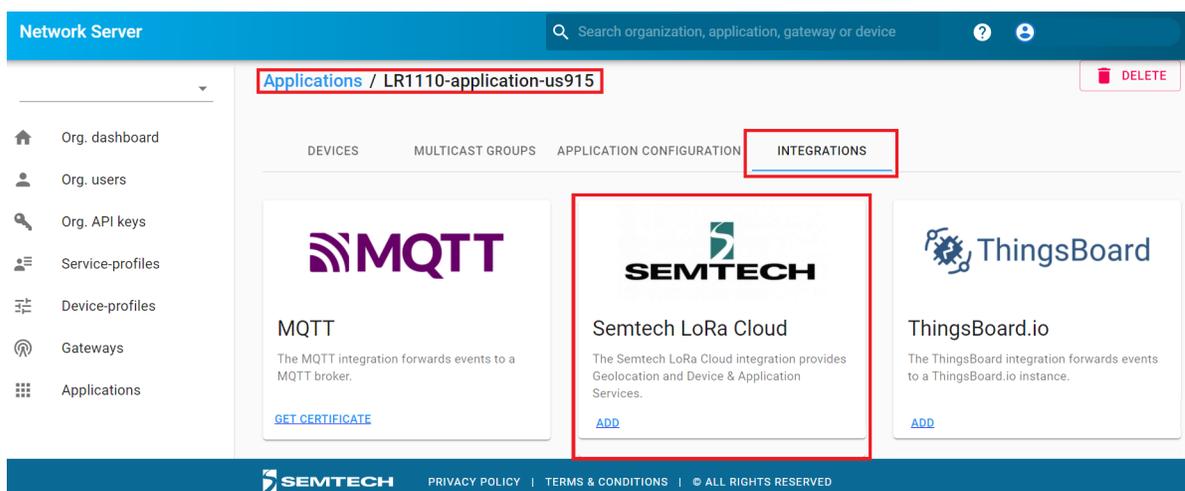


Figura A.49: Pestaña de *integrations*, dentro de la aplicación creada anteriormente, y la opción para integrar *LoRa Cloud™*.

Dentro de la integración de *LoRa Cloud™*, se seleccionó la pestaña **DEVICE & APPLICATION SERVICES**, para posteriormente marcar la opción **DAS enabled** y rellenar los siguientes campos; **Token** que corresponde al *token* previamente copiado desde *LoRa Cloud™*, el **DAS GNSS port (FPort)** se dejó vacío, el **DAS Modem port (FPort)** se dejó como 199, y finalmente se debía marcar la opción **Resolve geolocation records data stream**. El resto de los campos se dejó como estaban y finalmente se seleccionó la opción **ADD INTEGRATION**, así como se observa en la Figura A.50.

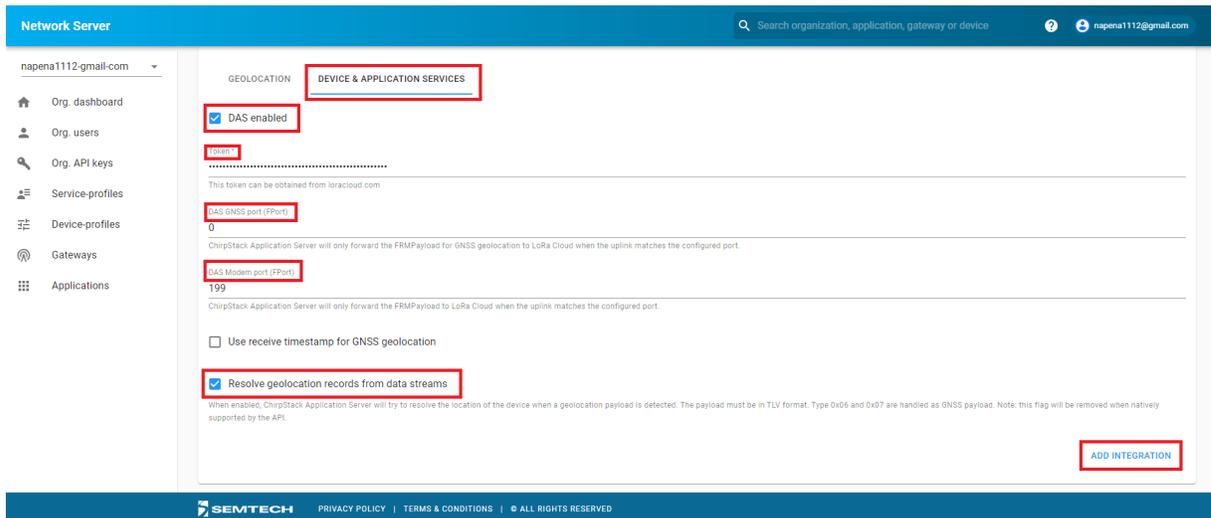


Figura A.50: Campos a rellenar para la integración de *LoRa Cloud™* y opción para agregar la integración.

Con la integración de *LoRa Cloud™* dentro del **Network Server**, ya era posible realizar algunas pruebas y divisar los resultados de la transmisión de paquetes directamente desde el *Network Server*. De esta forma ya estaba configurado todo lo necesario para poder comenzar con las pruebas de la red LoRaWAN que se estaba armando.

## A.10. Pruebas de localización del *Development Kit*

El transmisor del **Development Kit** tiene una modalidad de geolocalización, lo cual corresponde a encontrar la posición geográfica desde donde se están enviando los datos. Dicha modalidad se basa en utilizar 3 tipos de escaneo: el de GNSS autónomo, el de GNSS asistido y el de redes Wi-Fi. Básicamente se emplean el escaneo de alguna señal para poder triangular la posición desde donde se esta enviando el paquete, ya sea por medio de las IP de las redes Wi-Fi escaneadas o mediante la distancia hacia los satélites que se logren encontrar. Antes de comenzar con las pruebas de localización, se tuvo que conectar el transmisor con el gateway y permitir el acceso al *LoRa Cloud™*.

