



UNIVERSIDAD LAICA ELOY ALFARO DE MANABI

FACULTAD CIENCIAS DE LA VIDA Y TECNOLOGIAS

CARRERA DE TECNOLOGIA DE LA INFORMACION

Tema:

**IMPLEMENTACIÓN DE UN SISTEMA DE RIEGO AUTOMATIZADO A TRAVÉS DE
SENSORES A LOS EXTERIORES DEL BLOQUE DE INFORMÁTICAS EN LA
UNIVERSIDAD LAICA ELOY ALFARO DE MANABÍ**

AUTORES:

ANCHUNDIA ANCHUNDIA JAMILETH ESTELA

VERA CASTRO CAROL PAMELA

UNIDAD ACADÉMICA:

MATRIZ MANTA

CARRERA:

TECNOLOGÍAS DE LA INFORMACIÓN

TUTOR:

ING. HENRY NEURIO MERO BRIONES

Certificación del director de trabajo de graduación

CERTIFICACIÓN

En calidad de docente tutor de la Facultad Ciencias de la Vida y Tecnologías de la Universidad Laica "Eloy Alfaro" de Manabí, CERTIFICO:

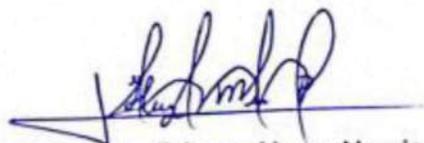
Haber dirigido, revisado y aprobado preliminarmente el Trabajo de titulación bajo la autoría los estudiantes Anchundia Anchundia Jamileth Estela y Vera Castro Carol Pamela legalmente matriculados en la carrera de Tecnologías de la Información, período académico 2024-2025, cumpliendo el total de 400 horas, cuyo tema del proyecto es **"Implementación de un sistema de riego automatizado a través de sensores a los exteriores del bloque de informática en la Universidad Laica Eloy Alfaro de Manabí "**.

La presente investigación ha sido desarrollada en apego al cumplimiento de los requisitos académicos exigidos por el Reglamento de Régimen Académico y en concordancia con los lineamientos internos de la opción de titulación en mención, reuniendo y cumpliendo con los méritos académicos, científicos y formales, y la originalidad del mismo, requisitos suficientes para ser sometida a la evaluación del tribunal de titulación que designe la autoridad competente.

Particular que certifico para los fines consiguientes, salvo disposición de Ley en contrario.

Manta, 10 de enero de 2025.

Lo certifico,



Ing. Mero Briones Henry Neurio.
Docente Tutor(a)

TRABAJO DE TITULACIÓN MODALIDAD: PROYECTO INTEGRADOR

Previo a la obtención del título de:

INGENIERO EN TECNOLOGÍAS DE LA INFORMACIÓN

Título del proyecto:

“Implementación de un sistema de riego automatizado a través de sensores a los exteriores del bloque de informática en la Universidad Laica Eloy Alfaro de Manabí”

El tribunal examinador declara APROBADO el trabajo presentado para optar al grado de: INGENIERO EN TECNOLOGÍAS DE LA INFORMACIÓN, de las señoritas:

Anchundia Anchundia Jamileth Estela

Vera Castro Carol Pamela

Declaración expresa de autoría

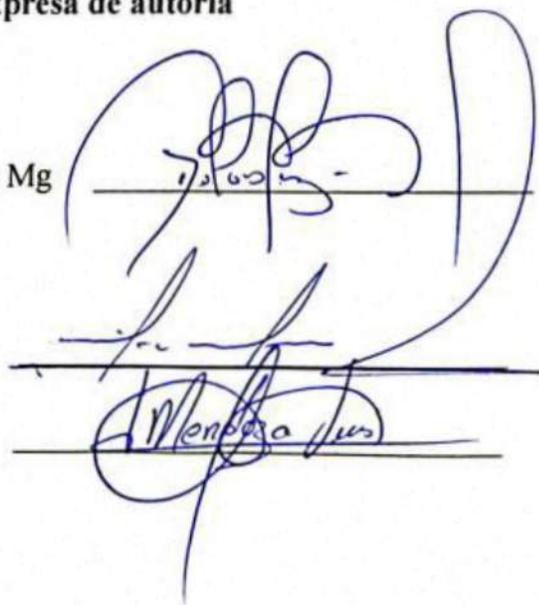
Presidente del tribunal:

Dra. Dolores Esperanza Verduga Muñoz, Mg

Miembro del tribunal:

Ing. Mike Paolo Machuca Avalos, Mg.

Ing. Luis Jacinto Mendoza Cuzme, Mg.



The image shows three handwritten signatures in blue ink, each written over a horizontal line. The top signature is the most complex and loops around the line. The middle signature is more fluid and cursive. The bottom signature is also cursive and appears to be 'Luis Jacinto Mendoza Cuzme'.

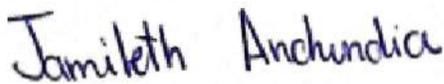
DECLARACIÓN EXPRESA DE AUTORÍA

Nosotras, **Anchundia Anchundia Jamileth Estela**, con cédula de ciudadanía **13461377-7** y **Vera Castro Carol Pamela**, con cédula de ciudadanía **135102398-9**, en calidad de autoras del trabajo de titulación:

“Implementación de un sistema de riego automatizado a través de sensores a los exteriores del bloque de informáticas en la Universidad Laica Eloy Alfaro de Manabí”.

Declaramos que este trabajo es original y de nuestra autoría, hemos citado correctamente las fuentes utilizadas y respetado las leyes vigentes sobre derechos de autor.

Autorizamos a la Universidad Laica "Eloy Alfaro" de Manabí a usar este trabajo con fines académicos o investigativos, así como a publicarlo en su repositorio digital.



Anchundia Anchundia Jamileth Estela

C.I: 131461377-7



Vera Castro Carol Pamela

C.I: 135102398-9

DEDICATORIA

A Dios, por guiarme y darme la fortaleza y sabiduría necesarias para superar cada desafío. Su presencia ha sido una luz constante en mi vida, iluminando mi camino y brindándome paz y esperanza.

A mi madre Rosa Elena Castro Lucas, cuyo amor incondicional, sacrificio y guía han sido los pilares sobre los que he construido mis logros. Su fortaleza y dedicación han inspirado cada paso de este arduo camino. Le agradezco profundamente por inculcarme el valor del esfuerzo y la perseverancia.

A mi hermano Jorge Luis Vera Castro, por ser mi compañero de vida, por su inquebrantable apoyo y por estar siempre dispuesto a ofrecer una palabra de aliento. Su presencia constante ha sido una fuente invaluable de fortaleza y motivación.

A mis hermanas de corazón la Ing. Valeria y Genesis Menéndez quienes han sido más que familiares, verdaderas amigas y confidentes. Agradezco cada momento compartido, su comprensión y su respaldo incondicional. Su apoyo ha sido fundamental en mi vida.

A mis queridos tíos Johnny Menéndez y Alexandra Castro, quienes me criaron con el mismo amor y cuidado que a una hija. Gracias por brindarme un hogar lleno de cariño, por ser mis mentores y por su constante guía y apoyo. Su influencia ha sido esencial en mi desarrollo personal y académico.

A mi amado Omar Adrián Ponce Quiroz, cuya paciencia, amor y apoyo constante han sido esenciales en este viaje. Gracias por estar a mi lado en cada momento, por animarme a seguir adelante y por brindarme la fortaleza necesaria para superar cualquier obstáculo, y también a mi angelito que está en el cielo, al cual voy a recordar siempre como lo mejor que me pudo pasar.

A todos ustedes, les dedico este trabajo con el más profundo agradecimiento y estima. Su presencia y apoyo han sido fundamentales para la culminación de esta etapa académica. Sin su constante aliento y amor, este logro no habría sido posible.

Carol Pamela Vera Castro

DEDICATORIA

Con profundo respeto y agradecimiento dedico este proyecto en primer lugar a Dios, cuya guía divina y protección han sido mi luz en los momentos más oscuros y mi fortaleza en los momentos de debilidad.

A mi querido papá Lider Inocencio Anchundia Delgado, en reconocimiento a tu inquebrantable compromiso y sacrificio. Tus enseñanzas sobre la importancia del trabajo duro y la perseverancia. Gracias por ser un modelo ejemplar de integridad y dedicación. Tu apoyo constante y tus palabras de aliento han sido fundamentales para mi éxito.

A mi amada mamá Deysi Beatriz Anchundia Anchundia, cuya paciencia y amor infinito han sido mi refugio en los momentos de adversidad. Gracias por tu constante apoyo y consejos en mis capacidades, por tus innumerables sacrificios y por ser mi mayor apoyo emocional.

A mi hermano Bryant Jimpson Anchundia Anchundia, mi amigo leal, dedico este trabajo en reconocimiento a tu constante apoyo y ánimo.

A mis queridas mascotas Thobby, Lemmy, Ammy, Shoppy, a mis familiares y amigos ya que ellos han sido un pilar fundamental en algunas u otra forma.

Jamileth Estela Anchundia Anchundia

AGRADECIMIENTO

En primer lugar, queremos expresar nuestro más profundo agradecimiento a Dios, quien nos ha dado la fortaleza, sabiduría y salud para llegar a esta etapa tan significativa en nuestras vidas académicas. Su guía nos ha permitido superar cada desafío con confianza y determinación.

A nuestras familias, quienes, con su apoyo incondicional, amor y sacrificio, han sido nuestra mayor inspiración y motivación para continuar. Agradecemos especialmente a nuestros padres y hermanos, que, con su ejemplo y constante aliento, han estado presentes en cada paso de este recorrido.

Nuestro especial reconocimiento al Ing. Mike Paolo Machuca Avalos, cuya orientación, paciencia y valiosos consejos fueron fundamentales en el desarrollo de nuestra tesis. Su apoyo no solo nos guió en los aspectos técnicos, sino que también nos ayudó a crecer como profesionales.

Al Ing. Henry Neurio Mero Briones, agradecemos su rol como tutor de tesis, reconociendo su esfuerzo y dedicación en este proyecto.

A nuestra prestigiosa Universidad Laica Eloy Alfaro de Manabí, por ser el espacio donde hemos adquirido las herramientas y conocimientos necesarios para nuestra formación profesional. Las experiencias vividas en sus aulas y laboratorios han marcado un antes y un después en nuestro desarrollo.

También extendemos nuestra gratitud a nuestros amigos y compañeros de estudio, quienes con su compañía, apoyo y consejos hicieron este camino más llevadero y enriquecedor. Sus palabras de ánimo y momentos compartidos nos han dejado recuerdos que llevaremos siempre con nosotras.

Finalmente, a todas aquellas personas que, de una u otra forma, contribuyeron a la realización de esta tesis, ya sea con su tiempo, conocimientos o palabras de aliento. A cada uno de ustedes, gracias por ser parte de este logro que es tan nuestro como suyo.

