

RED DE DATOS INALÁMBRICA BASADA EN UAV PARA APLICACIONES
BÁSICAS DE E-AGRICULTURA

JUAN SEBASTIAN RODRIGUEZ RODRIGUEZ

UNIVERSIDAD DEL VALLE
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA
SANTIAGO DE CALI

2023

RED DE DATOS INALÁMBRICA BASADA EN UAV PARA APLICACIONES
BÁSICAS DE E-AGRICULTURA

JUAN SEBASTIAN RODRIGUEZ RODRIGUEZ

Trabajo de grado para optar por el título de
Ingeniero Electrónico

Director

Fabio German Guerrero Moreno, M. Sc.

UNIVERSIDAD DEL VALLE
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA
SANTIAGO DE CALI

2024

RESUMEN

La falta de conectividad en el sector agrícola representa una barrera para la implementación de dispositivos IoT en el monitoreo y análisis de cultivos. Actualmente, las tecnologías disponibles suelen estar asociadas a infraestructuras costosas, lo que limita su acceso para los pequeños y medianos productores. En este contexto, surge la necesidad de desarrollar un sistema accesible que integre tecnologías modernas a un costo reducido, facilitando la digitalización de la agricultura.

Este trabajo tiene como objetivo diseñar un sistema (Internet of Things - Unmanned Aerial Vehicle) de bajo costo orientado a la E-agricultura de cultivos de baja producción en el Valle del Cauca enfocado en cultivos distintos a la caña y de menor volumen departamental, por ejemplo lulo, mora, maracuyá, chontaduro, zapallo y guayaba, identificados por la UPRA (Unidad de Planificación Rural Agropecuaria) y la Gobernación como alternativas priorizadas para pequeños y medianos productores. Para ello, se integraron sensores para la medición de temperatura, humedad, luminosidad y humedad del suelo, cuyos datos son recolectados mediante una Raspberry Pi y transmitidos a un UAV (Unmanned Aerial Vehicle). Este UAV cuenta con otra Raspberry Pi con un módem USB 3G, permitiendo la conectividad a la nube.

Los datos recopilados se almacenan en una base de datos PostgreSQL, desde donde son procesados y enviados a una interfaz web mediante una API con WebSockets, permitiendo la visualización en tiempo real utilizando conexión celular. Se realizaron pruebas en cada etapa del sistema, validando su funcionamiento y eficiencia. Los resultados obtenidos demuestran que esta solución es una alternativa viable y escalable, contribuyendo al avance de la E-agricultura mediante el uso combinado de las tecnologías UAV, IoT y redes celulares.

Palabras clave: UAV, IoT, base de datos, desarrollo web, E-agricultura, red celular

ABSTRACT

The lack of connectivity in the agricultural sector represents a barrier to the implementation of IoT devices for crop monitoring and analysis. Currently, available technologies are often associated with costly infrastructures, limiting access for small and medium-sized producers. In this context, the need arises to develop an accessible system that integrates modern technologies at a reduced cost, facilitating the digitalization of agriculture.

This study aims to design a low-cost system based on the Internet of Things (IoT) and Unmanned Aerial Vehicles (UAVs), oriented towards E-agriculture for low-production crops in the Valle del Cauca region. To achieve this, sensors were integrated to measure temperature, humidity, luminosity, and soil moisture, with data collected by a Raspberry Pi and transmitted to a UAV. This UAV is equipped with another Raspberry Pi and a USB 3G modem, enabling cloud connectivity.

The collected data is stored in a PostgreSQL database, where it is processed and sent to a web interface through a WebSocket API, allowing real-time visualization. System tests were conducted at each stage to validate its functionality and efficiency. The results demonstrate that this solution is a viable and scalable alternative, contributing to the advancement of E-agriculture through the combined use of UAV, IoT, and cellular networks.

Keywords: *UAV, IoT, database, web development, E-agriculture, cellular network.*

DEDICATORIA

Agradezco profundamente a Dios por darme la fortaleza y sabiduría necesarias para superar cada desafío en este camino. Sin Su guía, este logro no habría sido posible.

Dedico este trabajo con todo mi amor y gratitud a mis padres, Florentino Rodríguez Ortiz y Adriana Patricia Wiesner, y a mi abuela, Teresa Wiesner, quienes, aunque ya no me acompañan en vida, siguen siendo mi mayor inspiración. Sé que desde donde estén, comparten conmigo la alegría de este logro y estarían orgullosos del camino que he recorrido.

También quiero expresar mi sincero agradecimiento a mi hermana Luz Melba Rodríguez Pulgarín, a mi tía Rocío Rodríguez Wiesner, mi prima Laura María Ayala y su esposo Oscar Hernández, quienes han sido una parte fundamental en mi crecimiento.

A mis sobrinos, Ana María Salazar Rodríguez y Daniel Humberto Salazar Rodríguez, gracias por su cariño y por brindarme un apoyo incondicional que ha sido fundamental en mi proceso de aprendizaje a lo largo de esta carrera.

Extiendo mi gratitud a mis primos, tíos y demás familiares, cuya confianza y motivación han sido clave para alcanzar uno de los logros más importantes de mi vida.

Finalmente, a Lina María Villate Real, por su amor inquebrantable y apoyo incondicional, acompañándome en cada momento significativo de mi carrera.

AGRADECIMIENTOS

Quiero expresar mi más profundo agradecimiento a mi director de trabajo de grado, M.Sc. Fabio Germán Guerrero Moreno, por su compromiso, apoyo y orientación a lo largo de este proceso. Su dedicación y excelente gestión de los recursos fueron fundamentales para la culminación de este trabajo.

Asimismo, extendiendo mi gratitud a la Universidad del Valle y a la Escuela de Ingeniería Eléctrica y Electrónica, así como a todos los profesores que compartieron su conocimiento y experiencia, permitiéndome alcanzar este logro académico y profesional.

Por último, quiero agradecer a mis amigos: John Wilmar Ortiz Villegas, Yobin David Vergara Valencia, Camilo Vargas Pantoja, Sebastián Quintero Corzo, Héctor Fabio Hernández López, Felipe Moreno Gómez, Jerson Lewis Orobio Lerma, Jesús Castrillón Vélez, Johan Martínez, Duván Becerra, José Aguirre, Brandon Ortiz y Mateo Ortiz. Su apoyo incondicional, compañía y los momentos inolvidables que compartimos hicieron de esta etapa una experiencia invaluable, y siempre llevaré en mi corazón cada instante vivido junto a ustedes.

Tabla de contenido

1.	INTRODUCCIÓN.....	12
2.	PLANTEAMIENTO DEL PROBLEMA	13
3.	ANTECEDENTES.....	15
4.	JUSTIFICACIÓN.....	18
5.	OBJETIVOS.....	19
5.1	OBJETIVO GENERAL	19
5.2	OBJETIVOS ESPECIFICOS.....	19
6.	MARCO TEORICO	20
6.1	IoT	20
6.2	UAV.....	20
6.3	E-Agriculture	21
6.4	Raspberry PI	21
6.5	Modelo OSI	22
6.6	Comunicación	23
6.6.1	<i>Tecnologías de Comunicación.....</i>	23
6.6.2	<i>Protocolos de comunicación.....</i>	26
6.7	Sensores para Aplicaciones de E-Agricultura.....	28
6.8	<i>Interfaz Web en aplicaciones IoT.....</i>	28
6.9	Bases de datos	30
6.9.1	<i>Tipos de bases de datos.....</i>	30
7.	METODOLOGIA.....	33
7.1	<i>Sistema embebido</i>	33
7.1.1	<i>Sensores.....</i>	34
7.2	<i>Dispositivo UAV</i>	47
7.2.1	<i>Dron F40</i>	47
7.2.2	<i>Raspberry PI</i>	54
7.3	<i>Desarrollo página web</i>	60
7.3.1	<i>API WebSockets.....</i>	60
7.3.3	<i>Base de datos</i>	61

7.3.4	JSON Web Token	65
7.3.5	Migración a la nube.....	68
8.	RESULTADOS	71
8.1	Datos sensores	71
8.2	Transmisión de datos al UAV.....	71
8.3	Almacenamiento de datos en la nube.....	72
8.4	Transmisión de datos hacia la interfaz web	72
8.5	Validación del sistema en un cultivo	76
8.6	Comparación de costos	78
9.	ANALISIS.....	81
9.1	Datos sensores:	81
9.2	Transmisión de datos al UAV.....	81
9.3	Migración a la nube.....	82
9.4	Transmisión de datos a la interfaz web.....	82
9.5	Validación en un cultivo	83
9.6	Comparación de costos	83
10.	CONCLUSIONES.....	84
11.	TRABAJOS FUTUROS	86
12.	REFERENCIAS	87

Figuras

Figura 1. Flujo de información utilizando API. Imagen tomada de [33]	30
Figura 2. Diagrama general del sistema. Elaboración propia	33
Figura 3. Diagrama de conexiones del sistema embebido. Elaboración propia	34
Figura 4. Pseudocódigo sensor de temperatura y humedad. Elaboración propia .	36
Figura 5. Pseudocódigo sensor de luminosidad. Elaboración propia	38
Figura 6. Pseudocódigo sensor capacitivo de humedad. Elaboración propia.....	40
Figura 7. Carcasa con el espacio para la Raspberry Pi. Elaboración propia	44
Figura 8. Carcasa con el espacio para la Powerbank. Elaboración propia	44
Figura 9. Parte frontal carcasa. Elaboración propia.	45
Figura 10. Pseudocódigo general de la Raspberry Pi. Elaboración propia	46
Figura 11. Controlador Pixhawk. Imagen tomada de [34]	47
Figura 12. Marco F450. Imagen tomada de [30]	48
Figura 13. Batería LiPo 11.1V 5200mAh. Imagen tomada de [35].....	49
Figura 14. Cargador baterías LiPo. Imagen tomada de [30]	50
Figura 15. Módulo de potencia GM V1.0. Imagen tomada de [31].....	50
Figura 16. ESC, motores y hélices. Imagen tomada de [31]	51
Figura 17. Control y receptor. Imagen tomada de [32].....	52
Figura 18. Dron F450. Elaboración propia	52
Figura 19. Modem 3G conectado a la Raspberry PI. Elaboración propia	56
Figura 20. Comandos de conexión modem 3G. Elaboración propia.....	57
Figura 21. Parámetros para conexión mediante PPP. Elaboración propia.	58
Figura 22. Pseudocódigo script Raspberry Pi UAV. Elaboración propia.....	59
Figura 23. Flujo de datos interfaz web, API y base de datos. Elaboración propia .	64
Figura 24. Interfaz inicio de sesión. Elaboración propia.....	65
Figura 25. Generación JWT. Elaboración propia	66
Figura 26. Interfaz página web principal. Elaboración propia	67
Figura 27. Interfaz sensor temperatura. Elaboración propia.	67
Figura 28. Comunicación de las interfaces. Elaboración propia.	68
Figura 29. URL base de datos y JWT_SECRET. Elaboración propia.....	69

Figura 30. Medición sensores sistema embebido	71
Figura 31. JSON temporal en la Raspberry Pi UAV. Elaboración propia.....	72
Figura 32. Registro de eventos API Express en render. Elaboración propia	72
Figura 33. Interfaz web recibiendo los datos. Elaboración propia.....	75
Figura 34. Archivo .csv sensor de temperatura. Elaboración propia.....	76

Tablas

Tabla 1. Costos de los componentes. Elaboración propia	79
Tabla 2. Comparación tecnologías. Elaboración propia.....	80

1. INTRODUCCIÓN

La E-agricultura es una actividad en la cual se juntan los procesos de la agricultura y la tecnología, esto ayuda a sistematizar actividades mediante sensores, sistemas de riego automatizados, alarmas, protocolos de comunicación entre otras. Se han implementado varias tecnologías en la agricultura para medir diferentes parámetros en la tierra, plantas y el ambiente para tener un control sobre la cosecha más eficaz y sistematizado. Una opción para complementar las tecnologías implementadas en los cultivos han sido los Unmanned Aerial Vehicle (UAV), permiten tener una comunicación más robusta entre el dispositivo y el usuario, también mediante una cámara se puede ver el estado del cultivo, se puede llegar a implementar un sistema de riego en el UAV para rociar la cosecha entre otras opciones más.

La complementación de todas las tecnologías anteriormente mencionadas es muy costosa para los agricultores que tienen cultivos de mediana a baja escala para el sostenimiento propio, la falta de acceso a maquinaria especializada, sistemas de monitoreo agrícola y herramientas de gestión de cultivos complica la tarea diaria de estos agricultores, quienes se ven obligados a depender en gran medida de métodos tradicionales. Este obstáculo no solo afecta la eficiencia y la productividad de sus cultivos, sino que también contribuye a la vulnerabilidad económica de estos agricultores frente a las fluctuaciones del mercado y los impactos climáticos impredecibles.

Se propone realizar un dispositivo de bajo costo para la medición de parámetros fundamentales en un cultivo con la ayuda de un UAV. Al finalizar el proyecto se busca obtener el correcto funcionamiento del dispositivo en un cultivo, teniendo una medición y transmisión exitosa de los diferentes parámetros como la humedad, temperatura, luminosidad hacia una interfaz web con la ayuda de un UAV el cual va a tener implementado un dispositivo de comunicación. En función de entender la propuesta planteada se presenta una breve descripción acerca de los trabajos anteriormente realizados en el sector agrícola y un marco teórico en donde se definen las variables más relevantes para este trabajo.

2. PLANTEAMIENTO DEL PROBLEMA

La agricultura es un conjunto de actividades conducente a la producción de alimentos tanto para producción industrial como para consumo personal. A nivel mundial la agricultura es una de las producciones más importantes en el ámbito económico, representando desde el 4% del PIB hasta el 25% para países menos desarrollados [1]. Se estima que para el año 2050, la población mundial aumente más de un tercio; por ende, la demanda de alimentos crecerá anualmente, por lo que se espera tener un incremento, con respecto a la actualidad, del 70%. Esto se conseguirá aumentando la superficie de tierra cultivable, aproximadamente, en 70 millones de hectáreas [2].

En el caso de Colombia se cuenta con 39.2 millones de hectáreas potenciales para cultivar de las cuales solo 5.3 millones se han utilizado, esto representa un potencial de 13.5%. En los últimos años se han potencializado nuevos cultivos diferentes a los tradicionales como aguacate, piña y cacao [3]. Se requiere tener una buena proyección y planeación para mantener la tendencia de crecimiento con innovación de sistemas y sistematización de tareas con el fin de aumentar las exportaciones agrícolas.

Uno de los desafíos más importantes de la agricultura a lo largo de la historia ha sido proporcionar efectivamente los productos con respecto al crecimiento de la población, el aumento en el consumo, la degradación de la tierra y las variaciones climáticas. Con el tiempo el desafío se ha incrementado por la falta de mano de obra en el sector y la escasa inversión en la estructura agrícola para una producción en masa. Los retos de la agricultura planteados no radican tanto en el incremento de las tierras cultivables como en el rendimiento e intensidad de los cultivos, por lo que es necesario invertir y destinar algunos recursos públicos y privados a modernizar el sector agrícola dejando entrar nuevas tecnologías [4].

Es en este contexto en donde la implementación de la tecnología es un apoyo significativo para el aumento de la productividad. Se han implementado varias estrategias de medición en la E-agricultura (uso innovador de las TIC y soluciones digitales para impulsar el sector agrícola). mediante sensores remotos para poder medir diferentes parámetros como la humedad del suelo, datos del cultivo, temperatura, entre otras. Gracias a estos datos se pueden tomar decisiones más acertadas logrando así una cosecha más productiva y sistematizada. Para poder lograr las soluciones planteadas se requiere un sistema de comunicación eficaz entre el dispositivo recolectando los datos y el receptor.

Los UAV son una opción bastante viable por su bajo costo, permite cubrir una extensión geográfica mayor y por consiguiente permite recolectar los datos del terreno para enviarlos al dispositivo receptor con un índice de fallo menor en la comunicación. El envío de la información se logra con la implementación del Internet of Things (IoT) el cual establece una conexión directa entre los sensores con sus mediciones específicas y el dispositivo receptor mediante internet [5].

El problema en Colombia radica en que la mayoría de los agricultores son campesinos, lo que representa al 30% de la población colombiana. Esta población carece de los recursos económicos necesarios para adquirir un sistema de monitoreo para sus cultivos, principalmente debido a la desinformación en torno a la E-agricultura y al elevado costo en relación con sus ingresos [6]. Por otra parte, no hay muchas empresas nacionales que distribuyan estos sistemas en conjunto a bajo costo. Así mismo en zonas rurales se suma la limitación de cobertura de red celular, lo que dificulta la transmisión directa de datos desde el campo.

Por ende, se busca implementar un dispositivo con sensores de medición que transmitan los datos mediante un sistema de comunicación hacia un dron que sobrevuele por los cultivos y luego envíe la información a una aplicación web mediante la implementación de IoT, con un bajo costo en los componentes para enfocarse a una pequeña o quizá mediana escala.

De acuerdo con lo anterior se formula la siguiente pregunta problema ¿Cómo implementar un sistema de monitorización de bajo costo en los cultivos de mediana a baja escala permitiendo al agricultor obtener datos mediante un dron utilizando IoT?

3. ANTECEDENTES

En este apartado se presentan trabajos previos relacionados con la implementación de un sistema de monitoreo hacia los cultivos mediante una transmisión de datos hacia un dispositivo. Se consultaron bases de datos en la Biblioteca Digital de la Universidad del Valle utilizando IEEE Xplore, Agriculture science database y Agriculture.

En el trabajo de grado **“Agricultura de precisión en el mundo y en Colombia: Revisión bibliográfica”** de la Universidad del Valle (2020) [7], se realiza un estudio a profundidad sobre la agricultura de precisión en donde primeramente se identifican las ideas principales a trabajar definiendo la agricultura de precisión, recolección de datos, interpretación de datos y demás conceptos para luego darles un enfoque a las herramientas para su desarrollo, como la importancia de los sensores remotos, herramientas geoespaciales y tecnología de la comunicación e información. Este trabajo de investigación ayuda a comprender el avance que ha tenido la agricultura de precisión en los últimos años y las estrategias más utilizadas a nivel mundial esto con el objetivo de tener una mejor gestión de recursos y evitar las pérdidas.

En la propuesta de trabajo de grado sobre **“Sistema de riego inteligente usando el Internet de las Cosas (IoT) orientado a pequeños agricultores de cultivos hidropónicos y/o agricultura urbana.”** De la Universidad del Valle (2023), se implementa un dispositivo de riego inteligente en un cultivo utilizando la técnica de hidroponía, se construye un cultivo prototipo en donde las plantas se encuentran un tanque pequeño y debajo se encuentra otro tanque donde se almacena el agua con los nutrientes y nivel de oxígeno necesarios. Se miden parámetros de temperatura, humedad, nivel de PH y luminosidad en las plantas los cuales van a un sistema de potencia que le suministre adecuadamente el agua mediante una bomba de agua. Se implementó una capa de red mediante Wifi para enviar los datos de los sensores a una aplicación móvil y web. La interfaz web cuenta con un actuador el cual va a encender o apagar el funcionamiento de la bomba de agua dependiendo los parámetros que el usuario vaya interpretando del cultivo.