Primero se habilito el acceso al *LoRa Cloud™*, por lo que se ingresó a dicho portal y se inicio sesión. Luego, dentro de la sección **LoRa Cloud Device Join**, se escogió la opción **APPLICATION OWNERS BINDINS** desde el menú disponible. Al tener solo un perfil de *application owner*, se desplegó inmediatamente las configuraciones del perfil creado previamente, lo que mostró la sección **Bindings for 'APP\_OWN\_NAME'**, donde **APP\_OWN\_NAME** corresponde al nombre del perfil. Para poder permitir el acceso del transmisor, por medio del *Network Server* de Semtech, se marcó la casilla correspondiente a **Accept all join requests from any Network Server (not recomend)** ya que es la opción más directa para permitir el acceso desde cualquier *network server*. También, se podía escoger la ultima casilla y seleccionar los *network server* “confiables” desde una lista. Con esto realizado, ya se podía realizar la conexión con el gateway. La configuración realizada se muestra en la Figura A.51.

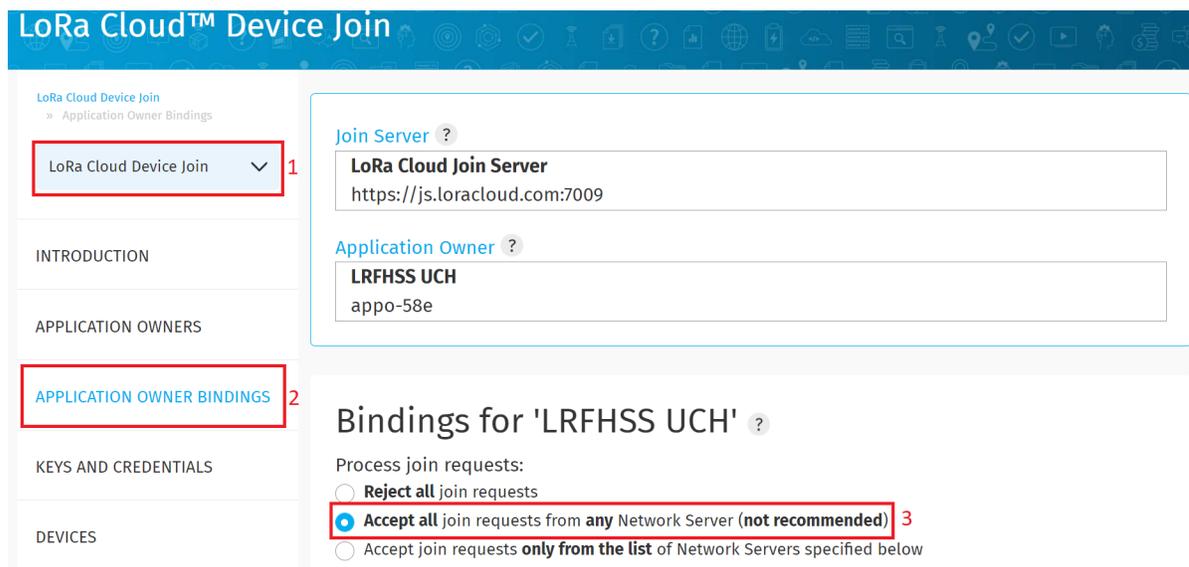


Figura A.51: Permiso necesario para poder utilizar los servicios de LoRa Cloud™.

Previo a la conexión con el gateway, primero se ingresó en el *Network Server* de Semtech y se dirigió a la sección **Applications**, luego se escogió la aplicación creada anteriormente y el dispositivo enlazado previamente.

Dentro del dispositivo en cuestión, se seleccionó la pestaña **DEVICE DATA**, que correspondía a la pestaña donde se desplegaba toda la información de los paquetes que fueran enviados y recibidos desde el dispositivo enlazado. Esta pantalla se debía mantener siempre abierta para que estuviera constantemente actualizada y poder verificar el comportamiento de los paquetes. La pestaña se muestra en la Figura A.52.

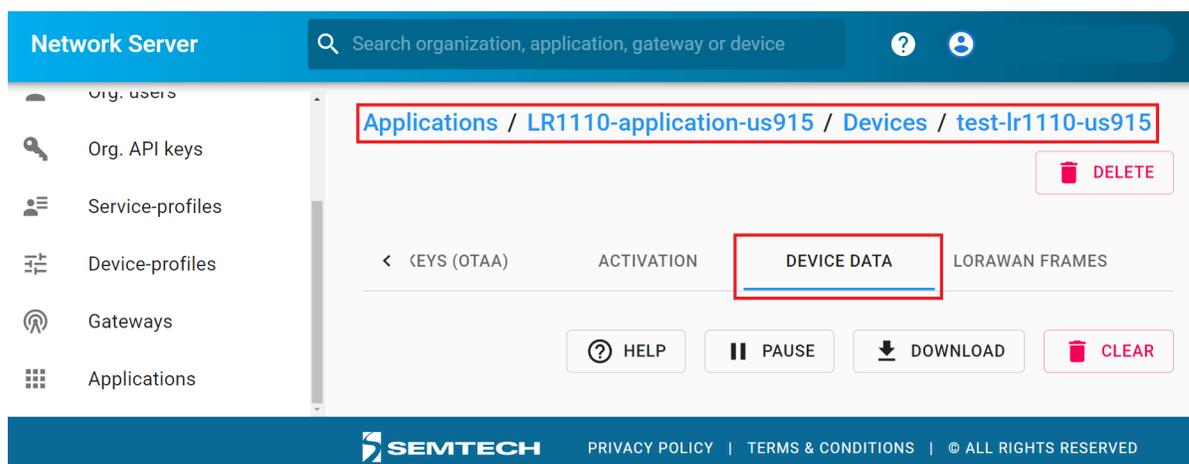


Figura A.52: Pestaña de **DEVICE DATA**, dentro del dispositivo enlazado previamente.

Para enlazar el transmisor del *Development Kit* con el gateway, era necesario energizar el transmisor y tener el gateway prendido, además de conectado con el *Network Server*. Se podía revisar la conexión del gateway yendo a la sección de **Gateways** y verificar que la última vez que haya sido visto fuera **a few seconds ago**. Luego, desde el transmisor, se tuvo que

elegir la opción **START** desde el menú de inicio y posteriormente la opción **GO!** que se encontraba al lado de la sección **CONNECTIVITY**, tal como se muestra en la Figura A.53



Figura A.53: Opción **CONNECTIVITY** dentro del transmisor del Development Kit

Una vez dentro, se desplegaba las opciones para realizar la conexión y el estado de esta. Para este caso se utilizó US915 para la opción **LoRaWAN region** y se dejó lo demás sin modificar. Finalmente se escogió la opción **JOIN** que se encontraba justo debajo del estado de conexión del transmisor. El proceso anterior se muestra en la Figura A.54.

