



UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE GUAYAQUIL
CARRERA DE MECATRÓNICA

**DESARROLLO DE UN SISTEMA DE MONITOREO INALÁMBRICO
DE TEMPERATURA Y HUMEDAD APLICADO A UN SISTEMA DE
RIEGO EN ZONAS RURALES BASADO EN ARDUINO Y
COMUNICACIÓN LORA WAN**

Trabajo de titulación previo a la obtención del
Título de Ingeniero en Mecatrónica

AUTORES: Julio Alberto Alaña Pután
Irving Airthon Moreira Calle
TUTOR: Ing. Jorge Bladimir Fariño Cedeño, Mgtr.

Guayaquil - Ecuador
2025

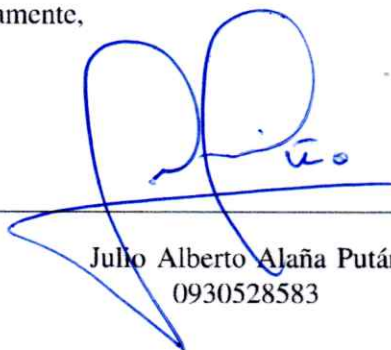
CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Nosotros, **Julio Alberto Alaña Pután** con documento de identificación N° **0930528583** y **Irving Airthon Moreira Calle** con documento de identificación N° **0952328649**; manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo.

Guayaquil, 20 de febrero de 2025

Atentamente,



Julio Alberto Alaña Pután
0930528583



Irving Airthon Moreira Calle
0952328649

**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA
UNIVERSIDAD POLITÉCNICA SALESIANA**

Nosotros, **Julio Alberto Alaña Pután** con documento de identificación N° **0930528583** y **Irving Airthon Moreira Calle** con documento de identificación N° **0952328649**, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del **Dispositivo Tecnológico: DESARROLLO DE UN SISTEMA DE MONITOREO INALÁMBRICO DE TEMPERATURA Y HUMEDAD APLICADO A UN SISTEMA DE RIEGO EN ZONAS RURALES BASADO EN ARDUINO Y COMUNICACIÓN LORA WAN**, el cual ha sido desarrollado para optar por el título de: Ingeniero en Mecatrónica, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Guayaquil, 20 de febrero de 2025

Atentamente,



Julio Alberto Alaña Pután
0930528583



Irving Airthon Moreira Calle
0952328649

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, **Jorge Bladimir Fariño Cedeño**, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **DESARROLLO DE UN SISTEMA DE MONITOREO INALÁMBRICO DE TEMPERATURA Y HUMEDAD APLICADO A UN SISTEMA DE RIEGO EN ZONAS RURALES BASADO EN ARDUINO Y COMUNICACIÓN LORA WAN**, realizado por **Julio Alberto Alaña Pután** con documento de identificación N° **0930528583** y por **Irving Airthon Moreira Calle** con documento de identificación N° **0952328649**, obteniendo como resultado final el trabajo de titulación bajo la opción **Dispositivo Tecnológico** que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Guayaquil, 20 de febrero de 2025

Atentamente,



Ing. Jorge Bladimir Fariño Cedeño
0914335484

DEDICATORIA

Dedico este trabajo a Dios, quien me ha guiado en cada paso de este camino, dándome fuerza, sabiduría y perseverancia para superar los obstáculos. Sin Su luz y apoyo constante, este logro no habría sido posible.

A mi amada esposa, Michelle, por su amor incondicional, paciencia y apoyo incansable, siempre creyendo en mí y estando a mi lado en cada etapa de este proceso. Junto a ella, mis hijos, quienes han sido mi mayor fuente de inspiración. Cada esfuerzo y sacrificio ha sido por ustedes, para demostrarles que con dedicación y perseverancia todo es posible.

Quiero agradecer a mis padres, Jacqueline y Julio, por su amor inquebrantable y por enseñarme que nunca debo rendirme. Sus palabras de aliento y su apoyo constante han sido una base sólida en este proceso. De igual forma, a mi tía Jessica, cuyas sabias enseñanzas y consejos me han permitido afrontar cada desafío con determinación y fortaleza.

No puedo dejar de reconocer el apoyo y ejemplo de mis hermanos, quienes me recordaron, con su cariño, que los sacrificios que hacemos en la vida nos permiten alcanzar nuestros sueños. Este logro también es para ustedes, con la esperanza de que se convierta en una fuente de inspiración para su propio camino.

Finalmente, dedico este trabajo a toda mi familia y seres queridos, quienes me han acompañado con su apoyo, confianza y amor en cada momento. Cada palabra de aliento y gesto de cariño han sido esenciales para llegar hasta aquí.

Julio Alberto Alaña Pután

Dedico este trabajo a mi querida familia, pilar fundamental en mi vida y en la consecución de este logro académico. A mis padres, Ana y Jair, quienes con su amor incondicional y sacrificios constantes me brindaron las herramientas necesarias para alcanzar mis metas. Su dedicación y esfuerzo han sido la luz que ha guiado mi camino, enseñándome el valor del trabajo arduo y la perseverancia. Cada consejo y enseñanza suya ha dejado una huella imborrable en mi formación, tanto personal como profesional.

A mi hermano Bryan, por ser mi compañero de vida, confidente y apoyo inquebrantable. Sus palabras de aliento y su fe en mis capacidades me motivaron a seguir adelante, incluso en los momentos más desafiantes. La complicidad y el cariño que compartimos han sido un refugio y una fuente constante de energía para superar cada obstáculo.

A mis abuelos, Lila y Emilio, cuyas historias y sabiduría me inspiraron a valorar el conocimiento y la educación. Sus vidas son testimonio de resiliencia y amor, y su legado vive en cada página de este trabajo. También a mis tíos y primos, por su constante interés y palabras de ánimo durante este proceso. Su presencia en mi vida ha enriquecido mi camino y me ha recordado la importancia de la unión familiar.

A cada miembro de mi familia, por ser el soporte emocional y la fuerza que me impulsó a no rendirme. Este logro es tanto mío como suyo, y refleja el amor y la unión que nos caracteriza.

Finalmente, a Allison, mi compañera de vida y fuente inagotable de inspiración. Tu amor y apoyo constante han sido fundamentales en la realización de este trabajo. Gracias por estar a mi lado en cada momento, brindándome ánimo y fortaleza para superar los desafíos que se presentaron en el camino. Tu paciencia y comprensión hicieron que este proceso fuera más llevadero y significativo. Este logro es tanto mío como tuyo, ya que, sin tu presencia y dedicación, no habría sido posible. Te dedico esta tesis con todo mi amor y gratitud, reconociendo el papel esencial que has desempeñado en mi vida y en mi formación profesional.

Irving Airthon Moreira Calle

AGRADECIMIENTO

Al concluir esta etapa tan importante de mi vida, quiero expresar mi más sincero agradecimiento a Dios, quien siempre estuvo a mi lado, y a todas las personas que, de una u otra manera, hicieron posible la culminación de esta tesis.

Mi agradecimiento al Ing. Fariño es profundo, ya que su orientación, conocimientos y apoyo fueron fundamentales para llevar a buen término este trabajo. Su paciencia y sus directrices me permitieron superar los desafíos y mejorar cada aspecto de esta investigación. A mis amigos Byron y Mario, por su apoyo incondicional, motivación y compañía en los momentos de duda. Su presencia hizo este proceso mucho más llevadero y me dio la fuerza necesaria para seguir adelante.

Finalmente, agradezco a todas las personas que he conocido a lo largo de este periodo, tanto en la universidad como en el trabajo, quienes, con sus enseñanzas y palabras de aliento, contribuyeron a mi crecimiento personal y profesional. A mi familia y seres queridos, por su amor, comprensión y apoyo incondicional, sin los cuales este logro no hubiera sido posible.

Julio Alberto Alaña Putan

Quiero expresar mi más sincero agradecimiento a todas las personas que hicieron posible la realización de esta tesis. Su guía, apoyo y cariño fueron fundamentales en cada etapa de este proceso.

En primer lugar, agradezco profundamente al Ing. Jorge Fariño, cuya paciencia, experiencia y valiosos consejos fueron cruciales para mi desarrollo académico. Sus observaciones y retroalimentaciones me ayudaron a alcanzar este logro.

También agradezco a la Universidad Politécnica Salesiana, cuya colaboración y recursos contribuyeron significativamente al desarrollo de este trabajo.

A mi familia, por ser mi fuente inagotable de motivación y fortaleza. Gracias por su amor incondicional, apoyo constante y por creer en mí, incluso en los momentos más difíciles. Su presencia fue mi mayor aliento a lo largo de todo este proceso.

A mis amigos, compañeros de estudio y colegas, quienes con sus palabras de ánimo, compañía y comprensión hicieron este viaje más llevadero y significativo.

Finalmente, a todas aquellas personas, tanto de manera directa como indirecta, que me brindaron su apoyo durante este proceso. Su influencia está presente en cada página de esta tesis, y por ello, les agradezco profundamente.

Irving Airthon Moreira Calle

RESUMEN

El propósito de este trabajo es diseñar e implementar un sistema automatizado de monitoreo y control de riego en zonas rurales, utilizando tecnología de comunicación inalámbrica LoRa WAN y microcontroladores. Este sistema integra sensores ambientales para medir temperatura, humedad del aire y humedad del suelo, además de relés que controlan bombas de agua en función de las condiciones detectadas o mediante comandos remotos.

El prototipo diseñado consta de una Matriz LoRa, basada en un ESP32, que gestiona la comunicación bidireccional con dos nodos remotos, implementados con LoRa. La Matriz LoRa solicita datos de los nodos, centraliza la información y permite el control manual de las bombas a través de una interfaz web desarrollada en HTML5 y CSS, accesible desde cualquier dispositivo con conexión a Internet. Por su parte, los nodos remotos recopilan datos ambientales con sensores DHT11 y FC-28 y operan de manera autónoma mediante un sistema de carga solar que utiliza un panel fotovoltaico y una batería de plomo-ácido. Todo el sistema fue programado utilizando el lenguaje basado en C++, que permitió una integración eficiente entre hardware y software.

El sistema propuesto no solo optimiza el uso del agua en sistemas de riego, sino que también garantiza un funcionamiento eficiente en zonas con acceso limitado a Internet y/o electricidad gracias a su alimentación con energía solar. Asimismo, reduce la dependencia de la intervención manual, minimiza errores humanos y mejora la productividad agrícola. Los resultados del prototipo validan su aplicabilidad en entornos rurales, consolidándose como una herramienta tecnológica sostenible y escalable para la gestión hídrica en la agricultura.

Palabras clave: Arduino, ESP32, LoRa WAN, monitoreo remoto, sistemas de riego, energía solar, sensores ambientales.

ABSTRACT

The purpose of this work is to design and implement an automated irrigation monitoring and control system in rural areas, utilizing LoRa WAN wireless communication technology and microcontrollers. This system integrates environmental sensors to measure temperature, air humidity, and soil moisture, as well as relays that control water pumps based on detected conditions or remote commands.

The designed prototype consists of a LoRa Matrix, based on an ESP32, which manages bidirectional communication with two remote nodes implemented with LoRa. The LoRa Matrix requests data from the nodes, centralizes the information, and allows manual control of the pumps through a web interface developed in HTML5 and CSS, accessible from any device with an internet connection. Meanwhile, the remote nodes collect environmental data using DHT11 and FC-28 sensors and operate autonomously through a solar charging system that uses a photovoltaic panel and a lead-acid battery. The entire system was programmed using a C++-based language, allowing for efficient hardware and software integration.

The proposed system not only optimizes water usage in irrigation systems but also ensures efficient operation in areas with limited access to the internet and/or electricity, thanks to its solar-powered supply. Additionally, it reduces the need for manual intervention, minimizes human errors, and improves agricultural productivity. The results of the prototype validate its applicability in rural environments, establishing it as a sustainable and scalable technological tool for water management in agriculture.

Keywords: Arduino, ESP32, LoRa WAN, remote monitoring, irrigation systems, solar energy, environmental sensors.

