UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES

TITULO DEL TRABAJO DE TITULACIÓN

IMPLEMENTACIÓN DE UN SISTEMA DE ALARMA COMUNITARIA APLICANDO TECNOLOGÍA LPWAN E IOT PARA EL BARRIO 5 DE JUNIO DEL CANTÓN LA LIBERTAD.

AUTOR

RICARDO PLÚAS SEBASTIÁN

OLAVES ROSALES ANDY

PREVIO A LA OBTENCIÓN DEL GRADO ACADÉMICO EN INGENIERO EN TELECOMUNICACIONES

TUTOR

ING. MONTAÑO BLACIO MANUEL ASDRUVAL. MSc.

La Libertad, Ecuador

Año

2025

TRIBUNAL DE SUSTENTACIÓN

Ing. Ronald Rovira Jurado, PhD.
DIRECTOR DE LA CARRERA

Ing. Manuel Montaño Blacio, MSc.
DOCENTE TUTOR

Ing. Fernando Chamba Macas, Mgt.
DOCENTE ESPECIALISTA

Ing. Luis Amaya Fariño, Mgt.
DOCENTE DE ÁREA

Ing. Corina Gonzabay De La A, Mgt.
SECRETARIA

UPSE

UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES

CERTIFICACIÓN

Certifico que luego de haber dirigido científica y técnicamente el desarrollo y estructura final del trabajo, este cumple y se ajusta a los estándares académicos, razón por el cual apruebo en todas sus partes el presente trabajo de titulación que fue realizado en su totalidad por Sebastián Ricardo y Andy Olaves, como requerimiento para la obtención del título de Ingeniero en Telecomunicaciones.

La Libertad, a los 30 días del mes de junio del año 2025

TUTOR

Ing. Manuel Montaño Blacio, MSc.

UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES
DECLARACIÓN DE RESPONSABILIDAD

Nosotros, Sebastián Ricardo y Andy Olaves

DECLARAMOS QUE:

El trabajo de Titulación, IMPLEMENTACIÓN DE UN SISTEMA DE ALARMA COMUNITARIA APLICANDO TECNOLOGÍA LPWAN E IOT PARA EL BARRIO 5 DE JUNIO DEL CANTÓN LA LIBERTAD, previo a la obtención del título en Ingeniero en Telecomunicaciones, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de nuestra total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

La Libertad, a los 30 días del mes de Julio del año 2025

AUTORES

Sebastián Ricardo

Andy Olaves

UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES

AUTORIZACIÓN

Nosotros, Sebastián Ricardo y Andy Olaves

Autorizo a la Universidad Estatal Península de Santa Elena, para que haga de este trabajo de titulación o parte de él, un documento disponible para su lectura consulta y procesos de investigación, según las normas de la Institución.[1]

Cedo los derechos en línea patrimoniales de artículo profesional de alto nivel con fines de difusión pública, además apruebo la reproducción de este artículo académico dentro de las regulaciones de la Universidad, siempre y cuando esta reproducción no suponga una ganancia económica y se realice respetando mis derechos de autor

La Libertad, a los 30 días del mes de Julio del año 2025

AUTORES

Sebastián Ricardo

Andy Olaves

APROBACIÓN DE DOCENTE ESPECIALISTA

En mi calidad de docente especialista del trabajo de Integración Curricular denominado: **"Implementación de un sistema de alarma comunitaria aplicando la tecnología LPWAN e IoT en el barrio 5 de junio del cantón la libertad "**, elaborado por **Sebastián Ricardo Plúas y Andy Olaves Rosales** estudiantes de la Carrera de Telecomunicaciones, Facultad de Sistemas y Telecomunicaciones de la Universidad Estatal Península de Santa Elena, previo a la obtención del título de Ingeniería en Telecomunicaciones, me permite declarar que, tras supervisar el desarrollo y estructura final del trabajo, este cumple y se ajusta a los estándares académicos. En consecuencia, lo considero apto en todos sus aspectos y listo para la sustentación del trabajo.

Atentamente,

Ing. Fernando Chamba, Mgt

DOCENTE ESPECIALISTA

AGRADECIMIENTO

A Dios por ser un pilar fundamental en mí vida, con su gracia pude superar muchos obstáculos y esclareció pensamientos en momentos adversos y llenos de dificultad.

A mis Padres, Abuelos, Hermanos y Novia por confiar en mí, en un desafío importante y lleno de retos como lo es mi carrera universitaria, estando siempre presente en los mejores y peores momentos de mi vida.

A mi tutor de tesis Ing. Manuel Montaño, quien con sus clases despertó en mí el interés del IoT. Como tutor de tesis, su habilidad para hallar soluciones eficaces a cada problema fue clave para culminar este trabajo.

A mis amigos Heidy, Carmen, Maeba y Erick que siempre nos mantuvimos apoyando y en esos momentos de libertad quedaron grandes recuerdos y risas con juegos de cartas y demás. Como olvidar a mi amigo y compañero de Tesis Andy que siempre se mantuvo en momentos importantes.

Finalmente, a la Universidad Estatal Península de Santa Elena, mi más sincero agradecimiento por brindarme la oportunidad de formarme en un ambiente académico tan enriquecedor. Gracias por proporcionarme las herramientas necesarias para crecer tanto profesional como personalmente.

Sebastián Ismael Ricardo Plúas

A Dios por otorgarme sabiduría y fortaleza, permitiendo enfrentar con valentía los desafíos presentes a lo largo de mi trayectoria universitaria.

A mis padres por su apoyo incondicional y sacrificios que me permitieron cumplir mi sueño académico. A mis hermanos y a mi novia, por su confianza, consejos y compañía en los momentos difíciles. A mi cuñado Edgar, por creer en mis capacidades y motivarme a ser decidido.

A mis amigos y compañeros, en especial a Sebastián, con quien he colaborado desde los inicios en diversos proyectos. Valoro profundamente la confianza depositada en mí y el respaldo constante a través de cada acción y gesto durante este proceso.

A mi tutor de tesis, por su tiempo y paciencia que me guio con su experiencia con el tema de la tecnología, permitiendo culminar el trabajo previsto.

Finalmente, expreso mi profundo agradecimiento a la Universidad Estatal Península de Santa Elena por brindarme la oportunidad de formar parte de esta carrera maravillosa, que ha fortalecido mis conocimientos. Reconozco y valoro el compromiso de cada uno de los docentes, quienes con su dedicación diaria han contribuido significativamente a mi formación profesional.

Andy Saul Olaves Rosales

DEDICATORIA

A mis padres, pilares de mi vida, gracias por creer en mis sueños con su amor sin medida y enseñarme, con su ejemplo diario, que la constancia abre caminos a grandes oportunidades, las enseñanzas y educación fueron y serán vitales para este y futuros logros en mi vida. A mis hermanos, compañeros de travesía, porque sus bromas y abrazos oportunos hicieron que incluso los días más largos encontraran luz. A mi pareja, por creer en mi cuando el cansancio pesaba y recordarme que las metas compartidas se vuelven más grandes. A mis amigos, que supieron aparecer justo cuando necesitaba una palabra de aliento o un respiro de risa sincera. Y a mis mascotas Nico y Chucho que, aunque son caninos sentía que me comprendían más de lo necesario.

Sebastián Ismael Ricardo Plúas

Este logro dedico a todas las personas que estuvieron desde los primeros semestres haciendo grupo conmigo quienes a pesar tuvimos inconvenientes, siguen ahí, a mis padres por darme un hogar con pocos lujos, que dejaban de comprar unas cosas que le hacía falta y darme ese dinero para comprar cosas que me hacían falta y por apoyar en terminar de pagar mi laptop, gracias por confiar en cada etapa de mi vida.

A mis sobrinos, por darme esa alegría que por llamada me preguntaban cuando vienes a visitarlo y yo lleno de deberes decía pronto.

Y finalmente a quienes no están físicamente, pero sé que desde el cielo me están viendo y están orgullo de su nieto, dejaron una herida, pero sé que tarde o temprano nos vamos a ver.

Andy Saul Olaves Rosales

INDICE

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

RESUMEN

Este trabajo aborda el desarrollo de un sistema de alarma comunitaria basado en tecnología LoRa e Internet de las Cosas, la finalidad de este proyecto es acortar los lapsos de trámites para tomar una emergencia . Se implementó un circuito de botones de pánico, un Gateway LoRaWAN, plataformas como TTN y Node-Red para activación y desactivación de la alarma. Mediante pruebas de laboratorio, las alertas se transmitieron en menos de 5 segundos y mantuvieron estabilidad de enlace. Los resultados evidencian la viabilidad técnica y económica de la propuesta y muestran que la participación vecinal, respaldada por soluciones IoT de bajo costo, puede fortalecer la coordinación con las autoridades e intentar disuadir la delincuencia en sectores conflictivos. Se concluye que el sistema es escalable y constituye una herramienta efectiva para mejorar la seguridad ciudadana.

Palabras claves: LoRaWAN, IoT y Node-Red.

ABSTRACT

This work addresses the development of a community alarm system based on LoRa technology and the Internet of Things. The purpose of this project is to shorten the processing times for emergency response. A panic button circuit, a LoRaWAN Gateway, and platforms such as TTN and Node-Red were implemented for alarm activation and deactivation. Through laboratory tests, alerts were transmitted in less than 5 seconds and maintained link stability. The results demonstrate the technical and economic viability of the proposal and show that neighborhood participation, supported by low-cost IoT solutions, can strengthen coordination with authorities and attempt to deter crime in conflict-ridden areas. It is concluded that the system is scalable and constitutes an effective tool for improving citizen security.

Keywords: LoRaWAN, IoT and Node-Red.

INTRODUCCIÓN

En los últimos años, la provincia de Santa Elena, específicamente el cantón La Libertad y el barrio 5 de junio, ha experimentado un notable incremento en los índices de inseguridad. Los habitantes de esta zona sufren de manera recurrente robos, actos vandálicos y enfrentamientos, que generan un ambiente de temor y desconfianza. Una de las principales problemáticas identificadas es la falta de sistemas tecnológicos eficientes para reportar emergencias o situaciones sospechosas, limita la capacidad de respuesta y coordinación entre los ciudadanos y las autoridades. La comunicación oral, aún predominante, suele presentar demoras debido a los protocolos de verificación y recopilación de información, agrava la sensación de vulnerabilidad y desprotección en la comunidad.

Ante este contexto, el desarrollo tecnológico, en particular el avance de las tecnologías LPWAN y el Internet de las Cosas, ofrece una oportunidad para optimizar la gestión de alertas. El presente proyecto propone la implementación de un sistema de alarma comunitaria basado en LPWAN e IoT, con el objetivo de reducir los tiempos de respuesta ante emergencias y fortalecer coordinación entre los habitantes y las instituciones responsables de la seguridad pública. Esta solución permitirá a los ciudadanos emitir alertas, integrando la ubicación del incidente o de los eventos reportados, cuidando su integridad y la de su familia.

1 CAPÍTULO I

1.1 Identificación del problema

En la actualidad, la provincia de Santa Elena en el cantón La Libertad se observa un aumento en los índices de inseguridad. Los habitantes del barrio 5 de junio enfrentan constantemente robos, actos vandálicos y enfrentamientos, generando un ambiente de temor y desconfianza en la comunidad.

Uno de los principales problemas actuales es la falta de un sistema tecnológico eficiente para reportar situaciones sospechosas o emergencias. Debido que en el barrio se viene presentado varios incidentes y es uno de los lugares más peligrosos del año 2023[1]. La ausencia de herramientas adecuadas para una comunicación y coordinación limita la capacidad de los habitantes para obtener ayuda en casos de su vulnerabilidad y la sensación de desprotección.

Hoy en día, se observa una desconfianza entre los habitantes y las instituciones responsables de la seguridad pública. A pesar de que la comunicación oral sigue siendo el método predominante para alertar a las autoridades, este enfoque conlleva frecuentes demoras debido a los protocolos de verificación y recopilación de información.

1.2 Antecedentes

El Ecuador se encuentra sumergido en un gran problema de inseguridad dentro del país se intenta mitigar de diferentes maneras. Sin embargo, el conseguir reducir el índice delincuencial es limitado. El desarrollo y crecimiento de las tecnologías a nivel mundial es notable el situar la tecnología dentro de esta problemática puede disminuir el conflicto del sector.

En [2], se realizó un sistema de seguridad para vigilar y poder intervenir el acceso en un mercado, además de detectar robos o daños. Utiliza una Raspberry Pi 3 para el procesamiento, 6 cámaras IP ONVIF P2P en las entradas del mercado para la vigilancia, se colocó una sirena de emergencia que suena si detecta movimiento fuera del horario establecido. También notifica al guardia de seguridad mediante una llamada telefónica y envía alertas por Telegram y WhatsApp a través de una red local con FreePBX. Esto garantiza una vigilancia efectiva y alerta ante intrusiones para los usuarios y las autoridades policiales, mejorando la seguridad y protección de la comunidad, y el incremento de la confianza y tranquilidad entre comerciantes

En [3], el describe la fase de transmisión de información entre los nodos LoRa, que recolectan datos de telemetría de las luminarias. Este sistema actúa como un complemento, una vez que los datos son recopilados, el nodo se comunica con el Gateway y se enlaza a través de la plataforma TTN. Estos datos pueden ser posteriormente enviados a un servidor central y, finalmente, se pueden visualizar en un sitio web la información recopilada por los nodos[3].

En [4], el proyecto busca disminuir la delincuencia y robos en la ciudad de Cuenca mediante una plataforma que administra y monitorea alarmas residenciales. Se decodificó una alarma DSC y se formó un sistema comunitario usando el dispositivo CubeCell AB02 con tecnología LoRaWAN. Se agregaron luces, sirenas y botones de pánico móviles para disuadir delitos y enviar alertas al sistema comunitario. Se usó un servidor en AWS (ChirpStack) para la red LORAWAN y el protocolo MQTT para enviar datos a una aplicación móvil desarrollada en Flutter, que envía alertas a los usuarios.

1.3 Descripción del Proyecto

El proyecto está enfocado en el desarrollo de un sistema de alarma comunitaria sustanciado por la tecnología LPWAN con la finalidad de optimizar el aviso a las autoridades ante emergencias. Este sistema intenta disminuir la inseguridad ciudadana mediante una tecnología de bajo consumo energético y gran cobertura.

Con los avances de la tecnología y el actual el uso del Internet de las Cosas, las autoridades pueden recibir información del incidente, incorporando la ubicación exacta del usuario. Además, todos los eventos alertados a las autoridades quedan registradas, lo que facilita el seguimiento del sector e intentar mejorar la capacidad de respuesta en situaciones futuras.

Objetivos del proyecto

1.3.1 Objetivo General:

Desarrollar un sistema de alarma comunitaria con LPWAN e IoT para mejorar la seguridad y respuesta a emergencias en el Barrio 5 de junio.

1.3.2 Objetivos Específicos:

- Realizar una comparativa tecnológica de las necesidades de seguridad y las vulnerabilidades del Barrio 5 de junio, identificando áreas críticas e incidentes recurrentes.
- Investigar y seleccionar las tecnologías LPWAN adecuadas para la implementación del sistema de alarma comunitaria, considerando factores como el alcance de la red y el consumo de energía.
- Diseñar una topología de red LPWAN adaptada a las necesidades específicas de interacción entre los nodos sensores, con el objetivo de asegurar una comunicación eficiente.
- Desarrollar un entorno controlado IoT para la detección de intrusiones utilizando dispositivos como botones de pánico.
- Configurar el Gateway para la comunicación entre los nodos sensores y la nube, asegurando una transmisión de los datos.
- Integrar el sistema IoT en una plataforma de gestión centralizada de alertas, permitiendo una respuesta coordinada ante situaciones de emergencia.
- Desplegar los sensores IoT y los dispositivos de comunicación en ubicaciones estratégicas dentro del Barrio 5 de junio, asegurando una cobertura adecuada y una comunicación segura entre los nodos de la red.
- Realizar pruebas de funcionamiento y simulacros de emergencia para evaluar la eficiencia y la fiabilidad del sistema de alarma comunitaria.

1.4 Justificación

El Ecuador enfrenta desafíos significativos en términos de seguridad, afectando al bienestar de sus residentes. La implementación de un sistema de alarma comunitaria basado en LPWAN no solo contribuirá a disuadir la delincuencia, sino que también fomentará una mayor participación y solidaridad entre los vecinos, quienes podrán colaborar activamente en la protección mutua y la respuesta ante emergencias.

La seguridad ciudadana es un tema de alta importancia en la época actual, ya que la actividad delictiva se ha incrementado en varios países. Los sistemas de alarma comunitaria convencional exhiben deficiencias, como la exigencia de cables y la dependencia energética, lo que puede influir su utilidad.

En este entorno, el IoT se muestra como una solución original. Esta tecnología posibilita el enlace de dispositivos físicos a Internet esto logra la recopilación y la transferencia de datos de manera simultánea. Gracias a su capacidad de cobertura y su ahorro energético, los dispositivos IoT mejoran la atención y la comunicación breve entre el usuario y las autoridades correspondientes, mejorando los tiempos de respuesta en circunstancias de emergencia.

La inversión en un sistema de alarma comunitaria resulta conveniente a largo plazo, al compartir los costos de implementación y mantenimiento entre los miembros de la comunidad, hace más accesible la inversión en seguridad de todos. Esta distribución equitativa de los recursos económicos garantiza que ningún individuo asuma una carga económica excesiva y promueve la participación colectiva en la protección del vecindario.

Un sistema de alarma utilizando la tecnología LPWAN e IoT fortalece la seguridad y la confianza entre los miembros de la comunidad al proporcionar una respuesta en una emergencia. Ofrece una cobertura amplia y garantiza una comunicación segura y confiable. Esta iniciativa promueve la colaboración y la solidaridad entre los vecinos, involucra la participación de la comunidad en la protección del barrio.

1.5 Alcance del Proyecto

El sistema de alarma comunitaria tiene como objetivo reducir los tiempos de alerta ante emergencias mediante una red de comunicación eficiente, permitiendo la notificación inmediata a las autoridades y a la comunidad. Para lograrlo, se implementará tecnología LoRa debido a su bajo consumo energético y a su amplia cobertura en entornos urbanos.

El sistema estará compuesto por un Gateway LoRa, que funcionará como enlace principal para la transmisión de datos, y botones de pánico inalámbricos que, al ser activados, enviarán una señal para disparar una alarma sonora en la comunidad y notificar de inmediato a los organismos de respuesta. Cada dispositivo estará previamente registrado con su ubicación fija, de modo que, al activarse, la plataforma central mostrará en tiempo real un punto en un mapa interactivo indicando exactamente dónde se originó la alerta. Así, tanto los vecinos como las autoridades podrán visualizar de forma instantánea el lugar del incidente, optimizando la coordinación y la rápida movilización de recursos durante la atención de la emergencia.

1.6 Marco contextual

Este proyecto se centra en mitigar las relaciones de los problemas sociales de la provincia de Santa Elena. Para mejorar las condiciones sociales es un componente fundamental que debe ser gestionado de manera eficiente para evitar problemas en la comunidad. Para lograr una intervención efectiva es crucial realizar un diagnóstico de las situaciones que se presenta actualmente y comprender las expectativas de la población.

En el primer capítulo de esta tesis se establece el contexto necesario de la justificación y selección de la tecnología, en esto incluye un análisis de las situaciones como la infraestructura de la tecnología que persiste en la provincia de Santa Elena para notificar a las autoridades. Además, se evaluaron las políticas y programas clásicos de los gobiernos en tema de Seguridad Social, los sistemas tradicionales ventajas y desventajas con este análisis ayuda a entender los aspectos que han sido utilizados de las cuales aún se pueden mejorar. Se evaluaron diferentes opciones de tecnologías las cuales fueron viables debido a los costos y los beneficios para menor los costos de la comunidad.

2 CAPÍTULO II

2.1 Marco Conceptual

El marco conceptual de este proyecto abarca diversos temas fundamentales para la implementación de un sistema de alarma comunitaria aplicando tecnología LPWAN e IoT. Se explorará la problemática de la inseguridad y la necesidad de sistemas de respuesta eficientes. Además, se abordarán las diferentes tecnologías LPWAN, como LoRa y LoRaWAN, y su aplicación en soluciones de seguridad ciudadana, incluyendo la comunicación de largo alcance y el bajo consumo energético. Finalmente, se analizarán los componentes tanto físicos como lógicos del sistema, tales como el Gateway LoRa, los botones de pánico, y las plataformas como The Things Network, Node-RED y MQTT, determinando su viabilidad para mejorar la seguridad y la respuesta a emergencias en la comunidad. Este marco conceptual proporciona una base teórica y técnica esencial para el desarrollo y la implementación del proyecto.

2.2 Marco Teórico

2.2.1 IoT: En la seguridad de la ciudadanía.

El avance tecnológico ha permitido el desarrollo del internet de las cosas en donde diversos equipos como electrodomésticos y sensores se conectan de forma inalámbricas para realizar diversas funciones. El IoT ofrece una gran ventaja que el ser humano realice en menor tiempo actividades, la incorporación de la tecnología realiza con menos esfuerzo [18].

2.2.2 La seguridad de los dispositivos IoT

Los dispositivos que están conectados a internet pueden ser vulnerables a los ataques cibernéticos, se exponen a la privacidad y seguridad de los usuarios. La Internet

Society publicó información sobre la seguridad de la tecnología donde destaca la importancia de adaptar mejores estándares para fortaleces los datos de los dispositivos. Los elementos que se encuentran en el campo permiten obtener la información de los dispositivos conectados, interruptores, sensores, antenas y actuadores, esto se encuentra presente en modelo OSI en la capa de física, permitiendo conectar al Gateway, para la recopilación de los datos[19].