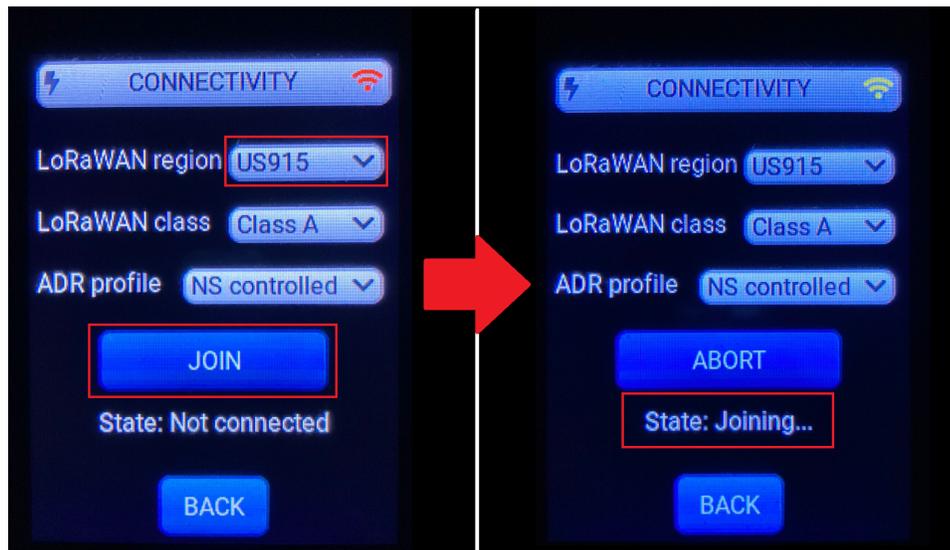


Figura A.54: Conectando el transmisor a la red LoRaWAN.

El proceso de conexión entre el transmisor y el gateway/NT se demora unos minutos. Una vez enlazado, se pudo ver el estado de **Conected** en la pantalla del transmisor y también se pudo visualizar en el *network server* el *joining request* que se envió, tal como se observa en

la Figura A.55 y la Figura A.56.

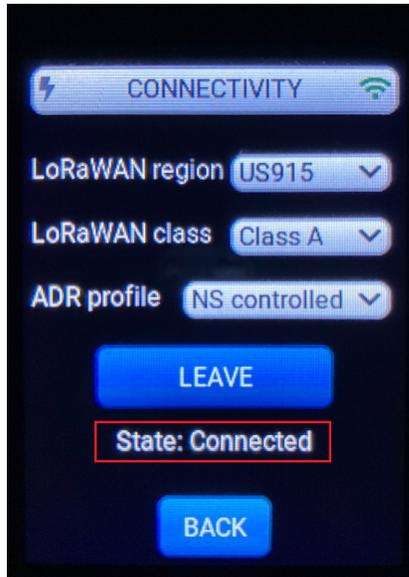


Figura A.55: Transmisor conectado a la red LoRaWAN.

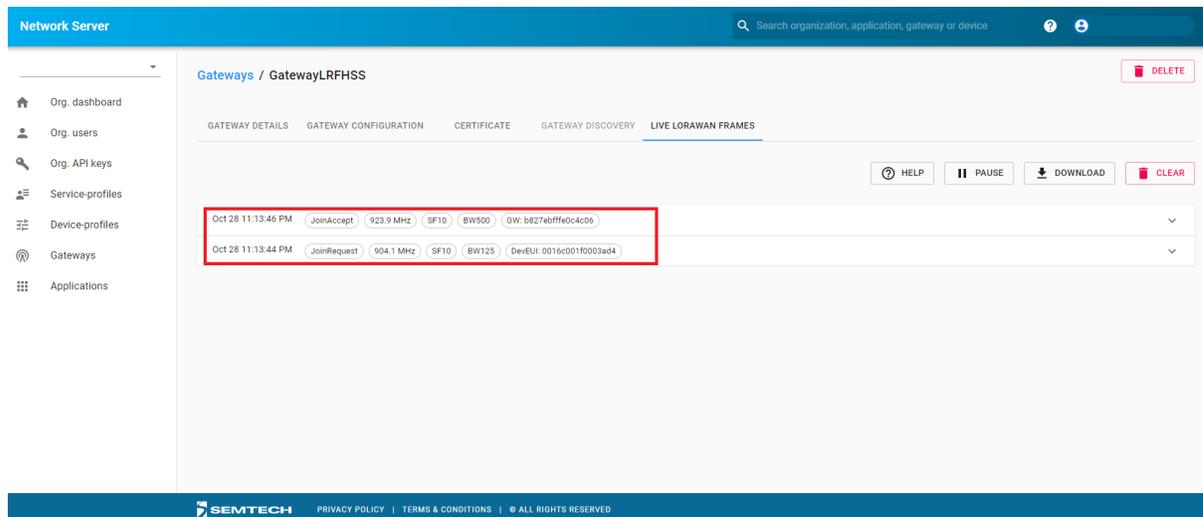


Figura A.56: Solicitud de unirse a la red LoRaWAN, hecha por el transmisor.

Con el transmisor conectado a la red LoRAWAN, ya se podía comenzar con las demostraciones con las cuales cuenta el *Development Kit*. Primero se comenzó con las pruebas de escaneo de Wi-Fi, lo cual se realizó desde la opción **Demostrations, Geolocation** y luego **Wi-Fi scan**, tal como se muestra en la Figura A.57.



Figura A.57: Seleccionando el modo de escaneo de Wi-Fi dentro del transmisor del Developmetn Kit.

Dentro de esta demostración se podía comenzar con el escaneo y luego enviar los datos hacia el *network server*, con los bonotes **START** y **SEND** respectivamente. Se debe mencionar que para poder obtener una ubicación precisa, se necesitaba encontrar al menos 3 MACs diferentes. Un escaneo, exitoso, se muestra en la Figura A.58.



Figura A.58: Escaneo de redes Wi-Fi y envío de resultados a la red LoRaWAN.