ÍNDICE

I.	Introducción	1
II.	Problema de estudio	2
III.	Justificación	3
IV.	Objetivos	4
IV-A.	Objetivo general	4
IV-B.	Objetivos específicos	4
V.	Fundamentos Teóricos	5
V-A.	Sistema de Riego Automatizados	5
V-B.	Monitoreo de riego automatizados	5
V-C.	Comunicación LoRa y LoRa WAN	6
V-D.	Uso de microcontroladores en Sistema IoT	7
VI.	Marco Metodológico	8
VI-A.	Definición de los requerimientos	8
VI-B.	Arquitectura del Sistema	8
VI-C.	Conexiones del Sistema de Monitoreo	9
VI-D.	Construcción del Sistema de Monitoreo	10
VI-D1.	Matriz LoRa	10
VI-D2.	Nodo Lora	10
VI-D3.	Implementación	11
VI-E.	Diseño para la Gestión de Alimentación Solar	13
VI-F.	Sistema de Riego	14
VI-G.	Programación de los micro-controladores	15
VI-G1.	Programación Nodo LoRa	15
VI-G2.	Programación de la Matriz LoRa	17
VII.	Resultados	22
VII-A.	Monitoreo y control de riego utilizando LoRa WAN	22
VII-A1.	Resultados de la comunicación LoRa	22
VII-A2.	Control automático y remoto de las bombas de riego	23
VII-A3.	Eficiencia energética y autonomía	23
VIII.	Cronograma	24
IX.	Presupuesto	26
X.	Conclusiones	27
XI.	Recomendaciones	27
	Referencias	28
XII.	Anexos	31

ÍNDICE DE FIGURAS

1.	Interconexión de dispositivos LoRa WAN. Fuente: Los Autores.	8
2.	Diagrama de Bloques Conexiones. Fuente: Los Autores.	9
3.	Esquema de conexiones de la matriz LoRa. Fuente: Los Autores.	10
4.	Esquema de conexiones del nodo LoRa. Fuente: Los Autores	10
5.	Placa del Nodo LoRa. Fuente: Los Autores.	11
6.	Caja metálica con componentes del Nodo LoRa. Fuente: Los Autores.	11
7.	Placa para la Matriz LoRa. Fuente: Los Autores.	12
8.	Placa para distribuir la alimentación del sistema. Fuente: Los Autores.	12
9.	Esquema de Gestión de Alimentación Solar para los Nodos. Fuente: Los Autores.	13
10.	Alimentación del sistema LoRa. Fuente: Los Autores.	13
11.	Bomba de agua utilizada en el sistema de riego. Fuente: Los Autores.	14
12.	Conexión de la bomba de agua al sistema. Fuente: Los Autores.	14
13.	Librerías y definición de Pines del Nodo. Fuente: Los Autores.	15
14.	Identificadores de la Matriz y Nodo 1.	15
15.	Configuración inicial del Nodo.	15
16.	Recibir Mensaje del Nodo.	16
17.	Procesamiento de datos del Nodo.	16
18.	Control Automático y Manual Nodo. Fuente: Los Autores	16
19.	Librerías y definición de Pines de la Matriz. Fuente: Los Autores.	17
20.	Identificadores de la Matriz y Nodo 1.	17
21.	Variables para datos del servidor web y control de bombas. Fuente: Los Autores.	18
22.	Estructura del código de configuración inicial. Fuente: Los Autores.	18
23.	Subrutina para el envío de datos mediante LoRa Matriz. Fuente: Los Autores.	19
24.	Subrutina para la recepción de datos mediante LoRa. Fuente: Los Autores.	19
25.	Subrutina para enviar datos de esclavos al servidor web. Fuente: Los Autores.	20
26.	Interfaz Web para Control de Nodos. Fuente: Los Autores.	20
27.	Monitor serial de la Matriz. Fuente: Los Autores.	20
28.	Dirección IP de la página web proporcionada por la Matriz. Fuente: Los Autores.	21
29.	Página principal del sistema de monitoreo LoRaWAN. Fuente: Los Autores.	21
30.	Interfaz web para el monitoreo y control de riego	22
31.	Control de riego desde la Web	23
32.	Cronograma de Actividades del Proyecto. Fuente: Los Autores.	25
33.	Diseño 3D soporte para el Panel Solar. Fuente: Los Autores.	31
34.	Diseño 3D adaptador para la bomba de agua. Fuente: Los Autores.	31
35.	Diseño 3D adaptador para el desfogue del agua. Fuente: Los Autores.	31
36.	Fijación y Conexión del Panel Solar. Fuente: Los Autores.	32
37.	Red de Tubería para Desagüe en los Contenedores Plásticos. Fuente: Los Autores.	32
38.	Dispensor de Agua del Prototipo. Fuente: Los Autores.	33
39.	Prueba de los Nodos LoRa en Zona Aislada. Fuente: Los Autores.	34
40.	Monitoreo de datos del sistema LoRa WAN en la web. Fuente: Los Autores.	34

ÍNDICE DE TABLAS

I.	Requerimientos del Sistema de Monitoreo y Riego LoRa WAN	8
II.	Presupuesto de materiales utilizados en el proyecto	26

I. INTRODUCCIÓN

Para la agricultura, el acceso al agua y el manejo eficiente de estos recursos hídricos son retos críticos en zonas rurales, donde las condiciones climáticas y la falta de infraestructura limitan la productividad agrícola. El presente trabajo aborda estos desafíos mediante el desarrollo de un sistema de monitoreo remoto que combina tecnología LoRa, sensores ambientales y control automatizado de riego.

Este proyecto tiene como enfoque la implementación de dos nodos, ambos basados en Arduino, que utilizan sensores para medir parámetros claves como la temperatura, humedad ambiental y humedad del suelo. Estos datos recopilados son enviados a un nodo matriz que está basado en ESP32 y actúa como servidor web, permitiendo a los usuarios visualizar la información en tiempo real y controlar el riego de manera remota.

La propuesta integra energías renovables mediante el uso de un panel solar para alimentar el sistema, promoviendo la sostenibilidad en zonas donde el acceso a la electricidad es limitado o nulo. El uso de LoRa WAN permite una comunicación eficiente y de largo alcance, superando las limitaciones de cobertura en áreas rurales.

Este documento presenta la justificación, objetivos, diseño, implementación y evaluación del sistema, destacando su impacto en la optimización del riego y su potencial como solución escalable para otras comunidades rurales.

II. PROBLEMA DE ESTUDIO

El cambio climático y el crecimiento poblacional han aumentado la presión sobre los recursos hídricos en la agricultura, intensificando la necesidad de prácticas sostenibles y eficientes. Los sistemas tradicionales de riego a menudo generan un uso ineficiente del agua, ocasionando sobreirrigación o deficiencias hídricas, lo que afecta tanto la productividad como el costo de los cultivos. Esto subraya la necesidad de soluciones basadas en tecnologías avanzadas como IoT, que pueden mejorar la eficiencia en el uso del agua en actividades agrícolas [1].

Además, la falta de conectividad en áreas rurales es otro desafío significativo, limitando el acceso a sistemas de monitoreo en tiempo real. Tecnologías como LoRa WAN han mostrado ser una opción efectiva, ya que facilitan la transmisión de datos a larga distancia mientras mantienen un bajo consumo de energía, lo cual es ideal para entornos agrícolas remotos [2]. Sin embargo, la implementación de estas tecnologías aún enfrenta obstáculos relacionados con la capacitación de los agricultores y los costos iniciales [3].

III. JUSTIFICACIÓN

La implementación de sistemas de monitoreo ambiental en la agricultura es fundamental para enfrentar los desafíos actuales de gestión del agua y mejorar la productividad agrícola. La combinación de sensores, como los de humedad y temperatura, con plataformas de análisis de datos, permite una toma de decisiones precisa y basada en información en tiempo real [4].

Tecnologías como LoRa y LoRa WAN no solo facilitan la transmisión eficiente de datos desde sensores instalados en zonas remotas, sino que también reducen el consumo de energía, garantizando sostenibilidad operativa y un impacto ambiental reducido [5]. Además, estos sistemas pueden ser alimentados mediante fuentes de energía renovable, como paneles solares, lo que refuerza su viabilidad en áreas rurales [6].

El uso de sistemas inteligentes de riego basados en IoT permite el monitoreo remoto y la distribución del agua al identificar las necesidades específicas de cada cultivo. Esto no solo mejora la calidad y cantidad de los productos agrícolas, sino que también reduce significativamente el desperdicio de recursos, contribuyendo a un manejo sostenible de los mismos [7]. La integración de estas tecnologías en la agricultura tiene el potencial de transformar prácticas tradicionales en soluciones modernas, más precisas y escalables, asegurando la seguridad alimentaria a largo plazo [8].

IV. OBJETIVOS

IV-A. Objetivo general

Desarrollar un sistema de monitoreo remoto de temperatura y humedad, para el control de riego en zonas agrícolas y rurales, utilizando tecnología LoRa WAN para la comunicación inalámbrica.

IV-B. Objetivos específicos

- Desarrollar un prototipo para el sistema de control y monitoreo remoto, tal que funcione correctamente gracias a la programación y configuración de su controlador.
- Desarrollar el enlace respectivo para la comunicación inalámbrica utilizando tecnología LoRa WAN.
- Implementar la página web donde se podrán visualizar los datos en tiempo real.

V. FUNDAMENTOS TEÓRICOS

V-A. Sistema de Riego Automatizados

Los sistemas de riego automatizados representan una innovación clave para mejorar la eficiencia en el aprovechamiento del agua en la agricultura, especialmente en regiones donde la escasez hídrica es un desafío importante. Estos sistemas integran tecnologías avanzadas, como sensores y microcontroladores, para garantizar una distribución precisa y eficiente del agua según las necesidades de los cultivos [9].

Los sensores capacitivos representan una herramienta eficaz para el monitoreo de la humedad del suelo en sistemas de riego automatizados. Estos dispositivos funcionan midiendo la constante dieléctrica del suelo, lo que permite obtener datos precisos sobre su contenido de agua y, en consecuencia, optimizar la distribución del recurso hídrico. En un estudio sobre riego por goteo en el Valle de Tarma, se implementó un sistema de sensores capacitivos que facilitó el control eficiente de la humedad del suelo, permitiendo una irrigación más precisa y reduciendo el desperdicio de agua [10]. Además, estos sensores son esenciales para disminuir el consumo excesivo de agua, especialmente en sistemas de riego por goteo, que distribuyen el agua de manera directa a las raíces de las plantas [11].

La implementación de microcontroladores, como el Arduino y el ESP32, permite procesar los datos recolectados por los sensores y controlar válvulas y bombas de agua de manera automatizada. Estos sistemas no solo aseguran la activación del riego en momentos específicos, sino que también son programables para ajustar los volúmenes de agua según las condiciones del suelo y del clima [9]. Un ejemplo exitoso es el uso del ESP8266 en sistemas de irrigación, el cual integra conectividad Wi-Fi para transmitir datos y operar remotamente, asegurando una gestión eficiente [11].

En áreas rurales con acceso limitado a la electricidad, la integración de paneles solares como fuente de energía para sistemas de riego automatizados es una solución ideal. Este enfoque no solo disminuye la dependencia de la red eléctrica, sino que también permite que los sistemas operen de manera sostenible y autónoma [10]. La combinación de tecnologías de sensores y energía solar ha demostrado ser eficaz en la optimización del uso del agua en diversas zonas agrícolas, como se evidencia en el proyecto basado en energías híbridas solares e hidrocinéticas [12].

V-B. Monitoreo de riego automatizados

El monitoreo ambiental en la agricultura es esencial para mejorar la productividad de los cultivos y asegurar un uso sostenible de los recursos naturales. Tecnologías avanzadas basadas en sensores facilitan la recolección de datos en tiempo real acerca de los parámetros como temperatura, humedad, salinidad del suelo y condiciones climáticas, integrándose con plataformas IoT para análisis y control [13]. Los sensores inteligentes, incluidos sistemas contextuales y autoalimentados, facilitan la medición de nutrientes y humedad del suelo, adaptándose a las condiciones específicas de cada entorno agrícola, lo que resulta en soluciones sostenibles y de bajo costo, especialmente útiles en zonas rurales con acceso limitado a energía [14].

Los avances en IoT han transformado el monitoreo agrícola al conectar múltiples sensores mediante plataformas en la nube, lo que permite monitorear factores críticos como temperatura, humedad y fertilización en tiempo real, optimizando decisiones fundamentadas y reduciendo el desperdicio de recursos [15]. Además, tecnologías modernas han ampliado las capacidades de monitoreo al incluir mediciones de luz, calidad del aire y compuestos orgánicos volátiles, lo que contribuye al control del entorno de cultivo y a la prevención de enfermedades [16].