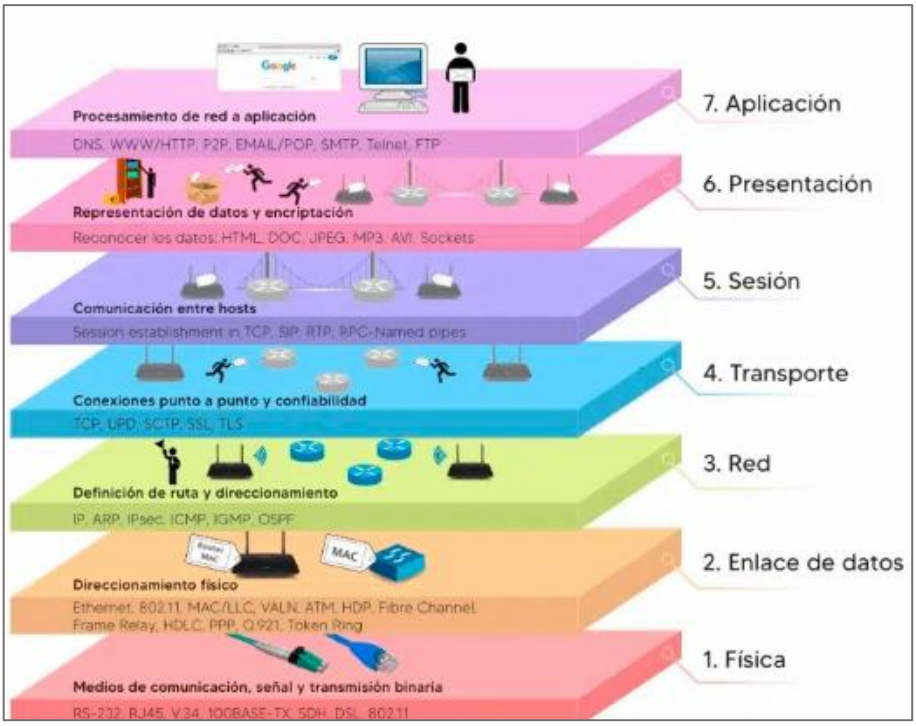


Figura 1 - Capas de comunicación OSI en IoT [20].

2.2.3 Las contraseñas predeterminadas en aplicaciones con IoT

Muchas de las personas que utilizan diversas aplicaciones, colocan el usuario y contraseña con su mismo nombre que es fácil de hackear, es muy esencial cambiarla para prevenir los ataques masivos. Una contraseña segura mínimo tiene que ser de 12 caracteres que incluya mayúscula. Un administrador de contraseñas puede facilitar el registro de sus credenciales con el inicio de sesión[20].

2.2.4 Las aplicaciones IoT en la seguridad ciudadana

Las aplicaciones se comenzaron a utilizar en el 2020 en la seguridad ciudadana como un sistema de alarma comunitaria con el uso de plataformas que permiten a las fuerzas públicas trabajar en conjunto para minimizar la delincuencia. En Guatemala la utilizaron datos de la Policía Nacional Civil y el de las instituciones Nacionales de estadísticas para analizar las situaciones de violencia e inseguridad[21].

En la figura 4 se observa que dispositivos y objetos se puede conectar al internet de las cosas y controlarlo mediante un celular o Tablet, observando el comportamiento en las decisiones que toma su configuración en aplicaciones móviles al igual que la industria realizando el respectivo proceso.

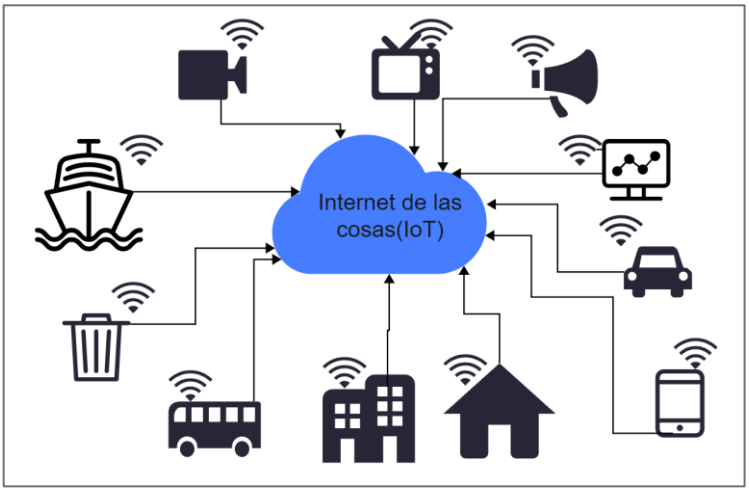


Figura 2 - Dispositivos conectados u objetos que pueden trabajar con IoT.

2.2.5 Arquitectura IoT

En la arquitectura se compone de varias capas que facilitan la conexión, y la gestión de la comunicación de los dispositivos inteligentes y sensores[22], [23].

Capa de dispositivos: Sensores para recopilar la información de los datos en su entorno, los actuadores para ejecutar la acción y los dispositivos móviles junto con el Gateway que estarán conectados con la red.

Capa de conectividad: Las tecnologías, como la red WIFI y LPWAN, que estarán conectados con el Gateway para la gestión de la comunicación con la nube. Aquí se incluyen los protocolos de comunicación con MQTT que facilita la transmisión eficiente de los datos entre los dispositivos y la nube.

Capa de procesamiento de datos: Existen 2 proceso Edge Computing en el procesamiento es el más cercano a la fuente. El Fog Computing optimiza el tráfico de los datos hacia la nube.

Capa de aplicación: Utiliza aplicaciones móviles, interfaz para los usuarios finales y Dashboards para su visualización de los datos.

Capa de seguridad: el cifrado de los datos y la autenticación y verificación de la identificación del usuario.

Capa de gestión: El monitoreo y el cuidado de los dispositivos y de la red. Control y la configuración de dispositivos.

2.2.6 Tecnologías emergentes para abordar la inseguridad en el Internet de las Cosas

Las tecnologías de área amplia y de baja potencia, LPWAN en ellas se presentan diversas tecnologías Sigfox, LoRa y NB-IoT. Estas tecnologías sirven para abordar el tema de seguridad con el internet de las cosas presentadas en tesis relacionas que se enfoca con la seguridad ciudadana utilizando diversas tecnologías.

Gateway

Es un dispositivo de alta importancia en el ámbito de las redes LoRaWAN, ayuda en la comunicación bidireccional de los dispositivos finales LoRa con la infraestructura de la red central, recibe los datos de los dispositivos IoT de baja potencia y de largo alcance utilizando una amplia gama de aplicaciones, desde monitores hasta la gestión de los entornos urbanos/rurales. En la siguiente tabla 1 se muestra varias marcas reconocidas que tienen sus versiones mejoras de la red WIFI 2G, 3G y 5G[25], [26].

Tabla 1 - Equipos Gateway LoRa

Equipo	Frecuencia	Características
Kerlink Wirnet iFemtoCell-evolution	868 MHz	Alto rendimiento, arquitectura software segura, creación de red privada
Kerlink Wirnet iBTS	868 MHz	Monitoreo remoto, Backhaul mono o dual-WAN
Heltec Wifi LoRa 32	868 MHz	Fácil de instalar, amplia compatibilidad
Sensecap M2	915 MHz	Robusto, fácil de instalar, amplia compatibilidad

2.2.7 La comunicación utilizando LoRa

Es una comunicación inalámbrica que trabaja en la capa física del modelo OSI, para proteger la información tiene 2 proceso de inicio de sesión una es con llave y encriptado AES-CTR esto se realiza para proteger la información, la estructura de comunicación con los dispositivos IoT y en la red son utilizados como nodos finales y su función es recopilar información y enviar datos, el Gateway que es el que va actuar con los puntos de acceso en conjunto con los servidores de la red que congestiona la información [27], [28], [29].

2.2.8 Activación de los dispositivos finales con LORA

Este proceso asegura que los dispositivos finales puedan comunicarse de manera segura y eficiente con la red LoRaWAN, permitiendo la trasmisión de los datos de manera bidireccional entre los dispositivos o aplicaciones que estén asociadas a la red, este proceso puede variar según el tipo de activación que se utilice[4][30].

ABP (Activación por personalizada): Este método se programa directamente en el dispositivo durante su configuración inicial. Los parámetros incluyen la dirección de la red y las claves de sesión que es otorgado por el propietario. El dispositivo utiliza este método para iniciar la comunicación sin necesidad de autenticación adicional cada vez que se conecta.

Activación inalámbrica (OTAA): En este método los dispositivos finales se activan mediante el servidor de la red LoRaWAN. Esto implica que el dispositivo trasmita su identificador único (DevEUi) y otras clases de seguridad (AppEUI, Appekey) al servidor. Con estos parámetros el dispositivo puede establecer una comunicación segura con la red LoRaWAN y trasmitir datos de manera bidireccional.

La capacidad de trabajar será muy diferente debido a las clases de los dispositivos presentes, como se muestra en la tabla 2.

Tabla 2 - Clases de la tecnología LoRaWAN.

Clase	**Descripción**	**Ventajas**	**Desventajas**
Clase A	Tiene una comunicación bidireccional.	Recepción en cada trasmisión.	Latencia es alta para recibir los datos
Clase B	Sincronización con la red con horarios específicos para recibir datos.	Mayor capacidad para la recepción de los datos.	Mayor consumo de energía.
Clase C	Siempre está listo para recibir los datos	Baja latencia para recibir los datos	Consume más energía

Las frecuencias con la que se utiliza LoRaWAN va a depender en que región se encuentra debido puede ocasionar problemas legales por el incumplimiento de la frecuencia debido a la interferencia. En la tabla 3 se muestra las frecuencias y la potencian máxima de cada una de ellas.

Tabla 3 - Frecuencias de cada región

Frecuencia	**Región**	**Potencia máxima**
868 MHz	Europa	+14 dBm
903-923 MHz	América del sur	+30 dBm
433 MHz	China	+20dBm
902 MHz	Estados unidos	+30dBm

2.2.9 Tecnologías inalámbricas usadas en el IoT

LoRa

Es una tecnología de capa física que puede ser utilizada con espectro sin licencia, es de un protocolo abierto con una comunicación apropiada para realizar diversos proyectos en la industria la asociación LoRa Alliance, está encargada de añadir los últimos modelos de IoT, siendo una tecnología de baja potencia puede alcanzar hasta 15km esto depende del equipo que se vaya a utilizar en los entornos urbanos-rurales[24].

Sigfox

Es una red de comunicación inalámbrica que es utilizada para los dispositivos celulares que tiene bajo consumo, muy útil para dar soluciones con IoT. El objetivo de esta tecnología es conectar los equipos que más utilizan las personas a bajo costo y muy eficiente, permitiendo una trasmisión de pequeños intervalos, sin embargo, esta red ofrece varias alternativas que se utilizan en esta como la tecnología GSM, 3G y 4G de bajo costo, bajo consumo de energía lo que permiten que los dispositivos operen en largos periodos con baterías y sin necesidad de reemplazarlo. Además, Sigfox optimiza la cobertura utilizando un número mínimo para las estaciones base que se conectan a internet, para obtener una comunicación adecuada de Maquina a Máquina (M2M). Esto convierte una solución de conectividad de baja ancho de banda, pero con una amplia cobertura y bajo costo, ideales para diversas aplicaciones en el perímetro de las comunicaciones M2M e IoT[31].

NB-IoT

Es una tecnología LPWAN utilizada para IoT, opera en bandas licenciadas de baja potencia con un ancho de banda estrecho, es ideal para los dispositivos que requieran transferir bajos volúmenes de datos en lugares de difícil acceso. Desarrollada 3GPP por el grupo de asociaciones de telecomunicaciones, que proporciona una cobertura amplia y estable, esto permite una conexión a varios clientes. Se basa en la tecnología LTE, evitando el consumo exagerado de energía, y ofreciendo una cobertura extensa, inclusive en áreas rurales. Esto también tiene una desventaja porque la velocidad es menor de trasmisión de LTE-M, es eficiente y económica para muchas aplicaciones IoT[32].

2.2.10 Evaluación de la Eficiencia y Seguridad de Tecnologías LPWAN en Aplicaciones IoT

Las tecnologías LPWAN han destacado en el ámbito del internet de las cosas. La tecnología LORA, fue desarrollada en el 2013, utilizada por su cobertura amplia y de bajo consumo de energía, es ideal para aplicaciones como monitores ambientales en ciudades inteligentes que utilizando el cifrada AES. Sigfox, que fue lanzada en el 2010, ofrece el bajo consumo y cobertura global, es apropiada para el rastreo, sensores industriales, utiliza una cifra de autenticación. NB-IoT fue creada por la asociación del grupo de telecomunicaciones (3GPP) del año 2016, opera en bandas licenciadas, esta ofrece una excelente cobertura de alta densidad de conexiones, utilizada para medidores inteligentes o monitoreo de salud, con una seguridad robusta basada en LTE[33]. Se muestra en la tabla 4 una comparativa de las tecnologías LPWAN.

Tabla 4 - Una comparativa de las tecnologías

	NB- IoT	**LoRaWAN**	**Sigfox**
Asociación o empresa	3GPP	Lora Alliance	Empresa Sigfox
Banda de frecuencia	Bandas licenciadas	Bandas sin licencias	Bandas sin licencias
Rango de trasmisión	10km (zonas urbanas) 40km (zonas rurales)	5km (zonas urbanas) 20km (zonas rurales)	1km (zonas urbanas) 10km (zonas rurales)
Consumo de energía	Baja	Bajo	Ultra bajo
La velocidad de datos	20 kbps	50 kbps	40kbps
Banda ancha	Banda estrecha	Banda estrecha	Banda ultra-estrecha
Modulación	BPSK	CSS	QPSK

2.2.11 Topología de tecnología LPWAN

Topología de WSN

Una red de sensores inalámbricos (WSN) está configurada físicamente con los nodos y dispositivos que la componen, puede tener una configuración lógica que está ubicada en la capa 3 del modelo OSI. Esto permite la conexión de elementos como la estación base o los servidores. En una WSN la topología física defino como se van a trasmitir los datos. La topología en estrella se utiliza para determinar la forma en cómo serán trasmitidos los datos por medio de un modelo broadcast o topología punto a punto para transmisiones de tipo UNICAST en donde pueden enviar un mensaje a un receptor

especifico. Por lo tanto, las topologías mallas y árbol son utilizadas en aplicaciones que involucran grandes cantidades de sensores [34].

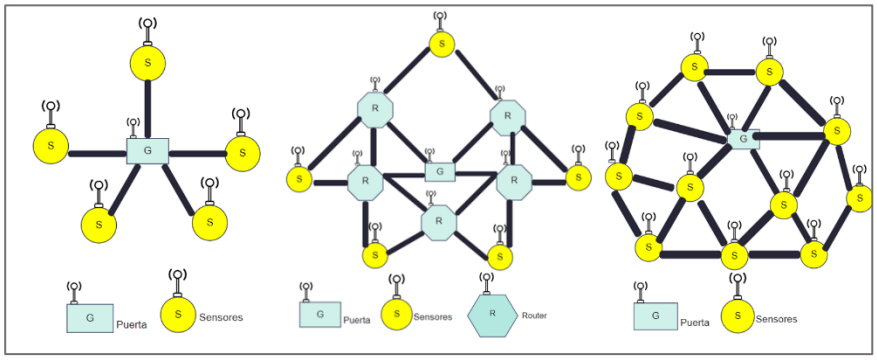


Figura 3 - Topologías más utilizadas LPWAN estrella, árbol y malla.

2.2.12 Servidores compatibles LoRaWAN

Son lo más importante dentro de la gestión de los nodos que están conectados con el Gateway, permitiendo activar y desactivar utilizando la tecnología LoRaWAN, en la trasmisión de paquetes y la comunicación de todos los dispositivos. Existen varias organizaciones que utilizan esta tecnología en sus servidores, están revolucionado al mundo, siendo muy útil para la optimizar como, envió de datos, control de equipos a larga distancias y monitoreo en lugares remotos. Sin embargo, tiene un Chips Semtech que es utilizado para de la comunicación LoRaWAN desde la serie SX12XX para dispositivos finales, así mismo tiene una serie de SX13XX para la comunicación de Gateway. Las más utilizadas por LoRaWAN son, TTN, Loriot, Actility, Everynet y ChirpStark [4].

The Things Network (TTN)

TTN tiene una versión gratuita, dentro de una red pública que se puede monitorear, es muy fácil de utilizar, esta plataforma nos permite registrar el Gateway y añadir los dispositivos finales, también ofrece otras versiones más completas, pero tiene un valor por la suscripción, es muy utilizada a nivel mundial, tiene sus funciones como el

servidor de red y se puede registrar una organización que se puede crear dentro de TTN y permitiendo el monitoreo del servidor de aplicaciones, para controlar los sensores finales [4].

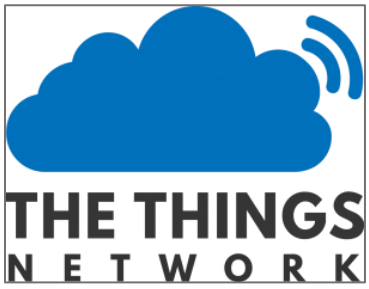


Figura 4 - Servidor The Things Network[35]

Loriot

Loriot esta plataforma permite gestionar los dispositivos IoT, tiene una versión gratuita, con funciones básicas, limitado para agregar dispositivos es útil para proyectos universitarios, esta proporciona una conexión segura y escalable, útil para una variedad de aplicaciones industriales y comerciales[35].

Figura 5 - Servidor Loriot [35]

ThingPark

ThingPark fue una de la fundadora de alianza de LoRaWAN, ofrece un servidor de red para operadores de telecomunicaciones, conocido ThingPark, constituye de una infraestructura que facilita conectar Gateway de la marca Kerlink, Cisco y Multitech, utilizado para una implementación empresarial[35].

Figura 6 - Servidor Actility[35]

Everynet

Everynet está conformada por elementos de red como, Network Server, una Radio Access Network, entre otros, todo estos operados en diferentes países, ofrece una única configuración de la red LoRaWAN para operadores como MMOs, MvNO que son modelos económicos [35].

Figura 7 - Servidor Everynet[35]

ChirpStack

ChirpStack es la más robusta conformada por el software, Open Source desarrollado por Broccar, este servidor requiere cierto nivel de experiencia, su configuración no es simple tampoco la gestión de la red, aunque es utilizado por empresas tecnológicas que están conformado por expertos informáticos sofisticados en conocimiento de LoRaWAN[35].

Figura 8 - Servidor ChirpStack[35]

A continuación, se realiza una comparativa de los servidores que son utilizado

Tabla 5 - Comparativa de varios Servidores LoRaWAN

SERVIDOR	PROTOCOLOS DE COMUNICACIÓN	PROTOCOLOS DE LORAWAN	AUTENTICACIÓN
TTN	MQTT, HTTP, AWS, Azure, Google Cloud	LoRaWAN 1.0.3 / 1.1	OAuth2, TLS/SSL, API Keys
CHIRPSTACK	MQTT, Redis, PostgreSQL, Prometheus, Grafana	LoRaWAN 1.0.2 / 1.0.3 / 1.1	API Tokens, TLS/SSL
LORIOT	MQTT, HTTP, Azure, AWS	LoRaWAN 1.0.2 / 1.0.3	TLS/SSL, AES128-CTR
THINGPARK	Múltiples plataformas	LoRaWAN 1.0.2 / 1.0.3 / 1.1	TLS/SSL, VPN, SIM-based Auth
EVERYNET	Operadores móviles con nubes privadas	LoRaWAN 1.0.3 / 1.1	TLS/SSL, SIM-Based, HSM

2.2.13 Node-Red

Es una herramienta muy usada en la actualidad por las funciones que cumple, es la ideal para diversas aplicaciones en seguridad, gestión de datos e incluso en la domótica. Esta herramienta fue creada en el año 2013 por la empresa mundialmente conocida IBM, la misión principal de esta herramienta de programación visual es simplificar las conexiones al momento incluir un hardware con otros servicios [36].

Esta interfaz es de código abierto y su sencilla interacción ayuda a simplificar el proceso de conexión del hardware con dispositivos que gestiona y trabaja en su interfaz con nodos y estos son organizados por flujos en donde la conexión es simultánea. Estos nodos permiten realizar tareas desde llamadas HTTP o activar un pulsador para cumplir con una acción definida[37].

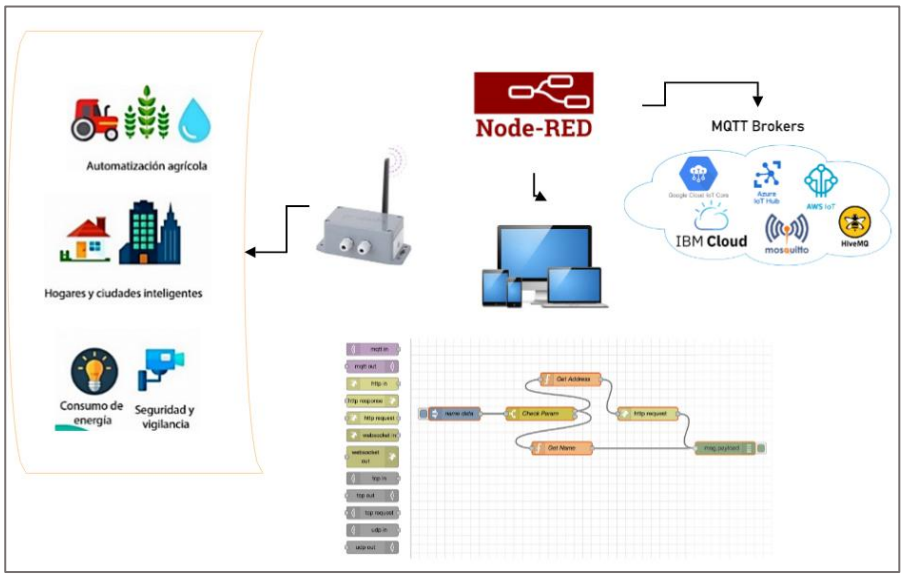


Figura 9 - Node-RED usos, componentes e interfaz

Esta herramienta llena de ventajas también es implementada dentro de la gestión y procesamiento de datos en tiempo real, este proceso consiste en pasar de lógica de programación a un flujo sencillo, la sencillez con la que cuenta Node-Red ayuda a trabajar

con casi nulos conocimientos dentro del mundo de la programación. Además de ello también se pueden implementar sistemas de marketing digital o realizar procesos que conlleven inteligencia artificial[38].

En ámbitos de seguridad esta interfaz aparece destacando las funciones con las que cuenta, actúa con sensores, enviar notificaciones, programación y demás. Las utilidades de estos van desde sensores, actuadores, servicios WEB y dispositivos IoT hasta realizar alguna acción de manera automática como apagar las luces o bloquear un acceso [39].

Para realizar estas acciones debe llevar una parte importante dentro de las comunicaciones, los protocolos son una parte fundamental para que Node-Red funcione en donde aparece HTTP para hacer peticiones a servicios web [40]. También aparece MQTT como otras de las pioneras en conexión, muy usado en los dispositivos IoT por su eficiencia, permite la suscripción y publicación enlazando dispositivos IoT[41].