Luego de enviar el resultado del escaneo hacia el *network server*, en este último se pudo visualizar el resultado. Se debe recordar que aun se mantenía abierta la pestaña de **DEVICE DATA** para poder analizar los paquetes dentro de la red LoRaWAN. Lo que se recibió desde el transmisor/gateway fue un paquete donde salía toda la info de este mismo, por lo cual contaba con datos tales como el RSSI, SNR, posición, etc, además de contar con las configuraciones utilizadas para enviar el paquete. También se debía recibir un mensaje titulado **location** lo cual era el resultado de encontrar la posición geográfica del transmisor mediante las IP de las MAC registradas en el escaneo realizado, así como se observa en la Figura A.59.

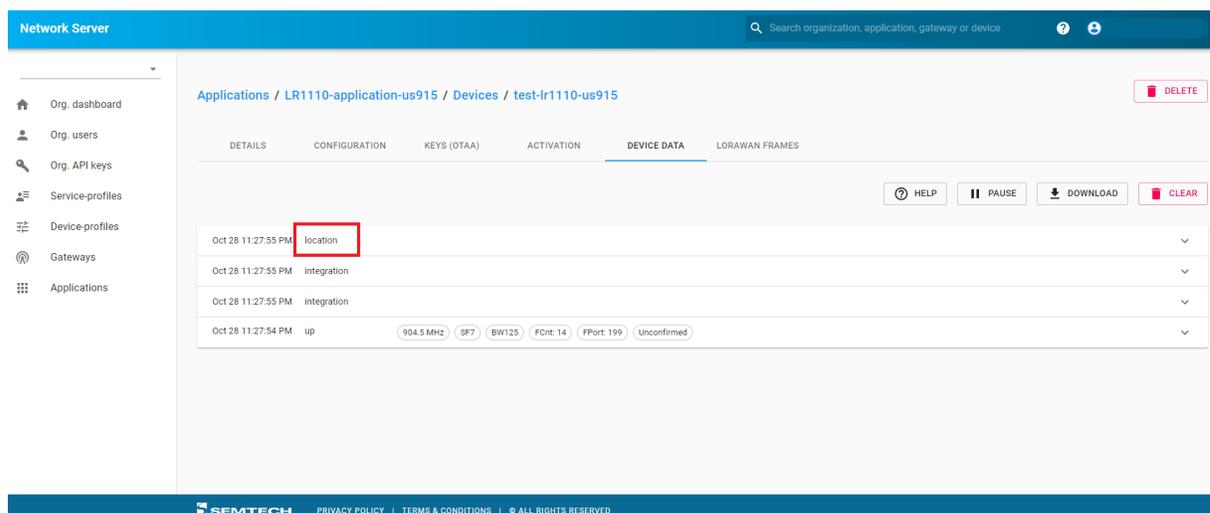


Figura A.59: Paquete recibido en el *network server*, con la posición obtenida por el escaneo de Wi-Fi.

Para el caso del escaneo de GNSS autónomo y asistido, ambos son muy similares, sin embargo el más fácil y corto es el autónomo, por lo que solo se explicara detalladamente el escaneo GNSS asistido. Cabe mencionar que el escaneo de GNSS no se podían realizar si es que el transmisor se había prendido recientemente, ya que se debía esperar un pequeño tiempo en que se sincronizara algunos parámetros. El escaneo de GNSS autónomo se realiza desde el menú de **Demostration**, *Geolocation* y eligiendo la opción **GNSS autonomous**. Luego simplemente se realizaba el escaneo con el botón **START** y se enviaron los resultados con el botón **SEND**.

En el caso del escaneo GNSS asistido, se tuvo que seleccionar la opción **GNSS assisted** dentro del menú de **Geolocation**, como se muestra en la Figura A.60. Sin embargo, antes de poder realizar el escaneo fue necesario ir al botón **ASSIST**, donde se entraba a configurar la posición geográfica desde donde se estaba realizando el escaneo, y de esta forma poder utilizar el registro de GNSS previamente actualizado. Para efectos de este trabajo, se utilizó la posición geográfica del gateway y que se podía rescatar desde el *network server* y revisando las características del gateway que se había registrado. La configuración de posición realizada se muestra en la Figura A.61.

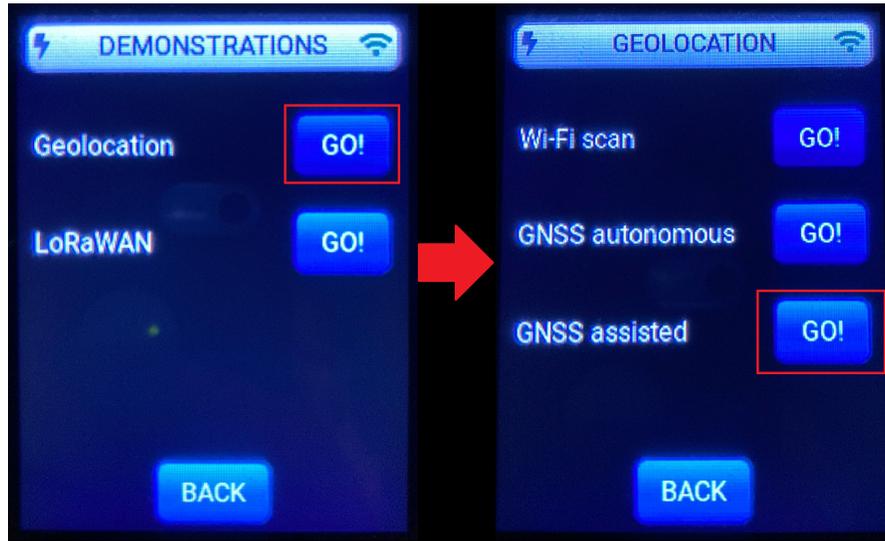


Figura A.60: Selección de escaneo de GNSS asistido.

