

Universidad ORT Uruguay

Facultad de Ingeniería

Monitoreo de Ganado mediante Redes

Entregado como requisito para la obtención del título de Ingeniero en Electrónica

Martín Fajardo - 229101

Octavio Revetria - 232745

Tutor: Diego Sáez

2024

Declaracion de Autoria

Nosotros, Martín Fajardo y Octavio Revetria declaramos que el trabajo que se presenta en esta obra es de nuestra propia mano. Podemos asegurar que:

- La obra fue producida en su totalidad mientras realizábamos el Proyecto final de Ingeniería en Electrónica;
- Cuando hemos consultado el trabajo publicado por otros, lo hemos atribuido con claridad;
- Cuando hemos citado trabajos de otros, hemos indicado las fuentes. Con excepción de estas citas, la obra es enteramente nuestra;
- En la obra, hemos acusado recibo de las ayudas recibidas;
- Cuando la obra se basa en trabajo realizado conjuntamente con otros, hemos explicado claramente qué fue contribuido por otros, y qué fue contribuido por nosotros;
- Ninguna parte de este trabajo ha sido publicada previamente a su entrega, excepto donde se han realizado las aclaraciones correspondientes;



Martín Fajardo

19 de Setiembre 2024



Octavio Revetria

19 de Setiembre 2024

Dedicatoria

Nuestro proyecto está dedicado a nuestros familiares y amigos, que nos han apoyado a lo largo de la carrera y durante todo el proyecto confiando plenamente en nosotros.

Agradecimientos

Queremos agradecer en primera instancia a nuestros familiares y amigos, que gracias a su apoyo incondicional fueron una fuerza de motivación a lo largo de los años. En especial, de parte de Octavio a Ricardo Revetria, Janine Schuch y Florencia Izquierdo que lo acompañaron durante todo el camino.

De parte de Martín a Laura D'Argenio por su apoyo en todo el camino, a Constantina De Barbieri por acompañar en esta etapa final. Finalmente una mención especial a Manuel Francia que fue una persona esencial en la vida y en la carrera entera, Q.E.P.D.

También queremos agradecer a un par de compañeros de facultad que comenzaron como conocidos y se volvieron grandes amigos e ingenieros como lo son Ignacio López, Gabriel Secinero y Valentina Alonso.

A la Facultad ORT por todo el apoyo, enseñanzas y conocimientos otorgados a lo largo de la carrera.

A los profesores en general por la paciencia y disposición, en particular a André Fonseca, nuestro coordinador y guía, dentro y fuera del salón de aulas. Milton Parada, excelente profesor que nos inició en este camino de la electrónica. Y por último Freddy Rabin Cymerman, gran profesor y persona que dejó una increíble huella en estos 6 años.

A la empresa AutoLink por brindar su producto y recursos, facilitando las estaciones de prueba y conocimientos para el desarrollo del proyecto.

Un agradecimiento a nuestro tutor Diego Sáez por su apoyo en este proyecto.

Abstract

En este trabajo se presenta el desarrollo de una solución integrada para la captura, transmisión y visualización de datos, con un enfoque en la optimización de energía y la eficiencia de comunicación.

Se exploraron diversas tecnologías de hardware y software, evaluando su desempeño en función de los requerimientos específicos del proyecto. A través de un enfoque modular, se diseñó un sistema que emplea protocolos de comunicación de bajo consumo, junto con una estrategia de gestión de energía para maximizar la autonomía.

El propósito principal de esta solución es asistir al productor rural en el monitoreo y control de su ganado, facilitando la localización y supervisión de los animales en áreas extensas, reduciendo así los costos operativos y mejorando la eficiencia en la gestión de los recursos.

Los resultados obtenidos demuestran la viabilidad de la solución para aplicaciones en entornos donde la optimización de recursos y la fiabilidad en la transmisión de datos son fundamentales.

Este documento detalla las fases del desarrollo, desde la investigación inicial hasta la implementación y pruebas, destacando las decisiones técnicas clave y las herramientas utilizadas.

Palabras Clave

IoT;LoRA;ISM;GPS;MQTT;RTU;PMU;FDM

Glosario

AMQP – *Advanced Message Queuing Protocol*

CAD – *Computer-Aided Design*

CAM – *Computer-Aided Manufacturing*

CAE – *Computer-Aided Engineering*

CoAP – *Constrained Application Protocol*

FDM – *Fused Deposition Modeling*

GLONASS – *Sistema Global de Navegación por Satélite*

GMT – *Greenwich Mean Time*

GNSS – *Global Navigation Satellite System*

GPS – *Global Positioning System*

HTTP – *Hypertext Transfer Protocol*

ISM – *Industrial, Scientific, and Medical*

IOT – *Internet of Things*

LTE-M – *Long Term Evolution for Machines*

LoRa – *Long Range*

MCU – *Microcontroller Unit*

MQTT – *Message Queuing Telemetry Transport*

MySQL – *Structured Query Language for Databases*

NB-IoT – *Narrowband Internet of Things*

NFC – *Near Field Communication*

OSI – *Open Systems Interconnection*

OTA – *Over-the-Air*

PMU – *Power Management Unit*

RFID – *Radio Frequency Identification*

RTOS – *Real-Time Operating System*

RTU – *Remote Terminal Unit*

SCADA – *Supervisory Control and Data Acquisition*

SLA – *Stereolithography*

TCP – *Transmission Control Protocol*

UART – *Universal Asynchronous Receiver-Transmitter*

WiFi – *Wireless Fidelity*

WWW – *World Wide Web*

Índice General

Declaracion de Autoria.....	2
Dedicatoria.....	3
Agradecimientos.....	4
Abstract.....	5
Palabras Clave.....	6
Glosario.....	7
1. Introducción:.....	13
1.1. Internet of Things.....	13
1.2. Motivación del proyecto.....	13
1.3. Objetivo General.....	14
1.3.1. Objetivos Específicos.....	14
2. Descripción General.....	15
2.1. Concepto.....	15
3. Investigación.....	18
3.1. Tecnologías de Comunicación.....	18
3.1.1. LTE-M.....	18
3.1.2. NB-IoT.....	21
3.1.3. Sigfox.....	22
3.1.3.1. ISM.....	22
3.1.3.1.1. Homologación de Frecuencias ISM en Uruguay.....	23
3.1.4. LoRa.....	23
3.1.5. ESP-NOW.....	24
3.1.6. RFID.....	28
3.1.7. NFC.....	30
3.1.8. GPS.....	32
3.1.8.1. Componentes del Sistema GPS.....	33
3.1.8.2. Precisión y Factores que Afectan al GPS.....	33
3.1.8.3. Constelaciones de Navegación Global.....	34
3.1.8.4. Ventajas de Usar Múltiples Constelaciones.....	35
3.2. Plataformas IoT.....	35
3.2.1. Dashboard.....	35
3.2.2. Thingsboard.....	37
3.2.3. NodeRed.....	39
3.2.4. Grafana.....	40
3.3. Protocolos.....	41
3.3.1. RS232.....	41
3.3.2. RS485.....	42
3.3.3. MQTT.....	43

3.3.4. HTTP.....	44
3.3.5. AMQP.....	45
3.4. Hardware.....	46
3.4.1. Microcontrolador.....	46
3.4.2. Antena.....	46
3.4.3. Gateway.....	46
3.5. Firmware.....	47
3.5.1. Framework.....	47
3.5.2. Real Time Operating System.....	48
4. Justificación de Tecnologías Elegidas.....	50
4.1. Requerimientos Técnicos.....	50
4.2. LoRa.....	50
4.3. ESP-NOW.....	50
4.4. Thingsboard.....	51
4.5. Entorno de desarrollo - Arduino IDE.....	51
4.6. Gateway.....	52
5. Desarrollo del Proyecto.....	53
5.1. Hardware elegido.....	53
5.1.2. GPS.....	54
5.1.3. Semtech SX1276.....	56
5.1.4. Batería.....	57
5.1.5. Alimentación.....	58
5.1.6. PMU AXP2101.....	59
5.2. Diseño software.....	60
5.2.1. Firmware.....	60
5.2.2. Pasamanos MQTT.....	63
5.2.3. Payload MQTT.....	64
5.3. Diseño de Carcasa.....	65
5.3.1. Fusion 360.....	65
5.3.2. Tecnología.....	67
5.3.2.1. SLA.....	67
5.3.2.2. FDM.....	68
5.3.2.3. Justificación.....	69
5.3.2. Concepto.....	71
5.3.3. Diseño Exterior.....	71
5.3.4. Protección.....	72
5.3.5. Interior.....	73
5.4. Lógica para procesamiento de los datos.....	74
5.5. Esquema de Solución.....	76
6. Pruebas de Concepto.....	78

6.1. Prueba de alcance.....	78
6.1.1. Objetivo de la Prueba.....	78
6.1.2. Justificación.....	78
6.1.3. Procedimiento.....	78
6.2. Prueba de batería.....	78
6.2.1. Objetivo de la Prueba.....	78
6.2.2. Justificación.....	79
6.2.3. Procedimiento.....	79
6.3. Prueba de GPS.....	80
6.3.1. Objetivo de la Prueba.....	80
6.3.2. Justificación.....	80
6.3.3. Procedimiento.....	80
6.4. Prueba de comunicación con Gateway.....	81
6.4.1. Objetivo de la Prueba.....	81
6.4.2. Justificación.....	81
6.4.3. Procedimiento.....	81
6.5. Prueba de integración de datos con Thingsboard.....	82
6.5.1. Objetivo de la Prueba.....	82
6.5.2. Justificación.....	82
6.5.3. Procedimiento.....	82
6.6. Configuración de Alarmas de Salida de Terreno (Geovallado).....	83
6.6.1. Objetivo de la Prueba.....	83
6.6.2. Justificación.....	83
6.6.3. Procedimiento.....	83
7. Resultados.....	84
7.1. Alcance de distancia.....	84
7.1.2. Observaciones sobre el Uso de la Carcasa.....	86
7.1.3. Observación sobre la Frecuencia de Operación.....	86
7.2. Alcance de batería.....	87
7.2.1. Despertar de Deep Sleep.....	87
7.2.2. Búsqueda de coordenadas de GPS.....	88
7.2.3. Envío por LoRa.....	89
7.2.4. Deep Sleep.....	90
7.2.5. Alcance de la autonomía del sistema.....	91
7.3. Alcance de GPS.....	92
7.4. Alcance de comunicación con gateway.....	94
7.5. Alcance de integración de datos con Thingsboard.....	96
7.6. Alcance de Alarmas de Salida de Terreno (Geovallado).....	97
8. Conclusión del MVP.....	100
9. Caso de Uso en el Monitoreo de Ganado y Plan de Mantenimiento.....	102
9.1. Implementación Práctica del Caso de Uso.....	102

9.2. Plan de Mantenimiento de los Dispositivos.....	102
10. Otros posibles usos.....	104
10.1. Tracking de Flotas Vehiculares.....	104
10.2. Monitoreo de Infraestructura Crítica.....	104
10.3. Gestión de Inventarios en Almacenes.....	104
10.4. Optimización de Riego en Agricultura.....	105
11. Mejoras para el futuro.....	106
11.1. Hardware.....	106
11.1.1. Construcción de una Placa Propia.....	106
11.1.2. Incorporación de Módulos de Energía Solar Optimizada.....	106
11.1.3. Sensores Adicionales y Mayor Capacidad de Monitoreo.....	106
11.1.4. Optimización del Diseño para Robustez Física.....	107
11.1.5. Evaluación de una Arquitectura Tipo Mesh.....	107
11.2. Software.....	107
11.2.1. Desarrollo de Algoritmos de Gestión de Energía más Avanzados.....	107
11.2.2. Ampliación de la Plataforma IoT con Análisis de Datos.....	107
11.2.3. Implementación de Actualizaciones de Firmware OTA.....	108
11.2.4. Mejoras en la Seguridad de los Datos.....	108
12. Referencias Bibliográficas.....	109
Anexos.....	117
Anexo 1: Gestión de Proyecto.....	117
Anexo 2: Problemas Técnicos con la Placa T-Beam y Cambio de Frecuencia de Operación.....	120
Anexo 3: Estimación de Costo Total.....	123

1. Introducción:

1.1. Internet of Things

Según el IERC (*European Research Cluster on the Internet of Things*) IoT (*Internet of Things*) se define como “*una infraestructura dinámica de red global con capacidades de autoconfiguración basada en protocolos de comunicación estándar e interoperables, donde los objetos físicos y virtuales tienen identidades, atributos físicos y personalidades virtuales, y utilizan interfaces inteligentes, y se integran sin problemas en la red de información.*” [1]

Bajando a tierra esta definición, IoT es una red global de dispositivos físicos que están equipados con sensores, software y diversas tecnologías de comunicación para conectar y compartir datos con otros dispositivos y sistemas a través de Internet. El propósito de IoT es permitir que estos dispositivos interactúen entre sí y con los usuarios, mejorando la eficiencia y ofreciendo nuevas funcionalidades y servicios en una variedad de contextos, desde el hogar inteligente hasta la industria y la atención médica. Lo más interesante de esta tecnología es que está al alcance de cualquier persona, ha permitido que el ciudadano común alcance logros en su hogar de manera barata y eficiente, lo que antes solo se podía con técnicos profesionales y con equipos no accesibles a cualquiera.

Esta tecnología permite una mayor automatización y control al proporcionar datos en tiempo real y permitir la toma de decisiones basada en la información recopilada por los dispositivos conectados.

Si bien esta tecnología brinda muchas ventajas, un punto controversial es la creciente preocupación por la seguridad y la privacidad. Gracias a la gran masividad de dispositivos conectados puede aumentar el riesgo de vulnerabilidades y ataques cibernéticos, que pueden dejar al alcance datos personales que posiblemente antes no eran accesibles mediante internet.

1.2. Motivación del proyecto

La necesidad de monitorear y controlar diferentes aspectos del ganado en grandes extensiones de campo, nace por dificultades en varias áreas de la ganadería. Algunas de ellas son: monitoreo de la salud del animal, su ubicación geográfica, el control del abigeato. Un

ejemplo de las necesidades previamente mencionadas se vio reflejado después de comunicarnos con la empresa Smartway, la cual manifestó la existencia de clientes con interés en soluciones de trazabilidad para el sector agropecuario. En resumen, creemos que todas estas aristas requieren de una solución eficiente y confiable en cuanto a conectividad y eficiencia energética. La posible implementación con redes celulares nos podría brindar una solución de bajo costo y bajo consumo energético, permitiendo a su vez la interacción entre los diversos componentes y la nube.

Hoy en día, los sistemas de chips en las orejas del ganado más adoptados mundialmente, utilizan RFID o NFC pero estas tecnologías cuentan con varios problemas:

- Necesidad de equipos costosos e infraestructura para la obtención de datos.
- No es posible lograr una conectividad en tiempo real.
- Carecen de medios eficientes para controlar el abigeato, ya que no pueden brindar alertas.

1.3. Objetivo General

Diseñar e implementar un sistema basado en redes para el monitoreo y control ganadero en áreas rurales del Uruguay.

1.3.1. Objetivos Específicos

- Diseñar e implementar un sistema basado en redes para el monitoreo y control ganadero en áreas rurales del Uruguay.
- Investigar las tecnologías capaces de brindar conectividad en las distintas áreas remotas.
- Diseñar e implementar un dispositivo que transmita datos empaquetando dicha información en un mensaje, para enviar por la red elegida para luego poder ser consumidos por un cliente remoto.
- Realizar pruebas de integración donde se obtengan datos de sensores, se los procese y se envíe a la unidad receptora.
- Lograr una total integración con el software.

2. Descripción General

En este capítulo se describirán de manera básica los principales elementos que conforman la solución creada.

2.1. Concepto

Analizando el problema, tenemos animales que se mueven por una zona muy amplia y queremos saber su ubicación. Es una zona en la cual no hay electricidad ni conectividad normal, sea celular o WiFi. Para ello tenemos que crear una red, a la cual los diversos nodos (animales) puedan conectarse para enviar su ubicación exacta.

Una vez obtenidos los datos en las unidades centralizadoras, estos se suben a cualquier nube/servidor para que sean consumidos.

Si bien no es la solución de IoT completa, el proyecto tiene como objetivo y alcance la obtención de información para ser enviada. Para ello el dispositivo consta de un módulo de recepción de datos, el módulo de procesamiento ordena los datos y les da un formato determinado, luego se los envía al módulo de comunicación que es el responsable de enviar toda esta información a un servicio en la nube.

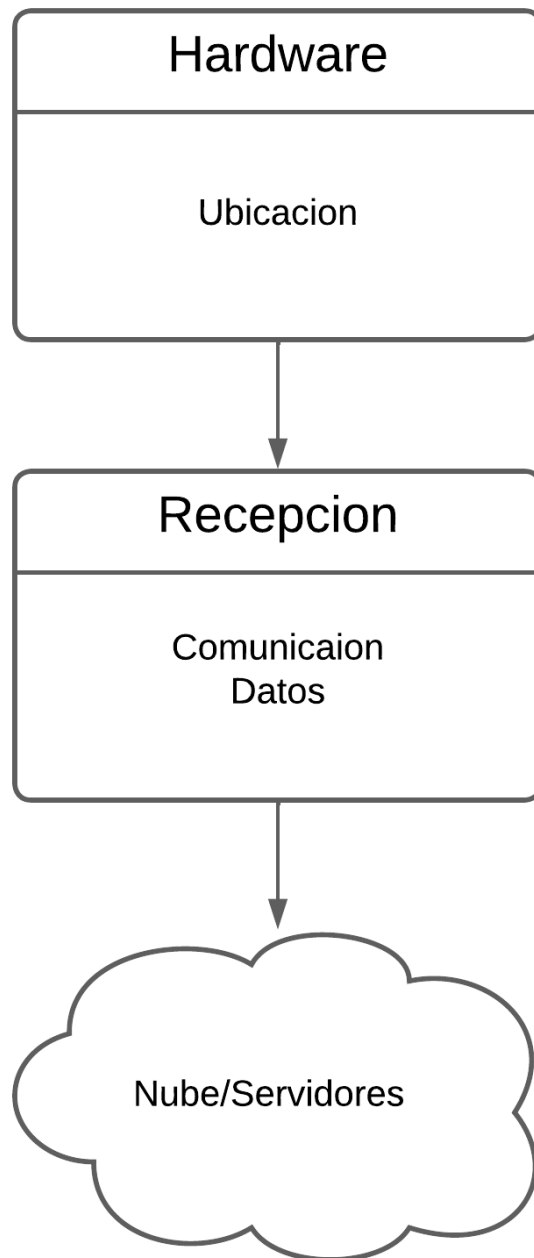


Figura 2.1: Concepto solución

Para dejarlo en claro la solución se basa en unidades que envían datos, se reciben en un centralizador y de allí se suben.

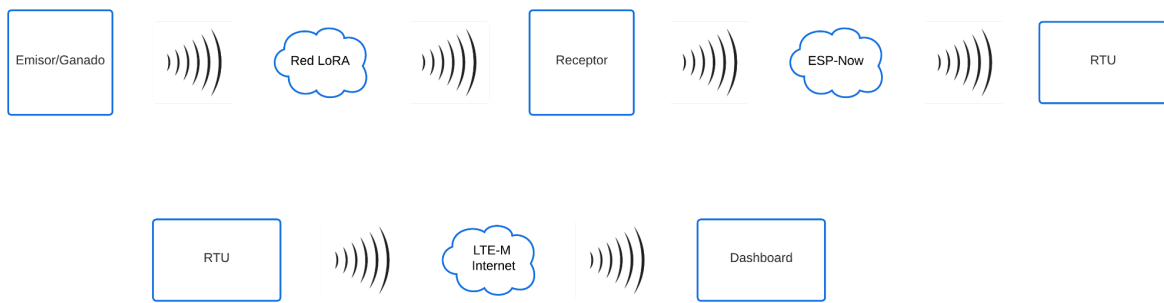


Figura 2.2: Concepto solución

3. Investigación

En este capítulo de investigación se presentarán las distintas tecnologías evaluadas durante la fase inicial del proyecto. Se exploraron diversas opciones de comunicación y plataformas para la visualización de datos, así como múltiples componentes de hardware. El objetivo fue analizar las características de cada opción y comprender sus ventajas y limitaciones en función de las necesidades del proyecto. Cabe destacar que en este capítulo no se tomó ninguna decisión definitiva, sino que se realizó una revisión exhaustiva de las alternativas disponibles, brindando un marco técnico amplio que facilitó la selección de las soluciones más adecuadas en etapas posteriores.

3.1. Tecnologías de Comunicación

3.1.1. LTE-M

LTE-M, (*Long Term Evolution for Machines*) categoría M1, es una tecnología de comunicación diseñada específicamente para el Internet de las Cosas (IoT) y otras aplicaciones de comunicación de máquina a máquina (M2M). Es una variante de la red LTE que optimiza la conectividad para dispositivos IoT que requieren una eficiencia en el consumo de energía, una cobertura amplia y una comunicación de bajo costo.

Al mismo tiempo, permite la reutilización de la base instalada de LTE. Como resultado, se obtiene una mayor vida útil de la batería para una amplia gama de casos de uso.

Algunas de sus principales características son:

- Amplio espectro de dispositivos.
- Duración prolongada de la batería.
- Cobertura mejorada.
- Moderada velocidad de datos.
- Compatibilidad con infraestructura existente [2]

En resumen, LTE-M es una tecnología de comunicación robusta y eficiente para IoT, especialmente adecuada para aplicaciones que requieren bajo consumo de energía, cobertura amplia, y transmisión de datos moderada.

Particularmente en Uruguay, las frecuencias asignadas para las Telecomunicaciones Móviles Internacionales, IMT por sus siglas en inglés, están dadas por la URSEC.

Asignatario	Cond. asignación	Plazo	Vencimiento	Sub-bloques Ida	Sub-bloques Vuelta
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	28 Sep 2037	703 MHz - 708 MHz	758 MHz - 763 MHz
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	28 Sep 2037	708 MHz - 713 MHz	763 MHz - 768 MHz
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	28 Sep 2037	713 MHz - 718 MHz	768 MHz - 773 MHz
A.M. WIRELESS URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	28 Sep 2037	718 MHz - 723 MHz	773 MHz - 778 MHz
A.M. WIRELESS URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	28 Sep 2037	723 MHz - 728 MHz	778 MHz - 783 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 20 años	28 Sep 2037	728 MHz - 733 MHz	783 MHz - 788 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 20 años	28 Sep 2037	733 MHz - 738 MHz	788 MHz - 793 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 20 años	28 Sep 2037	738 MHz - 743 MHz	793 MHz - 798 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 20 años	28 Sep 2037	743 MHz - 748 MHz	798 MHz - 803 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional, precaria y revocable	Sin plazo		824 MHz - 835 MHz	869 MHz - 880 MHz
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	24 Jul 2024	835 MHz - 845 MHz	880 MHz - 890 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional, precaria y revocable	Sin plazo		845 MHz - 846.5 MHz	890 MHz - 891.5 MHz
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	24 Jul 2024	846.5 MHz - 849 MHz	891.5 MHz - 894 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 20 años	30 Apr 2033	910 MHz - 915 MHz	955 MHz - 960 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 20 años	01 Jul 2023	1710 MHz - 1715 MHz	1805 MHz - 1810 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 20 años	01 Jul 2023	1715 MHz - 1720 MHz	1810 MHz - 1815 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 20 años	31 Dec 2026	1720 MHz - 1725 MHz	1815 MHz - 1820 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 25 años	28 Dec 2045	-	1820 MHz - 1830 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 25 años	28 Dec 2045	-	1830 MHz - 1840 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 25 años	28 Dec 2045	-	1840 MHz - 1850 MHz
A.M. WIRELESS URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	10 Apr 2033	1855 MHz - 1860 MHz	1935 MHz - 1940 MHz

Figura 3.1: Asignaciones de frecuencias [3]

Asignatario	Cond. asignación	Plazo	Vencimiento	Sub-bloques Ida	Sub-bloques Vuelta
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	10 Apr 2033	1860 MHz - 1865 MHz	1940 MHz - 1945 MHz
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	10 Apr 2033	1865 MHz - 1870 MHz	1945 MHz - 1950 MHz
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	10 Apr 2033	1870 MHz - 1875 MHz	1950 MHz - 1955 MHz
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	10 Apr 2033	1875 MHz - 1880 MHz	1955 MHz - 1960 MHz
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	24 Jul 2024	1880 MHz - 1885 MHz	1960 MHz - 1965 MHz
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	11 Dec 2042	1885 MHz - 1890 MHz	1965 MHz - 1970 MHz
A.M. WIRELESS URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	10 Apr 2033	1890 MHz - 1895 MHz	1970 MHz - 1975 MHz
A.M. WIRELESS URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	23 Jun 2024	1895 MHz - 1900 MHz	1975 MHz - 1980 MHz
A.M. WIRELESS URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	23 Jun 2024	1900 MHz - 1910 MHz	1980 MHz - 1990 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 20 años	28 Sep 2037	1920 MHz - 1925 MHz	2110 MHz - 2115 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 20 años	17 Jul 2027	1925 MHz - 1930 MHz	2115 MHz - 2120 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 20 años	28 Sep 2037	-	2120 MHz - 2125 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 20 años	30 Apr 2033	1725 MHz - 1730 MHz	2125 MHz - 2130 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 20 años	30 Apr 2033	1730 MHz - 1735 MHz	2130 MHz - 2135 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 20 años	30 Apr 2033	1735 MHz - 1740 MHz	2135 MHz - 2140 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 20 años	30 Apr 2033	1740 MHz - 1745 MHz	2140 MHz - 2145 MHz
A.M. WIRELESS URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	10 Apr 2033	1745 MHz - 1750 MHz	2145 MHz - 2150 MHz
A.M. WIRELESS URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	10 Apr 2033	1750 MHz - 1755 MHz	2150 MHz - 2155 MHz
A.M. WIRELESS URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	28 Sep 2037	1755 MHz - 1760 MHz	2155 MHz - 2160 MHz
A.M. WIRELESS URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 20 años	28 Sep 2037	1760 MHz - 1765 MHz	2160 MHz - 2165 MHz
A.M. WIRELESS URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 25 años	12 Mar 2045	1765 MHz - 1770 MHz	2165 MHz - 2170 MHz
A.M. WIRELESS URUGUAY S.A.	Exclusiva, nacional y con plazo	Por 25 años	12 Mar 2045	1770 MHz - 1775 MHz	2170 MHz - 2175 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 25 años	28 Dec 2045	2510 MHz - 2515 MHz	2630 MHz - 2635 MHz

Figura 3.2: Asignaciones de frecuencias [3]

Asignatario	Cond. asignación	Plazo	Vencimiento	Sub-bloques Ida	Sub-bloques Vuelta
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 25 años	28 Dec 2045	2515 MHz - 2520 MHz	2635 MHz - 2640 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 25 años	28 Dec 2045	2520 MHz - 2525 MHz	2640 MHz - 2645 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 25 años	28 Dec 2045	2525 MHz - 2530 MHz	2645 MHz - 2650 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional y con plazo	Por 25 años	28 Dec 2045	2530 MHz - 2535 MHz	2650 MHz - 2655 MHz
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, en 13 departamentos y con plazo	Por 25 años	12 Aug 2045	2535 MHz - 2540 MHz	2655 MHz - 2660 MHz
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, en 6 Departamentos y con plazo	Por 25 años	31 Dec 2046	2535 MHz - 2540 MHz	2655 MHz - 2660 MHz
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, en 13 Departamentos y con plazo	Por 25 años	12 Aug 2045	2540 MHz - 2545 MHz	2660 MHz - 2665 MHz
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, en 6 Departamentos y con plazo	Por 25 años	31 Dec 2046	2540 MHz - 2545 MHz	2660 MHz - 2665 MHz
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, en 13 Departamentos y con plazo	Por 25 años	12 Aug 2045	2545 MHz - 2550 MHz	2665 MHz - 2670 MHz
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, en 6 Departamentos y con plazo	Por 25 años	31 Dec 2046	2545 MHz - 2550 MHz	2665 MHz - 2670 MHz
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, en 13 Departamentos y con plazo	Por 25 años	12 Aug 2045	2550 MHz - 2555 MHz	2670 MHz - 2675 MHz
TELEFONICA MOVILES DEL URUGUAY S.A.	Exclusiva, en 6 Departamentos y con plazo	Por 25 años	31 Dec 2046	2550 MHz - 2555 MHz	2670 MHz - 2675 MHz
ADMINISTRACION NACIONAL DE TELECOMUNICACIONES	Exclusiva, nacional, precaria y revocable	Sin plazo		27.5 GHz - 28.35 GHz	27.5 GHz - 28.35 GHz

Figura 3.3: Asignaciones de frecuencias [3]

5G/4G Bands			
Band Number	5GNR / FR1 / LTE / LTE-Advanced / WCDMA / HSPA / HSPA+ / TD-SCDMA		
	Uplink	Downlink	Covered
1	UL: 1920 to 1980	DL: 2110 to 2170	✓
2	UL: 1850 to 1910	DL: 1930 to 1990	✓
3	UL: 1710 to 1785	DL: 1805 to 1880	✓
4	UL: 1710 to 1755	DL: 2110 to 2155	✓
5	UL: 824 to 849	DL: 869 to 894	✓
7	UL: 2500 to 2570	DL: 2620 to 2690	✓
8	UL: 880 to 915	DL: 925 to 960	✓
9	UL: 1749.9 to 1784.9	DL: 1844.9 to 1879.9	✓
11	UL: 1427.9 to 1447.9	DL: 1475.9 to 1495.9	✗
12	UL: 699 to 716	DL: 729 to 746	✓
13	UL: 777 to 787	DL: 746 to 756	✓
14	UL: 788 to 798	DL: 758 to 768	✓
17	UL: 704 to 716	DL: 734 to 746	✓
18	UL: 815 to 830	DL: 860 to 875	✓
19	UL: 830 to 845	DL: 875 to 890	✓
20	UL: 832 to 862	DL: 791 to 821	✓
21	UL: 1447.9 to 1462.9	DL: 1495.9 to 1510.9	✗
22	UL: 3410 to 3490	DL: 3510 to 3590	✗
23	UL: 2000 to 2020	DL: 2180 to 2200	✓
24	UL: 1625.5 to 1660.5	DL: 1525 to 1559	✓
25	UL: 1850 to 1915	DL: 1930 to 1995	✓
26	UL: 814 to 849	DL: 859 to 894	✓
27	UL: 807 to 824	DL: 852 to 869	✓
28	UL: 703 to 748	DL: 758 to 803	✓
29	UL: -	DL: 717 to 728	✓
30	UL: 2305 to 2315	DL: 2350 to 2360	✓
31	UL: 452.5 to 457.5	DL: 462.5 to 467.5	✗
32	UL: -	DL: 1452 - 1496	✓
35		1850 to 1910	✓
38		2570 to 2620	✓
39		1880 to 1920	✓
40		2300 to 2400	✓
41		2496 to 2690	✓
42		3400 to 3600	✗
43		3600 to 3800	✗
48		3550 to 3700	✗
66	UL: 1710-1780	DL: 2110-2200	✓
71		617 to 698	✗
74/75/76		1427 to 1518	✗
78		3300 to 3800	✗
79		4400 to 5000	✗
85	698-716	728-746	✓

Figura 3.4: Rango de frecuencias por banda [4]

Observando las figuras 3.1, 3.2, 3.3 y 3.4, se concluye que en Uruguay se encuentran asignadas las siguientes bandas:

- B28
- B5
- B8
- B3
- B2
- B4 / B66
- B7

3.1.2. NB-IoT

NB-IoT (*Narrowband IoT*) es una tecnología de comunicación inalámbrica diseñada específicamente para aplicaciones de Internet de las Cosas que requieren bajo consumo de energía, cobertura amplia y la capacidad de conectar un gran número de dispositivos. Es parte de las redes LPWA (*Low Power Wide Area*), que operan en zonas extensas utilizando un ancho de banda estrecho y baja potencia. [5]

NB-IoT opera utilizando frecuencias no utilizadas por las redes de comunicación tradicionales, lo que reduce las interferencias y mejora la cobertura. Una de las ventajas clave de esta tecnología es su capacidad para operar con una potencia de transmisión muy baja, lo que extiende la vida útil de las baterías de los dispositivos y permite una cobertura teórica global. [6]

Tomando en cuenta que no se necesitan grandes velocidades de conexión, sino conexiones estables, el estándar permite transferencias de hasta 250 kbit/s y una latencia de entre 1.6 y 10 segundos, y dependiendo del uso, la batería de los chips puede alargarse hasta los 10 años. [7]

En resumen, las características principales de NB-IoT son:

- Eficiencia energética
- Muy buena cobertura
- Bajo Costo de materiales
- Compatibilidad y escalabilidad

3.1.3. Sigfox

Sigfox es una tecnología de comunicación diseñada específicamente para el Internet de las Cosas, enfocada en ofrecer conectividad de bajo costo y bajo consumo de energía para dispositivos que requieren enviar pequeñas cantidades de datos de manera intermitente. A diferencia de otras tecnologías de comunicación que se centran en ofrecer altas velocidades de transmisión y grandes anchos de banda, Sigfox se especializa en cubrir grandes áreas geográficas con una infraestructura simple y eficiente.