2.2.14 Protocolos IoT y Bróker

Protocolos IoT

El internet de las cosas ha logrado revolucionarse en el transcurso del tiempo, esto reside en un conjunto de protocolos para realizar la comunicación. Estos protocolos actúan como un lenguaje unificador. que permite a los dispositivos inteligentes recopilar datos para poder transformar nuestro entorno.

Empezando con Wi-Fi que es casi omnipresente realizando su aparición en nuestros hogares hasta las largas comunicaciones que encontramos con LoRaWAN en campos agrícolas para fomentar una agricultura inteligente. Cada protocolo juega un papel fundamental en el crecimiento de las comunicaciones modernas.

Tabla 6 - Protocolos IoT

Protocolo	Velocidad de Datos	Rango	Frecuencia
Wi-Fi 802.11	11 Mbps - 1 Gbps	50 m	2.4 GHz - 5 GHz
Bluetooth	1 Mbps	50-150 m	2.4 GHz
Zigbee	250 kbps	10-100 m	2.4 GHz
Z-Wave	120 kbps	30 m	900 MHz
LoRaWAN	0.3 kbps - 50 kbps	5-15 km	Varias
Red Telefonica (2G, 3G, 4G, 5G)	35 kbps - 10 Mbps	50 km	900/1800/1900/ 2100 MHz

802.11 (Wi-Fi)

Wi-Fi, basado en el estándar 802.11, es ampliamente utilizado para la conectividad inalámbrica en el hogar, oficinas y espacios públicos. La capacidad de transmisión que maneja a altas velocidades lo hace adecuado para aplicaciones que

requieren un ancho de banda significativo y una transferencia de datos altas, como el streaming de video en alta definición, las videollamadas y el intercambio rápido de archivos. En el ámbito del IoT, resulta crucial para dispositivos que deben enviar grandes paquetes de información en una nube para su análisis o almacenamiento, como cámaras de seguridad inteligentes o sistemas multimedia.

Bluethoot Low Energy

Este protocolo está enfocado en las conexiones a cortas distancias y posee un bajo consumo energético. En el ámbito del IoT, Bluetooth Low Energy ha adquirido un papel importante en equipos que se comunican con teléfonos inteligentes, tabletas, etc. BLE resulta ideal para aplicaciones que incluyen sensores de proximidad, dispositivos de salud y balizas de localización en interiores. Su capacidad para establecer conexiones personales y de bajo consumo lo convierte en una opción popular para dispositivos IoT que interactúan directamente con los usuarios.

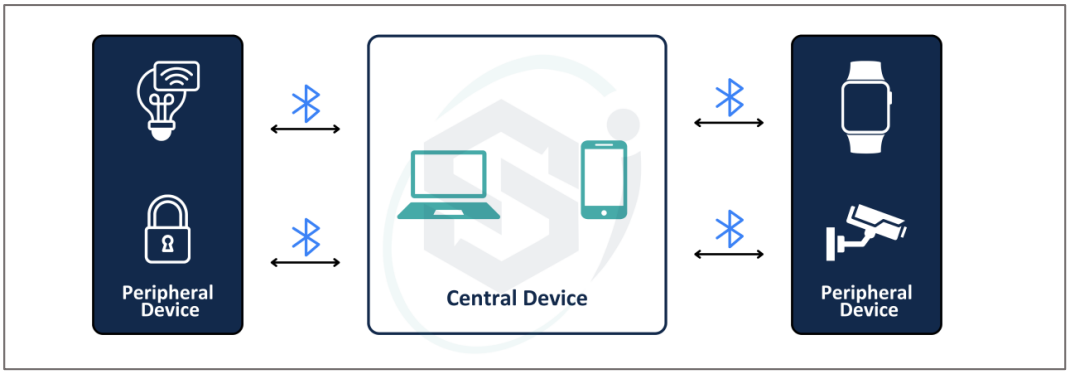


Figura 10 - Bluethoot Low Energy[42]

Zigbee y Z-Wave

Son protocolos diseñados específicamente para redes de dispositivos interconectados que se comunican entre sí en una topología de malla. Esta característica

les permite superar las limitaciones de alcance de un solo dispositivo, ya que los mensajes pueden saltar de un nodo a otro hasta llegar a su destino. Ambos protocolos son ampliamente utilizados en aplicaciones de domótica, donde dispositivos como luces, termostatos, cerraduras y sensores se comunican entre sí y con un controlador central. Su bajo consumo de energía y su capacidad para formar redes robustas los hacen ideales para entornos donde la conectividad confiable es esencial.

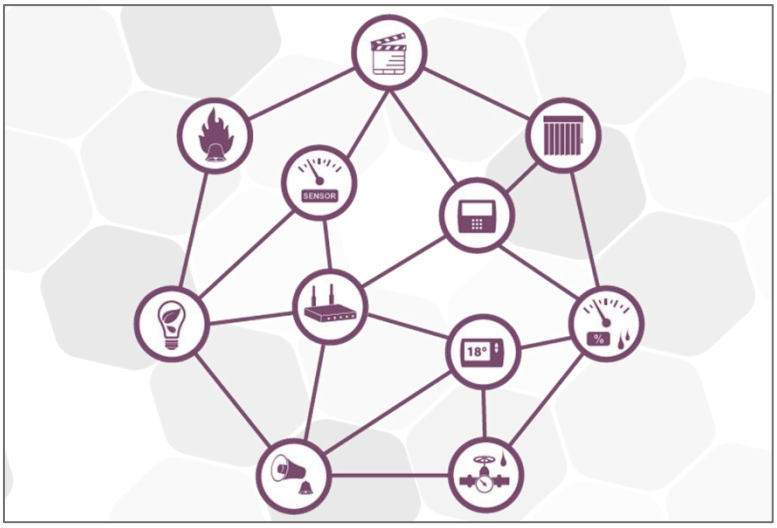


Figura 11 - Zigbee y Z-wave con la topología [42]

LoRaWAN

Este popular protocolo es el ideal de la conectividad a largas distancias y bajo consumo de energía en el mundo de IoT. Su capacidad para enviar datos a kilómetros de distancia con baterías que duran años lo hace perfecto para aplicaciones que requieren cobertura extensa y autonomía energética. LoRaWAN es la opción ideal para conectar sensores en campos agrícolas, dispositivos de seguimiento de activos en áreas remotas y sistemas de monitoreo ambiental en ciudades y zonas rurales.

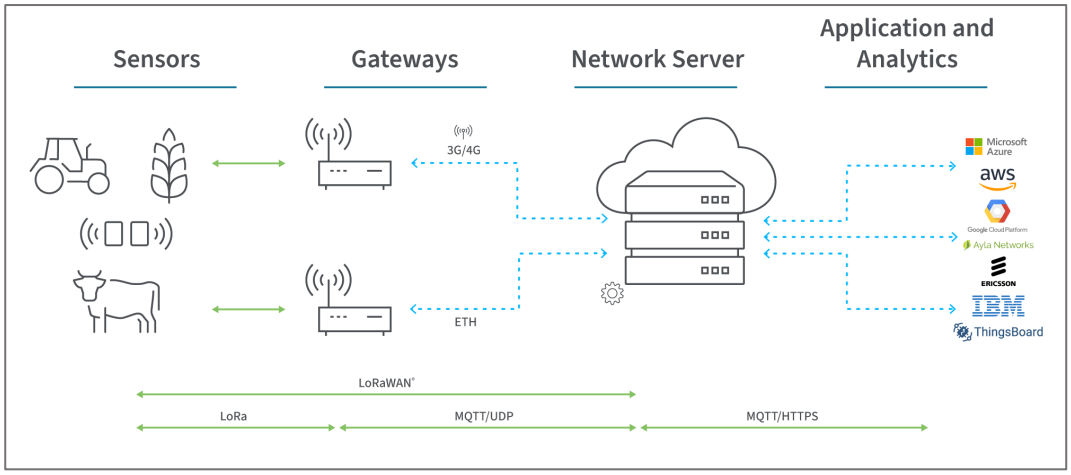


Figura 12 - Arquitectura LoRaWAN[43]

Red Celular

Las redes celulares, con su cobertura global y su capacidad para transmitir datos a alta velocidad, se han convertido en una herramienta poderosa para conectar dispositivos IoT que requieren movilidad y ancho de banda. Desde automóviles conectados hasta drones y dispositivos de seguimiento de flotas, las redes celulares permiten la comunicación en tiempo real y la transferencia de grandes cantidades de datos. Si bien ofrecen una conectividad ubicua y de alta velocidad, su mayor consumo de energía y costo en comparación con otros protocolos los limitan en aplicaciones IoT que requieren baterías de larga duración y bajo costo.

Figura 13 - Red celular[44]

2.2.15 Protocolos de comunicación

Los protocolos de comunicación son pautas que trabajan en guiar instrucciones para realizar el intercambio de información esto en la comunicación es de vital importancia para que se realice una transmisión de datos efectiva, en la actualidad existen muchos protocolos que pueden ser omitidos todo se reduce al uso que requiera este tipo de normas de comunicación[45] .

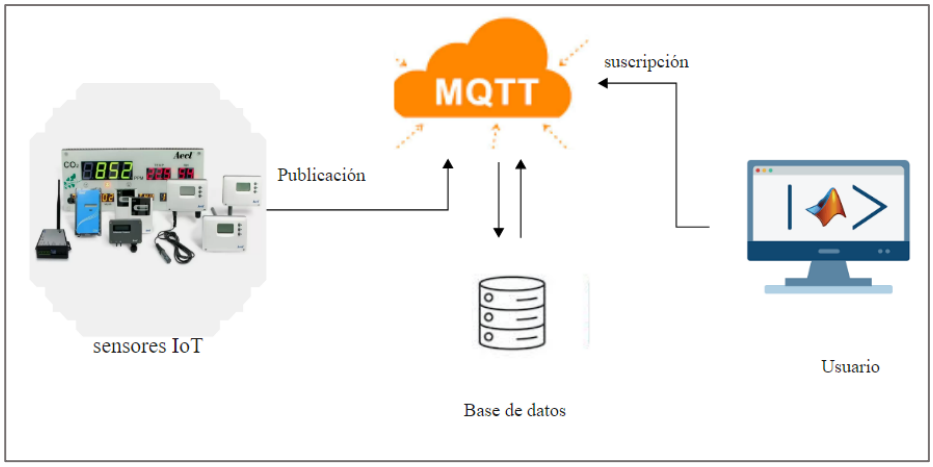


Figura 14 - Comunicación de los dispositivos IoT con el usuario

Existen casos que tienen errores dentro de la comunicación y los protocolos determinan en donde se puede encontrar el desperfecto para tener una rápida respuesta y por ende dar una solución al problema de manera eficaz. Existen diferentes tipos de protocolos los cuales son:

2.2.16 Protocolos de aplicación

Estos protocolos dentro del IoT son fundamentales por el hecho de que ellos son los encargados de que los dispositivos intercambien datos de una manera en donde no haya inconvenientes. Son clasificados según la función que cumplen, nivel que pertenecen al modelo OSI.

MQTT

Este es un protocolo que se basa en mensajería, al igual que cualquier otro protocolo tiene reglas, teniendo labores fundamentales dentro del internet de las cosas, la facilidad con la que cuentan para implementación lo hace uno de los más usados en el ámbito del IoT pues realiza la comunicación del dispositivo a la nube y de la nube al dispositivo.

La eficiencia y lo ligero de este protocolo además de los recursos que usa, son pocos, lo convierten en el pionero de las aplicaciones IoT pues este protocolo se puede usar hasta en pequeños microcontroladores, tomando como ejemplo que un mensaje puede llegar a tener la capacidad de dos bytes. Este protocolo basado en la publicación y suscripción, que incorporan topics, estos hacen que la comunicación sea más fácil de establecer determinando un tema en publicación y suscripción.

HTTP/HTTPS

Este protocolo basado en realizar una petición de datos o recursos que sean necesarios dependiendo lo requerido, esto puede llegar a ser documento HTML

. El protocolo HTTP es una base fundamental dentro de cualquier intercambio de datos dentro del internet y tiene una estructura que es cliente/servidor estos se comunican entre sí por el intercambio de mensajes. Este protocolo nació en la década de 1990 pues este se ha ido adaptando a los cambios que ha sufrido la tecnología, la web y demás trabaja dentro del protocolo TCP/IP o también puede trabajar dentro del protocolo de encriptación conocido como TLS.

En la figura 15 se puede apreciar que la estructura que compone el protocolo HTTP es cliente/servidor, pero en esta interacción aparece el proxy, este es un

intermediario que proporciona una puerta de enlace para la comunicación y transferencia de datos.

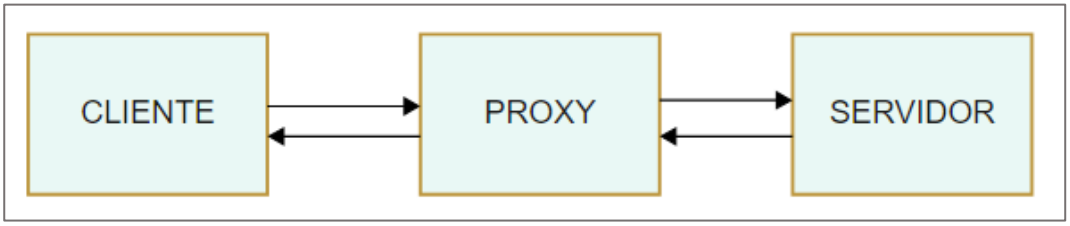


Figura 15 - Elementos que tiene la estructura del protocolo HTTP

Estos protocolos mencionados no son los únicos además de ellos existe CoAP, AMQP y DDS que a diferencia de MQTT y HTTP no son protocolos extensamente usados, pero de igual manera cumplen funciones importantes en la comunicación. AMPQ y DDS destacan por encima de CoAP, AMPQ por la infraestructura con eficiente en mensajería de sistemas en el ámbito empresarial y DDS con un rendimiento en tiempo real.

3 CAPÍTULO III

3.1 Componentes de la propuesta

3.2 Elementos de hardware

Los elementos físicos son fundamentales para que el proyecto funcione correctamente, y a continuación se describen en detalle.

3.2.1 Placa de desarrollo CubeCell Plus HTCC-AB02

En la figura 16 se presenta el Heltec CubeCell AB02 Plus, una placa que incorpora un chip LoRa optimizado para operar con redes LoRaWAN versión 1.0.2. Su función principal dentro del sistema es emitir la señal de activación. Es compatible con las bandas de frecuencia 433, 470, 868 y 915 MHz; en este proyecto se ha configurado específicamente para la banda EU915 MHz, correspondiente a la región Estados Unidos. Gracias a su bajo consumo energético y capacidad de comunicación a largas distancias, resulta ideal para activar el sistema de alarma comunitaria.

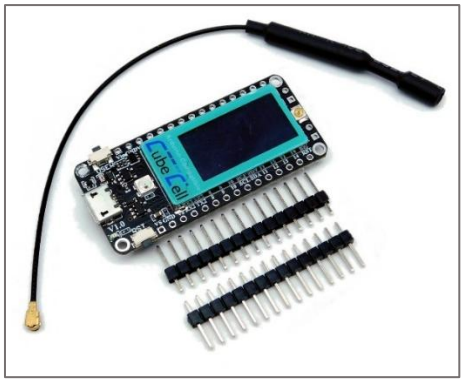


Figura 16 - Heltec CubeCell - AB02

La tabla 7 presenta una descripción detallada de las características esenciales del Heltec CubeCell AB02.

Tabla 7 - Especificaciones Técnicas del Heltec CubeCell AB02.

CARACTERÍSTICAS	VALOR
Procesador	ASR605x
Memoria RAM	16 KB
Memoria Flash	128 KB
Alimentación	2.1 ~ 3.6 V
Pines GPIO	21
Pines PWM	4
Pines ADC	3 (12 bits)
Interfaces de comunicación	I2C, UART, SPI
Interfaz LoRa	LoRaWAN 1.0.2
Frecuencia de operación	433/470/868/915 MHz

3.2.2 Pulsador eléctrico

Los pulsadores eléctricos, se representan en la figura 17, son componentes ampliamente reconocidos en el ámbito de la electrónica como en la industria de la seguridad. Tratan de pulsadores normalmente abiertos, lo que significa necesariamente ser presionados para cerrar el circuito. Este tipo de componente suelen frecuentar

empleando en sistemas de seguridad, maquinaria industrial y sistemas de control de procesos.

En consecuencia, el pulsador eléctrico ha sido seleccionado para este proyecto, ya que es el encargado de ejecutar la instrucción que activa la sirena.

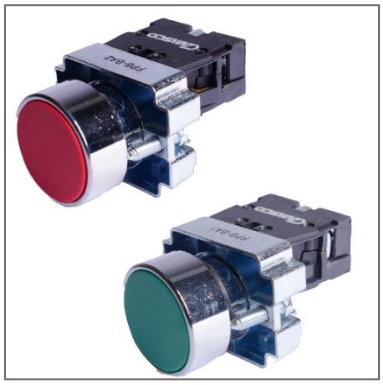


Figura 17 - Pulsador eléctrico.

3.2.3 Batería de litio 3.7 V 1500 mAh

La batería de litio cumplirá el objetivo de mantener en funcionamiento el Heltec CubeCell AB02 en situaciones donde no se disponga de un suministro eléctrico continuo, la incorporación de la batería recargable de 3,7 V y 1500 mAh, como se muestra en la figura 18. Estas clases de baterías son comunes en dispositivos portátiles, gracias a su diseño compacto.

La incorporación al sistema permite que el módulo opere de manera autónoma, esto resulta primordial en aplicaciones que requieren monitoreo constante o transmisión de datos.

Figura 18 - Batería de Litio.

3.2.4 Módulo cargador de batería TP4056

Se incorporó al sistema un módulo cargador TP4056, como se puede observar en la figura 19. Este componente permite la recarga de la batería de forma segura a través de una conexión USB tipo C, gestionando la recarga de manera automática. Al momento de recargar la batería brinda protección contra sobrecargas y descargas profundas.

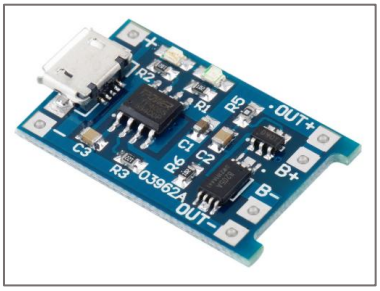


Figura 19 - Modulo cargador de baterías.

3.2.5 Gateway Sensecap M2

En la figura 20 se muestra el Gateway Sensecap M2, este equipo permite comunicaciones de largo alcance. La transmisión de información que se recopila utiliza la tecnología LoRa, mientras que la conexión a la plataforma TTN se realiza con la red LoRaWAN. El Gateway opera en un rango de frecuencia de 902-928 MHz, siendo la frecuencia base de 915 MHz.

Figura 20 - Gateway Sensecap M2

Tabla 8 - Especificaciones del Gateway Sensecap M2

Procesador	**Broadcom BCM2711, Quad-Core Cortex-A72 (ARM v8) a 1.5 GHz**
Memoria RAM	128MB
Almacenamiento	32MB
Conectividad	Wifi 2.4 GHz, Ethernet10/100 Mbps
Fuente de alimentación	12V / 2A
Protocolos soportados	LoRaWAN 1.0.2
Chip de puerta de enlace	SX1302

3.2.6 Esp8266

El Esp8266, mostrado en la figura 21, es el encargado de gestionar la señal de activación o desactivación de la sirena comunitaria mediante un algoritmo, proceso que se lleva a cabo gracias al uso de plataformas específicas que permiten dicha funcionalidad. Este módulo, además, opera con un bajo consumo energético, lo que convierte en una opción eficiente para sistemas que requieren funcionamiento continuo.

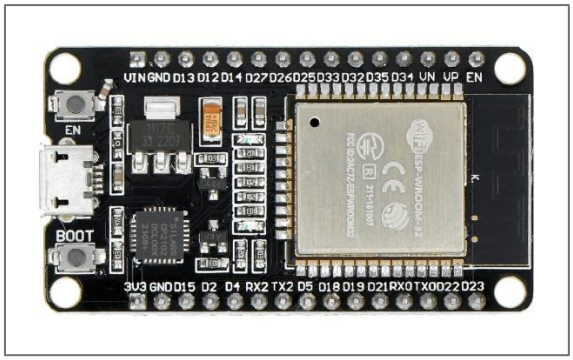


Figura 21 – Placa de desarrollo Esp8266.

De acuerdo con la información presentada en la Tabla 9, se detallan las especificaciones técnicas.

Tabla 9 - Especificaciones del Esp8266.

Parámetro	**Especificación**
CPU	RISC-V 32-bit, 160 MHz
Núcleos	1
SRAM	400 KB
ROM	384 KB

Flash	4 MB
Wi-Fi	IEEE 802.11 b/g/n (2.4 GHz)
Bluetooth	Bluetooth 5.0
UART / SPI / I2C / I2S	UART, 1 SPI, 1 I2C, 1 I2S
USB	USB 2.0
Voltaje de operación	3.0V a 3.6V

3.2.7 Sirena Comunitaria

La sirena comunitaria es un dispositivo electromecánico con 30W de potencia sonora que funciona a 12V DC. Su función principal es emitir una alerta sonora para informar a la comunidad situaciones de emergencia. La activación de esta sirena se realiza de forma remota a través de un sistema de alerta basado en la tecnología LoRa. Cuando recibe la señal de activación por los botones eléctricos, este dispositivo cobra importancia dentro del sistema de seguridad.

Figura 22 - Sirena comunitaria

3.3 Elementos de software

3.3.1 The Things Network (TTN)

La red LoRaWAN se gestionó usando TTN como servidor, facilitando el enlace entre los nodos finales y la infraestructura en la nube. El Gateway Sensecap M2 envía paquetes cifrados a TTN vía LoRa, lo que asegura la integridad de la comunicación. A continuación, un fragmento de código permite visualizar la información recibida. Asimismo, TTN soporta MQTT para encaminar estos datos a plataformas de supervisión, por ejemplo, Node-Red.

3.3.2 Arduino IDE

El Arduino IDE sirvió como plataforma de desarrollo para el Heltec AB02 PLUS, encargándose de la lógica de activación y detección de los botones eléctricos. De igual forma, el Esp8266 fue configurado y programado mediante este entorno. Con este software se compilan y cargan rutinas a medida para gestionar el encendido y apagado de los componentes del sistema.

La configuración del Cubecell y el Esp8266 se pueden ver en los anexos 1 y 2.