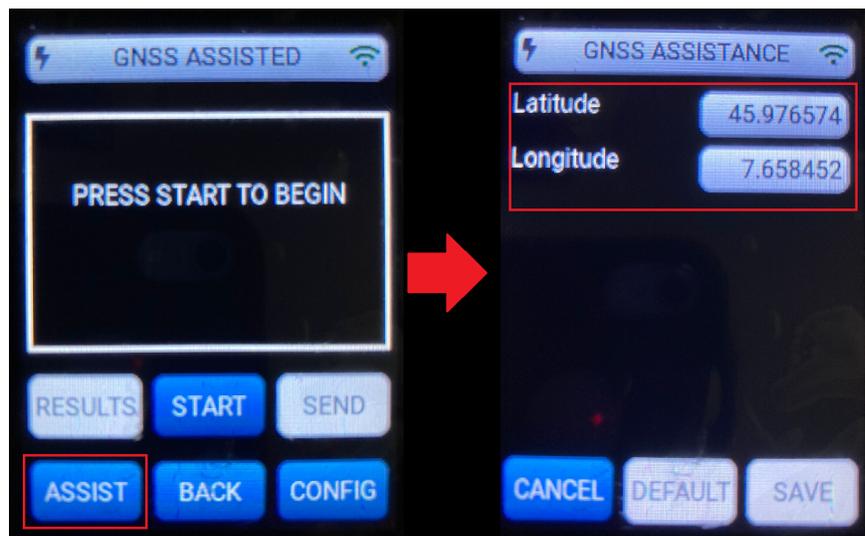


Figura A.61: “Posición” por defecto del transmisor, dentro de las opciones del escaneo asistido de GNSS.

Luego de ingresar la posición, se guardo y se selecciono el botón **CONFIG** para poder realizar las ultimas configuraciones. Tal como se muestra en la Figura A.62, dentro de este menú se seleccionó la opción **Best effort** en vez de **Low power**, para que al realizar el escaneo se intentara encontrar la mayor cantidad de satélites en vez de intentar utilizar la menor cantidad de energía posible. Esto se realizó ya que para poder encontrar la posición del transmisor, era necesario poder encontrar al menos 6 satélites.

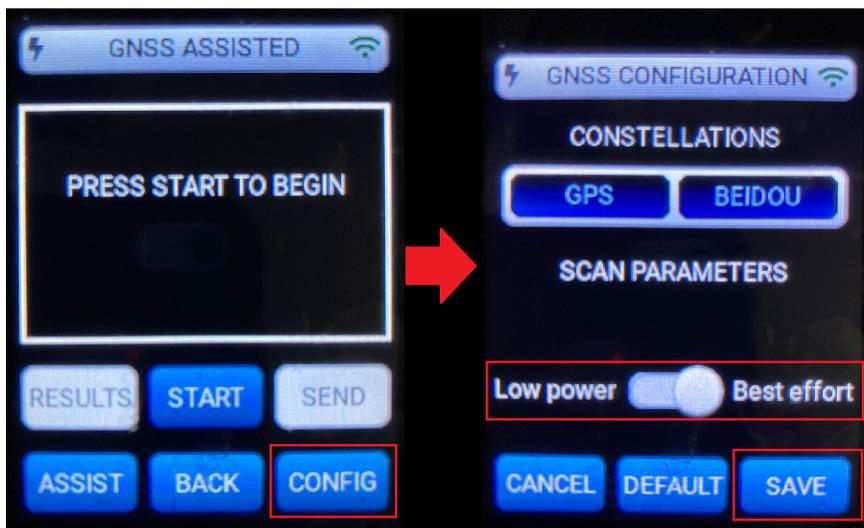


Figura A.62: Configuración el escaneo del GNSS asistido para realizar el mejor esfuerzo en vez de ahorrar energía.

Finalmente, ya era posible realizar el escaneo y esperar un buen resultado. Para poder comenzar se utilizó el botón **START** y luego de mostrar el resultado, este se mando con el botón **SEND**, tal como se puede observar en la Figura A.63. Al igual que con el escaneo de Wi-Fi, se podía visualizar los paquetes enviados y recibidos desde el transmisor, por medio de la pestaña de **DEVICE DATA** del *network server*, así como se muestra en la Figura A.64.

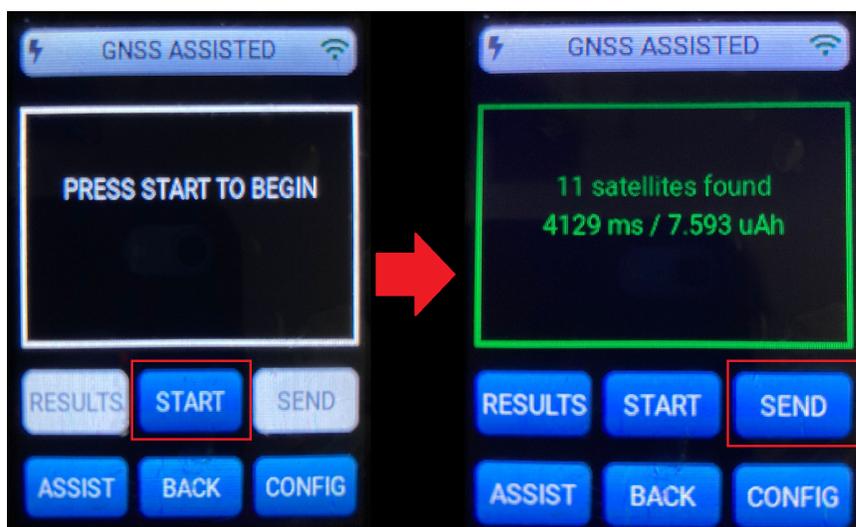


Figura A.63: Realización del escaneo asistido de GNSS y envío por medio de la red LoRaWAN.