En el trabajo de grado **“Diseño e implementación de un sistema de riego automatizado y monitoreo de variables ambientales mediante IoT en los cultivos urbanos de la fundación mujeres empresarias Marie poussepin”** de la Universidad Católica de Colombia (2020) [8], se implementó un diseño de un dispositivo electrónico con el fin de monitorear variables para implementar un sistema de riego a baja escala el cual envía los datos a una aplicación móvil.

Definieron los elementos a usar como los tipos de riego, sensores, variables a medir y métodos de intercomunicación. El método de riego utilizado fue por aspersión mediante un sistema de control con la ayuda del microcontrolador del Arduino uno, la variable de control es la humedad del suelo la cual indica que si esta menor del 65% encienda el riego y además en una pantalla LCD se visualiza la humedad con la temperatura. Para el envío de datos en la nube se utilizó un módulo wifi ESP8266 con el cual cualquier dispositivo con internet puede estar revisando las variables a tiempo real. Por otra parte, se puede apreciar el funcionamiento del sistema de riego con sensores y de qué manera está ubicado el dispositivo de control en el cultivo para el buen funcionamiento a largo plazo.

Para la implementación de un proyecto laboral titulado **“Desarrollo de dispositivo IoT para monitoreo agrícola de bajo coste”** publicado en el Centro de investigación e innovación en tecnologías de la información y comunicación (2020) [9], se diseñó un sistema embebido IoT autónomo mediante energía solar para monitorizar variables específicas en la agricultura con un dispositivo de bajo costo y comunicación continua a tiempo real. El dispositivo tiene como objetivo medir las variables de los sensores de humedad, índice de clorofila, concentración de iones de hidrógeno (PH) e iluminancia. El diseño electrónico está conformado por los sensores anteriormente mencionados, un microcontrolador, un GPS, un circuito integrado para tomar la medición del estado de la batería y un transceptor con el protocolo de comunicación LoRa. El protocolo de comunicación LoRa es muy eficaz para el proyecto por el bajo consumo de energía y la correcta transmisión de los datos permitiendo una mejor autonomía en el dispositivo. Se registran todos los resultados y análisis con pruebas de funcionamiento de los sensores y del sistema de comunicación permitiendo visualizar los datos en la plataforma web Thingspeak.

En el artículo titulado **“IoT Sensing Platform for e-Agriculture in África”** publicado en la conferencia IST-África (2020) [10], se propone una plataforma de detección mediante IoT con el objetivo de implementar tecnologías de detección y comunicación, esto con el fin de identificar el estado del suelo y orientar al pequeño agricultor en la aplicación de un fertilizante. Los sensores integrados en el dispositivo son de humedad, PH, luz y temperatura ambientes, también se incorporó una cámara (Raspberry Pi Camera v2) la cual tiene como función adquirir imágenes del suelo, aplicando fuentes de luz ultravioleta (UV) o infrarroja (IR) para detectar características que no son fácilmente perceptibles a través de la luz visible. Para la prueba del funcionamiento de los sensores se utilizó un microcontrolador común como el Arduino UNO, luego en el desarrollo del proyecto se utilizó una Raspberry Pi en la cual no se identifica el programa o interfaz usada en la implementación.

En el documento titulado “**E-agricultura inteligente basada en iot monitoreo con aplicación Android**” publicado en la conferencia internacional 2022 sobre tendencias emergentes en ingeniería y ciencias médicas (2022) [11], se plantearon los principales problemas que tienen los pequeños agricultores en la India como las enfermedades en las plantas, humedad del terreno y ataque de mamíferos. Por consiguiente, se identificaron los tipos de sensores requeridos tales como el sensor de humedad, nivel de agua, temperatura e infrarrojo. El sensor del nivel de agua arroja una medida que va a controlar un sistema de extracción de agua mediante una bomba en los casos en donde sobrepase el nivel por las lluvias e inundaciones, el sensor infrarrojo tiene como objetivo proteger el cultivo de los mamíferos mediante un led o una alarma la cual le indique al agricultor algún movimiento. Se instalaron los sensores en un cultivo a pequeña escala para corroborar su funcionamiento mediante un microcontrolador Arduino Mega, los datos recibidos al Arduino se enviaron a una aplicación móvil para tener un constante monitoreo. El proyecto tiene como objetivo reemplazar métodos comunes como mallas eléctricas para la protección de los mamíferos y también tener un control del cultivo a baja escala más preciso.

Se publicó un artículo en la Conferencia Internacional sobre Computación Avanzada y Tecnologías Innovadoras en Ingeniería titulado “**Review of IoT Technologies used in Agriculture**” (2022) [12], en donde se revisan las diferentes aplicaciones de IoT sobre la agricultura en la India las cuales se utilizan para el control de plagas y crecimiento de las plantas. Analizan el sistema de riego controlado mediante variables medidas e internet, el monitoreo de suelo utilizando la técnica 6LoWPAN para saber utilizar diferentes tipos de fertilizantes y por último el monitoreo del agua para saber el nivel de alcalinidad, temperatura, oxígeno disuelto, entre otras. Se analiza las capas utilizadas en la E-agriculture, explicando los sensores utilizados en la capa física con los diferentes tipos de conexión a internet, la siguiente es la capa de red proporcionando comunicación entre el middleware y la capa física mediante Internet, la capa de servicios proporciona datos estadísticos como la información de los cultivos, por último, la capa de análisis suministra información sobre el rendimiento de los cultivos, clima, detección de plagas entre otros.

4. JUSTIFICACIÓN

En la actualidad, la agricultura busca sistematizar procesos mediante tecnologías para lograr cultivos más productivos y rentables. La implementación de sensores en los cultivos ha resultado ser una herramienta viable para las industrias agrícolas. Sin embargo, los cultivos de mediana a baja escala han enfrentado limitaciones en el acceso a estos dispositivos debido a sus altos costos y su difícil implementación. En respuesta a esta problemática, se propone que el bajo costo del dispositivo sea una característica fundamental, junto con su fácil manejo, para que cualquier persona tenga la posibilidad de adquirirlo y usarlo en su cultivo.

Reconociendo dichas limitaciones se propone diseñar un sistema de bajo costo implementado las tecnologías IoT y UAV, orientado a la agricultura de mediana a baja escala. Este sistema va complementado con una página web en donde se visualiza y almacena los datos medidos en la nube durante determinado tiempo. La complementación de los sensores remotos con la tecnología UAV permite reducir las limitaciones de medición en cultivos amplios y tener una comunicación más eficaz entre el dispositivo y la interfaz web.

5. OBJETIVOS

5.1 OBJETIVO GENERAL

Diseñar un sistema (Internet of Things - Unmanned Aerial Vehicle) de bajo costo orientado a la E-agricultura de cultivos de mediana a baja producción en el Valle del Cauca.

5.2 OBJETIVOS ESPECIFICOS

1. Investigar las variables a monitorear en el cultivo usando sensores de bajo costo y justificar su elección.
2. Implementar un sistema prototipo de bajo costo IoT para realizar la lectura de los sensores desde un UAV y transmitir los datos leídos a una página web.
3. Desarrollar una aplicación web con servicios de almacenamiento en la nube que permita consultar la información del sistema de e-agricultura en todo momento y desde cualquier lugar.
4. Validar el funcionamiento del sistema prototipo en un cultivo de prueba.

6. MARCO TEORICO

El desarrollo de este proyecto involucra diversas tecnologías y metodologías que permiten la integración eficiente entre hardware y software. En este apartado, se abordan los conceptos fundamentales que sustentan su funcionamiento, desde la comunicación entre dispositivos hasta el procesamiento y transmisión de datos. Se exploran tecnologías como el Internet de las Cosas (IoT), los Vehículos Aéreos No Tripulados (UAV), la E-Agricultura y el uso de plataformas como Raspberry Pi. Además, se presentan los protocolos de comunicación esenciales y la estructura del modelo OSI, elementos clave para garantizar la interconectividad y la gestión eficiente de la información en el sistema.

6.1 IoT

El Internet de las Cosas (IoT) es una red de dispositivos conectados entre sí que permite la comunicación y transferencia de datos. Esta red se compone de tres elementos esenciales: los dispositivos inteligentes, la aplicación IoT y la interfaz gráfica. Los dispositivos inteligentes, equipados con sensores y tecnologías de comunicación, son los encargados de recopilar la información del entorno y enviarla en tiempo real. La aplicación IoT utiliza los datos recolectados para la toma de decisiones, empleando diversos métodos como el aprendizaje automático (machine learning) y el diseño de software. Finalmente, la interfaz gráfica, conectada a la nube, es crucial para la visualización y gestión de los dispositivos, permitiendo a los usuarios leer y analizar la información, así como controlar y monitorear los dispositivos conectados [13].

6.2 UAV

Los Unmanned Aerial Vehicles (Vehículos Aéreos No Tripulados) combinan la robótica con la aeronáutica, formando dispositivos aéreos que comúnmente se controlan desde tierra. Este campo es muy amplio y abarca diversas clases de dispositivos, cuya infraestructura y funcionamiento varían según su propósito. Los UAV pueden utilizarse para una variedad de fines, incluyendo comerciales, científicos y recreativos.

En el ámbito comercial, los UAV son frecuentemente equipados con cámaras y estabilizadores de vuelo para capturar imágenes y videos aéreos de alta calidad. Por otra parte, estos dispositivos son utilizados en sectores como la agricultura, donde facilitan la monitorización de cultivos y envío de datos; en la

construcción, para inspeccionar y mapear terrenos; y en la industria cinematográfica, para obtener tomas aéreas impresionantes. Además, hay muchos más campos en donde se pueden utilizar, como en la entrega de productos, la geología, la gestión de recursos naturales, etc.

Fuera del ámbito comercial, los UAV tienen aplicaciones en la vigilancia y seguridad, en operaciones de búsqueda y rescate, en el análisis ambiental y en el sector militar. Cada aplicación específica determina las características técnicas del UAV, como su tamaño, capacidad de carga, autonomía de vuelo y sistemas de control [14].

6.3 E-Agriculture

La E-Agriculture es la implementación de nuevas tecnologías en el sector agrícola, abarcando un campo extenso con diversas aplicaciones en los cultivos. Según la Organización de las Naciones Unidas para la Alimentación y la Agricultura (FAO), se define como un campo emergente que se centra en la mejora del desarrollo agrícola y rural mediante tecnologías de la información y la comunicación (TIC). Entre estas aplicaciones se incluyen la medición de parámetros, la toma de imágenes y la geolocalización.

Estas actividades implican una serie de prácticas que requieren la integración de múltiples tecnologías, adaptadas a las necesidades específicas de cada proyecto. Esto se logra mediante el intercambio de datos, que permite optimizar los procesos agrícolas, mejorar la eficiencia en el uso de recursos y aumentar la productividad de los cultivos. Además, facilita la toma de decisiones basada en datos precisos y actualizados, mejora la gestión de riesgos y promueve prácticas agrícolas sostenibles. La E-Agriculture transforma la manera en que se gestionan los cultivos, contribuyendo al desarrollo rural y a la seguridad alimentaria a nivel global [15].

6.4 Raspberry Pi

La Raspberry Pi es una computadora de placa única (SBC) a baja escala, compuesta por diferentes componentes que se encuentran en un ordenador, como un procesador, un chip gráfico, memoria RAM, conectividad mediante Wi-Fi, Bluetooth y Ethernet, además de puertos USB y HDMI.

En el ámbito práctico, la Raspberry Pi es muy utilizada debido a que ofrece un dispositivo completo y compacto, el cual se puede integrar en diferentes

proyectos. Una de sus características más destacadas es la inclusión de un conjunto de pines GPIO (General Purpose Input/Output). La Raspberry Pi 4, por ejemplo, cuenta con 40 pines GPIO, divididos en pines de energía (5V, 3.3V y GND) y pines de propósito general. Estos pines permiten realizar una variedad de funciones como entradas y salidas digitales, y comunicación serial (UART), síncrona (SPI) y I2C. Esto posibilita la conexión con una amplia gama de sensores, actuadores y otros dispositivos electrónicos, ampliando las capacidades de los proyectos desarrollados con la Raspberry Pi, desde la robótica y la domótica hasta aplicaciones de Internet de las Cosas (IoT) [16]. En el presente proyecto se implementarán dos Raspberry Pi 3b+, cada una con funciones específicas para optimizar el monitoreo y la comunicación de datos. La primera Raspberry Pi 3b+ se encargará de la medición de los sensores instalados en el cultivo. Esta unidad ejecutará un script diseñado para la comunicación mediante el protocolo MQTT, lo que permitirá transmitir los datos recolectados de manera eficiente y en tiempo real. La segunda Raspberry Pi 3b+ estará montada en un dron, y su función principal será recibir los datos enviados desde la primera Raspberry Pi 3b+. Posteriormente, estos datos serán transmitidos a un servicio en la nube, donde se almacenarán y procesarán.

6.5 Modelo OSI

El modelo OSI es una forma de entender cómo se comunican los dispositivos en una red. Está dividido en siete capas, que van desde lo más básico, como los cables y señales eléctricas, hasta lo más cercano al usuario, como las aplicaciones que usamos para navegar por internet o enviar correos. Cada capa tiene una función específica: unas se encargan de enviar los datos físicamente, otras los organizan, y otras se aseguran de que lleguen correctamente a su destino. Por ejemplo, la capa de red se ocupa de buscar la mejor ruta para que los datos lleguen al otro dispositivo, mientras que la de enlace verifica que no haya errores en la transmisión.

Las capas más altas trabajan con la información que usamos directamente. La capa de transporte se asegura de que los datos lleguen completos, ya sea de forma confiable como con TCP o más rápida pero sin garantía, como con UDP. Luego están las capas de sesión y presentación, que ayudan a que la comunicación se mantenga estable y que los datos se entiendan correctamente, incluso si están comprimidos o cifrados. Finalmente, la capa de aplicación es la que nos permite usar servicios como páginas web, correos electrónicos o chats. En conjunto, todas las capas hacen posible que la comunicación digital funcione

de manera ordenada y eficiente, aunque no lo notemos al usar nuestros dispositivos [17].

6.6 Comunicación

Los protocolos de comunicación son sistemas que definen un conjunto de reglas y estándares para permitir la transferencia de información entre dispositivos en una red. Estos protocolos operan en distintas capas del modelo OSI, cada una desempeñando funciones específicas para asegurar que la comunicación sea eficiente, segura y confiable. Dependiendo del propósito del protocolo, pueden encargarse de tareas como el control de la señal física, la gestión de errores, el direccionamiento de los datos, o la organización de mensajes entre aplicaciones.

La implementación de estos protocolos en el ámbito de IoT es esencial para garantizar una comunicación eficaz y confiable entre diferentes dispositivos. Por otra parte, existen tecnologías de comunicación que proporcionan la infraestructura para la transmisión de datos y permiten la implementación de diversos protocolos según la capa en la que operen.

6.6.1 Tecnologías de Comunicación

- Wi-Fi (802.11)

Wi-Fi es una de las tecnologías de comunicación más reconocidas y eficientes debido a su alta capacidad de transmisión de datos y su facilidad de implementación. Funciona en las capas física y de enlace de datos del modelo OSI y no maneja capa de red ya que se basa en el uso de protocolos como TCP/IP para gestionar la transferencia de datos. Sin embargo, en comparación con otras tecnologías utilizadas en IoT, Wi-Fi consume una cantidad considerable de energía y tiene un alcance limitado, lo que lo hace menos adecuado para aplicaciones que requieren transmisión continua de datos a largas distancias o en ubicaciones remotas. Por otro lado, es ideal para aplicaciones que necesitan una conexión rápida y estable en distancias cortas, especialmente cuando el consumo de energía no es una limitación crítica [18].

- LoRa

LoRa es una tecnología de comunicación inalámbrica que utiliza una modulación basada en Chirp Spread Spectrum (CSS) para transmitir datos en radiofrecuencias de baja potencia. Su principal ventaja en comparación con otras tecnologías es su eficiencia energética, permitiendo que un módulo LoRa funcione durante hasta 10 años con una pequeña batería. Además, LoRa es capaz de transmitir información a distancias de hasta 15 km en condiciones óptimas utilizando una tipología de red en estrella, lo que la hace especialmente útil para aplicaciones en áreas rurales o entornos donde los dispositivos deben estar muy dispersos, como en la agricultura o la monitorización ambiental.

Sin embargo, una de sus desventajas más significativas es su limitada capacidad para transmitir grandes volúmenes de datos. Debido a que opera en bandas de frecuencia bajas para maximizar el alcance y reducir el consumo de energía, LoRa no es adecuada para aplicaciones que requieren un alto ancho de banda o la transmisión de datos complejos, como video o archivos grandes [37].

- Zigbee

Zigbee es una tecnología de comunicación inalámbrica que utiliza una modulación de radiofrecuencia de espectro ensanchado. Se destaca por su bajo consumo de energía, lo que permite que los dispositivos alimentados por baterías pequeñas puedan operar durante varios años, haciéndola ideal para aplicaciones como la automatización del hogar, redes de sensores y control industrial. Una característica clave de Zigbee es su capacidad para formar redes en malla, donde los dispositivos pueden interconectarse y retransmitir datos, lo que amplía la cobertura y mejora la estabilidad de la red.

Por otro lado, una de las principales desventajas de Zigbee es su alcance limitado, que suele variar entre 10 y 100 metros en interiores, dependiendo del entorno y los obstáculos. Además, la tecnología tiene una capacidad de transmisión de datos relativamente baja, lo que la hace adecuada para aplicaciones que requieren el envío de pequeñas cantidades de datos, como la información de sensores, pero no es

ideal para aplicaciones que demandan un ancho de banda elevado [19].

- Bluetooth

Es una tecnología que permite establecer conexiones punto a punto o punto a multipunto para el intercambio de datos entre dispositivos, operando en las capas física y de enlace de datos del modelo OSI. Tiene un rango de conexión efectivo de hasta 10 metros, más allá del cual pueden comenzar a experimentarse interferencias y pérdida de calidad en la transmisión de datos, lo que limita su uso en aplicaciones que requieren comunicaciones de largo alcance.

Bluetooth utiliza un chip especializado que ofrece una velocidad máxima de transmisión de aproximadamente 3 Mbps (dependiendo de la versión de Bluetooth), aunque las velocidades pueden ser menores en aplicaciones típicas. Opera en la banda de frecuencia global de 2,4 GHz, que es compartida con muchas otras tecnologías inalámbricas, lo que a veces puede aumentar las interferencias en entornos congestionados. Aunque ha habido avances en términos de eficiencia energética, como el desarrollo de Bluetooth Low Energy (BLE), las versiones clásicas de Bluetooth pueden tener un consumo de energía más alto. Esto puede no ser ideal para dispositivos que requieren conexiones continuas durante largos períodos de tiempo, especialmente si se busca optimizar la duración de la batería [20].

- Red celular

La red celular es un método de comunicación eficaz que utiliza una infraestructura de torres de telecomunicaciones para transmitir y recibir datos. Estas torres operan en diferentes bandas de frecuencia, las cuales varían según la ubicación geográfica y el operador móvil. Existen varias generaciones de tecnologías celulares, que se diferencian en términos de cobertura y capacidad de transmisión de datos: 2G, 3G, 4G LTE, y 5G.

- 2G es la tecnología más antigua y obsoleta, adecuada solo para llamadas y mensajes de texto básicos.