Jamileth Estela Anchundia Anchundia

Carol Pamela Vera Castro

Índice de Contenidos

Resumen.....	22
Abstract.....	23
CAPÍTULO I.....	24
1. Introducción.....	24
1.1. Presentación del tema.....	25
1.2. Ubicación y contextualización de la problemática.....	25
1.3. Planteamiento de problema.....	26
1.3.1 Problematización.....	26
1.3.2 Génesis del problema.....	26
1.3.3 Estado actual del problema.....	27
1.4 Diagrama causa – efecto del problema.....	28
1.5 Objetivos.....	29
1.5.1 Objetivo general.....	29
1.5.2 Objetivos específicos.....	29
1.6 Justificación.....	30
1.7 Impactos esperados.....	30
1.7.1 Impacto tecnológico.....	30
1.7.2 Impacto Social.....	31
1.7.3 Impacto Ecológico.....	31

CAPITULO II	32
2. Marco teórico de la investigación	32
2.1 Antecedentes Históricos	32
2.1.1 Desde cuando existe la automatización en riego.....	32
2.1.2 Antecedentes de investigaciones relacionadas al tema presentado	33
2.2 Definiciones conceptuales (Contexto teórico)	33
2.2.1 Sistema de Riego.....	33
2.2.2 Riego por goteo.....	34
2.2.3 Riego por superficie.....	34
2.2.4 Automatización.....	35
2.2.5 Arquitectura IOT.....	35
2.2.6 Red de Sensores Inalámbricos (WSN).....	36
2.2.7 Heltec WiFi LoRa 32 V2	37
2.2.8 WIFI.....	38
2.2.9 Tecnología LoRa	39
2.2.10 Protocolo LoRaWAN	39
2.2.10 Yanmis Sonda de sensor.....	40
2.2.11 BME680 Temperatura Humedad Presión.....	42
2.2.12 Electroválvula	44
2.2.13 Sensor de flujo de agua	44

2.2.14	Raspberry Pi.....	45
2.2.15	Protocolo MQTT.....	46
2.2.16	Eclipse Mosquitto.....	47
2.2.17	Mango.....	47
2.2.18	Vetiver.....	48
CAPITULO III.....		50
3.	Marco Investigativo.....	50
3.1	Introducción.....	50
3.2	Tipos de Investigación.....	50
3.2.1	Cuantitativa.....	50
3.2.2	Aplicada.....	51
3.3	Métodos de Investigación.....	52
3.3.1	Método Lógico Inductivo.....	52
3.4	Fuentes y métodos para la obtención de datos.....	53
3.4.1	Fuentes de datos directos.....	53
3.4.2	Fuentes secundaria.....	54
3.5	Plan Estratégico para la Captura de Datos.....	54
3.5.1	Plan de recolección de datos.....	54
3.5.2	Población Objetivo y Estrategia de Muestreo.....	55

3.5.3 Evaluación de los procedimientos y métodos para la obtención de información	56
3.5.3.1 Encuesta	56
3.6 Presentación y análisis de datos	61
3.6.1 Respuestas obtenidas encuesta de diagnóstico de necesidades:	61
3.7 Análisis y Resultados: Informe Final	70
CAPITULO IV	72
4. Marco Propositivo	72
4.1 Introducción	72
4.2 Descripción de la Propuesta	72
4.3 Determinación de Recursos	74
4.3.1 Humanos	74
4.3.2 Tecnológicos	74
4.3.3 Económicos	76
4.4 Etapas operativas para el desarrollo de la propuesta	78
4.4.1 Enfoques metodológicos empleados en el desarrollo de la propuesta	78
4.4.2 Aplicación de la metodología	83
4.4.2.1 Fase preliminar del proyecto	84
4.4.2.2 Fase de planificación y estimación	86
4.4.2.3 Definición de funcionalidades	87

4.4.3 Fase de implementación.....	97
4.4.3.1 Control del administrador	97
4.4.3.2 Control del usuario.....	105
4.4.4.2 Herramientas y Librerías Utilizadas:	111
4.4.4.3 Entorno de Desarrollo	112
4.4.4.4 Revisión y retrospectiva.....	128
CAPITULO V	130
5. Evaluación de Resultados	130
5.1 Introducción	130
5.2 Presentación de resultados	131
5.2.1 Resultados según objetivos planteados	131
5.2.3 Selección de sensores y librerías.....	139
5.2.4 Establecimiento de sensores.....	140
5.2.5 Configuración Lora	141
5.2.6 Inicialización y Lectura de Sensores.....	141
5.2.7 Transferencia de información al servidor.....	142
5.2.8 Comprobación del enlace y la comunicación.....	144
5.2.9 Pruebas de lectura de datos	145
5.2.10 Transmisión y documentación de datos	146
5.2.11 Prueba de integración con la plataforma de node-red	148

5.2.12 Pruebas de Estabilidad y Robustez	152
5.2.12.1 Procedimiento y Resultados.....	152
5.3 Interpretación objetiva	152
CAPÍTULO VI.....	154
6. Conclusiones y Recomendaciones	154
6.1 Conclusiones	154
6.2 Recomendaciones.....	156
Bibliografía	158
Anexos.....	164

Índice de Ilustraciones

Ilustración 1. Ubicación del sistema	26
Ilustración 2. Fuente propia del diagrama.....	28
Ilustración 3. Esp32-lora	38
Ilustración 4. wifi	38
Ilustración 5. lorawan-lora	39
Ilustración 6. Yanmis Sonda de sensor.....	42
Ilustración 7. BME680	43
Ilustración 8. Electroválvula	44
Ilustración 9. Ilustración. Sensor flujo de agua.....	45
Ilustración 10. Raspberry Pi.....	46
Ilustración 11. Protocolo MQTT	46
Ilustración 12. Eclipse Mosquitto	47
Ilustración 13. planta de mango	48
Ilustración 14. Planta de vetiver.....	49
Ilustración 15. Ocupación actual del encuestado	61
Ilustración 16. Interpretación de la experiencia de los encuestados	63
Ilustración 17. Interpretación de nivel de interés	64
Ilustración 18. Consideración de mejorar el uso de agua a los encuestados.....	65
Ilustración 19. Experiencia sobre tecnología de comunicación a los encuestados	66

Ilustración 20. Nivel de conocimiento sobre sensores a los encuestados	67
Ilustración 21. Importancia sobre una retroalimentación.....	68
Ilustración 22. Opinión sobre aceptación y apoyo	69
Ilustración 23. Arquitectura de la solución plantada.....	73
Ilustración 24. Inicio del VNC	98
Ilustración 25. Comprobación de identidad.	98
Ilustración 26. Autenticación en VNC server.....	99
Ilustración 27. Página principal.....	99
Ilustración 28. Dirigirse a Node-Red.	100
Ilustración 29. Iniciar las credenciales de Node-RED.	101
Ilustración 30. Configuración de Node-RED.....	102
Ilustración 31. Configuración de Node-Red	102
Ilustración 32. Visualización en Dashboard.....	103
Ilustración 33. Usuario-Contraseña MARIADB.....	103
Ilustración 34. Ingreso de la base de datos.....	104
Ilustración 35. Verificación Base de datos.	104
Ilustración 36. Verificación Dashboard.....	105
Ilustración 37. Plano de la estructura	106
Ilustración 38. Circuito integrado	109
Ilustración 39. Modelo conceptual de la base de datos.....	111

Ilustración 40. codigo Json dashboard.	113
Ilustración 41. codigo Json dashboard.	114
Ilustración 42. codigo Json dashboard.	114
Ilustración 43. codigo Json dashboard.	115
Ilustración 44. codigo Json dashboard.	115
Ilustración 45. codigo Json dashboard.	116
Ilustración 46. codigo Json dashboard.	116
Ilustración 47. codigo Json dashboard.	117
Ilustración 48. codigo Json dashboard.	117
Ilustración 49. codigo Json dashboard.	118
Ilustración 50. codigo Json dashboard.	119
Ilustración 51. codigo Json dashboard.	120
Ilustración 52. Código del ESP32 sobre el EMISON	121
Ilustración 53. Código del ESP32 sobre el EMISON	121
Ilustración 54. Código del ESP32 sobre el EMISON	122
Ilustración 55. Código del ESP32 sobre el EMISON	122
Ilustración 56. Código del ESP32 sobre el EMISON	123
Ilustración 57. Código del ESP32 sobre el EMISON	123
Ilustración 58. Código del ESP32 sobre el EMISON	124
Ilustración 59. Código del ESP32 sobre el EMISON	124

Ilustración 60. Código del ESP32 sobre el SERVIDOR.....	125
Ilustración 61. Código del ESP32 sobre el SERVIDOR.....	125
Ilustración 62. Código del ESP32 sobre el SERVIDOR.....	126
Ilustración 63. Código del ESP32 sobre el SERVIDOR.....	126
Ilustración 64. Código del ESP32 sobre el SERVIDOR.....	127
Ilustración 65. Código del ESP32 sobre el SERVIDOR.....	127
Ilustración 66. Levantamiento de información	132
Ilustración 67. Selección e investigación.....	133
Ilustración 68. Diseño del sistema	134
Ilustración 69. Programación del ESP32	135
Ilustración 70. Implementación y ensamblar componentes del sistema	136
Ilustración 71. Implementación y ensamblar componentes del sistema2	136
Ilustración 72. Implementación y ensamblar componentes del sistema 3	137
Ilustración 73. Prueba de funcionamiento.....	138
Ilustración 74. Prueba de funcionamiento del lugar asignado	138
Ilustración 75. Prueba de funcionamiento del lugar asignado 2	139
Ilustración 76. Librerías necesarias.....	140
Ilustración 77. Definición de sensores.....	140
Ilustración 78. Configuración lora	141
Ilustración 79. Inicio y lectura	142

Ilustración 80. Envío de datos.....	143
Ilustración 81. Comprobación de conexión y comunicación entre el ESP32 y los sensores.	144
Ilustración 82. Examen de lectura de información de los sensores y su procesamiento por el ESP32 -LORA para su representación visual	146
Ilustración 83. Comprobación de respuesta en Node-Red.....	147
Ilustración 84. Comprobación en la base de datos en MariaDB	147
Ilustración 85. Datos de temperatura	149
Ilustración 86. Datos de humedad.....	149
Ilustración 87. Datos presión.....	150
Ilustración 88. Datos del caudal	150
Ilustración 89. Total, de litros.....	151
Ilustración 90. Estado del flujo tenemos los dos casos encendido y apagado	151
Ilustración 91. Instalación en diversos entornos	164
Ilustración 92. Instalación del riego	165
Ilustración 93. Instalación del riego y detalles del riego.....	166
Ilustración 94. Tablero y servidor del riego	167
Ilustración 95. Datos recogidos visualizados en la base de datos visualización del 4 al 5 de diciembre del 2024.....	168
Ilustración 96. Datos recogidos visualizados en la base de datos visualización del 6 al 7 de diciembre del 2024.....	169

Ilustración 97. Datos recogidos visualizados en la base de datos visualización del 7 al 8 de diciembre del 2024.....	170
Ilustración 98. Datos recogidos visualizados en la base de datos visualización del 9 de diciembre del 2024.....	171
Ilustración 99. El antes y después del terreno que se eligió para la automatización.....	172
Ilustración 100. Automatización del riego con sus respectivas plantas	173
Ilustración 101. Automatización del riego en la actualidad.	174