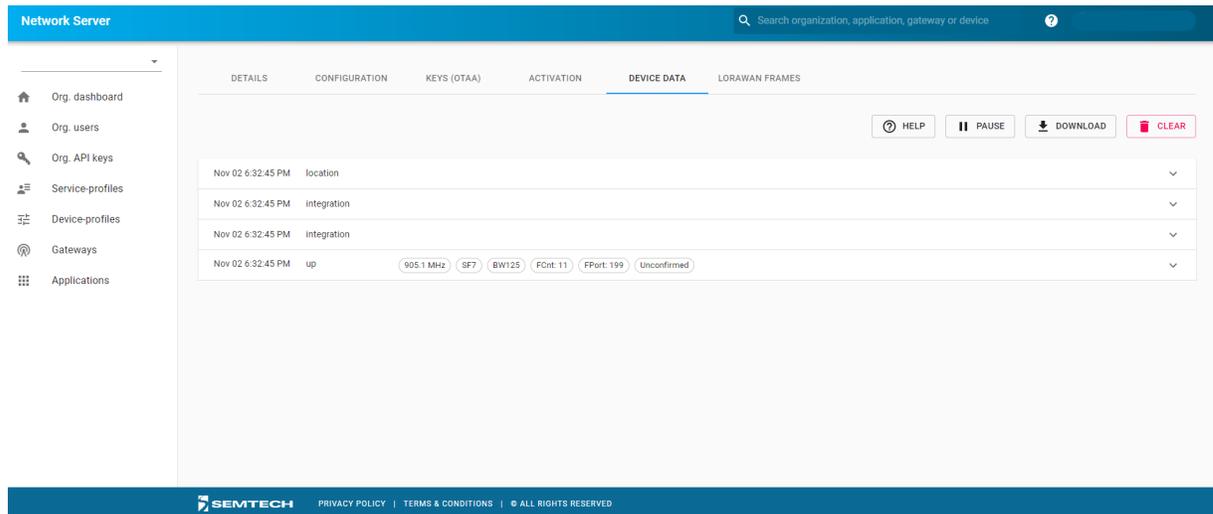


Figura A.64: Paquete recibido en el *network server*, con la posición obtenida por el escaneo asistido de GNSS.

Luego de realizar las demostraciones, se realizaron algunas mediciones y pruebas para entender como funcionaba el *Development Kit*. Los resultados se mostraran más adelante y mediante la plataforma de *ThingsBoard*.

## A.11. Visualización mediante *ThingsBoard*

Para la visualización de los datos enviados hacia el *network server*, se optó por utilizar *ThingsBoard* ya que era de simple incorporación y, además, se contaba con la seguridad de que la opción de intrigarlo por medio de las aplicaciones del *network server*.

Primero se tuvo que ingresar a la página de *ThingsBoard Demo*, la cual es <https://demo.thingsboard.io/login> e iniciar sesión. Luego de ingresar, se tuvo que dirigir a la pestaña de **Dispositivos** para poder agregar el transmisor del *Development Kit*, lo cual se lograba al seleccionar la opción + y **Agregar nuevo dispositivo**, así como se observa en la Figura A.65.

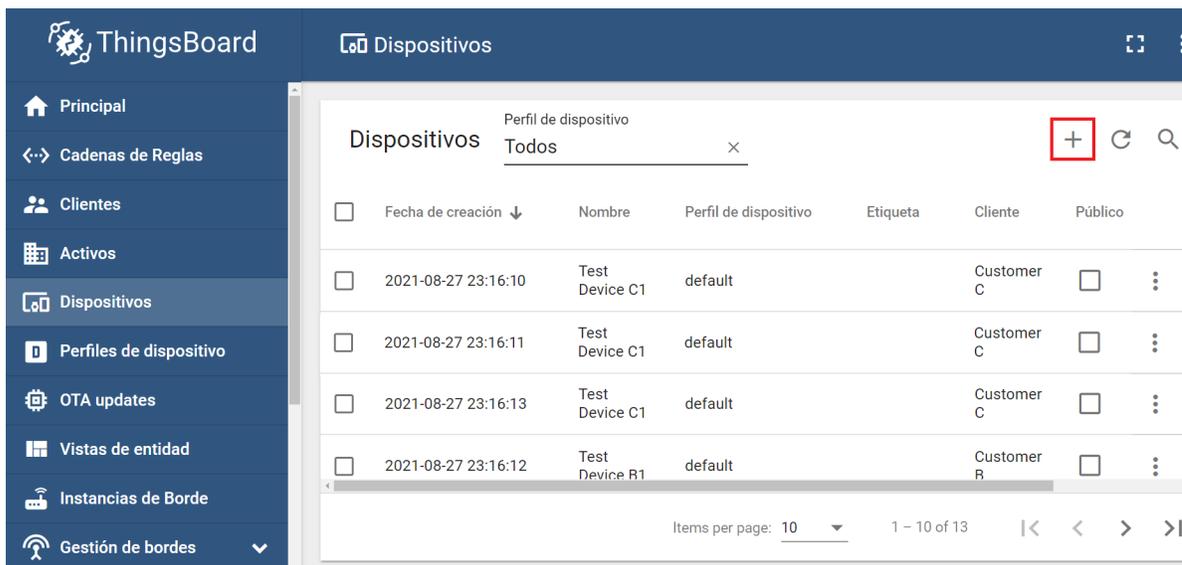


Figura A.65: Opción de agregar un nuevo dispositivo dentro de la pestaña de Dispositivos en la plataforma de *ThingsBoard Demo*.

Se tuvo que completar los campos; **Nombre** con el nombre del dispositivo, seleccionar la casilla **Crear un nuevo perfil de dispositivo** y escribir al lado el nombre de este nuevo perfil, y finalmente seleccionar la pestaña el **Configuración de transporte** y seleccionar **Por defecto**. Los demás campos se dejaron como estaban. Para agregar el dispositivo se seleccionó la opción **Agregar**, tal como se muestra en la Figura A.66.

Figura A.66: Datos necesarios para agregar un nuevo dispositivo en la plataforma de *ThingsBoard Demo*.

Una vez agregado el dispositivo, este debía aparecer dentro del listado de la pestaña **Dispositivos**. Al encontrarlo, se tuvo que ir a la casilla con un escudo, el cual representaba las credenciales del registro. Dentro de las credenciales se tuvo que almacenar el parámetro **Access token**, así como se muestra en la Figura A.67.