- 3G mejora la velocidad y permite la transmisión de datos para aplicaciones como la navegación web y el envío de correos electrónicos.
- 4G LTE ofrece una conexión rápida y eficiente, ideal para transmitir video en alta definición y aplicaciones que requieren un ancho de banda mayor.
- 5G es la tecnología más avanzada, proporcionando velocidades ultrarrápidas, baja latencia, y capacidad para conectar una gran cantidad de dispositivos simultáneamente.

En el ámbito de IoT, la red celular es particularmente eficiente para la comunicación con la nube, ya que no requiere módems ni puntos de acceso adicionales. Los dispositivos IoT pueden conectarse directamente a la red a través de un módulo celular, facilitando el envío de datos a servidores remotos [21]. Sin embargo, en zonas rurales la cobertura de red suele ser limitada, lo que puede dificultar o impedir la transmisión continua de datos, haciendo necesario implementar soluciones alternativas como repetidores, antenas de alta ganancia o el uso de UAVs para la recolección y envío de información.

6.6.2 Protocolos de comunicación

- HTTP/HTTPS

Para transmitir datos a la nube, el método más común es utilizar HTTP/HTTPS. HTTP (Hypertext Transfer Protocol) es un protocolo de capa de aplicación que define cómo se envían y reciben datos a través de la web. HTTPS (HTTP Secure) es la versión segura de HTTP, que encripta la comunicación para proteger la información sensible y garantizar la privacidad. Esto es especialmente importante en aplicaciones IoT que manejan datos críticos o confidenciales [22].

- MQTT

Message Queuing Telemetry Transport (MQTT) es una de las tecnologías más reconocidas en la capa de aplicación para el intercambio de información en dispositivos IoT. Al operar en la capa

más alta del modelo OSI, permite un intercambio eficiente de datos entre los sensores y la nube. Utiliza un modelo de publicación/suscripción, en el que los datos se publican en un broker, como Mosquitto, y otros dispositivos o aplicaciones pueden suscribirse para recibir esta información en tiempo real.

MQTT es muy flexible en cuanto al tipo de datos que maneja, ya que envía información en forma de secuencias de bytes. Esto permite el uso de diversos formatos, como JSON (para una estructura clara y fácil de interpretar), texto plano (TXT), datos binarios, y XML (para aplicaciones específicas). Esta flexibilidad lo convierte en una opción ideal para aplicaciones IoT, donde es crucial manejar diferentes tipos de datos de manera eficiente.

El broker Mosquitto utiliza siempre TCP/IP para la comunicación, lo que garantiza una transmisión confiable de los mensajes. Puede alojarse de manera local en un dispositivo para recibir y distribuir los datos temporalmente o puede estar alojado en la nube, lo que amplía su aplicación al permitir el acceso remoto y la conectividad a través de la web [38].

- SCP

El Protocolo de Copia Segura (SCP, Secure Copy Protocol) es un método de transferencia de archivos que permite enviar datos de manera segura entre un equipo local y uno remoto, o entre dos equipos remotos, utilizando el protocolo SSH y el puerto TCP 22. Su principal ventaja es la protección de la información mediante cifrado y autenticación, evitando accesos no autorizados. Para establecer la conexión, SCP requiere credenciales SSH o una clave pública autorizada. Es compatible con Windows, macOS y Linux y se utiliza comúnmente en entornos donde se prioriza la seguridad y la rapidez en la transmisión de datos. Aunque en algunos casos ha sido reemplazado por SFTP, sigue siendo una opción eficiente para la transferencia puntual de archivos sin necesidad de interfaces gráficas [23].

- Websocket

WebSocket es un protocolo de comunicación que permite una conexión bidireccional y persistente entre un servidor y un cliente, a diferencia de HTTP, que requiere solicitudes constantes. Opera sobre TCP y permite el envío de datos en tiempo real con baja latencia. Tras una conexión inicial mediante HTTP, el canal se mantiene abierto, evitando la sobrecarga de múltiples solicitudes. Este protocolo es ampliamente utilizado en aplicaciones que requieren transmisión continua de datos, como juegos en línea, sistemas de monitoreo en tiempo real, sensores IoT y plataformas de mensajería instantánea [24].

6.7 Sensores para Aplicaciones de E-Agricultura

Los sensores son esenciales para cualquier aplicación IoT en la agricultura, ya que permiten la monitorización eficiente de las condiciones del cultivo. Según las necesidades de cada tipo de planta, se utilizan diferentes sensores, como los de humedad del suelo, temperatura ambiental, humedad relativa, luminosidad, y pH del suelo. Estos sensores proporcionan datos críticos que permiten optimizar el uso de recursos, mejorar la calidad de los alimentos y aumentar la producción agrícola. Al analizar la información recopilada, los agricultores pueden tomar decisiones informadas, ajustando el riego, la fertilización y otras prácticas para maximizar la eficiencia y sostenibilidad de la producción.

6.8 Interfaz Web en aplicaciones IoT

En los sistemas IoT, una interfaz web es una herramienta esencial para visualizar y gestionar los datos recopilados por dispositivos y sistemas embebidos. Estas interfaces permiten a los usuarios interactuar con la información de forma intuitiva y accesible, mostrando los datos en diversos formatos visuales, como gráficos, tablas y diagramas. Esto facilita no solo el análisis, sino también la comprensión de grandes volúmenes de datos, haciendo que las decisiones sean más precisas y basadas en información en tiempo real [25].

Para que estas interfaces sean accesibles desde cualquier lugar, se alojan en la nube, lo que garantiza que los usuarios puedan acceder a la información de

forma remota y segura. Esto es especialmente importante en aplicaciones IoT que requieren monitoreo y control constante, como la gestión de sistemas agrícolas o el control de dispositivos domésticos inteligentes.

Además de la visualización de datos, las interfaces web ofrecen funcionalidades para controlar dispositivos de manera remota. Esto se logra mediante comandos que se envían desde la interfaz a los sistemas embebidos. Por ejemplo, en un entorno de hogar inteligente, los usuarios pueden encender o apagar luces con solo presionar un botón en la interfaz. En el ámbito agrícola, si los sensores detectan que la humedad del suelo ha caído por debajo de un nivel crítico, la interfaz puede permitir que un usuario active manualmente un sistema de riego o programar que este proceso se realice automáticamente. Estas capacidades hacen que las interfaces web sean herramientas poderosas para la automatización y el control remoto, mejorando la eficiencia y la capacidad de respuesta en una amplia variedad de aplicaciones.

La interfaz web se comunica con los dispositivos IoT y las bases de datos utilizando protocolos de comunicación seguros, como HTTPS, y mediante APIs (Interfaces de Programación de Aplicaciones). Las APIs sirven como intermediarios que permiten que la interfaz envíe y reciba información de manera estructurada y eficiente. Por ejemplo, cuando un usuario interactúa con la interfaz para enviar un comando a un dispositivo IoT, la solicitud se envía a través de una API, que a su vez se encarga de transmitirla al dispositivo correspondiente. De manera similar, los datos recolectados por los sensores se envían a la nube, donde se almacenan en bases de datos; la interfaz puede acceder a estos datos usando APIs para mostrarlos al usuario en tiempo real. Este sistema de comunicación asegura que todas las interacciones entre la interfaz, los dispositivos y la base de datos sean rápidas, seguras y bien organizadas [26].

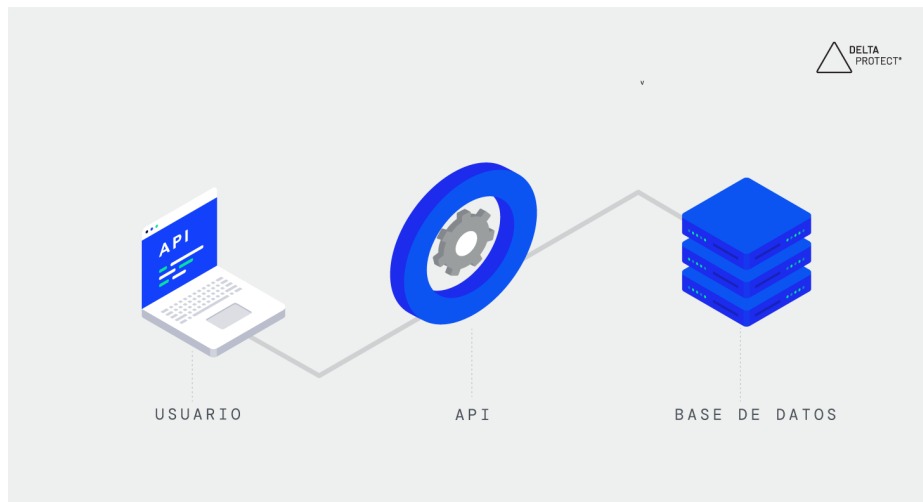


Figura 1. Flujo de información utilizando API. Imagen tomada de [34].

6.9 Bases de datos

Las bases de datos son un pilar fundamental en el manejo de la información en sistemas informáticos. Se pueden entender como un conjunto estructurado de datos organizados de manera que faciliten su almacenamiento, consulta y actualización. En esencia, su propósito es permitir que la información se gestione de forma eficiente, asegurando su integridad y disponibilidad cuando se necesite.

Para administrar estos datos, se utilizan Sistemas de Gestión de Bases de Datos (SGBD), que actúan como intermediarios entre el usuario y la base de datos. Estos sistemas permiten realizar operaciones como agregar, modificar, eliminar y consultar información, garantizando además la seguridad y la consistencia de los datos almacenados.

6.9.1 Tipos de bases de datos

A lo largo del tiempo, han surgido distintos modelos de bases de datos, cada uno diseñado para responder a necesidades específicas. Algunos de los principales son:

- Bases de datos relacionales

Las bases de datos relacionales son las más utilizadas en la actualidad debido a su estructura organizada en tablas. Cada tabla se compone de filas (registros) y columnas (atributos), lo que permite almacenar datos de manera ordenada y con relaciones bien definidas entre distintas tablas.

Este modelo se basa en el uso de llaves primarias y foráneas para vincular información de diferentes tablas, asegurando la integridad de los datos. Además, su manipulación se realiza a través del lenguaje SQL (Structured Query Language), que facilita la consulta y gestión de la información. Ejemplos de bases de datos relacionales incluyen MySQL, PostgreSQL, SQL Server y Oracle.

- Bases de Datos Jerárquicas

Este modelo organiza los datos en una estructura de árbol, donde cada registro tiene una relación de dependencia con un único registro superior. Es eficiente en sistemas con información estructurada de manera fija, pero su rigidez lo hace menos flexible en comparación con las bases de datos relacionales.

- Bases de Datos en Red

A diferencia del modelo jerárquico, las bases de datos en red permiten que un mismo registro tenga múltiples relaciones con otros registros, formando una estructura de nodos interconectados. Este enfoque es útil cuando se requiere modelar relaciones complejas entre datos.

- Bases de Datos Orientadas a Objetos

En este modelo, la información se almacena en forma de objetos, de manera similar a los lenguajes de programación orientados a objetos como Java y Python. Esto permite que cada objeto conserve sus atributos y métodos, facilitando la integración con aplicaciones que utilizan este paradigma.

- Bases de Datos NoSQL

Las bases de datos NoSQL fueron diseñadas para manejar grandes volúmenes de información que no necesariamente tienen una estructura fija. Se dividen en distintas categorías según su forma de almacenar los datos:

- Clave-valor: Guardan la información en pares clave-valor (ejemplo: Redis).
- Basadas en documentos: Almacenan datos en formatos flexibles como JSON o BSON (ejemplo: MongoDB).
- De grafos: Modelan relaciones entre datos de manera visual y eficiente (ejemplo: Neo4j) [27].

7. METODOLOGIA

En este capítulo se describe el diseño e implementación del sistema, detallando cada uno de sus componentes principales: el dispositivo en tierra con sensores, el UAV, la base de datos y la interfaz web. Se especificarán los aspectos de hardware, software y los protocolos de comunicación utilizados en cada etapa del desarrollo. A continuación, se presenta un diagrama general del sistema, donde se visualizan sus diferentes componentes y la forma en que interactúan entre sí.

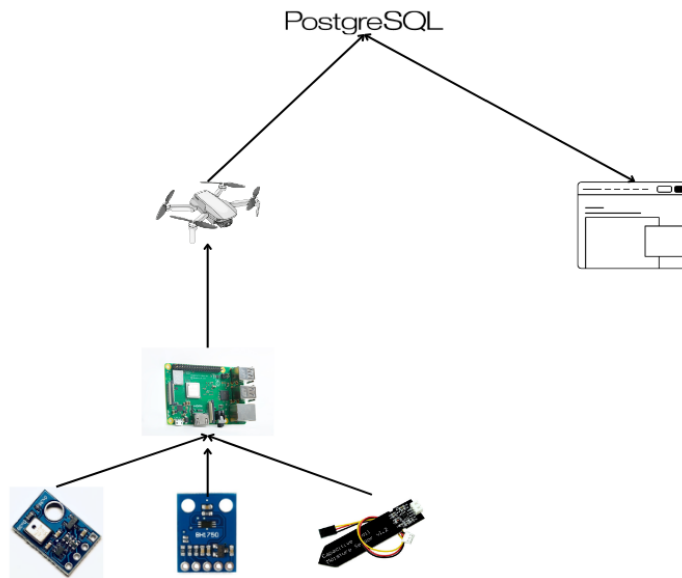


Figura 2. Diagrama general del sistema. Elaboración propia

7.1 Sistema embebido

El sistema embebido está compuesto por una Raspberry Pi 3B+ y tres sensores: dos digitales y uno analógico, los cuales están conectados a través del protocolo de comunicación I2C.

Los sensores digitales, AHT10 (sensor de temperatura y humedad) y BH1750 (sensor de luminosidad), están conectados directamente a los puertos SDA y SCL de la Raspberry Pi, permitiendo una comunicación eficiente mediante el bus I2C.

La elección de estos sensores se basó en su bajo costo, tamaño compacto, facilidad de integración gracias al protocolo I2C y su buena eficiencia en la medición de variables ambientales. Estas características los hacen especialmente adecuados para cultivos de mediana y baja escala, donde se requiere un sistema confiable y accesible que no demande infraestructura compleja.

Por otro lado, el sensor analógico, un sensor capacitivo de humedad del suelo, requiere de un convertidor Analógico-Digital (ADC) para transformar la señal analógica en valores digitales que puedan ser interpretados por la Raspberry Pi. Este ADC retransmite la señal al microprocesador mediante, facilitando su procesamiento y conexión al dispositivo.

Este sistema embebido se encarga de capturar datos en tiempo real y transmitirlos para su posterior análisis, formando una parte esencial del sistema general de monitoreo.

7.1.1 Sensores

A continuación, se presenta un diagrama del hardware de los sensores y la Raspberry Pi.

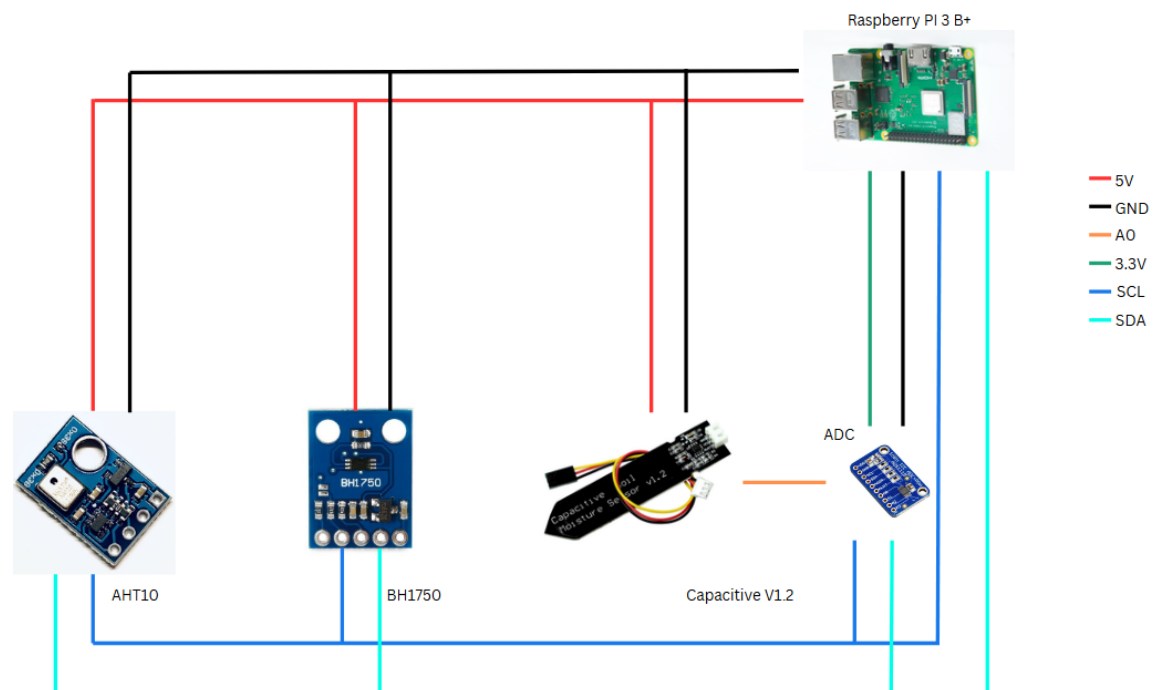


Figura 3. Diagrama de conexiones del sistema embebido. Elaboración propia.

AHT10

El sensor AHT10 es un dispositivo de medición de temperatura y humedad ambiental, diseñado para aplicaciones de monitoreo ambiental con alta precisión y estabilidad a largo plazo.

Características Principales:

- Rango de medición: Proporciona un rango desde -40 C° a 85 C° , con una precisión de $\pm 0.3\text{ C}^{\circ}$ para la temperatura. Para la humedad relativa tiene un rango de 0% a 100% RH (Relative humidity) con una precisión de $\pm 2\%$ RH.
- Voltaje de Operación: El voltaje permitido va desde 1.8V a 6V.
- Protocolo de comunicación: I2C [28].

Estos parámetros de temperatura y humedad ambiental permite conocer las condiciones micro climáticas que influyen directamente en procesos como la transpiración y la respiración de las plantas. Estos datos ayudan a prevenir estrés térmico o hídrico y a optimizar prácticas como el riego o la ventilación.

Se implementó la siguiente configuración en la Raspberry Pi para la medición de temperatura y humedad. Gracias a su bajo consumo energético, el sensor AHT10 es ideal para una conexión directa a la Raspberry Pi sin necesidad de circuitos adicionales, permitiendo una integración eficiente y estable en el sistema.