Índice de Tabla

Tabla 1. Arquitectura IOT	36
Tabla 2. Algunos ejemplos de sensores o transductores comunes	42
Tabla 3. Tabulación de datos de encuestado primera pregunta	62
Tabla 4. Tabulación de datos segunda pregunta	63
Tabla 5. Tabulación de datos tercera pregunta	64
Tabla 6. Tabulación de datos cuarta pregunta	65
Tabla 7. Tabulación de datos quinta pregunta	66
Tabla 8. Tabulación de datos sexta pregunta.....	67
Tabla 9. Tabulación de datos séptima pregunta.....	69
Tabla 10. Tabulación de datos octava pregunta.....	70
Tabla 11. Recursos humanos y funciones	74
Tabla 12. Recursos tecnológicos	76
Tabla 13. Tabla sobre los económico	78
Tabla 14. Proyecto en equipo sobre la aplicación de la metodología	84
Tabla 15. Descripción de la visión del proyecto	85
Tabla 16. Definición de épicas o fases según Scrum	85
Tabla 17. Definición de los criterios de finalización.....	86
Tabla 18. Desglose de épicas y sus tareas	87
Tabla 19. Historia de usuario1	88

Tabla 20. Criterios de aceptación y casos límites CA1	89
Tabla 21. Historia de usuario 2	89
Tabla 22. Criterios de aceptación y casos límites CA2	90
Tabla 23. Historia de usuario 3	91
Tabla 24. Criterios de aceptación y casos límites CA3	91
Tabla 25. Historia de usuario 4	92
Tabla 26. Criterios de aceptación y casos límites CA4	93
Tabla 27. Historia de usuario 5	93
Tabla 28. Criterios de aceptación y casos límites CA5	94
Tabla 29. Historia de usuario 6	95
Tabla 30. Criterios de aceptación y casos límites CA6	96
Tabla 31. Historia de usuario 7	96
Tabla 32. Criterios de aceptación y casos límites CA7	97

Resumen

Este proyecto de titulación se enfoca en el diseño e implementación de un sistema automatizado de riego destinado a los cultivos de mango y vetiver, instalado en los exteriores del bloque de Informática de la Universidad Laica Eloy Alfaro de Manabí. Este sistema emplea sensores con tecnología ESP32 LoRa para optimizar la gestión del riego mediante el monitoreo en tiempo real de variables clave como la humedad del suelo, la temperatura ambiente y el flujo de agua, promoviendo la eficiencia y la sostenibilidad en la agricultura.

La base de datos MariaDB se utiliza para almacenar la información recopilada, mientras que un dashboard desarrollado en Node-RED permite la visualización en tiempo real del estado del sistema. A través de encuestas realizadas a 67 participantes, incluyendo estudiantes, docentes e ingenieros agrónomos, se identificaron necesidades y expectativas relacionadas con la automatización del riego, lo que permitió el diseño de un sistema orientado al usuario. La revisión de antecedentes investigativos resaltó la importancia de la automatización en la agricultura moderna, particularmente mediante la incorporación de tecnologías IoT para el monitoreo y control de variables críticas en los cultivos.

Los resultados obtenidos evidenciaron un elevado interés entre los encuestados, especialmente estudiantes, por la implementación de sistemas automatizados de riego, destacando la relevancia de integrar soluciones tecnológicas tanto en la educación como en el sector agrícola, fomentando prácticas sostenibles y eficientes.

En conclusión, el sistema desarrollado representa una herramienta práctica para el aprendizaje y la experimentación, al tiempo que constituye un modelo demostrativo aplicable a cultivos reales. Se sugiere realizar un mantenimiento periódico del sistema y considerar mejoras futuras, como la integración de algoritmos predictivos y el control automatizado de válvulas, con el fin de maximizar la eficiencia del riego. Este proyecto impulsa el desarrollo de una agricultura más sostenible y promueve la capacitación de profesionales especializados en la aplicación de tecnologías innovadoras.

Abstract

This thesis project focuses on the design and implementation of an automated irrigation system for mango and vetiver crops, installed outside the Informatics block at the Universidad Laica Eloy Alfaro de Manabí. This system uses ESP32 LoRa sensors to optimize irrigation management by monitoring key variables in real time, such as soil moisture, ambient temperature, and water flow, promoting efficiency and sustainability in agriculture.

The MariaDB database is used to store the collected information, while a dashboard developed in Node-RED allows real-time visualization of the system's status. Through surveys conducted with 67 participants, including students, teachers, and agronomists, needs and expectations related to irrigation automation were identified, enabling the design of a user-centered system. The review of previous research highlighted the importance of automation in modern agriculture, particularly through the incorporation of IoT technologies for monitoring and controlling critical variables in crops.

The results showed a high level of interest among the respondents, especially students, in the implementation of automated irrigation systems, underscoring the importance of integrating technological solutions in education and the agricultural sector, fostering sustainable and efficient practices.

In conclusion, the developed system represents a practical tool for learning and experimentation, while also serving as a demonstrative model applicable to real crops. Regular maintenance of the system is recommended, along with future improvements, such as the integration of predictive algorithms and automated valve control, to maximize irrigation efficiency. This project drives the development of more sustainable agriculture and promotes the training of professionals specialized in the application of innovative technologies.

CAPÍTULO I

1. Introducción

La agricultura y el manejo eficiente del agua son fundamentales para el desarrollo sostenible, especialmente en regiones donde los recursos hídricos son limitados. La implementación de tecnologías innovadoras se convierte en una herramienta esencial para optimizar el uso del agua en los cultivos. En la Universidad Laica Eloy Alfaro de Manabí (ULEAM), el área exterior del bloque de Informática alberga cultivos de mango y vetiver, que actualmente dependen de un sistema de riego manual mediante mangueras conectadas a un pozo. Este método tradicional no solo resulta ineficiente, sino que también provoca un desperdicio considerable de agua, ya que no se tiene control sobre la cantidad necesaria para cada cultivo. Además, la topografía del terreno, caracterizada por una loma, obliga a los trabajadores a realizar un esfuerzo físico significativo al desplazarse constantemente para irrigar las plantas, limitando su capacidad para llevar a cabo otras tareas.

Como estudiantes de tecnología, hemos identificado la necesidad de implementar un sistema de riego automatizado. Este sistema estará diseñado y desarrollado utilizando un microprocesador ESP32 para los exteriores del bloque de Informáticas. Esta solución busca no solo optimizar el uso del agua, sino también facilitar la labor de los trabajadores, reduciendo la carga física y mejorando la eficiencia operativa.

A través de un levantamiento de información exhaustivo, la investigación de tecnologías adecuadas y el diseño e implementación de un sistema que integre sensores y actuadores, pretendemos transformar el proceso de riego en el área, alineándolo con las demandas actuales de sostenibilidad y eficiencia.

El objetivo de esta tesis es desarrollar e implementar un sistema de riego automatizado que permita el monitoreo en tiempo real de las necesidades hídricas de los cultivos, promoviendo así un uso más responsable y eficiente del agua en la ULEAM. Esperamos que esta investigación contribuya al desarrollo de prácticas agrícolas más sostenibles y al bienestar de la comunidad universitaria.

1.1. Presentación del tema

SISTEMA DE RIEGO AUTOMATIZADO A TRAVÉS DE SENSORES A LOS EXTERIORES DEL BLOQUE DE INFORMÁTICA EN LA UNIVERSIDAD LAICA ELOY ALFARO DE MANABÍ

1.2. Ubicación y contextualización de la problemática

El espacio interactivo está localizado en la Universidad Laica Eloy Alfaro de Manabí (ULEAM), concretamente ubicado en la parte exterior del bloque de informática. Este espacio tiene poca producción de agua, en la cual el proceso de riego se realiza de forma manual, demandando gran esfuerzo físico y excesivo tiempo.

En el exterior del bloque de informática se puede visualizar que hay un pequeño espacio de terreno que consta con pocos cultivos y el riego lo hacen de manera manual.

En este espacio del terreno en esta área presenta condiciones áridas y una textura arenosa que limita su capacidad para retener humedad. Esto obliga al personal a realizar riegos manuales frecuentes, una labor que puede resultar monótona y desgastante, especialmente durante los periodos de alta temperatura, afectando potencialmente su bienestar físico.

Para solucionar esta situación, se plantea la implementación de un sistema automatizado de riego, que facilite el control y monitoreo de la irrigación en esta zona. El proyecto en mente refleja el compromiso de la universidad con la tecnología de vanguardia y el cuidado del medio ambiente, al tiempo que mejora la calidad de vida de la comunidad universitaria. La iniciativa busca promover el desarrollo saludable de la flora en el campus, con el fin de que en el futuro los estudiantes y empleados puedan disfrutar de sus beneficios, contribuyendo así al bienestar común y a un entorno más sostenible.

Ilustración 1. *Ubicación en donde se colocará el sistema de riego*

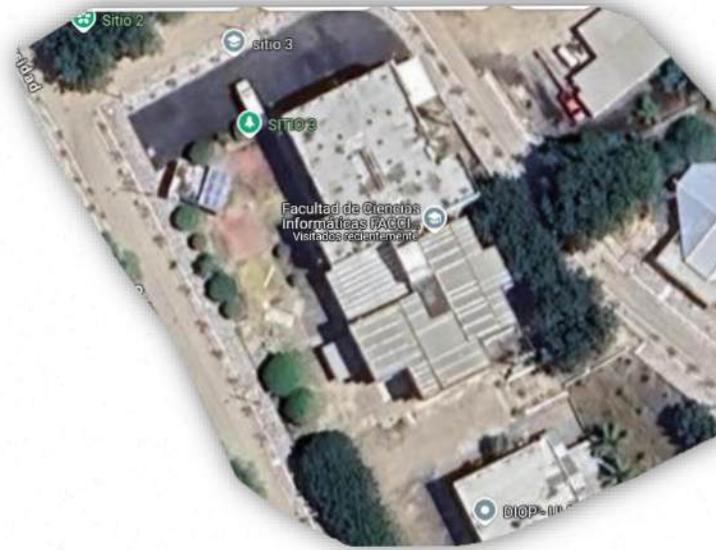


Ilustración 1. Ubicación del sistema

Nota: Imagen del bloque informática en la ULEAM

Fuente: Elaboración Propia de los Autores

1.3. Planteamiento de problema

1.3.1 Problematicación

¿Por qué es importante la implementación de un sistema de riego automatizado a través de sensores en los exteriores del bloque de Informática de la Universidad Laica Eloy Alfaro de Manabí?

1.3.2 Génesis del problema

La eficiencia en el uso de agua de riego, además de su control de manera precisa, puede llegar a ser un desafío crucial para los trabajadores en la Universidad Laica Eloy Alfaro de Manabí (ULEAM). En el caso del espacio interactivo ubicado en la parte exterior del bloque de

informática, enfrenta una clara problemática: la falta de un sistema de riego inteligente y automatizado.

1.3.3 Estado actual del problema

El área exterior del bloque de Informática en la Universidad Laica Eloy Alfaro de Manabí (ULEAM) se caracteriza por una topografía irregular, con una loma que alberga cultivos de mango y vetiver. Actualmente, el riego de estas plantas se realiza de manera manual mediante mangueras conectadas a un pozo. Este método de irrigación resulta ineficiente por varias razones.