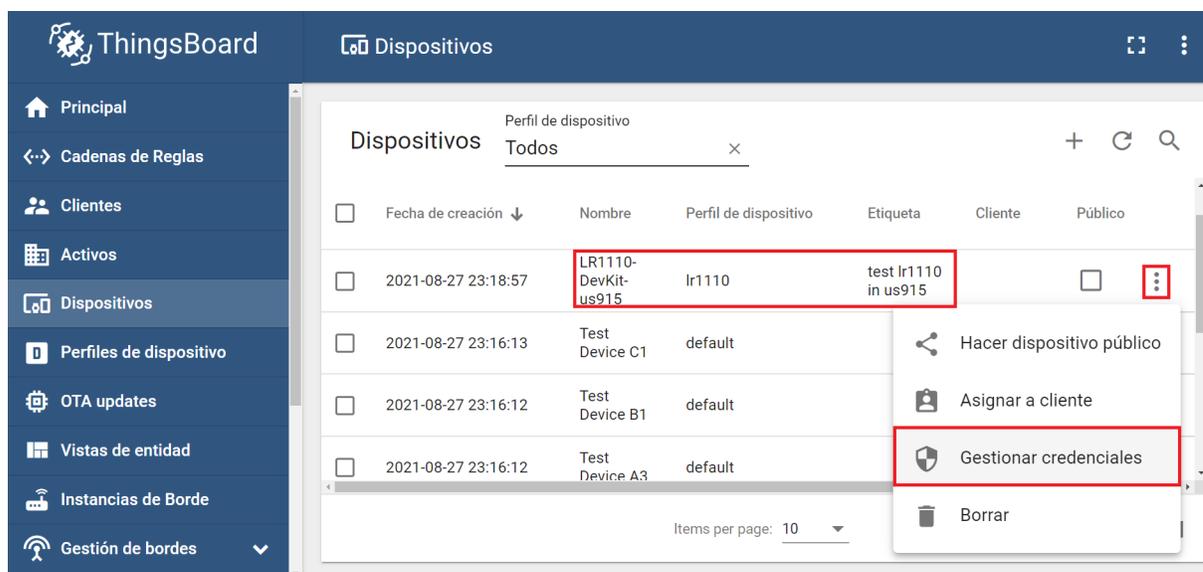


Figura A.67: Gestión de las credenciales del dispositivo agregado a la plataforma de *ThingsBoard Demo*.

Dentro del *Network Server* de Semtech, se tuvo que ir a la opción de **Applications**, luego seleccionar la aplicación utilizada anteriormente e ir a la pestaña de **CONFIGURATIONS**. Dentro de esta pestaña se tuvo que ir a la opción **VARIABLES** y seleccionar la opción **ADD VARIABLE**. El procedimiento se muestra en la Figura A.68.

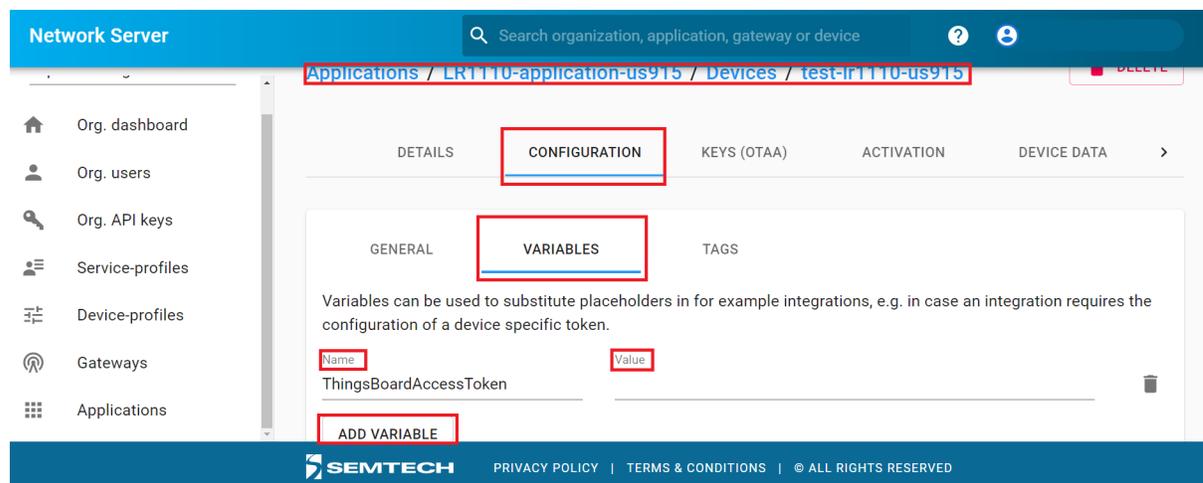


Figura A.68: Agregando el **Access token** de *ThingsBoard Demo* en el *network server* de Semtech, dentro de la aplicación creada previamente.

Para añadir una variable se tenía que especificar el nombre de esta y el valor, que para este caso se utilizó **Name:** *ThingsBoardAccessToken* y en **Value:** se introdujo el *token* previamente almacenado desde *ThingsBoard*. Finalmente se seleccionó la opción **UPDATE DEVICE**.

Continuando con la integración de *ThingsBoard*, se tuvo que ir al *Network Server* de Semtech, luego ir a la aplicación creada previamente creada y seleccionarla, para, posteriormente,

ir a la pestaña de **INTEGRATIONS**, tal como se observa en la Figura A.69.

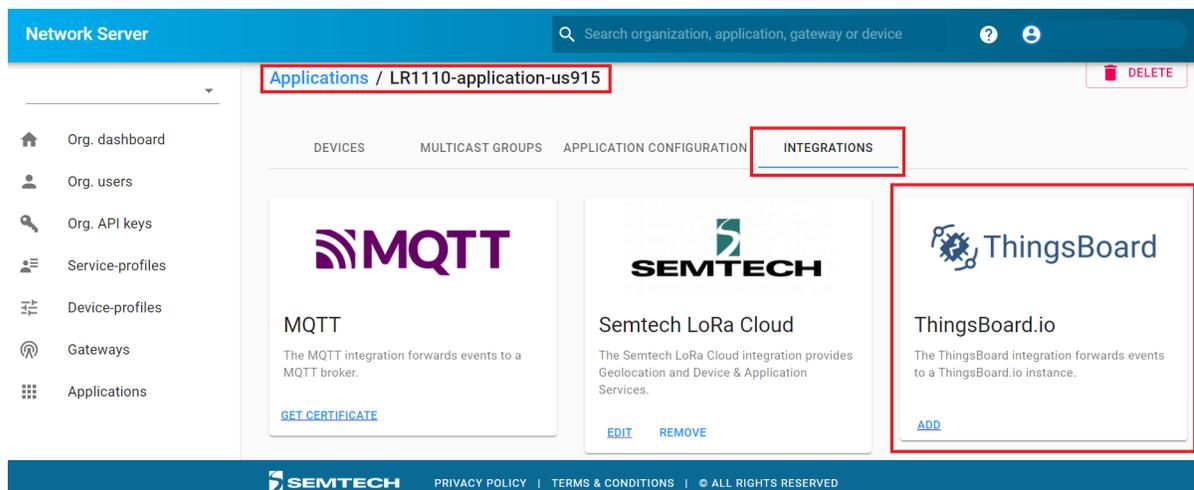


Figura A.69: Seleccionando la opción para integrar *ThingsBoard* en la aplicación del *network server* de Semtech.