Sensores de temperatura y humedad, esenciales en la agricultura de precisión, ajustan las condiciones ambientales para optimizar la calidad de los cultivos y reducir costos operativos [17]. Tecnologías como sistemas portátiles de monitoreo ambiental ofrecen soluciones prácticas para medir parámetros en tiempo real con flexibilidad y facilidad de uso [18]. Asimismo, en invernaderos, sensores integrados con microcontroladores como el ESP32 permiten controlar variables climáticas, optimizando la producción agrícola y minimizando el impacto ambiental [19].

V-C. Comunicación LoRa y LoRa WAN

La tecnología LoRa (Long Range) y el protocolo LoRa WAN se han consolidado como pilares en las comunicaciones para redes de sensores del Internet de las Cosas (IoT). Su éxito radica en la combinación de un amplio alcance, bajo consumo de energía y la capacidad de adaptación a distintos entornos. Mientras que LoRa opera en la capa física, ofreciendo una modulación robusta frente a interferencias, LoRa WAN complementa esta funcionalidad al nivel de la capa MAC, proporcionando un protocolo estándar para la gestión eficiente de redes distribuidas y de gran escala [20].

El principio de funcionamiento de LoRa se basa en el uso de la modulación por espectro ensanchado con chirp, que facilita la transmisión eficiente de datos incluso en entornos adversos. Los parámetros ajustables, como el factor de expansión (SF) y el ancho de banda (BW), ofrecen un balance entre alcance, tasa de datos y consumo energético. Esto hace que LoRa sea ideal para aplicaciones que requieren una alta autonomía y conectividad en áreas remotas [21]. Por su parte, LoRa WAN establece un modelo de red en estrella, donde los dispositivos finales se comunican con gateways, que a su vez conectan con un servidor central. Esta arquitectura es particularmente efectiva en escenarios de alta demanda de datos y control, como emergencias o redes de sensores masivas [22].

Las aplicaciones de LoRa y LoRa WAN son diversas, y su impacto abarca múltiples sectores. En la agricultura, por ejemplo, estas tecnologías permiten la automatización mediante redes locales de sensores, optimizando la monitorización de cultivos y mejorando la productividad [23]. En entornos urbanos, LoRa se ha utilizado para evaluar el impacto de la densidad de edificios en la calidad de la señal, lo que facilita la planificación de redes en ciudades densamente pobladas [24]. De igual manera, en zonas rurales o remotas, LoRa WAN ha sido clave para la gestión eficiente de sistemas como la iluminación pública en áreas con sombras de señal, destacándose por su capacidad de operar en condiciones adversas [25].

Un desafío importante en la implementación de LoRa es la interferencia electromagnética, especialmente en entornos industriales y ferroviarios. Estudios recientes han demostrado que la integración de contramedidas en la capa MAC de LoRa WAN mejora la resiliencia frente a fuentes de ruido como jammers y transitorios eléctricos [20]. Además, para superar limitaciones de cobertura, se han desarrollado arquitecturas híbridas que combinan LoRa WAN con redes malla, lo que permite adaptarse automáticamente a variaciones dinámicas del entorno y mantener la conectividad [26]. Por otro lado, en aplicaciones especializadas, como la comunicación subacuática, un sistema híbrido que combina LoRa con tecnologías ópticas ha demostrado ser una solución eficiente para superar los problemas de atenuación y reflexión de señales en interfaces agua-aire [27].

V-D. Uso de microcontroladores en Sistema IoT

Los microcontroladores son elementos clave en las aplicaciones del Internet de las Cosas (IoT) debido a su bajo costo, flexibilidad y eficiencia energética. Dispositivos como el ESP32, STM32 y la familia ATmega se utilizan ampliamente en proyectos que requieren procesamiento de datos en tiempo real y conectividad inalámbrica. Estos microcontroladores permiten el desarrollo de soluciones integradas en diversos ámbitos, como la salud, la automatización industrial, la gestión del tráfico y otros escenarios críticos en IoT [28], [29], [30].

En el ámbito de la salud, el ESP32 ha demostrado ser una herramienta poderosa gracias a su capacidad de recopilar y transmitir datos fisiológicos en tiempo real. Estos sistemas, que emplean sensores avanzados, pueden monitorear parámetros como la frecuencia cardíaca, la temperatura y los niveles de oxígeno en sangre. La información recopilada se envía a plataformas en la nube para su análisis y almacenamiento, facilitando la detección temprana de complicaciones y mejorando la gestión de enfermedades crónicas. Por ejemplo, un sistema de monitoreo basado en ESP32 alcanzó precisiones promedio de 97.72 % en mediciones críticas [28].

La versatilidad de los microcontroladores también los hace ideales para dispositivos multipropósito. Un ejemplo destacado es un sistema basado en ESP32 que combina sensores de presión atmosférica, humedad y temperatura, con la capacidad de adaptarse a funciones como el control de drones o sistemas domóticos. Esta modularidad no solo permite personalizar el dispositivo según las necesidades, sino que también ofrece oportunidades de aprendizaje en el campo de los sistemas embebidos [31].

En la automatización inteligente, los microcontroladores permiten optimizar recursos y reducir el consumo energético. En proyectos como sistemas de iluminación pública, dispositivos como el STM32 y el ESP32 trabajan en conjunto para controlar el brillo de las luces según las condiciones ambientales. Sensores de luz y movimiento integrados en estos sistemas aseguran que la iluminación se ajuste automáticamente, demostrando la capacidad de estas tecnologías para habilitar soluciones sostenibles y escalables [29].

En el sector industrial y la gestión del tráfico, microcontroladores como el ATmega328p y el ATmega2560 son esenciales. Estos dispositivos se utilizan en sistemas IoT para automatizar procesos industriales y controlar el flujo de tráfico en áreas urbanas. Integran sensores inteligentes y comunicaciones inalámbricas para garantizar operaciones seguras y eficientes, destacándose en la administración de cargas múltiples y en la regulación de zonas de alta densidad vehicular [32], [33].

Por último, el ESP32 ha sido objeto de estudios comparativos para evaluar su efectividad frente a otros microcontroladores en términos de costo y rendimiento. Los resultados han demostrado que este dispositivo es una alternativa económica y eficiente para aplicaciones que requieren conectividad constante y procesamiento en tiempo real. Esto refuerza su posición como una herramienta ideal para proyectos embebidos de IoT [30].

VI. MARCO METODOLÓGICO

VI-A. Definición de los requerimientos

A continuación, se plantean los requerimientos que debe cumplir el sistema de monitoreo y riego basado en tecnología LoRa WAN:

Tabla I
REQUERIMIENTOS DEL SISTEMA DE MONITOREO Y RIEGO LoRa WAN

Funcionalidad	Descripción
Mediciones ambientales	Monitorear los datos ambientales mediante sensores confiables como DHT11 y FC-28 para evaluar las condiciones climáticas y del terreno.
Comunicación inalámbrica	Garantizar una transmisión de datos eficiente y confiable entre la Matriz LoRa (estación base) y los Nodos LoRa mediante la tecnología LoRa WAN, asegurando un alcance adecuado en zonas rurales.
Control de bombas	Gestionar de manera automática o manual el encendido y apagado de bombas de agua según los niveles de humedad detectados o mediante comandos desde una interfaz web.
Alimentación autónoma	Asegurar la operación continua del sistema en entornos remotos mediante el uso de paneles solares y baterías de plomo-ácido con gestión de carga optimizada.
Visualización remota	Proporcionar acceso en tiempo real a los datos recolectados y posibilitar la gestión remota del sistema de riego mediante una interfaz web accesible desde cualquier dispositivo con conexión a internet.

VI-B. Arquitectura del Sistema

Se puede observar la arquitectura del sistema propuesto en la figura 1. En esta, la matriz central solicita periódicamente datos a los nodos, los cuales recopilan información ambiental mediante los sensores y responden enviando los datos recolectados a la matriz. La matriz central actúa como un servidor web el cual mediante conexión a Internet permite la visualización de los datos y la gestión remota de los nodos.

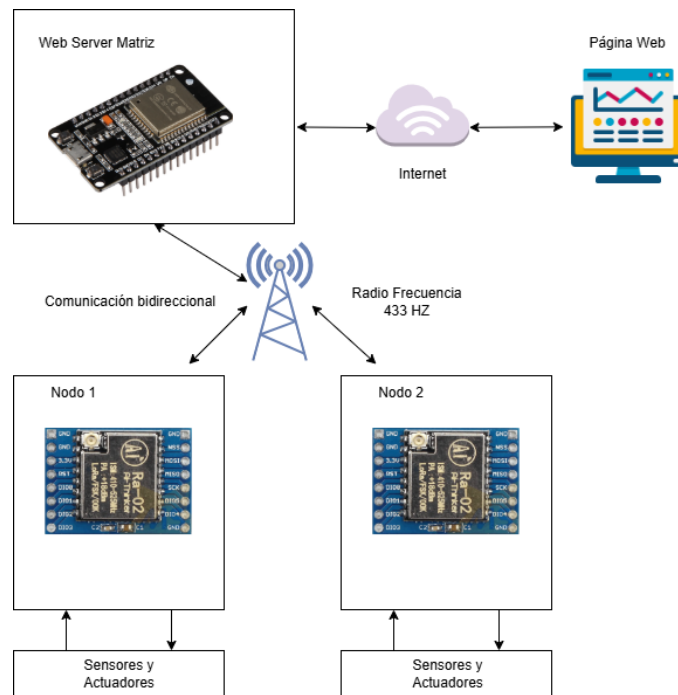


Figura 1. Interconexión de dispositivos LoRa WAN. Fuente: Los Autores.

VI-C. Conexiones del Sistema de Monitoreo

Se muestra el diagrama de bloques del sistema propuesto en la figura 2 , que ilustra las conexiones físicas y los componentes principales utilizados en el diseño. Este diagrama permite visualizar cómo se integran los módulos de hardware para implementar el sistema de monitoreo y riego LoRa WAN.

- **Nodos LoRa:** Cada nodo incluye un microcontrolador Arduino Uno, sensores ambientales (DHT11 y FC-28), un módulo LoRa SX1278, y un sistema de alimentación autónomo compuesto por un panel solar, un controlador de carga MPPT, y baterías.
- **Matriz LoRa:** Actúa como estación base utilizando un microcontrolador ESP-WROOM-32, gestionando la recepción y envío de datos desde los nodos mediante comunicación LoRa WAN.
- **DHT11:** Sensor de temperatura y humedad que proporciona datos sobre las condiciones ambientales, permitiendo el monitoreo efectivo del clima en el área de riego.
- **FC-28:** Es un sensor de humedad del suelo, el cual ayudara a determinar cuándo es necesario activar el riego.
- **LM2596:** Es un regulador de voltaje ajustable que permite convertir voltajes de entrada más altos a un voltaje de salida más bajo y estable, asegurando que los componentes del sistema reciban la alimentación adecuada y protegiendo así su funcionamiento, en este caso regulada a 9V para los 2 Nodos.
- **Módulo LoRa SX1278:** Es el componente de comunicación inalámbrica encargado de transmitir y recibir datos entre los nodos y la matriz mediante tecnología LoRa WAN, operando en la banda de 433 MHz.
- **Panel solar:** Genera la energía necesaria para el sistema.
- **Controlador de carga MPPT:** Gestiona la carga solar en la batería.
- **Baterías:** Almacenan energía para asegurar el funcionamiento continuo en zonas remotas.