Tomando en cuenta que no se tiene control sobre la cantidad exacta de agua que requieren los cultivos, lo que genera un desperdicio significativo de este recurso.

Además, la topografía irregular del terreno obliga a los trabajadores a desplazarse repetidamente cuesta arriba y cuesta abajo para realizar el riego. Esta actividad demanda un esfuerzo físico considerable y un tiempo excesivo, afectando otras labores que los trabajadores deben realizar.

Por lo tanto, se plantea la necesidad de implementar un sistema de riego automatizado, que permita optimizar el uso del agua, mejorar la eficiencia en el riego y reducir el esfuerzo físico del personal. Este sistema incluiría sensores para monitorear las necesidades hídricas de los cultivos en tiempo real, lo que permitiría un riego más preciso y eficiente, acorde a las características específicas del terreno y de las plantas.

1.4 Diagrama causa – efecto del problema

Ilustración 2. Diagrama Causas-Efectos.

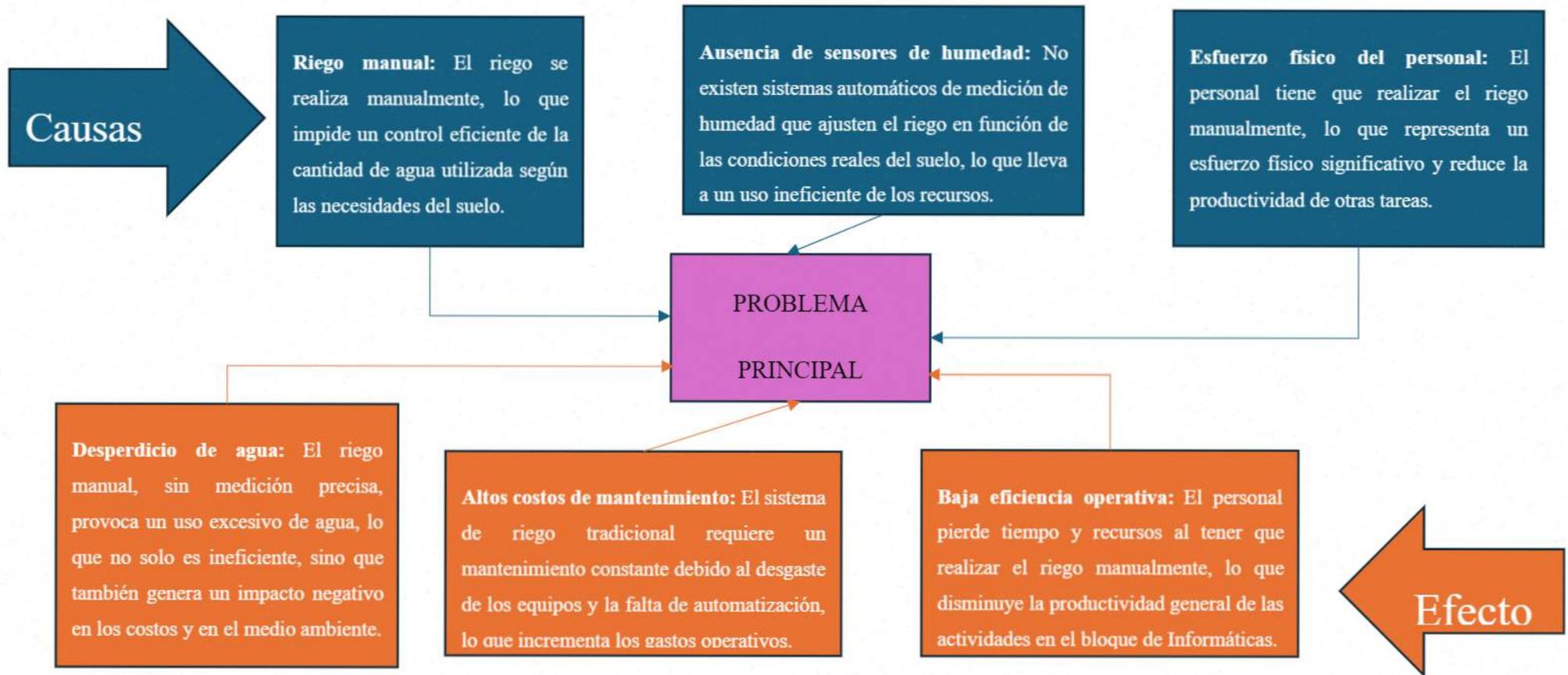


Ilustración 2. Fuente propia del diagrama

Nota: Imagen del diagrama Causa-Efecto.

Fuente: Elaboración Propia de los Autores

1.5 Objetivos

1.5.1 Objetivo general

Implementar un sistema de riego automatizado a través de sensores a los exteriores del bloque de informáticas en la Universidad Laica Eloy Alfaro de Manabí

1.5.2 Objetivos específicos

- Realizar el levantamiento de información y análisis de los procesos necesarios para la automatización del sistema de riego en los exteriores del bloque de Informática.
- Investigar y seleccionar las tecnologías y sensores más adecuados para monitorear las necesidades hídricas del área.
- Diseñar un sistema de riego automatizado utilizando microcontroladores ESP32 con tecnología Lora, sensores y actuadores para su óptimo funcionamiento.
- Programar el ESP32 para la gestión de los datos provenientes de los sensores mediante la tecnología Lora y configurar el Raspberry Pi para procesar y almacenar esta información así permitiendo la supervisión remota y eficiente del sistema de riego automatizado.
- Implementar y ensamblar los componentes del sistema, incluyendo las partes eléctricas y electrónicas, para garantizar el monitoreo preciso de las condiciones de riego.
- Realizar pruebas de funcionamiento del sistema de riego, asegurando la conectividad a internet y el monitoreo preciso de los niveles de humedad del suelo.

1.6 Justificación

La necesidad de abordar la problemática existente en el sistema de riego en el espacio interactivo de la Universidad Laica Eloy Alfaro de Manabí (ULEAM), ubicado en la parte exterior del bloque de informática, se basa en cumplir el objetivo de mejorar la eficiencia en el uso del agua y optimizar la productividad de los cultivos que se encuentran allí.

El sistema de riego utilizado en este espacio se basa en métodos convencionales de riego manual, lo cual genera ineficiencia y un uso excesivo del agua. Esta falta de eficiencia se debe a la ausencia de automatización y control del riego, causando que los trabajadores dejen de realizar otras labores y se centren en el riego manual. Por tal motivo, implementar un sistema de riego inteligente controlado mediante un prototipo ESP32 y monitoreado a través de un aplicativo web es fundamental.

Además, el agua es un recurso limitado y valioso, especialmente en las zonas urbanas. Con este proyecto, podremos optimizar y evitar el uso excesivo de este recurso, contribuyendo a la conservación de los recursos hídricos y con el aplicativo web, se podrá tener un mayor control y monitoreo del sistema de riego en tiempo real desde cualquier ubicación, brindando una mayor flexibilidad y eficiencia en la gestión del riego.

Con todo lo propuesto, la implementación de un sistema de riego inteligente controlado mediante un prototipo ESP32 en el espacio interactivo del bloque de informática de la ULEAM se ve justificada. Esto permitirá mejorar la eficiencia del riego, ahorrar agua, reducir costos operativos, y garantizar un mayor control y monitoreo. Además, este proyecto tendrá un impacto positivo en la universidad al impulsar la innovación en los sistemas de riego.

1.7 Impactos esperados

1.7.1 Impacto tecnológico

La implementación de un sistema de riego inteligente en el espacio exterior del bloque de informática de la ULEAM permitirá la automatización del proceso de riego, lo que mejorará la eficiencia en el uso del agua. Además, el uso de un aplicativo móvil facilitará el acceso y control remoto del riego, apoyando la adopción de tecnologías móviles en este ámbito. Esto podría llevar

a la futura implementación de soluciones similares en otros contextos dentro de la universidad y más allá.

Con la implementación de este proyecto, se contribuirá al desarrollo de una cultura tecnológica en la comunidad universitaria, fomentando el uso de soluciones tecnológicas en otros aspectos de la vida académica y administrativa. Las capacitaciones y el desarrollo de habilidades tecnológicas necesarias para operar el sistema serán clave para este avance.

El éxito de este proyecto podría servir de modelo para que otras instituciones educativas o comunidades adopten sistemas similares para enfrentar desafíos relacionados con el manejo del riego. Esto tendría un impacto tecnológico más amplio, extendiéndose a nivel provincial e incluso nacional.

1.7.2 Impacto Social

El impacto social abarcará varios aspectos, tales como: el bienestar de los trabajadores y estudiantes, la promoción de prácticas sostenibles, y el desarrollo de una comunidad universitaria más eficiente y tecnológicamente avanzada. La elaboración de este tipo de proyectos puede mejorar las condiciones de vida y de trabajo en la universidad, además de promover una cultura de sostenibilidad y resiliencia.

1.7.3 Impacto Ecológico

Implementar soluciones tecnológicas llevará a un uso eficiente del agua, reduciendo su consumo innecesario y el desperdicio. Esto ayudará a conservar y preservar las fuentes de agua locales, contribuyendo a la sostenibilidad de los ecosistemas acuáticos.

La optimización del riego y la entrega precisa de agua protegerán el suelo de la erosión y mantendrán la biodiversidad, lo que contribuirá a la reactivación del suelo como sistema biológico. En cuanto a la productividad, esta aumentará y se disminuirá el laboreo, beneficiando así a la comunidad universitaria en su conjunto.

CAPITULO II

2. Marco teórico de la investigación

2.1 Antecedentes Históricos

2.1.1 Desde cuando existe la automatización en riego.

A lo largo del tiempo, las técnicas de riego han evolucionado de manera significativa, desempeñando un rol clave en la transformación de la agricultura. Desde los métodos tradicionales utilizados por civilizaciones antiguas hasta los sistemas modernos como el riego por goteo, cada avance ha mejorado la eficiencia en el uso del agua y la productividad agrícola. En nuestra tesis, desarrollada conjuntamente, abordamos la problemática actual de la escasez de agua, un desafío que exige soluciones innovadoras, sostenibles y accesibles. Nos enfocamos en la implementación de tecnologías adaptables que promuevan la optimización de los recursos hídricos, contribuyendo así a la sostenibilidad y el desarrollo agrícola en el contexto local. (Ecuador, 2022).

El riego automatizado ha revolucionado la agricultura al ofrecer soluciones eficientes para la distribución del agua. Según (Olmos, 2025), estos sistemas no solo optimizan el uso del agua, sino que también mejoran la calidad y el rendimiento de los cultivos al asegurar que las plantas reciban la cantidad exacta de agua necesaria en todo momento. Gracias a la integración de tecnologías avanzadas como sensores, controladores y sistemas de monitoreo, el riego automatizado se adapta a las condiciones específicas del suelo y el clima, lo que permite un manejo más preciso y sostenible de los recursos hídricos. Estos avances son fundamentales para hacer frente a la escasez de agua, un problema crítico en muchas áreas agrícolas del mundo, garantizando la sostenibilidad y eficiencia en el uso del recurso. (Olmos, 2025)

2.1.2 Antecedentes de investigaciones relacionadas al tema presentado

El estudio de uso de sensores de humedad del suelo para la programación automatizada del riego en un cultivo de ciruelas realizado en suiza está muy relacionado con nuestro trabajo de titulación, ya que los resultados mostraron que el uso de sensores no solo optimiza el consumo de agua, sino que también mejora el rendimiento del cultivo. Además, se encontró que la automatización puede reducir significativamente la cantidad de agua utilizada sin afectar la producción agrícola (Millán, Casadesús, Carlos, Moñino, & Prieto¹, 2019).