La red Sigfox pertenece al grupo de redes LPWAN (*Low Power Wide Area Network*), que son redes de baja potencia y amplio alcance. Este tipo de red está optimizado para dispositivos que operan con baterías y que necesitan durar varios años sin mantenimiento. Sigfox logra esto operando en la banda ISM (*Industrial, Scientific, and Medical*), una banda de frecuencia libre de licencia, lo que permite que los dispositivos transmitan datos utilizando muy poca energía. [8]

Una de las razones por las que Sigfox es tan eficiente es su enfoque en la simplicidad. Los dispositivos Sigfox no necesitan realizar conexiones complicadas o intercambiar grandes volúmenes de datos. En cambio, envían pequeños mensajes que la red recoge y transmite a las aplicaciones de los usuarios finales. Esta simplicidad no solo reduce el consumo de energía, sino que también minimiza los costos de operación y mantenimiento.

3.1.3.1. ISM

Las bandas ISM son un conjunto de bandas de radiofrecuencia que se reservan globalmente para el uso en aplicaciones industriales, científicas y médicas, pero también se utilizan ampliamente en dispositivos de comunicación inalámbrica, como los que operan en redes Wi-Fi, Bluetooth, y tecnologías IoT como Sigfox y LoRa.

Frecuencias ISM	Longitud de onda
433,92 MHz \pm 0.2%	69,14 cm
915 MHz \pm 13 MHz	32,75 cm
2450 MHz \pm 50 MHz	12,24 cm
5800 MHz \pm 75 MHz	5,17 cm
24125 MHz \pm 125 MHz	1,36 cm

Figura 3.5: Frecuencias ISM [9]

3.1.3.1.1. Homologación de Frecuencias ISM en Uruguay

En Uruguay, las bandas de frecuencia designadas para aplicaciones industriales, científicas y médicas (ISM) están reguladas y homologadas, lo que permite su uso dentro del marco establecido. Según el cuadro de atribución de frecuencias y la normativa vigente de la URSEC, la banda de 902-928 MHz está homologada para aplicaciones ISM en la Región 2, y su uso debe respetar las disposiciones de aceptación de interferencias perjudiciales que puedan resultar de estas aplicaciones. [10]

3.1.4. LoRa

LoRa (Long Range) es una tecnología desarrollada en 2012 y luego adquirida por Semtech, quien fue responsable de mejorarla e impulsarla. Usa un tipo de modulación de amplio espectro, ideal para tolerar el ruido y para que una señal realice caminos múltiples. Presenta una tasa de datos reducida, pero adaptado a las necesidades de los dispositivos que ayudan a conectar. Esta tecnología se destaca por su capacidad para transmitir datos a largas distancias utilizando muy poca energía, lo que la hace ideal para conectar dispositivos que operan en lugares remotos o donde la eficiencia energética es una prioridad. [11]

LoRa funciona en bandas de frecuencia no licenciadas, como las bandas ISM (Industrial, Scientific, and Medical), que están disponibles globalmente sin necesidad de pagar por licencias específicas. Esto contribuye a reducir los costos de implementación y operación, lo que es una de las razones por las cuales LoRa es tan popular en proyectos IoT de gran escala.

Las frecuencias que son utilizadas varían según la regiones donde se utilizan, por ejemplo 868 MHz en Europa, 915 MHz en América del Norte y 433 MHz en Asia y otras regiones. Aparte se puede utilizar 2400MHz pero esta frecuencia es menos utilizada en el mundo para este tipo de aplicaciones, si bien pertenece a las bandas ISM, es comúnmente utilizada por WiFi (Wireless Fidelity) y Bluetooth, entonces en LoRa es menos frecuente para evitar interferencias y por tener menor alcance respecto a las bandas más bajas. [12]

Cada una de estas bandas de frecuencia tiene sus propias ventajas y limitaciones. Las bandas más bajas, como 433MHz, ofrecen una mejor penetración y mayor alcance, lo que es ideal para aplicaciones en entornos con muchos obstáculos. Sin embargo, el menor ancho de banda limita la velocidad de transmisión de datos. Las bandas más altas, como 868 MHz y 915 MHz, permiten un equilibrio entre alcance y capacidad de transmisión, siendo ampliamente utilizadas en aplicaciones IoT que requieren una cobertura confiable en áreas mixtas, tanto urbanas como rurales.

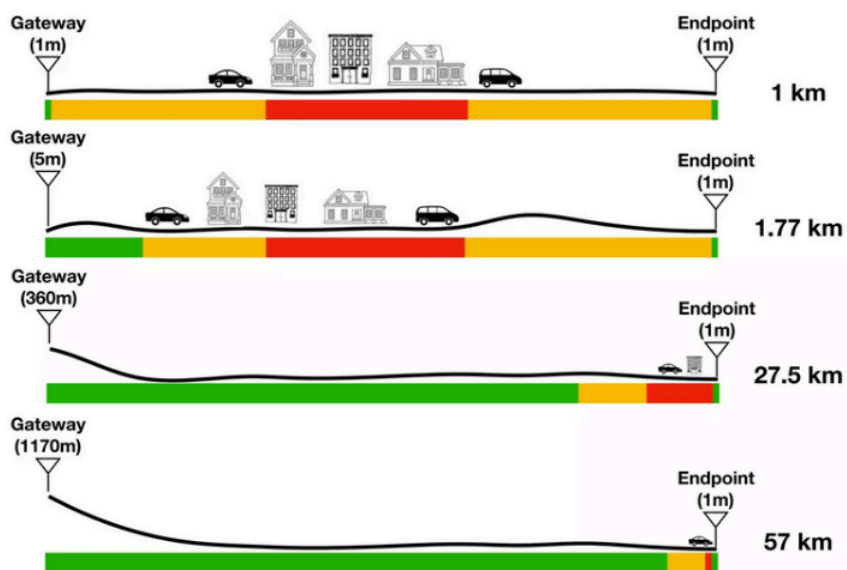


Figura 3.6: Ejemplos de Alcances [13]

3.1.5. ESP-NOW

ESP-NOW es un protocolo de comunicación inalámbrica desarrollado por Espressif. Este protocolo permite la comunicación directa entre dispositivos ESP sin la necesidad de un punto de acceso Wi-Fi, lo que lo hace ideal para aplicaciones que requieren una comunicación rápida y de bajo consumo de energía.

A diferencia de las conexiones Wi-Fi tradicionales, donde los dispositivos deben conectarse a un router o punto de acceso, ESP-NOW permite que los dispositivos ESP se comuniquen entre sí directamente en una configuración peer-to-peer. Esto se logra mediante la utilización de la capa física del estándar Wi-Fi (IEEE 802.11), pero sin establecer una conexión completa a la red. Como resultado, los dispositivos pueden enviar y recibir datos con muy baja latencia, lo que es crucial en aplicaciones donde la velocidad de respuesta es importante.

Una de las características más destacadas de ESP-NOW es su eficiencia energética. Los microcontroladores ESP pueden permanecer en modo de bajo consumo (deep sleep) y despertarse solo para enviar o recibir mensajes a través de ESP-NOW. Esto es particularmente útil en aplicaciones IoT que requieren dispositivos con una vida útil de batería prolongada, como redes de sensores distribuidos. El consumo de energía reducido se debe a que los dispositivos no tienen que mantener una conexión Wi-Fi activa en todo momento, lo que es un proceso energéticamente costoso.

En términos de capacidad, ESP-NOW permite que un dispositivo ESP se comunique con hasta 20 otros dispositivos simultáneamente. Este protocolo utiliza direcciones MAC para identificar y dirigir los mensajes a los dispositivos correspondientes, lo que facilita la gestión de múltiples conexiones. Sin embargo, la capacidad de transmisión de datos es limitada, ya que ESP-NOW está diseñado para enviar pequeños paquetes de datos, típicamente de hasta 250 bytes. Esta limitación lo hace más adecuado para aplicaciones que requieren la transmisión de información breve, como actualizaciones de estado o comandos de control, en lugar de grandes volúmenes de datos.

ESP-NOW también incorpora opciones de seguridad, incluyendo la encriptación de los datos transmitidos. Esto asegura que la comunicación entre los dispositivos esté protegida contra accesos no autorizados. Para establecer una comunicación segura, los dispositivos deben intercambiar claves de encriptación antes de la transmisión, lo que garantiza que sólo los dispositivos autorizados puedan participar en la comunicación.

En cuanto al alcance, ESP-NOW ofrece un rango de comunicación similar al de las conexiones Wi-Fi tradicionales. En espacios abiertos, el alcance puede llegar a varios cientos de metros, aunque en entornos interiores o en áreas con muchas obstrucciones, el alcance

efectivo puede verse reducido. Este rango es adecuado para muchas aplicaciones IoT, aunque no es tan extenso como el que ofrecen otras tecnologías como LoRa.

Sin embargo, a pesar de sus ventajas, ESP-NOW tiene algunas limitaciones. No es adecuado para aplicaciones que requieren la transmisión de grandes volúmenes de datos o comunicaciones de largo alcance. Además, está restringido a la familia de microcontroladores ESP, lo que significa que no puede ser utilizado directamente con otros tipos de microcontroladores sin adaptaciones adicionales. [14] [15]



Figura 3.7: Tipos de Comunicación soportados [16]



Figura 3.8: Tipos de Comunicación soportados [16]



Figura 3.9: Tipos de Comunicación soportados [16]



Figura 3.10: Tipos de Comunicación soportados [16]

3.1.6. RFID

RFID (Identificación por Radiofrecuencia) es una tecnología que permite la comunicación inalámbrica para la identificación y seguimiento de objetos, personas o animales a través de ondas de radio. Este sistema consta principalmente de etiquetas RFID, lectores RFID y antenas. Las etiquetas, o tags, contienen un microchip y una antena que reciben y envían señales de radio. Pueden ser pasivas, activas o semipasivas.

Las etiquetas pasivas son las más comunes y no tienen una fuente de energía propia; dependen del campo electromagnético generado por el lector para transmitir la información almacenada en su chip. Estas etiquetas son más económicas y duraderas, pero su alcance es limitado, típicamente de unos pocos centímetros a varios metros dependiendo de la frecuencia utilizada. Por otro lado, las etiquetas activas contienen una batería interna, lo que les permite transmitir señales a distancias mayores, que pueden alcanzar hasta varios cientos de metros. Sin embargo, estas son más costosas y requieren mantenimiento para reemplazar la batería.

En cuanto a las frecuencias, RFID opera en varias bandas que determinan el alcance y la aplicación de la tecnología:

- Baja Frecuencia (LF) - 125-134 kHz: Las etiquetas que operan en esta banda tienen un alcance de aproximadamente 10 cm a 30 cm. Son menos susceptibles a interferencias causadas por sólidos y líquidos, lo que las hace adecuadas para

aplicaciones de corto alcance, como la identificación de animales y sistemas de control de acceso.

- Alta Frecuencia (HF) - 13.56 MHz: Con un alcance de entre 10 cm y 1 metro, las etiquetas HF son ampliamente utilizadas en aplicaciones como tarjetas de transporte, pagos sin contacto e identificación de productos. Esta banda es conocida por su buena penetración en materiales no metálicos, lo que facilita su uso en diversas aplicaciones comerciales y de seguridad.
- Ultra Alta Frecuencia (UHF) - 860-960 MHz: Estas etiquetas tienen un alcance de 1 metro a 12 metros, siendo ideales para la gestión de inventarios, el seguimiento de activos y sistemas de peaje automáticos. Aunque ofrecen un mayor rango de lectura, son más susceptibles a interferencias por metales y líquidos.
- Frecuencias de Microondas - 2.45 GHz: Utilizadas principalmente en etiquetas activas, estas frecuencias permiten un alcance de 30 metros. Se emplean en aplicaciones que requieren un rango de lectura muy largo y altas tasas de transferencia de datos, como el rastreo de vehículos y el monitoreo de contenedores en grandes almacenes.

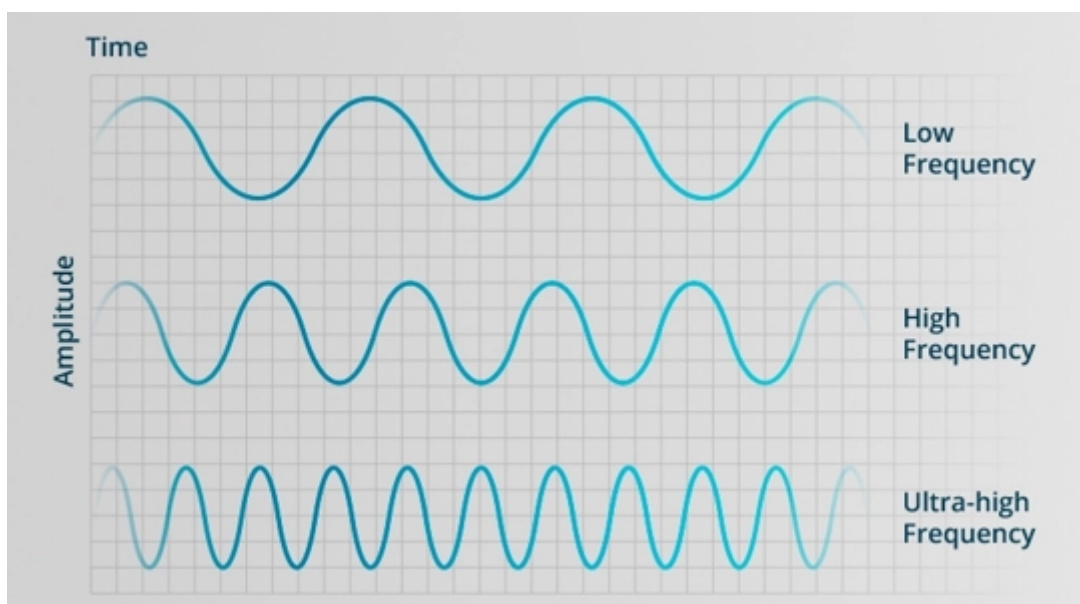


Figura 3.11: Frecuencia y Amplitudes [17]

La distancia de lectura varía según el tipo de etiqueta y la frecuencia en la que operan. Las etiquetas pasivas, al depender de la energía del lector, tienen un alcance más limitado, mientras que las etiquetas activas, con su fuente de energía propia, pueden cubrir distancias

significativamente mayores. En aplicaciones donde se requiere un seguimiento a larga distancia, como la logística o el rastreo de vehículos, las etiquetas activas son preferidas. En cambio, para aplicaciones de corto alcance, como el control de acceso o la identificación de productos en tiendas, las etiquetas pasivas son más comunes debido a su menor costo y mayor durabilidad. [18]

3.1.7. NFC

NFC (*Near Field Communication*) es una tecnología inalámbrica que permite la comunicación entre dispositivos cuando están a una distancia muy corta, generalmente menos de 10 centímetros. Basada en la tecnología RFID (Identificación por Radiofrecuencia), NFC opera en la banda de 13.56 MHz y se utiliza en una variedad de aplicaciones modernas, como pagos móviles, acceso sin contacto, y transferencia de datos.

NFC funciona mediante la inducción electromagnética entre dos dispositivos. Cuando un dispositivo NFC activo, como un smartphone, se acerca a otro dispositivo NFC (ya sea otro dispositivo activo o una etiqueta pasiva), se genera un campo electromagnético. Este campo induce una corriente en la antena del dispositivo receptor, lo que permite la transferencia de datos entre los dos dispositivos.

Una de las características más importantes de NFC es su capacidad para establecer la comunicación de manera extremadamente rápida y sencilla. A diferencia de otros métodos de conexión, como el emparejamiento de Bluetooth, NFC no requiere configuraciones complicadas. Los dispositivos simplemente necesitan estar lo suficientemente cerca para iniciar la comunicación, lo que la hace ideal para aplicaciones que requieren rapidez y simplicidad, como los pagos móviles.

NFC opera principalmente en dos modos diferentes, que permiten diferentes tipos de interacción entre los dispositivos:

- **Modo Pasivo o Solo Lectura:** En este modo, el dispositivo activo genera un campo electromagnético que energiza a un dispositivo pasivo, como una etiqueta NFC. El dispositivo pasivo no necesita una fuente de alimentación propia, ya que utiliza la energía generada por el dispositivo activo para funcionar. Este modo es común en aplicaciones donde el dispositivo pasivo contiene información almacenada que el

dispositivo activo puede leer. Un ejemplo típico es el uso de tarjetas de identificación o etiquetas inteligentes que proporcionan información adicional al ser escaneadas.

- **Modo Activo o Lectura/Escritura:** En este modo, el dispositivo NFC puede no solo leer, sino también escribir información en una etiqueta o dispositivo NFC pasivo. Esto es útil en situaciones donde se necesita actualizar o modificar la información almacenada en una etiqueta, como en el caso de inventarios o precios en tiendas minoristas. El dispositivo activo no solo extrae información del dispositivo pasivo, sino que también puede enviar datos para ser almacenados en la etiqueta.

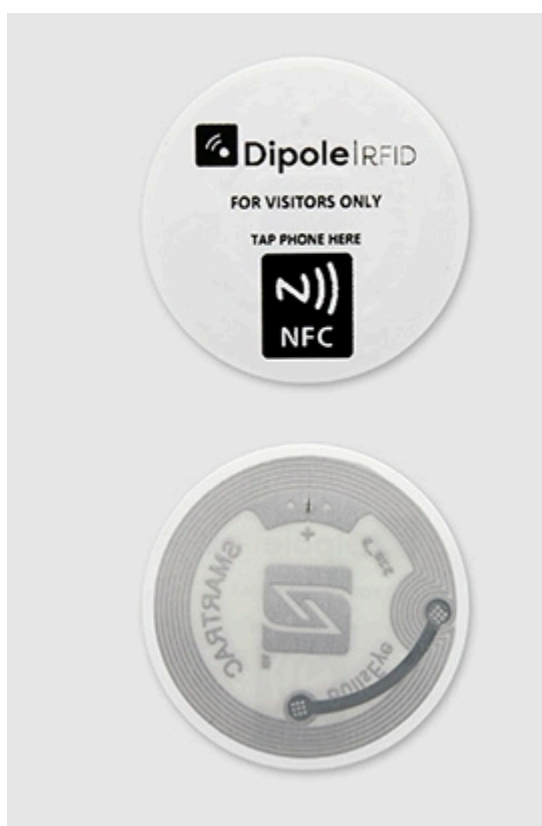


Figura 3.12: NFC Pasivo [19]

NFC opera en la banda de 13.56 MHz, con tasas de transferencia de datos de 106, 212 o 424 kbps. Aunque la velocidad de transferencia es relativamente baja en comparación con otras tecnologías como Wi-Fi o Bluetooth, es más que suficiente para las aplicaciones para las que NFC está diseñado, como la transmisión de pequeñas cantidades de datos en pagos móviles o la autenticación de dispositivos.

La seguridad es una preocupación fundamental en las aplicaciones de NFC, especialmente en los pagos móviles. La proximidad física requerida para que los dispositivos se comuniquen actúa como una barrera natural contra muchos tipos de ataques, ya que los dispositivos deben estar muy cerca entre sí, en otras palabras, el alcance es muy limitado. Además, las transacciones suelen estar encriptadas, lo que añade una capa adicional de seguridad para proteger los datos sensibles.

NFC es una tecnología eficiente en términos de energía. En el modo pasivo, el dispositivo pasivo no requiere una fuente de alimentación propia, ya que obtiene la energía necesaria del campo electromagnético generado por el dispositivo activo. Esto es especialmente útil en aplicaciones donde se utilizan etiquetas o tarjetas que deben ser operativas durante largos periodos sin necesidad de mantenimiento. [20]

3.1.8. GPS

El GPS, o Sistema de Posicionamiento Global, es una tecnología de navegación basada en una red de satélites que orbitan la Tierra. Desarrollado inicialmente por el Departamento de Defensa de los Estados Unidos, el GPS permite a los usuarios determinar su posición geográfica con gran precisión en cualquier lugar del mundo y bajo casi cualquier condición climática.

El sistema GPS funciona recibiendo señales de radio de al menos tres satélites en órbita. Estos satélites transmiten continuamente datos sobre su posición y la hora exacta. Un receptor GPS en la Tierra capta estas señales y calcula su distancia a cada satélite en función del tiempo que tardan en llegar las señales. Utilizando la técnica de trilateración, el receptor determina su ubicación exacta en términos de latitud, longitud y, en algún caso, altitud. Para obtener altitud se necesita de un satélite más, subiendo el mínimo a cuatro.



Figura 3.13: Esquema de Trilateración [21]

3.1.8.1. Componentes del Sistema GPS

- Segmento Espacial: Consiste en al menos 24 satélites que orbitan la Tierra a una altitud de aproximadamente 20,200 kilómetros. Estos satélites están distribuidos de manera que al menos cuatro sean visibles en cualquier lugar del mundo en cualquier momento.
- Segmento de Control: Este segmento incluye estaciones de control ubicadas en la Tierra que monitorizan y gestionan la órbita y la salud de los satélites GPS. Estas estaciones también ajustan los relojes de los satélites para mantener la precisión del sistema.
- Segmento de Usuario: Compuesto por todos los receptores GPS, desde dispositivos en automóviles y smartphones hasta equipos especializados en aviación, marina, y agricultura. Estos receptores utilizan las señales de los satélites para calcular la posición del usuario en la Tierra.

3.1.8.2. Precisión y Factores que Afectan al GPS

La precisión del GPS puede variar dependiendo de varios factores. En condiciones ideales, el GPS puede proporcionar una precisión de hasta unos pocos metros. Sin embargo, esta precisión puede verse afectada por:

- **Interferencias Atmosféricas:** La señal GPS debe atravesar la ionosfera y la troposfera, donde puede ser distorsionada, reduciendo la precisión.
- **Multipath:** Ocurre cuando las señales de GPS rebotan en edificios u otras superficies antes de llegar al receptor, lo que puede causar errores en la ubicación calculada.
- **Visibilidad de Satélites:** Cuantos más satélites sean visibles para el receptor, mayor será la precisión de la ubicación. En áreas con muchas obstrucciones, como en cañones urbanos o bosques densos, la precisión puede disminuir debido a la menor visibilidad de los satélites.

3.1.8.3. Constelaciones de Navegación Global

Además del sistema GPS desarrollado por los Estados Unidos, existen otras constelaciones de satélites de navegación global que funcionan de manera similar y, en conjunto, permiten una cobertura global más robusta y precisa. Estas constelaciones son operadas por diferentes países y regiones, y muchas veces los dispositivos modernos son compatibles con múltiples sistemas de satélites para mejorar la precisión y la confiabilidad del posicionamiento.

- **GPS - Estados Unidos:** El sistema GPS es la constelación original de navegación global, con al menos 24 satélites operativos. Es utilizado en todo el mundo para aplicaciones civiles y militares, proporcionando una precisión de posicionamiento de entre 5 y 10 metros en condiciones normales.
- **GLONASS - Rusia:** GLONASS es el sistema de navegación por satélite operado por Rusia, similar al GPS en su estructura y función. Consta de 24 satélites operativos y ofrece una precisión comparable al GPS, siendo especialmente efectivo en latitudes altas.
- **Galileo - Unión Europea:** Galileo es la constelación de satélites de navegación desarrollada por la Unión Europea. Diseñado para ofrecer una precisión mejorada de hasta 1 metro, es compatible con otros sistemas como GPS y GLONASS, lo que permite una mayor precisión al combinar las señales de múltiples constelaciones.
- **BeiDou - China:** BeiDou es el sistema de navegación por satélite de China, que ha evolucionado desde una cobertura regional a un sistema global completo con más de 30 satélites. Ofrece una precisión comparable a las otras constelaciones globales y es cada vez más utilizado a nivel mundial.

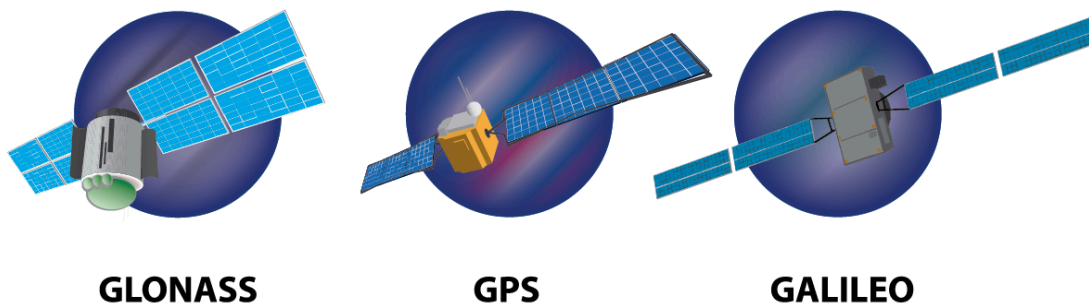


Figura 3.14: Distintas Constelaciones [22]

3.1.8.4. Ventajas de Usar Múltiples Constelaciones

El uso de múltiples constelaciones de satélites en un solo dispositivo, conocido como GNSS (Global Navigation Satellite System), proporciona varias ventajas. La principal es una mejora en la precisión y la confiabilidad del posicionamiento, ya que el dispositivo puede recibir señales de más satélites, lo que ayuda a mitigar los efectos de interferencias o bloqueos de señal en entornos difíciles. Además, el acceso a múltiples constelaciones permite una mayor redundancia, lo que es crucial para aplicaciones críticas en las que se requiere un posicionamiento exacto y constante.

En resumen, el GPS es una tecnología esencial para la determinación precisa de la ubicación en cualquier lugar del mundo. Además del sistema GPS original, otras constelaciones como GLONASS, Galileo, BeiDou y QZSS complementan y mejoran la cobertura y precisión global, proporcionando una herramienta fundamental en la navegación y el posicionamiento global. [22] [23]

3.2. Plataformas IoT

3.2.1. Dashboard

Un dashboard, o tablero de control, es una herramienta visual que presenta información clave de manera gráfica e interactiva, permitiendo a los usuarios monitorear, analizar y tomar decisiones basadas en datos en tiempo real. Los dashboards son ampliamente utilizados en

diversos sectores, desde los negocios hasta la tecnología, incluyendo aplicaciones en proyectos de IoT, análisis financiero, y gestión empresarial.

La principal función de un dashboard es la visualización de datos. En lugar de analizar grandes cantidades de información cruda o trabajar con hojas de cálculo complejas, un dashboard permite a los usuarios ver los datos de forma organizada y comprensible. Esto se logra mediante gráficos, tablas, medidores, mapas y otros tipos de representaciones visuales que facilitan la identificación rápida de patrones, tendencias y posibles problemas.

La interactividad es otra característica fundamental de los dashboards. En muchos casos, los usuarios pueden interactuar con el tablero, filtrando información, profundizando en detalles específicos o ajustando la vista de los datos según sus necesidades. Esto es especialmente útil para la toma de decisiones en tiempo real, ya que permite un análisis más detallado cuando sea necesario.

En entornos que requieren monitorización continua, como en IoT o en la gestión empresarial, los dashboards ofrecen la ventaja de mostrar datos en tiempo real. Esto significa que los usuarios pueden ver el estado actual de sistemas, dispositivos o procesos a medida que ocurren, lo que es crucial para detectar y responder a problemas rápidamente.

Además, los dashboards pueden ser personalizados para mostrar solo la información más relevante para un usuario o contexto específico. Por ejemplo, un gerente puede configurar su dashboard para monitorear KPIs (*Key Performance Indicator*) como por ejemplo ventas o crecimiento de clientes, mientras que un operador de TI puede enfocarse en el monitoreo de la red y los servidores.

En la práctica, los dashboards son utilizados en una amplia gama de aplicaciones. En el ámbito empresarial, permiten a los gerentes seguir el desempeño de sus equipos y ajustar estrategias basadas en datos actuales. En IoT, los dashboards facilitan la visualización en tiempo real de datos provenientes de sensores y dispositivos, como el consumo de energía o el estado de maquinaria. En la ciencia de datos, los dashboards ayudan a los analistas a identificar patrones y anomalías en grandes conjuntos de datos, mientras que en TI, son esenciales para el monitoreo continuo de infraestructuras tecnológicas. [24]

3.2.2. Thingsboard

ThingsBoard es una plataforma de código abierto diseñada para la gestión integral de proyectos de Internet de las Cosas (IoT). Esta plataforma permite la recopilación, procesamiento, visualización y gestión de datos provenientes de dispositivos conectados, lo que la convierte en una herramienta esencial para el desarrollo de soluciones IoT a cualquier escala. Desde pequeños proyectos experimentales hasta implementaciones industriales masivas, ThingsBoard ofrece la flexibilidad y robustez necesarias para manejar la complejidad de los entornos IoT modernos.

Una de las características más destacadas de ThingsBoard es su arquitectura modular y escalable. Esto significa que la plataforma puede crecer con el proyecto, comenzando con unos pocos dispositivos y expandiéndose a millones de ellos a medida que la necesidad aumenta. Además, ThingsBoard puede desplegarse tanto en entornos locales (on-premise) como en la nube, ofreciendo a los usuarios la flexibilidad de elegir la configuración que mejor se adapte a sus necesidades.

La plataforma soporta múltiples protocolos de comunicación, incluyendo MQTT, CoAP y HTTP, lo que facilita la integración de una amplia variedad de dispositivos y sensores, independientemente del fabricante o la tecnología utilizada. Esta capacidad de interoperabilidad es crucial en un ecosistema IoT, donde los dispositivos suelen provenir de diferentes proveedores y tecnologías. Al permitir la conexión fluida de estos dispositivos, ThingsBoard asegura que los datos puedan ser recopilados y procesados de manera eficiente.

ThingsBoard también proporciona herramientas avanzadas para la gestión de dispositivos y telemetría. Los usuarios pueden configurar dispositivos de forma remota, actualizar firmware y visualizar datos en tiempo real a través de paneles personalizables. La telemetría, o la transmisión de datos desde dispositivos a la plataforma, es una función clave que permite a los usuarios monitorizar continuamente el estado y el rendimiento de sus dispositivos, reaccionando rápidamente a cualquier anomalía.

Una característica fundamental de ThingsBoard es su motor de reglas, que permite la automatización de procesos en función de los datos recibidos. Por ejemplo, se pueden configurar reglas para enviar alertas o activar acciones automáticas (como encender o apagar un dispositivo) cuando se cumplen ciertas condiciones. Esta capacidad de automatización es

esencial para optimizar operaciones y reducir la necesidad de intervención manual, especialmente en entornos industriales.