Las librerías empleadas, junto con sus respectivas versiones, se detallan en la Tabla 10.

Tabla 10 - Tabla de recursos usados en Arduino IDE

Librería	**Descripción**	**VERSIÓN**
CubeCell-Arduino	SDK oficial para programar CubeCell en Arduino IDE.	1.7.0
MCCI LORAWAN LMIC LIBRARY	Biblioteca LoRaWAN para Arduino, compatible con dispositivos LoRa y The Things Network.	5.0.1
IBM LMIC FRAMEWORK	Implementación de LoRaWAN para dispositivos LoRa.	1.5.1
PubSubClient	Biblioteca de cliente MQTT para Arduino, ideal para comunicación con brokers MQTT.	2.8
Esp32	Core de ESP32 para Arduino IDE, proporciona soporte para el desarrollo con ESP32	3.2.0

Recursos usados para la comunicación LoRaWAN.

Librería de las placas CubeCell

GitHub - HelTecAutomation/CubeCell-Arduino: Heltec CubeCell Series (based on ASR6501, ASR6502 chip) Arduino support.

Librerías y complementos LoRaWAN

GitHub - ricaun/SimpleLMIC: SimpleLMIC uses the MCCI LoRaWAN LMIC library behind the scene and makes Arduino-friendly

Librería de Conexión MQTT

GitHub - knolleary/pubsubclient: A client library for the Arduino Ethernet Shield that provides support for MQTT.

Librería del ESP-32

GitHub - espressif/arduino-esp32: Arduino core for the ESP32

Node-Red

Node-Red se configura como servidor destinado a enlazar hardware, su entorno gráfico muestra nodos con íconos específicos que representan funciones diversas, como la supervisión de flujos de datos. Estos flujos se serializan en JSON, lo que facilita conectar orígenes y destinos de información. Asimismo, Node-Red permite realizar análisis de datos y gestionar dispositivos de notificación. En la edición 22.14.0, resulta esencial para comunicarse con TTN, gestionando los paquetes que activan o desactivan la alarma comunitaria, por lo que se considera una herramienta adecuada para el desarrollo.

Mosquitto

Mosquitto se emplea como bróker MQTT de código abierto (licencia EPL/EDL), reconocido por su rendimiento y adaptabilidad. Opera como servidor para interconectar equipos IoT mediante el esquema de publicador-suscriptor del protocolo MQTT. En la arquitectura propuesta, Mosquitto sirve de intermediario entre Node-Red y el módulo ESP8266, garantizando el tránsito de mensajes entre estos componentes.

3.4 Diseño de la propuesta

El diseño plantea un sistema de alarma comunitaria que utiliza LoRa y plataformas IoT para desencadenar remotamente una señal audible a través de pulsadores físicos enlazados a una placa LoRa. El sistema se divide en componentes autónomos para simplificar tanto su despliegue como su mantenimiento. Cada componente asume una labor específica: captura de eventos, procesamiento, envío de datos y activación de alarmas. El procedimiento inicia en los nodos que emiten señales codificadas hacia la red;

posteriormente, una herramienta lógica como Node-Red procesa esa información y genera la acción correspondiente

3.4.1 Diseño y conexión de los componentes

La arquitectura del sistema de alarma comunitaria (ver figura 23) se distribuye en capas que funcionan como etapas de detección y respuesta, aprovechando el potencial del Internet de las cosas. Se distinguen cuatro niveles básicos: percepción, red, middleware y aplicación.

Capa de percepción: Cuando alguien presiona el botón eléctrico, el CubeCell AB02 PLUS capta la señal de inmediato y la prepara para enviarla por LoRa. Esta capa actúa como los "sentidos" del sistema, detectando el evento físico que inicia todo el proceso.

Capa de red: El Gateway Sensecap M2 recibe el paquete LoRa y lo reenvía a la infraestructura IoT mediante LPWAN. Aquí la información viaja de forma segura hacia el siguiente eslabón, como un mensajero que traslada la alerta.

Capa middleware: La plataforma TTN gestiona esos mensajes y los dirige a Node-Red. En este ambiente de procesamiento se aplican las reglas lógicas: se evalúa la situación y se decide qué respuesta corresponde. Funciona como el "cerebro" que interpreta la señal recibida.

Capa de aplicación: Un ESP32 WROOM-32 suscrito vía MQTT espera la instrucción de Node-Red; al recibirla, activa la sirena comunitaria. Es la etapa final donde la lógica se convierte en acción tangible, generando la alerta sonora ante la emergencia.

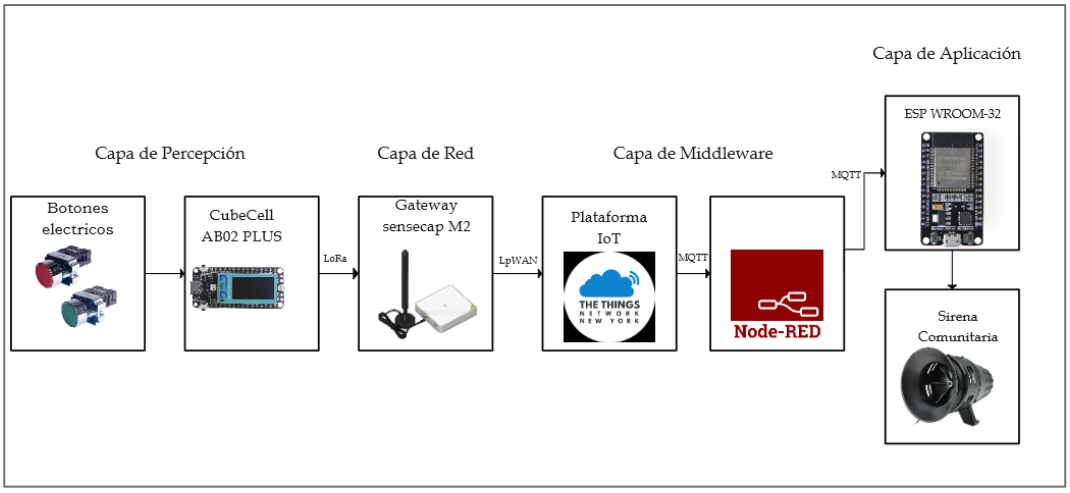


Figura 23 - Arquitectura del sistema de alarma comunitaria.

3.4.2 Diseño electrónico

El diseño eléctrico del sistema de alarma comunitaria fue desarrollado utilizando el software Fritzing, reconocido por su utilidad en la elaboración de circuitos y conexiones eléctricas. Esta herramienta facilita la planificación y el montaje eficiente del circuito, reduciendo así la posibilidad de errores durante el proceso de fabricación. A continuación, se presenta un análisis detallado del procedimiento para la creación de los componentes de modelo.

Selección de componentes

Botones eléctricos con CubeCell AB02 PLUS

Los botones eléctricos integran una batería de litio de 3,7 V, administrada por un módulo TP4056. Gracias al puerto USB-C, la batería se recarga de forma sencilla y el TP4056 se encarga de protegerla ante sobrecargas y de informar sobre su nivel de carga, de modo que el sistema mantenga siempre un voltaje estable incluso en ubicaciones remotas ver figura (24).

Las conexiones del diseño son:

- El TP4056 recarga la batería a través del USB-C y supervisa el estado de carga.
- El pin D1 queda asignado como entrada digital para vigilar eventos de seguridad ciudadana; el pin D2, como entrada digital para emergencias por incendio.

Se comprobó que ambos canales respondieran correctamente en distintos escenarios de carga de la batería. El TP4056 logró mantener la alimentación dentro de los márgenes esperados, de modo que incluso cuando la batería baja de nivel, el sistema sigue operando.

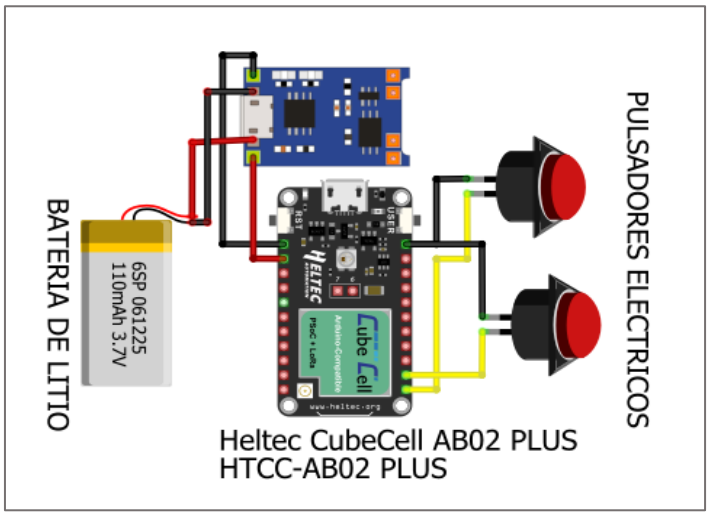


Figura 24 - Diseño de los pulsadores eléctricos con el CubeCell AB02 plus.

3.4.3 Esp8266 con sirena comunitaria

El mecanismo de la alarma comunitaria funciona sobre una arquitectura que controla la sirena de 12 V mediante un relé. Node-Red envía las órdenes que determinan cuándo activar o desactivar la alarma, lo que permite programar la sirena y asegurar una respuesta ágil en emergencias. El microcontrolador conecta el relé de la siguiente manera: la alimentación del relé se toma del pin de 5 V del sistema de control; el GND del

microcontrolador y el del módulo de relé comparten la referencia de tierra; y el pin D1 se emplea como línea de mando para enviar las señales que encienden o apagan la sirena.

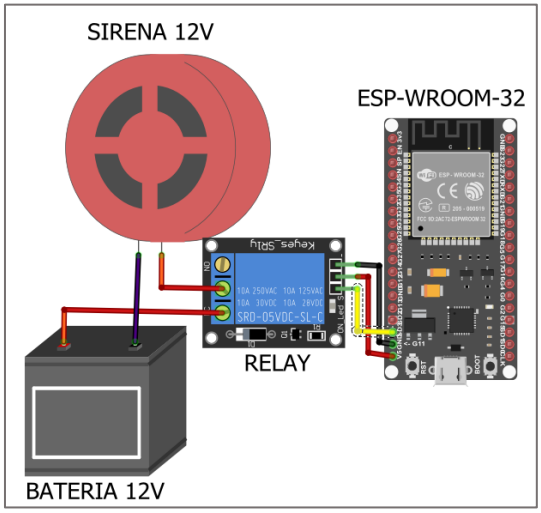


Figura 25 - Diseño de la sirena 12V.

3.5 Configuraciones del hardware y software

3.5.1 Gateway Sensecap m2

La configuración del Gateway Sensecap es un paso esencial para establecer la comunicación LoRa con los nodos finales, como se muestra en las figuras 26 a 33. Para ingresar a la interfaz de configuración del dispositivo, se deben utilizar las credenciales ubicadas en la parte posterior del equipo. El acceso inicial se realiza a través de la dirección IP predeterminada: 192.168.168.1.

Figura 26 - Credenciales de acceso e información del Gateway Sensecap M2

Dentro de la interfaz del Gateway, se debe acceder a la sección correspondiente a LoRa y, posteriormente, ubicar la opción denominada Channel Plan.

Figura 27 - Interfaz principal del Gateway.

A continuación, se debe seleccionar la región US902-928, dentro del apartado correspondiente. Seguidamente, se configura la frecuencia del plan en FSB1, canal 0. Esta configuración permite establecer una sincronización precisa con el servidor designado para la visualización del flujo de datos.

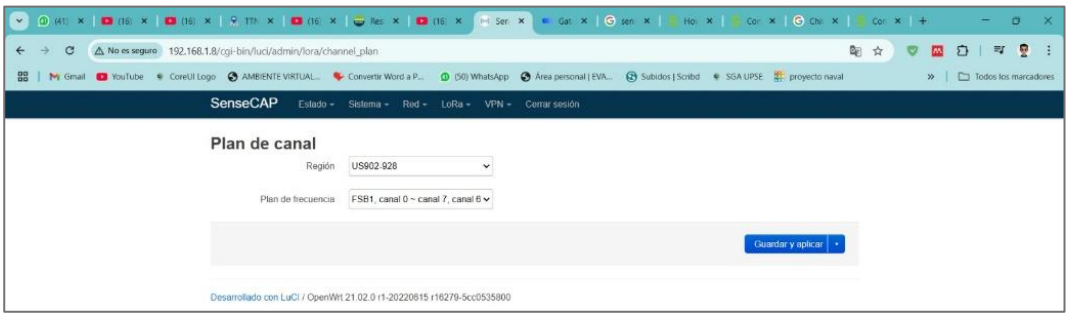


Figura 28 - Interfaz de selección del plan de canal.

Para configurar el dispositivo dentro de una red inalámbrica, se debe acceder a la opción Network (1) y posteriormente seleccionar Wireless (2). A partir de allí, se procede a vincular el equipo con una red Wi-Fi disponible.

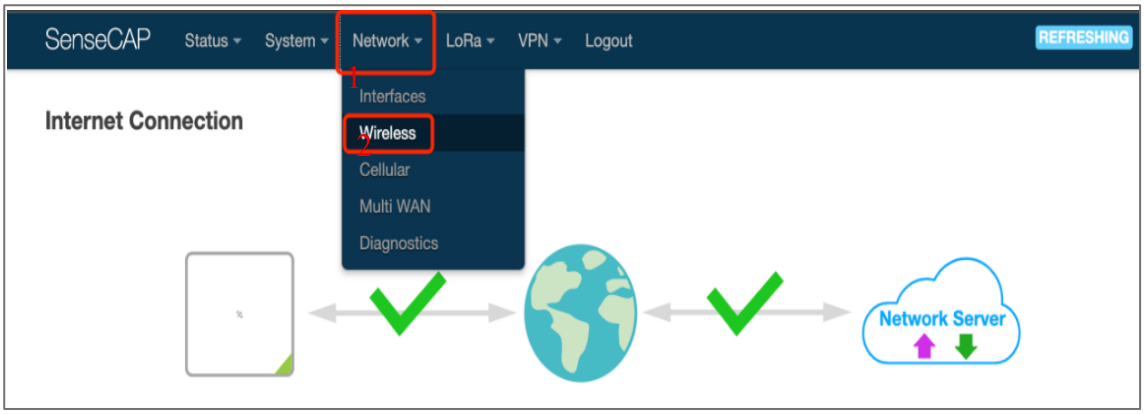


Figura 29 - Entrar a la configuración de red en el Gateway Sensecap M2.

Se realiza un escaneo de las redes disponibles haciendo clic en la pestaña Scan, tras lo cual se selecciona la red Wi-Fi a la que se desea conectar el Gateway.

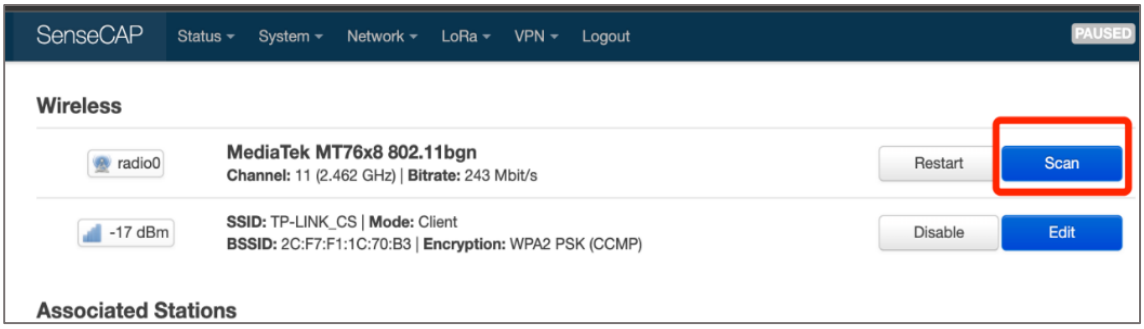


Figura 30 - Escaneo de redes inalámbricas en el Gateway.

Una vez finalizado el escaneo, se despliega un listado con las redes Wi-Fi disponibles; desde este listado, se debe seleccionar la red deseada y hacer clic en **Join Network** para establecer la conexión.

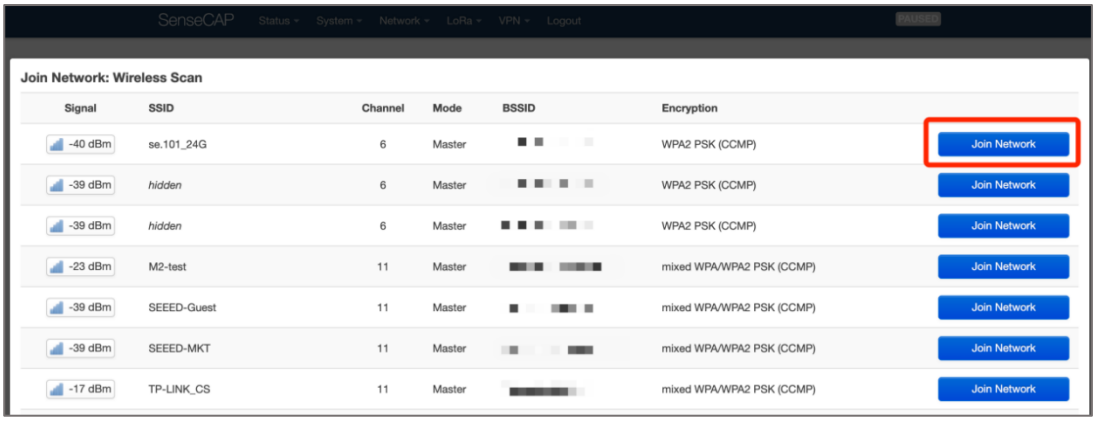


Figura 31 - Selección de red inalámbrica.

Como paso final en la configuración, es necesario ingresar las credenciales correspondientes en el campo indicado (1) y, posteriormente hacer clic en **submit** (2) para completar el proceso de conexión a la red inalámbrica.

Figura 32 - Insertar credenciales para conectarse a la red inalámbrica.

Para establecer la comunicación mediante LoRa, el Gateway permite configurar distintos modos de operación, tales como estación de base, reenvío de paquetes o servidor de red local. En este caso, se emplea el modo reenvío de paquetes el cual requiere configurar el servicio en la dirección nam1.cloud.thethings.network con el fin de asegurar la comunicación con la plataforma TTN.

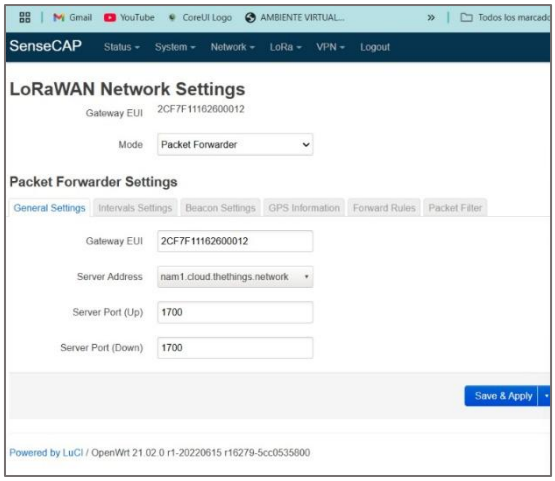


Figura 33 - Configuración LoRaWAN.

3.5.2 Configuración de la plataforma The Things Network

Para habilitar el envío de datos, es necesario crear una cuenta en la plataforma TTN. Una vez realizado el registro se debe iniciar sesión y acceder a la pestaña consola para comenzar con la configuración correspondiente.

Figura 34 - Inicio de sesión en la plataforma The Things Network.

Después de iniciar sesión, se debe seleccionar la región, tal como se muestra en la figura 35. En este caso, se elige la región North America 1.

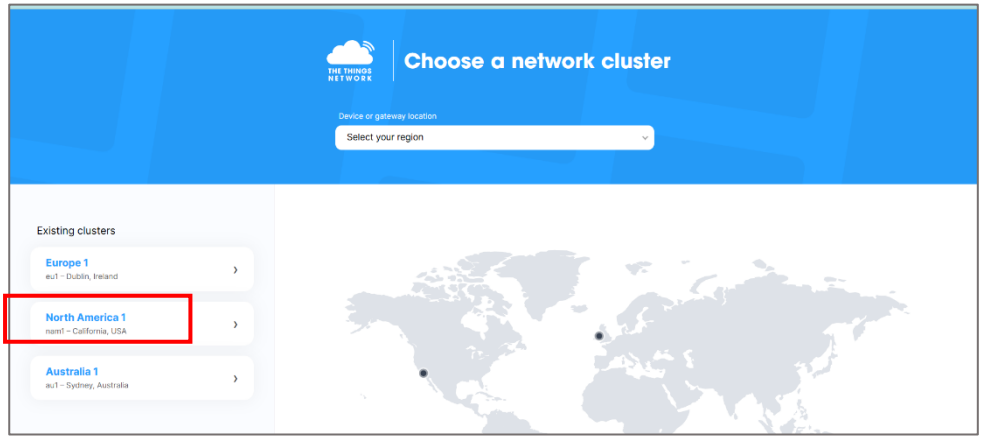


Figura 35 - Selección de región en The Things Network.

3.5.3 Configuración del Gateway en la interfaz The Things Network

Se presenta la interfaz principal de la plataforma, desde donde se debe acceder a la pestaña Gateway. Dentro de esta sección se procede a registrar el dispositivo en TTN, tal como se ilustra en la figura 36.

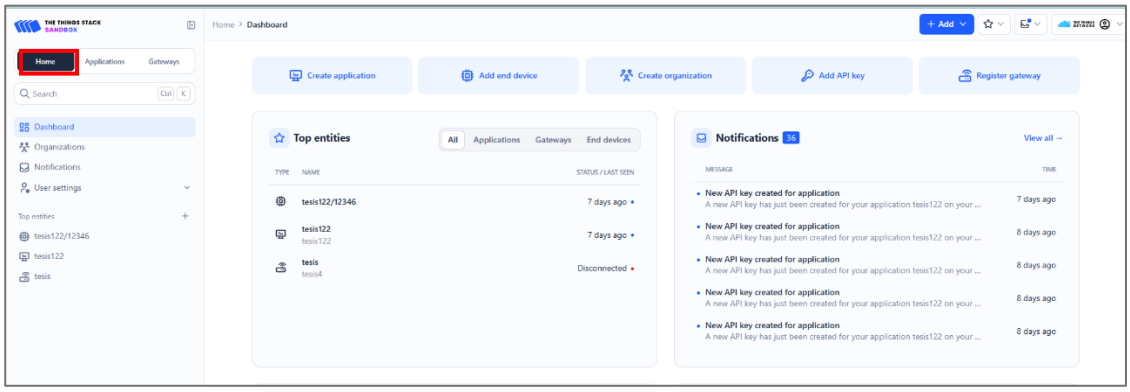


Figura 36 - Interfaz inicial de The Things Network.