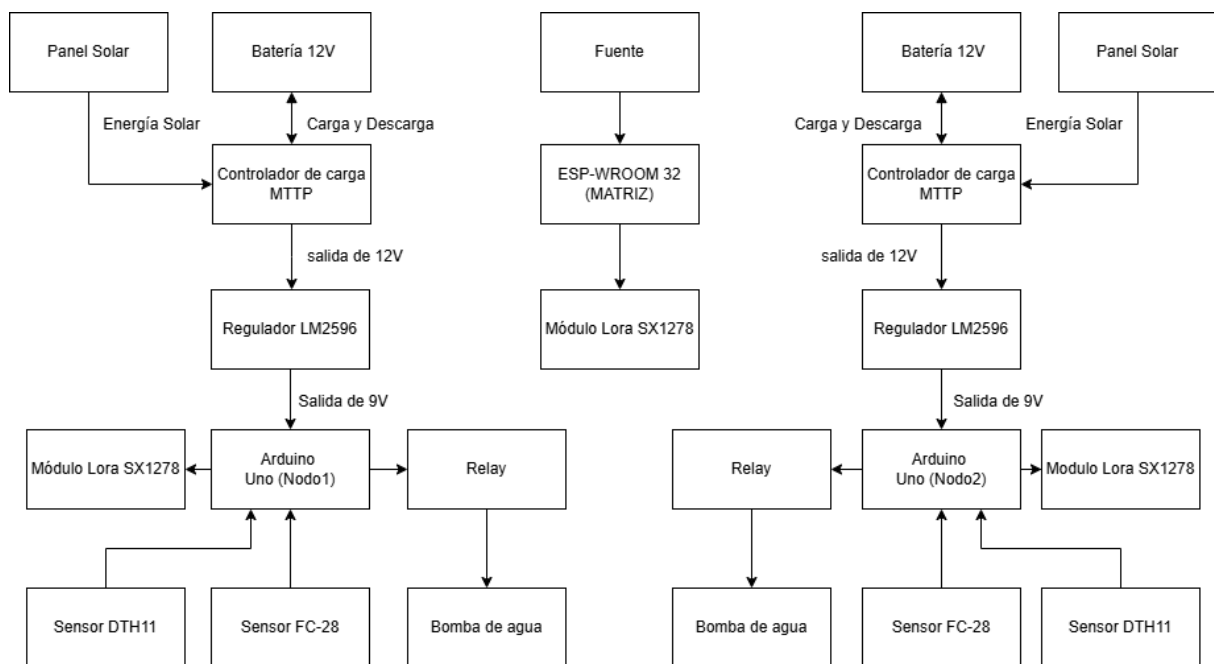


Figura 2. Diagrama de Bloques Conexiones. Fuente: Los Autores.

VI-D. Construcción del Sistema de Monitoreo

La construcción del sistema de monitoreo se realizó siguiendo el diagrama de bloques presentado anteriormente, el cual define claramente los componentes principales y sus interacciones.

VI-D1. Matriz LoRa: Se presentan las conexiones eléctricas de la matriz LoRa en la figura 3, compuesta por un módulo ESP32-WROOM-32 y un módulo de comunicación LoRa SX1278. Este esquema detalla cómo se integran ambos dispositivos para recibir y enviar datos.

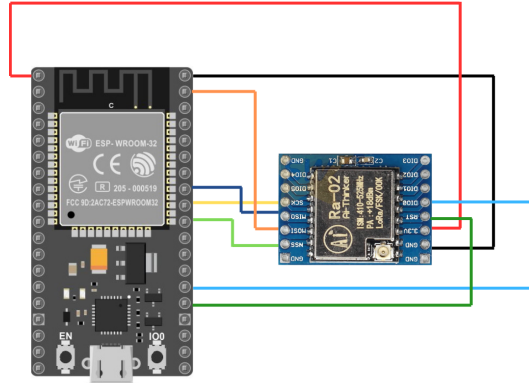


Figura 3. Esquema de conexiones de la matriz LoRa. Fuente: Los Autores.

VI-D2. Nodo Lora: Para cada nodo, se utilizó el microcontrolador Arduino Uno, el sensor FC-28 y el sensor DHT11 para registrar la humedad del suelo, humedad del aire y la temperatura. Estos sensores permiten controlar un relé que activa automáticamente la bomba de agua cuando se cumplen ciertas condiciones que indican que el suelo necesita riego. Para la comunicación entre el nodo y la matriz, se emplea el módulo LoRa SX1278. En la figura 4 se presentan las conexiones de todos los componentes.

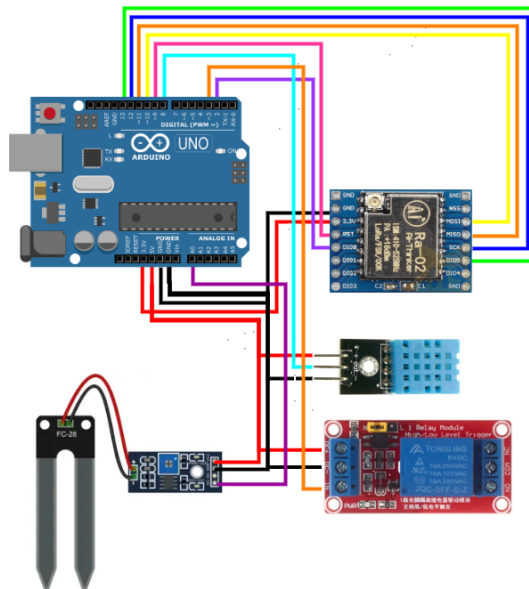


Figura 4. Esquema de conexiones del nodo LoRa. Fuente: Los Autores

VI-D3. Implementación: Después de diseñar los esquemas de conexiones eléctricas, se procedió a implementar físicamente el sistema de monitoreo.

Se fabricó una placa PCB personalizada para el Arduino basada en los esquemas diseñados. Esta placa cuenta con un socket para el módulo SX1278, una bornera para la alimentación y tres borneras adicionales. Estas permiten conectar el módulo relé, encargado de controlar la bomba de agua, y los dos sensores ambientales (FC-28 y DHT11), como podemos ver en la figura 5.

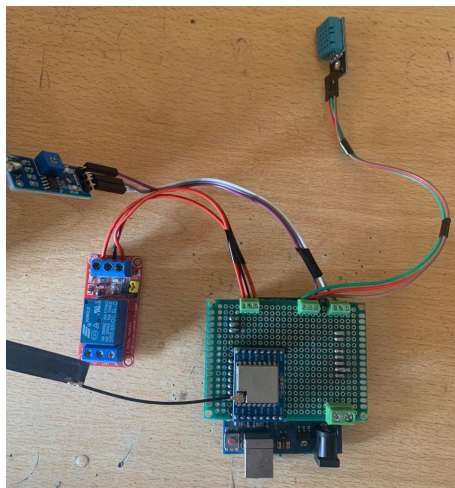


Figura 5. Placa del Nodo LoRa. Fuente: Los Autores.

Para proteger los componentes, se diseñó una caja metálica con orificios estratégicos que alberga todos los elementos del Nodo LoRa, dejando expuestos únicamente los sensores ambientales, la alimentación del microcontrolador a través de un regulador LM2596 calibrado a 9V (de alta eficiencia), la bomba de agua de 12V conectada al relé y la antena del módulo LoRa SX1278 para la comunicación. La figura 6 muestra el diseño de la caja.

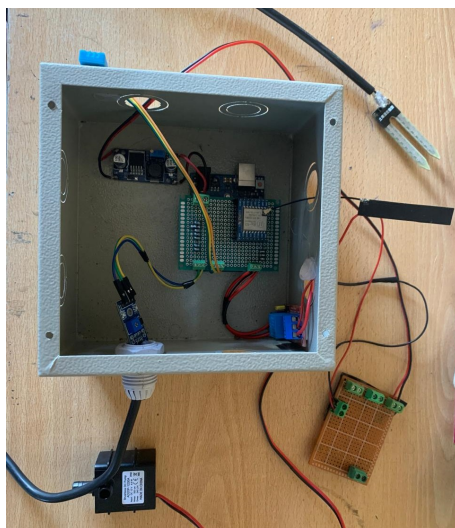


Figura 6. Caja metálica con componentes del Nodo LoRa. Fuente: Los Autores.

Después de finalizar el nodo, se procede a diseñar y construir una placa PCB específica para la matriz, como se puede ver en la figura 7. Esta placa será responsable de monitorear y gestionar la comunicación con los nodos, contando con acceso a internet. A diferencia de los nodos, no requiere una batería de alimentación, ya que estará conectada a una fuente de energía constante.

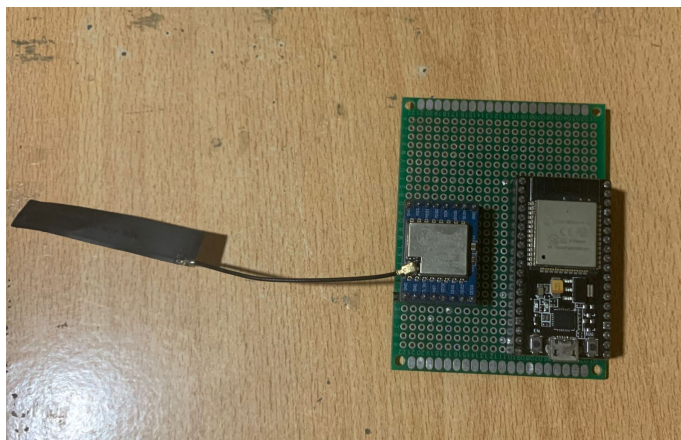


Figura 7. Placa para la Matriz LoRa. Fuente: Los Autores.

Para alimentar los nodos y los relés que controlan el encendido y apagado de las bombas del prototipo, se diseñó un circuito de distribución de energía. Este circuito tiene una entrada principal conectada a la batería y cuatro salidas que suministran energía a los diferentes elementos del sistema. La configuración de este circuito se puede visualizar en la figura 8.

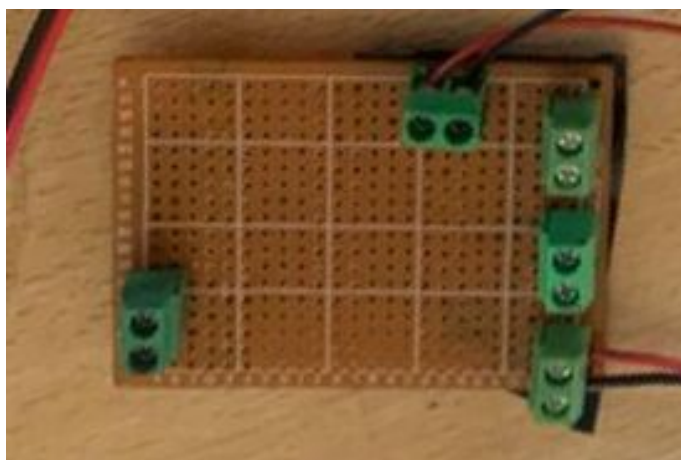


Figura 8. Placa para distribuir la alimentación del sistema. Fuente: Los Autores.

VI-E. Diseño para la Gestión de Alimentación Solar

Para garantizar la independencia energética de los nodos, se diseñó un esquema de gestión y distribución de alimentación mediante energía solar. Este esquema incluye las conexiones y componentes necesarios para optimizar el suministro de energía, como paneles solares, controladores de carga y baterías, asegurando el funcionamiento continuo de los nodos. El diseño se puede visualizar en la figura 9.

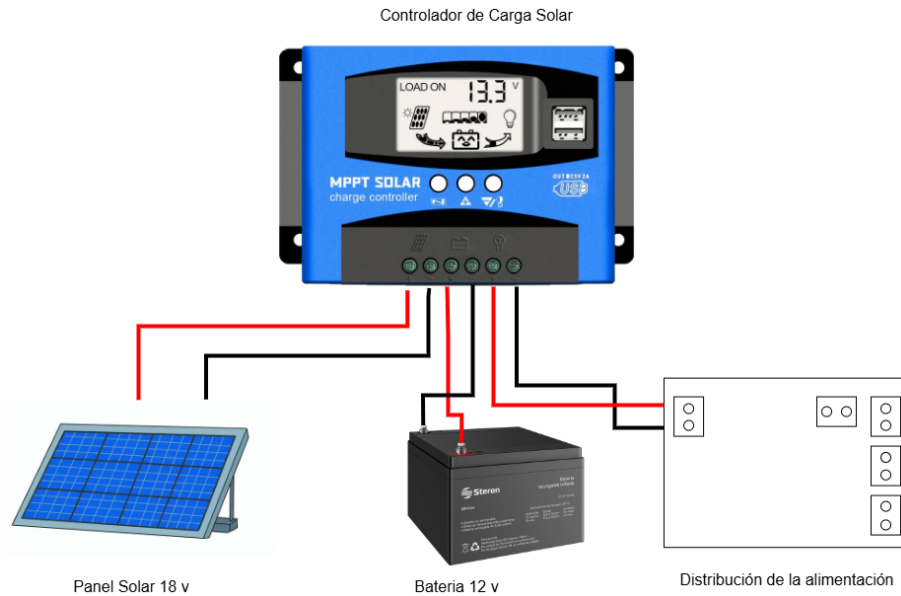


Figura 9. Esquema de Gestión de Alimentación Solar para los Nodos. Fuente: Los Autores.