En cuanto a la visualización de datos, ThingsBoard ofrece potentes herramientas que permiten a los usuarios crear paneles interactivos y personalizados. Estos paneles pueden incluir una variedad de widgets, como gráficos, mapas y tablas, que facilitan el análisis de los datos en tiempo real. Esta capacidad de visualización es especialmente útil en aplicaciones donde el monitoreo continuo es crucial para el éxito de la operación.

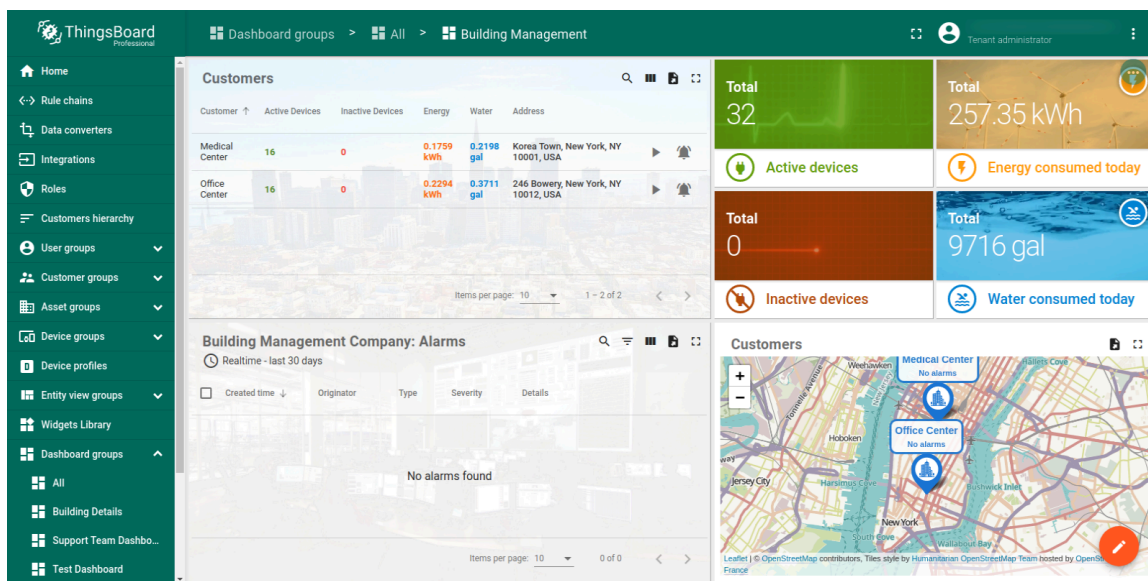


Figura 3.15: Ejemplo de Dashboard en Thingsboard[25]

La plataforma también incluye robustas características de seguridad y gestión de usuarios. Los administradores pueden definir roles y permisos, asegurando que solo los usuarios autorizados tengan acceso a funciones o datos específicos. Esta gestión granular de accesos es vital en aplicaciones empresariales donde la seguridad de los datos es una prioridad.

Además, ThingsBoard se integra fácilmente con otras herramientas y servicios externos mediante APIs, webhooks y conectores. Esto permite ampliar las capacidades de la plataforma y su integración en flujos de trabajo más amplios, haciendo de ThingsBoard una solución versátil y adaptable a diversas necesidades empresariales. [25]

3.2.3. NodeRed

Node-RED es una herramienta de desarrollo basada en flujos diseñada para conectar dispositivos, servicios y APIs de manera sencilla y visual. Desarrollado originalmente por IBM, Node-RED es una plataforma de código abierto que permite a los usuarios crear flujos de trabajo automatizados mediante un editor visual. Estos flujos están compuestos por nodos, que son componentes individuales que realizan tareas específicas, como recibir datos de un sensor, procesarlos, y enviarlos a otro servicio.

Node-RED es especialmente popular en el mundo del IoT debido a su flexibilidad y facilidad de uso. Los usuarios pueden arrastrar y soltar nodos en el editor para construir flujos que recojan datos de dispositivos IoT, los procesen y los envíen a bases de datos, aplicaciones en la nube, o interfaces de usuario como dashboards. Además, Node-RED cuenta con una amplia biblioteca de nodos disponibles que permite la integración con una variedad de servicios y protocolos, incluyendo MQTT, HTTP, WebSocket, y muchos más.

Una característica importante de Node-RED es su dashboard integrado, que permite a los usuarios crear interfaces visuales interactivas para monitorizar en tiempo real los datos recolectados por los dispositivos conectados. Esto lo convierte en una solución ideal para proyectos IoT, donde la capacidad de visualizar y controlar dispositivos es fundamental. [26]
[27]

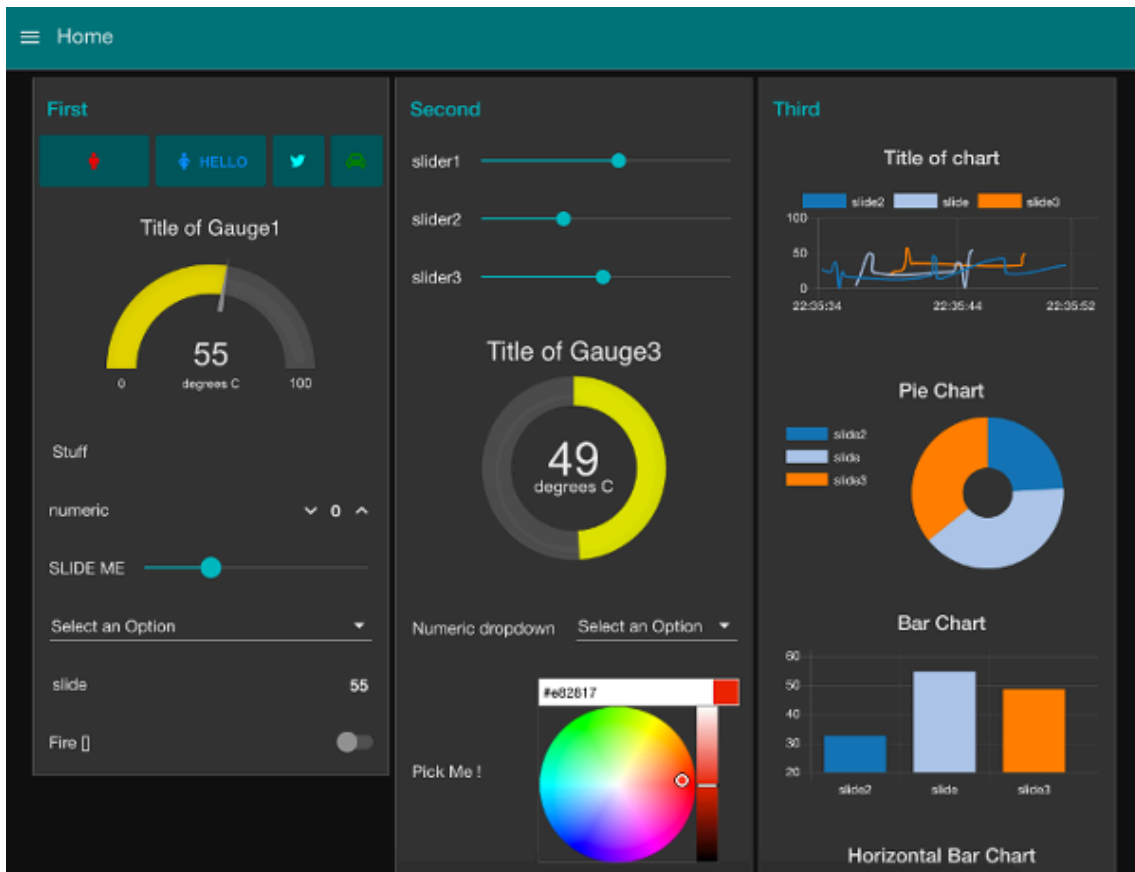


Figura 3.16: Ejemplo de Dashboard en Node Red [28]

3.2.4. Grafana

Grafana es una plataforma de análisis y visualización de datos que permite a los usuarios crear dashboards interactivos y personalizables. Es una herramienta muy popular para la monitorización y análisis de sistemas, servicios y aplicaciones, especialmente en entornos donde se manejan grandes volúmenes de datos en tiempo real.

Grafana se integra con una amplia variedad de fuentes de datos, como bases de datos de series temporales (por ejemplo, InfluxDB, Prometheus), bases de datos SQL (como MySQL, PostgreSQL), y muchas otras. Esta capacidad de integración permite a los usuarios visualizar datos provenientes de múltiples fuentes en un solo dashboard, facilitando el análisis y la correlación de datos.

Una de las principales fortalezas de Grafana es su capacidad para personalizar dashboards según las necesidades del usuario. Los dashboards de Grafana pueden incluir gráficos de líneas, barras, tablas, mapas de calor, diagramas de flujo, entre otros, lo que permite a los

usuarios crear visualizaciones que se adapten específicamente a los datos que están monitoreando.

Grafana es ampliamente utilizado en operaciones de TI, DevOps, y monitoreo de infraestructura, pero también se aplica en otros campos como IoT, finanzas, y análisis de rendimiento de aplicaciones. Su capacidad para alertar a los usuarios cuando se detectan anomalías o cuando se alcanzan ciertos umbrales de datos es particularmente valiosa en entornos donde el monitoreo continuo y la respuesta rápida son críticos. [29] [30]

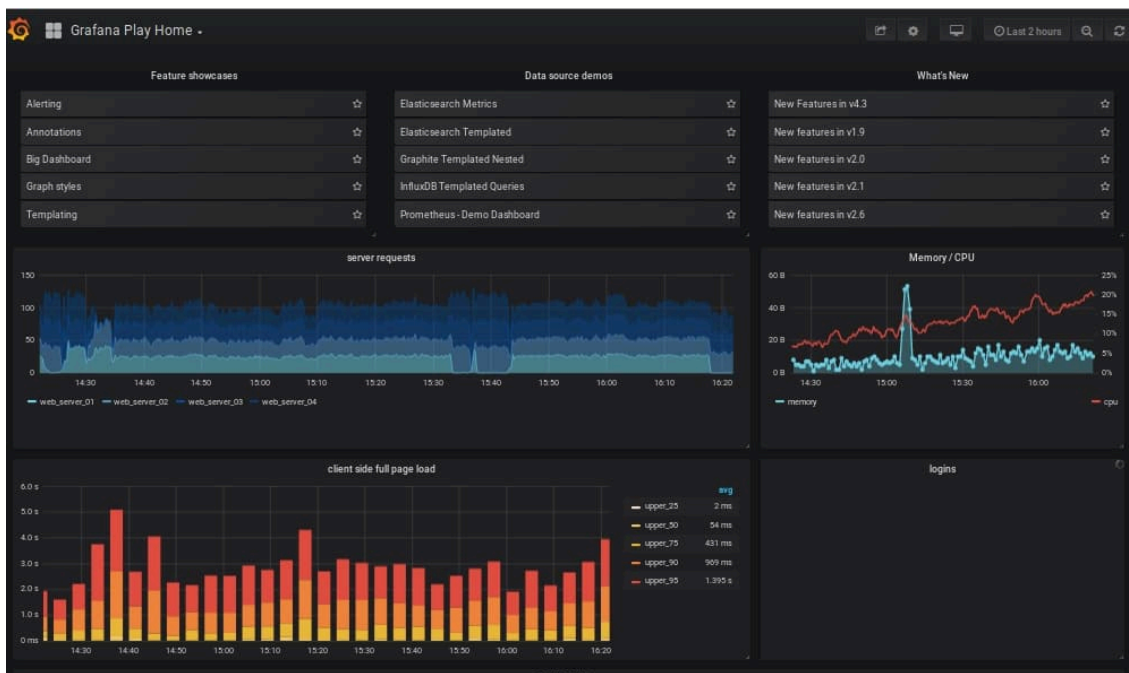


Figura 3.17: Ejemplo de Dashboard en Grafana [29]

3.3. Protocolos

3.3.1. RS232

El "estándar recomendado 232", más comúnmente conocido como RS-232, es un protocolo de comunicación en serie asíncrono establecido en 1960. Originalmente, se usaba comúnmente para conexiones a módems, impresoras, ratones y otros dispositivos periféricos. También es importante tener en cuenta que, a diferencia de los muy similares RS-485 protocolo, RS-232 no es adecuado para acomodar una red de tres o más dispositivos; es únicamente para la comunicación entre dos dispositivos.

Los dispositivos que se comunican a través de RS-232 se comunican a través de dos conjuntos de cables dentro de un cable. Las señales salientes son señales Tx y las señales entrantes son señales Rx. Estas señales de salida suelen estar entre $\pm 5V$ y $\pm 25V$. Para tener en cuenta las caídas de voltaje a lo largo del cable, el dispositivo receptor interpretará cualquier voltaje por encima de +3 V como un 0 y cualquier voltaje por debajo de -3 V como un 1, comunicación binaria simple. [31]

3.3.2. RS485

RS485 es una interfaz estándar de la capa física de comunicación, un método de transmisión de señales, el 1er nivel del modelo OSI. La conexión serie RS485 es realizada utilizando un cable de dos o tres hilos: un hilo de datos, un hilo con datos invertidos y, a menudo, un hilo cero (tierra, 0 V). De este modo, los transmisores y los receptores intercambian los datos a través de un cable de par trenzado de hilos rígidos.

Sus principales características son:

- Intercambio de datos bidireccional a través de un par de hilos trenzados.
- Admite varios transeptores conectados a la misma línea, es decir, permite crear una red.
- Gran longitud de la línea de comunicación.
- Alta velocidad de transmisión.

Por la naturaleza de la interfaz RS-485, los dispositivos RS-485 no pueden transmitir y recibir datos al mismo tiempo, ya que se produce un conflicto de transmisores. Por lo tanto, es necesario adoptar un comportamiento determinista para evitar la colisión de los paquetes de datos. [32]

3.3.3. MQTT

MQTT es un protocolo de mensajería basado en estándares, o un conjunto de reglas, que se utiliza para la comunicación de un equipo a otro. Los sensores inteligentes, los dispositivos portátiles y otros dispositivos de Internet de las Cosas generalmente tienen que transmitir y recibir datos a través de una red con recursos restringidos y un ancho de banda limitado. Estos dispositivos IoT utilizan MQTT para la transmisión de datos, ya que resulta fácil e implementar y puede comunicar datos IoT de manera eficiente. MQTT admite la mensajería entre dispositivos a la nube y la nube al dispositivo.

MQTT implementa el modelo de publicación o suscripción mediante la definición de clientes y agentes, tal y como se muestra a continuación:

- **Cliente MQTT:** Es cualquier dispositivo, desde un servidor hasta un microcontrolador, que ejecuta una biblioteca MQTT. Si el cliente envía mensajes, actúa como editor, y si recibe mensajes, actúa como receptor. Básicamente, cualquier dispositivo que se comunique mediante MQTT a través de una red puede denominarse dispositivo cliente MQTT.
- **Agente MQTT:** Es el sistema de back-end que coordina los mensajes entre los diferentes clientes. Las responsabilidades del agente incluyen recibir y filtrar mensajes, identificar a los clientes suscritos a cada mensaje y enviarles los mensajes. También se encarga de otras tareas como la autorización y autenticación de clientes MQTT, pasar mensajes a otros sistemas para su posterior análisis, el control de mensajes perdidos y sesiones de clientes.

Algunas características principales son:

- **Ligero y eficiente:** la implementación de MQTT en el dispositivo IoT requiere recursos mínimos, por lo que se puede usar incluso en pequeños microcontroladores.

- Escalable: la implementación de MQTT requiere una cantidad mínima de código que consume muy poca energía en las operaciones. El protocolo también tiene funciones integradas para admitir la comunicación con una gran cantidad de dispositivos IoT.
- Fiable: muchos dispositivos IoT se conectan a través de redes celulares poco fiables con bajo ancho de banda y alta latencia. MQTT tiene funciones integradas que reducen el tiempo que tarda el dispositivo IoT en volver a conectarse con la nube.
- Seguro: MQTT facilita a los desarrolladores el cifrado de mensajes y la autenticación de dispositivos y usuarios mediante protocolos de autenticación modernos, como OAuth, TLS1.3, certificados administrados por el cliente, etc.[33]

3.3.4. HTTP

HTTP (*HyperText Transfer Protocol*) es un protocolo de comunicación que permite la transferencia de datos en la WWW (*World Wide Web*) y es esencial para el funcionamiento de internet.

Se basa en un modelo cliente-servidor, donde un cliente (generalmente un navegador web) realiza solicitudes a un servidor web y este último responde proporcionando los recursos solicitados, como páginas web, imágenes, videos o archivos.

Toda esta información se transporta a través de TCP (*Transfer Control Protocol*) que garantiza la fiabilidad de la entrega, y desglosa las grandes solicitudes y respuestas de datos en fragmentos manejables por red. TCP es un protocolo orientado a la conexión, lo que significa que cuando un cliente inicia un diálogo con un servidor, el protocolo TCP abrirá una conexión, sobre la cual los datos HTTP se transferirán de manera confiable, y cuando se complete el diálogo, esa conexión debe cerrarse.

HTTP está diseñado para la comunicación entre 2 sistemas a la vez. Si bien esto funciona bien para solicitar recursos de la web como usuario, no satisface las necesidades de la

configuración de IoT. En la mayoría de las aplicaciones IoT en grandes industrias y unidades de fabricación, hay una gran cantidad de sensores que generan datos al mismo tiempo y también quieren llevar esto al servidor al mismo tiempo. Por lo tanto, HTTP no satisface la necesidad de una a muchas comunicaciones entre los sensores y el servidor.

Además HTTP es unidireccional en el sentido de que solo un cliente o servidor puede enviar un mensaje al otro en cualquier momento. Esto se debe a que se basa en el modelo de solicitud-respuesta donde el cliente tiene que solicitar explícitamente recursos y luego el servidor responde con ellos. Sin embargo, en el caso de las aplicaciones IoT, es posible que necesitemos enviar datos en ambas direcciones simultáneamente.[34]

3.3.5. AMQP

El protocolo AMQP (*Advanced Message Queuing Protocol*) es un protocolo que resuelve varios problemas al mismo tiempo:

- El protocolo (en colaboración con un bróker de mensajería) se encarga de una transmisión sólida de datos.
- AMQP permite almacenar mensajes en una cola. Esto, a su vez, permite una comunicación asíncrona: transmisor y receptor no deben actuar al mismo ritmo. El receptor (consumidor) del mensaje no tiene por qué aceptar, procesar la información directamente y confirmar la recepción al emisor (productor). En su lugar, recuperará el mensaje de la cola cuando tenga capacidad disponible para ello. Esto ofrece al productor, entre otras cosas, la posibilidad de seguir trabajando y se evitan los tiempos de inactividad.
- Está basado en estándares de código abierto como TCP/IP.

Como desventajas se destacan su alto requerimiento de energía, procesamiento y memoria para dispositivos y el largo de los encabezados requeridos.[35]

3.4. Hardware

En esta sección se detallarán las opciones investigadas en el apartado de hardware.

3.4.1. Microcontrolador

Los microcontroladores MCU (MicroController Unit) se utilizan en productos y dispositivos controlados automáticamente, como sistemas de control de motores de automóviles, dispositivos médicos implantables, controles remotos, máquinas de oficina, electrodomésticos, herramientas eléctricas, juguetes y otros sistemas integrados.

Al reducir el tamaño y el costo en comparación con un diseño que utiliza un microprocesador, memoria y dispositivos de entrada / salida separados, los microcontroladores hacen que sea económico controlar digitalmente aún más dispositivos y procesos. [36]

Existen diversas marcas y modelos de estos MCU, en particular presentaremos 2 de ellas:

- Esp32: Para este MCU destacaremos su eficiencia energética, ya que fue diseñado para dispositivos móviles o IoT, proporciona integridad con bluetooth y wifi, además de su barato costo. [37]
- nRF9160: En este caso destacaremos su módem integrado compatible con LTE-M y NB-IoT y puede funcionar globalmente eliminando cualquier necesidad de variantes regionales. [38]

3.4.2. Antena

Según la tecnología aplicada será la antena que se propondrá. Se ampliará en otro capítulo.

3.4.3. Gateway

Un *gateway* IoT es un dispositivo esencial en las arquitecturas de Internet de las Cosas, que actúa como intermediario entre los dispositivos conectados, como sensores y actuadores, y la red central o la nube donde se procesan y almacenan los datos. Su principal función es facilitar la comunicación eficiente y segura entre los dispositivos IoT, que suelen utilizar diferentes protocolos y tecnologías, y los sistemas centrales que manejan la información.

Uno de los roles más importantes de un *gateway* IoT es la traducción de protocolos. Los dispositivos IoT a menudo emplean protocolos específicos como MQTT, CoAP, Zigbee o LoRa, que no son compatibles de forma nativa con las redes IP tradicionales. El *gateway* convierte estos protocolos en formatos que pueden ser entendidos por la infraestructura de red más amplia, permitiendo que los datos fluyan sin problemas desde los dispositivos IoT hasta la nube o el sistema central.

Además de la traducción de protocolos, un *gateway* IoT concentra y procesa los datos recopilados por múltiples dispositivos en un área local. Este procesamiento previo puede incluir el filtrado de información irrelevante y la agregación de datos, lo que no solo optimiza el uso del ancho de banda, sino que también asegura que solo la información más relevante llegue al sistema de análisis central. Este proceso es clave para gestionar de manera eficiente grandes volúmenes de datos generados por dispositivos IoT dispersos.

La seguridad es otro aspecto crucial que maneja un *gateway* IoT. Al actuar como una barrera entre los dispositivos IoT y la red externa, el *gateway* proporciona una capa adicional de protección. Gestiona la autenticación de los dispositivos conectados, asegurando que solo aquellos autorizados puedan comunicarse con la red. También puede cifrar los datos para protegerlos durante su transmisión, lo que es vital en aplicaciones donde la integridad y la confidencialidad de los datos son esenciales.

En cuanto a la gestión de dispositivos, un *gateway* IoT permite controlar y configurar remotamente los dispositivos conectados. Esto incluye la actualización de firmware y la configuración de parámetros operativos, lo que es fundamental en despliegues a gran escala donde los dispositivos están dispersos en diversas ubicaciones. [39]

3.5. Firmware

3.5.1. Framework

Un framework es un marco o esquema de trabajo generalmente utilizado por programadores para realizar el desarrollo de software. Utilizar un framework permite agilizar los procesos de desarrollo ya que evita tener que escribir código de forma repetitiva, asegura unas buenas prácticas y la consistencia del código.

Un framework es por tanto un conjunto de herramientas y módulos que pueden ser reutilizados para varios proyectos. [40]

En particular, ESP Arduino Core es un kit de desarrollo de software (SDK) diseñado para simplificar el proceso de programación de los microcontroladores ESP32 y ESP8266 utilizando el lenguaje y la plataforma de programación Arduino.

La plataforma Arduino se ha vuelto popular por su facilidad de uso y su interfaz amigable para principiantes. Y el ESP Arduino Core aporta estas ventajas a los microcontroladores ESP32 y ESP8266. Permite utilizar el mismo lenguaje de programación Arduino y el entorno de desarrollo, incluido el IDE Arduino, para escribir código para los microcontroladores ESP32 y ESP8266. Esto facilita que los principiantes y los aficionados comiencen con estos microcontroladores, especialmente si utilizaron Arduino antes, sin tener que aprender un nuevo conjunto de herramientas.

Por otro lado tenemos ESP-IDF, o framework de Espressif IoT, es un marco de desarrollo oficial para la serie ESP32 de microcontroladores.

El marco ESP IDF proporciona un conjunto de bibliotecas, herramientas y API para desarrollar aplicaciones basadas en microcontroladores ESP32. Ofrece una amplia gama de características, comúnmente necesarias para las aplicaciones de IoT y está diseñado para proporcionar una plataforma integral y flexible.

Una de las principales ventajas del marco ESP-IDF es que es totalmente compatible con los lenguajes de programación estándar C y estándar C ++, que permite escribir código eficiente y de alto rendimiento. [41]

3.5.2. Real Time Operating System

Un RTOS (*Real-Time Operating System*) es, como su propio nombre indica, un sistema operativo en tiempo real. Es un tipo de sistema operativo diseñado específicamente para aplicaciones en tiempo real, donde se requiere una respuesta rápida y determinista a eventos y tareas críticas. A diferencia de los sistemas operativos convencionales, que pueden priorizar la ejecución de tareas según su programación o prioridad. Por tanto, son más predecibles y

estables para controlar tareas específicas, y los procesos suelen residir de forma permanente en la memoria. [42]

En particular, FreeRTOS es un sistema operativo de código abierto en tiempo real para microcontroladores que facilita la programación, el despliegue, la protección, la conexión y la administración de dispositivos periféricos pequeños y de bajo consumo. FreeRTOS, que se distribuye de forma gratuita con la licencia de código abierto del MIT, incluye un kernel y un conjunto cada vez mayor de bibliotecas de software aptas para su uso en todos los segmentos y aplicaciones del sector. [43]

4. Justificación de Tecnologías Elegidas

En este capítulo se profundizará en las tecnologías seleccionadas para desarrollar la solución, enfocándonos en los aspectos técnicos y en la investigación que respaldó cada una de las decisiones tomadas. Se presentarán las herramientas y protocolos utilizados a lo largo del proyecto, proporcionando una visión general que permita comprender cómo cada tecnología contribuyó a alcanzar los objetivos planteados.

4.1. Requerimientos Técnicos

- Menor consumo posible
- Obtener ubicación lo más precisa posible
- Menor costo posible
- Mayor cobertura de conexión
- Gateway de datos
- Diseño compacto y robusto
- Acceso web y mobile

4.2. LoRa

LoRa se justifica principalmente por su capacidad de ofrecer un largo alcance de transmisión combinado con un consumo de energía extremadamente bajo. LoRa es especialmente adecuado para aplicaciones IoT donde los dispositivos están dispersos en grandes áreas, como la agricultura inteligente o la gestión de recursos en ciudades inteligentes, y donde la duración de la batería es crucial. Además, LoRa opera en bandas de frecuencia no licenciadas, lo que reduce los costos operativos en comparación con tecnologías que requieren el uso de bandas licenciadas. En resumen, es ideal cuando se necesita un equilibrio entre alcance, consumo de energía, y costos operativos, con una capacidad de datos moderada.

4.3. ESP-NOW

Elegir ESP-NOW sobre RS232 para la comunicación entre la placa receptora y el gateway se justifica principalmente por la naturaleza inalámbrica y el bajo consumo de energía de ESP-NOW, lo que facilita la implementación sin la necesidad de cables, permitiendo una

mayor flexibilidad en la ubicación de los dispositivos. ESP-NOW es un protocolo de comunicación de baja latencia y sin emparejamiento, que permite la transmisión rápida y eficiente de datos entre dispositivos ESP32 en un rango moderado, lo cual es ideal en entornos donde la instalación de cables RS232 puede ser costosa, complicada o simplemente inviable.

Por otro lado, RS232 es un protocolo de comunicación por cable, lo que implica limitaciones físicas en términos de alcance y movilidad, y puede resultar en costos adicionales para la instalación y mantenimiento de cables. Además, ESP-NOW permite la comunicación con múltiples dispositivos simultáneamente, mientras que RS232 es típicamente punto a punto, lo que limita su escalabilidad en aplicaciones donde se requiere la interconexión de múltiples nodos con un solo gateway.

4.4. Thingsboard

Se eligió ThingsBoard sobre otras plataformas principalmente por su flexibilidad y capacidad para manejar múltiples protocolos, lo que la convierte en una solución robusta para proyectos IoT de cualquier escala. Además, dada la previa experiencia con ThingsBoard, esto permitirá reducir significativamente el tiempo de desarrollo y centrarse en la implementación del proyecto, aprovechando al máximo sus herramientas de visualización y automatización sin enfrentar una gran curva de aprendizaje.

4.5. Entorno de desarrollo - Arduino IDE

El uso de Arduino IDE se justifica por su simplicidad y accesibilidad, especialmente en proyectos que requieren un desarrollo rápido y eficiente. Arduino IDE es una plataforma de desarrollo integrada que ofrece una interfaz sencilla y fácil de usar, lo que la convierte en una excelente opción para el proyecto. Su diseño intuitivo permite escribir, compilar y cargar código en una amplia gama de microcontroladores de manera rápida y sin complicaciones.

Además, Arduino IDE es compatible con una gran cantidad de bibliotecas y ejemplos de código, lo que facilita la implementación de diversas funciones y acelera el proceso de desarrollo.

4.6. Gateway

Se decidió por usar el RTU de Autolink como gateway, y se justifica por su robustez y confiabilidad en la comunicación de datos en entornos industriales. Además, el RTU de Autolink está diseñado para operar en condiciones adversas, asegurando un rendimiento constante y seguro. Su compatibilidad con múltiples protocolos de comunicación industrial y su capacidad para personalizar funciones específicas lo convierten en una opción ideal para proyectos que requieren un gateway confiable y eficiente.



Figura 4.1: RTU Autolink [44]

5. Desarrollo del Proyecto

En este capítulo, se detalla el proceso de desarrollo del proyecto, abarcando tanto la selección de los componentes de hardware como las soluciones de software implementadas. Inicialmente, se presentarán los dispositivos de hardware seleccionados, explicando las razones técnicas detrás de cada elección. A continuación, se describe el diseño y la integración del software, mostrando cómo se lograron optimizar las funciones del sistema para cumplir con los objetivos del proyecto. Finalmente, se incluirá el diseño de la carcasa, destacando su rol en la protección del equipo y la optimización de su funcionalidad en el entorno de operación.

5.1. Hardware elegido

Se decidió optar por una placa de desarrollo disponible en el mercado que cumpla con las características definidas en las secciones 3.4 y 4.1. Esta decisión se tomó principalmente para minimizar los costos de los dispositivos, ya que comprar los componentes por separado y luego fabricar la placa en China resultaría en un costo elevado. Elegir una solución ya existente nos permite reducir significativamente estos gastos.

A continuación se presenta una pequeña tabla con costos aproximados de los componentes más esenciales y relevantes:

Hardware	Modelo	Precio Aprox (USD)	Referencia
GPS	Neo-M8J	25	[45]
LoRa	SX1262	7.79	[46]
MCU	ESP32-C3-WROOM	2	[47]

El costo de los componentes esenciales asciende a 34.79 USD, sin incluir los componentes adicionales necesarios para el funcionamiento del circuito (como resistencias, bobinas, capacitores, etc.), ni el costo de fabricación del circuito. Por otro lado, se identificó una placa de desarrollo de la marca LilyGo T-beam SoftRF que integra todos los componentes necesarios, con un valor de 40.98 USD para la versión de 433 MHz y de 35.89 USD para la versión con antena IPEX y 915 MHz. Inicialmente, se optó por la versión de 433 MHz para

maximizar el alcance de LoRa, pero tras realizar pruebas exhaustivas, se detectó una falla en el módulo LoRa, lo que llevó a la adquisición de la versión de 868/915 MHz, que se presenta como una opción viable para un MVP. [48]

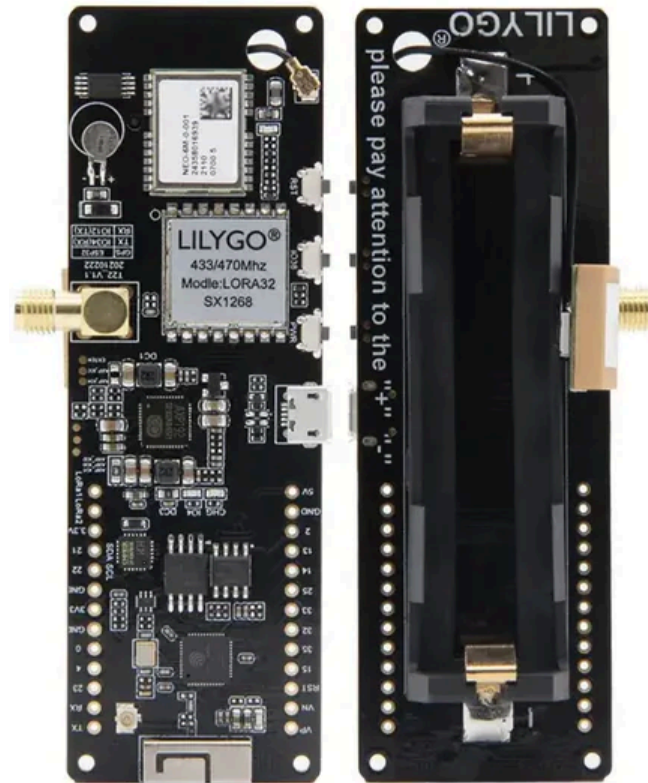


Figura 5.1: LilyGO T-Beam SoftRF [48]

5.1.2. GPS

El módulo NEO-M8 de u-blox es un dispositivo GNSS diseñado para aplicaciones que requieren un alto nivel de precisión en el posicionamiento y confiabilidad en la recepción de señales satelitales. Este módulo soporta la recepción simultánea de señales de múltiples constelaciones GNSS, incluyendo GPS, GLONASS, Galileo y BeiDou, lo que le permite ofrecer un posicionamiento preciso y robusto, incluso en entornos complejos como áreas urbanas densas.