El estudio sobre el uso de sensores de humedad del suelo para la gestión del riego en campos de soja se relaciona con nuestro trabajo de titulación, ya que demuestra cómo el uso de estos sensores ayuda a optimizar el consumo de agua. Los resultados indican que al integrar sensores con modelos de cultivos, se puede gestionar el riego de forma más precisa, reduciendo el uso de agua sin afectar la producción. Este enfoque permite una agricultura más sostenible y eficiente. (Hodges, Tagert, & Paz, 2022).

El monitoreo de la humedad del suelo es esencial para optimizar la producción agrícola, conservar agua y reducir impactos ambientales. Los sensores de humedad permiten gestionar el riego de manera eficiente, determinando la cantidad y el momento adecuado para aplicar agua. Esto no solo mejora el uso de recursos, sino que también contribuye a reducir costos y aumentar la rentabilidad de los cultivos. (Enciso, Porter, & Périès, 2007)

Estas investigaciones muestran cómo el uso de modelos de cultivos y sensores de humedad del suelo, de caudal y temperatura monitoreados y controlados por microcontroladores permiten el aumento en la productividad agrícola y la eficiencia del riego, de esta forma se puede combatir con los desafíos de la escasez de agua y mejorar la sostenibilidad en la agricultura.

2.2 Definiciones conceptuales (Contexto teórico)

2.2.1 Sistema de Riego.

Un sistema de riego se puede definir como un conjunto de estructuras y tecnologías diseñadas para suministrar agua a las plantas y cultivos de manera controlada y eficiente. Estos sistemas hacen posible el cultivo en diversas áreas geográficas, independientemente de las

condiciones climáticas locales o la disponibilidad natural de agua. La implementación de sistemas de riego permite la optimización del uso del agua, asegurando que los cultivos reciban la cantidad necesaria para su crecimiento y desarrollo sin desperdiciar recursos hídricos. (Serigio, 2021)

En el artículo de Innovaciones sobre sistemas de riego, se explica cómo estos sistemas permiten optimizar el uso del agua en la agricultura. A través de la implementación de tecnologías de riego, se puede asegurar que los cultivos reciban el agua adecuada para su desarrollo sin desperdiciar recursos. Estos sistemas son fundamentales para permitir la producción agrícola en regiones con escasez de agua o climas difíciles, contribuyendo así a la sostenibilidad y eficiencia en la agricultura. (Innovatione, 2019).

2.2.2 Riego por goteo

El riego por goteo es un sistema de irrigación altamente eficiente y localizado, diseñado para suministrar agua de manera precisa en forma de gotas directamente a la zona radicular de cada planta. Este método se implementa a través de una red de tuberías y pequeños emisores conocidos como goteros, que están estratégicamente ubicados para garantizar que el agua llegue exactamente donde se necesita. A diferencia de otros métodos de riego que pueden desperdiciar grandes cantidades de agua debido a la evaporación, escorrentía o infiltración profunda, el riego por goteo minimiza estas pérdidas al proporcionar una dosificación controlada y constante de agua (Valdivieso, 2024) el riego por goteo es particularmente beneficioso en regiones con limitaciones hídricas o con suelos de baja capacidad de retención de agua. Su capacidad para mantener una humedad uniforme y constante en la zona radicular mejora la eficiencia del uso de nutrientes, permitiendo que las plantas absorban de manera más efectiva los elementos esenciales para su crecimiento. Este sistema también puede ser automatizado y ajustado para adaptarse a las necesidades específicas de diferentes cultivos y condiciones del suelo, proporcionando una herramienta versátil y adaptable para los agricultores.

2.2.3 Riego por superficie

El riego por superficie es un método de irrigación ampliamente utilizado, especialmente recomendado para terrenos llanos o con pendientes muy suaves donde no es necesario realizar una explanación del suelo. La evitación de la explanación no solo reduce los costos asociados a la

preparación del terreno, sino que también previene posibles impactos negativos sobre la estructura y fertilidad del suelo. Este método de riego es el menos costoso tanto en términos de instalación como de mantenimiento, lo que lo convierte en una opción accesible para muchos agricultores, especialmente en regiones con limitaciones económicas (Gómez, y otros, 2010).

2.2.4 Automatización

La implementación de la automatización trae consigo numerosos beneficios, incluyendo la reducción de costos operativos, el incremento de la calidad y consistencia de los productos y servicios, y la posibilidad de liberar a los trabajadores de tareas repetitivas para que puedan enfocarse en actividades más estratégicas y creativas. No obstante, también plantea desafíos significativos, como la necesidad de capacitar a la fuerza laboral en nuevas habilidades y la adaptación a un entorno de trabajo en constante evolución tecnológica (Pérez, 2022).

2.2.5 Arquitectura IOT

La arquitectura de IoT es un sistema de elementos constituyentes, estructura de red y tecnologías de nube que funcionan de conformidad con los protocolos y estándares de seguridad de IoT establecidos.

Tabla 1. Arquitectura IOT

ARQUITECTURA IOT	
Capa de Detección	Sensores y objetos físicos que permiten la obtención de datos.
Capa de Intercambio de Datos	Transmisión eficiente y transparente de datos a través de diversas redes de comunicación.
Capa de integración de la información	Procesamiento de la información obtenida, filtrado de datos no relevantes y conversión de información clave en conocimiento útil para servicios y usuarios finales.

Tabla 1. Arquitectura IOT

Nota: Tabla relacionado sobre la arquitectura IOT.

Fuente: Internet de las cosas (Salazar & Santiago), página 16

2.2.6 Red de Sensores Inalámbricos (WSN)

El monitoreo ambiental desempeña un papel fundamental en la comprensión y gestión de los cambios en nuestro entorno, especialmente en un contexto donde la sostenibilidad y la protección del medio ambiente son prioridades esenciales. Una herramienta tecnológica de gran utilidad en este ámbito es la Red de Sensores Inalámbricos (WSN), la cual está compuesta por sistemas de dispositivos que recopilan y transmiten datos en tiempo real sobre distintas variables ambientales. Este blog explorará el funcionamiento de las WSN para el monitoreo ambiental y destacará las soluciones ofrecidas por "Sensor Vital" como integrador de sensores para el monitoreo automático del entorno. (Urdaneta, Pérez, & Custodio, 2014)

Las WSN están compuestas por múltiples sensores distribuidos en diferentes ubicaciones que comunican sus lecturas a través de conexiones inalámbricas. Estos sensores pueden ser desplegados en áreas remotas y de difícil acceso, permitiendo una cobertura amplia y detallada del entorno monitoreado. La recopilación y transmisión de datos en tiempo real permite la detección temprana de cambios y anomalías, facilitando una respuesta rápida y eficiente. (Urdaneta, Pérez, & Custodio, 2014)

Además, destacaremos las soluciones brindadas por "Sensor Vital", un integrador de sensores especializado en el monitoreo automático del entorno. "Sensor Vital" ofrece una gama de dispositivos y sistemas diseñados para proporcionar datos precisos y confiables, mejorando la capacidad de monitoreo y gestión ambiental. Sus soluciones abarcan desde sensores individuales hasta sistemas integrados que proporcionan una visión holística del estado del entorno (Urdaneta, Pérez, & Custodio, 2014).

2.2.7 Heltec WiFi LoRa 32 V2

WiFi LoRa 32 es una placa de desarrollo clásica de IoT diseñada y producida por Heltec Automation(TM). Es un producto altamente integrado basado en ESP32 + SX127x. Tiene funciones de Wi-Fi, BLE, LoRa, sistema de gestión de batería de Li-Po y OLED de 0,96 pulgadas. Es la mejor opción para ciudades inteligentes, granjas inteligentes, hogares inteligentes y fabricantes de IoT. (heltec, 2024).

Características principales

- Microprocesador: ESP32 (MCU de 32 bits de doble núcleo + núcleo ULP), con chip de nodo LoRa SX1276/SX1278.
- Interfaz micro USB con regulador de voltaje completo, protección ESD, protección contra cortocircuitos, blindaje RF y otras medidas de protección.
- Interfaz de batería SH1.25-2 incorporada, sistema de gestión de batería de litio integrado (gestión de carga y descarga, protección contra sobrecarga, detección de energía de la batería, conmutación automática de energía de USB / batería).
- La pantalla OLED de matriz de puntos 128*64 de 0,96 pulgadas incorporada se puede utilizar para mostrar información de depuración, nivel de batería y otra información. (“Placa De Desarrollo 0.96 Oled Sx1262 Wifi Ble Esp32 Lora32 V”)
- WiFi integrado, LoRa, tres conexiones de red Bluetooth, Wi-Fi integrado, antena de resorte de metal de 2,4 GHz dedicada a Bluetooth, interfaz IPEX (U.FL) reservada para uso de LoRa.
- Chip de puerto USB a serie CP2102 integrado, conveniente para descargar programas y depurar información de impresión.
- Apoyar el entorno de desarrollo Arduino.
- Con un buen diseño de circuito de RF y un diseño básico de bajo consumo (corriente de suspensión $\leq 800 \mu\text{A}$).

Ilustración 3. ESP32-Lora

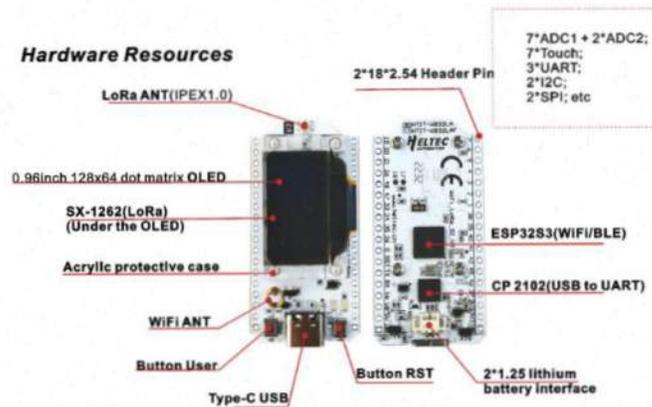


Ilustración 3. Esp32-lora

Nota: imagen sobre ESP32-Lora

Fuente: (heltec, 2024).

2.2.8 WIFI

Wi-Fi, abreviatura de Wireless Fidelity (fidelidad inalámbrica), es una tecnología de redes inalámbricas que facilita la conexión entre dispositivos electrónicos a través de frecuencias de radio. Esta tecnología, conocida como red inalámbrica de área local (WLAN, por sus siglas en inglés), permite que dispositivos como smartphones, tablets, laptops y computadoras de escritorio se conecten a internet y se comuniquen entre sí sin la necesidad de cables físicos, a diferencia de las conexiones realizadas mediante puertos Ethernet. (Proofpoint., 2024).