Las conexiones se realizaron de acuerdo con el diseño del esquema de alimentación solar. La salida del controlador de carga solar se conectó al circuito de distribución de energía, el cual proporciona alimentación a todos los componentes del sistema, incluidos los nodos, relés y demás elementos esenciales. La configuración completa puede observarse en la figura 10.



Figura 10. Alimentación del sistema LoRa. Fuente: Los Autores.

VI-F. Sistema de Riego

El proyecto está diseñado para ser adaptable a diferentes sistemas de riego, ya que los sensores se encargan de activar los relés. Esto permite integrar diferentes tipos de componentes electrónicos para el control del agua, como electroválvulas o bombas de agua. Por ejemplo, las electroválvulas operan utilizando presión de agua, mientras que las bombas extraen agua desde un pozo o depósito. Para la implementación y las pruebas del sistema, se utilizó una bomba de agua modelo 240L/H DC 12V 4.2W, ilustrada en la figura 11.

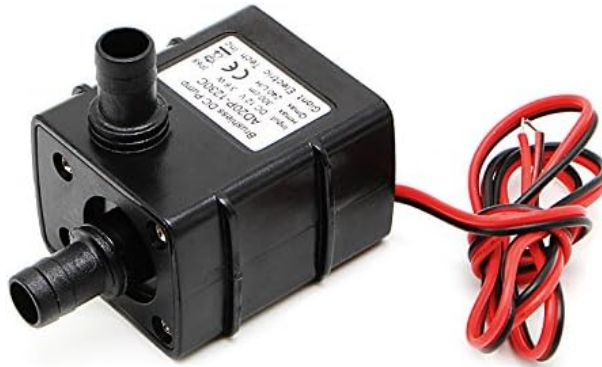


Figura 11. Bomba de agua utilizada en el sistema de riego. Fuente: Los Autores.

La implementación del sistema de riego consistió en conectar un tubo de vinil a la salida de la bomba de agua, al cual se adaptó un tubo de PVC con pequeños orificios, permitiendo así la dispersión del agua sobre el área de riego. El recipiente cuenta con un sistema de desagüe para evitar que el agua se acumule de forma no controlada. Esta configuración garantiza un flujo continuo y eficiente, como podemos ver en la figura 12.



Figura 12. Conexión de la bomba de agua al sistema. Fuente: Los Autores.

VI-G. Programación de los micro-controladores

La programación de los microcontroladores es una parte fundamental de este proyecto. En este caso, se utilizará Arduino Uno como nodos, que estarán encargados de realizar las lecturas ambientales con los sensores conectados a cada uno, como el sensor DHT11 para la humedad y temperatura del aire, y para la humedad del suelo el sensor FC-28. Los nodos, además de recolectar los datos de los sensores, se encargan de enviar la información hacia la matriz LoRa. Esta matriz actuará como un servidor web, solicitando constantemente los datos a los nodos y presentándolos en una página web, donde también se ofrecerá la opción de controlar manualmente las bombas de riego. A continuación, se detallará la programación del nodo.

VI-G1. Programación Nodo LoRa: Se incluyeron las librerías necesarias para manejar la comunicación LoRa y los sensores. También se definieron los pines de entrada/salida para los sensores y el relé de control de la bomba, como se muestra en la figura 13.

```
1  //----- Inclusión de Librerías.
2  #include <SPI.h> //para la comunicación SPI, que es esencial para la transmisión de datos
3  #include <LoRa.h> //para gestionar la comunicación LoRa entre el nodo y la matriz LoRa.
4  #include <DHT.h> //para la lectura de los datos del sensor DHT11 (temperatura y humedad)
5  //-----
6
7  //----- Definición de los Pines y Tipo de Sensor DHT11.
8  #define PIN_DHT 8 //Pin digital para la lectura de DHT11
9  #define TIPO_DHT DHT11 //Tipo de Sensor
10 #define PIN_SENSOR_SUELO A0 // Pin analógico al que está conectado el FC-28
11 #define PIN_BOMBA 3 // Pin que se usara para activar el relay conectado a la bomba.
12 #define UMBRAL_HUMEDAD_SUELO 30 // Umbral de humedad (en porcentaje) para activar la bomba automáticamente
13 //-----
```

Figura 13. Librerías y definición de Pines del Nodo. Fuente: Los Autores.

Se asignaron identificadores únicos a la matriz y a cada nodo para garantizar una comunicación diferenciada entre dispositivos, como se ilustra en la figura 14.

```
24 // Configuración de direcciones de LoRa
25 byte DireccionLocal = 0x02; // Dirección de este dispositivo (esclavo 1).
26 byte DireccionMaestro = 0x01; // Dirección del dispositivo maestro.
```

Figura 14. Identificadores de la Matriz y Nodo 1.

Dentro del bloque `setup`, se configuró la comunicación LoRa, estableciendo un ancho de banda de 433 MHz. La figura 15 muestra esta configuración.

```
150 //----- Configuración Inicial
151 void setup() {
152     Serial.begin(115200);
153     pinMode(PIN_BOMBA, OUTPUT);
154     reiniciarLoRa();
155     dht11.begin();
156 }
```

Figura 15. Configuración inicial del Nodo.

El nodo fue programado para detectar solicitudes de datos enviadas por la matriz. Si se recibe una solicitud, el nodo realiza las lecturas de los sensores, procesa los datos y los envía a la matriz. Además, si la solicitud incluye una petición para encender la bomba, el nodo activará el control de la bomba de acuerdo con la petición. Esta secuencia de pasos se detalla en la figura 16

```

88     if (Receptor != DireccionLocal) {
89         Serial.println("Este mensaje no es para mí.");
90         return;
91     } else {
92         Serial.println("Mensaje recibido: " + MensajeEntrante);
93         if (ModoMaestro == 1) procesarDatos();
94         if (ModoMaestro == 2) procesarControlBomba();
95     }
96 }

```

Figura 16. Recibir Mensaje del Nodo.

Se creó una función para procesar los datos recopilados por los sensores del nodo, estos datos se empaquetan y se envían a la matriz mediante el método `enviarMensaje()`. Este método incluye información sobre los sensores y el estado de la bomba, así como los identificadores tanto del nodo como de la matriz. La figura 17 muestra cómo se realiza el procesamiento y envío de los datos.

```

// _____ Subrutina para procesar datos a enviar
void procesarDatos() {
    byte EstadoBomba = digitalRead(PIN_BOMBA);

    MensajeSalida = Estado_Lectura_DHT11 + "," + String(HumedadAire) + "," + String(Temperatura) + "," + String(HumedadSuelo) + "," + String(EstadoBomba);

    Serial.println("Enviando mensaje al maestro: " + MensajeSalida);
    enviarMensaje(MensajeSalida, DireccionMaestro);
}

```

Figura 17. Procesamiento de datos del Nodo.

Finalmente se implementó en el nodo un mecanismo que permite activar la bomba automáticamente cuando el suelo está seco. Además, se agregó una función de control manual para activar la bomba durante 10 segundos, ignorando la condición del suelo, según lo mostrado en la figura 18.

```

185     // Control automático de la bomba por humedad del suelo
186     if (!BombaActivadaPorMatriz) {
187         if (HumedadSuelo <= UMBRAL_HUMEDAD_SUELO) {
188             digitalWrite(PIN_BOMBA, HIGH); // Activa la bomba automáticamente.
189         } else {
190             digitalWrite(PIN_BOMBA, LOW); // Apaga la bomba si no es necesario.
191         }
192     } else {
193         // Control manual desde la matriz (10 segundos)
194         if (millis() - TiempoInicioBomba >= DuracionBombaMatriz) {
195             BombaActivadaPorMatriz = false; // Regresa al control automático
196         }
197     }

```

Figura 18. Control Automático y Manual Nodo. Fuente: Los Autores

VI-G2. *Programación de la Matriz LoRa:* La matriz LoRa actúa como un servidor web para visualizar y controlar los nodos. Solicita datos a los nodos, los procesa y los muestra en una página HTML. Para ello, se utilizara librerías que habilitan la comunicación con el módulo LoRa, la creación de un servidor web asíncrono en ESP32, el manejo de datos JSON y la integración de la interfaz HTML del sistema. La figura 19 muestra las librerías empleadas para este propósito.

```
1  //----- Librerías incluidas
2  #include <SPI.h> // Comunicación SPI para el módulo LoRa.
3  #include <LoRa.h> // Control del módulo LoRa (SX1278).
4  #include <AsyncTCP.h> // Servidores y clientes TCP asíncronos (ESP32).
5  #include <ESPAsyncWebServer.h> // Servidor web asíncrono en ESP32.
6  #include <Arduino_JSON.h> // Manejo de datos JSON.
7  #include "PageIndex.h" // Archivo HTML para la página principal.
```

Figura 19. Librerías y definición de Pines de la Matriz. Fuente: Los Autores.

Se procede a definir las credenciales de red necesarias para la conexión Wi-Fi de la matriz, las variables para almacenar los datos enviados y recibidos, y la configuración de direcciones y modos de transmisión para la comunicación LoRa. Se establecieron direcciones únicas para identificar al maestro y a los nodos esclavos, así como modos específicos para solicitar datos de los sensores o controlar las bombas conectadas a los nodos. La figura 20 muestra su estructura.

```
14 //----- Credenciales de red
15 const char* ssid = "miRed";
16 const char* password = "miPassword";
17
18 //----- Variables para datos entrantes y salientes
19 String Incoming = "";
20 String Message = "";
21
22 //----- Configuración de transmisión de datos LoRa
23 byte LocalAddress = 0x01; // Dirección del maestro
24 byte Destination_Arduino_Slave_1 = 0x02; // Dirección de destino para el Esclavo 1
25 byte Destination_Arduino_Slave_2 = 0x03; // Dirección de destino para el Esclavo 2
26 const byte get_Data_Mode = 1; // Modo para obtener datos de los sensores y estados de los Bombas
27 const byte Bomba_Control_Mode = 2; // Modo para controlar las Bombas en el esclavo
28
```

Figura 20. Identificadores de la Matriz y Nodo 1.

Se definieron las variables necesarias para la gestión de datos del servidor web, lo que permite el control de las bombas conectadas a los nodos esclavos. Además, se inicializó el servidor web en el puerto 80 y se definió una fuente de eventos para la comunicación en tiempo real, junto con un objeto JSON para almacenar los datos recibidos. La figura 21 muestra los elementos configurados.

```

47 //----- Variables para datos del servidor web para controlar las Bombas
48 String Slave_Number = "";
49 String Bomba_Number = "";
50 String Bomba_Value = "";
51
52 byte Slv = 0; // Contador para esclavos
53 byte slave_Address; // Dirección del esclavo en uso
54 byte count_to_Rst_LORA = 0; // Contador para reiniciar LoRa
55
56 bool finished_Receiving_Message = false; // Indicador de finalización de recepción de mensajes
57 bool finished_Sending_Message = false; // Indicador de finalización de envío de mensajes
58 bool send_Control_Bomba = false; // Activador para envío de comandos de control Bomba
59
60 JSONVar JSON_All_Data_Received; // Inicializar JSONVar para datos recibidos
61 AsyncWebServer server(80); // Crear servidor web en puerto 80
62 AsyncEventSource events("/events"); // Crear una fuente de eventos en /events

```

Figura 21. Variables para datos del servidor web y control de bombas. Fuente: Los Autores.