El NEO-M8 se caracteriza por su alta sensibilidad de -167 dBm, lo que asegura un excelente rendimiento en la adquisición y el seguimiento de señales GNSS, crucial para aplicaciones donde la recepción de señales puede estar comprometida. Además, el módulo integra

funciones de mitigación de interferencias y detección de suplantación de señal (spoofing), mejorando la integridad y seguridad de los datos de posicionamiento.

En cuanto a su precisión, el NEO-M8 ofrece una exactitud horizontal de hasta 2.5 metros al utilizar señales GPS y GLONASS, y puede mejorar a 2 metros cuando se utiliza un sistema SBAS (*Satellite Based Augmentation System*). La frecuencia de actualización puede configurarse hasta 10 Hz, lo que permite obtener datos de posicionamiento con alta frecuencia, siendo ideal para aplicaciones en movimiento continuo.

El tiempo necesario para obtener la primera señal (Time-To-First-Fix) también es muy competitivo, con 26 segundos para un inicio en frío (cold start) y solo 1 segundo en situaciones de reinicio en caliente (hot start). Esto lo hace eficiente para aplicaciones que requieren un posicionamiento rápido tras el encendido.

El módulo está diseñado para operar en movimientos de alta velocidad, con un límite de hasta 500 m/s de velocidad y capacidad para manejar aceleraciones dinámicas de hasta 4 g, lo que lo hace adecuado para vehículos de alta velocidad y aplicaciones de seguimiento.

En términos de consumo de energía, el NEO-M8 está optimizado para aplicaciones donde la eficiencia energética es clave. El módulo cuenta con modos de operación que permiten reducir significativamente el consumo de energía, alcanzando niveles tan bajos como 21 mA en su modo de operación continuo, lo que lo hace adecuado para aplicaciones portátiles o alimentadas por batería.

El NEO-M8 es compatible con interfaces estándar como UART, USB, SPI e I2C, lo que facilita su integración en sistemas diversos. También soporta sistemas de aumento satelital (SBAS) como WAAS, EGNOS y MSAS, y ofrece funciones avanzadas como geofencing y detección de spoofing. Estas características, junto con la posibilidad de actualizaciones de firmware, lo convierten en una opción robusta y flexible para aplicaciones industriales y automotrices donde se requiere una alta precisión en el posicionamiento GNSS y un bajo consumo de energía. [49]



Figura 5.2: GNSS Neo -M8 [49]

5.1.3. Semtech SX1276

El SX1276 de Semtech es un transceptor de radiofrecuencia diseñado para operar en las bandas sub-GHz, específicamente entre 137 MHz y 1020 MHz, con un enfoque particular en las bandas de 868 MHz y 915 MHz. Este transceptor utiliza la tecnología LoRa, que permite la comunicación inalámbrica de largo alcance con un bajo consumo de energía. Gracias a su alta sensibilidad de hasta -148 dBm, el SX1276 es capaz de mantener comunicaciones efectivas a largas distancias, incluso en entornos con altos niveles de interferencia o cuando se requiere penetrar obstáculos como edificios o vegetación densa.

En términos de consumo de energía, el SX1276 está optimizado para aplicaciones donde la duración de la batería es crucial. El chip consume solo 9.9 mA en modo de recepción y 28 mA en modo de transmisión a +20 dBm, lo que lo convierte en una opción ideal para dispositivos alimentados por batería que necesitan operar durante largos periodos sin recarga. Además, incorpora modos de baja energía que pueden reducir aún más el consumo cuando el dispositivo no está en activo.

Tal como se mencionó en el apartado 5.1, se seleccionó este chip con las frecuencias de 868 MHz y 915 MHz debido a problemas técnicos encontrados en la versión anterior de la placa del fabricante. Esta decisión se tomó para asegurar la estabilidad y confiabilidad en las

comunicaciones, garantizando un rendimiento óptimo en las aplicaciones específicas del proyecto.

El SX1276 soporta la modulación LoRa, que ofrece una mayor robustez frente a la interferencia y un rango de comunicación superior en comparación con las tecnologías de modulación tradicionales. Esta característica, junto con funciones avanzadas como la corrección de errores y el manejo dinámico de datos, lo hace adecuado para aplicaciones que requieren alta confiabilidad y eficiencia energética. [50]



Figura 5.3: Módulo LoRa SX1276 [50]

5.1.4. Batería

La batería 18650 es una célula de iones de litio con una capacidad de 3500 mAh, operando a un voltaje nominal de 3.6 V o 3.7 V, con un máximo de 4.2 V en estado de carga completa. Este tipo de batería es comúnmente utilizado en aplicaciones donde se requiere una alta densidad de energía en un formato compacto. Su capacidad de 3500 mAh permite un suministro prolongado de energía, adecuado para dispositivos electrónicos de consumo y herramientas portátiles. La batería 18650 es capaz de manejar varios ciclos de carga, lo que la hace adecuada para aplicaciones que demandan durabilidad.



Figura 5.4: Batería NCR18650 [48]

5.1.5. Alimentación

Luego de medir el consumo promedio de la placa receptora, que se determinó en aproximadamente 148 mA, llegamos a la conclusión del panel solar necesario. Este cálculo se basó en la capacidad de la batería de 3500 mAh. Al calcular la potencia promedio consumida, se obtuvo un valor aproximado de 547.6 mW, considerando que la batería opera a un voltaje nominal de 3.7 V.

Para mantener la operación continua del sistema, el panel solar debe ser capaz de generar esta energía diariamente. Asumiendo unas 5 horas de luz solar directa al día, se determinó que el panel solar debe proporcionar al menos 13.15 Wh al día, lo que se traduce en una potencia mínima de 2.63 W. Sin embargo, para asegurar un margen adecuado que cubra variaciones en la disponibilidad de luz solar y posibles ineficiencias, se concluyó que un panel solar de al menos 3W a 5V o 6V sería la opción más adecuada. Este panel garantizará que la batería se recargue correctamente y mantenga la autonomía necesaria para el funcionamiento ininterrumpido de la placa receptora.

A modo de ejemplo, se seleccionó un panel solar de 6V y 5W, diseñado para aplicaciones que requieren una fuente de energía eficiente y robusta. El panel tiene una capacidad de suministro de corriente de 830 mA, adecuada para la carga de baterías y el suministro de energía a dispositivos de bajo consumo. Su clasificación IP67 asegura resistencia al polvo y al agua, lo que lo hace apto para uso en exteriores. Además, su construcción con materiales duraderos garantiza un rendimiento confiable en condiciones ambientales adversas.



Figura 5.5: Panel Solar Google [51]

5.1.6. PMU AXP2101

El AXP2101 es un PMU (*Power Management Unit*) diseñado para la gestión eficiente de energía en sistemas integrados. Este PMU es capaz de manejar múltiples fuentes de alimentación y distribuir la energía de manera precisa a diferentes subsistemas del dispositivo. Una de sus características principales es la inclusión de varios reguladores de voltaje integrados, que permiten generar tensiones de salida estables a partir de una entrada variable.

En cuanto a la gestión de baterías, el AXP2101 incorpora un cargador de baterías integrado, optimizado para baterías de iones de litio. Este módulo no solo supervisa la carga de la batería, sino que también permite la monitorización en tiempo real de parámetros críticos como el voltaje, la corriente y la temperatura de la batería, asegurando así una carga eficiente y segura.

El AXP2101 es compatible con modos de bajo consumo, lo que permite reducir el consumo energético durante los periodos de inactividad del dispositivo, una característica fundamental

para aplicaciones en las que la autonomía es una prioridad, como en dispositivos portátiles y sistemas IoT. La configuración y el control del PMU se realizan a través de interfaces estándar como I2C, lo que permite un ajuste fino de los parámetros operativos y una integración flexible en el sistema. [52]



Figura 5.6: PMU AXP2101 [52]

5.2. Diseño software

5.2.1. Firmware

En el apartado de firmware, se abordará el desarrollo y utilización del IDE de Arduino así como también las librerías específicas que fueron clave para la implementación del sistema: *LoRa*, *TinyGPS*, *XpowersLib* y *PMU Manager*. Cada una de estas librerías desempeñó un papel crucial en el diseño y funcionamiento del proyecto, y su integración se realizó en etapas para garantizar la correcta operación de todo el sistema.

Inicialmente, se comenzó con el diseño de la etapa de GPS utilizando la librería *TinyGPS*. Esta librería permite decodificar y procesar los datos recibidos del módulo GPS, proporcionando información precisa de la ubicación. Se realizaron pruebas exhaustivas para asegurar que la captura y el procesamiento de los datos de posición fueran confiables y precisos antes de avanzar a la siguiente etapa.

Posteriormente, se desarrolló la etapa de comunicación utilizando la librería *LoRa.h*. Esta fase se centró en integrar los datos obtenidos del GPS y transmitirlos mediante la tecnología

LoRa. Se realizaron ajustes para asegurar que la transmisión de datos fuera estable y eficiente, optimizando los parámetros de comunicación según las necesidades del sistema.

Luego, se abordó la etapa de gestión de energía utilizando la librería XpowersLib, en conjunto con la librería PMU Manager, que fue desarrollada específicamente para integrar y gestionar correctamente XpowersLib, asegurando la inicialización y manejo adecuado de los componentes del sistema. En esta fase, se implementaron estrategias de control de potencia y ahorro de batería, configurando el sistema para que el dispositivo se apague durante una hora después de enviar los datos. Este ciclo fue diseñado para preservar la batería, dado que no se requiere una ubicación constantemente, lo que maximiza la autonomía del sistema.

A continuación, se describe el orden de funcionamiento de los dispositivos emisores, detallando paso a paso el proceso de ejecución que sigue cada dispositivo desde su encendido hasta su regreso al estado de reposo.

Para optimizar el consumo de energía, en la fase inicial de encendido del dispositivo se desactivan los módulos que no serán utilizados y que presentan un consumo energético significativo, como lo son el módulo Bluetooth y WiFi. Posteriormente, se procede a activar el módulo GPS. Una vez encendido, el sistema entra en el bucle principal de ejecución. En esta etapa, se inicia el proceso de lectura de coordenadas válidas a través del sistema GPS, el cual tiene una duración estándar de entre uno y tres segundos.

En algunas situaciones, sucede que el sistema no logra visualizar el número mínimo de satélites, cuando esto sucede el proceso de lectura de coordenadas puede demorarse más de lo habitual. Es por esto que el dispositivo está configurado con distintos mecanismos para pasar al estado de reposo, sobre estos se expandirá más adelante. Una vez obtenidas las coordenadas y el número de satélites válidos, el módulo GPS se apaga para reducir el consumo energético y se procede a la etapa de transmisión. A partir de este punto, es que se activa el módulo LoRa.

Durante este proceso, se realizan hasta tres intentos de envío con un intervalo de 3 segundos entre cada uno.

El ciclo de funcionamiento de los emisores incluye dos condiciones para entrar en modo de reposo (Deep Sleep). La primera condición se cumple después de realizar tres intentos de

envío a través de LoRa. La segunda se activa cuando transcurren 30 segundos desde el encendido del dispositivo. Bajo condiciones normales, el dispositivo permanece activo entre 11 y 14 segundos, siendo la etapa de obtención de coordenadas GPS la que presenta mayor variación en el tiempo. Durante el periodo activo, se envían los tres mensajes con la información de ubicación obtenida antes de regresar al modo de reposo. En el peor de los casos, el dispositivo estará encendido durante 30 segundos antes de volver al estado de reposo.

Este escenario plantea dos posibles resultados antes de que dispositivo haga la transición al modo de reposo:

1. Si se obtiene una señal GPS válida en menos de 30 segundos, el dispositivo realizará al menos un intento de transmisión (payload estándar) con la información de su ubicación al receptor.
2. Si no se logra obtener una señal GPS válida en menos de 30 segundos, el dispositivo realizará un intento de transmisión con un payload sin coordenadas de ubicación, indicando la falta de cobertura GPS y de esa manera señalando que no fue posible obtener coordenadas precisas antes de volver a dormirse.

En la sección 5.2.3 se profundizará en el uso de diferentes formatos de payloads de transmisión.

En conclusión, se logró desarrollar un sistema eficiente que obtiene datos del GPS, los envía a través de LoRa y, posteriormente, entra en un estado de bajo consumo de energía durante treinta minutos para preservar la batería. Este enfoque asegura un equilibrio entre la necesidad de obtener la mayor cantidad de información geográfica posible y la optimización del consumo de energía.

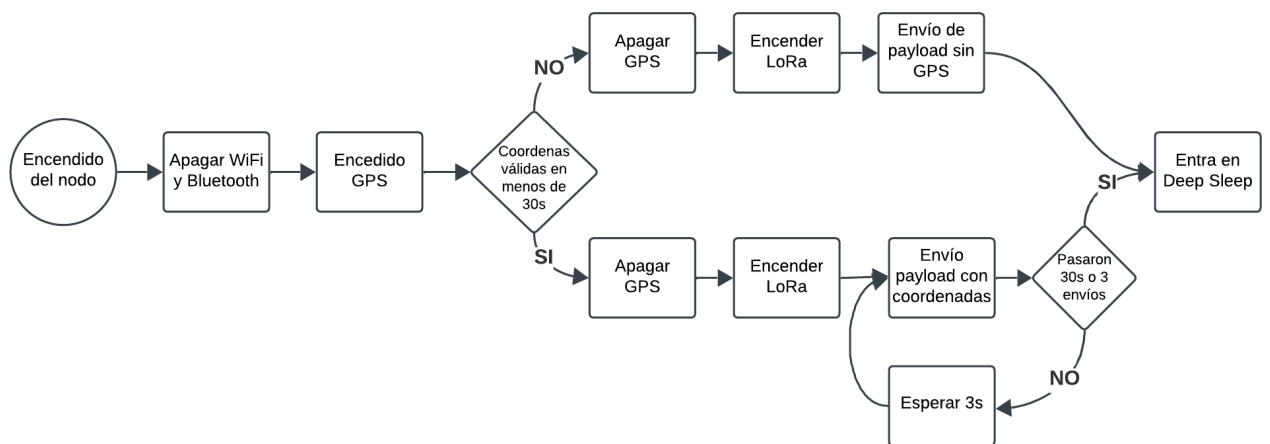


Figura 5.7: Flujo del emisor

5.2.2. Pasamanos MQTT

Inicialmente, es fundamental comprender hacia dónde deben vincularse nuestros mensajes MQTT para poder visualizarlos en la interfaz deseada. En nuestro caso, al trabajar con Thingsboard es necesario conectarse a su servidor de hosting MQTT. Este servidor se caracteriza por ser un sistema más robusto que maneja tópicos predefinidos y claves de acceso distintas para cada dispositivo.

El RTU de Autolink, que funciona como gateway en nuestra solución, ya está configurado para su servidor dedicado de hosting MQTT. Este gateway fue configurado y probado en conexión con un servidor de hosting MQTT particular, garantizando su buen funcionamiento con dicho servidor.

Si se optara por vincular directamente el gateway con el servidor de Thingsboard, sería necesario modificar las claves de acceso en el código del gateway cada vez que se cree un nuevo dispositivo en la plataforma. Además, habría que realizar ajustes en la configuración para vincular correctamente con el servidor de alojamiento de Thingsboard y modificar los tópicos en caso de que se requiriera enviar datos a una sección diferente dentro de la plataforma.

Es a partir de este problema que surge la necesidad de implementar un mecanismo de transición de un hosting a otro. Este pasamanos MQTT, permite migrar de manera controlada hacia un nuevo sistema, manteniendo la estabilidad del servidor actual del gateway.

El concepto de pasamanos implica que, en lugar de cambiar directamente la configuración del gateway, se añade un proceso intermedio que recoge los datos del hosting original y los redirige al nuevo hosting de Thingsboard. Esto permite realizar pruebas, probar distintas configuraciones o tópicos sin la necesidad de abrir el gateway y editar sobre un código que ya fue testado.

Para implementar el pasamanos se optó por desarrollar en Python, dado que es el lenguaje con la mayor cantidad de bibliotecas disponibles, lo que facilita la integración y el manejo de múltiples funcionalidades. En este caso particular, se utilizó la biblioteca paho para gestionar la comunicación MQTT. Paho es una biblioteca de cliente MQTT que permite conectar dispositivos a un broker MQTT, suscribirse a tópicos, publicar mensajes y recibir notificaciones de manera eficiente. Su flexibilidad y amplio soporte la hacen ideal para manejar la transferencia de datos entre múltiples sistemas. [53] [54]

5.2.3. Payload MQTT

Dado que la interpretación de los datos se realizará en la plataforma IoT, es importante identificar los valores de interés para representar la solución en una interfaz gráfica. Se estableció el contenido necesario para incluir en el payload del mensaje MQTT, con el objetivo de mantener el mensaje lo más liviano posible y minimizar el consumo asociado a su envío. El payload tiene un tamaño de 50 bytes y puede contener hasta cinco campos diferentes. Los campos que utiliza son:

- “lat” para la coordenada de latitud
- “lon” para la coordenada de longitud
- “sat” para el número de satélites visibles
- “trk” para el identificador de tracker o emisor
- “bat” para representar el porcentaje de batería disponible

Un ejemplo válido del payload, con el tipo de datos correcto sería: {“lat”:float, ”lon”:float, ”sat”:int, ”trk”:int, ”bat”:int}

En el caso de que el emisor no logre obtener coordenadas GPS válidas dentro del tiempo estipulado, el payload se reduce, y un ejemplo en esta situación sería: {“sat”:0, ”trk”:int, ”bat”:int}

El uso de diferentes formatos de payload permite recopilar una mayor cantidad de información sobre el estado actual de los emisores, lo que incrementa significativamente la capacidad de garantizar el correcto funcionamiento del sistema. Además, facilita un diagnóstico más preciso de posibles fallos, mejorando así la detección y resolución de errores.

5.3. Diseño de Carcasa

5.3.1. Fusion 360

Fusion 360 es una herramienta desarrollada por Autodesk que integra funciones de diseño CAD, CAE y CAM en una única plataforma. Esta solución permite a diseñadores, ingenieros y fabricantes trabajar de manera colaborativa, optimizando tanto la creación como la validación y fabricación de productos.

El software permite modelado 3D paramétrico y libre, lo que es ideal tanto para diseños mecánicos como para productos de consumo. Su capacidad para simular esfuerzos y deformaciones permite validar los diseños en un entorno virtual antes de proceder a la fabricación, reduciendo la necesidad de prototipos físicos. Fusion 360 también ofrece herramientas avanzadas para la fabricación CAM, facilitando la generación de trayectorias de mecanizado CNC para la producción de piezas, además de soporte para impresión 3D.

Una de las ventajas destacadas de Fusion 360 es su base en la nube, lo que permite a los usuarios acceder a sus proyectos desde cualquier ubicación, facilitando la colaboración entre equipos. Además, cuenta con herramientas de renderizado fotorrealista y creación de documentación técnica, lo que lo convierte en una solución completa para todo el ciclo de desarrollo de productos. [55] [56]

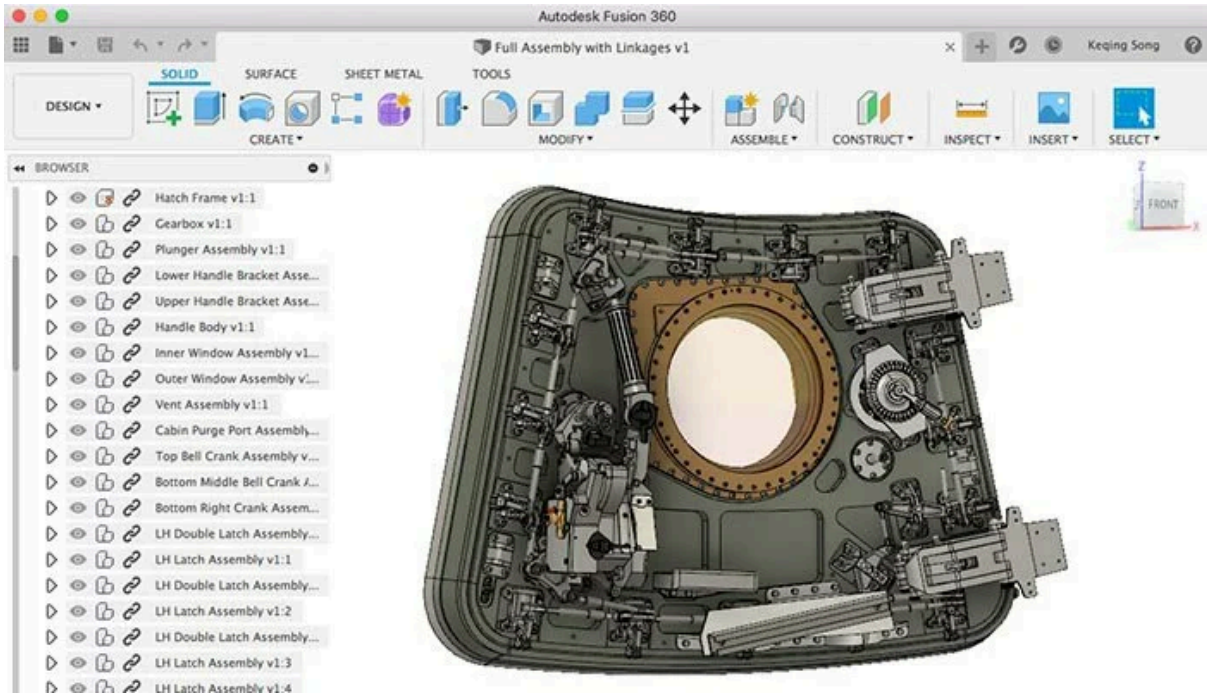


Figura 5.8: Visual de Fusion 360 [55]

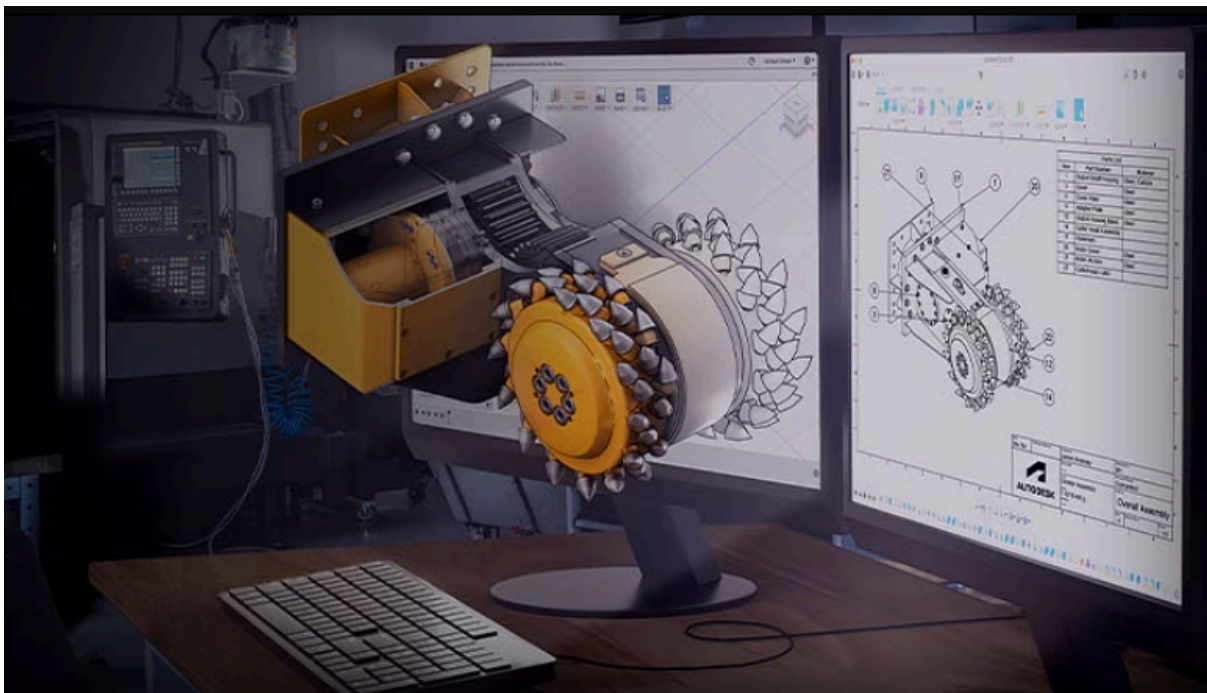


Figura 5.9: Animación de Fusion 360 [56]

5.3.2. Tecnología

5.3.2.1. SLA

La estereolitografía (SLA) es una tecnología de impresión 3D que utiliza un proceso de fotopolimerización para crear objetos sólidos a partir de una resina líquida. En SLA, un láser de luz ultravioleta (UV) se proyecta sobre la superficie de la resina fotosensible, lo que provoca que la resina se solidifique de manera controlada, capa por capa, siguiendo el diseño digital previamente cargado en la impresora.

El proceso se lleva a cabo en una cámara que contiene un tanque de resina líquida y una plataforma de construcción. El láser traza los contornos de cada capa del objeto sobre la resina, y una vez que se ha solidificado una capa, la plataforma desciende una fracción de milímetro para permitir que la siguiente capa sea expuesta al láser. Este proceso se repite hasta que la pieza está completamente formada.

Una de las ventajas técnicas de SLA es la alta resolución que ofrece, permitiendo producir detalles finos con precisión en las dimensiones. Las impresoras SLA pueden trabajar con resoluciones de capa que oscilan entre 25 y 100 micras, lo que permite obtener superficies lisas y detalles complejos en las piezas impresas. Esto hace que la tecnología sea adecuada para aplicaciones que requieren precisión, como en el desarrollo de prototipos o piezas con geometrías complejas.

En términos de materiales, las resinas utilizadas en SLA están formuladas específicamente para endurecerse con luz UV, y existen diferentes tipos de resinas con propiedades mecánicas, ópticas y térmicas ajustadas según la aplicación que se necesite. [57]

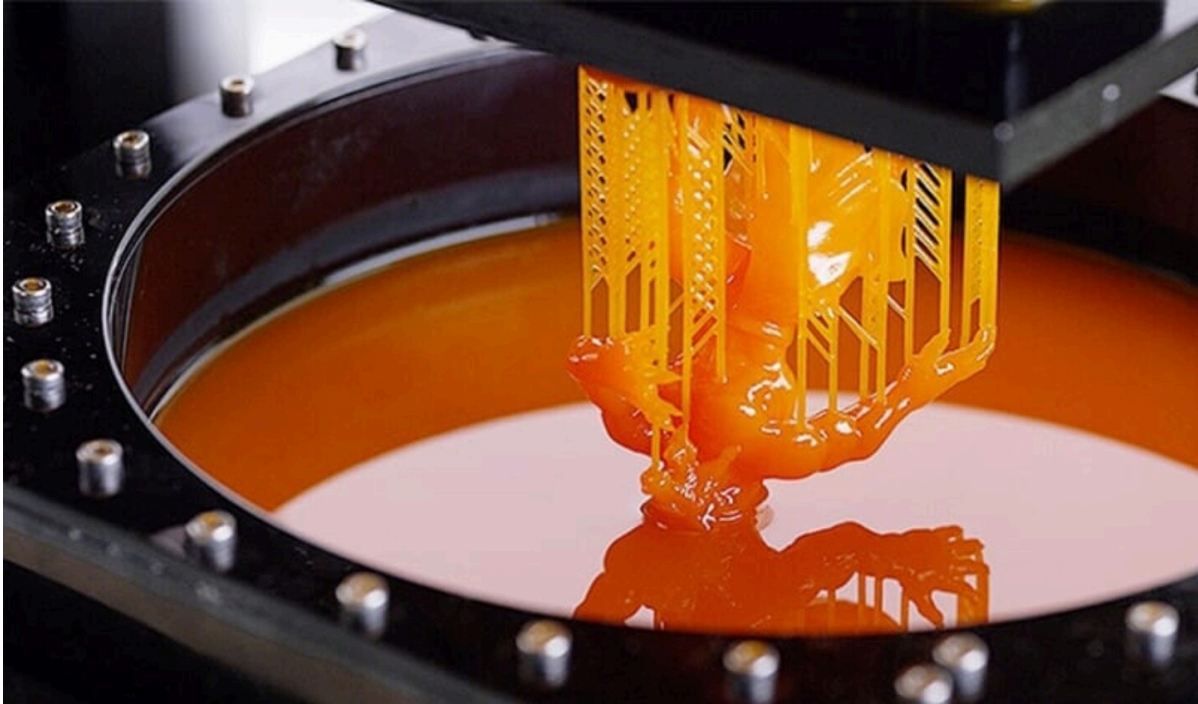


Figura 5.10: Impresión SLA [57]

5.3.2.2. FDM

La fabricación con FDM (*Fused Deposition Modeling*) es una de las tecnologías más comunes en impresión 3D, especialmente en la creación de prototipos y piezas funcionales. En FDM, el material de impresión, generalmente un termoplástico en forma de filamento, es fundido y extruido a través de una boquilla caliente que deposita el material en capas sucesivas para construir el objeto deseado.

El proceso comienza calentando el filamento en la boquilla, que lo funde y lo deposita de acuerdo con el diseño digital 3D. La plataforma de construcción desciende gradualmente conforme el material se deposita capa por capa, permitiendo que el objeto se forme de abajo hacia arriba. Una vez depositada cada capa, el material se solidifica rápidamente al enfriarse, lo que permite la construcción estable del modelo.

Una ventaja técnica de FDM es la variedad de materiales termoplásticos disponibles, como PLA, ABS, PETG y nylon, que ofrecen diferentes propiedades mecánicas y térmicas. Esto permite crear piezas que no solo son estructuralmente estables, sino que también pueden ser utilizadas en aplicaciones funcionales. Además, FDM es ideal para crear piezas con

estructuras internas como rellenos, lo que reduce el uso de material y el tiempo de impresión, sin comprometer la resistencia del objeto.