Al abrir la pestaña Gateway, se procede a seleccionar la opción Register Gateway (véase figura 37).

Figura 37 - Interfaz para proceder con el registro de la puerta de enlace.

La plataforma ofrece dos métodos para realizar el registro del dispositivo: mediante el escaneo del código QR o de forma manual, tal como se observa en la figura 38.

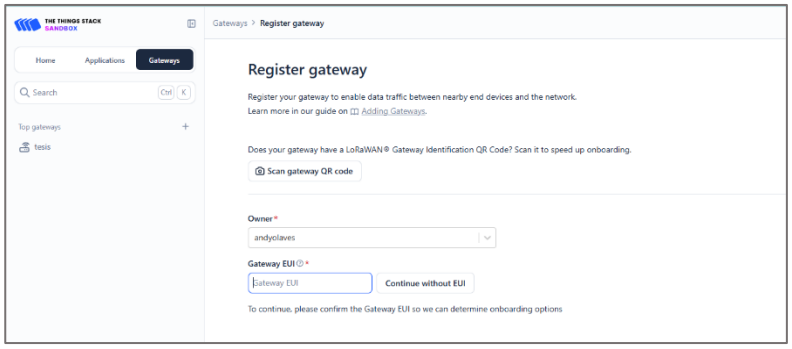


Figura 38 - Opciones de registro del Gateway.

Se deben completar todos los campos requeridos con la información correspondiente a la puerta de enlace, tal como se observa en la figura 39. Luego se hace clic en **Registrar Gateway** para finalizar.

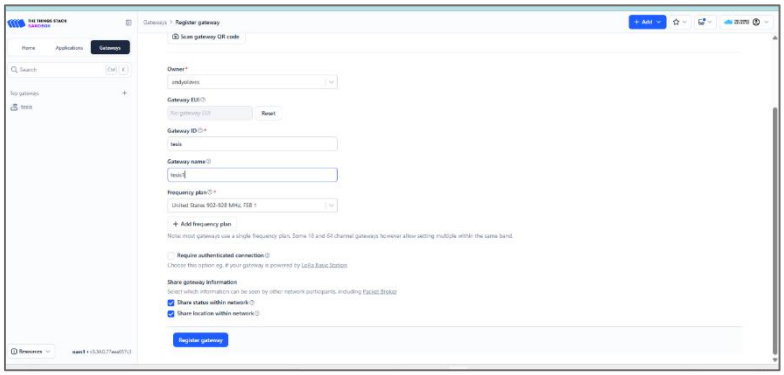


Figura 39 - Registro del Gateway.

En la plataforma, el dispositivo debe mostrarse con el ID correspondiente, la conexión establecida, como se observa en la figura 40.

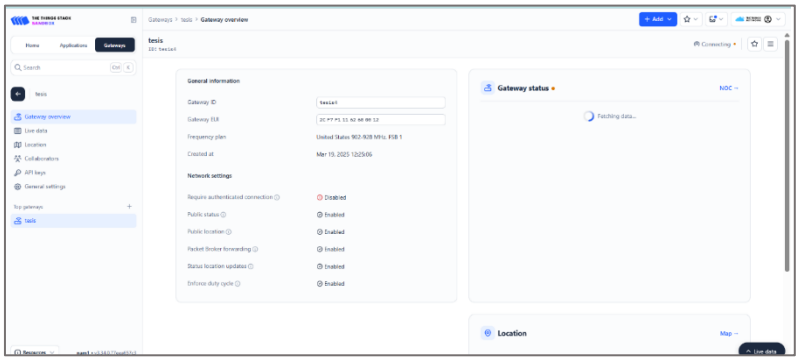


Figura 40 - Verificación del estado del Gateway dentro de la plataforma The Things Network.

3.5.4 Configuración de los nodos finales en The Things Network

Para incorporar nodos en TTN, es necesario seleccionar la opción **Applications** y luego hacer clic en **Create Application**, tal como se muestra en la figura 41.

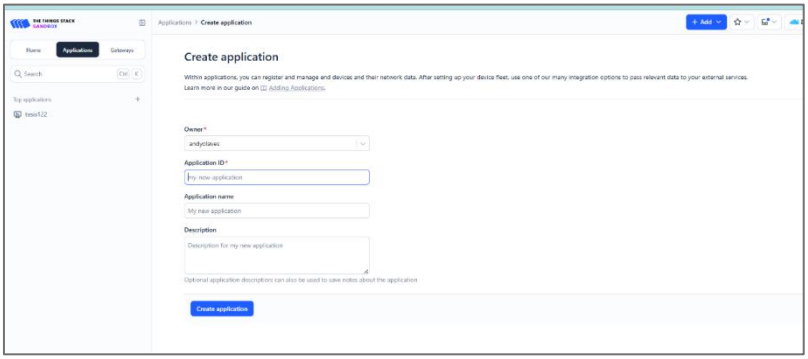


Figura 41 - Interfaz create application para agregar nodos en TTN.

Se completan los campos con la información correspondiente al módulo que se usara. Luego se hace clic en **Register End Device**, se selecciona el modelo y la clase del módulo, en este caso, HTCC-AB02 Clase A ABP, tal como se observa en la figura 42.

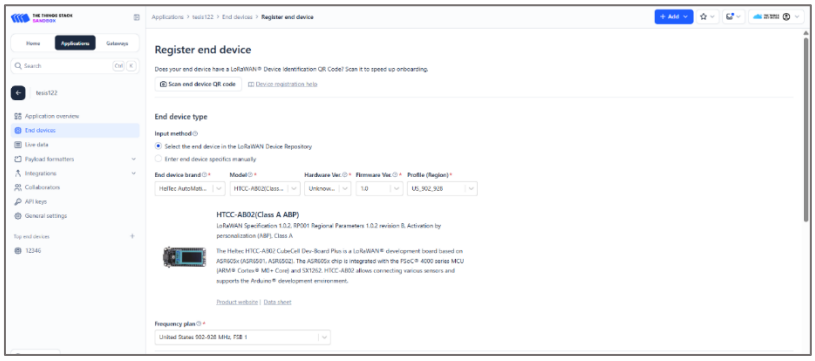


Figura 42 - Registro del nodo a The Things Network.

El dispositivo requiere credenciales que son DevEUI, device address, AppSkey y NwkSkey. Para generar estas credenciales, se debe hacer clic en **Generate** (1) y clic en **Register end device** (2), como se muestra en la figura 43.

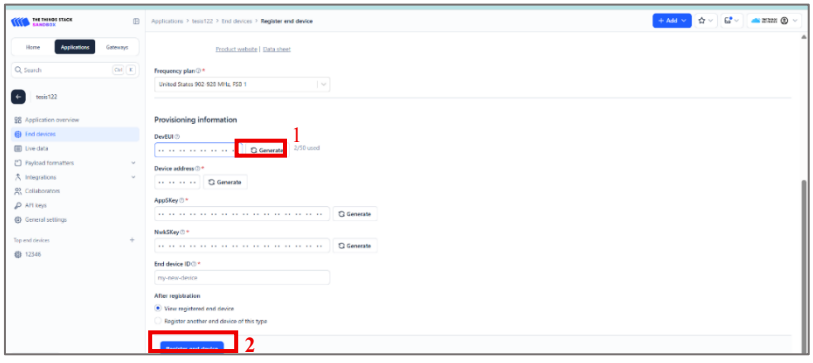


Figura 43 - Creación de las credenciales del módulo Heltec Cubecell AB02.

Este debe aparecer en estado Connected para saber si el dispositivo está correctamente sincronizado con la plataforma (ver figura 44).

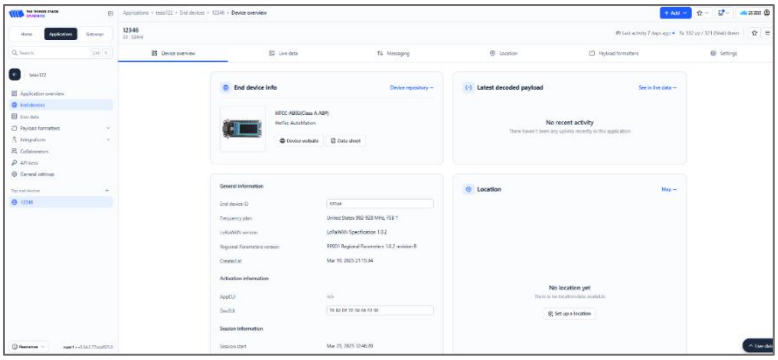


Figura 44 - Dispositivo ya configurado en la plataforma TTN.

3.5.5 Decodificación de los mensajes de The Things Network

La decodificación correcta de los mensajes mostrados en la plataforma es fundamental, ya que durante la transmisión de datos pueden presentarse errores debido a incompatibilidades.

Esta configuración se ilustra en la figura 45, donde se debe acceder al apartado Payload Formatters (1) y configurar el formato del enlace ascendente como JavaScript personalizado (2). Esta selección dependerá del tipo de datos que se estén transmitiendo, ya sea en formato ASCII o binario.

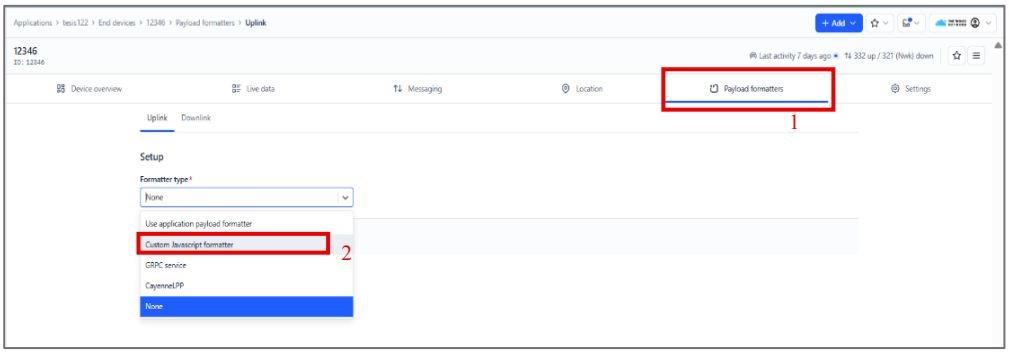


Figura 45 - Configuración de uplink en la plataforma TTN.

Una vez completada la configuración, se debe hacer clic en Guardar cambios. Posteriormente, se llevan a cabo las pruebas necesarias para verificar el funcionamiento.

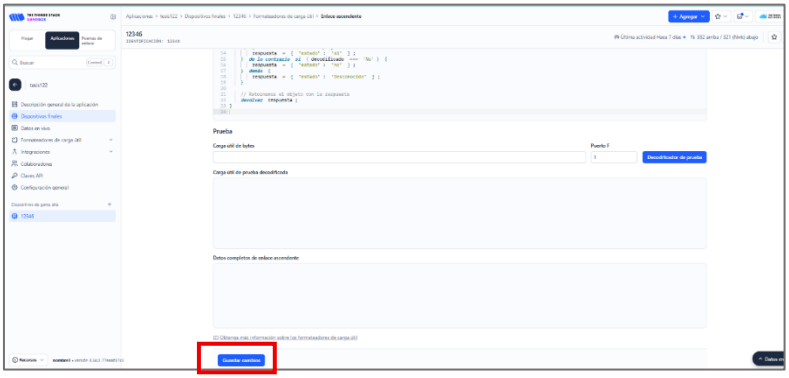


Figura 46 - Configuración de la decodificación de mensajes en TTN.

3.5.6 Configuración de Node-Red

Para ejecutar el servidor, se utiliza el Node.JS Command Prompt, donde se ingresa el comando Node-Red. Después de unos segundos, el servidor se inicializa, cargando sus componentes, como la paleta de nodos y flujos, tal como se muestra en la figura 47. Posteriormente, se accede a la interfaz desde un navegador web introduciendo la dirección IP proporcionada por el servidor.

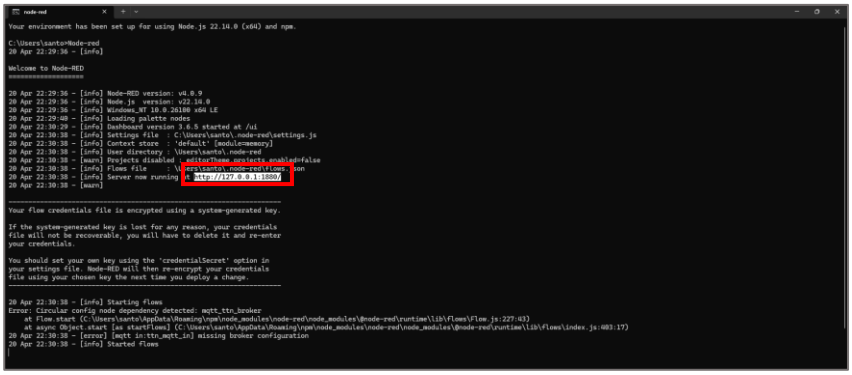


Figura 47 - Ejecución de Node Red.

Dentro de la interfaz, el siguiente paso consiste en descargar las librerías necesarias. Para ello, se debe acceder al Menú de opciones, ubicado en la parte superior derecha, como se observa en la figura 48 y posteriormente hacer clic en Administrar paleta.

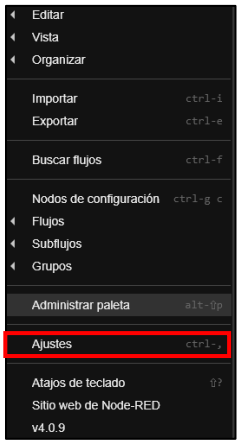


Figura 48 - Apartado de administrar paleta en Node Red.

Dentro del menú desplegado, se debe acceder a la pestaña Instalar, tal como se muestra en la figura 49. En esta sección, se buscan las librerías requeridas para proceder con su instalación.

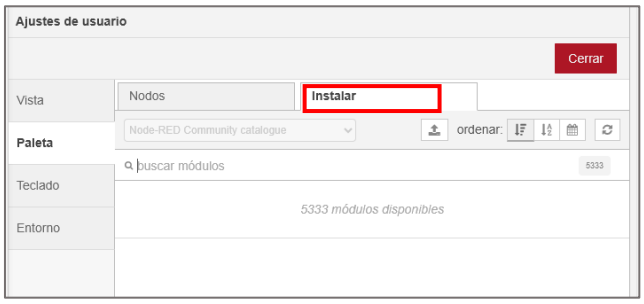


Figura 49 - Instalar librerías en Node Red.

Librerías usadas en Node-Red

NODE-RED-MQTT

NODE-RED-CONTRIB-TELEGRAMBOT

NODE-RED-DASHBOARD

NODE-RED WORLDMAP

Una vez finalizada la instalación de las librerías, se debe acceder a la paleta de nodos, como se observa en la figura 50, para localizar los nodos requeridos.

Figura 50 - Paleta de nodos.

Los Nodos para utilizar son:

MQTT IN

MQTT OUT

WORLDMAP

DEBUG

FUNCTION

TELEGRAM-SENDER

TELEGRAM-RECEIVER

Configuración de los nodos

Para llevar a cabo la activación de la alarma comunitaria, es imprescindible configurar adecuadamente los nodos ubicados en el área de trabajo.

3.5.7 Configuración de los nodos por MQTT

Nodo MQTT IN

El nodo MQTT IN se encarga de establecer la comunicación con la plataforma TTN. Al hacer doble clic sobre este nodo, se despliega una ventana, como la que se muestra en la figura 51, donde se realiza su configuración correspondiente.

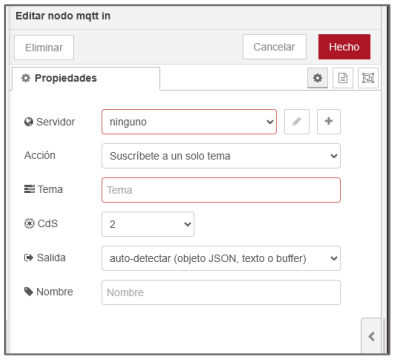


Figura 51 - Ventana de Configuración del Nodo Mqtt in.

Para garantizar una comunicación adecuada entre plataformas, es necesario configurar el tema dentro del nodo y establecer el servidor correspondiente. Para ello, se hace clic en el símbolo “+” y se define el servidor nam1.cloud.thethings.network con el

puerto 1883, que corresponde al servidor North America 1 en TTN (Véase en la figura 52).

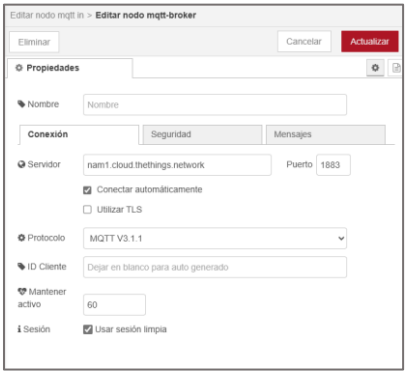


Figura 52 - Configuración del servidor y puerto dentro del nodo MQTT IN.

En la pestaña de seguridad del nodo MQTT IN, se debe configurar la información conforme a las credenciales proporcionadas por TTN, tal como se muestra en la figura 53.

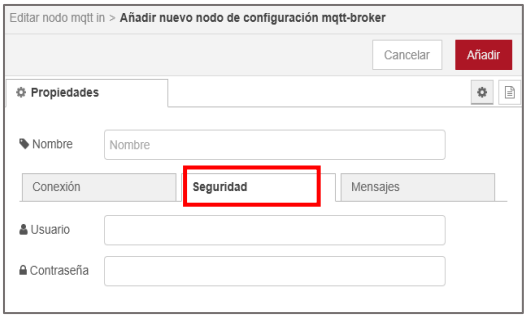


Figura 53 - Configuración de la pestaña seguridad en MQTT IN.

Para obtener las credenciales requeridas por el nodo MQTT IN, es necesario ingresar a la plataforma TTN y dentro de la pestaña Aplicaciones, localizar el protocolo MQTT, como se muestra en la figura 54.

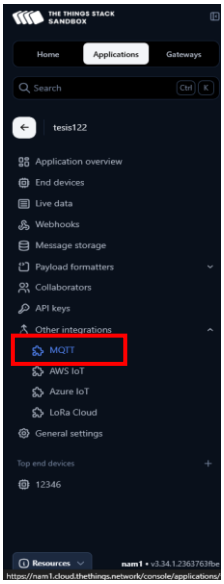


Figura 54 - MQTT en The Things Network.

Al seleccionar la pestaña MQTT, se despliega la sección mostrada en la figura 55, donde se visualizan las credenciales necesarias, incluyendo el usuario y la contraseña.

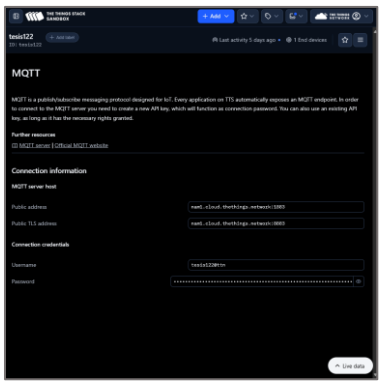


Figura 55 - Credenciales de usuario y contraseña.

Posteriormente, se completan los campos con la información obtenida de la plataforma TTN y se hace clic en Actualizar, como se muestra en la figura 56.

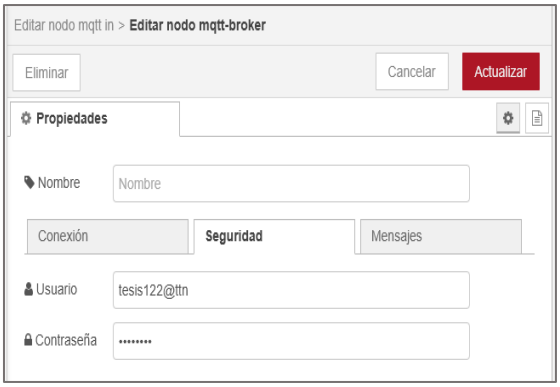


Figura 56 - Configuración de usuario y contraseña en el nodo MQTT IN.

Respecto al tema o topic, es necesario seguir las instrucciones recomendadas por TTN. En este caso, dado que se recogerán datos de subida, se selecciona el modelo de tema indicado en la figura 57.

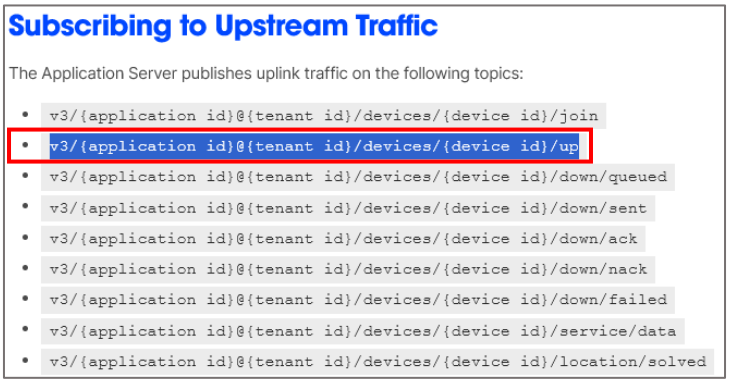


Figura 57 - Modelo de tema en The Things Network.

Una vez creado el tema, se completa el último campo requerido para el nodo MQTT IN, como se observa en la figura 58.

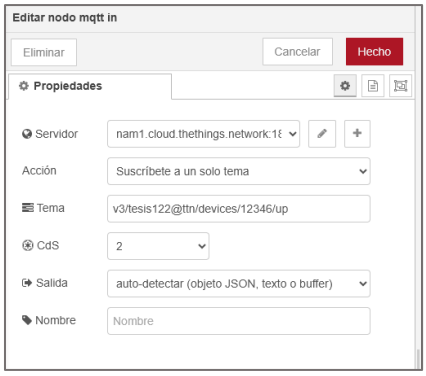


Figura 58 - Configuración final del nodo MQTT IN.

Nodo MQTT OUT

Este nodo se configura para el envío de datos hacia un microcontrolador, con el propósito de activar la sirena comunitaria.

Al hacer doble clic en el nodo MQTT OUT, se despliega la ventana mostrada en la figura 59, desde donde se realiza el proceso de configuración.