Dentro del setup, se configuró el ESP32 en modo estación (STA), habilitando la conexión Wi-Fi y la comunicación LoRa. Además, se programaron eventos asíncronos para manejar solicitudes GET desde la página web, como se detalla en la figura 22.

```

224 void setup() {
225 // Configuración inicial que se ejecuta una vez:
226 Serial.begin(115200); // Inicia la comunicación serial a 115200 baud
227
228 //----- Limpia los valores iniciales de los arreglos Temp, Humd y SoilMoisture
229 for (byte i = 0; i < 2; i++) {
230   Humd[i] = 0; // Humedad del aire
231   Temp[i] = 0.00; // Temperatura
232   SoilMoisture[i] = 0; // Humedad del suelo
233 }
234
235 //----- Configura el modo WiFi en "STA" (Estación)
236 Serial.println();
237 Serial.println("-----");
238 Serial.println("Modo WIFI : STA");
239 WiFi.mode(WIFI_STA);
240 Serial.println("-----");
241
242 delay(100); // Pequeña espera para estabilidad
243
244 //----- Conectar a la red WiFi en modo STA
245 Serial.println("-----");
246 Serial.println("Conectando en modo WIFI STA");
247 Serial.print("Conectando a ");
248 Serial.println(ssid);
249 WiFi.begin(ssid, password); // Inicia la conexión WiFi

```

Figura 22. Estructura del código de configuración inicial. Fuente: Los Autores.

Se implementaron subrutinas para enviar mensajes a través del módulo LoRa (Ra-02), permitiendo la transmisión de datos entre nodos. Recibe como parámetros el mensaje a enviar, la dirección de destino y el modo de envío (para obtener datos o controlar bombas). La subrutina prepara y envía un paquete LoRa que incluye la dirección de destino, la dirección del remitente, la longitud del mensaje y el tipo de modo. Después de enviar el mensaje, establece la variable (`finishedSendingMessage`) como `true` para indicar que la transmisión ha finalizado. La figura 23 muestra el flujo de la subrutina.

```

64 // Subrutinas para enviar datos (LoRa Ra-02)
65 void sendMessage(String Outgoing, byte Destination, byte SendMode) {
66     finished_Sending_Message = false;
67
68     Serial.println();
69     Serial.println("Intento hacia: 0x" + String(Destination, HEX));
70     Serial.print("Modo: ");
71     if (SendMode == 1) Serial.println("Obtener datos");
72     if (SendMode == 2) Serial.println("Controlar Bomba");
73     Serial.println("Mensaje: " + Outgoing);
74
75     LoRa.beginPacket(); // Inicia paquete
76     LoRa.write(Destination); // Dirección destino
77     LoRa.write(LocalAddress); // Dirección del remitente
78     LoRa.write(Outgoing.length()); // Longitud del mensaje
79     LoRa.write(SendMode); // Modo de envío
80     LoRa.print(Outgoing); // Datos a enviar
81     LoRa.endPacket(); // Envía paquete
82
83     finished_Sending_Message = true;
84 }

```

Figura 23. Subrutina para el envío de datos mediante LoRa Matriz. Fuente: Los Autores.

Luego, se implementó una subrutina para recibir datos mediante el módulo LoRa (Ra-02). Este verificará si hay un paquete disponible y, de ser así, leerá su cabecera, que incluye la dirección del destinatario, la dirección del remitente y la longitud del mensaje. Posteriormente, recopilará los datos entrantes y verificará la consistencia de la longitud para evitar errores. Si el mensaje está destinado a este dispositivo (según la dirección local), se imprimirán los detalles del remitente y el mensaje recibido, se actualizará la dirección del remitente como el esclavo activo, y se llamará a otra subrutina para procesar los datos. La figura 24 ilustra el flujo de esta subrutina.

```

87 // Subrutinas para recibir datos (LoRa Ra-02)
88 void onReceive(int packetSize) {
89     if (packetSize == 0) return; // No hay paquete
90
91     finished_Receiving_Message = false;
92
93     //----- Leer cabecera del paquete
94     int recipient = LoRa.read(); // Dirección del destinatario
95     byte sender = LoRa.read(); // Dirección del remitente
96     byte incomingLength = LoRa.read(); // Longitud del mensaje entrante
97
98     Incoming = ""; // Limpia variable de datos entrantes
99
100     //----- Obtener todos los datos entrantes
101     while (LoRa.available()) {
102         Incoming += (char)LoRa.read();
103     }
104
105     // Reinicia el contador para reiniciar LoRa si se recibe un mensaje
106     count_to_Rst_LORA = 0;
107 }

```

Figura 24. Subrutina para la recepción de datos mediante LoRa. Fuente: Los Autores.

Para enviar al servidor web los datos recopilados de los nodos en formato JSON se creó el siguiente método, ver la figura 25. Éste asigna valores como la identificación del esclavo, estado de lectura del DHT11, humedad del aire, temperatura, humedad del suelo y estado de la bomba a un objeto JSON. Luego, convierte el objeto en una cadena JSON y lo transmite al servidor mediante un evento web asíncrono identificado como ‘allDataJSON’. Esto permite actualizar la información de los sensores y estados en tiempo real en la interfaz web.

```

176 // Subrutina para enviar datos recibidos de los esclavos al servidor web.
177 void Send_Data_to_WS(char ID_Slave[5], byte Slave) {
178     JSON_All_Data_Received["ID_Slave"] = ID_Slave;
179     JSON_All_Data_Received["StatusReadDHT11"] = receive_Status_Read_DHT11;
180     JSON_All_Data_Received["Humd"] = Humd[Slave-1];
181     JSON_All_Data_Received["Temp"] = Temp[Slave-1];
182     JSON_All_Data_Received["SoilMoisture"] = SoilMoisture[Slave-1]; // Nuevo valor de humedad del suelo
183     JSON_All_Data_Received["LED1"] = Bomba_State_bool;
184
185     String jsonString_Send_All_Data_received = JSON.stringify(JSON_All_Data_Received);
186     events.send(jsonString_Send_All_Data_received.c_str(), "allDataJSON", millis());
187 }

```

Figura 25. Subrutina para enviar datos de esclavos al servidor web. Fuente: Los Autores.

La página web fue desarrollada utilizando HTML y CSS, almacenada en un archivo denominado "PageIndex.h", diseñado para ser compatible con microcontroladores. Este archivo permite que el ESP32 WROOM-32 levante un servidor web, proporcionando una interfaz gráfica para visualizar los datos de los nodos y gestionar su control. La figura 26 muestra cómo se presenta esta funcionalidad en la interfaz.

```

Matrix_WebServer.ino PageIndex.h
1 const char MAIN_page[] PROGMEM = R"=====(
2 <!DOCTYPE HTML>
3 <html lang="es">
4   <head>
5     <meta charset="UTF-8">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7     <title>Monitoreo de Riego LoraWAN</title>
8     <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.7.2/css/all.css" integrity="sha384-fnmOqcB77117186J8L64J99nc0siRcVclpfhqzw05hv1rkTh860rWbd848648" crossorigin="anonymous">
9     <link rel="icon" href="data:;"/>
10    <style>
11      html {font-family: Arial; display: inline-block; text-align: center;}
12      p {font-size: 1.2rem;}
13      h4 {font-size: 0.8rem;}
14      body {margin: 0;}
15      .topnav {overflow: hidden; background-color: #00000f; color: white; font-size: 1.2rem;}
16      .content {padding: 5px;}
17      .card {background-color: white; box-shadow: 0px 0px 10px 1px rgba(140,140,140,.5); border: 1px solid #00000f; border-radius: 15px;}
18      .card.header {background-color: #00000f; color: white; border-bottom-right-radius: 0px; border-bottom-left-radius: 0px; border-top-right-radius: 12px; border-top-left-radius: 12px;}
19      .cards {max-width: 700px; margin: 0 auto; display: grid; grid-gap: 2rem; grid-template-columns: repeat(auto-fit, minmax(300px, 1fr));}
20      .reading {font-size: 1.3rem;}
21      .packet {color: #bebebe;}

```

Figura 26. Interfaz Web para Control de Nodos. Fuente: Los Autores.

Se realizó una prueba de comunicación en la que se pudo observar, haciendo uso del monitor serial de Arduino, el flujo de datos recibidos desde uno de los nodos. La figura 27 muestra esta visualización.

```

Output Serial Monitor x
Message (Enter to send message to 'ESP32-WROOM-DA Module' on 'COM12')

Modo: Obtener datos
Mensaje:

Intento hacia: 0x2
Modo: Obtener datos
Mensaje:

Recibido de: 0x2
Mensaje: s,68,30.70,0,1

```

Figura 27. Monitor serial de la Matriz. Fuente: Los Autores.

Durante el proceso, se utilizó el monitor serial para obtener la dirección IP asignada al ESP32, permitiendo el acceso a la página web alojada en el servidor. Por ejemplo, la dirección ‘http://192.168.100.123’ fue mostrada, como se puede apreciar en la figura 28.

```
Iniciando el servidor.  
  
Reiniciar LoRa...  
Iniciando LoRa...  
LoRa iniciado con éxito.  
  
-----  
Dirección IP del ESP32: 192.168.100.123  
  
Visita la dirección IP anterior en tu navegador para abrir la página principal.
```

Figura 28. Dirección IP de la página web proporcionada por la Matriz. Fuente: Los Autores.

Finalmente, se accedió a la página principal del sistema de monitoreo, donde se visualizaron los datos de los dos nodos. Estos incluyen la temperatura, la humedad del aire y la humedad del suelo. Además, se integró un botón para el control manual de la bomba. En la parte inferior de la página, se mostraron otros detalles, como el tiempo de recepción de datos, el estado de los sensores y el estado de la bomba, como se observa en la figura 29.



Figura 29. Página principal del sistema de monitoreo LoRaWAN. Fuente: Los Autores.

VII. RESULTADOS

VII-A. Monitoreo y control de riego utilizando LoRa WAN

El prototipo implementado con la tecnología LoRa WAN, en combinación con sensores ambientales FC-28 y DTH11, funcionó correctamente durante las pruebas realizadas en distintas condiciones. La matriz LoRa, equipada con un ESP32 y módulos LoRa SX1278, pudo enviar solicitudes a los nodos distribuidos en zonas rurales, los cuales respondieron con los datos de los sensores de manera eficiente. La comunicación inalámbrica a través de LoRa permitió la transmisión de datos a distancias significativas, sin la necesidad de infraestructura de red adicional.

VII-A1. Resultados de la comunicación LoRa: Los módulos LoRa en ambos nodos y la matriz permitieron un monitoreo remoto efectivo. Los datos de temperatura, humedad del aire y suelo, fueron recibidos por la matriz LoRa de manera fiable. La comunicación se mantuvo estable incluso a distancias superiores a 1 km, lo que demuestra la eficacia del protocolo LoRa en ambientes rurales, donde la cobertura de red celular es limitada o nula. La matriz, como servidor web, actualizó la información en tiempo real y facilitó el manejo remoto de las bombas de riego..

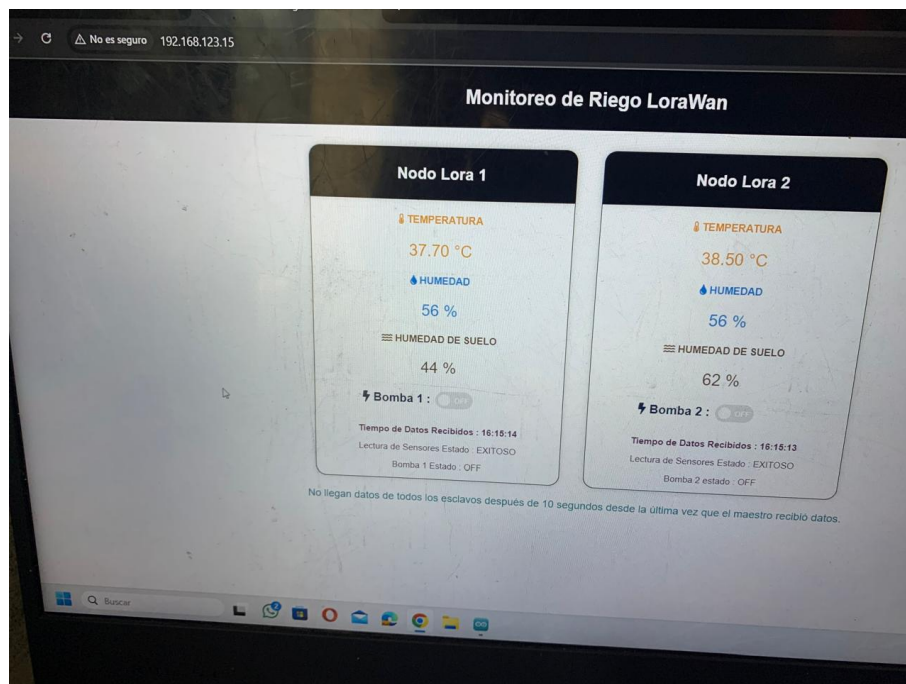


Figura 30. Interfaz web para el monitoreo y control de riego

VII-A2. Control automático y remoto de las bombas de riego: El sistema de riego se activó correctamente cuando la humedad del suelo descendió por debajo del umbral establecido del 40 %. Además se pudieron encender y apagar las bombas manualmente a través de la interfaz web. Este control híbrido entre automático y manual permitió una flexibilidad adicional en el proceso de riego, asegurando que se aplicara agua sólo cuando era necesario, optimizando el uso del recurso hídrico.