FDM también destaca por su capacidad de crear objetos de gran tamaño, dependiendo del tamaño de la impresora, y por la facilidad para cambiar y utilizar diferentes materiales y colores de filamento. Sin embargo, la resolución de las piezas en FDM suele estar en el rango de 100 a 300 micras, lo que resulta en un nivel de detalle menor en comparación con otras tecnologías de impresión 3D como SLA. [58]

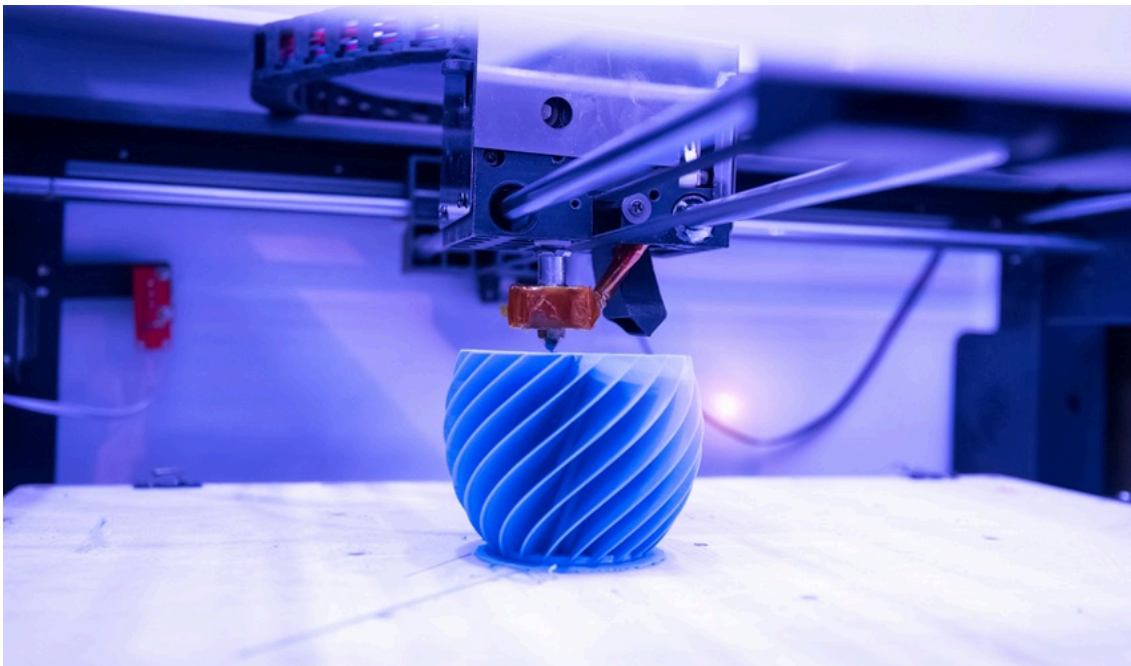


Figura 5.11: Impresión FDM [59]

5.3.2.3. Justificación

Decidimos utilizar la tecnología FDM en el proyecto principalmente debido a su mayor resistencia mecánica en comparación con SLA, lo que es fundamental para las piezas que necesitan soportar esfuerzos o cargas en aplicaciones funcionales. Además, FDM permite el uso de una amplia variedad de materiales termoplásticos como ABS, PLA y PETG, que ofrecen propiedades mecánicas superiores para estructuras robustas, lo que resulta ideal para nuestro caso.

Otra razón importante es el costo. FDM es generalmente una opción más económica en términos de materiales y equipos. Los filamentos termoplásticos utilizados en FDM son

considerablemente más accesibles que las resinas especializadas que requieren SLA. Además, las impresoras FDM suelen ser menos costosas y más accesibles, lo que reduce los costos tanto en la adquisición de la tecnología como en la producción de las piezas.

Finalmente, FDM también nos permitió imprimir piezas de mayor tamaño de manera más eficiente, y su proceso es menos dependiente de post procesamiento en comparación con SLA, que requiere curado adicional bajo luz UV. Esto, junto con la capacidad de producir piezas resistentes a un costo menor, hizo que FDM fuera la opción más adecuada para el proyecto.

En particular, para este proyecto se decidió utilizar la impresora Bambu Lab P1S, que ofrece una excelente combinación de velocidad, precisión y capacidad para trabajar con diversos materiales. La Bambu Lab P1S es una impresora FDM avanzada, diseñada para imprimir con alta calidad y eficiencia. Una de sus ventajas principales es la capacidad para manejar múltiples tipos de filamentos, como PLA, ABS, y PETG, lo que nos permitió seleccionar el material más adecuado para cada pieza en función de las necesidades mecánicas del proyecto.

Además, la Bambu Lab P1S se caracteriza por su rapidez en la impresión, lo que nos permitió reducir significativamente los tiempos de producción sin comprometer la calidad de las piezas. Esta impresora también incluye características avanzadas como la detección de fallos y el control de temperatura preciso, lo que mejora la confiabilidad del proceso de fabricación.[60]



Figura 5.12: Bambu Lab P1S [60]

5.3.2. Concepto

Para el diseño de la carcasa pensamos en algo que sea compacto para no obstaculizar al animal y cerrado para brindar protección al dispositivo contra condiciones ambientales adversas. Además, debe ser robusta para poder resistir posibles impactos, considerando que será usada por animales con comportamientos impredecibles. Por último, es importante que la carcasa no obstaculice el funcionamiento del dispositivo interno, asegurando un rendimiento estable y acorde a lo esperado.

5.3.3. Diseño Exterior

Para el exterior evitamos los vértices puntiagudos ya que podían molestar o lastimar al animal. Los ojales o alas eran necesarias para poder colocar el dispositivo en el animal. En combinación con un elástico o cinta el dispositivo podría colocarse en alguna de las extremidades del mismo. Para elegir el lado sobre el cual colocar las alas optamos por los

lados de mayor longitud de la base rectangular, porque entendemos que tienen menos probabilidad de romperse en comparación con los lados cortos. Para poder cerrar la base y la tapa se colocaron cuatro ranuras para tornillos M2 de 6mm. Esta medida de tornillos es compatible con orificios que ya trae la placa en sus cuatro vértices.

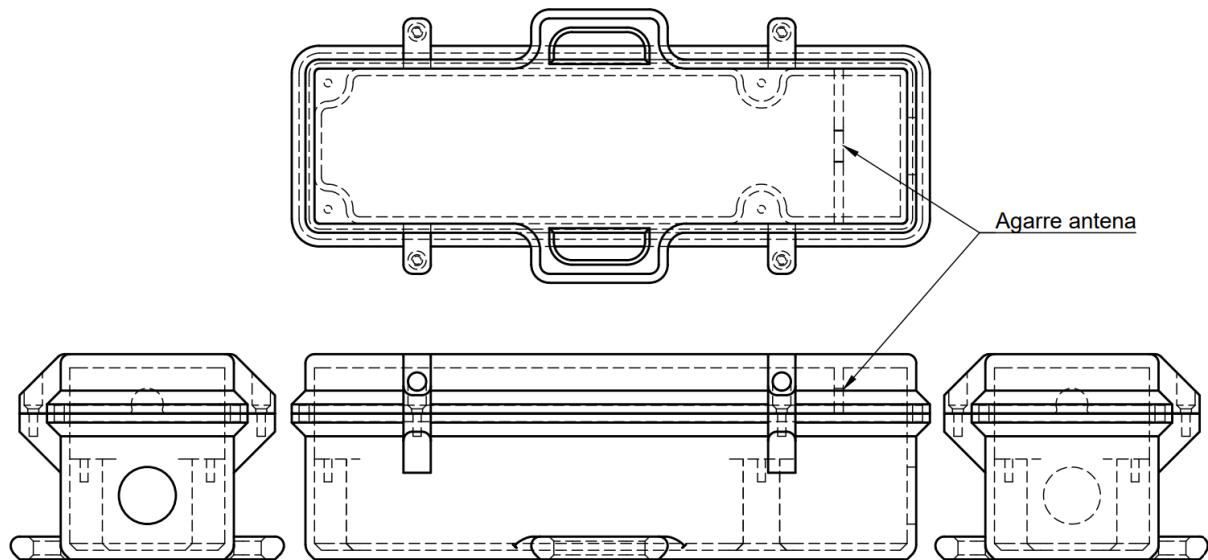


Figura 5.13: Plano del exterior de la carcasa

5.3.4. Protección

En el caso de la carcasa del emisor, priorizamos que sea hermética, por lo que no cuenta con orificios. Sin embargo, para la carcasa de recepción fue necesario agregar un orificio de entrada para la alimentación. Para aumentar su protección, añadimos un bullet hembra y lo recubrimos con silicona alrededor del orificio de entrada.

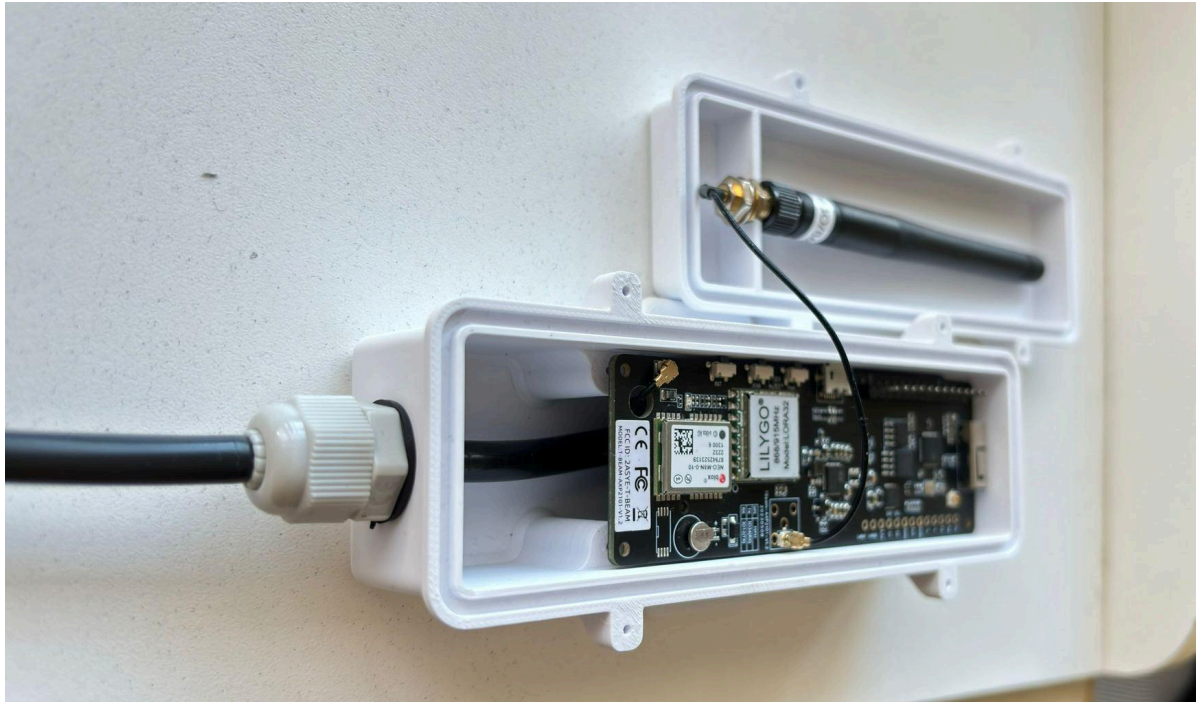


Figura 5.14: Protección para alimentación de carcasa receptora

5.3.5. Interior

Para el interior de la carcasa, se tomaron como referencia inicial las dimensiones de la placa utilizada, que son 32,9 mm x 100,15 mm. Además, se colocaron cuatro soportes para fijar la placa con ranura para tornillos M2, a las dimensiones iniciales se les adicione un 10% de longitud para facilitar la conexión de la alimentación. Finalmente, con el fin de evitar que la antena quedará suelta en el interior, se diseñó un encastre en la tapa para fijarla, evitando así que la antena se mueva o desconecte.

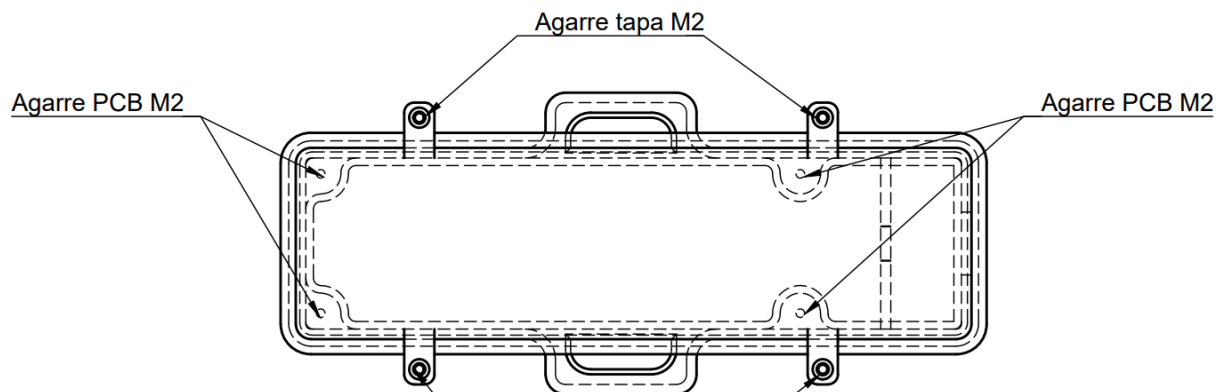


Figura 5.15: Plano del interior de la carcasa

5.4. Lógica para procesamiento de los datos

Todo el procesamiento, análisis y almacenamiento de los datos obtenidos se realizan en la plataforma de Thingsboard. Esto implica que esta etapa del proyecto se desarrollará con las herramientas existentes en la plataforma.

En primer lugar, se identificó que la interacción directa con la plataforma se realiza a través de un único dispositivo, en la solución que se plantea este dispositivo es una RTU. Esta condición ofrece una ventaja en términos de escalabilidad, ya que solo será necesario ajustar las credenciales de conexión MQTT cuando se desee añadir una nueva RTU al servidor de Thingsboard. Para la solución planteada solo sería necesario ajustar las credenciales la primera vez, porque el sistema utiliza una sola RTU.

La clase Dispositivo es una de las clases predeterminadas de la plataforma. Esta clase será utilizada para configurar el dispositivo central de nuestra solución al que llamaremos Gateway dentro de Thingsboard. Para visualizar los distintos emisores o nodos, se emplea la clase Activo. Los activos son útiles para agrupar la información. Y a su vez, facilitan la visualización gráfica de manera segmentada en la interfaz gráfica.

Siguiendo los mecanismos de vinculación de clases de Thingsboard se utilizó el sistema representado en la figura 5.16.

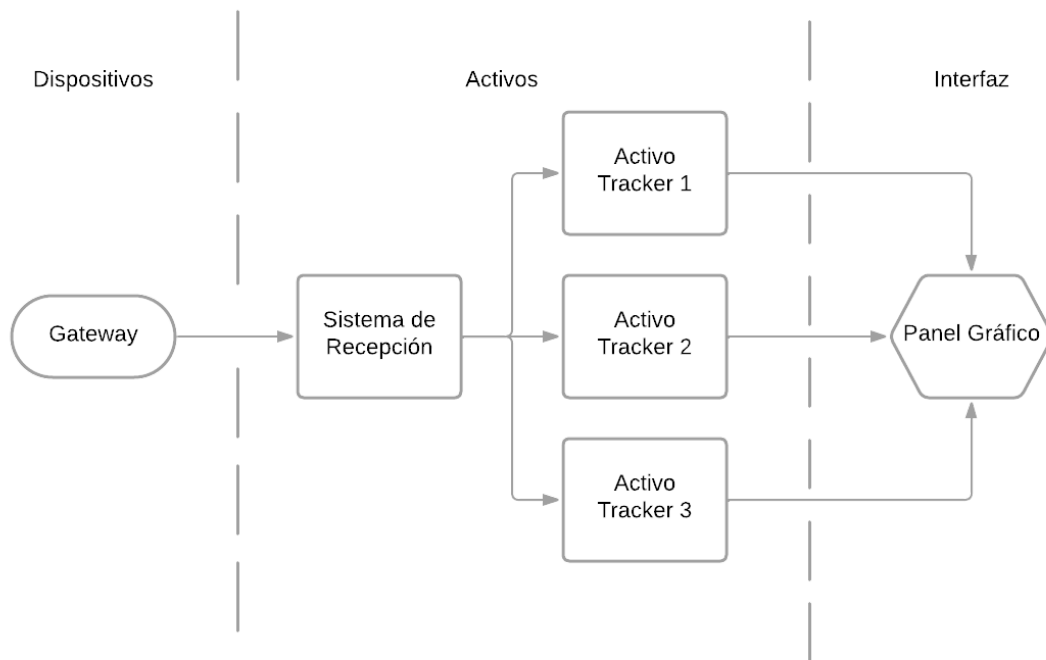


Figura 5.16: Diagrama de flujo para Thingsboard

El dispositivo gateway almacena toda la información recibida por el RTU del receptor, lo cual también aplica al activo Sistema de Recepción. A partir de esta información, se utiliza el campo “tracker id” (trk) para distribuir los datos hacia una entidad específica, asociada con los datos exclusivos de cada nodo o tracker.

Para interpretar los datos de manera lógica, es esencial emplear las cadenas de reglas que proporciona la plataforma. En este proyecto, utilizamos estas reglas para identificar el origen del mensaje, validar las coordenadas, detectar si un emisor carece de señal GPS, verificar si los nodos están dentro del terreno y generar alertas para notificar al usuario.

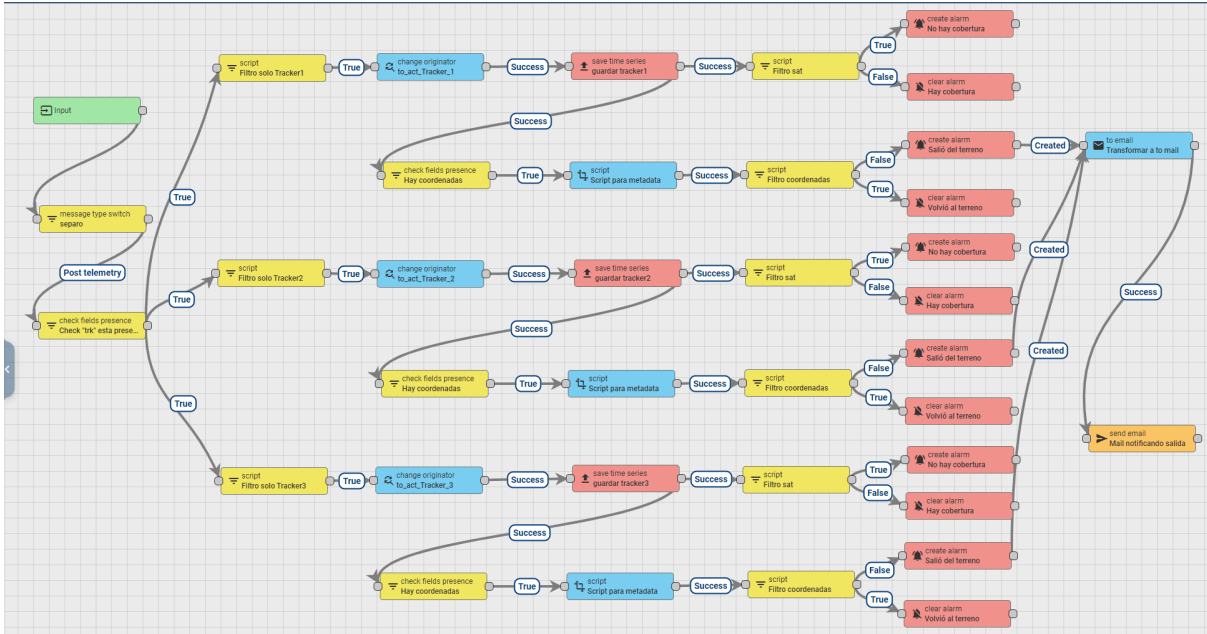


Figura 5.17: Cadena de reglas principal del proyecto

Además, una de las principales ventajas de implementar la lógica en la plataforma es que permite gestionar un número indefinido de activos y dispositivos. Esto facilita que el proyecto crezca en cantidad sin alterar la lógica predefinida ni comprometer la infraestructura, garantizando una expansión escalable sin necesidad de modificaciones en las futuras fases de la solución.

5.5. Esquema de Solución

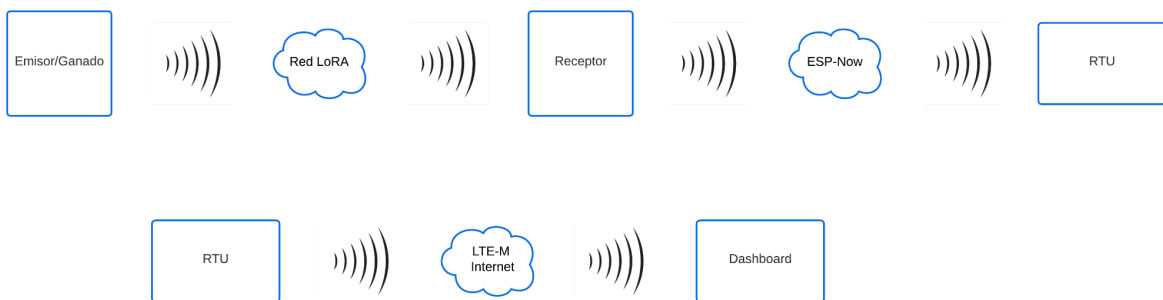


Figura 5.18: Diagrama de solución

En el siguiente diagrama se ilustra cómo las diversas áreas y componentes del productor rural pueden interactuar para implementar la solución planteada en la figura 5.18.

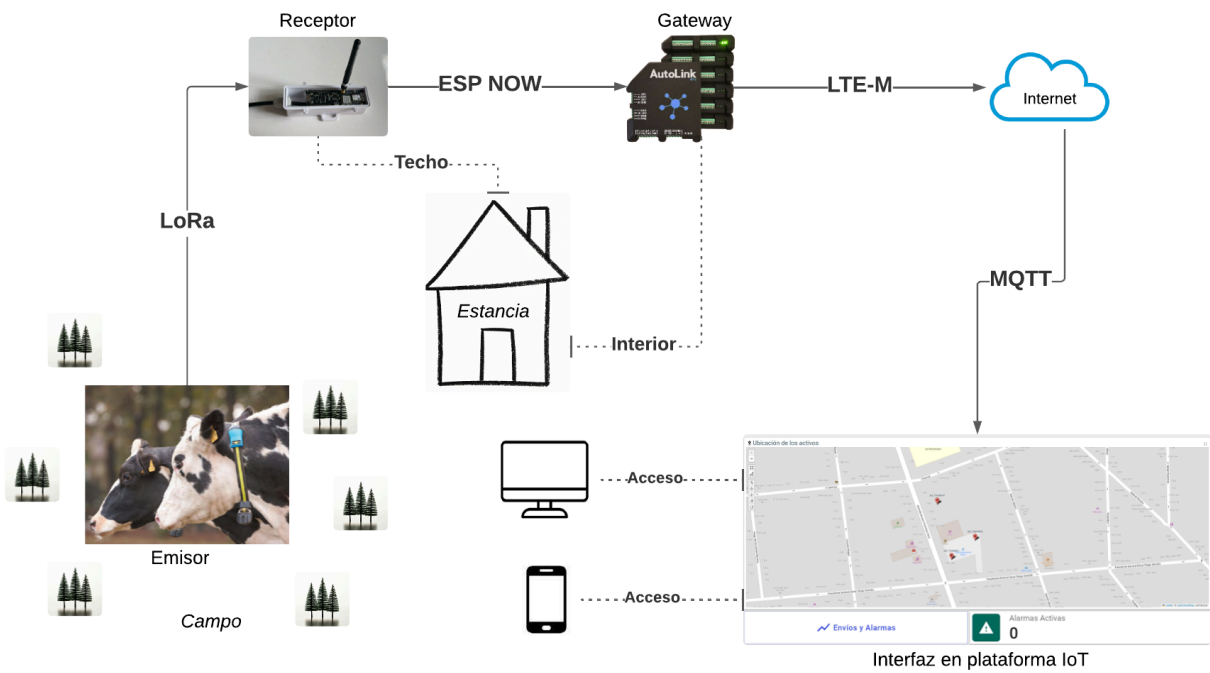


Figura 5.19: Entidades de la solución para entorno rural

Para el dispositivo emisor, decidimos colocarlo en paralelo al eje del cuello del animal, de modo que la carcasa se apoye por completo sobre la superficie del cuello. Esto impide que el dispositivo gire, manteniéndolo siempre en el lomo del animal. En el siguiente diagrama se ilustra la ubicación propuesta para su correcta disposición sobre el animal.

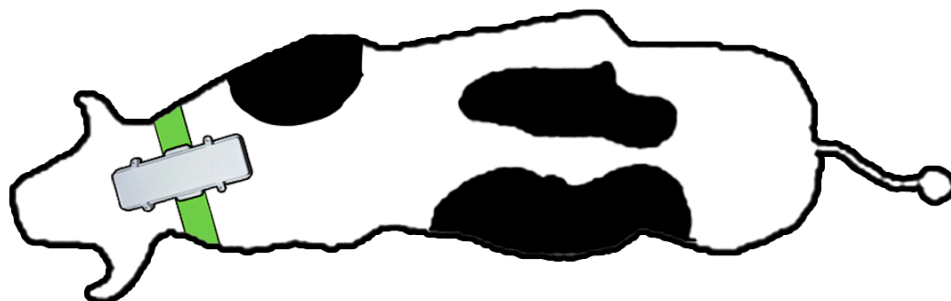


Figura 5.20: Diagrama del dispositivo montado en el ganado

6. Pruebas de Concepto

Las pruebas de concepto son de utilidad porque permiten validar la viabilidad de una idea o solución antes de la implementación completa. Ayudan a identificar posibles problemas técnicos, permiten evaluar la funcionalidad de las distintas partes de manera modular y ajustar el proyecto o solución en base a resultados preliminares. Esto asegura que los recursos se inviertan de manera eficiente y aumenta las posibilidades de las distintas partes del proyecto.

6.1. Prueba de alcance

6.1.1. Objetivo de la Prueba

El objetivo de esta prueba es evaluar la máxima distancia alcanzable de la comunicación emisor-receptor mediante LoRa en los entornos disponibles.

6.1.2. Justificación

Para poder entender los límites de la solución es necesario poner a prueba la comunicación de LoRa en sus distintas configuraciones para poder alcanzar la mayor distancia posible.

6.1.3. Procedimiento

Para estudiar la distancia, colocaremos el receptor en un punto lo suficientemente alto dentro de nuestro alcance. Al mismo tiempo, alejaremos el emisor manteniendo una configuración donde los envíos por LoRa ocurran con una alta frecuencia. Partiendo de una altura conocida para el receptor, podremos determinar la distancia máxima bajo las condiciones urbanas en la que es posible establecer una comunicación estable a través de LoRa.

6.2. Prueba de batería

6.2.1. Objetivo de la Prueba

El objetivo de esta prueba es evaluar la duración de la batería del emisor bajo diferentes condiciones de uso para estimar su autonomía en un entorno real.

6.2.2. Justificación

Es necesario determinar la duración de la batería, ya que el emisor será utilizado en situaciones donde es importante maximizar su tiempo de funcionamiento sin tener que recargar. Además, al ser un dispositivo compuesto por varios módulos y funciones que interactúan entre sí, pasará por diferentes estados de operación. Entender todos los estados posibles y sus respectivos consumos, es clave para obtener una estimación de vida útil lo más precisa posible.

6.2.3. Procedimiento

Para llevar a cabo las pruebas, utilizaremos distintos códigos que permitirán someter al emisor a todos los estados cuyo consumo queremos medir. El consumo será registrado mediante el software nRF Connect, específicamente con su herramienta Power Profiler. Este programa, en conjunto con el dispositivo de hardware Power Profiler Kit II (PPK2), nos permite medir con gran precisión el consumo de energía en dispositivos que utilizan fuentes de bajo voltaje.



Figura 6.1: Power Profiler Kit II [61]

Los estados de mayor interés son de dos tipos: los primeros son aquellos con el potencial de generar un mayor consumo de energía y los segundos corresponden a los estados en los que el sistema pasa la mayor parte del tiempo.

Los estados a estudiar serían:

- Despertar de deep sleep del dispositivo
- Búsqueda de coordenadas de GPS
- Envío por LoRa
- Deep Sleep

Una vez determinados los consumos de corriente para cada estado, sólo resta determinar la distribución de tiempo que el dispositivo pasará en cada estado para poder estimar su autonomía. Al conocer con detalle el consumo de cada estado, nos permite pensar el funcionamiento del sistema de manera de consumir lo menos posible.

6.3. Prueba de GPS

6.3.1. Objetivo de la Prueba

El objetivo de esta prueba es evaluar el rendimiento del módulo GPS en diferentes velocidades de movimiento y en distintos entornos.

6.3.2. Justificación

Es importante probar el módulo GPS en distintas situaciones para asegurar que sea capaz de recibir señales satelitales y que la precisión de los datos geográficos sea confiable en distintos entornos. Estos entornos pueden incluir obstáculos como árboles, llanuras o la presencia de posibles interferencias electromagnéticas

6.3.3. Procedimiento

Se inició configurando el módulo GPS mediante la librería específica, ajustando los parámetros básicos de operación, como la frecuencia de actualización y el formato de salida de los datos (coordenadas de latitud y longitud, entre otros). Una vez configurado, se realizaron pruebas en movimiento. Se montó el dispositivo en un vehículo para simular

desplazamientos en diferentes entornos y a distintas velocidades, con el objetivo de comprobar la precisión de la ubicación y la estabilidad de la señal. Durante estas pruebas, se evaluó si el GPS podría seguir la trayectoria sin pérdida de señal o errores significativos en los datos de ubicación.

6.4. Prueba de comunicación con Gateway

6.4.1. Objetivo de la Prueba

La meta de esta prueba es lograr establecer una comunicación estable y eficiente entre el dispositivo receptor y el gateway de envío de datos.

6.4.2. Justificación

Esta comunicación es clave en la solución propuesta, ya que será utilizada para transmitir todos los mensajes provenientes de los múltiples emisores del proyecto. Si este medio no es confiable o estable, toda la solución se verá afectada, ya que cada mensaje que visualicemos en la interfaz pasará por esta vía.

6.4.3. Procedimiento

Al principio, pensamos transmitir lo recibido en el receptor con TTL y adaptarlo para el gateway. Analizando los componentes, la ESP32 tiene pines UART, lo que nos permitía enviar información en formato TTL. El gateway, por su parte, cuenta con pines Rx y Tx para comunicación RS-232. Es así que resolvimos utilizar un conversor de TTL al protocolo RS-232, con este podríamos enviar la información entre ambos dispositivos.

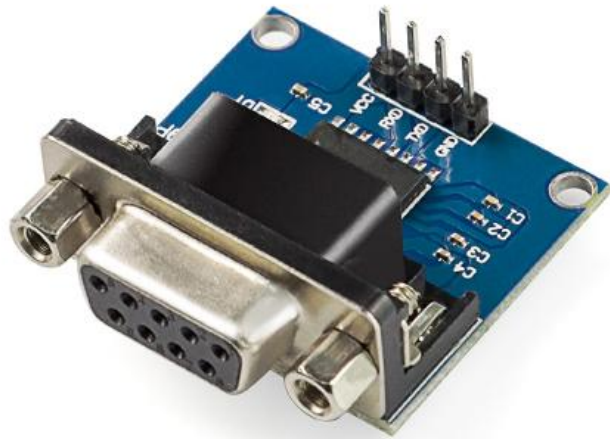


Figura 6.2: Conversor RS-232 a TTL [62]

6.5. Prueba de integración de datos con Thingsboard

6.5.1. Objetivo de la Prueba

Con esta prueba buscamos comprender cómo manejar y almacenar los datos obtenidos en la plataforma IoT. Priorizando que sean precisos, visualizables y que aporten valor al proyecto a partir de la información que está siendo recolectada.

6.5.2. Justificación

Es fundamental comprender las reglas, objetos y clases que utiliza Thingsboard para poder aprovechar al máximo su potencial para el proyecto. Al entender las herramientas que tiene, es posible implementar paneles gráficos y gestionar alarmas. Permitiendo así contar con más información para la toma de decisiones y prevenir problemas antes de que sucedan.