Finalizando la integración, solo se tubo que llenar el campo de **ThingsBoard.io server** con `https://demo.thingsboard.io` y luego seleccionar **ADD INTEGRATION**. Al igual que como se muestra en la Figura A.70.

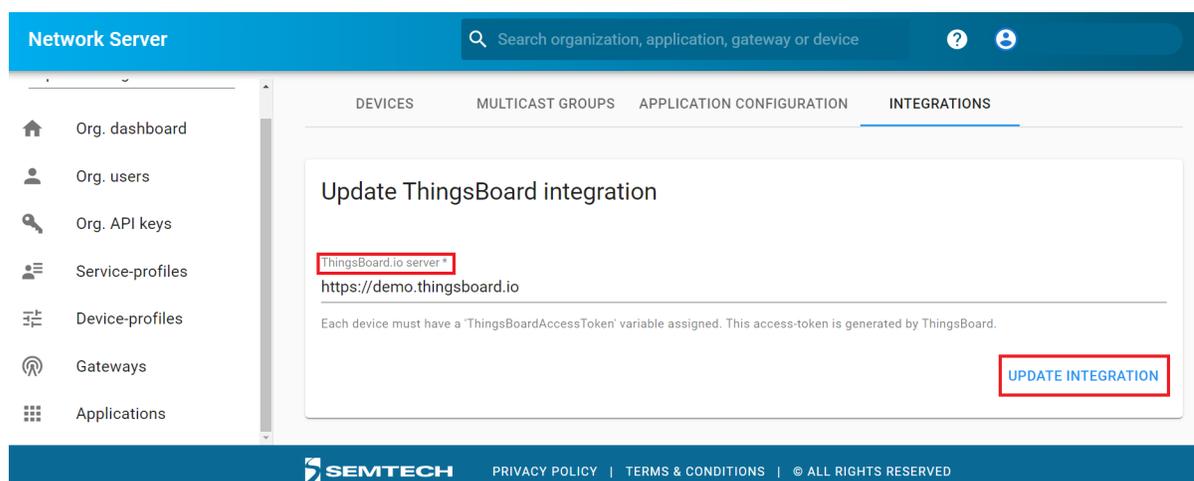


Figura A.70: Completando los campos necesarios para integrar *ThingsBoard* en la aplicación del *network server* de Semtech.

Pensando en la visualización de los resultados obtenidos en las pruebas que se realizaron previamente, primero se tuvo que seleccionar los parámetros del paquete que se iban a mostrar y la manera de visualizarlos. De esta forma, se pensó pertinente visualizar las siguientes variables

- Posición geográfica (latitud y longitud)
- RSSI

- SNR

En cuanto a la visualización, se pensó prudente mostrarlo de la siguiente forma

- Gráficos: Tanto como para el RSSI y el SNR, para los que se mostraría en su valor en función del tiempo y permitiría ver el historial.
- Valor numérico: Tanto como para el RSSI y el SNR, para los que se mostraría el último valor registrado de cada uno.
- Mapa: Se utilizaría para la posición, marcando la ultima posición registrada con un indicador y se dejaría marcado el “camino” recorrido por el transmisor, que representaría el historial de posiciones registradas

Los resultados de las configuraciones para las visualizaciones en *ThingsBoard Demo*, se muestran en la Figura A.71.

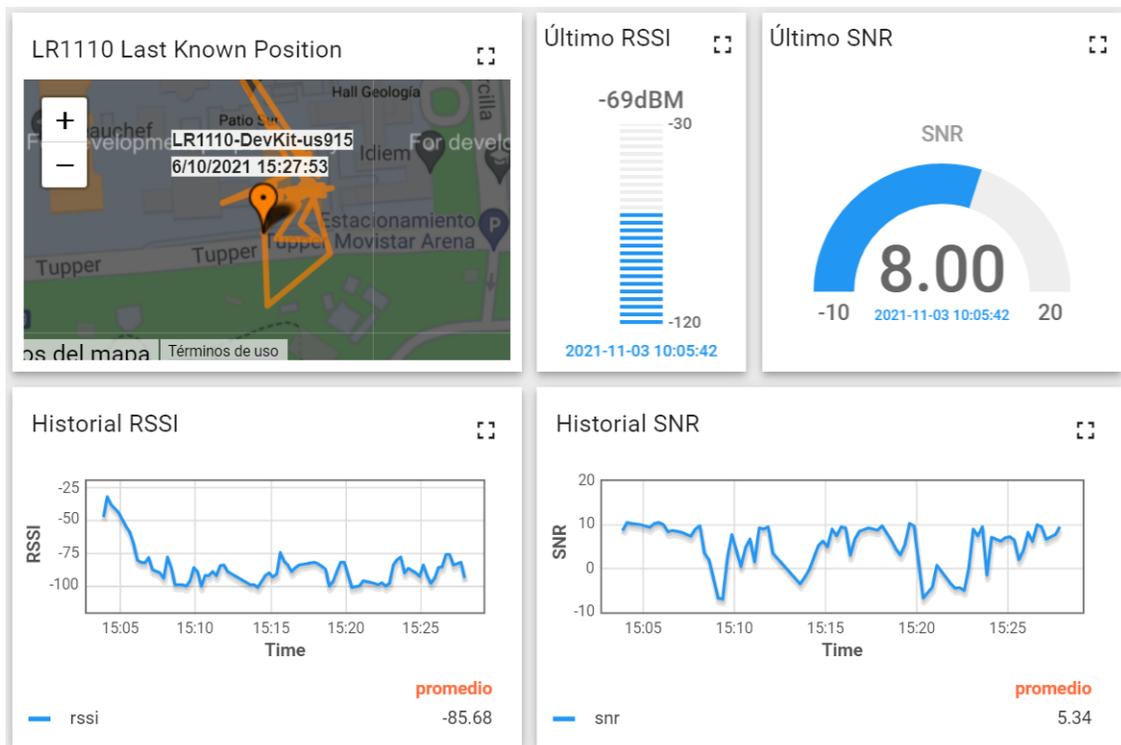


Figura A.71: Panel creado en *ThingsBoard Demo* para poder visualizar en tiempo real las variables previamente seleccionadas.