Ilustración 4. Wifi



Nota: Wifi

Ilustración 4. wifi

Fuente: (Domodesk, Domótica,

IoT, WIFI6, 2019)

2.2.9 Tecnología LoRa

La tecnología LoRa (Long Range) facilita la implementación de redes IoT, cuya demanda sigue creciendo. Para su funcionamiento, es fundamental contar con soluciones que optimicen el consumo energético y garanticen un amplio alcance de la red, capaz de conectar todos los dispositivos involucrados. Con el propósito de entender su operación en profundidad, este artículo detalla las características y beneficios que ofrece LoRa. (Artículo, 2022)

2.2.10 Protocolo LoRaWAN

LoRaWAN es el protocolo que dicta como se envían y reciben los paquetes de datos de sensores u otros componentes de la red y la forma de tratar esos paquetes. (Artículo, 2022)

Ilustración 5. *Lorawan-lora*



Ilustración 5. lorawan-lora

Nota: imagen sobre Lorawan-lora

Fuente: (Lorawan-lora, 2024)

2.2.10 Yanmis Sonda de sensor

Los sensores de temperatura desempeñan un papel crucial en la climatización de interiores, siendo ampliamente utilizados en viviendas, oficinas y otros espacios. Su función principal radica en medir la temperatura del aire o del agua, proporcionando datos precisos que permiten mantener condiciones ambientales confortables y eficientes. También denominados sondas de temperatura, estos dispositivos operan mediante componentes electrónicos capaces de generar señales o corrientes en función de la temperatura registrada. Destacan por su rápida capacidad de respuesta al medir la temperatura seca del aire.

Para obtener una medición más completa de la entalpía del aire, que incluye la humedad, es indispensable integrar un sensor combinado de temperatura y humedad relativa. Estas sondas permiten determinar parámetros fundamentales del aire, como la temperatura seca, la humedad absoluta y el punto de rocío, a través del uso de un diagrama psicrométrico. Su implementación resulta esencial para garantizar un control ambiental preciso en aplicaciones donde la eficiencia energética y el confort térmico son prioritarios. (S&P, 2024).

Los sensores o transductores son sensibles a una propiedad física y la convierten en un nivel de voltaje proporcional al valor de la propiedad que se mide. Un ejemplo común es una sonda de corriente. Convierte la corriente eléctrica en un voltaje eléctrico donde el nivel de voltaje es proporcional al nivel de corriente medido. (Spectrum, 2024)

Tabla 2. Algunos ejemplos de sensores o transductores comunes

Sensores		
Tipo de sensor/transductor	Propiedad física medida	Unidades de medida típicas
Sensores de corriente Sonda de corriente, derivación o transformador	Actual	Amperio

Acelerómetro		Aceleración	de g
Celda de carga o de fuerza		Fuerza	Newtons o libras fuerza
Transductor de presión		Presión	Pascales o libras por pie cuadrado (psi)
Micrófono		Nivel de presión sonora	Pascales
Termómetro	Termopar, Termistor o Detectores de Temperatura de Resistencia (RTD)	Temperatura	Grados Celsius o Fahrenheit
Tacómetro	Captación óptica, Captación por reluctancia	Velocidad rotacional	Revoluciones por minuto (RPM)
Sensores	ópticos	Iluminancia (potencia óptica incidente sobre una superficie)	Lux (lúmenes/m ²)
Fotodiodos, fotomultiplicadores, fototransistores o CCD			
Sensor de efecto Hall de campos magnéticos		Densidad de flujo magnético (B)	Tesla
Detectores de partículas Multiplicadores de		Radiación y partículas	Contar

electrones Detectores de radiación

Tabla 2. Algunos ejemplos de sensores o transductores comunes

Nota: tabla relaciona sobre algunos jemplos de sensores o transductores comunes

Fuente: Uso de sondas y sensores con digitalizadores modulares (Spectrum, 2024)

Ilustración 6. *Yanmis Sonda de sensor*



Ilustración 6. Yanmis Sonda de sensor

Nota: imagen de Yanmis Sonda de sensor

fuate (Sonda De Sensor Interfaz I2c Temperatura Del Suelo Sens..., 2024)

2.2.11 BME680 Temperatura Humedad Presión

BME680 Sensor es un pequeño módulo de presión atmosférica de bajo consumo, permite detectar precio, temperatura, humedad y gas. Este sensor cuenta con 6 pines para facilitar la conexión con MCUs o placas de desarrollo de Arduino y otras, tiene 2 interfaces de comunicación por I2C y SPI. (UNIT, 2024).

Tarjeta sensor de humedad temperatura y presión. Es capaz de detectar también concentraciones de gases y alcoholes. Pueder ser usado en drones y también en aplicaciones de contorl de calidad del aire interior. (Electrónicas, 2024). Sus características incluyen :

- Voltaje de funcionamiento: 1.7-3.6V
- Interfaz de comunicación: I2C (de hasta 3.4MHz) y SPI (3 y 4 cables de hasta 10MHz)

- Los sensores de humedad, presión y gas pueden ser habilitados o deshabilitados de forma individual
- De ultra-bajo consumo de corriente
- Temperatura de funcionamiento: -40 a 85°C
- Dimensiones: 18.6x16.6m

Pines de conexión:

- VCC: voltaje de alimentación
- GND: tierra
- SCL: Serial clock Input I2C Clock / SPI-CLOCK, pin de reloj o clock para I2C y SPI.
- SDA: Serial Data Input I2C Data /SPI-MOSI: Microcontroller Out Sensor In pin, pin de data en I2C y MOSI en SPI.
- SDO: Serial Data Out / SPI-MISO: Microcontroller In Sensor Out pin, pin MISO en SPI, no usado en I2C
- CS: SPI-chip select, no usado en I2C

Ilustración 7. BME680

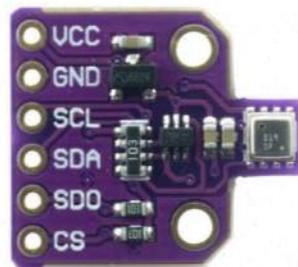


Ilustración 7. BME680

Nota: imagen relacionada a BME680 que nos ayuda a visualizarla precio, temperatura, humedad y gas.

Fuente: (Naylamp, 2024)

2.2.12 Electroválvula

En la mayoría de los sistemas destinados al control de caudales, resulta fundamental poder iniciar o detener el flujo del fluido en el circuito para gestionar su presencia en el sistema. Para este propósito, se emplean comúnmente electroválvulas, dispositivos que, al ser activados mediante un solenoide, permiten su instalación en ubicaciones remotas y su manejo a través de interruptores eléctricos simples.

Las electroválvulas representan uno de los componentes de control más utilizados en la tecnología de fluidos. Su funcionalidad incluye tareas como interrumpir, liberar, dosificar, distribuir o mezclar diferentes fluidos, lo que las hace indispensables en una amplia variedad de aplicaciones. Estas válvulas destacan por su capacidad de ofrecer conmutaciones rápidas y seguras, además de ser altamente duraderas, fiables y de bajo consumo energético. Su diseño compacto también las convierte en una solución eficiente y práctica para diferentes entornos industriales y tecnológicos. (Buerkert, 2024).

Ilustración 8. *Electroválvula*



Ilustración 8. Electroválvula

Nota: *imagen sobre Electroválvula*

Fuente: (electrovalvula, 2024)

2.2.13 Sensor de flujo de agua

Los sensores de flujo, también conocidos como "detector de flujo" o "interruptor de caudal" son equipos para monitoreo de fluidos en tuberías y funcionan con el desplazamiento de un pistón magnético que indica el aumento o disminución del flujo de líquido, accionando el contacto de un

interruptor de láminas (reed switch). El pistón es controlado por un resorte y regresa a la posición inicial cuando no hay fluido, incluso si hay presión en la tubería.

Los sensores de flujo Eicos son versátiles, permiten configurar en la propia pieza el caudal de disparo, por eso el mismo sensor puede ser utilizado en aplicaciones de diferentes caudales (dentro del rango de sensibilidad del sensor), así se obtiene un mejor control y monitoreo de los fluidos. (Eicos, 2024)

Ilustración 9. *Sensor flujo de agua*



Ilustración 9. Ilustración. Sensor flujo de agua

Nota: imagen sobre el Sensor flujo de agua

Fuente: (sensor flujo de agua, 2024)

2.2.14 Raspberry Pi

Es un ordenador de bajo coste y formato compacto destinado al desarrollo para hacer accesible la informática a todos los usuarios. La Raspberry Pi también se caracteriza por ser muy utilizada para desarrollar pequeños prototipos y para la formación sobre informática y electrónica en los colegios. (Solé, 2024)

Todos los diseños de Raspberry Pi se basan en el hardware libre y habitualmente se utilizan también sistemas operativos libres basados en GNU/Linux. Para este microordenador se ha desarrollado Raspberry Pi OS (antes conocido como Raspbian) que es una versión personalizada de Debian. (Solé, 2024).

Ilustración 10. *Raspberry pi*



Ilustración 10. Raspberry Pi

Nota: imagen sobre el Raspberry pi

Fuente: (raspberry pi, 2024)

2.2.15 Protocolo MQTT

MQTT es un protocolo de comunicación basado en estándares, diseñado específicamente para la transmisión de mensajes entre equipos. Es ampliamente utilizado en dispositivos de Internet de las cosas (IoT), como sensores inteligentes y dispositivos portátiles, que operan en redes con recursos limitados y ancho de banda reducido. La elección de MQTT en el ámbito IoT se debe a su facilidad de implementación y su capacidad para transmitir datos de manera eficiente. Este protocolo permite la mensajería tanto de los dispositivos hacia la nube como desde la nube hacia los dispositivos, convirtiéndose en una herramienta clave para la comunicación en sistemas IoT. (AWS, 2024).

Ilustración 11. Protocolo MQTT

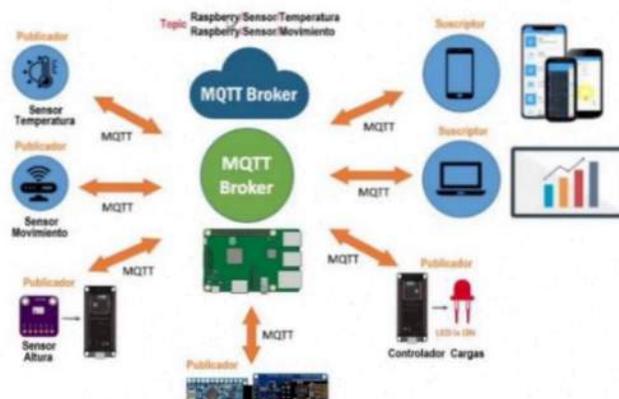


Ilustración 11. Protocolo MQTT

Nota: imagen de Protocolo MQTT

Fuente: (protocolo MQTT, 2024)

2.2.16 Eclipse Mosquitto

Eclipse Mosquitto es un broker de mensajería de código abierto, licenciado bajo EPL/EDL, que implementa las versiones 5.0, 3.1.1 y 3.1 del protocolo MQTT. Este software se caracteriza por su diseño liviano, lo que lo hace ideal para su implementación en una amplia gama de dispositivos, desde computadoras de placa única de bajo consumo hasta servidores de alto rendimiento. Su flexibilidad y eficiencia lo convierten en una herramienta clave para aplicaciones IoT que requieren una comunicación fiable y escalable. (Mosquitto, 2024)

El protocolo MQTT proporciona un método liviano para enviar mensajes mediante un modelo de publicación/suscripción. Esto lo hace adecuado para la mensajería de Internet de las cosas, por ejemplo, con sensores de bajo consumo o dispositivos móviles como teléfonos, computadoras integradas o microcontroladores. (Mosquitto, 2024).