6.5.3. Procedimiento

Experimentar con la plataforma, partiendo desde un solo dispositivo con sus respectivas entidades asociadas, agregando paneles, cadena de reglas y alarmas. Una vez entendidas las interacciones entre las distintas partes, diagramar cómo funcionan todas en conjunto para llevar a cabo la solución.

6.6. Configuración de Alarmas de Salida de Terreno (Geovallado)

6.6.1. Objetivo de la Prueba

El objetivo de esta prueba es configurar y verificar el funcionamiento de las alarmas de geovallado en la plataforma Thingsboard. Se busca asegurar que el sistema pueda identificar cuando un dispositivo o activo sale de un área geográfica predefinida y que envíe una notificación de alerta o tome acciones automatizadas en respuesta.

6.6.2. Justificación

La implementación de geovallado es crucial para mejorar el control y la seguridad de los activos monitoreados. Comprender cómo configurar y personalizar estas alarmas permite optimizar la gestión del proyecto, ya que facilita la detección de movimientos no autorizados o inesperados fuera de las zonas de operación. Esta funcionalidad es esencial para prevenir posibles problemas logísticos y asegurar una rápida toma de decisiones.

6.6.3. Procedimiento

Para realizar esta prueba, primero se configura un área geográfica en la plataforma Thingsboard, definiendo los límites de la zona de operación mediante coordenadas GPS. Luego, se simula el movimiento del dispositivo más allá de los límites configurados para verificar si la alarma se activa correctamente. Tras la simulación, se comprueba que la plataforma envíe la notificación configurada por correo electrónico y que se genere una notificación en el dashboard en el momento de la activación de la alarma. Finalmente, se revisan los registros de eventos y se ajustan los parámetros de las alarmas si es necesario, con el fin de optimizar su funcionamiento.

7. Resultados

7.1. Alcance de distancia

Se seleccionó como punto de instalación del receptor la azotea de un edificio de diez pisos, con una altura estimada de 33 metros desde la vereda. Este punto se utilizó como referencia en el mapa, registrando su ubicación precisa. El sistema fue configurado para transmitir su posición con alta frecuencia mediante una configuración diseñada para la prueba.

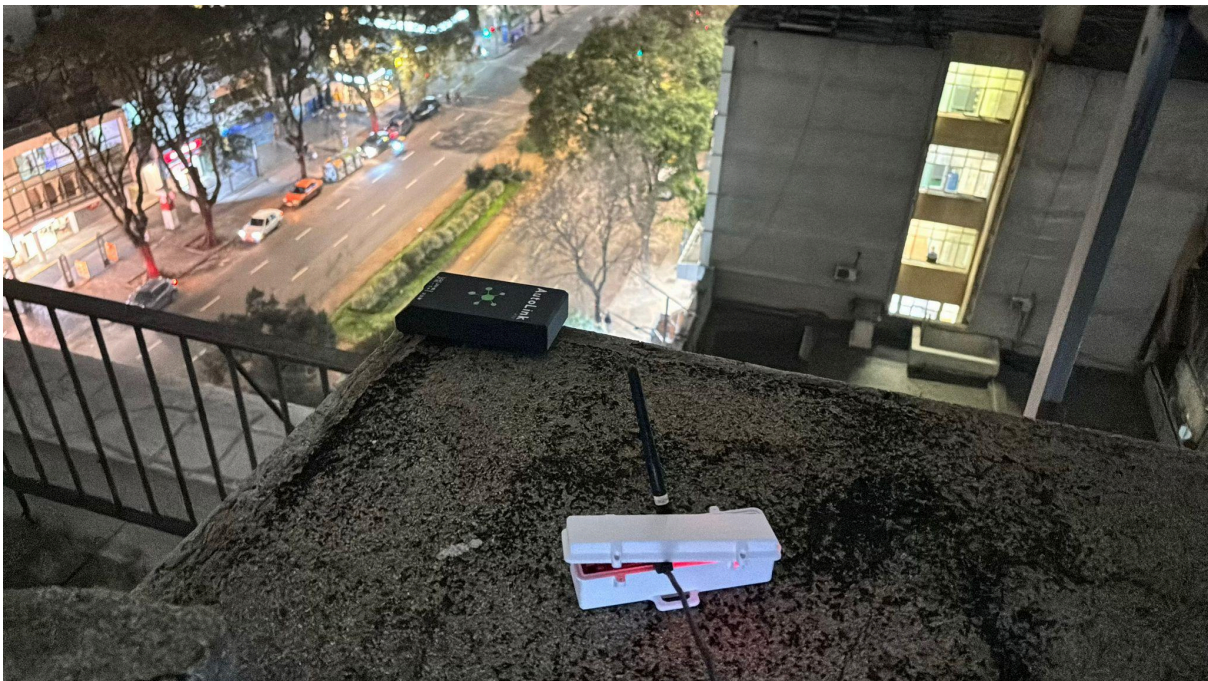


Figura 7.1 : Fotografía del receptor y gateway en la azotea del edificio

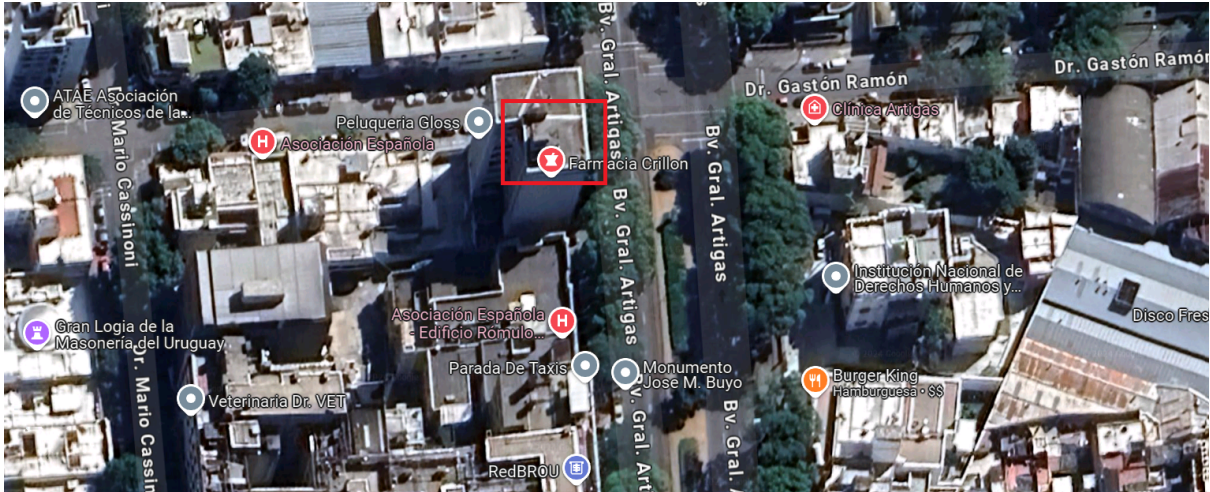


Figura 7.2: Recorte de Google Maps con la ubicación del receptor marcada

A medida que nos desplazamos de manera progresiva, observamos cómo la señal de ubicación se recibe de forma continua en la interfaz de nuestro celular. Durante todo el recorrido, el mensaje se mantuvo estable y la comunicación no presentó interrupciones.

Logramos alcanzar una distancia horizontal de 733,3 metros, tomando en cuenta una componente vertical de 33 metros, lo que nos da un alcance total de 734 metros. Este resultado es alentador, ya que la prueba se realizó en un entorno urbano con numerosos edificios y árboles que podrían haber interferido con la señal.

Dado el rendimiento observado, esperamos obtener un alcance aún mayor en un entorno rural, donde las obstrucciones serán significativamente menores.

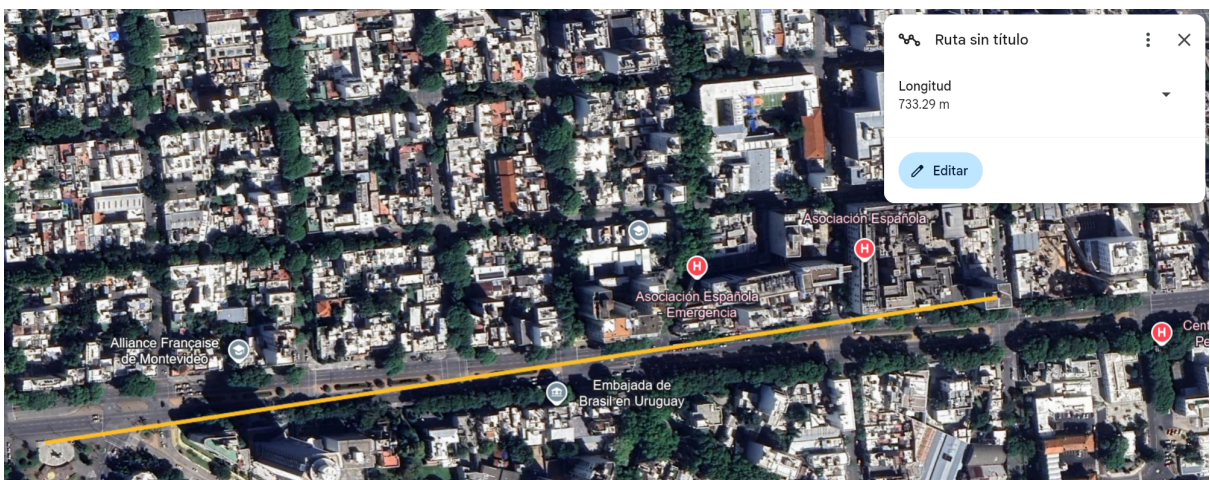


Figura 7.3: Distancia entre receptor y emisor

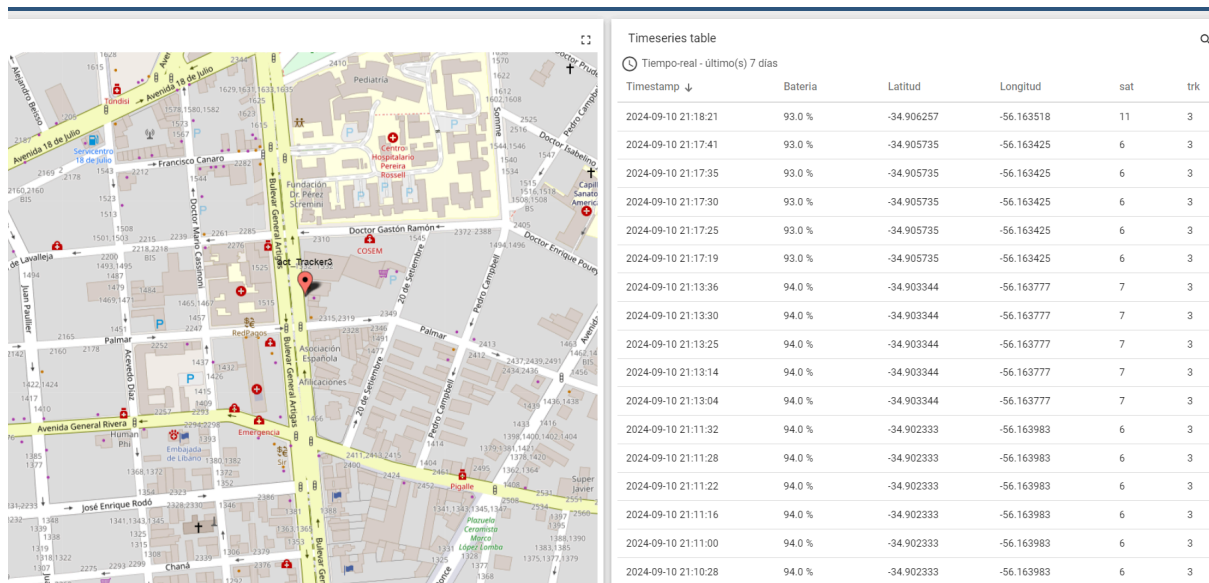


Figura 7.4: Recorte de la pantalla del interfaz en ThingsBoard

7.1.2. Observaciones sobre el Uso de la Carcasa

Aunque las pruebas de alcance de LoRa se realizaron con la carcasa abierta, también se llevaron a cabo pruebas adicionales con la carcasa cerrada para validar el diseño y confirmar que no hubiera diferencias significativas en el rendimiento. Estas pruebas demostraron que la configuración con la antena paralela al piso, según la forma de uso propuesta, no causó ninguna afectación notable en la calidad de la señal ni en la transmisión de datos. Por lo tanto, se puede concluir que el diseño es adecuado y no presenta variaciones de desempeño al estar cerrado.

7.1.3. Observación sobre la Frecuencia de Operación

Inicialmente, la prueba de alcance de LoRa estaba planificada para realizarse en la frecuencia de 433 MHz. Sin embargo, debido a problemas técnicos en la placa, que presentó fallos y dejó de funcionar correctamente, se optó por utilizar otra placa cuya frecuencia de operación era de 915 MHz. Este cambio en la frecuencia y las razones técnicas detrás de la decisión se detallan en el Anexo 2.

7.2. Alcance de batería

Después de varias iteraciones del código, enfocadas en maximizar la autonomía del dispositivo, se llevaron a cabo las pruebas de concepto para cada uno de los estados del sistema. Con base en este enfoque, se procedió a analizar los resultados de consumo obtenidos en los distintos estados del dispositivo. Las capturas que se presentarán a continuación fueron obtenidas utilizando la herramienta Power Profiler mencionada anteriormente.

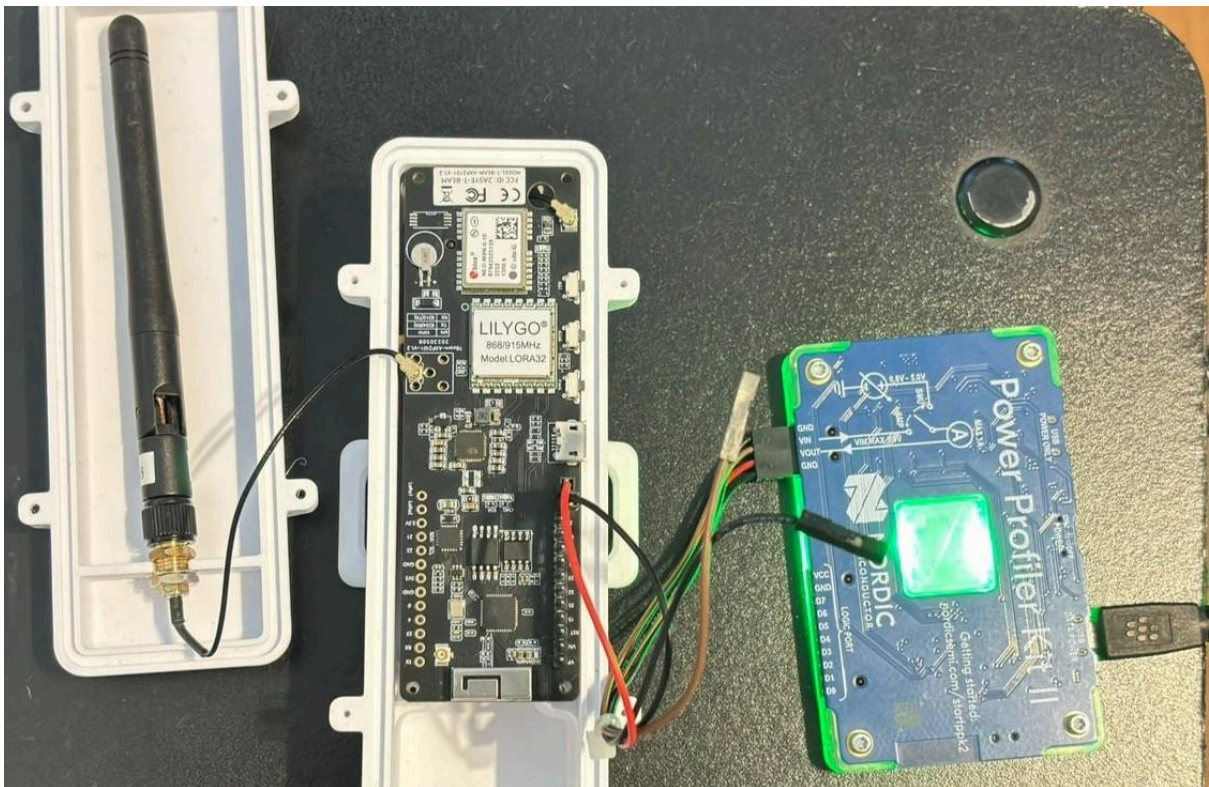


Figura 7.5: Fotografía de la prueba

7.2.1. Despertar de Deep Sleep

Para evaluar el consumo en este estado, primero necesitamos que el dispositivo entre en modo deep sleep y luego despertarlo utilizando un temporizador, ya que ese será el mecanismo de activación empleado en la versión final. A continuación, en la figura 7.6 podremos ver la corriente de consumo en este estado. Esta captura del programa nRF Connect corresponde a una muestra de 10 segundos en la que se observa la transición del

estado de deep sleep al estado de funcionamiento activo del dispositivo. Como referencia, la corriente durante el proceso de despertar está resaltada con un recuadro violeta, mientras que los estados de deep sleep y activo están marcados en naranja y verde, respectivamente.

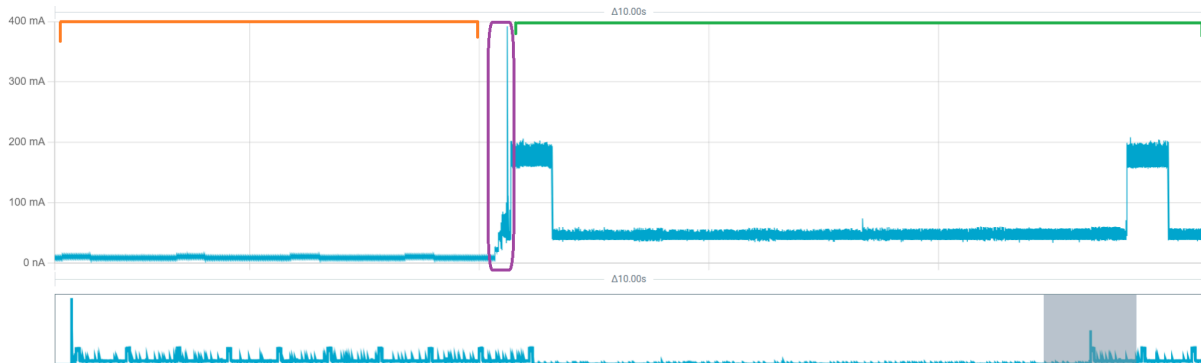


Figura 7.6: Transición de Deep Sleep a Activo

Al ampliar la escala a una muestra de 1 segundo, centrada en el recuadro violeta, obtenemos la captura que se muestra en la figura 7.7. En el programa, se permite seleccionar una zona específica de la muestra para visualizar las variables de interés, la cual está indicada con un sombreado gris. Podemos observar que este estado dura 135 milisegundos y tiene un consumo promedio de 45 mA.

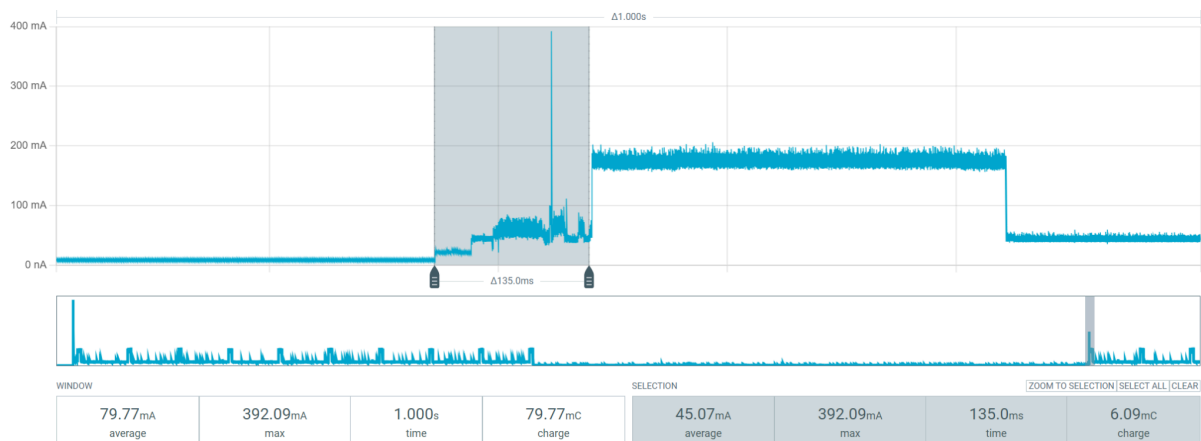


Figura 7.7: Transición de Deep Sleep a Activo ampliado

7.2.2. Búsqueda de coordenadas de GPS

Un estado de alto consumo en la solución se presenta cuando no se logra obtener una coordenada GPS de manera inmediata. Esto ocurre porque el sistema de GPS necesita

visualizar un número mínimo de satélites para calcular la ubicación, como se detalló anteriormente en la sección 3.1.8.2.

Al colocar el dispositivo en un lugar completamente aislado y bloqueado como lo es un sótano, logramos alcanzar un estado donde el sistema de GPS no logra calcular su ubicación repetidas veces. Simulando así una posible situación de alto consumo para la solución.

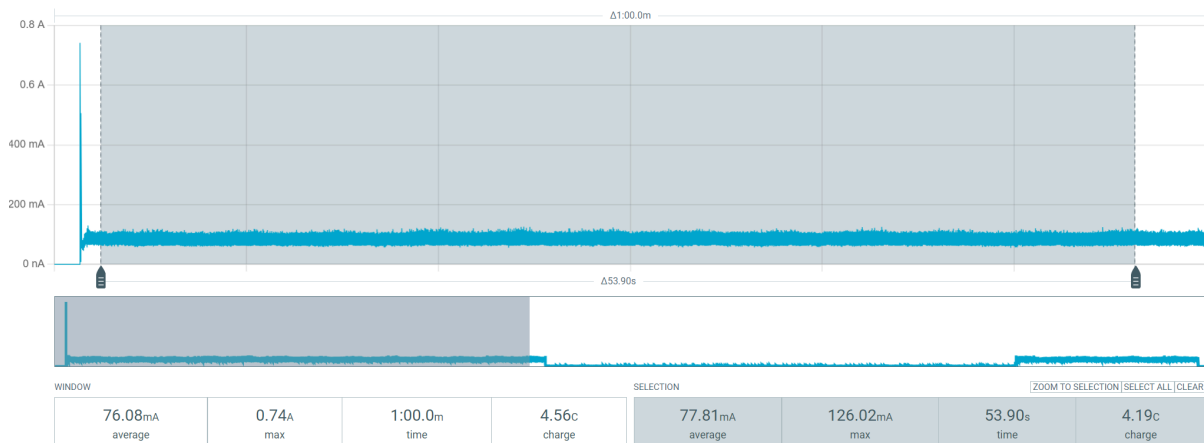


Figura 7.8: Búsqueda de coordenadas GPS

Este estado se mantendrá mientras dure activo el sistema, por lo tanto buscamos controlar su duración en el código. El consumo de corriente promedio de la solución con el GPS encendido es de 77,81 mA como podemos ver en la figura 7.8.

7.2.3. Envío por LoRa

Como buscamos recopilar la mayor cantidad de información posible sobre nuestro sistema, es necesario realizar muchos envíos por día. Por esta razón, el envío a través de LoRa será una tarea frecuente para nuestro dispositivo. La duración promedio de los envíos es de 358 milisegundos y su consumo de 175 mA. Por otro lado, entre los envíos con el transceptor encendido el consumo promedio es de 43,30 mA.

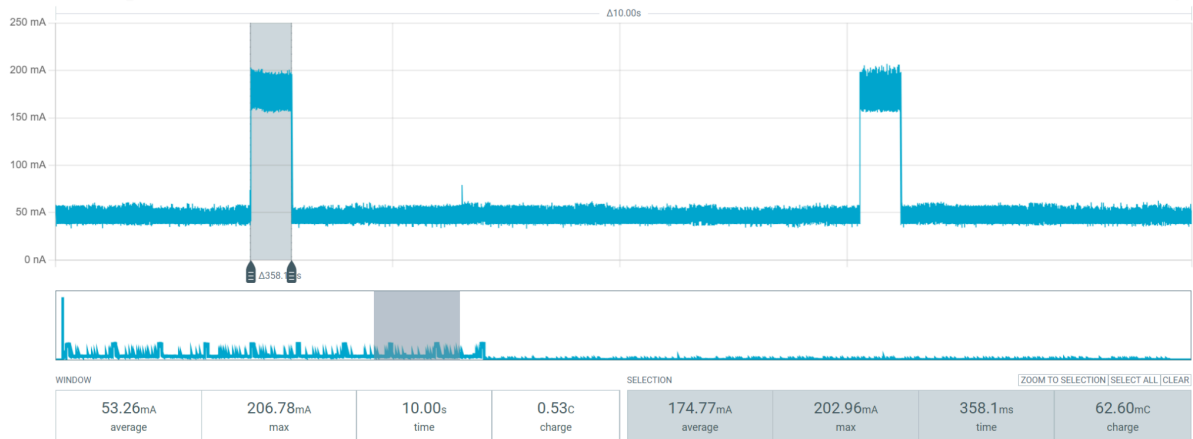


Figura 7.9: Consumo de envío por LoRa

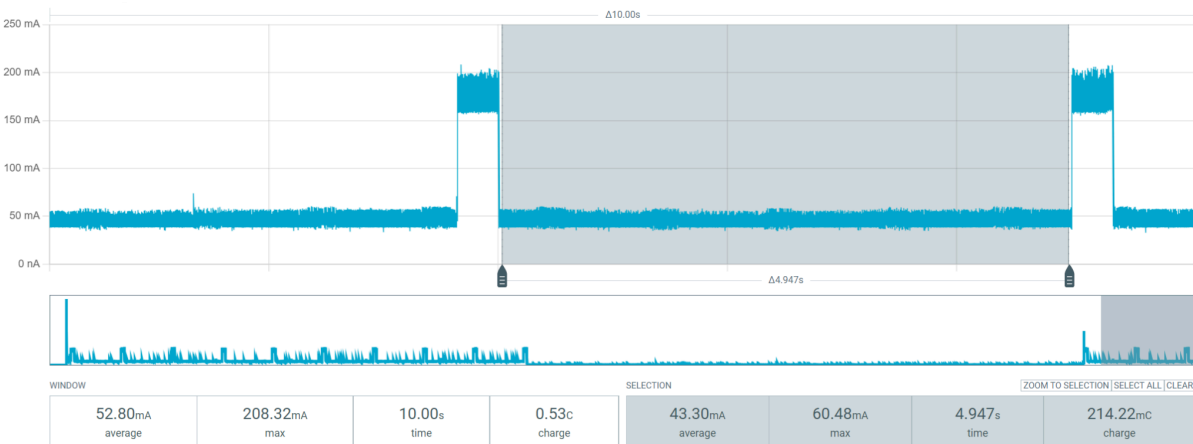


Figura 7.10: Consumo entre envíos

7.2.4. Deep Sleep

Este es el estado en el que el dispositivo pasa la mayor parte del tiempo, por lo que su consumo será determinante para la autonomía del mismo. Su consumo promedio en estado de deep sleep es de 9,04 mA.

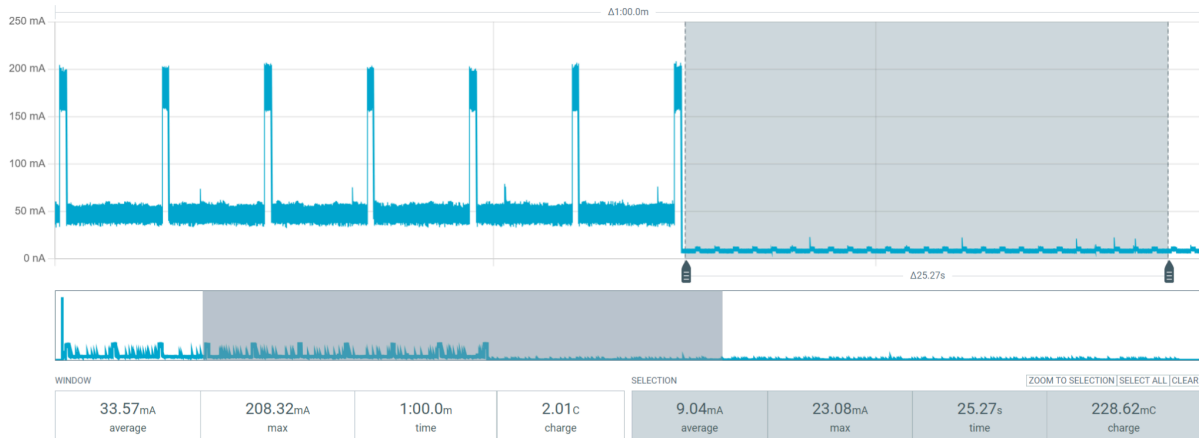


Figura 7.11: Consumo en estado de Deep Sleep

7.2.5. Alcance de la autonomía del sistema

Posibles estados	Duración de estado (ms)	Consumo (mA)	Ocurrencias (cada 24 horas)	% de tiempo en cada estado (Estándar)	% de tiempo en cada estado (Peor caso)
Deep Sleep a Activo	135	45	48	0.0075%	0.0075%
Módulo GPS encendido	3000 - 30000	77.81	48	0.1667%	1.6667%
Envío LoRa	358	175	144	0.0597%	0.0597%
LoRa encendido	10000	43.3	48	0.5556%	0.5556%
Deep Sleep	1.800.000	9.04	-	99.2106%	97.7106%
			Consumo promedio (mAh)	9,45	10,48
			Autonomía (días)	15,44	13,92

Figura 7.12: Tabla de resumen de consumos

La autonomía del sistema presenta diferencias significativas en función de dos escenarios evaluados:

- En el escenario estándar, el consumo promedio es de 9.45 mAh, lo que resulta en una autonomía aproximada de 15,44 días.
- En el peor caso, el consumo promedio aumenta a 10,48 mAh, reduciendo la autonomía a 13,92 días.

El principal factor que diferencia estos escenarios es el tiempo de encendido del módulo GPS. En el escenario estándar, el GPS permanece activo diez veces menos que en el peor

caso, lo que permite que el sistema pase una mayor proporción del tiempo en estados de menor consumo. Este análisis queda respaldado por la tabla de consumos, que muestra claramente que la reducción de aproximadamente 1,5 días en la autonomía se debe al mayor porcentaje de tiempo en que el GPS está encendido en el peor caso

Para maximizar la autonomía, se ha implementado una gestión eficiente de los estados de alto consumo, como lo son el encendido de los módulos GPS y LoRa. En cuanto al módulo LoRa, se optó por reducir la cantidad de envíos por ciclo de encendido. Esta decisión se basa en el hecho de que los mensajes enviados en intervalos de tan solo tres segundos muestran una alta redundancia, lo que los hace innecesarios en términos operativos y perjudiciales para la autonomía del sistema.

Por su parte, el módulo GPS ha sido configurado con un sistema de apagado basado en un tiempo límite, lo que evita que el dispositivo permanezca encendido innecesariamente si no logra obtener suficientes señales para calcular las coordenadas correctas. Al aplicar estas estrategias, se minimiza el tiempo que el sistema pasa en estados de alto consumo, permitiendo que se mantenga en el estado de menor consumo (Deep Sleep) el mayor tiempo posible.

7.3. Alcance de GPS

Los resultados de las pruebas indicaron que el GPS funcionaba de manera correcta tanto en estado estacionario como en movimiento. Los datos de ubicación fueron precisos y no se observaron pérdidas significativas de señal.