Figura 31. Control de riego desde la Web

VII-A3. Eficiencia energética y autonomía: El sistema demostró una alta eficiencia energética, gracias a la integración del panel solar de 18V 20W y la batería de plomo de 12V 5Ah. La matriz LoRa, conectada directamente a la red eléctrica, y los nodos alimentados por la batería recargada, operaron de manera continua sin necesidad de intervención humana. El tiempo de autonomía estimado fue de 1 a 2 días bajo condiciones de funcionamiento normal, lo que indica que el sistema es autosuficiente para operar en zonas rurales aisladas, donde la recarga solar garantiza la continuidad del servicio.

VIII. CRONOGRAMA

Para la creación del prototipo del sistema de monitoreo inalámbrico, cada objetivo específico se ha estructurado como una fase, desde el diseño y construcción hasta la implementación y validación. Este enfoque metódico asegura una ejecución eficiente y la resolución anticipada de desafíos.

- Objetivo 1: Diseñar el sistema de monitoreo inalámbrico de temperatura y humedad basado en LoRa.
 - Act._1.1_ Determinar los requisitos técnicos y funcionales del sistema (2 días).
 - Act._1.2_ Diseñar el esquema del sistema, incluyendo nodos y matriz (3 días).
 - Act._1.3_ Seleccionar los componentes y realizar un análisis de costos (3 días).
 - Act._1.4_ Crear el diseño lógico del sistema en software de simulación (5 días).
 - Act._1.5_ Realizar pruebas iniciales en simulación (3 días).
- Objetivo 2: Construir e implementar el sistema de monitoreo inalámbrico.
 - Act._2.1_ Adquirir y verificar los componentes electrónicos (4 días).
 - Act._2.2_ Ensamblar los nodos y la matriz con sus respectivos módulos LoRa (5 días).
 - Act._2.3_ Programar los nodos para enviar y recibir datos (6 días).
 - Act._2.4_ Desarrollar la página web para la visualización de datos (6 días).
 - Act._2.5_ Integrar el sistema con sensores y relés (4 días).
 - Act._2.6_ Realizar pruebas de comunicación entre matriz y nodos (4 días).
 - Act._2.7_ Ajustar el sistema según los resultados de las pruebas (3 días).
- Objetivo 3: Validar el sistema mediante pruebas en condiciones reales.
 - Act._3.1_ Configurar el sistema en un entorno de prueba (3 días).
 - Act._3.2_ Evaluar la precisión de los datos proporcionados por los sensores (6 días).
 - Act._3.3_ Analizar la eficiencia de los relés y bombas de agua en su activación automática (4 días).
 - Act._3.4_ Evaluar la estabilidad de la comunicación LoRa en campo (5 días).
 - Act._3.5_ Documentar las pruebas y resultados obtenidos (4 días).
 - Act._3.6_ Realizar ajustes finales en el sistema (3 días).

Como se muestra en el cronograma, se puede determinar que el tiempo aproximado requerido para completar el proyecto es de 102 días. Si el desarrollo del proyecto comienza el 1 de noviembre de 2024, tendría una fecha de finalización estimada para el 10 de febrero de 2025. Es importante considerar que estos plazos son aproximados y podrían variar según las circunstancias que surjan durante el desarrollo del proyecto.

Cronograma de Actividades																	
Nro.	Actividad	Noviembre				Diciembre				Enero				Febrero			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Planificación del proyecto	X															
2	Diseño de circuitos	X															
3	Adquisición de materiales		X														
4	Ensamblaje de hardware		X														
5	Programación de microcontrolador			X													
6	Pruebas de funcionamiento			X													
7	Ajustes y calibración				X												
8	Documentación del proyecto					X	X										
9	Revisión de diseño funcional							X									
10	Análisis de costos y proveedores								X								
11	Ensayo de hardware LoRa									X							
12	Pruebas LoRa a distancia										X						
13	Optimización energética											X					
14	Pruebas con sensores											X					
15	Ajustes web de visualización												X				
16	Ajustes e integración del sistema												X				
17	Evaluación de rendimiento													X			
18	Revisión final y entrega														X		

Figura 32. Cronograma de Actividades del Proyecto. Fuente: Los Autores.

IX. PRESUPUESTO

Tabla II
PRESUPUESTO DE MATERIALES UTILIZADOS EN EL PROYECTO

Cantidad	Nombre del elemento	Descripción	Costo Unit (\$)	Valor Total (\$)
1	ESP-WROOM-32	Microcontrolador Wi-Fi y Bluetooth	9.00	9.00
2	Arduino Uno	Microcontrolador	10.00	20.00
3	Módulo LoRa SX1278	Módulo de comunicación inalámbrica	16.00	48.00
2	Módulo relé	Control de dispositivos de potencia	2.00	4.00
2	Sensor DHT11	Sensor de temperatura y humedad	3.00	6.00
2	Sensor FC-28	Sensor de humedad del suelo	2.50	5.00
2	Bomba de agua 12V	240L/H DC 12V 4.2W	12.00	24.00
1	Batería de plomo 12V 7Ah	Fuente de energía para el sistema	30.00	30.00
2	Regulador LM2596	Regulador de voltaje ajustable	3.00	6.00
4	Placas PCB acrílico 5x7cm	Soporte para circuitos electrónicos	0.80	3.20
6	Borneras de 3 pines 2.54mm	Conectores para circuitos electrónicos	0.40	2.40
7	Borneras de 2 pines 5mm	Conectores para circuitos electrónicos	0.30	2.10
2	Metro de manguera	Para transporte de agua en riego	1.00	2.00
1	Metro de cable arcoíris 40 pines	Cableado para conexiones de sensores	3.50	3.50
5	Metros de cable 20 AWG	Cableado para conexiones eléctricas	0.30	1.50
1	Panel solar 18V 20W	Energía renovable para el sistema	35.00	35.00
1	Controlador MPPT	Gestor de energía solar	25.00	25.00
12	Perno 10mm de cola puntiaguda	Elementos de fijación	0.20	2.40
1	Placa de acero inoxidable	Plancha de 1.2 mm de grosor	100.00	100.00
TOTAL				329.10

X. CONCLUSIONES

El prototipo del sistema de monitoreo remoto de temperatura, humedad del aire y suelo, implementado con Iot y comunicación LoRa WAN, ha logrado cumplir su objetivo de automatizar el riego en zonas rurales. La comunicación entre los nodos y la matriz, basada en LoRa, ha demostrado ser efectiva para la transmisión de datos en tiempo real y para el control remoto de las bombas de riego. Además, la autonomía energética proporcionada por el panel solar y la batería de plomo ha asegurado la operación continua del sistema en entornos rurales, donde el acceso a energía eléctrica es limitado o nulo. La integración de una interfaz web facilita la supervisión y el control del sistema de manera remota, permitiendo al administrador revisar los parámetros desde cualquier lugar del mundo.

XI. RECOMENDACIONES

Para mejorar la autonomía energética del sistema, sería conveniente considerar baterías de mayor capacidad o alternativas como las baterías LiFePO₄, que ofrecen una vida útil más prolongada y son más eficientes en la carga. Además, la optimización del consumo energético de los nodos y el módulo LoRa podría mejorar la eficiencia general, especialmente durante los períodos de inactividad. Usar modos de bajo consumo en el hardware contribuiría a maximizar la duración de la batería.

Además, para garantizar la escalabilidad del sistema a medida que se incorporen más nodos, es importante optimizar la estructura de la comunicación LoRa WAN. A medida que el número de nodos crezca, podrían surgir problemas de congestión en la red. Por lo tanto, es necesario asegurarse de que el protocolo de comunicación sea eficiente y robusto, y que el sistema esté preparado para manejar múltiples nodos sin interferencias.

Finalmente, aunque el sistema ha funcionado bien en las pruebas iniciales, es recomendable realizar pruebas a largo plazo bajo condiciones reales. Esto permitirá identificar posibles fallos o áreas de mejora en el sistema y asegurar su fiabilidad en entornos de campo. Adicionalmente, implementar un sistema de monitoreo de fallos podría evitar inconvenientes y permitir un mantenimiento predictivo, evitando paradas inesperadas del sistema y asegurando su funcionamiento continuo.

REFERENCIAS

- [1] P. Badoni, S. Kaur, B. Sharma y R. Walia, «Enhancing Water Efficiency and Crop Yield in Agriculture Sector using IoT,» en *2023 International Conference on Advances in Computation, Communication and Information Technology (ICAICCIT)*, 2023, págs. 1039-1044. DOI: 10.1109/ICAICCIT60255.2023.10466092.
- [2] A. Andreadis, G. Giambene y R. Zambon, «Low-Power IoT Environmental Monitoring and Smart Agriculture for Unconnected Rural Areas,» en *2022 20th Mediterranean Communication and Computer Networking Conference (MedComNet)*, 2022, págs. 31-38. DOI: 10.1109/MedComNet55087.2022.9810376.
- [3] T. Akter, T. Mahmud, R. Chakma, N. Datta, M. S. Hossain y K. Andersson, «IoT-based Precision Agriculture Monitoring System: Enhancing Agricultural Efficiency,» en *2024 Second International Conference on Inventive Computing and Informatics (ICICI)*, 2024, págs. 749-754. DOI: 10.1109/ICICI62254.2024.00126.
- [4] A. P. Jaywant y A. S. Desai, «Cloud-Integrated Precision Agriculture with LoRaWAN and ESP32,» en *2024 5th International Conference on Smart Electronics and Communication (ICOSEC)*, 2024, págs. 619-624. DOI: 10.1109/ICOSEC61587.2024.10722074.
- [5] K. Barnes, S. Kulkarni, A. Sthalekar, J. S. Isaac y A. Kotrashetti, «LoraWAN Enabled Solar Photovoltaic Energy Monitoring System,» en *2021 IEEE Bombay Section Signature Conference (IBSSC)*, 2021, págs. 1-6. DOI: 10.1109/IBSSC53889.2021.9673295.
- [6] J. T, A. K. M, C. D, G. P, P. V. Anandan y S. O. Husain, «Precision Smart Agricultural Monitoring using LoRaWAN,» en *2023 International Conference for Technological Engineering and its Applications in Sustainable Development (ICTEASD)*, 2023, págs. 279-281. DOI: 10.1109/ICTEASD57136.2023.10585007.
- [7] B. Y. S. S. Murthy, C. B. K. Reddy, S. Jilani y M. Sindhwani, «Smart Irrigation System,» en *2022 1st International Conference on Sustainable Technology for Power and Energy Systems (STPES)*, 2022, págs. 1-4. DOI: 10.1109/STPES54845.2022.10006434.
- [8] A. Abusukhon y F. Altamimi, «Water Preservation Using IoT: A proposed IoT System for Detecting Water Pipeline Leakage,» en *2021 International Conference on Information Technology (ICIT)*, 2021, págs. 115-119. DOI: 10.1109/ICIT52682.2021.9491667.
- [9] M. B. Tephila, R. A. Sri, R. Abinaya, J. A. Lakshmi y V. Divya, «Automated Smart Irrigation System using IoT with Sensor Parameter,» en *2022 International Conference on Electronics and Renewable Systems (ICEARS)*, 2022, págs. 543-549. DOI: 10.1109/ICEARS53579.2022.9751993.
- [10] P. C. C. Veli, J. S. F. Bados, J. M. A. Villanueva, D. V. Y. Villa y C. A. C. Rojo, «Design of an Automated Drip Irrigation System in Tarma Valley by Controlling Soil Moisture,» en *2021 4th International Conference on Robotics, Control and Automation Engineering (RCAE)*, 2021, págs. 236-240. DOI: 10.1109/RCAE53607.2021.9638833.
- [11] S. Maurya, K. Bhagwat, M. Shivhare, S. Tomar y K. Markam, «Automatic Irrigation System,» en *2023 1st International Conference on Innovations in High Speed Communication and Signal Processing (IHCSP)*, 2023, págs. 519-522. DOI: 10.1109/IHCSP56702.2023.10127128.
- [12] P. A. F. Tapiceria y E. G. V. Magwili, «Hybrid Solar-Hydrokinetic Powered Automated Irrigation System,» en *2021 IEEE 13th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, 2021, págs. 1-5. DOI: 10.1109/HNICEM54116.2021.9731960.
- [13] K. Zhang, G. Liu, L. Nan, W. Hua y J. Han, «A Long Distance Environmental Monitoring System Based on Low Power IoT,» en *2024 6th International Conference on Communications, Information System and Computer Engineering (CISCE)*, 2024, págs. 1392-1396. DOI: 10.1109/CISCE62493.2024.10653076.
- [14] K.-K. Kee, R. Rashidi, O. K.-H. Kee, A. B. Han, I. Z. Patrick y L. M. Bawen, «Context-aware self-powered intelligent soil monitoring system for precise agriculture,» *International Journal of Electrical and Computer Engineering*, vol. 15, n.º 1, 1123 – 1131, 2025, Cited by: 0. DOI: 10.11591/ijece.v15i1.pp1123-1131.
- [15] M. Raja. G., B. Deepa, Nijanthan, B. Swapna, B. Divya y P. J. Chate, «Internet of Things (IoT) Assisted Smart Agriculture Monitoring and Summarization System using NodeMCU and Efficient Sensor Unit,» en *2023 9th International Conference on Smart Structures and Systems (ICSSS)*, 2023, págs. 1-6. DOI: 10.1109/ICSSS58085.2023.10408069.