Ilustración 12. *Eclipse Mosquitto*



Ilustración 12. Eclipse Mosquitto

Nota: imagen relacionada al Eclipse Mosquitto

Fuente: (Eclipse Mosquitto, 2024)

2.2.17 Mango

El mango es una fruta tropical rica en vitaminas y minerales, especialmente vitamina C y A, lo que la convierte en un excelente complemento para una dieta saludable. Además, el mango es conocido por sus propiedades antioxidantes y digestivas. Esta fruta es versátil en la cocina,

utilizada tanto en platos dulces como salados, y es especialmente apreciada en diversas culturas por su sabor único y su aporte nutricional. (García, 2025).

Ilustración 13. *Planta de mango*



Ilustración 13.planta de mango

Nota: imagen referente a la planta de mago

Fuente: (Editorial Prensa Alicantina, 2022)

2.2.18 Vetiver

El vetiver es una planta tropical reconocida por su capacidad para mejorar la calidad del suelo, prevenir la erosión y ser una fuente sostenible de aceite esencial. En la agricultura, se utiliza para estabilizar terrenos y mejorar la fertilidad del suelo, lo que la convierte en una herramienta clave en la agroindustria y la conservación ambiental. Además, el vetiver posee propiedades medicinales y aromáticas, siendo también valiosa en la industria cosmética y perfumera. (Agronet, 2022).

Ilustración 14. *Planta de vetiver*



Ilustración 14. Planta de vetiver

Nota: imagen referente a la planta de vetiver

Fuente: (Estévez, 2019)

2.3 CONCLUSIONES RELACIONADAS AL MARCO TEÓRICO EN REFERENCIA AL TEMA PLANTEADO.

En el desarrollo del marco teórico, se han identificado y analizado una serie de conceptos y tecnologías clave para la implementación de un sistema de riego automatizado. En primer lugar, se abordaron los diferentes sistemas de riego existentes, lo que nos permitió comprender las características fundamentales de cada uno. Esta información es crucial, ya que la elección del tipo de sistema influirá directamente en la eficiencia del uso del agua y los recursos en el proyecto.

Entre los sistemas de riego más utilizados, encontramos el riego por goteo, el riego por aspersión y el riego subterráneo. Cada uno de estos sistemas ofrece ventajas específicas dependiendo de las condiciones climáticas, el tipo de cultivo y las características del terreno. La revisión de los diferentes tipos de riego nos brindó una base sólida para decidir cuál es el más adecuado para nuestra implementación. Por ejemplo, el riego por goteo se destaca por su eficiencia en la distribución del agua, especialmente en zonas áridas, mientras que el riego por aspersión es ideal para áreas más extensas con cultivos uniformes.

En conclusión, el estudio teórico nos ha ofrecido una guía valiosa para la implementación de un sistema de riego automatizado eficiente y adaptado a las necesidades actuales. Las conclusiones extraídas de la revisión de la literatura no solo facilitan la toma de decisiones técnicas, sino que también nos orientan hacia una implementación sostenible y eficaz del proyecto.

CAPITULO III

3. Marco Investigativo

3.1 Introducción

En este capítulo se describe el proceso de selección de los métodos para la recolección de datos, fundamentado en la consulta de diversas fuentes de información, tanto primarias como secundarias. El propósito principal de esta selección fue garantizar la calidad y pertinencia de los datos recopilados. Para ello, se diseñó un plan estratégico que empleó técnicas reconocidas y validadas en el ámbito académico y profesional, asegurando la exactitud y representatividad de la información obtenida.

El enfoque metodológico implementado permitió emplear herramientas analíticas y métodos estadísticos adecuados para el tratamiento de la información. Los resultados fueron organizados de manera clara y estructurada, presentándose mediante tablas, gráficos y otras representaciones visuales que facilitan su comprensión e interpretación.

De esta manera, no solo se logró una recopilación exhaustiva de información relevante, sino también una presentación efectiva que respalda la toma de decisiones basadas en datos confiables y sólidos.

3.2 Tipos de Investigación

3.2.1 Cuantitativa

El enfoque cuantitativo se caracteriza por la medición y el análisis numérico de datos, recolectando información que puede ser cuantificada y luego analizada mediante técnicas

estadísticas. A diferencia del enfoque cualitativo, que se centra en comprender fenómenos desde la perspectiva de los participantes, el cuantitativo busca obtener resultados que permitan identificar relaciones causales y patrones a gran escala. Un ejemplo de su uso es un estudio sobre los efectos de un programa educativo en el rendimiento académico, donde se analizan datos como calificaciones y puntajes en pruebas estandarizadas. Su principal fortaleza es la capacidad para generar resultados generalizables y comparables, permitiendo tomar decisiones basadas en evidencia. Sin embargo, este enfoque puede no captar la complejidad de la experiencia humana como lo hace el enfoque cualitativo (Zúñiga, 2023) .

Se optó por este enfoque de investigación debido a la necesidad de abordar el problema desde una perspectiva que garantizara la obtención de datos precisos y específicos, lo cual resultaba fundamental para el desarrollo del sistema. La adopción de este método cuantitativo no solo proporcionó resultados medibles, sino que también permitió estructurar y llevar a cabo el proceso de investigación de manera eficiente y ordenada.

Para la recolección de datos, se realizó un diagnóstico basado en la identificación de necesidades y la determinación de parámetros óptimos, a través de encuestas cerradas. La información obtenida fue clave para avanzar en el desarrollo del sistema y asegurar que las soluciones implementadas respondieran a los requerimientos identificados. Este enfoque permitió que el proyecto se desarrollara de forma estructurada, basándose en datos concretos y respaldados.

3.2.2 Aplicada

La investigación aplicada recurre a los conocimientos ya alcanzados en la investigación básica para encaminarlos al cumplimiento de objetivos específicos; por tanto, este tipo de investigación considera todo el conocimiento existente en un área concreta, que será aplicado en el intento de solucionar problemas específicos.

"Son trabajos originales destinados a la obtención de nuevos conocimientos enmarcados en la solución de un problema específico basados en los resultados de la investigación básica; estos se pueden obtener en un periodo más corto, y su impacto en la sociedad es más destacado. (Grunbaum, 2023).

La investigación aplicada emerge como una respuesta concreta a la necesidad de interpretar y transformar una realidad social cada vez más compleja y dinámica. A diferencia de otros enfoques centrados en la generación de conocimiento teórico, la investigación aplicada se apoya en una base epistemológica robusta que le permite tender un puente entre el conocimiento teórico y su aplicación práctica. En este contexto, el conocimiento científico se convierte en una herramienta esencial para abordar problemas específicos y mejorar situaciones reales en diversos campos. Este enfoque se orienta, por tanto, a la traducción del conocimiento en soluciones prácticas que tengan un impacto directo y significativo en la sociedad (Grunbaum, 2023).

La decisión de adoptar un enfoque de investigación aplicada en este proyecto responde a la necesidad de establecer una base sólida y metodológica que permita el desarrollo estructurado del sistema. Este proyecto de riego automatizado, que emplea sensores de humedad y temperatura junto con la tecnología de comunicación LoRa en el microcontrolador ESP32, ha sido desarrollado y evaluado conforme a las recomendaciones y directrices provistas por Espressif, la empresa fabricante del ESP32. En este contexto, la Guía de Programación de ESP-IDF (Espressif IoT Development Framework) ha constituido una fuente de referencia esencial, brindando una descripción exhaustiva de las funcionalidades y capacidades del microcontrolador, además de proporcionar ejemplos prácticos que facilitan la implementación y configuración de aplicaciones IoT. Esta documentación ha permitido una integración eficaz de los sensores y módulos de comunicación, asegurando que el sistema de riego responda de manera automatizada y precisa a las variaciones en las condiciones ambientales, optimizando así el uso de recursos hídricos.

3.3 Métodos de Investigación

3.3.1 Método Lógico Inductivo

Con el fin de diagnosticar las necesidades relacionadas con la implementación de un sistema automatizado de riego mediante sensores, se llevaron a cabo encuestas dirigidas a estudiantes y personal académico. Las encuestas realizadas tuvieron como finalidad determinar el nivel de conocimiento y las expectativas respecto a la agricultura de precisión y los sistemas de riego automatizados. Además, se efectuaron observaciones minuciosas del espacio destinado a la instalación del sistema de riego, analizando las condiciones actuales del área y las herramientas disponibles para los estudiantes. A partir de los resultados obtenidos, se planteó la hipótesis de que

la implementación de un sistema de riego automatizado, diseñado de acuerdo con las necesidades identificadas y adaptado a las particularidades del espacio evaluado, contribuirá significativamente a optimizar la gestión del recurso hídrico. Además, se proyecta que dicho sistema constituirá una herramienta de aprendizaje y práctica para los estudiantes, potenciando su comprensión de las tecnologías de riego inteligente y su impacto en la optimización de recursos.

Para comprobar esta hipótesis, se diseñó sistema de riego automatizado, equipado con sensores de humedad y temperatura, los cuales fueron seleccionados cuidadosamente para monitorear los parámetros críticos definidos en el diagnóstico inicial. Este fue probado en un entorno controlado, verificando su precisión y efectividad en la recolección de datos en tiempo real, con el fin de asegurar su funcionalidad antes de su implementación definitiva en el área de estudio.

3.4 Fuentes y métodos para la obtención de datos

3.4.1 Fuentes de datos directos

Como fuente principal de información, se optó por realizar observaciones directas en el área de la ULEAM destinada a la colocación de mangos y vetiver, con el fin de identificar las necesidades fundamentales y evaluar las condiciones actuales para la implementación de un sistema de riego automatizado. Así mismo, se diseñaron encuestas dirigidas a profesionales y estudiantes, orientadas a identificar requerimientos específicos y a evaluar la aceptación y viabilidad de un sistema de automatización de riego en este contexto particular. Estas encuestas, estructuradas en torno a preguntas centradas en la experiencia actual con el riego y las expectativas sobre la automatización en el área, permitieron obtener datos precisos y específicos.

Dado que la automatización del riego puede ser un concepto novedoso para algunos en el entorno de estudio, fue esencial abordar estos aspectos de forma clara y detallada. Esta combinación de metodologías permitió no solo identificar los desafíos actuales en el manejo hídrico del área, sino también evaluar la pertinencia de implementar una solución automatizada adaptada a las necesidades específicas de los mangos y el vetiver en el área de estudio en la ULEAM.