Figura 59 - Ventana de configuración del nodo MQTT OUT.

La configuración se realiza considerando el servidor que se utilizará; en este caso se emplea el servidor Mosquitto.

El servidor corresponde a la dirección IP donde se ejecuta Mosquitto, y el tema debe estar sincronizado con el microcontrolador previamente configurado, tal como se muestra en la figura 60.

Figura 60 - Configuración del nodo MQTT OUT.

3.5.8 Telegram

Chatbot BotFather

Para configurar los nodos de Telegram en Node-Red, es necesario utilizar un bot que genere las credenciales requeridas para establecer comunicación entre cualquier red IoT y Telegram. Esto facilita el intercambio de mensajes para la gestión de dispositivos.

Primero, se busca el bot llamado BotFather dentro del buscador de Telegram. Este bot permitirá realizar la interacción necesaria para vincular las plataformas Node-Red y Telegram, como se observa en la figura 61.

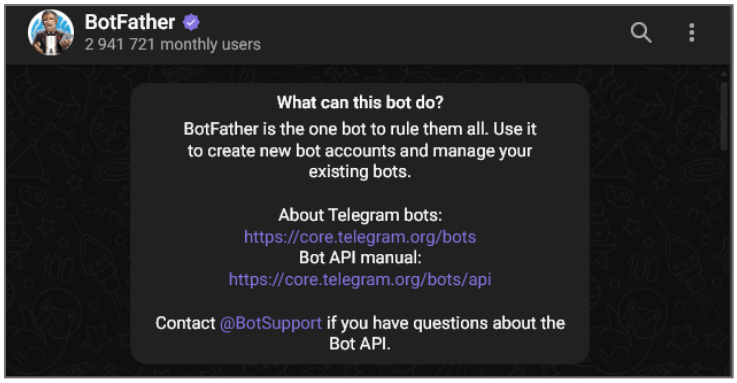


Figura 61 - Creador de un chat bot BotFather.

Posteriormente, se hace clic en **Start** para desplegar un menú de opciones, desde donde se selecciona la opción **Newbot**, tal como se muestra en la figura 62. A continuación, se deben seguir las instrucciones para asignar un nombre al bot.

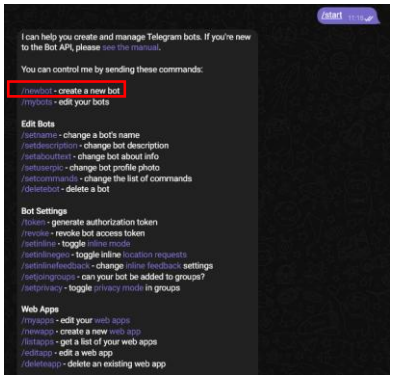


Figura 62 - Creación del Chatbot.

En esta etapa, el bot recibe el nombre de **alarmacombot** y se genera un token, el cual es imprescindible para su configuración en Node-Red y para establecer la comunicación entre ambas plataformas (véase en la figura 63).

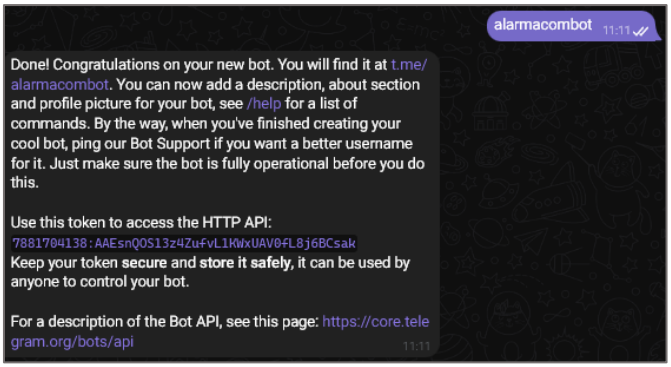


Figura 63 - Creación del bot "alarmacombot".

El siguiente paso consiste en obtener el Chat ID, que funciona como un identificador único para el chat y permite enviar y recibir mensajes desde Node-Red. Para ello, se debe hacer clic en **start** para iniciar la conversación con el bot "alarmacombot" (véase en la figura 64).

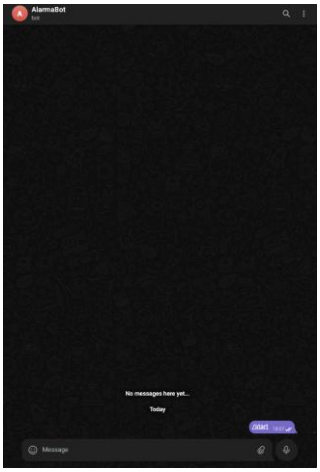


Figura 64 - Chat creado desde Botfather.

Para obtener el Chat ID, es necesario enviar un mensaje al chat creado. A continuación, se debe regresar al apartado de BotFather y hacer clic en el enlace proporcionado por este bot. En el mensaje que aparece como se muestra en la figura 65, se podrá visualizar en Chat ID.

Figura 65 - Mensaje del botfather con las credenciales del chat.

Con esta información, se completa la configuración necesaria para conectar los nodos Telegram Sender y Telegram Receiver. Se accede al nodo de Telegram y se añade una nueva configuración, donde se ingresa el nombre del bot y el token proporcionado, tal como se muestra en la figura 65.

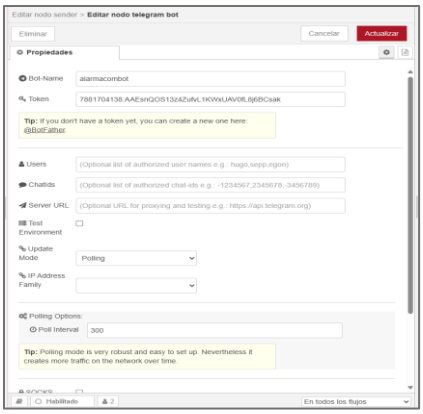
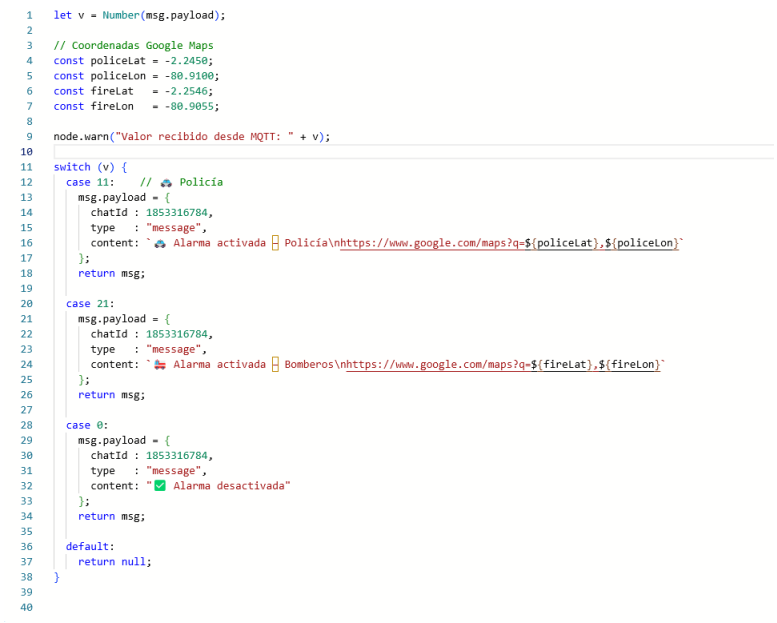


Figura 66 - Configuración del nodo telegram sender.

Para establecer una comunicación adecuada con Telegram, es necesario utilizar el modo Function, en el cual se especifican el Chat ID y el mensaje que se desea enviar a la plataforma de Telegram (véase en la figura 67).

Una vez realizadas estas configuraciones, el flujo estará completo para permitir la comunicación efectiva con la plataforma.



```
let v = Number(msg.payload);

// Coordenadas Google Maps
const policeLat = -2.2450;
const policeLon = -80.9100;
const fireLat   = -2.2546;
const fireLon   = -80.9055;

node.warn("Valor recibido desde MQTT: " + v);

switch (v) {
  case 11:    // 🚓 Policía
    msg.payload = {
      chatId : 1853316784,
      type   : "message",
      content: `🚓 Alarma activada - Policía\nhttps://www.google.com/maps?q=${policeLat},${policeLon}`
    };
    return msg;

  case 21:
    msg.payload = {
      chatId : 1853316784,
      type   : "message",
      content: `🚒 Alarma activada - Bomberos\nhttps://www.google.com/maps?q=${fireLat},${fireLon}`
    };
    return msg;

  case 0:
    msg.payload = {
      chatId : 1853316784,
      type   : "message",
      content: "✅ Alarma desactivada"
    };
    return msg;

  default:
    return null;
}
```

Figura 67 - Nodo Function.

Configuración del mapa

Para visualizar eventuales incidentes en el sistema, se emplea el nodo worldmap. Este nodo fue configurado para separar las instrucciones provenientes de los botones eléctricos, identificados como policía 11 y bombero 21. En este caso se utilizan tres funciones que permiten diferenciar las instrucciones y capturar las coordenadas asociadas a cada botón.

La función encargada de la separación de instrucciones se muestra en la figura 68; esta permite distinguir entre los datos correspondientes a policía y bombero mediante códigos 11 y 21 respectivamente.

Figura 68 - Función de separación de datos.

Para que el nodo worldmap funcione correctamente, es necesario que cada nodo incluya las coordenadas, el ícono y el mensaje que se desea mostrar. La configuración correspondiente a la función se puede observar en la figura 69.

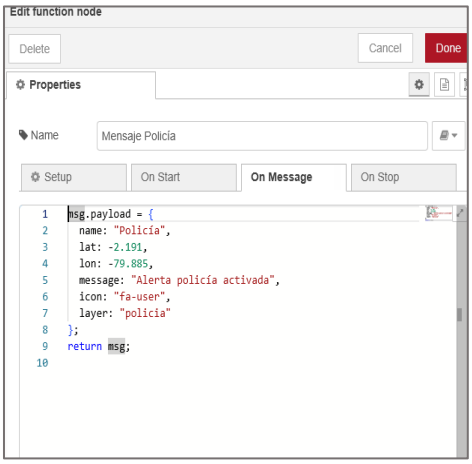


Figura 69 - Configuración de la función policía.

Continuando con la función destinada a bomberos, su configuración se presenta en la figura 70.

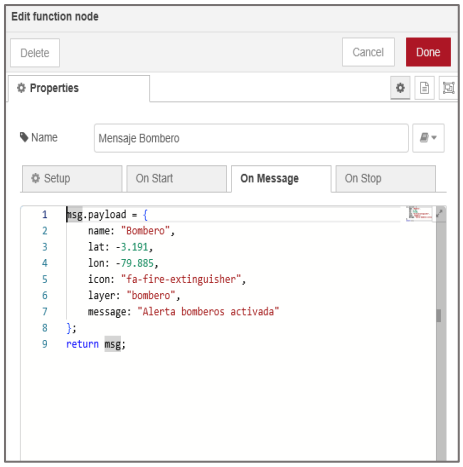


Figura 70 - Configuración de la función bombero.

En la configuración final, se presenta el nodo worldmap, como se observa en la figura 71. En este nodo es fundamental definir una zona de inicio que centre la ubicación en el mapa, además de configurar las capas del mapa y los menús para la interacción del usuario.

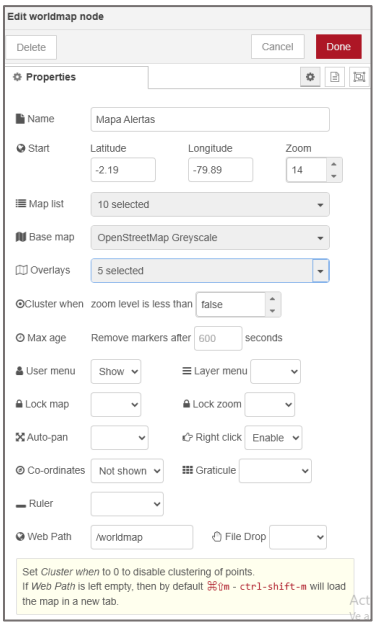


Figura 71 - Nodo Worldmap.

3.5.9 Flujo final en Node-Red

En la figura 72 se muestra el flujo diseñado para establecer la comunicación entre las plataformas TTN y Node-Red para el envío y recepción de datos (El diagrama de flujo se encuentra en el anexo 3). A través de este diseño, es posible visualizar la ubicación de los botones eléctricos al enviar una señal de activación, diferenciándolas entre policía y bomberos. El microcontrolador Esp8266 se encarga de activar y desactivar la alarma comunitaria, finalizando con el envío de notificaciones a Telegram.

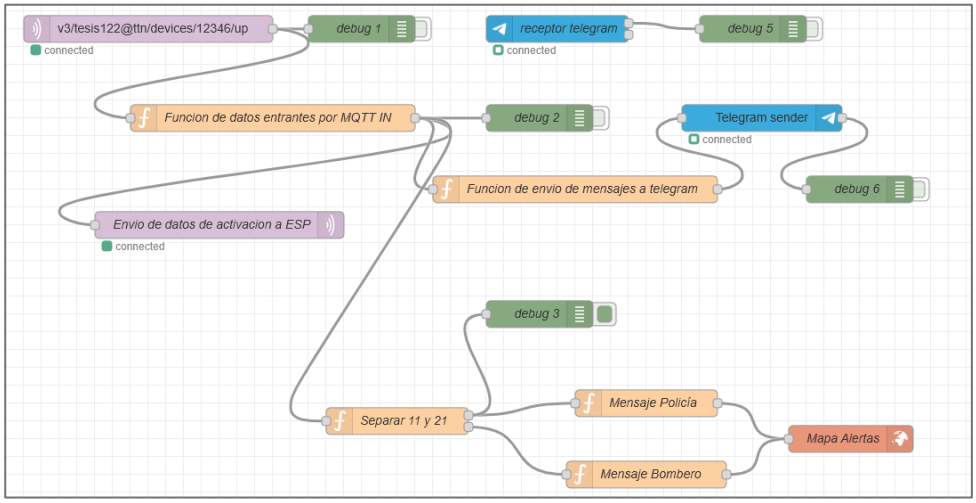


Figura 72 - Diagrama de flujo final en Node Red.

3.6 Estudio de factibilidad

3.6.1 Costos de la propuesta

En este apartado se detallan los costos de los materiales empleados para la implementación del prototipo. En la tabla 11 correspondiente se especifican el precio y la cantidad de cada uno de los equipos utilizados.

Tabla 11 - Presupuesto del material

Elementos	Cantidad	Precio por unidad	Precio total
Gateway Sensecap M2	1	$160,00	$160,00
Heltec HTCC ABC-02	1	$45,00	$45,00
Batería de litio 1500 mA	1	$12,00	$12,00
Batería de litio 5600 mA	1	$15	$15
Relé	1	$2,00	$2,00
Esp8266	1	$32,00	$32,00
Botones eléctricos	2	$5,00	$10,00

Alarma	1	$25,00	$25,00
Tablero	1	$15,00	$15,00
Presupuesto total			$316,00

4 CAPÍTULO IV

4.1 Resultados

Se realizaron pruebas a cada uno de los componentes del sistema de alarma comunitaria, construido sobre LPWAN e IoT. Primero se verificó que todos los dispositivos funcionaran correctamente en condiciones normales; luego se evaluó su comportamiento en diferentes instantes de tiempo simulando escenarios de emergencia. Con esto se comprobó que cada parte responde de manera fiable, lo que confirma que el diseño está preparado para entornos reales donde se requiere reaccionar de inmediato ante situaciones críticas.

4.1.1 Componentes implementados

El prototipo de los botones eléctricos se montó dentro de una caja con certificación IP-65, de modo que los componentes quedan protegidos frente a la humedad y otras condiciones adversas. En su interior, hay dos pulsadores uno rojo y otro azul conectado a una placa con tecnología LoRa. Esta placa toma la señal cuando se pulsa un botón y luego la envía con el fin de encender la sirena comunitaria.

Para mantenerlo operativo sin depender de corriente directa, el sistema usa una batería de litio recargable de 3.7 V y un módulo TP4056. Gracias al puerto USB, recargar resulta sencillo y seguro. El microcontrolador lleva una programación que detecta cuándo se presionan los pulsadores y actúa en consecuencia (ver figura 73).

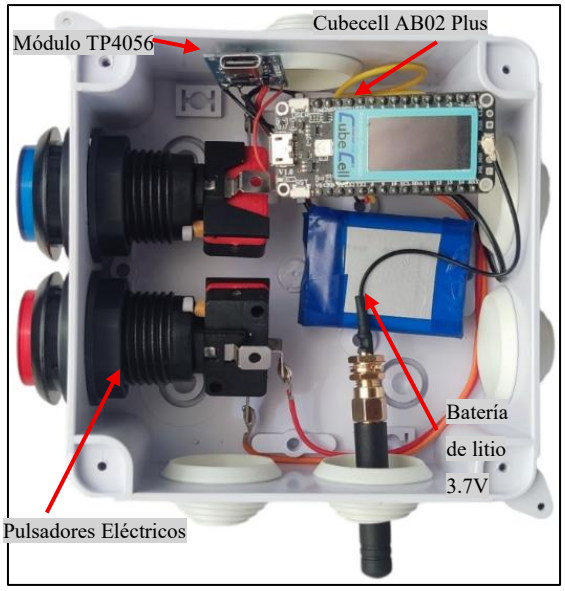


Figura 73 – Implementación de componentes de los botones eléctricos.

El dispositivo autónomo para activar la sirena (ver figura 74) incorpora varios módulos electrónicos que trabajan juntos. El corazón es el Esp8266, encargado de recibir desde Node-Red los valores MQTT (0, 11, 21), procesarlos e indicar al relé, conectado al pin D1 cuándo activar o desactivar la sirena comunitaria.

Para alimentar todo esto se utiliza una batería litio de 3,7 V y 5600 mAh. Con un regulador DC-DC esa tensión sube a 5 V cuando hace falta para ciertos elementos, y otra salida dedicada de 3,3 V garantiza el voltaje estable que requiere el Esp8266 y los circuitos lógicos.

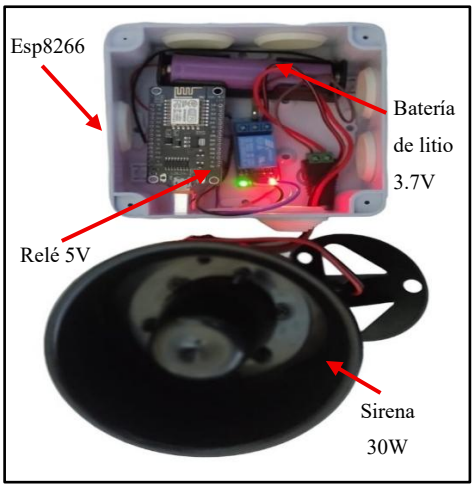


Figura 74 - Implementación de los componentes de la sirena comunitaria.

Tal como muestra la figura 75, el prototipo final incluye, de izquierda a derecha, la antena acoplada al Gateway Sensecap M2 y luego el propio Gateway, que conecta el nodo CubeCell a la red LoRa. A la derecha se encuentra el circuito de la sirena dentro de su caja certificada IP-65, junto a la unidad de botones de pánico. Estos botones aparecen marcados con flechas: la flecha azul apunta a la opción de "Policía" y la flecha roja a la de "Bomberos".

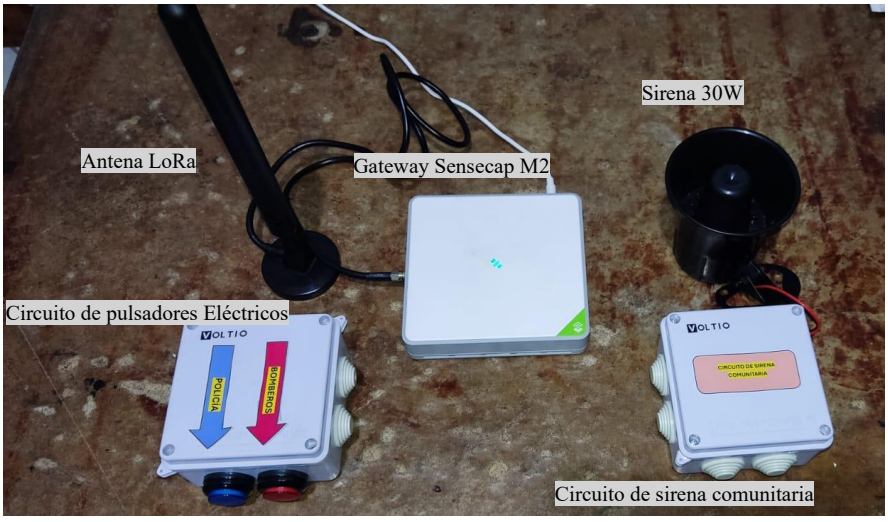


Figura 75 - Prototipo final del sistema de alarma comunitaria aplicando la tecnología Lora.

4.2 Resultados de las pruebas funcionales

4.2.1 Prueba de activación de alarma

Para validar el sistema se realizaron pruebas funcionales usando la placa de desarrollo configurada en modo de activación ABP sobre LoRaWAN 1.0.2. Mediante el monitor serial se mostraba el DevAddr, las claves de sesión y los contadores de trama, lo que confirma que el nodo está listo para transmisión.

Para evitar disparos accidentales, se añadió una lógica de confirmación en los botones de pánico con un retardo de 5 segundos: cuando se presiona el botón, el microcontrolador espera ese tiempo y, si la pulsación se mantiene, envía la alerta.

Al enlazar el dispositivo con el Gateway, se registraron parámetros como RSSI de –101 dBm y SNR de +9 dB, señal de que la cobertura es estable. No se detectaron rebotes, lo cual confirma que el retardo de confirmación filtra eficazmente pulsaciones involuntarias.

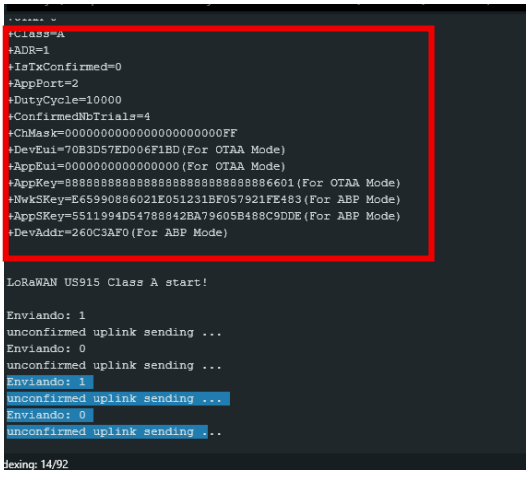


Figura 76 - Monitoreo de datos en el Heltec CubeCell AB02 PLUS.