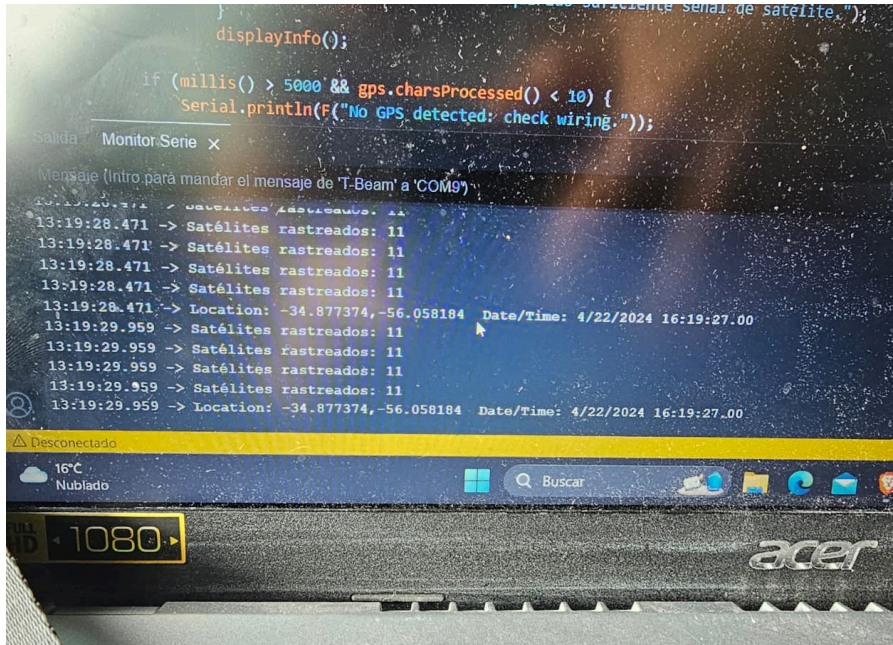


Figura 7.13: Ubicación recibida por GPS

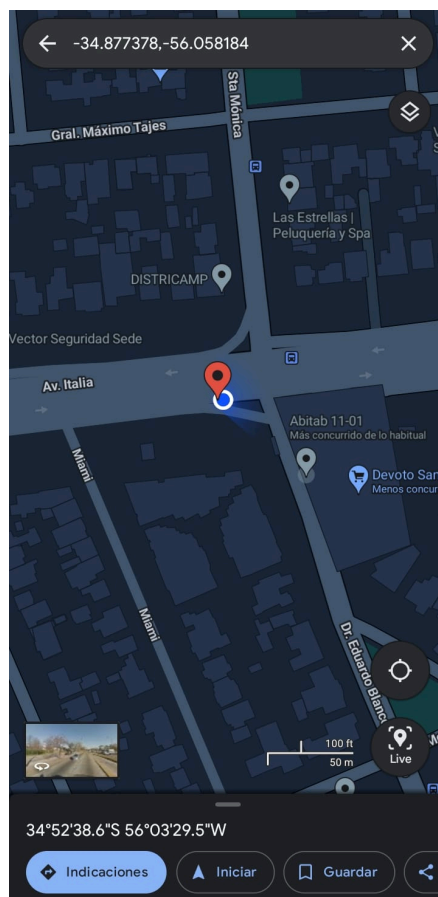


Figura 7.14: Corroboración de la Ubicación

Durante las pruebas, se observó que la hora entregada por el módulo GPS estaba desfasada respecto a nuestra zona horaria local. Esto se debe a que el GPS proporciona la hora en el estándar GMT 0 (*Greenwich Mean Time*), mientras que nuestra ubicación se encuentra en la zona horaria GMT -3.

También se notó que el GPS comenzaba a proporcionar ubicaciones precisas a partir de la conexión con al menos 3 satélites. A partir de este umbral, las lecturas de posición se estabilizaron y mostraron un alto grado de precisión. Este comportamiento es típico en los sistemas GPS, donde la triangulación básica de señales requiere un mínimo de tres satélites para determinar una ubicación bidimensional fiable (latitud y longitud), mientras que una mayor cantidad de satélites mejora la precisión y añade la capacidad de cálculo en tres dimensiones.

7.4. Alcance de comunicación con gateway

Probamos e iteramos con el protocolo planteado inicialmente (TTL a RS-232), simulando los posibles mensajes enviados del receptor al gateway con el programa de monitoreo serial Device Monitoring Studio.

Este procedimiento permite diagnosticar con facilidad los posibles fallos en el proceso de conversión. Además, validamos lo recibido en el gateway transmitiendo con RS-232 (ver figura 7.15). Más adelante, se utilizaron mensajes de diferentes tamaños para evaluar la velocidad de lectura y determinar los límites o restricciones de esta comunicación.

Durante el transcurso de la prueba surgieron varios problemas, como falsos contactos constantes y el sobrecalentamiento del conversor, lo que provocaba interrupciones en la comunicación. Además, quedó en evidencia la necesidad de mantener el gateway y el receptor conectados físicamente, exigiéndonos diseñar un soporte o carcasa extra para almacenarlos juntos.

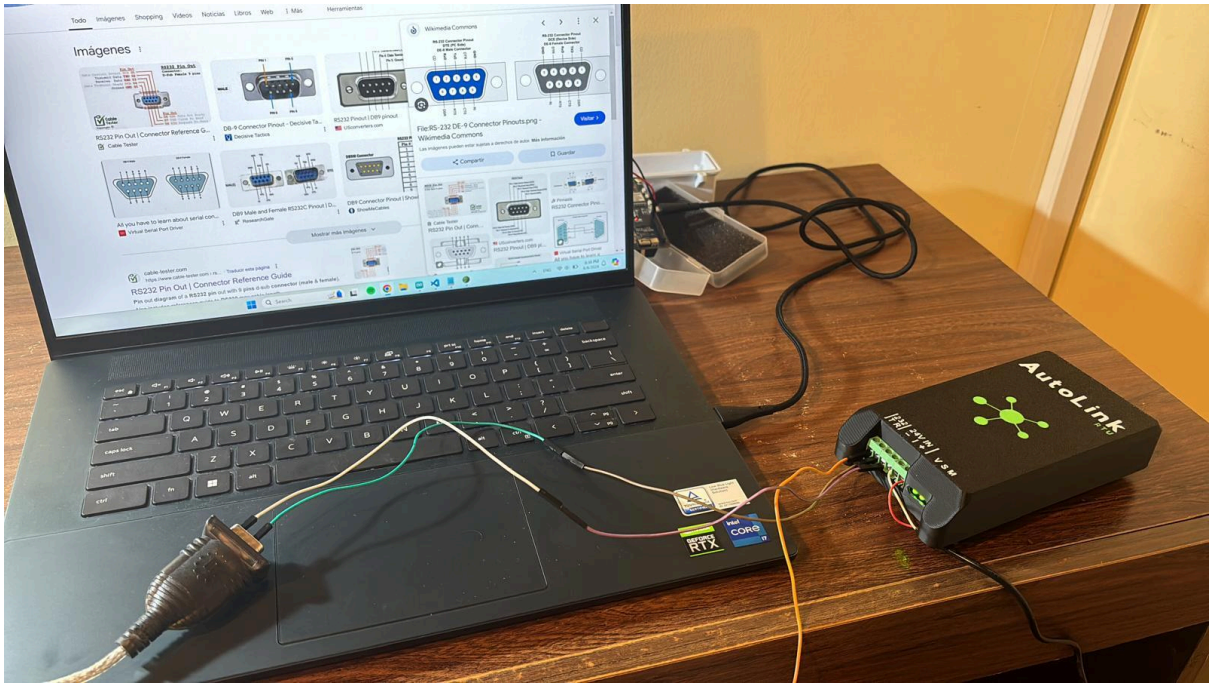


Figura 7.15: Prueba de recepción de RS232 del gateway

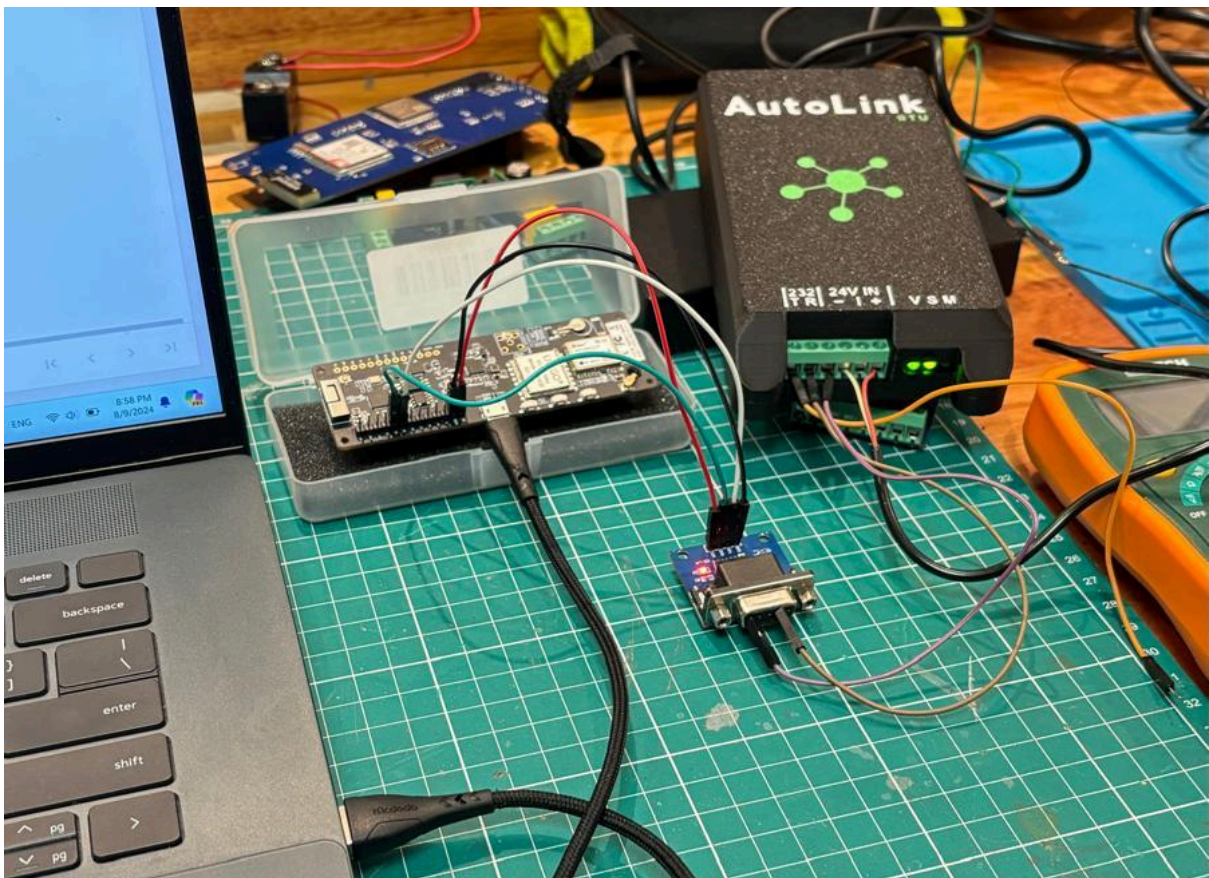


Figura 7.16: Receptor-gateway con comunicación TTL a RS232

Por todas estas limitantes, es que decidió seguir investigando para encontrar alguna manera de evitar esta combinación. Esto llevó a explorar nuevas vías de comunicación entre microcontroladores, especialmente porque tanto el emisor como el gateway tienen un microcontrolador ESP32 como parte de su arquitectura.

Como resultado de lo expuesto anteriormente, se procedió a experimentar con el protocolo de comunicación ESP-NOW. El uso de ESP-NOW elimina la limitante de requerir una conexión directa entre el receptor y el gateway. Además, resuelve los problemas de parseo y los inconvenientes de falso contacto asociados al uso de un conversor. Al realizar pruebas de funcionamiento, se logró una comunicación estable hasta un rango de 80 metros, similar a la cobertura de una red WiFi. Estos resultados confirman la viabilidad de ESP-NOW como protocolo de comunicación entre el gateway y el receptor.

7.5. Alcance de integración de datos con Thingsboard

Las pruebas realizadas permitieron validar las diversas funcionalidades del sistema. En particular, se emplearon solicitudes HTTP que facilitaron la simulación de interacciones con la plataforma, permitiendo analizar detalladamente las respuestas del servidor. Este enfoque resultó clave para depurar las secuencias lógicas implementadas en la plataforma. Se utilizó el comando curl para enviar las solicitudes, aprovechando la capacidad de la plataforma de operar en modo debug, lo que permitió registrar y monitorear los eventos a medida que estos atravesaban cada uno de los bloques de la cadena lógica. De esta manera, fue posible identificar con precisión los problemas relacionados con las integraciones, así como la configuración de dispositivos y tokens. Las funcionalidades validadas con este método fueron:

- Alarma por falta de cobertura GPS de un emisor
- Alarma en caso de salida del terreno de un emisor
- Notificación vía mail por salidas del terreno
- Visualización de emisores de manera simultánea en el mapa
- Relación de telemetrías tanto en clase Dispositivo como en clase Activo

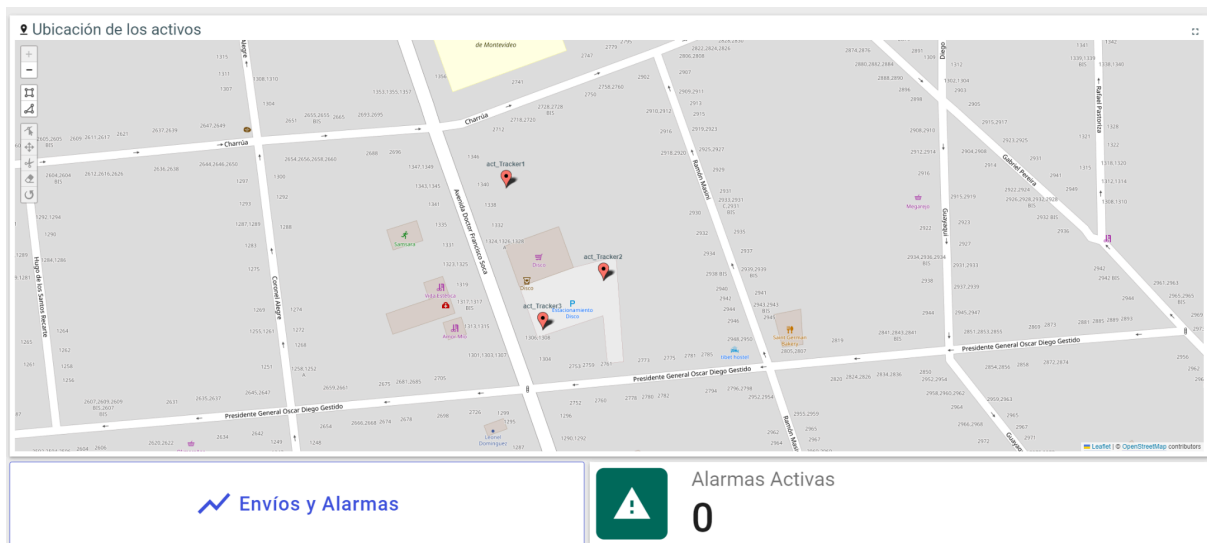


Figura 7.17: Panel gráfico inicial en la plataforma

En caso de pérdida de cobertura GPS, el sistema genera una alarma con una notificación en la plataforma, indicando cuál de los emisores ha perdido cobertura. Por otro lado, para detectar la salida de un área determinada, se utiliza el concepto de geovallado, que consiste en establecer un perímetro virtual basado en coordenadas geográficas. Este perímetro es definido en la lógica de la plataforma cargando las coordenadas específicas para determinar el límite. Si un activo vinculado a la plataforma cruza esta zona delimitada, se envía una notificación por correo electrónico al usuario asociado, indicando que el activo ha abandonado el área de geovallado.

7.6. Alcance de Alarmas de Salida de Terreno (Geovallado)

En este apartado nos centraremos específicamente en la configuración y funcionamiento de las alarmas en Thingsboard, un aspecto clave que complementa la integración de datos discutida previamente. Mientras que en el apartado anterior abordamos cómo manejar y almacenar datos en la plataforma, ahora exploramos la capacidad de configurar y gestionar alarmas, destacando su importancia en la detección de eventos críticos y en la toma de decisiones en tiempo real.



Figura 7.18: Captura de la regla de cadena con el bloque script “Filtro coordenadas”



Figura 7.19: Código en el bloque “Filtro coordenadas” para validar el geovallado

Durante la prueba, se logró configurar exitosamente el área geográfica y las reglas de alarma en Thingsboard. El sistema respondió correctamente al detectar la salida del dispositivo del área establecida, enviando una notificación por correo electrónico y generando una alerta visible en el dashboard. Este comportamiento demostró la efectividad de la configuración para notificar cambios de ubicación no autorizados. En la sección de reglas en cadena de la plataforma, se implementó el área de geovallado mediante un script que define las coordenadas de latitud y longitud que delimitan la zona a monitorear. Cuando la ubicación recibida está fuera de estos límites, se genera automáticamente una alarma como se muestra en las figuras a continuación.

Notificación

Marcar todas como leídas

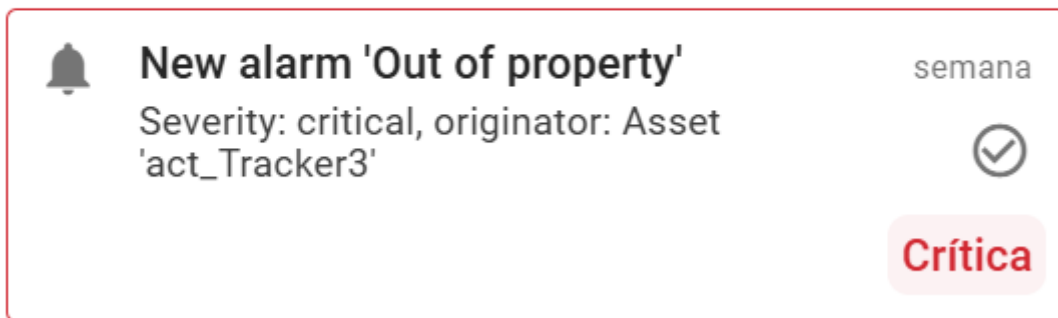


Figura 7.20: Captura de la notificación en la plataforma

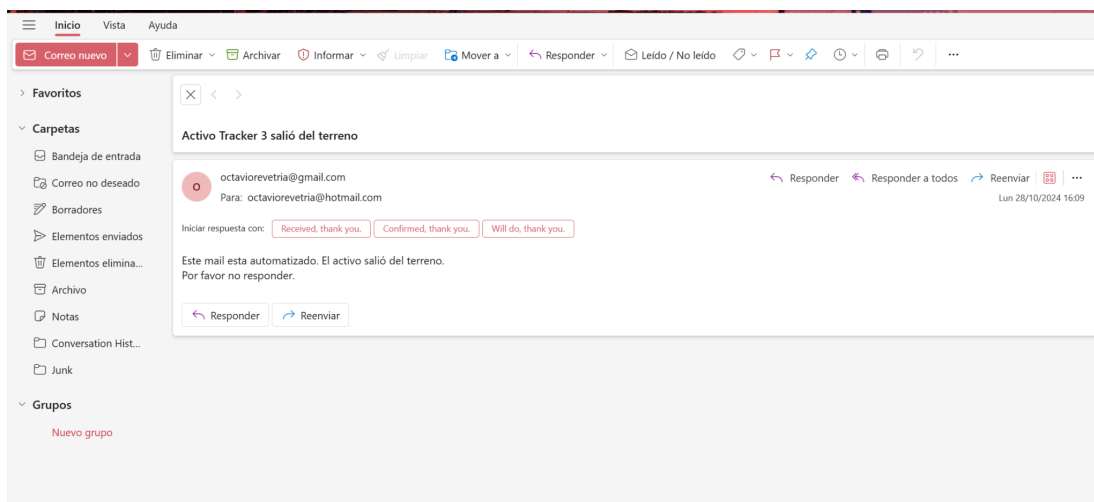


Figura 7.21: Captura de la notificación por correo electrónico

8. Conclusión del MVP

En conclusión, el desarrollo de nuestro Producto Mínimo Viable ha logrado cumplir con creces los objetivos establecidos.

El diseño compacto y robusto se ha cumplido con éxito. Empezando por la elección de una placa de desarrollo de tamaño reducido, se diseñaron dos carcasas especializadas: una para la transmisión y otra para la recepción. Este punto es validado, ya que para ambos módulos se obtuvo una solución con dimensiones de 46x139x53 milímetros y con un peso de 124 gramos, cumpliendo así con los requisitos de portabilidad planteados.

La capacidad de obtener una ubicación lo más precisa posible también se ha alcanzado de manera sobresaliente, garantizando datos confiables y exactos en todo momento. Al utilizar un módulo GPS como el NEO-M8 la solución ofrece una exactitud horizontal de hasta 2.5 metros, lo cual es ideal más que suficiente para entornos rurales donde no se necesita un nivel de precisión muy alto. Con esta capacidad, el sistema garantiza que los datos de localización sean precisos, compartiendo una correcta ubicación de los distintos nodos.

Sistema resistente y fácil de implementar en diversos entornos. El sistema asegura una cobertura eficiente tanto en la transmisión de datos como en la comunicación entre dispositivos. En cuanto al gateway, la conectividad LTE proporciona una cobertura que abarca la mayor parte del territorio de Uruguay, facilitando la transmisión estable de datos. Para la comunicación entre dispositivos, el uso de LoRa para la transmisión de datos locales garantiza un alcance teórico superior a 1,5 km, que puede incrementarse en áreas rurales debido a la baja cantidad de obstáculos, optimizando aún más la cobertura del sistema (ver sección 3.1.4). En resumen, hemos logrado una conectividad amplia y estable, incluso en áreas remotas, cumpliendo ampliamente con este objetivo. [63]

El sistema ha logrado mantener un costo accesible sin comprometer su funcionalidad, lo que refuerza la viabilidad del proyecto. A pesar de que las soluciones IoT en la industria rural están aún en desarrollo, lograr las mismas funcionalidades de forma independiente en un placa diseñada desde cero habría implicado un costo considerablemente mayor que el obtenido con esta solución. Esto evidencia que las placas de desarrollo y las herramientas

gratuitas disponibles pueden proporcionar una solución económica sin comprometer la calidad y la eficiencia requeridas en entornos rurales.

Se priorizó la optimización energética del sistema para prolongar la autonomía en entornos remotos. El sistema ha demostrado tener una autonomía de entre 14 y 15 días, lo que resulta adecuado para un sistema de transmisión a distancia con las dimensiones de nuestra solución y reduce la necesidad de intervenciones frecuentes para reemplazar los emisores por causa de batería agotada, disminuyendo así la mano de obra asociada al funcionamiento de la solución.

El gateway de datos ha funcionado de manera fiable, logrando una transmisión continua hacia la nube. Esto asegura que los datos recolectados se almacenen de forma segura y accesible para análisis posteriores, lo que refuerza la capacidad del sistema para integrarse plenamente dentro de la solución final sin complicaciones técnicas.

El sistema ha sido diseñado para ofrecer una interfaz intuitiva y fácil de usar, accesible tanto desde una plataforma web como desde dispositivos móviles. Esto garantiza que los usuarios puedan monitorear el estado del sistema en tiempo real, independientemente de su ubicación, mejorando la capacidad de gestión y la usabilidad del sistema en diferentes entornos.

Finalmente, el producto resuelve varios de los problemas planteados desde un inicio. Como lo son la falta de cobertura en áreas remotas por medio de cobertura LTE, comunicación LoRa y ESP-NOW. Aporta herramientas para combatir el abigeato con alarmas que notifican por correo electrónico si el ganado ha abandonado el terreno. A su vez, plantea un sistema intuitivo y accesible tanto vía mobile como web para la comodidad del usuario.

9. Caso de Uso en el Monitoreo de Ganado y Plan de Mantenimiento

En este proyecto, cada animal en el grupo de ganado se equipará con un dispositivo de monitoreo que permitirá la recopilación y transmisión de datos importantes para la gestión de los recursos y la seguridad de los animales. La relación dispositivo/animal se ha diseñado para garantizar que cada unidad pueda captar información relevante, como la posición GPS y transmitirla de forma eficiente a un nodo central. Esta asignación de un dispositivo por animal busca optimizar la cobertura y la precisión de los datos, permitiendo una monitorización en tiempo real y facilitando la toma de decisiones.

9.1. Implementación Práctica del Caso de Uso

En el futuro, la implementación del sistema podría comenzar con la colocación de los dispositivos en el ganado, asegurando que cada dispositivo esté adecuadamente fijado y protegido contra factores ambientales, como la lluvia y el polvo. Una vez instalados, los dispositivos se configurarán para transmitir datos a intervalos regulares hacia un nodo central, que se encargará de recopilar y procesar la información de múltiples animales.

9.2. Plan de Mantenimiento de los Dispositivos

El plan de mantenimiento para futuros desarrollos incluiría revisiones periódicas y acciones correctivas en caso de fallos. Las tareas de mantenimiento previstas se realizarán cada dos semanas, coincidiendo con el ciclo de duración de las baterías de los dispositivos, que es de aproximadamente 15 días. Durante estas revisiones, se aprovecharía para inspeccionar visualmente los dispositivos, detectar posibles daños físicos o mal funcionamiento debido al desgaste o exposición al medio ambiente, y realizar el cambio de baterías de manera eficiente.

Para llevar a cabo esta tarea, sería necesario ubicar al ganado en un espacio adecuado donde se facilite el acceso a los dispositivos. Este lugar es comúnmente conocido como manga o corral de manejo, donde los animales se colocan en fila, uno detrás de otro, para poder ser manipulados con seguridad. Esto permitiría realizar las revisiones y los cambios de baterías

de forma organizada y segura, minimizando el estrés en los animales y optimizando el tiempo de mantenimiento.

10. Otros posibles usos

En este capítulo, se explorarán otros posibles usos de las tecnologías investigadas y de la solución desarrollada durante este proyecto. Aunque la implementación original fue diseñada para un caso específico, estas herramientas y sistemas pueden aplicarse a diversas problemáticas en otros ámbitos. El objetivo es destacar la flexibilidad y adaptabilidad de la solución, mostrando cómo los componentes de hardware, los protocolos de comunicación, y la plataforma de software pueden ser reutilizados o adaptados para resolver diferentes desafíos en áreas que requieren soluciones tecnológicas similares.

10.1. Tracking de Flotas Vehiculares

Una aplicación alternativa de la solución desarrollada es el seguimiento de flotas vehiculares. Utilizando el GPS integrado, el gateway y la plataforma IoT, se podría implementar un sistema de monitoreo en tiempo real para vehículos, lo que permitiría obtener la ubicación exacta de cada unidad, rastrear rutas y gestionar flotas de manera eficiente. Este sistema no solo optimizaría la logística, sino que también podría proporcionar datos clave sobre el rendimiento del vehículo, tiempos de inactividad y mantenimiento preventivo.

10.2. Monitoreo de Infraestructura Crítica

El sistema de GPS y comunicación LoRa podría ser adaptado para monitorear infraestructuras críticas, como puentes, presas o redes eléctricas. Sensores distribuidos podrían reportar datos sobre vibraciones, inclinaciones o cambios estructurales en tiempo real, permitiendo el mantenimiento preventivo y la detección temprana de fallos.

10.3. Gestión de Inventarios en Almacenes

Aplicando la misma tecnología de rastreo, el sistema podría ser utilizado para monitorizar en tiempo real el inventario de grandes almacenes o centros logísticos. Cada pallet o contenedor podría estar equipado con un dispositivo GPS y un módulo de comunicación, facilitando el seguimiento y optimización del flujo de productos dentro de los almacenes.

10.4. Optimización de Riego en Agricultura

El sistema podría ser aplicado para monitorear la humedad del suelo y gestionar de manera remota el riego de campos agrícolas extensos. A través de la combinación de GPS, sensores y comunicación LoRa, se podría optimizar el uso del agua y mejorar la eficiencia de los recursos hídricos, el cual en un ámbito de sequía tan importante como la vivida en el año 2022/2023 en Uruguay es fundamental el extremo control de los recursos hídricos.

11. Mejoras para el futuro

11.1. Hardware

En cuanto al hardware, se identifican varias áreas de mejora que podrían implementarse en versiones futuras del sistema, con el objetivo de optimizar su rendimiento, eficiencia y adaptabilidad a diversas aplicaciones. A continuación, se detallan algunos puntos clave que se podrían mejorar:

11.1.1. Construcción de una Placa Propia

Una de las mejoras más importantes sería el diseño y construcción de una placa de circuito personalizada. Esto permitiría seleccionar componentes más eficientes en términos de consumo energético, ajustados específicamente a las necesidades del sistema, sacrificando algo de costo a favor de una mayor autonomía y optimización del rendimiento. Al tener control total sobre el diseño del hardware, podríamos integrar sólo los módulos necesarios, eliminando redundancias y reduciendo el tamaño físico del dispositivo.

11.1.2. Incorporación de Módulos de Energía Solar Optimizada

A futuro, se podría integrar directamente en el sistema un controlador de carga solar más eficiente, diseñado específicamente para el tipo de batería y panel solar utilizados. Esto optimizaría la gestión de energía y garantiza una mejor autonomía en entornos remotos o con acceso limitado a fuentes de energía convencionales.

11.1.3. Sensores Adicionales y Mayor Capacidad de Monitoreo

Para aumentar las capacidades del sistema, se podría considerar la integración de sensores adicionales, como sensores de temperatura, humedad o presión, para ampliar las aplicaciones del dispositivo en distintos entornos. Estos sensores podrían monitorear el estado ambiental o estructural, mejorando la capacidad de recolección de datos.

11.1.4. Optimización del Diseño para Robustez Física

Se podría optimizar el diseño de la carcasa para mejorar su resistencia física y aumentar la protección del hardware frente a condiciones adversas, como polvo, agua o impactos. Esto es especialmente relevante para aplicaciones en entornos industriales, donde la durabilidad del equipo es crítica.

11.1.5. Evaluación de una Arquitectura Tipo Mesh

A futuro, y basándose en los datos recolectados y los patrones de comportamiento observados en el ganado, se evaluará la viabilidad de implementar una arquitectura tipo mesh. Esta estructura permitiría una mayor centralización de los datos y reduciría la necesidad de transmisores individuales en cada animal. El análisis se enfocará en determinar cuántos dispositivos podrían agruparse sin afectar la precisión y la rapidez de la transmisión de datos. Además, la implementación de esta arquitectura podría mejorar la relación costo-eficiencia al disminuir el número de transmisores requeridos y, por ende, el costo total del sistema.

11.2. Software

En el ámbito del software, existen varias áreas en las que se pueden introducir mejoras para optimizar el rendimiento del sistema, aumentar su funcionalidad y facilitar su mantenimiento a largo plazo. A continuación, se detallan algunos puntos clave que podrían mejorarse:

11.2.1. Desarrollo de Algoritmos de Gestión de Energía más Avanzados

Se podría implementar un sistema de gestión de energía más sofisticado, que ajuste dinámicamente el consumo de los diferentes componentes en función de las condiciones operativas y los requerimientos de cada momento. Esto podría incluir modos de suspensión más eficientes y la optimización de la transmisión de datos para maximizar la autonomía del dispositivo en situaciones de bajo consumo.

11.2.2. Ampliación de la Plataforma IoT con Análisis de Datos

Integrar capacidades de análisis de datos dentro de la plataforma IoT, permitiendo realizar procesamiento de los datos recolectados de manera más avanzada. Esto podría incluir análisis

predictivo o la detección de patrones, facilitando la toma de decisiones basadas en los datos recopilados y ofreciendo un valor añadido a la solución en términos de monitoreo y control.

11.2.3. Implementación de Actualizaciones de Firmware OTA

Añadir la capacidad de realizar actualizaciones de firmware de manera remota (OTA) sería una mejora significativa para la gestión del sistema, ya que permitiría realizar ajustes, correcciones de errores y añadir nuevas funcionalidades sin necesidad de acceso físico a los dispositivos, lo que es especialmente útil en entornos de difícil acceso.