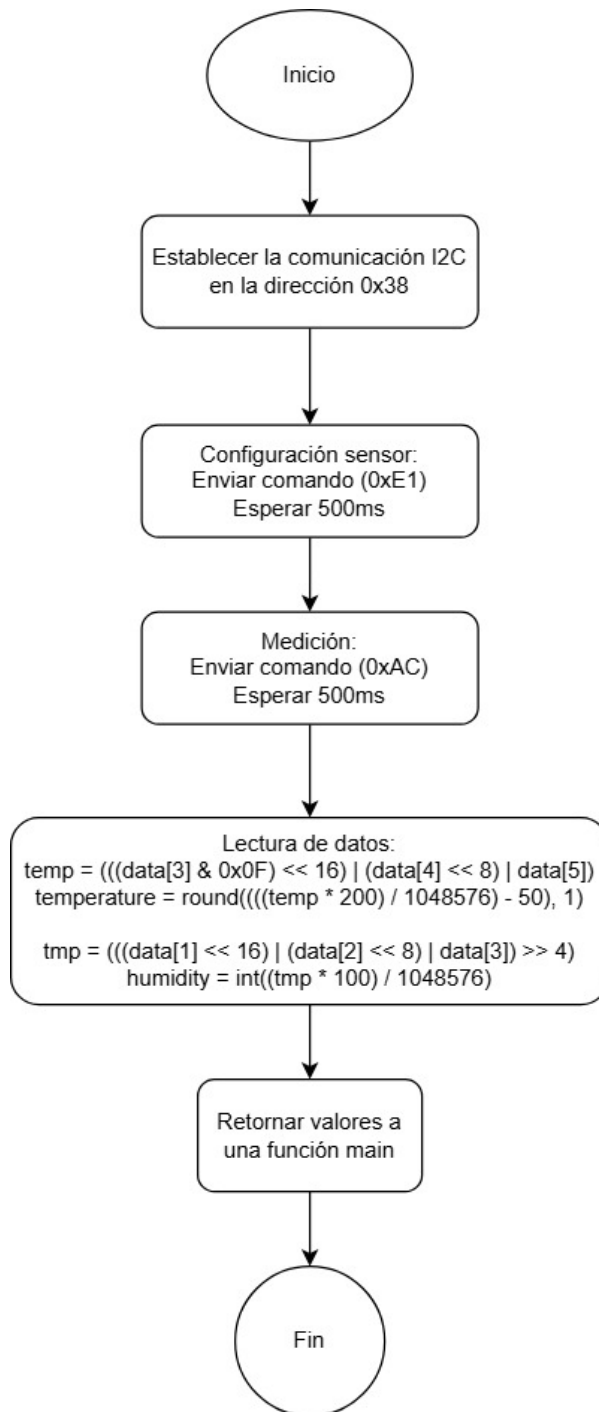


Figura 4. Pseudocódigo sensor de temperatura y humedad.
Elaboración propia.

BH1750

El sensor BH1750 mide la luminosidad ambiental en escala LUX la cual significa cantidad de luz (lumen) que hay en un metro cuadrado. Este pequeño dispositivo fue diseñado para aplicaciones de bajo consumo teniendo las siguientes características:

- Rango de medición: Entrega un rango desde 1 a 65535 Lux
- Voltaje de operación: Permite un voltaje desde 3.3V hasta 5V.
- Protocolo de comunicación: I2C [29].

Esta medición permite conocer la cantidad de luz disponible para el proceso de fotosíntesis. Este dato resulta esencial para decidir la ubicación óptima de los cultivos, definir el uso de mallas de sombra y programar la iluminación artificial en invernaderos, asegurando así un desarrollo adecuado de las plantas.

A continuación, se presenta la configuración del sensor:

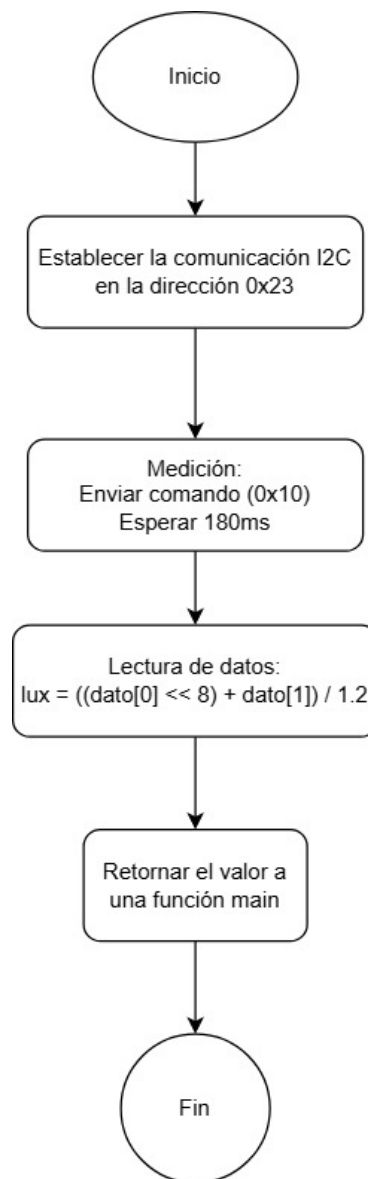


Figura 5. Pseudocódigo sensor de luminosidad. Elaboración propia.

Capacitive Soil Moisture Sensor V1.2

El sensor capacitivo de humedad del suelo es ampliamente utilizado en aplicaciones agrícolas y de monitoreo ambiental de largo plazo debido a su durabilidad, precisión y resistencia a la corrosión. A diferencia de los sensores resistivos, que pueden degradarse con el tiempo debido a la exposición al agua y los minerales del suelo, los sensores capacitivos ofrecen una medición más estable y confiable sin contacto directo con el medio.

Gracias a su bajo consumo energético y compatibilidad con microcontroladores y sistemas embebidos, como la Raspberry Pi, este sensor es ideal para sistemas de riego automatizado, optimización del consumo de agua y estudios sobre las condiciones del suelo en cultivos de diversa índole.

Algunas de las características más importantes a tener en cuenta son las siguientes:

- Voltaje de operación: Permite un voltaje de 3.3V a 5V.
- Rango de medición: Su voltaje máximo en la salida depende del voltaje con el que se está alimentando, siendo un rango de 0V a 5V si se alimenta con 5V. Por otra parte, si se alimenta con un voltaje de 3.3V su rango varía de 0V a 3.3V.

Para la implementación del sensor en el sistema, es necesario un convertidor ADC, el cual recibe una señal analógica de voltaje a través de un pin y la convierte en una señal digital con una resolución de 16 bits y una velocidad de hasta 860 muestras por segundo. Posteriormente, el ADC transmite la señal a la Raspberry Pi mediante el protocolo I2C, permitiendo su procesamiento.

En lo que respecta a la medición del suelo, este sensor proporciona información sobre el contenido de agua en el sustrato, lo que permite determinar el momento y la cantidad adecuada de riego. Su diseño capacitivo le otorga una mayor resistencia a la corrosión en comparación con los modelos resistivos, prolongando así su vida útil en condiciones de campo.

A continuación, se presenta el diagrama que describe el proceso de configuración del sensor, incluyendo la selección del canal de entrada y la conversión de la señal analógica a digital mediante el ADC:

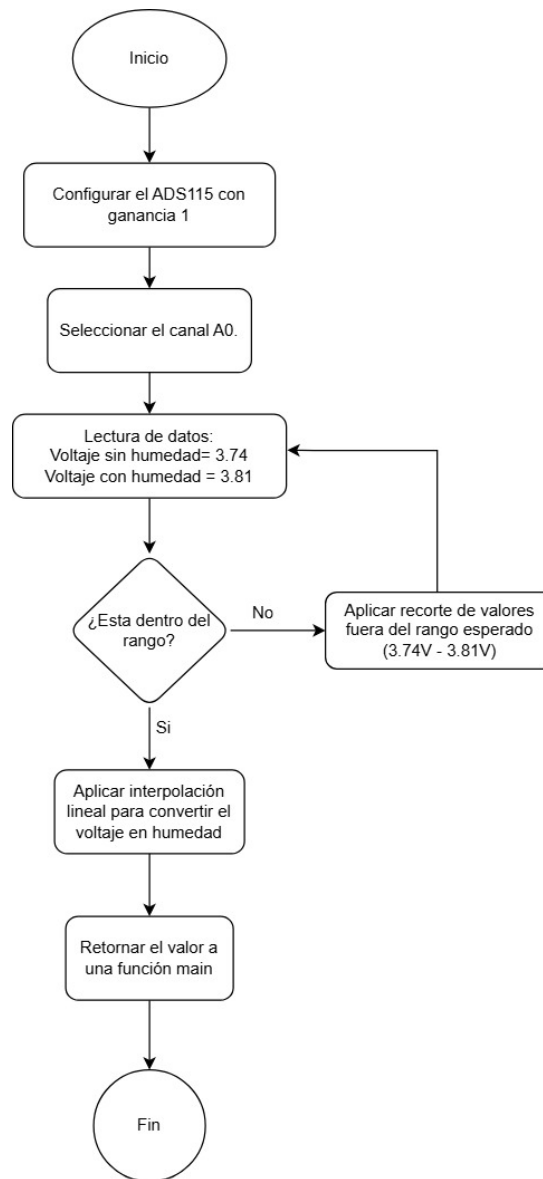


Figura 6. Pseudocódigo sensor capacitivo de humedad.
Elaboración propia

El sensor capacitivo de humedad del suelo no entrega un rango de voltaje de 0V a 5V como se esperaría, sino que opera en un intervalo más reducido. Al medir su salida, se observa que en suelo seco genera un voltaje de aproximadamente 3.74 V, mientras que en suelo completamente húmedo alcanza cerca de 3.81 V, lo que corresponde a una variación de solo 70 mV. Aunque este rango es limitado, el conversor ADS1115 configurado con ganancia 1 (± 4.096 V) ofrece una resolución de 0.125 mV por bit, lo que permite representar esta variación en alrededor de 560 niveles digitales

distintos. Gracias a esta precisión, es posible ajustar las mediciones en el código para que el sistema interprete correctamente los niveles de humedad. Para ello, se aplica una normalización del voltaje dentro de este rango y posteriormente se utiliza una interpolación lineal para convertir el valor obtenido en un porcentaje de humedad.

7.1.2 Fuente de Alimentación

Uno de los desafíos más importantes en los sistemas embebidos es la gestión eficiente del consumo energético y el suministro de una alimentación estable y adecuada. Para garantizar el correcto funcionamiento del sistema, es fundamental seleccionar una fuente de alimentación que cumpla con los requerimientos de potencia y estabilidad.

Existen diversas opciones para alimentar el dispositivo, entre las cuales se encuentran las siguientes:

- **Baterías Alcalinas:** Las baterías alcalinas son ampliamente utilizadas en dispositivos electrónicos de bajo consumo, ya que proporcionan un voltaje estable y son de fácil acceso. Sin embargo, presentan limitaciones significativas cuando se requiere alimentar sistemas con un mayor consumo energético.

En este caso, la Raspberry Pi requiere 5V y puede llegar a consumir hasta 3A, dependiendo de los periféricos conectados y la carga del sistema. Esto representa un problema, ya que la mayoría de las baterías alcalinas comerciales tienen 1.5V por celda, lo que significa que para alcanzar 5V se necesitarían al menos tres o cuatro celdas en serie. Además, la corriente suministrada por estas baterías es limitada, con valores típicos de 500mA a 1A, lo que podría ser insuficiente para alimentar la Raspberry Pi de manera estable.

- **Baterías LiPo:** Este tipo de baterías se utilizan ampliamente en el campo de la electrónica, especialmente en aplicaciones donde los dispositivos requieren altas corrientes con variaciones bruscas de demanda energética. A diferencia de las baterías de Litio, las LiPo tienen una mayor capacidad de descarga, lo que las hace ideales para sistemas que necesitan potencia instantánea y sostenida.

El uso de baterías LiPo en un sistema embebido conlleva diversos riesgos debido a su composición y características operativas. Estas baterías están formadas por celdas individuales, cada una con un rango de voltaje específico que varía según su capacidad. Si el voltaje de una celda cae por debajo del 30% de su capacidad, puede sufrir daños irreversibles, reduciendo drásticamente su vida útil.

Además, las baterías LiPo requieren un cargador especializado para evitar sobrecargas, ya que una carga inadecuada puede provocar sobrecalentamiento, hinchamiento o incluso incendios.

Si bien es posible alimentar una Raspberry Pi con este tipo de baterías, sería necesario un sistema de regulación de voltaje para garantizar un suministro estable de 5V constantes. No obstante, esto implicaría un enfoque distinto al habitual y no aprovecharía las ventajas principales de las baterías LiPo, que están diseñadas para aplicaciones de alta demanda energética con descargas rápidas y sostenidas, como drones y vehículos de radio control.

- **Baterías de Litio:** Las baterías de litio se utilizan ampliamente en dispositivos que requieren un suministro de energía estable y de larga duración, como celulares, tabletas y power banks. Se caracterizan por ser ligeras, compactas y de alta capacidad, además de tener la ventaja de ser recargables de forma rápida y eficiente, lo que permite reutilizarlas múltiples veces sin una pérdida significativa de rendimiento.

En este caso, el uso de una power bank resulta una opción ideal para alimentar una Raspberry Pi, ya que incorpora un sistema de conversión que eleva el voltaje de las celdas de litio (3.7V) a 5V estables, garantizando una alimentación continua y confiable. Además, las power banks pueden recargarse fácilmente mediante puertos USB, lo que permite mantener el sistema operativo sin interrupciones prolongadas. La duración de la energía suministrada dependerá tanto de la capacidad de la power bank como del consumo energético del sistema.

Por consiguiente, se decidió utilizar una power bank de 20,000 mAh con una salida de 5.1V y una potencia de hasta 18W, proporcionando una fuente de energía portátil y confiable para la Raspberry Pi.

Dado que la Raspberry Pi tiene un consumo estimado de 2.5A a 5.1V, se ha calculado que la PowerBank podrá mantener el sistema funcionando durante aproximadamente 7 horas antes de agotarse por completo. Esto la convierte en una solución eficiente para aplicaciones que requieren movilidad o respaldo energético durante períodos prolongados.

7.1.3 Diseño carcasa

Para la implementación de la Raspberry Pi en un cultivo, es fundamental contar con una carcasa de protección que resguarde los dispositivos electrónicos y proporcione una adecuada ventilación interna. Además, debe permitir la correcta ubicación de los sensores para garantizar mediciones precisas.

Por ejemplo, el sensor de luminosidad debe estar colocado en una posición estratégica dentro de la carcasa para obtener lecturas fiables, mientras que el sensor de temperatura debe ubicarse en un área con buen flujo de aire y contacto directo con el exterior para medir con precisión las condiciones ambientales. Adicionalmente, el sensor capacitivo de humedad del suelo debe situarse en la parte inferior de la estructura para que pueda enterrarse en la tierra y realizar mediciones exactas del nivel de humedad del sustrato.

Tomando en cuenta estos aspectos, se diseñó el siguiente modelo de carcasa.

- **Ubicación Raspberry**

Para comenzar, se diseñó una carcasa que se ajusta perfectamente a la Raspberry Pi 3 B+, teniendo en cuenta las dimensiones exactas y la ubicación de los orificios para atornillarla de forma segura. Además, se incorporó una rejilla de ventilación para mejorar la circulación del aire y evitar el sobrecalentamiento, garantizando así un funcionamiento estable del sistema.

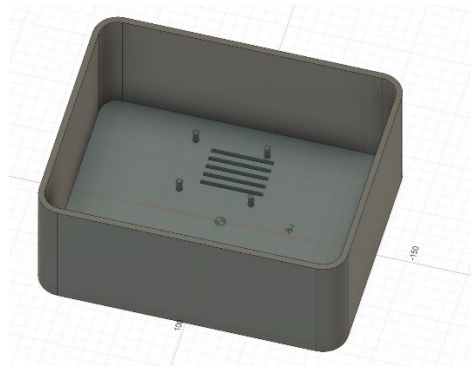


Figura 7. Carcasa con el espacio para la Raspberry Pi.
Elaboración propia

- **Ubicación power bank**

Se diseñó un espacio específico debajo de la Raspberry Pi en forma de compartimento rectangular para alojar y sostener de manera segura la power bank. Además, se dejó un espacio en la parte derecha de la carcasa para facilitar el acceso a los conectores.

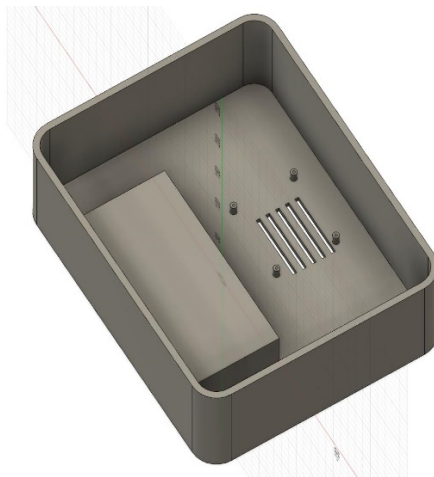


Figura 8. Carcasa con el espacio para la Powerbank.
Elaboración propia

- **Parte frontal.**

La parte superior de la carcasa fue diseñada con una rejilla de ventilación para permitir la circulación de aire dentro del sistema. Además, se incorporó un soporte específico para la instalación del sensor de temperatura y humedad, asegurando su correcta ubicación en un área con buen flujo de aire para obtener mediciones precisas del entorno

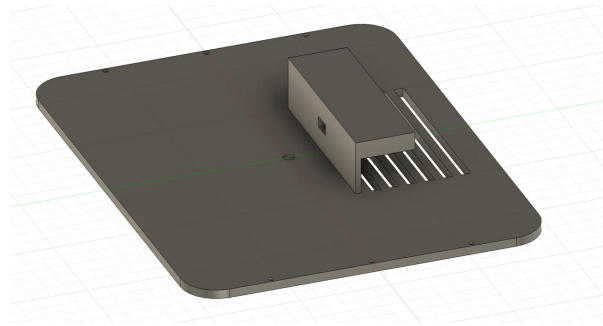


Figura 9. Parte frontal carcasa. Elaboración propia.

Una vez que los sensores están en funcionamiento, ubicados en la carcasa y con una fuente de alimentación, se implementa un script el cual recopile todas las mediciones de manera organizada para almacenarlas de manera temporal y continuar con las siguientes etapas del sistema.

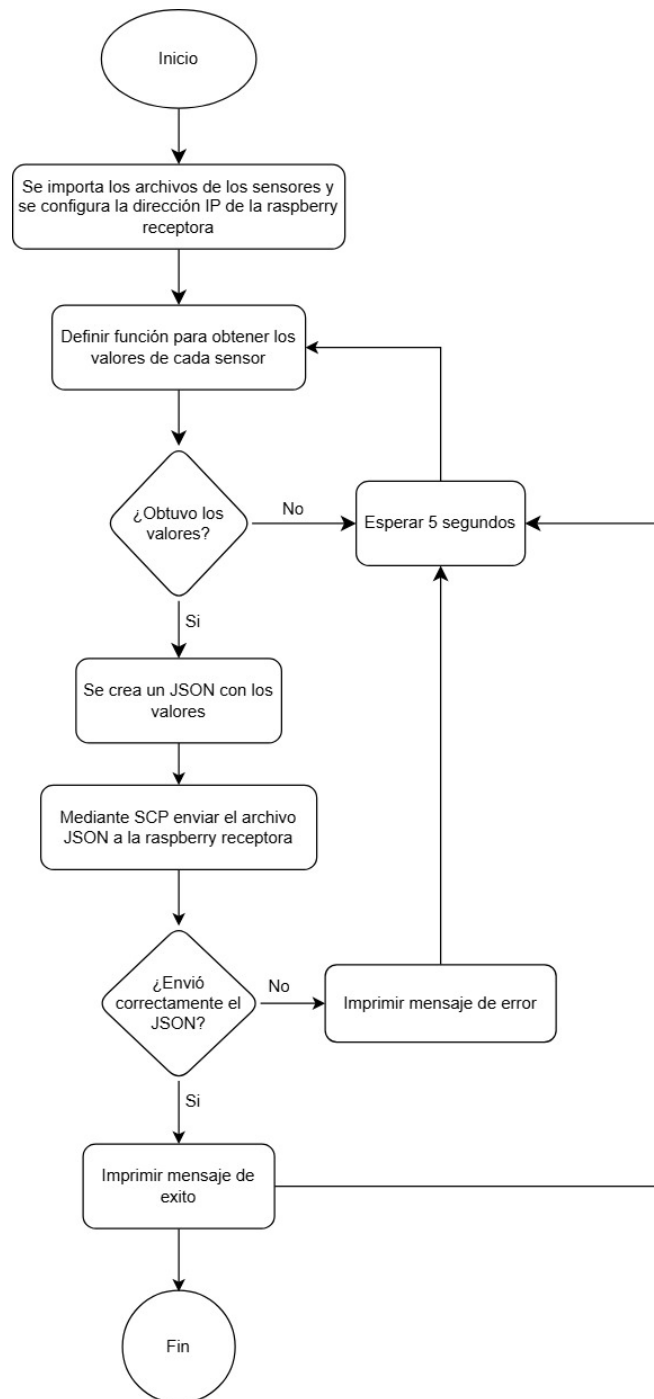


Figura 10. Pseudocódigo general de la Raspberry Pi.
Elaboración propia.