La figura 77 muestra cómo se comporta el tráfico LoRa gestionado por el Gateway Sensecap M2 durante las pruebas del sistema de alarma comunitaria. Se distinguen tres

grupos de barras apiladas, cada uno correspondiente a una serie de activaciones del botón de alerta. En cada grupo, las barras verdes (paquetes recibidos) y las azules (paquetes transmitidos como confirmación) crecen de manera paralela: por cada paquete que arriba, el Gateway emite su respuesta sin que falte ni sobre ninguno. Esto sugiere que los parámetros de enlace como la potencia de transmisión, la configuración de modulación y las claves ABP están bien calibrados para mantener la comunicación estable.

- Verde indica los paquetes rx que llegan desde el nodo.
- Azul representa los paquetes tx que el Gateway envía de vuelta como confirmación.
- Rojo refleja el total procesado en cada intervalo (suma de rx y tx).

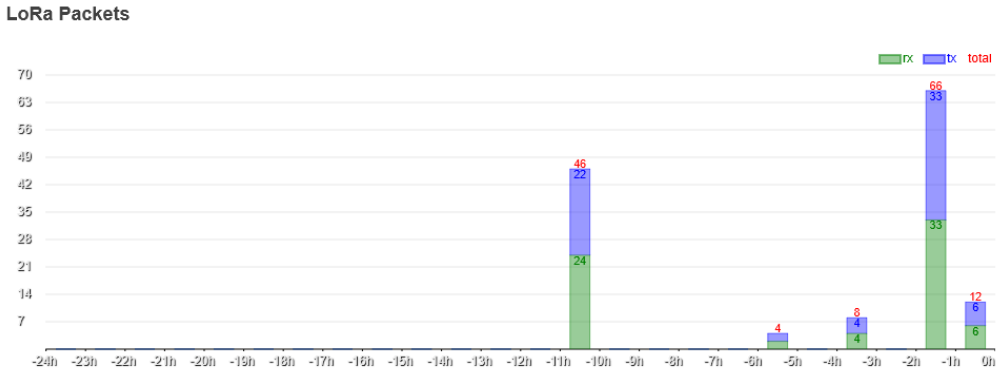


Figura 77 - Paquetes recibidos desde el Heltec al Gateway.

En la figura 78 se visualiza el apartado de Application Data en TTN manifestando cómo llegan los mensajes Uplink. El contador de tramas va subiendo uno tras otro, lo que indica que las transmisiones entran de forma ordenada y sin cortes. Al mismo tiempo, el RSSI ronda los –88 dBm y el SNR está cerca de –3 dB, valores dentro de lo normal para que LoRaWAN funcione bien.

Que los mensajes lleguen decodificados sin problemas y la señal se mantenga estable da confianza de que el enlace está funcionando. Además, al poder ver esto en tiempo real, se tiene la certeza de que el sistema de alarma comunitaria opera correctamente y se puede detectar al instante si algo se sale de lo esperado.

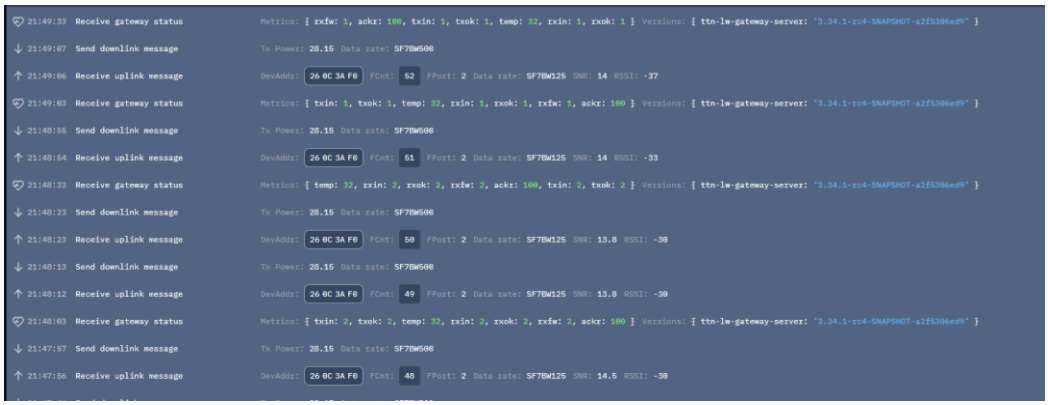


Figura 78 - Visualización de datos en TTN.

En la figura 79, se observa el panel de depuración de Node-Red confirmando que la conexión MQTT ya está activa. Los mensajes que llegan desde TTN aparecen sin faltar ninguno, entran al flujo, se procesan y se envían a Telegram. En el registro se ven tres códigos: el 11 para la alerta de Policía, el 21 para Bomberos y el 0 cuando todo está en reposo o la sirena queda apagada.

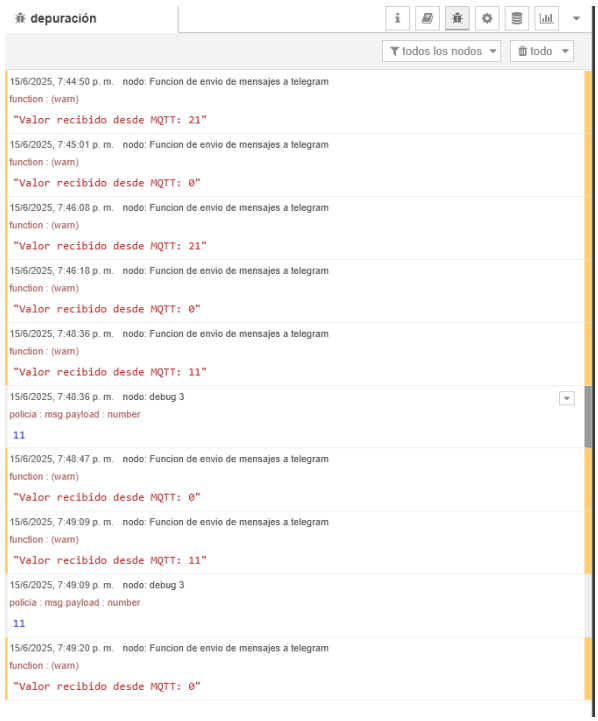


Figura 79 - Recibimiento de datos a Node-Red.

En la ventana de chat del bot de Telegram se distinguen claramente tres notificaciones: “Alarma activada – Bomberos”, “Alarma activada – Policía” y “Alarma desactivada”. Cada mensaje nace del código que envía el nodo CubeCell (21, 11 y 0); Node-Red traduce ese número a texto antes de mandarlo a Telegram. En la secuencia que aparece al lado se observa que los avisos llegan de inmediato, sin demoras apreciables. Esto demuestra que la información recorre la ruta completa desde que se pulsa el botón, pasa por TTN y luego por el flujo MQTT en Node-Red hasta llegar al usuario en Telegram de forma puntual y fiable (ver figura 80).

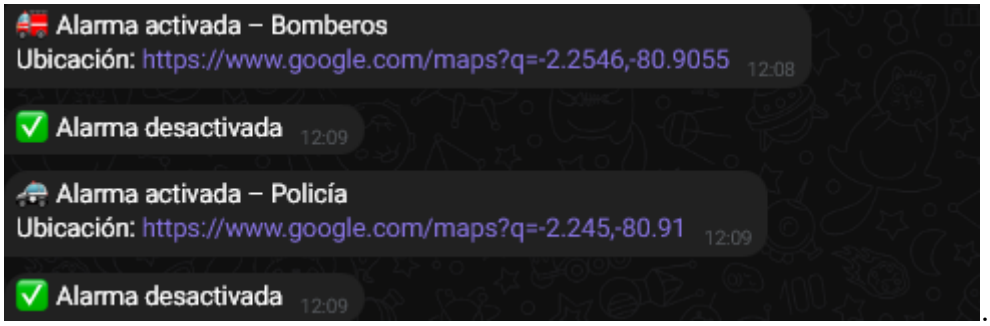


Figura 80 - Señal de alarma activada y desactivada en telegram.

En la consola serial del ESP8266 se evidencia que tras conectarse a la red Wi-Fi y suscribirse al tópico MQTT, el módulo confirma la "Conexión a MQTT exitosa". Asimismo, registra claramente los comandos 11 y 21, correspondientes a la activación de la sirena para Policía y Bomberos respectivamente, así como el comando 0 para su desactivación. La recepción precisa de estos comandos confirma el microcontrolador interpreta y ejecuta correctamente las ordenes enviadas desde Node-Red, cerrando de esta manera el ciclo de control del sistema de alarma (véase figura 81).

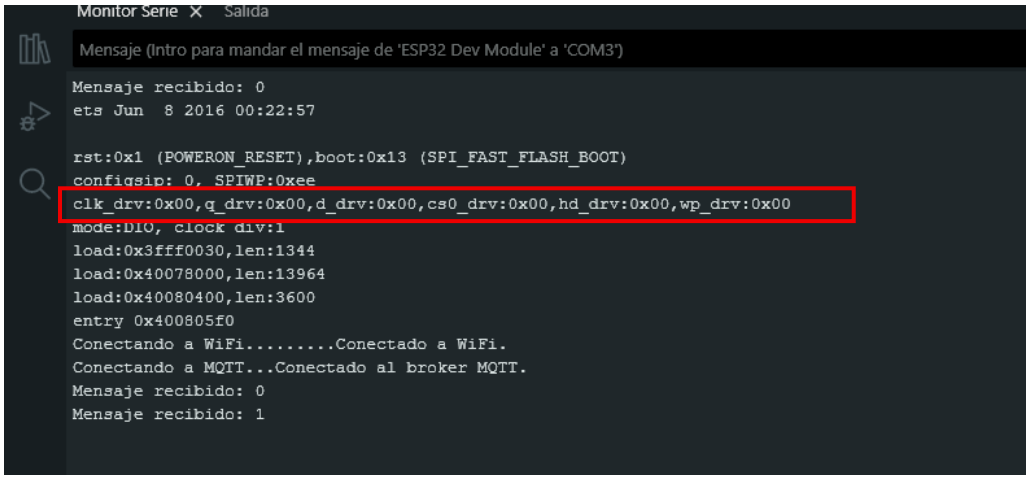


Figura 81 - Monitoreo del Esp8266.

Una vez recibido el aviso en Telegram, el sistema visualiza el evento mediante el panel Worldmap de Node-Red, que posiciona un marcador en las coordenadas transmitidas junto con el código de activación. Cada marcador se identifica mediante un ícono específico que distingue si corresponde a Policía o Bomberos, permitiendo al operador reconocer de forma inmediata el servicio involucrando y la ubicación exacta de la alerta. Cuando el flujo recibe el código de desactivación, el marcador se atenúa o elimina, cerrando visualmente el incidente.

Esta representación geográfica complementa la notificación textual, proporcionando a las autoridades una referencia precisa para la rápida y eficiente movilización de sus recursos (véase en la figura 82).

Figura 82 - Ubicación de la activación del sistema de alarma.

4.2.2 Consumo de energía

Cálculo de consumo energético en botones eléctricos

En el prototipo se llevaron a cabo pruebas de funcionamiento destinadas a evaluar la autonomía en diversos escenarios de operación, con el objetivo de asegurar un desempeño estable óptimo del sistema (véase la tabla 12).

Tabla 12 – Consumo del circuito botones eléctricos.

Componente	Consumo
Cubecell AB02 PLUS (bajo consumo)	3.5 µA
Cubecell AB02 PLUS (Envío de datos por Lora)	55 mA
Total	55.0035 mA

Considerando los requerimientos de consumo del circuito de los pulsadores y las especificaciones de la batería de litio de 3.7 V y 1500 mAh, la autonomía del circuito se determina mediante la siguiente fórmula.

$$\boldsymbol{Autonomia = \frac{Capacidad\ de\ la\ bateria}{Consumo\ del\ dispositivo}}$$

$$\boldsymbol{Autonomia = \frac{1500\ [mAh]}{55.0035\ mA}}$$

$$\boldsymbol{Autonomia = 27.27\ H}$$

Traducido en días

$$\boldsymbol{Días\ de\ autonomia = \frac{27.27\ h}{24\ h} = 1.13\ días}$$

Cálculo de consumo energético en la sirena comunitaria

Se llevaron a cabo ensayos funcionales en diversas condiciones operativas, con el propósito de evaluar tanto la autonomía como la respuesta del sistema en cada escenario. Estos análisis facilitaron la optimización de los parámetros energéticos, garantizando así un desempeño estable y confiable durante el funcionamiento continuo.

Tabla 13 – Consumo del circuito de la sirena comunitaria.

Componente	Consumo
Esp8266 (Modo solo escucha)	40 mAh
Esp8266 (Recepción WI-FI)	56 mAh
Total	96 mAh

Considerando los requerimientos de consumo del circuito de control de la sirena, compuesto por el ESP8266, el regulador DC-DC y el relé, tomando en cuenta las características de la batería de ion-litio de 3,7 V y 5600 mAh integrada en el gabinete, la autonomía teórica del sistema se calcula mediante la siguiente expresión:

$$\boldsymbol{Autonomia = \frac{Capacidad\ de\ la\ bateria}{Consumo\ del\ dispositivo}}$$

$$\boldsymbol{Autonomia = \frac{5600\ [mAh]}{96\ mAh}}$$

$$\boldsymbol{Autonomia = 58.33\ H}$$

Traducido en días

$$\boldsymbol{Días\ de\ autonomia = \frac{58.33\ h}{24\ h} = 2.4\ días}$$

Para dimensionar adecuadamente la fuente de alimentación o la batería que sostenga el circuito, se establece una hipótesis de uso basada en la frecuencia de activaciones. Se asume que el Esp8266 se activa en promedio dos veces al día.

$$\boldsymbol{Autonomia = \frac{5600\ [mAh]}{(96x2)\ mAh}}$$

$$\boldsymbol{Autonomia = 116.66\ H}$$

Traducido en días

$$\boldsymbol{Días\ de\ autonomia = \frac{116.66\ h}{24\ h} = 4.86\ día}$$

Conclusiones

El estudio confirma que LPWAN e IoT permiten alarmas comunitarias innovadoras y de bajo costo, integrando dispositivos y fortaleciendo la comunicación vecinal–autoridades para una respuesta ágil.

El sistema de alarma comunitaria desarrollado a través de tecnología LPWAN e IoT optimiza la respuesta ante emergencias al permitir una comunicación eficiente y en tiempo real entre los ciudadanos y las autoridades. Esto reduce significativamente los tiempos de respuesta y contribuye a la seguridad.

La implementación de la tecnología LoRa y LPWAN ha demostrado ser eficaz, ya que ofrece un bajo consumo energético, una amplia cobertura en áreas urbanas y rurales, y una comunicación robusta, lo que la convierte en una solución ideal para la seguridad pública y emergencias en comunidades de difícil acceso.

Al integrar tecnología IoT y plataformas como Node-Red, se fortalece la participación de la comunidad en la seguridad, permitiendo que los ciudadanos no solo reporten emergencias, sino que también colaboren en la protección mutua, lo cual genera un entorno más seguro y cohesionador.

Recomendaciones

Supervisión de baterías y autonomía: Implementar monitoreo periódico del nivel de carga de los sensores y actuadores para anticipar recargas o reemplazos, y evaluar opciones de baterías de mayor capacidad o fuentes renovables en ubicaciones críticas, asegurando así continuidad de servicio sin depender únicamente de la red eléctrica.

Integración con autoridades de emergencia: Configurar flujos automáticos que envíen alertas estructuradas a despachos de policía y bomberos mediante los canales oficiales que utilicen definiendo con ellos formatos y prioridades claros, y realizando simulacros periódicos para validar la eficacia de las notificaciones.

Análisis de incidentes y mejora continua: Registrar cada activación y generar reportes resumidos que permitan identificar patrones o zonas con más actividad, de modo que se ajusten posiciones de dispositivos, sensibilidades y flujos de notificación, respaldando decisiones para optimizar y replicar el sistema

Bibliografías

[1] "Santa Elena: Estos son los sectores más peligrosos, según la policía," Agosto. Accessed: Apr. 28, 2024. [Online]. Available: https://suscripcion.extra.ec/?limit=true&continue=https://www.extra.ec/noticia/provincias/santa-elena-violencia-sectores-peligrosos-Libertad-90056.html

[2] Karla Gabriela Chicaiza Guachi, "'SISTEMA DE ALARMA COMUNITARIA PARA EL MERCADO SAN JUAN DE LA CIUDAD SANTIAGO DE PÍLLARO,'" UNIVERSIDAD TÉCNICA DE AMBATO, Ambato, 2020.

[3] J. C. F. R. Germán Baldo, "Telemetría de luminarias en aeropuertos utilizando LoRaWAN," 2020.

[4] Jonnathan Mauricio Ayabaca Espinoza Pablo Fernando Cochancela Solórzano, "DESARROLLO DE UNA PLATAFORMA PARA LA GESTIÓN Y MONITOREO DE ALARMAS RESIDENCIALES INTEGRADAS A UN SISTEMA DE ALARMA COMUNITARIA," Cuenca, 2023.

[5] F. Orellana Batallas and D. Alfonso Caveda, "Factores que contribuyen al aumento de la delincuencia en el Ecuador.," *REVISTA CIENTÍFICA ECOCIENCIA*, vol. 9, pp. 276–294, Dec. 2022, doi: 10.21855/ECOCIENCIA.90.766.

[6] Daniel Montalvo, "Ecuador registra los niveles más altos de crimen, inseguridad y delincuencia del continente," Ecuador registra los niveles más altos de crimen, inseguridad y delincuencia del continente. Accessed: Jun. 12, 2024. [Online]. Available: https://www.participacionciudadana.org/web/wp-content/uploads/2024/02/A1-Ecuador-registra-los-niveles-mas-altos-de-crimen.pdf

[7] PRIMICIAS, "La violencia migra a Santa Elena por presión a bandas en Guayas." Accessed: Jun. 12, 2024. [Online]. Available: https://www.primicias.ec/noticias/en-exclusiva/violencia-santa-elena-bandas-guayas/

[8] Alexander Garcia, "Santa Elena registra un incremento del 88% de inseguridad." Accessed: Jun. 12, 2024. [Online]. Available:

https://www.primicias.ec/noticias/seguridad/santa-elena-muertes-violencia-bandas-ecuador/

[9] PLAN V, “Cantón la Libertad como uno de los mas peligrosos.” Accessed: Jun. 12, 2024. [Online]. Available: https://www.planv.com.ec/historias/crimen-organizado/cuatro-cantones-ecuador-se-perfilan-ingresar-el-listado-ciudades-mas

[10] José Ignacio Ruiz, “Los efectos que la inseguridad .” Accessed: Jun. 12, 2024. [Online]. Available: https://periodico.unal.edu.co/articulos/los-efectos-que-la-inseguridad-y-el-miedo-al-crimen-generan-en-las-personas

[11] Hazel Villalobos Fonseca, “El desarrollo tecnológico en materia policial: una receta de éxito para la prevención del delito,” vol. 15(1), 2020, Accessed: Jun. 10, 2024. [Online]. Available: http://www.scielo.org.co/pdf/ries/v15n1/1909-3063-ries-15-01-79.pdf

[12] Alexandra Ramírez Castro, “Riesgo tecnológico y su impacto para las organizaciones parte I | Revista .Seguridad.” Accessed: Jun. 03, 2024. [Online]. Available: https://revista.seguridad.unam.mx/numero-14/riesgo-tecnol%C3%B3gico-y-su-impacto-para-las-organizaciones-parte-i

[13] Franklin Álvarez Salinas, “La investigación en la formación tecnológica de la Policía Nacional del Ecuador Research in the technological training of the National Police of Ecuador,” 2020. [Online]. Available: https://orcid.org/0000-0001-6766-9399

[14] Hector Muñoz, “La tecnología en la seguridad pública | LinkedIn,” 12/07/2023. Accessed: Jun. 08, 2024. [Online]. Available: https://www.linkedin.com/pulse/la-tecnolog%C3%ADa-en-seguridad-p%C3%BAblica-h%C3%A9ctor-mu%C3%B1oz/

[15] EXPOST, “El impacto de la tecnología en seguridad pública en México,” 02/06/2022. Accessed: Jun. 08, 2024. [Online]. Available: https://www.iexe.edu.mx/tecnologia/el-impacto-de-la-tecnologia-en-materia-de-seguridad-publica-en-mexico/

[16] Universidad de Galileo, “El Papel de la Tecnología en la Prevención del Delito: Algunas aplicaciones más notables - IES,” 04/12/2023. Accessed: Jun. 08, 2024. [Online]. Available: https://www.galileo.edu/ies/historias-de-exito/el-papel-de-la-tecnologia-en-la-prevencion-del-delito-algunas-aplicaciones-mas-notables/

[17] P. H. T. Jeniffer Herrera Araujo, “‘La Seguridad Ciudadana y la influencia de la participación ciudadana en las estrategias de prevención del delito en la frontera norte:

Caso de estudio Tulcán en el periodo 2014 – 2017,'" 2020. Accessed: Jun. 12, 2024. [Online]. Available: http://repositorio.upec.edu.ec/bitstream/123456789/1227/1/090-%20HERRERA%20JENIFFER-%20HUERA%20PEDRO.pdf

[18] M. Nevárez-Toledo, W. Mecía-Vélez, and V. Yánez-Ortiz, "Sistema de monitoreo delincuencial en viviendas basado en Internet de las Cosas," *3C Tecnología_Glosas de innovación aplicadas a la pyme*, vol. 8, no. 3, pp. 24–43, Sep. 2019, doi: 10.17993/3ctecno/2019.v8n3e31.24-43.