- [16] U. Garlando, L. Bar-On, A. Avni, Y. Shacham-Diamand y D. Demarchi, «Plants and Environmental Sensors for Smart Agriculture, an Overview,» en *2020 IEEE SENSORS*, 2020, págs. 1-4. DOI: 10.1109/SENSORS47125.2020.9278748.
- [17] P. Saraswathi, S. Anandhika, M. Hannah Nissi, M. Prabha, A. Poornima Srinithi y B. Dhiyanesh, «Precision Farming: A Temperature and Humidity Monitoring Approach,» en *2024 5th International Conference on Image Processing and Capsule Networks (ICIPCN)*, 2024, págs. 698-702. DOI: 10.1109/ICIPCN63822.2024.00121.
- [18] R. Jegan, A. B. Evangeline y W. Nimi, «Wearable Environmental Monitoring System for Measurement of Environmental Parameters: A Pilot study,» en *2022 International Conference on Augmented Intelligence and Sustainable Systems (ICAISS)*, 2022, págs. 1185-1191. DOI: 10.1109/ICAISS55157.2022.10010880.
- [19] P. Sharma, M. Pandkar y S. Nambiar, «Wireless Greenhouse Monitoring using ESP32 and Sensor Array for Sustainable Crop Production,» en *2024 3rd International Conference on Artificial Intelligence For Internet of Things (AIIoT)*, 2024, págs. 1-6. DOI: 10.1109/AIIoT58432.2024.10574742.
- [20] A. N. de São José et al., «A Comparative Analysis of LoRa and LoRaWAN in the Presence of Jammers and Transient Interference,» en *2022 International Symposium on Electromagnetic Compatibility – EMC Europe*, 2022, págs. 586-591. DOI: 10.1109/EMCEurope51680.2022.9901081.
- [21] M. O. Farooq, «Clustering-Based Layering Approach for Uplink Multi-Hop Communication in LoRa Networks,» *IEEE Networking Letters*, vol. 2, n.º 3, págs. 132-135, 2020. DOI: 10.1109/LNET.2020.3003161.
- [22] A. Dino, D. Garlisi, F. Giuliano, D. Croce e I. Tinnirello, «Dynamic Adaptation of LoRaWan Traffic for Real-time Emergency Operations,» en *2022 18th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2022, págs. 457-460. DOI: 10.1109/WiMob55322.2022.9941684.
- [23] V. V. Das, A. Sathyan y D. D S, «Establishing LoRa based Local Agri-Sensor Network through Sensor plugin modules and LoRaWAN Data concentrator for extensive Agriculture Automation,» en *2022 IEEE 19th India Council International Conference (INDICON)*, 2022, págs. 1-6. DOI: 10.1109/INDICON56171.2022.10040050.
- [24] B. Babayiğit y F. Doğan, «LoRa Communication Evaluation Based Building Density in Ankara City,» en *2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, 2022, págs. 1-4. DOI: 10.1109/HORA55278.2022.9800058.
- [25] J. Ferreira, J. Cardoso, J. P. Amaro y C. I. F. Agreira, «LoRaWAN Based Street Lighting for Remote Areas with Shadow Zones,» en *2024 IEEE 22nd Mediterranean Electrotechnical Conference (MELECON)*, 2024, págs. 762-767. DOI: 10.1109/MELECON56669.2024.10608473.
- [26] H. Kim, H. Kim, S. Baek, R. Melenchuk, J. Soroka y A. Smith, «Hybrid LoRa Network Architecture: Automatic Switching between LoRaWAN and LoRa Mesh Network in Environments with Dynamic Obstacle Variations,» en *2024 33rd International Conference on Computer Communications and Networks (ICCCN)*, 2024, págs. 1-6. DOI: 10.1109/ICCCN61486.2024.10637558.
- [27] J. Bolboli, M. Salman, R. Prasad Naik y W.-Y. Chung, «Design and Performance Evaluation of a Relay-Assisted Hybrid LoRa/Optical Wireless Communication System for IoT,» *IEEE Open Journal of the Communications Society*, vol. 5, págs. 4046-4060, 2024. DOI: 10.1109/OJCOMS.2024.3424419.
- [28] Z. K. Farej y H. Y. Al-hayaly, «Accuracy Evaluation of Healthcare Monitoring System Based on ESP32 Microcontroller with IoT,» en *2023 International Conference on Engineering, Science and Advanced Technology (ICESAT)*, 2023, págs. 90-94. DOI: 10.1109/ICESAT58213.2023.10347330.
- [29] D. S. Kumar, M. Samer, A. Abhinav, P. Tejasree, G. S. Varsha y D. Koushik, «Design and Implementation of an IoT-Enabled Smart Street Lighting System using STM32 Microcontroller and ESP32,» en *2023 International Conference on Sustainable Computing and Smart Systems (ICSCSS)*, 2023, págs. 977-981. DOI: 10.1109/ICSCSS57650.2023.10169466.
- [30] H. Kareem y D. Dunaev, «The Working Principles of ESP32 and Analytical Comparison of using Low-Cost Microcontroller Modules in Embedded Systems Design,» en *2021 4th International Conference on Circuits, Systems and Simulation (ICCSS)*, 2021, págs. 130-135. DOI: 10.1109/ICCSS51193.2021.9464217.

- [31] S. Dey y T. Bera, «Design and Development of a Smart and Multipurpose IoT Embedded System Device Using ESP32 Microcontroller,» en *2023 International Conference on Electrical, Electronics, Communication and Computers (ELEXCOM)*, 2023, págs. 1-6. DOI: 10.1109/ELEXCOM58812.2023.10370327.
- [32] P. M. E., H. P. S, H. L y K. E, «IoT based Industrial Automation for Various Load using ATmega328p Microcontroller,» en *2022 6th International Conference on Intelligent Computing and Control Systems (ICICCS)*, 2022, págs. 471-475. DOI: 10.1109/ICICCS53718.2022.9788258.
- [33] Y. Tan, A. R. Prasetyo, D. Dwi Putra, E. Wismiana, R. Gunawan y A. Munir, «IoT-based Sensor System for Stop Line Traffic Area Using ATmega2560 Microcontroller,» en *2020 14th International Conference on Telecommunication Systems, Services, and Applications (TSSA)*, 2020, págs. 1-4. DOI: 10.1109/TSSA51342.2020.9310876.

XII. ANEXOS

Imágenes de la Implementación del Sistema de Monitoreo Lora Wan:

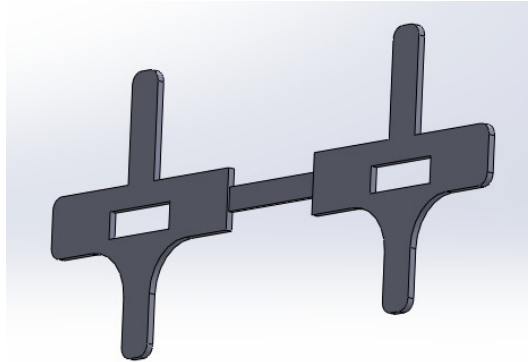


Figura 33. Diseño 3D soporte para el Panel Solar. Fuente: Los Autores.

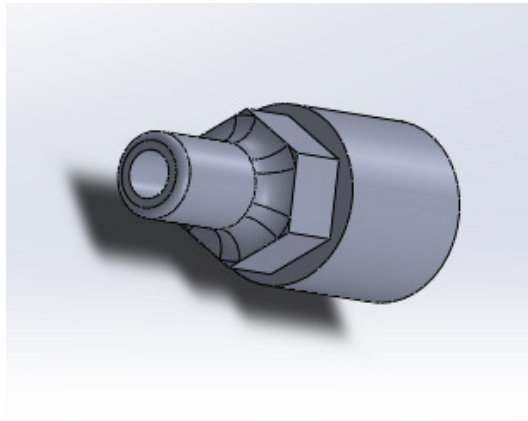


Figura 34. Diseño 3D adaptador para la bomba de agua. Fuente: Los Autores.

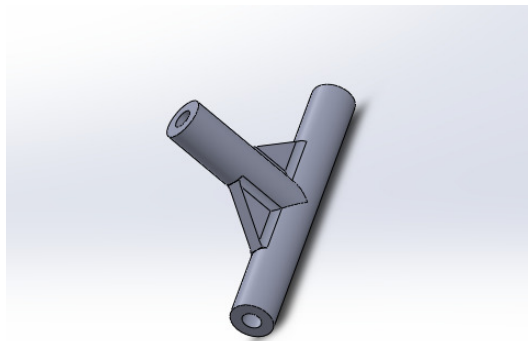


Figura 35. Diseño 3D adaptador para el desfogue del agua. Fuente: Los Autores.



Figura 36. Fijación y Conexión del Panel Solar. Fuente: Los Autores.



Figura 37. Red de Tubería para Desagüe en los Contenedores Plásticos. Fuente: Los Autores.



Figura 38. Dispensor de Agua del Prototipo. Fuente: Los Autores.



Figura 39. Prueba de los Nodos LoRa en Zona Aislada. Fuente: Los Autores.

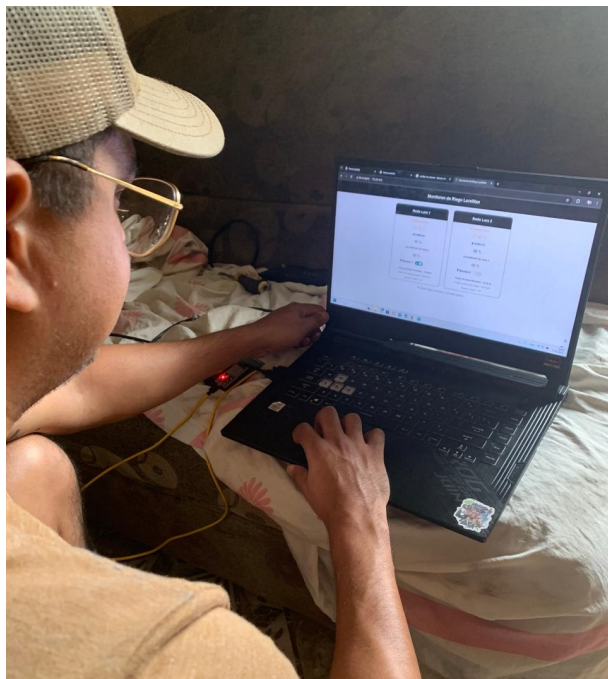


Figura 40. Monitoreo de datos del sistema LoRa WAN en la web. Fuente: Los Autores.