La Raspberry Pi receptora está ubicada en el UAV y tiene la función de almacenar temporalmente los datos recopilados antes de reenviarlos para su procesamiento.

Su propósito es y optimizar la transmisión de los datos, como se detalla en la siguiente sección

7.2 Dispositivo UAV

El dispositivo UAV es una parte fundamental en el sistema para la recolección de datos, brindando la comunicación unidireccional de los datos hacia la nube mediante una Raspberry PI y un módulo 3G.

7.2.1 Dron F40

El Dron F450 es usado en diferentes campos de la electrónica, cumpliendo diferentes propósitos. Es un dron con una amplia configuración tanto como en su Hardware como en su Software. Los componentes principales del dron son los siguientes:

- **Pixhawk 2.4.8**

El Pixhawk 2.4.8 es un controlador de vuelo ampliamente utilizado en sistemas de drones debido a su capacidad de procesamiento y versatilidad. Su función principal es gestionar y procesar las señales provenientes de los distintos sensores y módulos del dron para garantizar un vuelo estable y preciso.



Figura 11. Controlador Pixhawk. Imagen tomada de [35].

Este controlador cuenta con dos procesadores, múltiples entradas y salidas para la integración de diversos periféricos, y es compatible con hasta 14 motores mediante señales PWM. Además, ofrece una amplia variedad de interfaces de comunicación, incluyendo I2C, UART, SPI y

CAN, lo que permite la conexión de sensores, sistemas de navegación y otros módulos adicionales.

- **Marco F450**

El chasis F450 está compuesto por una placa central que integra una distribución de potencia mediante una PDB (Power Distribution Board), con conexiones XT60 y pads de soldadura para alimentar los ESC y motores. Cuenta con cuatro brazos estructurales diseñados para alojar motores y hélices, asegurando estabilidad y eficiencia aerodinámica. En la parte inferior dispone de un compartimiento para la batería, optimizando el centro de gravedad, y un tren de aterrizaje de cuatro patas que proporciona estabilidad durante el despegue y aterrizaje.



Figura 12. Marco F450. Imagen tomada de [30].

- **Batería LiPo 11,1V 5200mAh**

Esta batería está compuesta por tres celdas, cada una con un voltaje nominal de 3.7V, lo que da un total de 11.1V. Su rango de voltaje seguro oscila entre 12.4V en carga completa y un mínimo recomendado de 10V, evitando descargas excesivas que podrían dañar permanentemente las celdas. Si una celda cae por debajo de 3V, puede volverse inservible, lo que afecta el rendimiento y la vida útil de la batería.

Las baterías LiPo destacan por su alta capacidad de descarga, lo que las hace ideales para aplicaciones donde los motores requieren cambios bruscos de velocidad y, por ende, variaciones en la demanda de corriente. Su capacidad para suministrar corrientes elevadas en cortos períodos las convierte en la mejor opción para drones y otras aplicaciones que exigen una respuesta energética rápida y eficiente.



Figura 13. Batería LiPo 11.1V 5200mAh.
Imagen tomada de [36].

- **Cargador batería LiPo**

El cargador de baterías LiPo es clave para garantizar una carga segura y prolongar la vida útil de la batería. Su funcionamiento es bastante inteligente: detecta automáticamente cuántas celdas tiene la batería y ajusta el voltaje y la corriente para que cada una reciba la carga adecuada. Esto evita problemas como sobrecargas o desequilibrios entre las celdas, que podrían dañar la batería o incluso hacer que se sobrecaliente. Gracias a este sistema, la carga se distribuye de manera uniforme, asegurando un rendimiento óptimo y mayor seguridad en su uso.



Figura 14. Cargador baterías LiPo.
Imagen tomada de [30].

- **Módulo de potencia V1.0**

El módulo de potencia V1.0 se encarga de suministrar energía al dron, proporcionando dos salidas principales: una salida no regulada que entrega el voltaje de la batería a los ESC (Controladores de velocidad) para alimentar los motores, y una salida regulada de 5.3V conectada al puerto 'Power' del Pixhawk, permitiendo su alimentación y la monitorización del voltaje y corriente de la batería.



Figura 15. Módulo de potencia GM V1.0. Imagen tomada de [31].

- **ESC y motores**

Los controladores electrónicos de velocidad (ESC) están conectados a los motores mediante tres cables de fase, cuyo orden define el sentido de giro del motor. Los ESC reciben alimentación a través de la placa de distribución de potencia (PDB) mediante conectores XT60 y gestionan la velocidad de los motores en función de las señales de control. Cada ESC cuenta con una salida de señal de tres cables (señal, VCC y tierra), la cual se conecta a los puertos PWM del Pixhawk, permitiendo la comunicación para el control preciso de la velocidad y dirección de los motores.



Figura 16. ESC, motores y hélices. Imagen tomada de [31]

- **Modulo receptor y control**

El sistema de control del dron utiliza un módulo receptor de radiofrecuencia para la comunicación con el transmisor. En este caso, se emplea el receptor FlySky FS-IA6B, el cual opera en la banda de 2.4 GHz y está conectado al Pixhawk a través del puerto PPM, permitiendo la transmisión de múltiples canales de control mediante un único cable.

El control remoto utilizado es el FlySky FS-I6X, el cual establece la comunicación inalámbrica con el receptor FS-IA6B, enviando las señales necesarias para el control de vuelo del dron.



Figura 17. Control y receptor. Imagen tomada de [32].

Después de identificar e implementar las diferentes partes del dron se tiene el siguiente montaje.



Figura 18. Dron F450. Elaboración propia

Para configurar el dron, lo primero es preparar el Pixhawk 2.4.8. Para ello, es necesario descargar e instalar el software Mission Planner, que facilita la gestión y calibración del controlador de vuelo. Una vez instalado, simplemente se conecta el Pixhawk al computador a través de un cable USB, y el programa lo detecta automáticamente. A partir de ahí, se pueden ajustar los parámetros necesarios para garantizar un vuelo estable y preciso.

Los parámetros ajustados en el Mission Planner son los siguientes:

- Configuración del frame (marco): En este apartado se configura que tipo de dispositivo estamos configurando, en este caso se selecciona un cuadricóptero.
- Calibración del acelerómetro: Se activa la opción de calibrar acelerómetro en la pestaña de Acer Calibration. Para completarla, es necesario mover el dron en las diferentes posiciones indicadas por el software.
- Calibración de la brújula: En la pestaña "Compass", el software solicitará mover el dron en posiciones específicas para realizar la calibración correctamente.
- Calibración control RC: Generalmente, el control remoto ya viene emparejado con el módulo de radiofrecuencia. En este proceso, se configura en modo PPM dentro del apartado "Output Mode" del control. Luego, se calibra desde la aplicación y se verifica el movimiento de los joysticks en la interfaz.
- Calibración ESC: Para la calibración de los motores se puede realizar mediante dos formas, desde la aplicación o desde el control RC. En este caso se realizó mediante el control, el cual se debe seguir un protocolo de pasos. Primero, se enciende el control y se sube la palanca del acelerador al máximo. Luego, se enciende el dron y este detecta la señal de calibración. A continuación, se apaga y vuelve a encender el dron, momento en el que emite un pitido indicando que ha entrado en modo de calibración. Para finalizar, se baja la palanca del acelerador al mínimo y, tras unos segundos, se escucha otra serie de pitidos que confirman que la calibración se completó con éxito.

Después de completar las calibraciones iniciales, es necesario configurar los modos de vuelo del dron, los cuales dependen de los componentes instalados. Algunos modos, como Loiter, requieren GPS para funcionar correctamente, mientras que otros pueden operar sin él. En este caso, el dron se configura con los modos AltHold y Stabilize, los cuales pueden intercambiarse utilizando las palancas superiores del control RC. El modo AltHold permite que el dron mantenga automáticamente la altitud, mientras que el piloto tiene control sobre la inclinación y el giro. Por otro lado, el modo Stabilize mantiene el dron nivelado de forma automática, pero el piloto puede controlar la altitud, el giro y la inclinación. Para configurar estos modos en Mission Planner, es necesario acceder a la pestaña Flight Modes, donde se pueden asignar hasta seis modos de vuelo a las diferentes posiciones del interruptor del control RC, permitiendo alternarlos fácilmente durante el vuelo.

7.2.2 Raspberry Pi

Para lograr que el UAV pueda enviar datos a la nube y comunicarse con el sistema embebido en tierra, se decidió utilizar una Raspberry Pi 3B+. Este dispositivo es una excelente opción porque es compacto, ligero y tiene la capacidad de ejecutar múltiples procesos de forma eficiente. Una de sus principales ventajas es que permite correr código en Python, lo que facilita la implementación de algoritmos para el procesamiento y transmisión de datos, asegurando un flujo de información más fluido y estable. Además, cuenta con puertos USB, lo que abre la posibilidad de conectar periféricos como módems o unidades de almacenamiento externo, ampliando sus funcionalidades. También puede actuar como un punto de acceso WiFi, permitiendo la conexión con la Raspberry Pi del sistema embebido en tierra para mantener una comunicación constante. Gracias a su peso reducido, se integra fácilmente en el UAV sin afectar su rendimiento.

Para alimentar la Raspberry Pi desde el sistema eléctrico del dron, es necesario utilizar un convertidor DC-DC, el cual se encarga de reducir el voltaje de la batería del dron a un nivel adecuado para la Raspberry. En este caso, la batería del dron proporciona aproximadamente 11.7V, un valor que excede el rango de operación de la Raspberry, por lo que se requiere este convertidor para garantizar una alimentación segura y estable. La entrada del convertidor se conecta a la salida del módulo de potencia GM V1.0, la cual va conectada también al marco del dron. Para suministrar energía a la Raspberry Pi, se ha integrado un cable micro USB en la salida del

convertidor, facilitando una conexión directa y eficiente. Esto asegura un suministro de energía continuo y estable, permitiendo el funcionamiento óptimo de la Raspberry Pi durante la operación del dron.



Figura 19. Convertidor DC-DC. Imagen tomada de [33].

Luego de establecer la alimentación de la Raspberry Pi en el dron se procede a realizar el proceso para que funcione dentro del sistema.

7.2.2.1 Configuración punto de acceso (AP)

Para asegurar una conexión estable entre ambas Raspberry Pi, se utilizó el protocolo Wi-Fi. Gracias a su módulo de doble banda, compatible con 2.4 GHz y 5 GHz, los dispositivos pueden comunicarse de manera más eficiente, evitando interferencias y manteniendo una conexión fluida y confiable.

En la Raspberry Pi del dron se creó un punto de acceso asignándole un nombre y una contraseña para permitir que la Raspberry Pi del sistema embebido se conecte a esa red. Posteriormente de establecer la conexión mutua entre las Raspberry Pi se procede a hacer él envió de datos como se ilustra en la Figura 10, se estableció una comunicación utilizando la tecnología de comunicación SCP la cual es ideal para él envió de archivos punto a punto mediante una red Wi-Fi. Al solo necesitar enviar un JSON periódicamente esta tecnología es una de las mejores.

7.2.2.2 Configuración Modem 3G

Luego de establecer la comunicación mediante una red local entre ambas Raspberry Pi se necesita enviar los datos recibidos a la nube. En base a

ese requerimiento se utilizó un Modem 3G USB ya que este nos brinda una conectividad estable a internet utilizando una tarjeta SIM.



Figura 20. Modem 3G conectado a la Raspberry Pi. Elaboración propia
Después de la conexión del hardware se necesita configurarlo en la Raspberry Pi, por lo que se realizaron los siguientes pasos.

1. Configuración PPP (Protocol Point to Point)

Para establecer una conexión a la nube se requiere el protocolo punto a punto, este protocolo funciona sobre TCP/IP y establece una conexión directa con un punto para el flujo de datos.

Como primer paso, se instaló Minicom, una herramienta que permite verificar el correcto funcionamiento del módem y su comunicación con PPP. Una vez validada la conexión, es necesario modificar y comprobar el archivo GPRS dentro de la carpeta chatscripts, ya que este archivo es el encargado de la configuración del módem y su conexión con la red móvil.

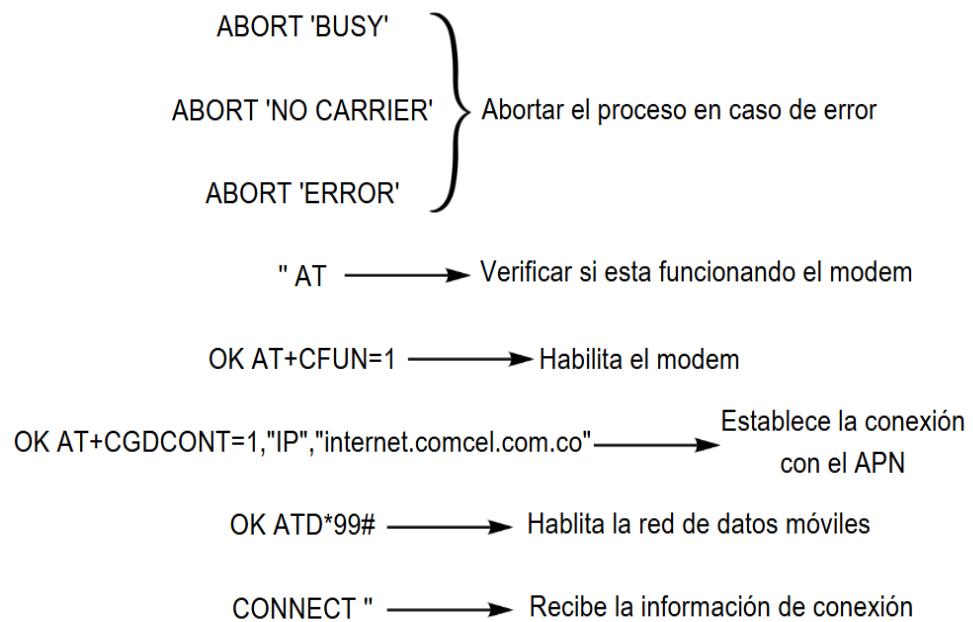


Figura 21. Comandos de conexión modem 3G. Elaboración propia

Una vez configurado el archivo GPRS en chatscripts, es necesario crear otro archivo de conexión con el mismo nombre en la carpeta del protocolo PPP, el cual contendrá los parámetros específicos para la conexión a Internet a través de este protocolo.

<code>/dev/ttyUSB2</code>	→	Especifica el puerto serie donde está conectado el módem
<code>115200</code>	→	Define la velocidad de transmisión de datos en baudios
<code>debug</code>	→	Habilita la salida de depuración para diagnosticar problemas
<code>nodetach</code>	→	Habilita la salida de depuración para diagnosticar problemas
<code>noauth</code>	→	No se requiere autenticación en la conexión.
<code>defaultroute</code>	→	Establece la conexión PPP como la ruta de red predeterminada
<code>usepeerdns</code>	→	Permite que el servidor DNS del proveedor sea utilizado automáticamente
<code>persist</code>	→	Mantiene la conexión intentando reconectar
<code>maxfail 0</code>	→	Evita que la conexión se cierre

Figura 22. Parámetros para conexión mediante PPP. Elaboración propia.

Luego de tener configurado los archivos GPRS inicia el servicio con el comando “sudo pon gprs” y se verifica el funcionamiento con “ifconfig ppp0”

2. Código en Python

La Raspberry Pi del sistema embebido genera datos en formato JSON y los almacena en una ubicación específica dentro de la Raspberry Pi instalada en el UAV. Para enviar estos datos a la nube, se desarrolló un script en Python que los recopila y los transmite mediante una solicitud HTTP POST.

Se eligió este método porque el envío de información no es continuo, sino que ocurre en momentos específicos. En lugar de mantener una conexión abierta todo el tiempo, HTTP POST permite enviar los datos solo cuando es necesario, optimizando así el uso de recursos y facilitando la integración con la base de datos en la nube.

A continuación, se muestra el código utilizado para realizar el requerimiento:

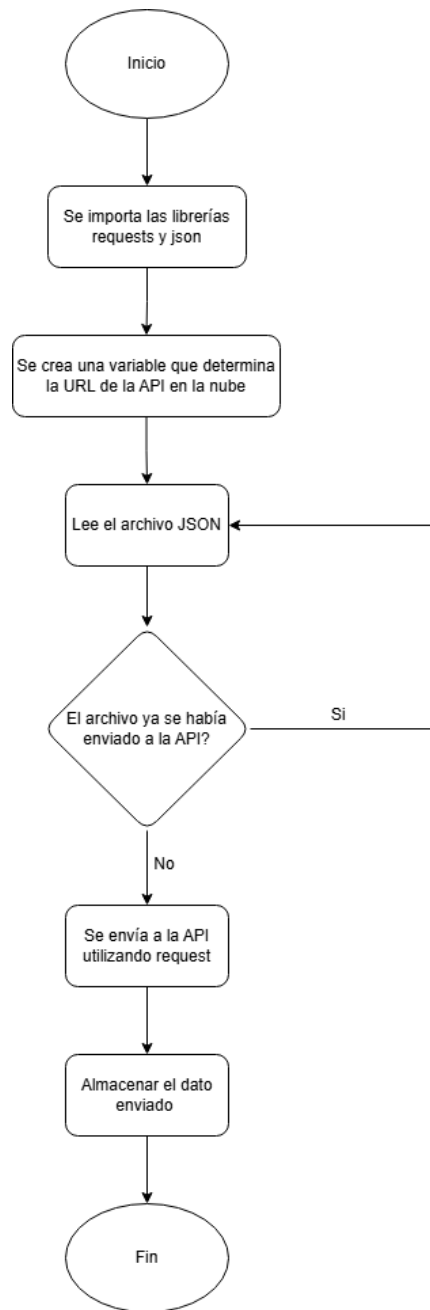


Figura 23. Pseudocódigo script Raspberry Pi UAV. Elaboración propia

Para completar el flujo de información, es necesario crear una aplicación web que permita visualizar en tiempo real los datos almacenados en la nube. Esta plataforma facilitará el acceso a la información enviada por la Raspberry Pi y brindará una interfaz clara para su consulta y análisis. En el siguiente capítulo, se explicará cómo se llevó a cabo su desarrollo, detallando la conexión con la

base de datos, la estructura de la página y las herramientas utilizadas para mostrar los datos de manera accesible y eficiente.