11.2.4. Mejoras en la Seguridad de los Datos

Implementar protocolos de seguridad más robustos, como cifrado de extremo a extremo y autenticación de múltiples factores, para garantizar que los datos transmitidos entre los dispositivos y la plataforma de control estén protegidos frente a interceptaciones o accesos no autorizados. Esta mejora es crítica en aplicaciones sensibles o en las que la integridad de los datos es prioritaria.

12. Referencias Bibliográficas

[1] IERC, “IERC - European Reaserch Cluster on the Internet of Things”, 2014.

[En línea]. Disponible en: http://www.internet-of-things-research.eu/about_iiot.htm.
[Último acceso: Agosto 2024]

[2]Alai Secure, “¿Qué es y cómo funciona LTE-M Technology?”, Agosto. 2024.

[En línea]. Disponible en: <https://alaisecure.es/glosario/que-es-y-como-funciona-lte-m-technology/>. [Último acceso: Agosto 2024]

[3] URSEC, “Asignación de frecuencia para IMT”, Agosto. 2024.

[En línea]. Disponible en: <https://www.gub.uy/unidad-reguladora-servicios-comunicaciones/datos-y-estadisticas/datos/asignacion-frecuencia-para-imt>. [Último acceso: Agosto 2024]

[4] Taoglas, “Low Profile LTE/Cellular 4G/3G/2G & GNSS SMD Antenna”, Agosto. 2024.

[En línea].Disponible en:<https://www.taoglas.com/datasheets/PCS.50.A.pdf>. [Último acceso: Agosto 2024]

[5] Telefonica, “¿Qué es NB-IoT y cómo funciona?”, Agosto. 2024.

[En línea]. Disponible en: <https://www.telefonica.com/es/sala-comunicacion/blog/que-es-nb-iiot-y-como-funciona/>. [Último acceso: Agosto 2024]

[6]EIMT, “¿Qué es NB-IoT?”, Nov. 2018.

[En línea].Disponible en:<https://blogs.uoc.edu/informatica/que-es-nb-iiot/>. [Último acceso: Agosto 2024]

[7]Carlos Espinoza, “¿Qué es NarrowBand IoT y Cómo Funciona?”, Mar. 2018.

[En línea]. Disponible en: <https://www.syscomblog.com/2019/03/que-es-narrowband-iiot-y-como-funciona.html>. [Último acceso: Agosto 2024]

[8]Sigfox, “¿Qué es Sigfox?”, 2021.

[En línea]. Disponible en: <https://www.sigfox.com.py/que-es-sigfox/>. [Último acceso: Septiembre 2024]

[9] Bemens, “¿Qué son las frecuencias ISM?”, 2022.

[En línea]. Disponible en: <https://www.bemens.com/que-son-las-frecuencias-ism/>. [Último acceso: Septiembre 2024]

[10] URSEC, "Cuadro de atribución de frecuencias de Uruguay," 2024.

[En línea]. Disponible en: <https://www.gub.uy/unidad-reguladora-servicios-comunicaciones/comunicacion/publicaciones/cuadro-atribucion-frecuencias>. [Último acceso: Noviembre 2024]

[11] Venco, “Qué es LoRa, cómo funciona y características principales”, Agosto, 2022.

[En línea]. Disponible en: <https://www.vencoel.com/que-es-lora-como-funciona-y-caracteristicas-principales>. [Último acceso: Septiembre 2024]

[12] CatSensors, “Tecnología LoRa y LoRaWAN”, Agosto, 2024.

[En línea]. Disponible en: <https://www.catsensors.com/es/lorawan/tecnologia-lora-y-lorawan>. [Último acceso: Septiembre 2024]

[13] Diego Fuentes, “LoRaWAN: La revolución de la conectividad en el Internet de las Cosas (IoT)”, Mayo, 2023.

[En línea]. Disponible en: <https://blog.zambia.cl/lorawan-que-debes-saber/>. [Último acceso: Septiembre 2024]

[14] M. A. Martínez, "ESP8266 y ESP32 con ESP-NOW: Comunicación inalámbrica rápida y eficiente," Emariete.com, Dec. 2020.

[En línea]. Disponible en: <https://emariete.com/esp8266-esp32-espnow/>. [Último acceso: Septiembre 2024]

[15] Espressif Systems, "ESP-NOW: Low Power Solutions," Espressif.com, Apr. 2019.

[En línea]. Disponible en: <https://www.espressif.com/en/solutions/low-power-solutions/esp-now>. [Último acceso: Septiembre 2024]

[16] Electrosoft, "ESP-NOW: Conecta dos o más ESP32/ESP8266," Electrosoftcloud.com, Jan. 2021.

[En línea]. Disponible en: <https://www.electrosoftcloud.com/esp-now-conecta-dos-o-mas-esp32-esp8266/>. [Último acceso: Septiembre 2024]

[17] Dipole RFID, "Tipos de Sistemas RFID," DipoleRFID.es, 2023.

[En línea]. Disponible en: <https://www.dipolerfid.es/blog-rfid/Tipos-Sistemas-RFID>. [Último acceso: Septiembre 2024]

[18] Dipole RFID, "¿Qué es RFID?," DipoleRFID.es, 2023.

[En línea]. Disponible en: <https://www.dipolerfid.es/blog-rfid/que-es-rfid>. [Último acceso: Septiembre 2024]

[19] Dipole RFID, "Etiquetas NFC Dipole," DipoleRFID.es, 2023.

[En línea]. Disponible en: <https://www.dipolerfid.es/producto/etiquetas-nfc-dipole>. [Último acceso: Septiembre 2024]

[20] Dipole RFID, "¿Qué es NFC?," DipoleRFID.es, 2021.

[En línea]. Disponible en: <https://www.dipolerfid.es/blog-rfid/que-es-nfc>. [Último acceso: Septiembre 2024]

[21] Tus Clases Particulares, "Descubriendo la Geometría Analítica," Tusclasesparticulares.com, 2023.

[En línea]. Disponible en: <https://www.tusclasesparticulares.com/blog/descubriendo-geometria-analitica>. [Último acceso: Septiembre 2024]

[22] Geotab, "¿Qué significa GPS?," Geotab.com, 2023.

[En línea]. Disponible en: <https://www.geotab.com/es-latam/blog/qu%C3%A9-significa-gps/>. [Último acceso: Septiembre 2024]

[23] Garmin, "Acerca de GPS," Garmin.com, 2023.

[En línea]. Disponible en: <https://www.garmin.com/es-ES/aboutgps/>. [Último acceso: Septiembre 2024]

[24] Cyberclick, "¿Qué es un dashboard?," Cyberclick.es, 2023.

[En línea]. Disponible en: <https://www.cyberclick.es/numerical-blog/que-es-un-dashboard>. [Último acceso: Septiembre 2024]

[25] Ibersontel, "ThingsBoard: Plataforma para monitorización de IoT," Ibersontel.com, 2023.

[En línea]. Disponible en: <https://ibersontel.com/thingsboard-plataforma-para-monitorizacion-de-iot/4730>. [Último acceso: Junio 2024]

[26] Node-RED, "About Node-RED," Node-RED.org, 2023.

[En línea]. Disponible en: <https://nodered.org/about/>. [Último acceso: Junio 2024]

[27] Grupo Sinelec, "¿Qué es Node-RED y para qué sirve?," Blog Grupo Sinelec, 2023.

[En línea]. Disponible en: <https://blog.gruposinelec.com/actualidad/que-es-node-red-y-para-que-sirve/>. [Último acceso: Junio 2024]

[28] Node-RED, "node-red-dashboard," Node-RED Flows, 2023.

[En línea]. Disponible en: <https://flows.nodered.org/node/node-red-dashboard>. [Último acceso: Junio 2024]

[29] Pandora FMS, "¿Qué es Grafana?," Pandora FMS Blog, 2023.

[En línea]. Disponible en: <https://pandorafms.com/blog/es/que-es-grafana/>. [Último acceso: Junio 2024]

[30] Ausum Cloud, "¿Qué es Grafana y cómo se usa en la monitorización?," Ausum Cloud, 2023.

[En línea]. Disponible en: <https://ausum.cloud/que-es-grafana-y-como-se-usa-en-la-monitorizacion/>. [Último acceso: Junio 2024]

[31] Advanced Motion Controls, "Comunicación RS-232", Julio. 2023.

[En línea]. Disponible en: <https://www.a-m-c.com/es/experiencia/technologies/peripheral-interface/rs-232/>. [Último acceso: Junio 2024]

[32] Olga Weis, "Guía de la comunicación RS485", Oct.2021.

[En línea]. Disponible en: <https://www.eltima.com/es/article/rs485-communication-guide/#:~:text=RS%2D485%20>. [Último acceso: Junio 2024]

[33] AWS, "Que es MQTT?", Marzo. 2023.

[En línea]. Disponible en: <https://aws.amazon.com/es/what-is/mqtt/>. [Último acceso: Junio 2024]

- [34] S. Bholá, “Why HTTP is not suitable for IOT applications”, Concurrency, Jun. 2019.
[En línea]. Disponible en: <https://www.concurrency.com/blog/june-2019/why-http-is-not-suitable-for-iot-applications>. [Último acceso: Junio 2024]
- [35] IONOS, “Advanced Message Queuing Protocol (AMQP)”, May.2019.
[En línea]. Disponible en: <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/advanced-message-queuing-protocol-amqp/>. [Último acceso: Junio 2024]
- [36] Drew, “MCU: Main control unit, ¿qué es?”, Junio. 2021.
[En línea]. Disponible en: <https://blog.wearedrew.co/mcu-main-control-unit-que-es>. [Último acceso: Junio 2024]
- [37] Espressif Systems, “Sistemas ESP32 Wi-Fi y Bluetooth MCU”, Julio. 2023
[En línea]. Disponible en: <https://www.espressif.com/en/products/socs/esp32>. [Último acceso: Junio 2024]
- [38] NORDIC Semiconductor, “nRF9160 - Semiconductor nordico”, Abril. 2023.
[En línea]. Disponible en: <https://www.nordicsemi.com/products/nrf9160>. [Último acceso: Junio 2024]
- [39] Arsys, "¿Qué es un gateway IoT?," Arsys Blog, 2023.
[En línea]. Disponible en: <https://www.arsys.es/blog/gateway-iot>. [Último acceso: Junio 2024]
- [40] Arimetrics, “Que es un Framework”, Noviembre. 2022.
[En línea]. Disponible en: <https://www.arimetrics.com/glosario-digital/framework#:~:text=Un%20framework%20es%20un%20marco,y%20la%20consistencia%20del%20c%C3%B3digo>. [Último acceso: Junio 2024]
- [41] ESPBoards, “ESP-IDF vs Arduino Core: qué marco elegir en 2023”, Enero. 2023
[En línea]. Disponible en: <https://www.espboards.dev/blog/esp-idf-vs-arduino-core/>. [Último acceso: Agosto 2024]

- [42] Isaac, "RTOS: que es un sistema operativo de tiempo real", Diciembre. 2020
[En línea]. Disponible en: <https://www.hwlibre.com/rtos/>. [Último acceso: Agosto 2024]
- [43] AWS, "Preguntas frecuentes sobre FreeRTOS", Julio. 2023.
[En línea]. Disponible en: <https://aws.amazon.com/es/freertos/faqs/#:~:text=FreeRTOS%20es%20un%20sistema%20operativo,peque%C3%B1os%20y%20de%20bajo%20consumo>. [Último acceso: Agosto 2024]
- [44] Autolink, "RTU," Autolink, 2023.
[En línea]. Disponible en: <https://autolink.com.uy/rtu>. [Último acceso: Agosto 2024]
- [45] u-blox, "NEO-M8J," Mouser Electronics, 2023.
[En línea]. Disponible en: <https://uy.mouser.com/ProductDetail/u-blox/NEO-M8J?qs=sGAEpiMZZMu3sxpav1qrlhWywTP0pnuh%252BmP8LamgMg%3D>. [Último acceso: Agosto 2024]
- [46] Semtech, "SX1262IMLTRT," Mouser Electronics, 2023.
[En línea]. Disponible en: <https://uy.mouser.com/ProductDetail/Semtech/SX1262IMLTRT?qs=XFmntM7Yc9eYt6yxTPughQ%3D%3D>. [Último acceso: Agosto 2024]
- [47] Espressif Systems, "ESP32-C3-WROOM-02-H4," Mouser Electronics, 2023.
[En línea]. Disponible en: <https://uy.mouser.com/ProductDetail/Espressif-Systems/ESP32-C3-WROOM-02-H4?qs=stqOd1AaK79%2FSA14oWtR%252Bw%3D%3D>. [Último acceso: Agosto 2024]
- [48] LilyGo, "T-beam SoftRF," LilyGo, 2023.
[En línea]. Disponible en: <https://www.lilygo.cc/products/t-beam-softrf?variant=43170139865269>. [Último acceso: Agosto 2024]
- [49] u-blox, "NEO-M8 Product Summary," u-blox, 2023.
[En línea]. Disponible en: https://content.u-blox.com/sites/default/files/products/documents/NEO-M8_ProductSummary_UBX-16000345.pdf. [Último acceso: Agosto 2024]

[50] Semtech, "SX1276 - LoRa Transceiver," Semtech, 2023.

[En línea]. Disponible en: <https://www.semtech.com/products/wireless-rf/lora-connect/sx1276>. [Último acceso: Agosto 2024]

[51] Amazon, "Producto disponible en Amazon," Amazon España, 2023.

[En línea]. Disponible en: https://www.amazon.es/dp/B0CMXF6V21?ref=emc_s_m_5_i_atc. [Último acceso: Septiembre 2024]

[52] Jinftry, "AXP2101 Datasheet PDF," Jinftry, 2023.

[En línea]. Disponible en: <https://es.jinftry.com/news/AXP2101-Datasheet-PDF>. [Último acceso: Septiembre 2024]

[53] EMQX, "How to Use MQTT in Python," *EMQX Blog*, 2023.

[En línea]. Disponible en: <https://www.emqx.com/en/blog/how-to-use-mqtt-in-python#why-choose-paho-mqtt-python-client>. [Último acceso: Septiembre 2024]

[54] DigitalOcean, "ThingsBoard Professional Edition," *DigitalOcean Documentation*, 2023.

[En línea]. Disponible en: <https://docs.digitalocean.com/products/marketplace/catalog/thingsboard-professional-edition/>. [Último acceso: Septiembre 2024]

[55] 3Dnatives, "Fusion 360: el software CAD/CAM todo en uno," 3Dnatives, 2020.

[En línea]. Disponible en: <https://www.3dnatives.com/es/fusion-360-software-29042020/>. [Último acceso: Septiembre 2024]

[56] Autodesk, "Fusion 360 Overview," Autodesk, 2023.

[En línea]. Disponible en: <https://www.autodesk.com/latam/products/fusion-360/overview?term=1-YEAR&tab=subscription>. [Último acceso: Septiembre 2024]

[57] Markforged, "FDM vs SLA: ¿Cuál es mejor para tu aplicación de impresión 3D?" Markforged Blog, 2023.

[En línea]. Disponible en: <https://markforged.com/es/resources/blog/fdm-vs-sla>. [Último acceso: Septiembre 2024]

[58] 3D Systems, "FDM (Fused Deposition Modeling)," 3D Systems, 2023.

[En línea]. Disponible en: <https://www.3ds.com/es/make/service/3d-printing-service/fdm-fused-deposition-modeling>. [Último acceso: Septiembre 2024]

[59] Impresion i3D, "Servicios de impresión 3D con tecnología FDM y sus ventajas," Impresion i3D, 2023. . [Último acceso: Septiembre 2024]

[En línea].Disponible en:<https://impresioni3d.com/servicios-de-impresion-3d-con-tecnologia-fdm-sus-ventajas/>. [Último acceso: Septiembre 2024]

[60] Bambu Lab, "P1S 3D Printer," Bambu Lab Store, 2023.

[En línea].Disponible en:<https://eu.store.bambulab.com/es-es/products/p1s?variant=49311572328796>. [Último acceso: Septiembre 2024]

[61] Power Profiler Kit II, "Power Profiler Kit II brings greater insight to wireless power consumption", 2021

[En línea].Disponible:<https://blog.nordicsemi.com/getconnected/power-profiler-kit-ii-brings-greater-insight-to-wireless-power-consumption>. [Último acceso: Septiembre 2024]

[62] AGE Electrónica, "RS232-TTL CONVERTER MODULE," *AGE Electrónica*, 2023.

[En línea]. Disponible en: <https://www.ageelectronica.com/detalle.php?p=RS232-TTL-CONVERTER-MODUL>. [Último acceso: Septiembre 2024]

[63] Antel, "Cobertura Móvil y Datos," *Antel*, 2023.

[En línea]. Disponible en: <https://www.antel.com.uy/empresas/movil-y-datos/cobertura>.

Anexos

Anexo 1: Gestión de Proyecto

Para la correcta planificación y gestión del proyecto se utilizaron diversas técnicas y aplicaciones.

Por un lado se utilizó un diagrama de Gantt para registrar y planificar cronológicamente los hitos que debían cumplirse, esto resultó de mucha ayuda, ya que surgieron cambios debido a imprevistos (importaciones, problemas personales, etc), falta de recursos, que provocaron un desvío del Gantt inicial y la herramienta permitió reprogramar las tareas de forma sencilla sin perder de vista los hitos preestablecidos.

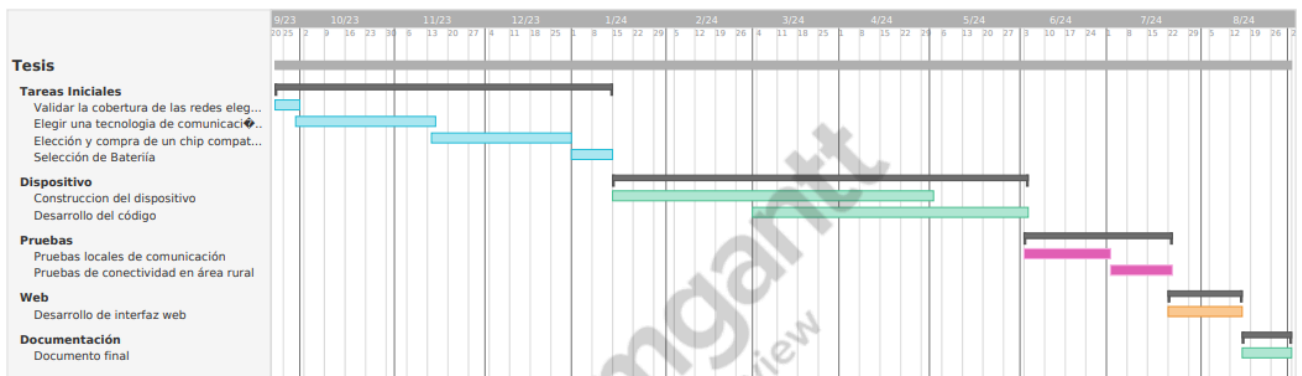


Figura 13.1: Plan de trabajo hasta el 11/03

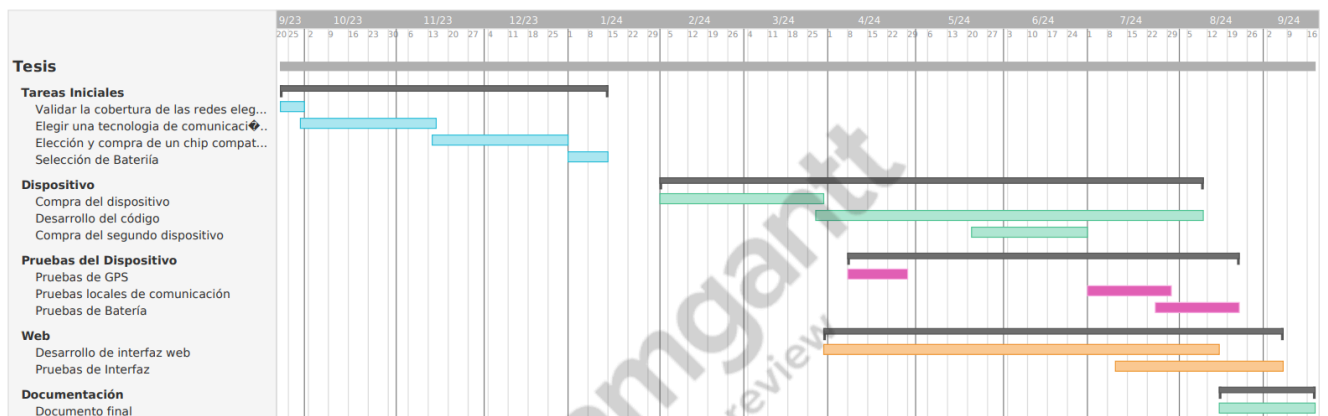


Figura 13.2: Plan de trabajo final

Al observar los diagramas de Gantt presentados en las Figuras 13.1 y 13.2, se pueden destacar las diferencias en la planificación y ejecución del proyecto a lo largo del tiempo. En la Figura 13.1, que representa el plan de trabajo hasta el 11/03, se delinearon las tareas

iniciales como la validación de la cobertura de redes, la selección de tecnología de comunicación y la compra del chip compatible. Estas tareas se planificaron en paralelo para optimizar el uso del tiempo y acelerar el proceso de inicio del proyecto.

A medida que el proyecto avanzó, surgieron situaciones inesperadas que requirieron ajustes en la planificación. Un ejemplo notable fue el fallo del módulo LoRa en el primer dispositivo, lo que llevó a la necesidad de adquirir un segundo dispositivo. Esta situación imprevista, aunque esencial para garantizar la continuidad del proyecto, causó un retraso en el cronograma, ya que cada envío y proceso de adquisición implicaba una demora considerable. Este hecho se refleja en la Figura 13.2, donde se observa la extensión de las tareas relacionadas con las pruebas y la construcción del dispositivo.

En esta segunda fase del cronograma, se enfocó más en etapas críticas como la integración de componentes y la validación del nuevo dispositivo, que operaba en una frecuencia diferente y utilizaba un módulo LoRa distinto. Las tareas relacionadas con las pruebas de GPS, comunicación y batería fueron redistribuidas para asegurar que el nuevo equipo cumpliera con los requisitos del proyecto.

Estas diferencias en los diagramas de Gantt no solo indican la evolución y adaptación del proyecto ante desafíos imprevistos, sino también la importancia de la gestión flexible del tiempo para mitigar riesgos y cumplir con los objetivos a tiempo.

A la hora de la programación y el desarrollo se utilizaron los repositorios, en particular se utilizó Github, el cual es un sistema de control, que permite comparar el código de un archivo para ver las diferencias entre las versiones, restaurar versiones antiguas si algo sale mal, y fusionar los cambios de distintas versiones.

Proyecto-Final-Electronica Private Unwatch 1 Fork 0 Star 0

main 1 Branch 0 Tags Add file Code **About**

octarevetria up 6aa7375 · now 11 Commits

Prueba1_Bateria_Sender	nueva version	last week
Prueba2_Bateria_Sender	nueva version	last week
Prueba_Distancia_Sender	nueva version	2 days ago
Receptor_v2	nueva version	last week
Sender_optimizado_v2	nueva version	2 days ago
Sender_optimizado_v3	Version de la ultima semana	1 hour ago
.gitattributes	up	now
mqtt_topic_pasamanos.py	nueva version	2 days ago

README

About

Codigo para la tesis

- Activity
- 0 stars
- 1 watching
- 0 forks

Releases

No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)

Languages

C++ 96.2% Python 3.8%

Figura 13.3: Repositorio del proyecto en Github
<https://github.com/octarevetria/Proyecto-Final-Electronica>

Anexo 2: Problemas Técnicos con la Placa T-Beam y Cambio de Frecuencia de Operación

Durante la realización de pruebas con la placa T-Beam, específicamente en la configuración y uso del módulo LoRa, se encontró un problema significativo en el que el módulo no encendía correctamente, generando repetidamente el mensaje de error "Starting LoRa failed!".

```
12:39:08.362 -> configsip: 0, SPIWP:0xee
12:39:08.362 -> clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:
12:39:08.362 -> mode:DIO, clock div:1
12:39:08.362 -> load:0x3fff0030,len:1184
12:39:08.362 -> load:0x40078000,len:13260
12:39:08.362 -> load:0x40080400,len:3028
12:39:08.362 -> entry 0x400805e4
12:39:08.542 -> initBoard
12:39:08.542 -> AXP2101 PMU init succeeded, using AXP2101 PMU
12:39:08.542 -> =====
12:39:08.542 -> DC1 : + Voltage: 3300 mV
12:39:08.574 -> DC2 : - Voltage: 0900 mV
12:39:08.574 -> DC3 : - Voltage: 0900 mV
12:39:08.574 -> DC4 : - Voltage: 1100 mV
12:39:08.574 -> DC5 : - Voltage: 1200 mV
12:39:08.574 -> ALDO1: - Voltage: 1800 mV
12:39:08.574 -> ALDO2: + Voltage: 3300 mV
12:39:08.574 -> ALDO3: + Voltage: 3300 mV
12:39:08.574 -> ALDO4: - Voltage: 2900 mV
12:39:08.574 -> BLDO1: - Voltage: 1800 mV
12:39:08.574 -> BLDO2: - Voltage: 2800 mV
12:39:08.574 -> =====
12:39:08.574 -> PowerKeyPressOffTime:4 Second
12:39:10.073 -> [SX1262] Initializing ... failed, code -12
```

Figura 13.4: Monitor Serial

Este fallo ocurrió incluso después de utilizar los ejemplos y el código disponible en el repositorio oficial de GitHub de LilyGo, así como de recibir asistencia de los ingenieros de la empresa, quienes proporcionaron orientación y verificación de la configuración.

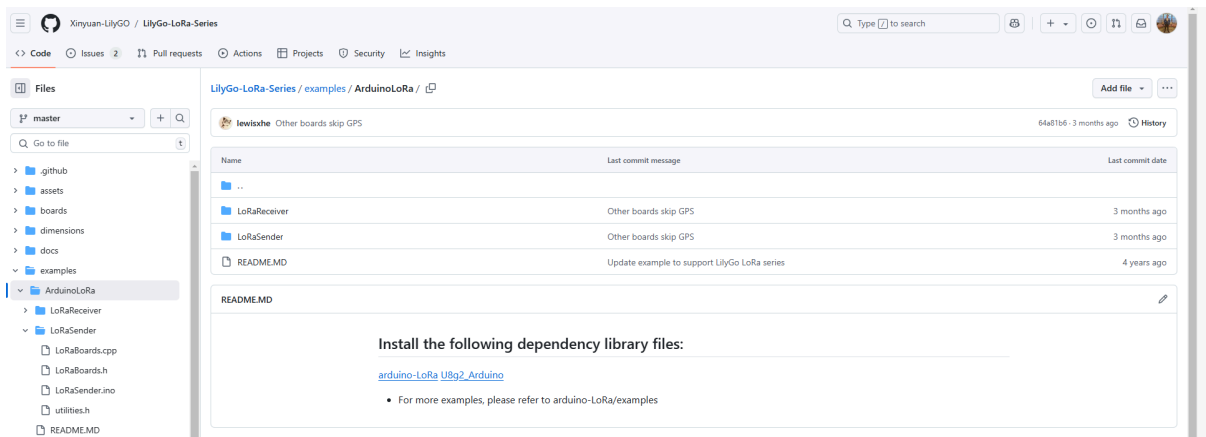


Figura 13.5: Repositorio de Github

<https://github.com/Xinyuan-LilyGO/LilyGo-LoRa-Series/tree/master/examples/ArduinoLoRa>

A pesar de los esfuerzos y múltiples pruebas realizadas, no se logró encontrar una solución al problema de inicialización del módulo. Ante esta situación, se tomó la decisión de cambiar la frecuencia de operación, que inicialmente era de 433 MHz, por una solución alternativa que minimizara los errores y redujera los tiempos de espera.

Dado que cada envío de nuevas placas podía demorar hasta un mes, y considerando el riesgo de recibir otra placa con el mismo módulo y potencialmente la misma falla, decidimos optar por una placa disponible con una frecuencia de 915 MHz. Además, se eligió la frecuencia de 915 MHz por estar homologada por la URSEC, lo que aseguraba su uso conforme a las normativas locales y permitía una operación segura y legal dentro del marco regulatorio de Uruguay.

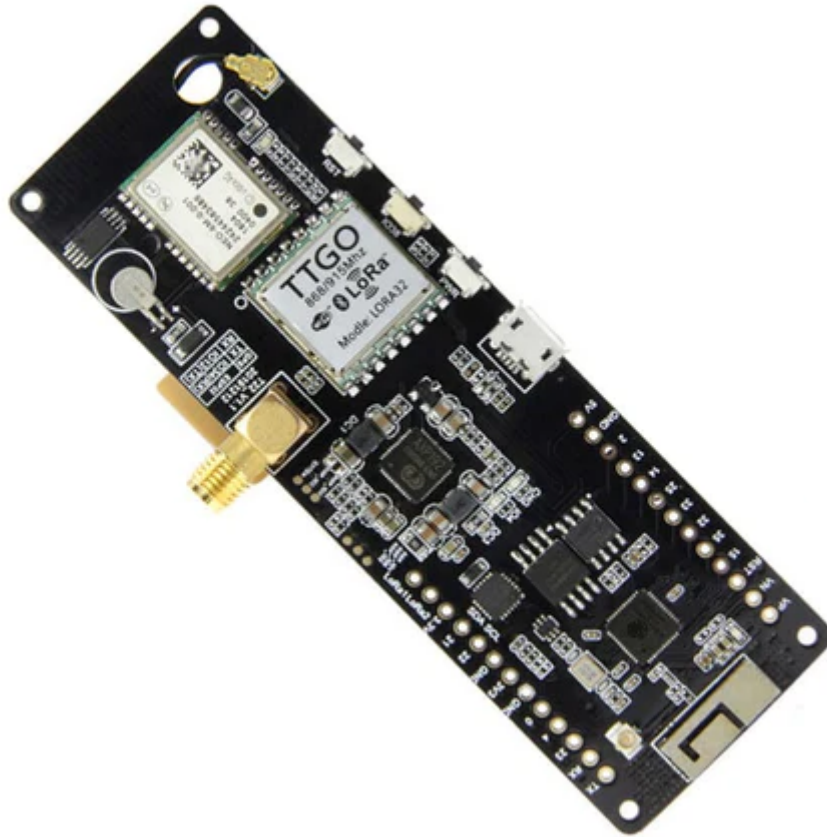


Figura 13.6: Placa con la versión de 915 MHz

Esta nueva placa incluía un módulo LoRa diferente, el modelo SX1276, y requería un código distinto para su uso. El cambio permitió continuar con las pruebas de manera eficiente y con un riesgo menor de interrupciones prolongadas.

Anexo 3: Estimación de Costo Total

Este anexo detalla la estimación de costos de producción de cada dispositivo, considerando todos los elementos necesarios para su operación, incluyendo la carcasa, la batería, y otros componentes relevantes. Se incluyen también los costos adicionales relacionados con la importación y los elementos necesarios para el receptor y gateway.

Elementos Básicos:

Descripción	Unidad	Costo (USD)
Placa Base	2	72
Importación	-	43
Carcasa	2	6
Batería	2	40
TOTAL	-	161

Elementos adicionales:

Descripción	Unidad	Costo (USD)
Caja Estanco	1	15
Gateway	1	220
TOTAL	-	235

Es importante destacar que estos costos representan la inversión necesaria para un solo activo. Si bien el costo inicial puede parecer elevado, hay que considerar que la inversión en el receptor y el gateway solo se realiza una vez. Para activos adicionales, el costo se limita principalmente a la adquisición de la placa base, la carcasa y la batería para cada dispositivo. Esto significa que, tras la compra del receptor y el gateway, el costo de escalabilidad por activo adicional se reduce considerablemente.