[19] Martín Álvarez., "Estándares del W3C para IoT | CTIC," 2019. Accessed: Jun. 12, 2024. [Online]. Available: https://www.fundacionctic.org/es/actualidad/estandares-del-w3c-para-iot

[20] Kaspersky, "Los riesgos de seguridad y las buenas prácticas de la Internet de las cosas," 2020. Accessed: Jun. 12, 2024. [Online]. Available: https://www.kaspersky.es/resource-center/preemptive-safety/best-practices-for-iot-security

[21] Infosegura, "Guatemala: Situación de seguridad ciudadana 2020," 2021. Accessed: Jun. 12, 2024. [Online]. Available: https://infosegura.org/en/guatemala/guatemala-2020s-citizen-security-situation

[22] Alec Jahnke, "Las 4 etapas de la arquitectura IoT | Digi International," 31 de julio. Accessed: Jun. 16, 2024. [Online]. Available: https://es.digi.com/blog/post/the-4-stages-of-iot-architecture

[23] David Cuartielles, "Arquitectura IoT, prototipando los dispositivos del futuro." Accessed: Jun. 16, 2024. [Online]. Available: https://programarfacil.com/podcast/arduino-wifi-proyectos-iot/

[24] L. González, O. Sofía, D. Laguía, E. Gesto, and K. Hallar, "Internet del Futuro – Estudio de tecnologías IoT," *Informes Científicos Técnicos - UNPA*, vol. 12, no. 3, pp. 105–137, Dec. 2020, doi: 10.22305/ict-unpa.v12.n3.744.

[25] Carlos Isaac Ramos Incháustegui, "'GATEWAY LoRa para la PLATAFORMA CLOUDINO,'" 2019.

[26] CATSENSORS, "Tecnología LoRA y LoRAWAN - Catsensors." Accessed: Jun. 19, 2024. [Online]. Available: https://www.catsensors.com/es/lorawan/tecnologia-lora-y-lorawan

[27] UNIR, “LoRaWAN: ¿en qué consiste este protocolo y para qué se emplea?,” 17 de Enero. Accessed: Jun. 13, 2024. [Online]. Available: https://www.unir.net/ingenieria/revista/lorawan/

[28] Venco Electrónica, “Qué es LoRa, cómo funciona y características principales,” 25 de Agosto. Accessed: Jun. 13, 2024. [Online]. Available: https://www.vencoel.com/que-es-lora-como-funciona-y-caracteristicas-principales/

[29] K. Andrés and B. Santos, “Pruebas de Seguridad Aplicadas a Infraestructura IoT,” 2023.

[30] I. O. Monfort, “Estudio de la arquitectura y el nivel de desarrollo de la red LoRaWAN y de los dispositivos LoRa”.

[31] Alfaiot, “Sigfox, la red de IoT más grande del mundo,” 15 de Enero. Accessed: Jun. 14, 2024. [Online]. Available: https://alfaiot.com/iot/sigfox-la-red-de-iot-mas-grande-del-mundo/

[32] Telefonica, “¿Qué es NB-IoT y cómo funciona?” Accessed: Jun. 16, 2024. [Online]. Available: https://www.telefonica.com/es/sala-comunicacion/blog/que-es-nb-iot-y-como-funciona/

[33] MOKO LORA, “Cómo las tecnologías LPWAN potencian el futuro de la conectividad IoT.” Accessed: Jun. 16, 2024. [Online]. Available: https://www.mokolora.com/es/how-lpwan-technologies-empower-the-future/

[34] Leonador Eugenio Vera Sanchez, “Implementación de un prototipo de red de sensores inalámbricos en topología lineal para la detección de eventos utilizando el estándar IEE 802.15.4,” 2022.

[35] IoT FACTORY, “Best LORAWAN Network Servers,” 2020. Accessed: Feb. 02, 2025. [Online]. Available: https://iotfactory.eu/products/software-platform/best-lorawan-network-servers/

[36] Luis del Valle Hernández, “Introducción a Node-RED y Raspberry Pi: sistema de alarma con Arduino.” Accessed: Jun. 16, 2024. [Online]. Available: https://programarfacil.com/blog/raspberry-pi/introduccion-node-red-raspberry-pi/

[37] L. P. ,Henry piña, Jesús Nieblas and J. O. Alfredo Cervantes, “Vista de Node-RED: Una Herramienta de Acceso Libre para el Control de Velocidad en Motores Trifásicos.” Accessed: Jun. 16, 2024. [Online]. Available: https://ciencialatina.org/index.php/cienciala/article/view/9076/13531

[38] Enpresa Digitala, “NODE-RED Gestión de datos en tiempo real (actualizado) - Enpresa Digitala.” Accessed: Jun. 17, 2024. [Online]. Available: https://enpresadigitala.spri.eus/es/node-red-gestioacuten-de-datos-en-tiempo-real/

[39] BORIS RAÚL CUTOS MANOTOA and MIGUEL ÁNGEL RECALDE CHÁVEZ, “UNIVERSIDAD POLITÉCNICA SALESIANA SEDE QUITO CARRERA DE INGENIERÍA DE SISTEMAS MONITORIZACIÓN Y CONTROL DE SEGURIDAD EN HOGARES,” 2022.

[40] Alexander Melian, “HTTP y HTTPS con Node.js. Hablaremos un poco sobre HTTP y HTTPS… | by Alexander Melian | Apr, 2024 | Medium.” Accessed: Jun. 17, 2024. [Online]. Available: https://medium.com/@alexandermelian/http-y-https-con-node-js-9ec30440fd7c

[41] INE4C Electronics, “Introducción MQTT con Node Red y Arduino - ine4celectronics.com.” Accessed: Jun. 17, 2024. [Online]. Available: https://ine4celectronics.com/introduccion-mqtt-con-node-red-y-arduino/

[42] “6 Tips To Ensure Your Next Z-Wave Install Is Successful.” Accessed: Feb. 02, 2025. [Online]. Available: https://www.clarecontrols.com/dealer-news/6-z-wave-tips-and-tricks

[43] “¿Qué es LoRaWAN? | Digi International.” Accessed: Feb. 02, 2025. [Online]. Available: https://es.digi.com/solutions/by-technology/lorawan

[44] “Tecnología de red celular de celda pequeña.” Accessed: Feb. 02, 2025. [Online]. Available: https://www.data-alliance.net/es/tecnolog%C3%ADa-de-red-celular-de-celda-peque%C3%B1a

[45] IKUSI, “Protocolos de comunicación: ¿cómo transmiten datos de forma eficiente?” Accessed: Jun. 17, 2024. [Online]. Available: https://www.ikusi.com/mx/blog/protocolos-de-comunicacion/

Anexos

Anexo 1 - Codificación del CubeCell AB02 PLUS con ARDUINO IDE

```
#include "LoRaWan_APP.h"

#include "Arduino.h"


/* OTAA configuration */

uint8_t devEui[] = { 0x70, 0xB3, 0xD5, 0x7E, 0xD0, 0x06, 0xF1, 0xBD };

uint8_t appEui[] = { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };

uint8_t appKey[] = { 0x88, 0x88, 0x88, 0x88, 0x88, 0x88, 0x88, 0x88, 0x88, 0x88, 0x88,
0x88, 0x88, 0x88, 0x66, 0x01 };


/* ABP configuration */

uint8_t nwkSKey[] = { 0xE6, 0x59, 0x90, 0x88, 0x60, 0x21, 0xE0, 0x51, 0x23, 0x1B, 0xF0,
0x57, 0x92, 0x1F, 0xE4, 0x83 };

uint8_t appSKey[] = { 0x55, 0x11, 0x99, 0x4D, 0x54, 0x78, 0x88, 0x42, 0xBA, 0x79, 0x60,
0x5B, 0x48, 0x8C, 0x9D, 0xDE };

uint32_t devAddr = (uint32_t)0x260C3AF0;


/* LoraWan settings */

uint16_t userChannelsMask[6] = { 0x00FF, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000 };

LoRaMacRegion_t loraWanRegion = ACTIVE_REGION;

DeviceClass_t loraWanClass = LORAWAN_CLASS;

bool overTheAirActivation = LORAWAN_NETMODE;

bool loraWanAdr = LORAWAN_ADR;

bool keepNet = LORAWAN_NET_RESERVE;

bool isTxConfirmed = LORAWAN_UPLINKMODE;

uint8_t appPort = 2;

uint8_t confirmedNbTrials = 4;
```

```
unsigned long appTxDutyCycle = 5000;
unsigned long tiempoPolicia = 0;
unsigned long tiempoBombero = 0;

#define BOTON_POLICIA_PIN GPIO1
#define BOTON_BOMBERO_PIN GPIO2

bool estadoPolicia = false;
bool estadoBombero = false;
bool estadoPoliciaAnterior = HIGH;
bool estadoBomberoAnterior = HIGH;

bool hayCambio = false;
int valorEnviar = 0;

static void prepareTxFrame(uint8_t port) {
    appDataSize = 1;
    appData[0] = valorEnviar;

    Serial.println(valorEnviar);
}
```

```
void setup() {
   Serial.begin(115200);
   pinMode(BOTON_POLICIA_PIN, INPUT_PULLUP);
   pinMode(BOTON_BOMBERO_PIN, INPUT_PULLUP);

   #if(AT_SUPPORT)
      enableAt();
   #endif
   deviceState = DEVICE_STATE_INIT;
   LoRaWAN.ifskipjoin();
}
void loop() {
   bool lecturaPolicia = digitalRead(BOTON_POLICIA_PIN);
   bool lecturaBombero = digitalRead(BOTON_BOMBERO_PIN);

   // BOTÓN POLICÍA
   if (estadoPoliciaAnterior == HIGH && lecturaPolicia == LOW) {
      delay(50);
      if (digitalRead(BOTON_POLICIA_PIN) == LOW) {
         estadoPolicia = true;
         estadoBombero = false;  // Desactiva bombero si estaba activo
         tiempoPolicia = millis();
         valorEnviar = 11;
         hayCambio = true;
      }
   }
```

```
/ BOTÓN BOMBERO
  if (estadoBomberoAnterior == HIGH && lecturaBombero == LOW) {
    delay(50);
    if (digitalRead(BOTON_BOMBERO_PIN) == LOW) {
      estadoBombero = true;
      estadoPolicia = false;  // Desactiva policía si estaba activo
      tiempoBombero = millis();
      valorEnviar = 21;
      hayCambio = true;
    }
  }

  estadoPoliciaAnterior = lecturaPolicia;
  estadoBomberoAnterior = lecturaBombero;

  // AUTO-DESACTIVAR a los 10 segundos
  if ((estadoPolicia && millis() - tiempoPolicia > 10000) ||
    (estadoBombero && millis() - tiempoBombero > 10000)) {
    estadoPolicia = false;
    estadoBombero = false;
    valorEnviar = 0;
    hayCambio = true;
  }
```

```
switch (deviceState) {
    case DEVICE_STATE_INIT:
      #if(LORAWAN_DEVEUI_AUTO)
        LoRaWAN.generateDeveuiByChipID();
      #endif
      printDevParam();
      LoRaWAN.init(loraWanClass, loraWanRegion);
      deviceState = DEVICE_STATE_JOIN;
      break;
    case DEVICE_STATE_JOIN:
      LoRaWAN.join();
      break;
    case DEVICE_STATE_SEND:
      if (hayCambio) {
        prepareTxFrame(appPort);
        LoRaWAN.send();
        hayCambio = false;
      }
      deviceState = DEVICE_STATE_CYCLE;
      break;
    case DEVICE_STATE_CYCLE:
      txDutyCycleTime = appTxDutyCycle + randr(0, APP_TX_DUTYCYCLE_RND);
      LoRaWAN.cycle(txDutyCycleTime);
```

```
 deviceState = DEVICE_STATE_SLEEP;
            break;
        case DEVICE_STATE_SLEEP:
            LoRaWAN.sleep();
            break;
        default:
            deviceState = DEVICE_STATE_INIT;
            break;
    }
}
```

Anexo 2 - Codificación del del Esp8266 en ARDUINO IDE

```
#include <ESP8266WiFi.h>
#include <PubSubClient.h>

// --- Configuración WiFi ---
const char* ssid     = "RICARDO_NETLIFE";
const char* password = "Riplu2025.1";

// --- Configuración MQTT ---
const char* mqtt_server = "192.168.100.7";
const int   mqtt_port   = 1883;
const char* mqtt_topic  = "esp8266/tesis";

WiFiClient    espClient;
PubSubClient  client(espClient);

const int relePin = 5;

void setup_wifi() {
  Serial.print("Conectando a WiFi");
  WiFi.begin(ssid, password);
  unsigned long start = millis();
  while (WiFi.status() != WL_CONNECTED && millis() - start < 10000) {
    delay(500);
    Serial.print(".");
  }
```

```
 if (WiFi.status() == WL_CONNECTED) {
   Serial.println(" OK");
   Serial.print("IP: ");
   Serial.println(WiFi.localIP());
  } else {
   Serial.println(" ERROR");
  }
}

void callback(char* topic, byte* payload, unsigned int length) {
  String msg;
  for (unsigned int i = 0; i < length; i++) {
   msg += char(payload[i]);
  }
  Serial.printf("Recibido en [%s]: %s\n", topic, msg.c_str());

  // Lógica invertida: 11 y 21 apagan (LOW), 0 enciende (HIGH)
  if (msg == "11" || msg == "21") {
   digitalWrite(relePin, LOW);   // Apagar relé
   Serial.println("Relé APAGADO");
  }
  else if (msg == "0") {
   digitalWrite(relePin, HIGH);  // Encender relé
   Serial.println("Relé ENCENDIDO");
  }
}
```

```
void reconnect() {
  while (!client.connected()) {
    Serial.print("Conectando a MQTT...");
    if (client.connect("ESP8266Client")) {
      Serial.println(" Conectado");
      client.subscribe(mqtt_topic);
    } else {
      Serial.printf(" Error: %d\n", client.state());
      delay(2000);
    }
  }
}

void setup() {
  Serial.begin(115200);
  pinMode(relePin, OUTPUT);
  digitalWrite(relePin, LOW);  // Estado inicial: relé APAGADO

  setup_wifi();

  client.setServer(mqtt_server, mqtt_port);
  client.setCallback(callback);
}

void loop() {
  if (WiFi.status() != WL_CONNECTED) {
```

Anexo 3 - JASON COMPACT DE NODE RED

[{"id":"3e1fc1219d3b2c08","type":"debug","z":"b98328f3a61f0e81","name":"debug 1","active":false,"tosidebar":true,"console":false,"tostatus":false,"complete":"payload","targetType":"msg","statusVal":"","statusType":"auto","x":500,"y":160,"wires":[]},{"id":"abc44b1143bf307a","type":"function","z":"b98328f3a61f0e81","name":"Funcion de datos entrantes por MQTT IN","func":"\nlet base64data = msg.payload.uplink_message.frm_payload;\n\nlet buffer = Buffer.from(base64data, 'base64');\n\nmsg.payload = buffer[0]; \n\nreturn msg;\n","outputs":1,"timeout":0,"noerr":0,"initialize":"","finalize":"","libs":[],"x":400,"y":260,"wires":[["f36ce86a1bf09131","c1f2f7f7dcb4aebf","a5cc4da190cd8df6","b59a2232bce53952"]]},{"id":"f36ce86a1bf09131","type":"debug","z":"b98328f3a61f0e81","name":"debug 2","active":false,"tosidebar":true,"console":false,"tostatus":false,"complete":"false","statusVal":"","statusType":"auto","x":700,"y":260,"wires":[]},{"id":"c1f2f7f7dcb4aebf","type":"mqtt out","z":"b98328f3a61f0e81","name":"Envio de datos de activacion a ESP","topic":"esp8266/tesis","qos":"0","retain":"false","respTopic":"","contentType":"","userProps":"","correl":"","expiry":"","broker":"a1c45f0a31f19bab","x":340,"y":380,"wires":[]},{"id":"c0348f3b76885d38","type":"debug","z":"b98328f3a61f0e81","name":"debug 5","active":false,"tosidebar":true,"console":false,"tostatus":false,"complete":"false","statusVal":"","statusType":"auto","x":940,"y":160,"wires":[]},{"id":"3cae1542b9ea84b6","type":"debug","z":"b98328f3a61f0e81","name":"debug 6","active":false,"tosidebar":true,"console":false,"tostatus":false,"complete":"false","statusVal":"","statusType":"auto","x":1060,"y":340,"wires":[]},{"id":"a5cc4da190cd8df6","type":"function","z":"b98328f3a61f0e81","name":"Funcion de envio de mensajes a telegram","func":"let v = Number(msg.payload); // valor recibido (0,11,21)\n\n// Coordenadas\nconst latPoli = -2.2450;\nconst lonPoli = -80.91;\nconst latBom = -2.2546;\nconst lonBom = -80.9055;\n\nswitch (v) {\n case 11:   // 🚓 Policía\n   // Link solo de policía\n   const linkPoli = `https://www.google.com/maps?q=${latPoli},${lonPoli}`;\n   msg.payload = {\n   chatId: 1853316784,\n   type : \"message\",\n   content: `🚓 Alarma activada – Policía\\nUbicación: ${linkPoli}`\n   };\n   return msg;\n\n case 21

🚒 Alarma activada – Bomberos\\nUbicación: ${linkBom}`\n };\n return msg;\n\n case 0: // ✅ Desactivada\n msg.payload = {\n chatId: 1853316784,\n type : \"message\",\n content: \"✅ Alarma desactivada\"\n };\n return msg;\n\n default:\n return null;\n}\n","outputs":1,"timeout":0,"noerr":0,"initialize":"","finalize":"","libs":[],"x":740,"y":340,"wires":[["353c9d98d38f9125"]]},{"id":"41659fd14fe91b32","type":"mqtt in","z":"b98328f3a61f0e81","name":"","topic":"v3/tesis122@ttn/devices/12346/up","qos":"2","datatype":"auto-detect","broker":"6eb4c4af7e986704","nl":false,"rap":true,"rh":0,"inputs":0,"x":260,"y":160,"wires":[["3e1fc1219d3b2c08","abc44b1143bf307a"]]},{"id":"1630dafeeffc6ad2","type":"telegram receiver","z":"b98328f3a61f0e81","name":"receptor telegram","bot":"452d1d9715a0287b","saveDataDir":"","filterCommands":false,"x":720,"y":160,"wires":[["c0348f3b76885d38"],[]]},{"id":"353c9d98d38f9125","type":"telegram sender","z":"b98328f3a61f0e81","name":"","bot":"452d1d9715a0287b","haserroroutput":false,"outputs":1,"x":950,"y":260,"wires":[["3cae1542b9ea84b6"]]},{"id":"2ad732b5d9714400","type":"debug","z":"b98328f3a61f0e81","name":"debug 3","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"payload","targetType":"msg","statusVal":"","statusType":"auto","x":760,"y":400,"wires":[]},{"id":"b59a2232bce53952","type":"function","z":"b98328f3a61f0e81","name":"Separar 11 y 21","func":"const val = Number(msg.payload);\nif (val == 11) {\n msg.topic = \"policia\";\n return [msg, null];\n} else if (val == 21) {\n msg.topic = \"bombero\";\n return [null, msg];\n} else {\n return null;\n}","outputs":2,"timeout":"","noerr":0,"initialize":"","finalize":"","libs":[],"x":560,"y":520,"wires":[["c068f1b40de75779","2ad732b5d9714400"],["f7fc5ab6f1519e57"]]},{"id":"c068f1b40de75779","type":"function","z":"b98328f3a61f0e81","name":"Mensaje Policía","func":"msg.payload = {\n name: \"alarma_policia\",\n lat: -2.2450,\n lon: -80.91,\n icon: \"fa-user\",\n layer: \"policia\",\n label: \"🚓 Policía: alerta activada\",\n popup: \"Alerta Policía activada
Ver en Google Maps\"\n};\nreturn msg;\n","outputs":1,"timeout":"","noerr":0,"initialize":"","finalize":"","libs":[],"x":780,"y":440,"wires":[["618378f3c9d602d8"]]},{"id":"f7fc5ab6f1519e57","type":"function","z":"b98328f3a61f0e81","name":"Mensaje Bombero","func":"msg.payload = {\n name: \"alarma_bombero\",\n lat: -2.2546,\n lon: -80.9055,\n icon:

\"fa-fire-extinguisher\",\n layer: \"bombero\",\n label: \"🚒 Bomberos: alerta activada\",\n popup: \"Alerta Bomberos activada
Ver en Google Maps\"\n};\nreturn msg;\n","outputs":1,"timeout":"","noerr":0,"initialize":"","finalize":"","libs":[],"x":790,"y":500,"wires":[["618378f3c9d602d8"]]},{"id":"618378f3c9d602d8","type":"worldmap","z":"b98328f3a61f0e81","name":"Mapa Alertas","lat":"-2.24462","lon":"-80.9055","zoom":"14","layer":"OSMG","cluster":"false","maxage":"","usermenu":"show","layers":"policia,bombero","hiderightclick":"false","coords":"none","path":"/worldmap","overlist":"DR,CO,RA,DN,HM","maplist":"OSMG,OSMC,EsriC,EsriS,EsriT,EsriO,EsriDG,NatGeo,UKOS,OpTop","mapname":"","mapurl":"","mapopt":"","mapwms":false,"x":990,"y":460,"wires":[]},{"id":"a1c45f0a31f19bab","type":"mqtt-broker","name":"","broker":"192.168.100.7","port":1883,"clientid":"","autoConnect":true,"usetls":false,"protocolVersion":4,"keepalive":60,"cleansession":true,"autoUnsubscribe":true,"birthTopic":"","birthQos":"0","birthRetain":"false","birthPayload":"","birthMsg":{},"closeTopic":"","closeQos":"0","closeRetain":"false","closePayload":"","closeMsg":{},"willTopic":"","willQos":"0","willRetain":"false","willPayload":"","willMsg":{},"userProps":"","sessionExpiry":""},{"id":"6eb4c4af7e986704","type":"mqtt-broker","name":"","broker":"nam1.cloud.thethings.network","port":1883,"clientid":"","autoConnect":true,"usetls":false,"protocolVersion":4,"keepalive":60,"cleansession":true,"autoUnsubscribe":true,"birthTopic":"","birthQos":"0","birthRetain":"false","birthPayload":"","birthMsg":{},"closeTopic":"","closeQos":"0","closeRetain":"false","closePayload":"","closeMsg":{},"willTopic":"","willQos":"0","willRetain":"false","willPayload":"","willMsg":{},"userProps":"","sessionExpiry":""},{"id":"452d1d9715a0287b","type":"telegram bot","botname":"alarmacombot","usernames":"","chatids":"","baseapiurl":"","testenvironment":false,"updatemode":"polling","pollinterval":300,"usesocks":false,"sockshost":"","socksprotocol":"socks5","socksport":6667,"socksusername":"anonymous","sockspassword":"","bothost":"","botpath":"","localbothost":"0.0.0.0","localbotport":8443,"publicbotport":8443,"privatekey":"","certificate":"","useselfsignedcertificate":false,"sslterminated":false,"verboselogging":false}]