7.3 Desarrollo página web

Como parte de la implementación del sistema, se requiere diseñar una interfaz web que permita a los usuarios visualizar los datos de cada sensor en tiempo real. Para ello, se decidió emplear una combinación de tecnologías ampliamente utilizadas en el desarrollo web: HTML (Lenguaje de Marcado de Texto), CSS (Hojas de Estilo en Cascada) y JavaScript (JS).

El lenguaje HTML es la base estructural de cualquier página web. Su función principal es organizar y definir los elementos que componen la interfaz, como títulos, párrafos, botones, imágenes y formularios. Sin embargo, HTML por sí solo no es suficiente para crear una interfaz visualmente atractiva o interactiva. Para mejorar la presentación y disposición de los elementos en la página, se utiliza CSS. Este lenguaje permite modificar la apariencia de los objetos definidos en HTML, ofreciendo opciones como el cambio de colores, tipografías, tamaños, alineaciones, márgenes y efectos visuales. Además, CSS permite aplicar animaciones y transiciones que hacen que la interfaz sea más dinámica para el usuario.

Por último, se emplea JavaScript (JS). A diferencia de HTML y CSS, que se centran en la estructura y el diseño, JavaScript permite agregar interactividad y dinamismo a la aplicación. Con este lenguaje, es posible programar eventos en botones, gestionar formularios, actualizar información sin necesidad de recargar la página y establecer conexiones con la base de datos mediante APIs. Por ejemplo, en este sistema, JavaScript se encarga de obtener y mostrar los datos en tiempo real desde la base de datos, manejar la lógica de los botones y actualizar la interfaz sin interrupciones.

7.3.1 API WebSockets

La API desarrollada utilizando WebSockets cumple un papel fundamental en la comunicación entre la base de datos y la página web. A través de este servicio, se procesan las solicitudes generadas por la interfaz web, permitiendo el acceso, consulta y actualización de los datos en tiempo real.

Para su implementación, se utilizó Node.js, un entorno de ejecución que facilita la gestión de múltiples solicitudes de manera eficiente. Además, se incorporaron WebSockets, una tecnología que mantiene una conexión constante entre el servidor y la página web, evitando la necesidad de realizar múltiples peticiones cada vez que se requiere actualizar la información. Gracias a esto, los datos provenientes del sistema embebido y del UAV se reflejan en la web en tiempo real, asegurando una visualización fluida y sin interrupciones.

Esta combinación de tecnologías optimiza el uso de recursos del servidor y mejora la experiencia del usuario, ya que cada nuevo dato registrado en la base de datos se actualiza automáticamente en la web sin necesidad de recargar la página o hacer solicitudes manuales.

7.3.2 API Express

Para gestionar la comunicación entre el UAV y la nube, se desarrolló una API utilizando Express.js, un framework ligero y eficiente para Node.js. Esta API permite recibir y procesar solicitudes HTTPS, proporcionando una interfaz confiable para la transmisión de datos. En este caso, el UAV envía periódicamente los datos en formato JSON, asegurando que la información sea recibida, validada e insertada en la base de datos en la nube.

El uso de Express.js es ideal para este tipo de aplicaciones, ya que permite manejar solicitudes de manera eficiente sin necesidad de mantener una conexión abierta de forma permanente. En lugar de establecer una comunicación en tiempo real continua, el UAV solo envía los datos cuando es necesario, optimizando el consumo de recursos y facilitando la escalabilidad del sistema.

7.3.3 Base de datos

La base de datos es un pilar fundamental en este proyecto, ya que permite almacenar de manera estructurada y organizada la información recopilada del sistema embebido. Para gestionar estos datos de forma eficiente, fue necesario seleccionar el tipo de base de datos más adecuado. En este caso, se optó por bases de datos relacionales, ya que permiten organizar la información en tablas interconectadas mediante relaciones definidas por

llaves primarias y foráneas, garantizando coherencia en los datos almacenados.

- **Diseño y estructura de la base de datos**

El número de tablas necesarias en una base de datos depende de la cantidad de procesos a gestionar y del volumen de información a almacenar. En este proyecto, se definieron tres tablas principales, cada una con una función específica y relacionadas entre sí mediante llaves foráneas para asegurar una estructura organizada.

1. Tabla de usuarios

Para gestionar los accesos a la página web, se creó una tabla dedicada a los usuarios, que almacena información esencial para el inicio de sesión. Esta tabla contiene los siguientes campos:

- id (llave primaria): Identificador único para cada usuario.
- username: Nombre con el que el usuario accede al sistema.
- password: Clave de acceso almacenada de manera segura.
- email: Dirección de correo electrónico asociada.
- role: Define el nivel de permisos dentro del sistema (ejemplo: administrador, usuario estándar, etc.).

2. Tabla de sensores

Esta tabla almacena la información de cada sensor utilizado en el sistema embebido. Contiene los siguientes campos:

- id (llave primaria): Identificador único del sensor.
- nombre: Nombre descriptivo del sensor (ejemplo: temperatura, humedad, luminosidad, etc.).

- descripcion: Información adicional sobre el tipo de sensor y su función dentro del sistema

3. Tabla de mediciones

Esta es la tabla principal donde se almacenan las mediciones realizadas por los sensores en el sistema embebido. Sus campos son:

- id (llave primaria): Identificador único de cada registro de medición.
- int_procesos_data_id (llave foránea): Hace referencia a la tabla de sensores para identificar a qué sensor pertenece la medición.
- valor: Dato numérico correspondiente a la lectura del sensor.
- fecha: Fecha en que se realizó la medición.
- hora: Hora exacta en la que se tomó la lectura.

El campo int_procesos_data_id en la tabla de mediciones funciona como llave foránea, estableciendo una relación con la tabla de sensores. Esto permite identificar con precisión qué tipo de sensor generó cada medición sin necesidad de almacenar información redundante en la tabla de mediciones.

Para esta implementación se utilizó PostgreSQL, una base de datos que destaca por su flexibilidad y robustez. Una de sus ventajas es que puede configurarse y administrarse fácilmente de manera local, permitiendo estructurar los datos de forma eficiente a través de SQL. Además, migrarla a la nube resulta sencillo, ya que muchos servicios de hosting la soportan, lo que facilita su despliegue sin necesidad de grandes cambios en la configuración.

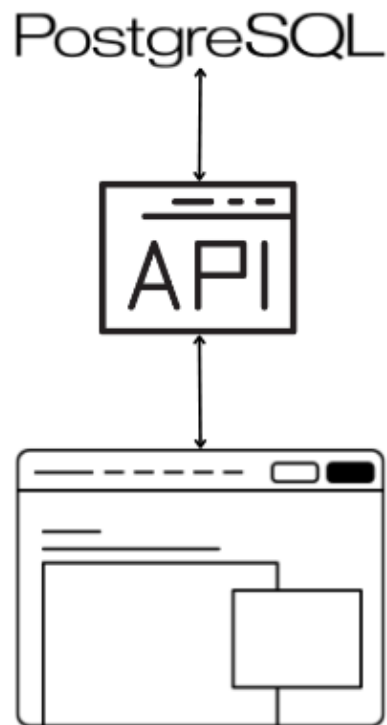


Figura 24. Flujo de datos interfaz web, API y base de datos. Elaboración propia.

Para comenzar, se diseñó la interfaz de manera local, asegurando que todo funcionara correctamente antes de su implementación en la web. El primer componente desarrollado fue la interfaz de inicio de sesión, que incluye los campos para ingresar usuario y contraseña. En la parte inferior se agregó un botón de acceso que, al ser presionado, envía la solicitud de autenticación a la base de datos a través de WebSocket. La API procesa la solicitud y ejecuta una consulta en lenguaje SQL para verificar si el usuario está registrado en la base de datos. Si las credenciales son correctas, la API responde con un mensaje de éxito, permitiendo así el acceso del usuario al sistema.

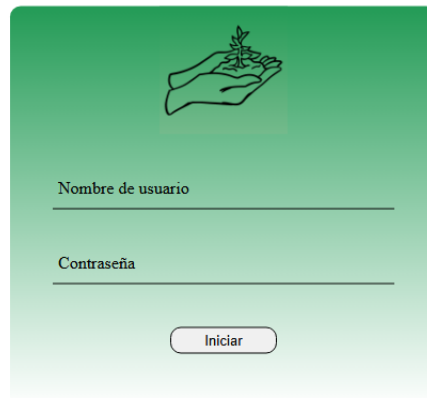


Figura 25. Interfaz inicio de sesión. Elaboración propia.

7.3.4 JSON Web Token

En el sistema, se implementó JWT (JSON Web Token) para gestionar la autenticación y mantener las sesiones de los usuarios de manera segura y eficiente. Cuando un usuario inicia sesión, el servidor verifica sus credenciales y, si son correctas, genera un token JWT firmado digitalmente, que contiene información del usuario, id, username y el tiempo de expiración. Este token es almacenado en localStorage del navegador y se utiliza en cada interacción con la API para autenticar las solicitudes.

El token está conformado por 3 partes fundamentales.

- Header: contiene el cifrado utilizado (HS256)
- Payload: contiene la información del mensaje, en este caso el id, username y tiempo de expiración
- Signature: Es una firma digital generada a partir del Header, el Payload y una clave secreta proporcionada por el servidor. Su función

es garantizar la autenticidad e integridad del token. Cada vez que un usuario inicia sesión, se genera una nueva firma, asegurando que el token no pueda ser modificado sin invalidarlo.

```
const jwtToken = jwt.sign(  
  { id: user.id, username: user.username },  
  SECRET_KEY,  
  { expiresIn: "10m" }  
);
```

Figura 26. Generación JWT. Elaboración propia.

En Node.js, para generar un token de autenticación, se utiliza la función `jwt.sign` del módulo `jsonwebtoken`. Esta función crea automáticamente el header, por lo que solo es necesario agregar el payload y una clave secreta para firmarlo.

La clave secreta es un valor único que garantiza que el token no pueda ser alterado por terceros. Para mantenerla segura, se almacena en un archivo `.env`, que permite gestionar variables de entorno sin exponer datos sensibles en el código fuente.

Además, el token tiene un tiempo de expiración, lo que significa que después de 10 minutos, dejará de ser válido y el usuario deberá volver a autenticarse.

Una vez diseñado el inicio de sesión, se procedió a desarrollar la página principal de la interfaz. En esta sección, se crearon cuatro botones, cada uno representando un tipo de sensor disponible. Al seleccionar un sensor, se muestra una imagen acompañada de una breve descripción de este. Además, en la parte inferior de la descripción, se añadió un botón que redirige al usuario a la página correspondiente para visualizar los datos obtenidos del sensor, como se muestra en la Figura 27.



Figura 27. Interfaz página web principal. Elaboración propia.

Después de diseñar la página principal, se procedió a desarrollar una interfaz web específica para cada sensor. Uno de los principales objetivos fue permitir que el usuario tuviera acceso en tiempo real a los datos recopilados. Para ello, se implementó una tabla dinámica utilizando `tableBody` y un gráfico interactivo generado con `Chart.js`. Estos elementos permiten visualizar los datos más recientes de cada sensor de manera dinámica mientras el sistema está en funcionamiento. En caso de que el sistema esté apagado, la interfaz mostrará únicamente los últimos datos almacenados.

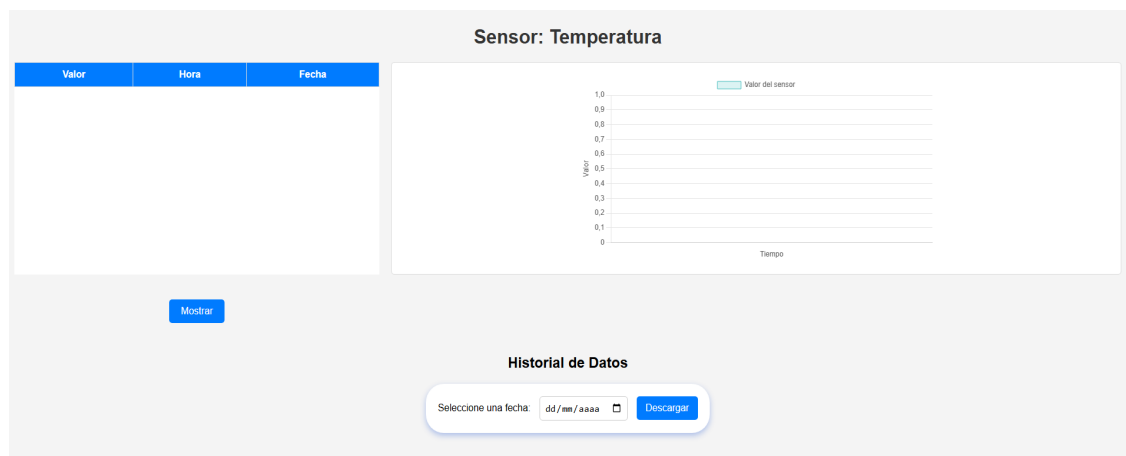


Figura 28. Interfaz sensor temperatura. Elaboración propia.

En la parte inferior de la interfaz, se agregó un apartado para consultar mediciones históricas en función de una fecha específica. El usuario puede seleccionar una fecha determinada y, si existen registros disponibles para ese día, el sistema genera automáticamente un archivo en formato `.csv`, que contiene el valor, la fecha y la hora de cada medición. Esto facilita el acceso y la descarga de información para su posterior análisis.

A continuación, se muestra un diagrama general en la comunicación de las interfaces.

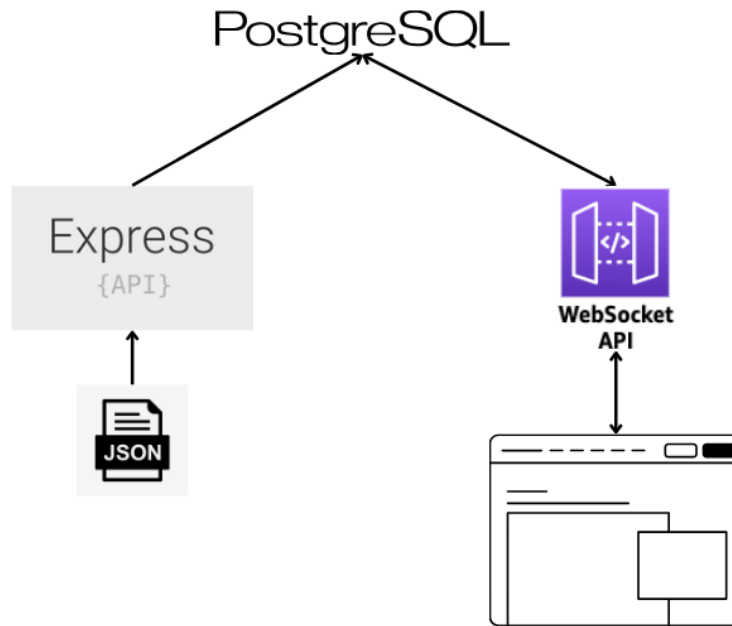


Figura 29. Comunicación de las interfaces. Elaboración propia.

7.3.5 Migración a la nube

Tras el desarrollo de la interfaz web y la implementación de los protocolos de comunicación en cada escenario, se requiere desplegar el sistema en la nube para garantizar su accesibilidad desde cualquier dispositivo y en cualquier momento.

Para lograr esto, se utilizó un servicio de hosting en la nube, permitiendo el despliegue de los servicios de manera independiente y asegurando su correcto funcionamiento. En este caso, se optó por Render, una plataforma que facilita la implementación de aplicaciones y bases de datos en la nube.

El proceso de migración inició con la subida de la base de datos en PostgreSQL. Para ello, fue necesario exportar la base de datos en un archivo SQL, el cual luego se importó en la instancia de PostgreSQL creada en

Render. Posteriormente, al configurar la instancia, la plataforma generó una cadena de conexión (Connection String), que es utilizada por la API para establecer una comunicación segura con la base de datos en la nube.

Una vez desarrolladas las APIs, el siguiente paso fue subirlas a la nube para que pudieran estar disponibles desde cualquier dispositivo. Primero, se subió la API con WebSockets, encargada de la comunicación en tiempo real entre la página web y la base de datos. Para que esto funcionara correctamente, fue necesario configurar el archivo .env, donde se guarda la URL de la base de datos para establecer la conexión y la SECRET_KEY, que se usa para la firma de los tokens JWT.

```
proyecto > Back_end > .env
1  JWT_SECRET=oiJfdaSobdv1u3012948238y5rhtgubfjsdkn9oisadj109238u4342yhrghbf1239
2  DATABASE_URL=postgresql://juansebastianrodriguez:Phwd2G0fzRR30eNwBvq2laTJ9NKtMC3f@dpg-cumgtbbqf0us73cjndpg-a.oregon-postgres.render.com/db2780
```

Figura 30. URL base de datos y JWT_SECRET. Elaboración propia.

Después de ajustar la configuración, se actualizó el código para que pudiera comunicarse con la base de datos a través de la URL en lugar de una conexión local. Con esto listo, se procedió a migrar el servicio web. Para hacerlo, se creó un repositorio en GitHub, lo que permitió enlazar el código con Visual Studio y el servicio de hosting. Gracias a esta integración, fue posible subir el código a la plataforma y hacer que Render lo obtuviera directamente desde el repositorio.

Una vez subido el código, hubo que configurar las variables del sistema en Render, asegurando que la plataforma pudiera acceder a la URL de la base de datos y la SECRET_KEY. Tras completar esta configuración, el servicio quedó completamente desplegado y listo para funcionar en la nube.

Para la segunda API, se desarrolló un nuevo servicio web encargado de gestionar la conexión entre el UAV y la base de datos. El proceso de implementación fue similar al de la API anterior, con la diferencia de que, en este caso, el archivo .env solo almacena la URL de la base de datos, ya que no es necesario incluir la SECRET_KEY para la firma de tokens.

Finalmente, la página web fue desplegada en Render, utilizando su servicio de alojamiento estático. Para ello, fue necesario modificar el código de todas las páginas que establecían comunicación con la API a través de WebSocket,

ajustando la conexión de una configuración local a una conexión en la nube. Esto implicó reemplazar las referencias locales de la API por la URL proporcionada por Render, permitiendo que la página web se comuniqué correctamente con la API alojada en la nube.

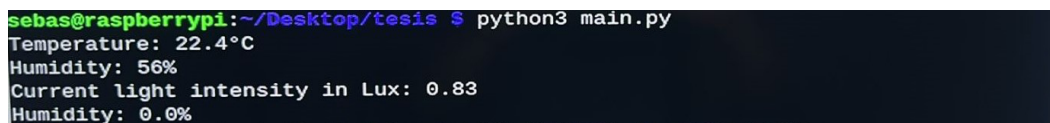
8. RESULTADOS

En este capítulo se presentan los resultados obtenidos tras la implementación del sistema en su totalidad. Se verifica el correcto funcionamiento de cada uno de los componentes: el sistema embebido, el UAV y la página web.

8.1 Datos sensores

Para verificar el correcto funcionamiento de los sensores, se implementó un proceso que muestra en la consola los datos generados en tiempo real, almacenándolos en un archivo JSON.

En la Figura 31 se pueden ver las lecturas obtenidas de los sensores, incluyendo valores de temperatura, humedad, intensidad de luz y humedad del suelo.



```
sebas@raspberrypi:~/Desktop/tesis $ python3 main.py
Temperature: 22.4°C
Humidity: 56%
Current light intensity in Lux: 0.83
Humidity: 0.0%
```

Figura 31. Medición sensores sistema embebido.

Luego de generar los datos de los sensores en un archivo JSON se procede a aplicar el protocolo SCP para la transmisión hacia la Raspberry Pi alojada en el UAV.

8.2 Transmisión de datos al UAV

El archivo JSON generado por la Raspberry Pi del sistema embebido se transmite mediante SCP. Para que ambas Raspberry Pi puedan comunicarse, primero es necesario instalar la librería adecuada y configurar su conexión, asegurando que puedan identificarse mutuamente a través de una clave de autenticación. Luego, se obtiene la dirección IP generada por el punto de acceso de la Raspberry Pi del UAV y se incorpora en el código de la Raspberry Pi del sistema embebido. Esto permite que la transferencia de datos se realice de forma automática utilizando comandos SCP. Por consiguiente, el archivo JSON se almacena temporalmente en la Raspberry Pi del UAV hasta que llegue otro archivo y lo sobrescribe.

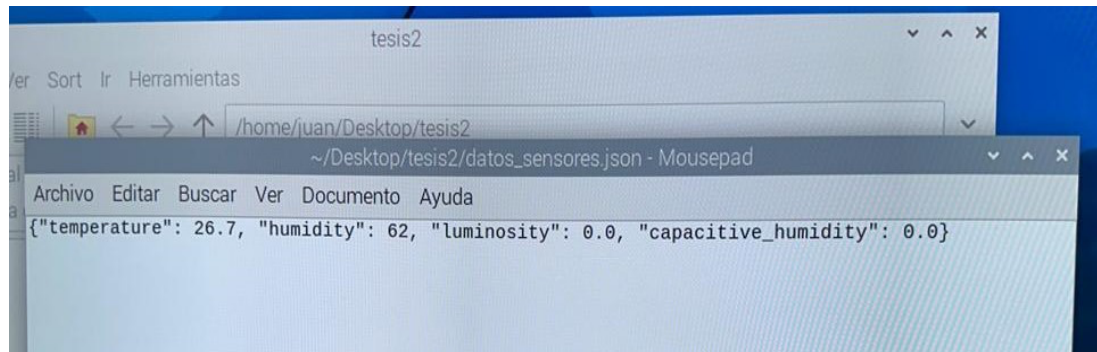


Figura 32. JSON temporal en la Raspberry Pi UAV. Elaboración propia.

8.3 Almacenamiento de datos en la nube

Después de almacenar temporalmente los datos en la Raspberry Pi del UAV, es necesario enviarlos a la base de datos alojada en la nube. Para ello, el sistema transmite la información en formato JSON a través de una solicitud HTTPS a la API desarrollada con Express. Esta API verifica que los nombres de los sensores coincidan con los registros establecidos en la base de datos. Una vez validada la información, se procede a insertar los datos mediante una segunda petición HTTPS.

A continuación, se presenta Figura 33 que evidencia la correcta inserción de los datos en la base de datos.

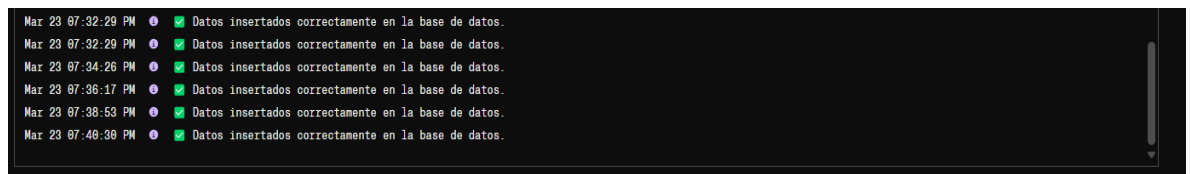


Figura 33. Registro de eventos API Express en render. Elaboración propia.

8.4 Transmisión de datos hacia la interfaz web

Una vez almacenados los datos en la base de datos, es necesario mostrarlos en la interfaz web en tiempo real. Para ello, el usuario primero debe iniciar sesión y seleccionar el sensor que desea analizar. Luego, al presionar el botón

de "Mostrar", se envía una solicitud a la API implementada con WebSockets para obtener la información del sensor seleccionado.

Cuando la API recibe la petición, ejecuta una consulta SQL para recuperar los últimos 10 datos almacenados en la base de datos. Estos datos son enviados de inmediato a la interfaz web y se actualizan dinámicamente en una tabla y un gráfico, proporcionando al usuario una visualización clara y en tiempo real de la información recolectada. Además, mientras el sistema permanece en funcionamiento, la API continúa enviando actualizaciones periódicas con los nuevos valores registrados, garantizando una visualización continua de los datos más recientes.

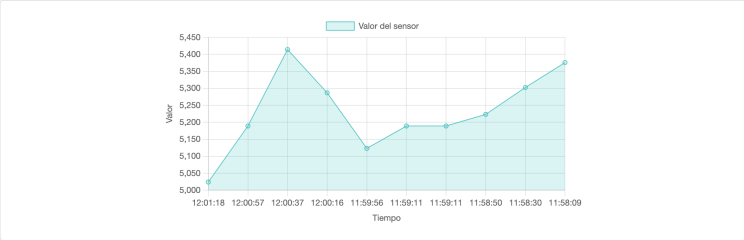
A continuación, se muestra cada página correspondiente a cada sensor obteniendo los datos provenientes desde el UAV.



Sensor: Luminosidad

Valor	Hora	Fecha
5024.17	12:01:18	2025-04-17T00:00:00.000Z
5189.17	12:00:57	2025-04-17T00:00:00.000Z
5414.17	12:00:37	2025-04-17T00:00:00.000Z
5286.67	12:00:16	2025-04-17T00:00:00.000Z
5123.33	11:59:56	2025-04-17T00:00:00.000Z
5189.17	11:59:11	2025-04-17T00:00:00.000Z
5189.17	11:59:11	2025-04-17T00:00:00.000Z
5223.33	11:58:50	2025-04-17T00:00:00.000Z
5302.5	11:58:30	2025-04-17T00:00:00.000Z
5375.83	11:58:09	2025-04-17T00:00:00.000Z

Mostrar



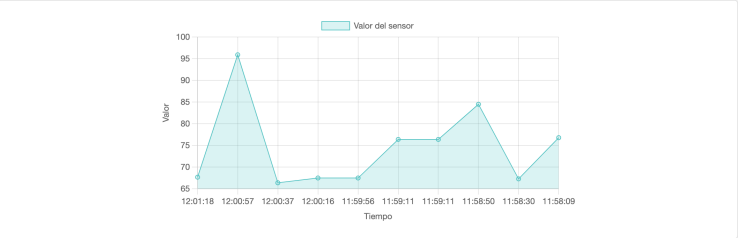
Historial de Datos

Seleccione una fecha:

Sensor: Humedad Relativa

Valor	Hora	Fecha
67.7	12:01:18	2025-04-17T00:00:00.000Z
95.9	12:00:57	2025-04-17T00:00:00.000Z
66.4	12:00:37	2025-04-17T00:00:00.000Z
67.5	12:00:16	2025-04-17T00:00:00.000Z
67.5	11:59:56	2025-04-17T00:00:00.000Z
76.4	11:59:11	2025-04-17T00:00:00.000Z
76.4	11:59:11	2025-04-17T00:00:00.000Z
84.5	11:58:50	2025-04-17T00:00:00.000Z
67.3	11:58:30	2025-04-17T00:00:00.000Z
76.8	11:58:09	2025-04-17T00:00:00.000Z

Mostrar



Historial de Datos

Seleccione una fecha:

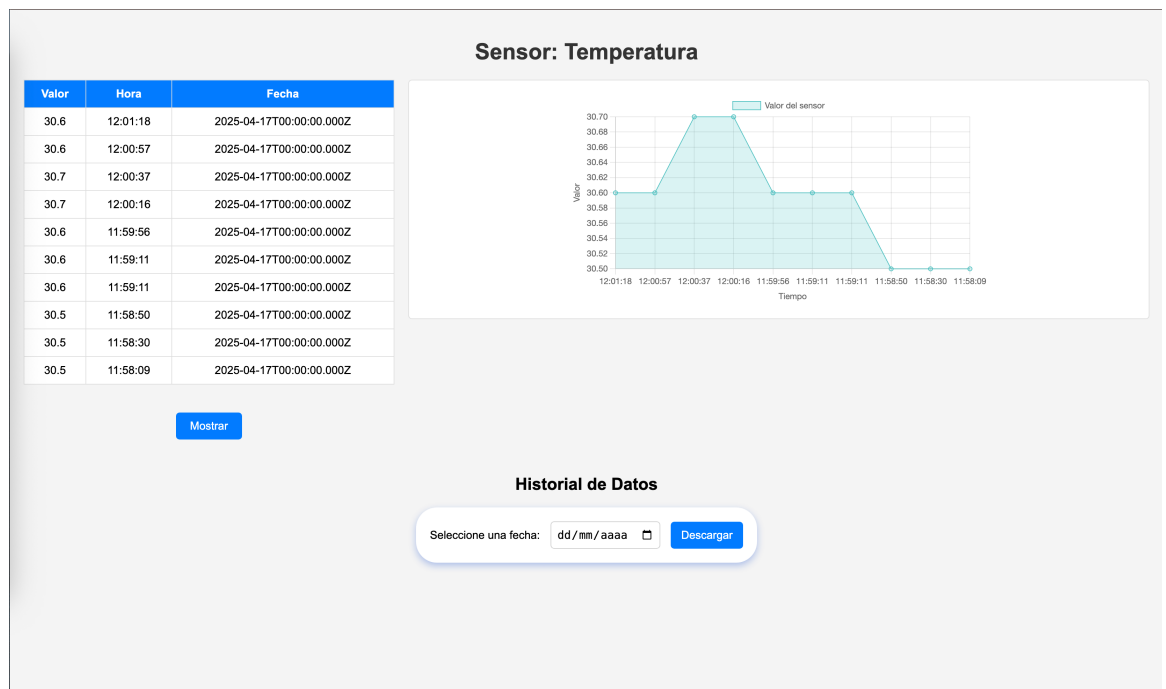


Figura 34. Pagina Sensor: Temperatura. Elaboración propia.

Posteriormente, se verificó el correcto funcionamiento del historial de datos. Para ello, se ingresó una fecha en la que previamente se habían registrado mediciones, y el sistema entrego el siguiente archivo .csv.

datos_sensor (4)		
valor	fecha	hora
27.1	2025-03-23T00:00:00.000Z	19:40:30
27	2025-03-23T00:00:00.000Z	19:38:53
26.9	2025-03-23T00:00:00.000Z	19:36:17
26.8	2025-03-23T00:00:00.000Z	19:34:25
26.7	2025-03-23T00:00:00.000Z	19:32:29
26.6	2025-03-23T00:00:00.000Z	19:32:29
26.7	2025-03-23T00:00:00.000Z	19:01:34
26.6	2025-03-23T00:00:00.000Z	19:01:13
26.7	2025-03-23T00:00:00.000Z	19:01:12
26.6	2025-03-23T00:00:00.000Z	19:00:52
26.6	2025-03-23T00:00:00.000Z	19:00:31
26.6	2025-03-23T00:00:00.000Z	18:59:52
26.6	2025-03-23T00:00:00.000Z	18:59:31
26.6	2025-03-23T00:00:00.000Z	18:59:09
26.6	2025-03-23T00:00:00.000Z	18:59:09
26.6	2025-03-23T00:00:00.000Z	18:58:47
26.5	2025-03-23T00:00:00.000Z	18:57:28
26.4	2025-03-23T00:00:00.000Z	18:53:59
26.4	2025-03-23T00:00:00.000Z	18:53:19
26.4	2025-03-23T00:00:00.000Z	18:53:19
26.3	2025-03-23T00:00:00.000Z	18:50:53
26.3	2025-03-23T00:00:00.000Z	18:50:53
26.1	2025-03-23T00:00:00.000Z	18:41:14
26	2025-03-23T00:00:00.000Z	18:38:14
26	2025-03-23T00:00:00.000Z	18:38:11
25.7	2025-03-23T00:00:00.000Z	18:29:26
25.7	2025-03-23T00:00:00.000Z	18:29:25
25.6	2025-03-23T00:00:00.000Z	18:29:14
25.2	2025-03-23T00:00:00.000Z	18:17:28
25.2	2025-03-23T00:00:00.000Z	18:17:28
25.1	2025-03-23T00:00:00.000Z	18:17:12
25.1	2025-03-23T00:00:00.000Z	18:17:12
25.1	2025-03-23T00:00:00.000Z	18:16:34
25.1	2025-03-23T00:00:00.000Z	18:16:33
25.1	2025-03-23T00:00:00.000Z	18:15:44

Figura 35. Archivo .csv sensor de temperatura. Elaboración propia.

8.5 Validación del sistema en un cultivo

Por último, se ubicó el sistema embebido en un cultivo para validar su funcionamiento junto al dron. A continuación, se presenta Figura 36 en donde se observa el sistema embebido en un cultivo de limones en el Valle del Cauca y Figura 37 del UAV obteniendo los datos y transmitiéndolos a la interfaz web.



Figura 36. Sistema embebido en el cultivo. Elaboración propia.



Figura 37. UAV Transmitiendo los datos a la nube. Elaboración propia.

8.6 Comparación de costos

A continuación, se muestra la tabla de costos de los componentes utilizados en la implementación del sistema. En esta Tabla 1 se detalla la cantidad de cada componente y su precio en pesos colombianos (COP).

Costos				
#	Componente	Cantidad	Precio unitario (COP)	Precio total (COP)
1	Raspberry PI 3B+	2	\$ 205.991,00	\$ 411.982,00
2	Sensor capacitivo de humedad V1.2	1	\$ 10.000,00	\$ 10.000,00
3	Sensor de temperatura y humedad AHT10	1	\$ 15.000,00	\$ 15.000,00
4	Sensor de luminosidad BH1750	1	\$ 7.000,00	\$ 7.000,00
5	PowerBank 20000mah Redmi 18w	1	\$ 109.200,00	\$ 109.200,00
6	Pixhawk 2.4.8	1	\$ 619.047,72	\$ 619.047,72
7	Marco dron F450	1	\$ 90.345,72	\$ 90.345,72
8	Batería LiPo 5200mAh 11,1V	1	\$ 145.000,00	\$ 145.000,00
9	Cargador batería LiPo LiPro Balance Charger IMAX B6	1	\$ 144.371,55	\$ 144.371,55
10	Módulo de potencia V1.0	1	\$ 56.502,00	\$ 56.502,00
11	4 motores sin escobillas, 4 ESC sin escobillas 30A y 4 hélices	1	\$ 231.085,28	\$ 231.085,28
12	Flysky FS-i6X Transmisor y receptor RC FS-iA6B iBus	1	\$ 222.830,76	\$ 222.830,76
13	Módem Zte Mf190 - 3.5G	1	\$ 54.900,00	\$ 54.900,00
14	Filamento Anycubic para impresora 3D	1	\$ 69.598,00	\$ 69.598,00

15	Mensualidad base de datos en la nube	1	\$ 26.972,14	\$ 26.972,14
TOTAL				\$ 2.213.835,17

Tabla 1. Costos de los componentes. Elaboración propia

A continuación, se presenta la Tabla 2 que compara diferentes tecnologías para la medición de variables en cultivos, junto con sus costos totales de implementación, considerando su viabilidad para agricultores de mediana y baja escala en el Valle del Cauca.

Dispositivo	Costo (COP)	Especificaciones
RED DE DATOS INALÁMBRICA BASADA EN UAV PARA APLICACIONES BÁSICAS DE E-AGRICULTURA	\$ 2.213.835,17	Sistema escalable y de bajo costo inicial, ya que la mayor inversión está en el UAV. Permite ampliar la cobertura agregando más dispositivos de medición sin necesidad de duplicar la inversión en UAV. Integración con redes celulares y WiFi para transmisión de datos en tiempo real. Compatible con sensores de temperatura, humedad, luminosidad y humedad del suelo. Almacenamiento y consulta de datos en la nube mediante API con WebSockets.
	\$ 2.489.487,03	Sistema fijo con sensores de temperatura, humedad, velocidad

Ambient Weather Estación meteorológica inteligente ultrasónica WS-5000		y dirección del viento. No permite una fácil expansión de cobertura sin comprar nuevas estaciones. Conectividad mediante WiFi y almacenamiento en la nube, pero sin integración con UAV. No adaptable para cambios en el terreno o monitoreo de distintas áreas sin adquirir otro dispositivo.
Libelium Smart Agriculture Xtreme Basic IoT Vertical Kit	\$ 41.243.000,00	Sistema fijo con sensores de temperatura, humedad, velocidad y dirección del viento. No permite una fácil expansión de cobertura sin comprar nuevas estaciones. Conectividad mediante WiFi y almacenamiento en la nube, pero sin integración con UAV. No adaptable para cambios en el terreno o monitoreo de distintas áreas sin adquirir otro dispositivo.

Tabla 2. Comparación tecnologías. Elaboración propia

9. ANALISIS

9.1 Datos sensores:

El proceso de adquisición de datos en tiempo real permitió comprobar que los sensores instalados en el sistema embebido funcionan de manera adecuada. Las lecturas de temperatura, humedad, luminosidad y humedad del suelo se mostraron de forma continua y sin interrupciones en la consola, lo que confirma que los sensores registran información de manera estable.

Al guardar los datos en un archivo JSON, se logró organizar la información de forma sencilla y eficiente, facilitando su manejo en las siguientes etapas de transmisión y procesamiento. Este formato también ayudó a mantener un flujo de datos ligero, ideal para las capacidades limitadas del sistema embebido.

Además, los valores obtenidos fueron coherentes con las condiciones del entorno de prueba, lo que indica que los sensores estaban correctamente calibrados. Tener mediciones fiables desde el principio fue clave para asegurar que toda la cadena de transmisión de datos posterior pudiera funcionar correctamente.

9.2 Transmisión de datos al UAV

La transmisión de datos desde la Raspberry Pi del sistema embebido hacia el UAV se realizó de forma sencilla y confiable. Configurar la conexión mediante SCP el cual facilitó que ambos dispositivos se reconocieran automáticamente, sin necesidad de intervención manual cada vez que se enviaba un archivo.

El hecho de incorporar la dirección IP del UAV directamente en el código permitió que la transferencia de los archivos JSON se hiciera de manera automática, agilizando el proceso y evitando posibles errores de conexión. Esta estrategia simplificó la comunicación entre dispositivos y ayudó a que el sistema fuera más robusto durante su funcionamiento en campo.

Finalmente, almacenar temporalmente el archivo en la Raspberry Pi del UAV sobrescribiendo el anterior permitió manejar los datos de forma práctica, sin saturar la memoria del dispositivo y asegurando que siempre se tuviera disponible la información más reciente.

9.3 Migración a la nube

El proceso de envío de los datos desde la Raspberry Pi del UAV hacia la base de datos en la nube se realizó de manera exitosa. La transmisión mediante solicitudes HTTPS a través de la API desarrollada con Express permitió garantizar que la información fuera validada antes de ser almacenada, asegurando así que solo se registraran datos correctos y organizados.

La validación previa de los nombres de los sensores resultó muy útil para evitar errores de consistencia en la base de datos, y el sistema respondió de forma estable durante las pruebas realizadas. La evidencia mostrada en el registro de eventos confirma que los datos fueron insertados correctamente y sin fallos en las solicitudes.

Implementar este tipo de comunicación también facilitó que la transmisión de los datos fuera rápida y segura, aprovechando un protocolo ampliamente utilizado para garantizar la integridad de la información en entornos de red.

9.4 Transmisión de datos a la interfaz web

La transmisión de los datos almacenados en la base de datos hacia la interfaz web funcionó de manera eficiente y estable. El uso de WebSockets permitió mantener una comunicación continua entre el servidor y el navegador, lo que facilitó la actualización dinámica de la información cada vez que se registraban nuevos datos.

La posibilidad de mostrar en tiempo real los últimos 10 registros, tanto en forma de tabla como de gráfico, mejoró significativamente la experiencia del usuario, ofreciendo una visualización clara y actualizada de las mediciones. Además, el envío periódico de nuevas actualizaciones permitió que la información en la interfaz se mantuviera siempre al día sin necesidad de refrescar la página o realizar nuevas solicitudes manuales.

Esta implementación no solo hizo que la interacción con el sistema fuera más fluida, sino que también demostró ser una solución práctica y escalable para manejar la visualización de datos en tiempo real en proyectos de monitoreo agrícola.

9.5 Validación en un cultivo

La prueba realizada en el cultivo permitió comprobar que el sistema embebido y el UAV trabajaron de forma conjunta y estable. Durante varios ensayos de aproximadamente 30 minutos, se logró capturar y transmitir los datos sin interrupciones, mostrando que la comunicación entre ambos dispositivos fue confiable en un entorno real.

Ubicar el sistema en el campo no presentó mayores dificultades y permitió verificar que podía adaptarse a las condiciones naturales, donde aspectos como la distancia o el ambiente podían haber afectado su desempeño. A pesar de estos factores, el sistema respondió bien, cumpliendo con el objetivo de recolectar la información y enviarla a la nube para su visualización en tiempo real.

Esta validación en condiciones reales de uso fue fundamental para confirmar que el diseño planteado no solo funciona en pruebas controladas, sino también en escenarios reales, donde su aplicación práctica tiene verdadero valor.

9.6 Comparación de costos

La comparación de costos permitió demostrar que la solución desarrollada ofrece una alternativa accesible frente a otras tecnologías comerciales de monitoreo agrícola. Los componentes del sistema fueron seleccionados buscando un equilibrio entre precio, funcionalidad y eficiencia, con el objetivo de construir una plataforma que no requiriera una inversión elevada, pero que mantuviera su efectividad en la recolección y transmisión de datos en el cultivo.

Uno de los principales beneficios observados es la escalabilidad que ofrece el sistema. A diferencia de las soluciones convencionales, que obligan a adquirir un nuevo dispositivo completo para cada área adicional que se desee monitorear, el enfoque basado en UAV permite agregar nuevos dispositivos de medición de forma sencilla y con un costo mucho menor. Gracias a que el UAV actúa como un canal de comunicación para todos ellos, el sistema puede expandirse sin necesidad de hacer grandes inversiones en infraestructura.

Esta flexibilidad representa una ventaja importante, especialmente para agricultores de mediana y pequeña escala, quienes podrían beneficiarse de una

herramienta eficiente, económica y adaptable a diferentes necesidades de monitoreo en campo.

10. CONCLUSIONES

A continuación, se presenta las conclusiones del presente trabajo:

- Se logró diseñar un sistema de bajo costo basado en tecnologías de Internet de las Cosas (IoT) y vehículos aéreos no tripulados (UAV), orientado al monitoreo de cultivos de baja producción. El sistema desarrollado permitió la medición, transmisión y visualización en tiempo real de variables agrícolas relevantes, garantizando un funcionamiento estable y accesible para pequeños y medianos agricultores.
- Se seleccionaron sensores de temperatura, humedad relativa, luminosidad y humedad del suelo, considerando criterios de bajo costo, disponibilidad y relevancia agronómica. Estas variables fueron escogidas por su impacto directo en el manejo de cultivos y por la posibilidad de obtener lecturas confiables mediante sensores accesibles para proyectos de agricultura de precisión a pequeña escala.
- Se implementó un sistema embebido basado en Raspberry Pi, que permitió la captura de datos de los sensores y su transmisión hacia el UAV mediante comunicación automática utilizando protocolos SCP. El UAV, equipado también con una Raspberry Pi y un módem USB, facilitó la conexión simultánea a redes WiFi y celulares, posibilitando la transmisión de datos a la nube en tiempo real. Esta configuración garantizó una comunicación estable, segura y eficiente, optimizando el flujo de información desde el cultivo hasta la infraestructura de almacenamiento remoto.
- Se construyó una aplicación web utilizando una API basada en Websockets, que permitió visualizar los datos recolectados en tiempo real y consultar registros históricos de manera eficiente. El sistema de almacenamiento en PostgreSQL aseguró la organización adecuada de la información, minimizando la latencia y mejorando la respuesta del sistema, lo que resultó en una experiencia de usuario ágil y funcional para la consulta de datos desde cualquier lugar y en cualquier momento.
- La validación en campo demostró que el sistema es funcional y escalable. La transmisión de datos desde el sistema embebido al UAV, y de allí a la nube y la interfaz web, se realizó sin interrupciones, incluso en condiciones reales

de uso. Se comprobó que el UAV amplía significativamente el rango de cobertura del sistema, permitiendo conectar múltiples dispositivos de medición ubicados en diferentes zonas del cultivo. Esto facilita la expansión del sistema sin necesidad de inversiones adicionales en infraestructura, haciendo que la solución propuesta sea flexible, accesible y adecuada para entornos agrícolas de pequeña y mediana escala.

11. TRABAJOS FUTUROS

En este capítulo se plantean posibles mejoras y extensiones del proyecto que podrían implementarse en el futuro. Estas ideas buscan optimizar el desempeño del sistema y explorar nuevas funcionalidades que amplíen su alcance.

- Implementar un sistema de alimentación sostenible, integrando energía solar para mejorar la autonomía del sistema embebido y reducir la dependencia de fuentes externas de energía.
- Integrar tecnologías de conectividad más avanzadas, como redes 4G o 5G, en la Raspberry Pi del UAV para mejorar la comunicación y transmisión de datos.
- Incorporar un módulo GPS en el UAV que permita habilitar modos de vuelo asistidos, facilitando el control y navegación del dron para el usuario.
- Reforzar la seguridad de la página web, almacenando el JSON Web Token (JWT) en una cookie segura e implementando mecanismos adicionales de verificación y protección de datos.

12. REFERENCIAS

- [1] “Agricultura y alimentos.” Accessed: Jun. 07, 2024. [Online]. Available: <https://www.bancomundial.org/es/topic/agriculture/overview>
- [2] “Global agriculture towards 2050 – High Level Expert Forum: How to Feed the World in 2050, FAO.” Accessed: Jun. 07, 2024. [Online]. Available: https://www.fao.org/fileadmin/templates/wsfs/docs/Issues_papers/HLEF2050_Global_Agriculture.pdf
- [3] “Solo se está aprovechando 13,5% de las 39,2 millones de hectáreas con potencial.” Accessed: Jun. 07, 2024. [Online]. Available: <https://www.larepublica.co/economia/del-34-del-area-potencial-para-cultivar-en-colombia-se-aprovecha-cerca-del-13-5-3391297>
- [4] “Los grandes retos de la agricultura - Tradecorp.” Accessed: Jun. 07, 2024. [Online]. Available: <https://tradecorp.es/retos-agricultura/>
- [5] Ó. A. Orozco and G. Llano Ramírez, “Sistemas de Información enfocados en tecnologías de agricultura de precisión y aplicables a la caña de azúcar, una revisión,” *Revista Ingenierías Universidad de Medellín*, vol. 15, no. 28, pp. 103–124, 2016, doi: 10.22395/rium.v15n28a6.
- [6] “Hay más de 15,2 millones de campesinos, que representan 30% de los colombianos | Agronegocios.co.” Accessed: Jun. 07, 2024. [Online]. Available: <https://www.agronegocios.co/agricultura/hay-mas-de-15-2-millones-de-campesinos-que-representan-un-30-de-los-colombianos-3628216>
- [7] L. Johanna, R. Gonzalez, P. Ingeniería, A. Santiago, and D. Cali, “AGRICULTURA DE PRECISIÓN EN EL MUNDO Y EN COLOMBIA: REVISIÓN BIBLIOGRÁFICA,” 2020.
- [8] V. Cortes and F. V. Garcia, “DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE RIEGO AUTOMATIZADO Y MONITOREO DE VARIABLES AMBIENTALES MEDIANTE IOT EN LOS CULTIVOS URBANOS DE LA FUNDACIÓN MUJERES EMPRESARIAS MARIE POUSSEPIN,” Bogota, 2020.
- [9] C. Aguirre, “DESARROLLO DE DISPOSITIVO IOT PARA MONITOREO AGRÍCOLA DE BAJO COSTE,” Aguascalientes, 2020.
- [10] P. (Paul M.) Cunningham and M. Cunningham, *IoT Sensing Platform for e-Agriculture in Africa*. 2020.

- [11] P. Deotale, "E-agricultura inteligente basada en IoT monitoreo con aplicación Android," in *Conferencia internacional 2022 sobre tendencias emergentes en ingeniería y ciencias medicas*, India, 2022, pp. 1–5.
- [12] H. Kaur, A. KR Shukla, and H. Singh, "Revisión de las tecnologías IoT utilizadas en la agricultura," in *2da Conferencia Internacional sobre Computación Avanzada y Tecnologías Innovadoras en ingeniería*, Mohali, Punjab, India, 2022, pp. 1–5.
- [13] "¿Qué es IoT? - Explicación del Internet de las cosas - AWS." Accessed: Jun. 07, 2024. [Online]. Available: <https://aws.amazon.com/es/what-is/iot/>
- [14] Y. TORAMAN and T. ÖZ, "The Use of New Technologies in Logistics: Drone (UAV) Use in Last Mile Delivery," *Sosyoekonomi*, vol. 31, no. 58, pp. 105–124, Oct. 2023, doi: 10.17233/sosyoekonomi.2023.04.05.
- [15] Narmilan A, "E-Agricultural Concepts for Improving Productivity: A Review," Sri Lanka, 2017.
- [16] L. Calvo, "¿Qué es una Raspberry PI y para qué sirve?" Accessed: Jun. 07, 2024. [Online]. Available: <https://www.godaddy.com/resources/es/tecnologia/que-es-raspberry-pi>
- [17] "The Seven Layers of the OSI Model," *Wireless Design & Development*, May 2006.
- [18] "Protocolos y tecnologías de IoT," Microsoft. Accessed: Nov. 01, 2024. [Online]. Available: <https://azure.microsoft.com/es-es/solutions/iot/iot-technology-protocols>
- [19] Aarón Wang, "LoRa contra Zigbee: ¿Cuál es la mejor tecnología para la conectividad IoT?," Moko Lora. Accessed: Nov. 01, 2024. [Online]. Available: <https://www.mokolora.com/es/lora-vs-zigbee-which-is-better/>
- [20] "Capítulo 2: Estándar IEEE 802.15.1 'Bluetooth'." Accessed: May 01, 2025. [Online]. Available: <http://www.ptolomeo.unam.mx/jspui/bitstream/>
- [21] Roel van der Meer, "Qué es una red celular." Accessed: Nov. 02, 2024. [Online]. Available: <https://es.simbase.com/iot-glossary-dictionary/cellular-network>
- [22] "¿Cuál es la diferencia entre HTTP y HTTPS?" Accessed: Nov. 02, 2024. [Online]. Available: <https://aws.amazon.com/es/compare/the-difference-between-https-and-http/>

- [23] Equipo editorial de IONOS, “¿Qué es el SCP protocol?,” IONOS. Accessed: Mar. 11, 2025. [Online]. Available: <https://www.ionos.com/es-us/digitalguide/servidores/known-how/scp-secure-copy/>
- [24] “WebSocket,” Javascript.info. Accessed: Mar. 11, 2025. [Online]. Available: <https://es.javascript.info/websocket>
- [25] “Interfaz de usuario (UI): Qué es, tipos, beneficios y más,” GoDaddy. Accessed: Nov. 04, 2024. [Online]. Available: <https://www.godaddy.com/resources/latam/desarrollo/que-es-interfaz-usuario-ui>
- [26] J. J. Molina Mosquera, J. P. Caviedes Silva, and J. M. Patuso Borda, “Diseño web con HTML5 para un sistema de monitoreo y control remoto por IoT,” *Investigación e Innovación en Ingenierías*, vol. 11, no. 1, pp. 77–89, May 2023, doi: 10.17081/invinno.11.1.6587.
- [27] O. Pérez Mora Carme Martín Escofet Marc Gibert Ginestà and L. Alberto Casillas Santillán, “Bases de datos,” May 2005. [Online]. Available: www.glo.org.mx
- [28] “Sensor de Temperatura y Humedad AHT10,” MCI Electronics. Accessed: Mar. 04, 2025. [Online]. Available: <https://mcielectronics.cl/shop/product/sensor-de-temperatura-y-humedad-aht10/>
- [29] “Tutorial módulo sensor de luz BH1750,” Naylamp Mechatronics. Accessed: Mar. 04, 2025. [Online]. Available: https://naylampmechatronics.com/blog/44_tutorial-modulo-sensor-de-luz-bh1750.html
- [30] “Hawks Work Marco De Drone F450, 17.717 In Distancia Entre E.,” Mercado Libre. Accessed: Mar. 06, 2025. [Online]. Available: https://articulo.mercadolibre.com.co/MCO-2810572626-hawks-work-marco-de-drone-f450-17717-in-distancia-entre-e-_JM#redirectedFromSimilar=https%3A%2F%2Farticulo.mercadolibre.com.co%2FMCO-2742682356-hawks-work-marco-de-drone-f450-17717-in-distancia-entre-e-_JM
- [31] Nuofany, “APM 2.5 2.6 2.8 para módulo de potencia Pixhawk 30V 90A con 5.3V DC BEC con enchufe T XT60 para RC Drone Helicopter Part Quadcopters Accesorios (XT60),” Amazon. Accessed: Mar. 06, 2025. [Online]. Available: https://www.amazon.com/-/es/potencia-BEC-Helicopter-Quadcopters-Accesorios/dp/B0BCJM3R5P/ref=pd_vtp_strm_d_sccl_2_1/143-4053545-

8840527?pd_rd_w=Ev6BT&content-id=amzn1.sym.8c8432aa-1bf0-4437-b5fa-12091a348861&pf_rd_p=8c8432aa-1bf0-4437-b5fa-12091a348861&pf_rd_r=M5CJXFYKHQ365S9NQCW5&pd_rd_wg=sLu6F&pd_rd_r=c1f3c5db-1b4b-4c84-beca-4a560ce0eafe&pd_rd_i=B0BCJM3R5P&th=1

- [32] XIWIYI, "Flysky FS-i6X Transmisor y receptor RC FS-iA6B iBus 10 canales o PWM 6CH 2.4GHz AFHDS 2A Sistema de control de radio para FPV Drone Multirotor, ala fija, coche RC, barco RC (Modo-2)," Amazon. Accessed: Mar. 06, 2025. [Online]. Available: https://www.amazon.com/-/es/Flysky-FS-i6X-Transmisor-receptor-FS-iA6B-Multirotor/dp/B0BRXM1K4X/ref=sr_1_2?crid=2NZLQ166IV7HU&dib=eyJ2ljojMSJ9.NiKYbEtFxBkh-lhdOLQ5iHcV2uNaigUGoaZWxhzhRC_yb_BFHvQ1lfHeW83V2UoO-j4_YmKRb7Y19M60hXKtocJgj3Lnp_SyAOAnnT_8Gu2ijHnNqUogxeVY4VAfbCQ5VJ5Om86RzLdo6qeGtyuxDA6v8a6ka_aunoXQRw0hqfFhKRB0ldUdg3pMfn6GSxe4zSnhlASx0voc68ZS0gMCvxbxDCcxmQiiHd8TOj2xtVtQtaXJd-n86clxnIRU642NJREDfK71gOeLBkHZ49g4ilu92fOTtBtscSEcaB33S-gQjUoFCPA046LhlOnUIPqgYBM0jYL6lmXpFBqjPZGIl7PR68AeWVWQcgB60HJoYlpGbc04COVyBTTkfY46eV2A3zeFMvmXoibl4Vd6ytK7S9dGxSEO7_TEWMqx_E_4G9k7T1ZjMUAcXduVUPOzZ63OH.iwG07Uwq_V_utdXKhaNZvnwT61X9he3IR_pC6i2N2Kc&dib_tag=se&keywords=fly+sky+rc+transmitter+and+receiver&qid=1741403254&srefix=FLY+SKY+%2Caps%2C217&sr=8-2
- [33] CITRONIA, "Convertidor De Voltaje Dc-dc Lm2596 Reductor Step Down." Accessed: Apr. 20, 2025. [Online]. Available: https://articulo.mercadolibre.com.co/MCO-2330652452-convertidor-de-voltaje-dc-dc-lm2596-reductor-step-down-_JM?matt_tool=26922515&matt_word=&matt_source=google&matt_campaign_id=22126928630&matt_ad_group_id=171342955177&matt_match_type=&matt_network=g&matt_device=c&matt_creative=729836846578&matt_keyword=&matt_ad_position=&matt_ad_type=pla&matt_merchant_id=328819879&matt_product_id=MCO2330652452&matt_product_partition_id=2394343693581&matt_target_id=aud-1918885411362:pla-2394343693581&cq_src=google_ads&cq_cmp=22126928630&cq_net=g&cq_plt=gp&cq_med=pla&gad_source=1&gbraid=0AAAAAD1DcoxmhQa0gOb4T71BCv9IYQAVe&gclid=Cj0KCQjw2ZfABhDBARIsAHFTxGy8ho2Vr3Bki-TeulrSUbOxC_biiVrx9p8t4iP573ghtGXD1SmEYdgaAlnvEALw_wcB
- [34] J. Santos, "¿Qué es una API?: Tipos y 5 ventajas para pymes y startups," Delta proyect. Accessed: Mar. 11, 2025. [Online]. Available: <https://www.deltaprotect.com/blog/que-es-una-api>

- [35] L. Moreno, "Readytosky Pixhawk Px4 Controlador De Vuelo Pixhawk 2.4.8 3.," Mercado Libre. Accessed: Mar. 06, 2025. [Online]. Available: https://articulo.mercadolibre.com.co/MCO-2532386402-readytosky-pixhawk-px4-controlador-de-vuelo-pixhawk-248-3-_JM
- [36] MLMTEY, "Batería lipo de 7,4 v, 11,1 v, 14,8 v, 1500mAh, 2200mah, 2800mah, 4200mah, 5200mah, 3s para coche de juguete, avión, helicóptero, barco a control remoto," AliExpress. Accessed: Mar. 06, 2025. [Online]. Available: https://es.aliexpress.com/item/1005007342393957.html?spm=a2g0o.order_list.order_list_main.9.3573194dNswwx&gatewayAdapt=glo2esp
- [37] "LoRa Technology – An Overview," *ResearchGate*. Accessed: Nov. 13, 2025. [Online]. Available: https://www.researchgate.net/publication/328460957_LoRa_Technology_-_An_Overview
- [38] S. Banks and E. Briggs, "MQTT Version 5.0," *OASIS Standard*, Mar. 2019. [Online]. Available: <https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html>