3.4.2 Fuentes secundaria

Las fuentes secundarias utilizadas en este proyecto incluyen tesis relacionadas con la automatización de sistemas de riego, especialmente aquellas que abordan la implementación de tecnologías avanzadas como LoRa. Se llevó a cabo una revisión exhaustiva de proyectos relevantes en los antecedentes de investigación, cuyo análisis contribuyó a la construcción de un marco teórico sólido y coherente con los objetivos de este estudio. El propósito principal de esta revisión bibliográfica fue consolidar las bases teóricas del proyecto, ofreciendo un respaldo actualizado sobre los avances, innovaciones y buenas prácticas en el campo de la automatización de sistemas de riego, aspecto clave para diseñar una solución eficiente y adaptada al contexto específico del área de estudio.

3.5 Plan Estratégico para la Captura de Datos

3.5.1 Plan de recolección de datos

1. Identificación de fuentes bibliográficas

Se llevó a cabo una selección rigurosa de literatura académica y técnica relevante, enfocada en sistemas de riego automatizados y tecnologías IoT, con el objetivo de respaldar teóricamente el estudio.

2. Determinación de necesidades del sector

A través de encuestas y observaciones, se definieron las necesidades específicas del área de estudio, considerando tanto los aspectos técnicos como las expectativas de los usuarios finales.

3. Selección de segmentación y tamaño de la muestra.

Se establecieron criterios para segmentar a los participantes y determinar un tamaño de muestra representativo que garantizara la validez de los datos recolectados.

4. Aplicación de herramientas de recolección de datos

Se implementaron instrumentos como encuestas, entrevistas estructuradas y observaciones de campo para obtener información precisa y detallada.

5. Elaboración de gráficos para la tabulación y visualización de datos.

Los datos obtenidos fueron tabulados y representados mediante gráficos, tablas y otras herramientas visuales, facilitando su interpretación y análisis para la toma de decisiones.

3.5.2 Población Objetivo y Estrategia de Muestreo

En este proyecto se empleó la técnica de muestreo no probabilístico por conveniencia, seleccionando una población compuesta por sujetos accesibles y pertinentes para el objetivo del estudio. La muestra estuvo conformada por profesionales del sector, agricultores, y estudiantes de las carreras de Agroindustria y Tecnologías de la Información. Este enfoque permitió recolectar datos directos y específicos de individuos con experiencia y conocimiento en el área, lo que facilitó una comprensión más profunda de las necesidades y expectativas relacionadas con la implementación del sistema de riego automatizado.

- **Profesionales**

Docentes de las carreras de Agroindustria y de Tecnologías de la Información de la Facultad de Ciencias de la Vida y Tecnología, junto con ingenieros de Agropecuaria con profundo conocimiento en principios de riego automatizado, participaron en este proyecto. Su experiencia y pericia resultaron invaluable para orientar el desarrollo y evaluar la viabilidad técnica y académica de la solución, especialmente en lo referente a la implementación de automatización con ESP32 y tecnología LoRa.

- **Estudiantes**

Estos estudiantes constituyen la próxima generación de profesionales en los campos de la agroindustria y las tecnologías de la información. Su participación en este proyecto brindó una perspectiva contemporánea y aportó ideas innovadoras sobre el potencial del riego automatizado en la agricultura del futuro. Gracias a los conocimientos adquiridos durante sus estudios y sus prácticas en el uso de tecnologías aplicadas a la gestión agrícola, ellos serán los principales

usuarios de este sistema, lo cual permitirá optimizar y agilizar los procesos de riego, incrementando la eficiencia en las actividades productivas del sector.

La división de la población en categorías permitió obtener datos más específicos y direccionados. Esto hizo posible identificar y analizar en detalle las necesidades y perspectivas propias de cada grupo, permitiendo ajustar tanto las estrategias de recolección de información como el procesamiento de los resultados. Así, se logró recopilar información esencial para el diseño y perfeccionamiento del sistema de riego automatizado. Esta segmentación aseguró que cada grupo contribuyera con su conocimiento particular, enriqueciendo la comprensión integral del sistema y sus potenciales aplicaciones.

3.5.3 Evaluación de los procedimientos y métodos para la obtención de información

Los datos recopilados fueron integrados en la sección de análisis, donde se procedió a su tabulación y se derivaron conclusiones específicas.

3.5.3.1 Encuesta

Objetivo

Identificar las necesidades, expectativas y nivel de conocimiento de la población seleccionada en relación con los sistemas de riego automatizado, con el propósito de desarrollar una solución eficiente, innovadora y adaptada a las exigencias del sector. Este sistema está diseñado para maximizar el uso eficiente de los recursos hídricos, aumentar la productividad y promover prácticas sostenibles en cultivos específicos. Los datos recopilados servirán para gestionar, a través de sensores y la tecnología ESP32, junto con una interfaz para la visualización de información, la medición y el monitoreo preciso de parámetros clave. Esto garantizará un seguimiento constante del estado de las áreas de riego y apoyará la toma de decisiones informadas, promoviendo un manejo hídrico más eficiente.

Diseño de la encuesta

La encuesta, compuesta por 8 preguntas, fue aplicada a un grupo de expertos en riego automatizado, conformado por ingenieros agrónomos, con el propósito de identificar sus necesidades y expectativas en torno a la automatización, de modo que el sistema pueda ser ajustado

a las condiciones y requerimientos específicos del sector. La información obtenida resultó fundamental para el diseño de un sistema de automatización integral que optimice la eficiencia en el uso de recursos hídricos, incremente la productividad y contribuya a la rentabilidad en áreas de riego.

Este sistema de automatización, basado en tecnología ESP32 y comunicación LoRa, se apoyará en sensores especializados para la medición y control preciso de parámetros críticos, tales como la humedad del suelo y la temperatura ambiental. Esta infraestructura permitirá reducir costos operativos, incrementar la eficiencia de los procesos y facilitar la gestión hídrica en beneficio de los agricultores.

Sistema de riego automatizado a través de sensores

1. Seleccione su ocupación actual:

Estudiante

Docente Uleam

Ingeniero Agrónomo o afines

Esta pregunta tiene como finalidad identificar la ocupación actual de los encuestados, con el propósito de analizar la distribución del público objetivo y evaluar sus niveles de conocimiento en relación con los sistemas de riego automatizados. Esto permitirá identificar las necesidades específicas de cada grupo ocupacional y evaluar su interés en la implementación de este sistema en los exteriores del bloque de Informática en la Universidad Laica Eloy Alfaro de Manabí.

2. ¿Cuenta con experiencia previa en la implementación de sistemas automatizados en cualquier sector, ya sea en el ámbito agrícola u otros?

Si

No

Esta pregunta tiene como propósito identificar si los encuestados poseen experiencia previa en la implementación de sistemas automatizados en cualquier sector, ya sea agrícola u otro. Esta información será clave para evaluar el grado de familiaridad y conocimientos prácticos del público objetivo, así como para detectar posibles necesidades de apoyo técnico o educativo que faciliten la adopción del sistema de riego automatizado en los exteriores del bloque de Informática de la Universidad Laica Eloy Alfaro de Manabí.

3. ¿Qué tan importante considera la implementación de sistemas automatizados para el riego en comparación con los métodos tradicionales de riego manual?

Muy importante

Importante

Moderadamente importante

Poco importante

Nada importante

El propósito de esta pregunta es analizar la percepción de los encuestados respecto a la importancia de los sistemas de riego automatizados en comparación con los métodos tradicionales. Esto ayudará a identificar el grado de aceptación y valoración que tiene el público objetivo hacia la tecnología propuesta, así como su disposición para incorporar sistemas más eficientes en el riego de áreas específicas, como los exteriores del bloque de Informática de la Universidad Laica Eloy Alfaro de Manabí.

4. ¿Considera que la automatización del riego mediante sensores podría mejorar la eficiencia en el uso del agua en espacios exteriores de la ULEAM?

Si

No

Esta pregunta tiene como propósito explorar la percepción de los encuestados acerca de la capacidad de los sistemas de riego automatizados basados en sensores para optimizar el uso del

agua en los espacios exteriores de la Universidad Laica Eloy Alfaro de Manabí. Los resultados permitirán analizar el nivel de aceptación y las expectativas en torno a los beneficios ambientales y operativos que la implementación de este sistema podría ofrecer.

5. ¿Ha tenido alguna experiencia con la integración de tecnologías de comunicación (como redes inalámbricas, LoRa, ¿Wi-Fi) en sistemas automatizados?

Si

No

No estoy seguro

Esta pregunta tiene como finalidad determinar si los encuestados poseen experiencia previa en la implementación de tecnologías de comunicación, como redes inalámbricas, LoRa o Wi-Fi, en sistemas automatizados. Esta información será clave para evaluar el nivel de conocimiento técnico del público objetivo y detectar posibles oportunidades de capacitación o apoyo para facilitar la adopción de tecnologías complementarias al sistema de riego automatizado en los exteriores del bloque de Informática de la Universidad Laica Eloy Alfaro de Manabí.

6. ¿Cuál es su nivel de conocimiento sobre los sensores utilizados en sistemas de riego automatizados (como sensores de humedad del suelo, temperatura, etc.)?

Amplio conocimiento

Conocimiento moderado

Conocimiento básico

Sin conocimiento

El objetivo de esta pregunta es determinar el nivel de conocimiento de los encuestados sobre los sensores empleados en sistemas de riego automatizados, como sensores de humedad del suelo o temperatura. Esto permitirá identificar la familiaridad técnica del público objetivo con los

componentes esenciales del sistema, así como diseñar estrategias de capacitación adecuadas para optimizar su comprensión y uso en los exteriores del bloque de Informática en la Universidad Laica Eloy Alfaro de Manabí.

7. ¿Qué tan importante considera que es la retroalimentación de los usuarios en la mejora de sistemas de riego automatizados?

Muy importante

Moderadamente importante

Poco importante

Nada importante

Esta pregunta tiene como propósito analizar la percepción de los encuestados acerca de la relevancia de la retroalimentación de los usuarios en la mejora continua de los sistemas de riego automatizados. Los resultados permitirán identificar el nivel de interés del público objetivo en contribuir activamente a la optimización del sistema, garantizando que este se ajuste de manera más precisa a las necesidades específicas de los exteriores del bloque de Informática de la Universidad Laica Eloy Alfaro de Manabí.

8. ¿Cuál es su opinión sobre la aceptación y el apoyo de la comunidad universitaria hacia la implementación de un sistema automatizado de riego y gestión de circuitos eléctricos?

Muy positiva

Positiva

Negativa

Muy negativa

El propósito de esta pregunta es explorar la percepción de los encuestados respecto a la aceptación y el respaldo de la comunidad universitaria hacia la implementación de un sistema automatizado de riego y gestión de circuitos eléctricos. Este análisis permitirá evaluar el nivel de

apoyo y las expectativas de la comunidad, identificando elementos clave para asegurar una integración exitosa del sistema en los exteriores del bloque de Informática de la Universidad Laica Eloy Alfaro de Manabí.

3.6 Presentación y análisis de datos

3.6.1 Respuestas obtenidas encuesta de diagnóstico de necesidades:

La encuesta tuvo como objetivo recopilar información precisa y organizada sobre el interés, la experiencia y las percepciones de los encuestados respecto a los sistemas de riego automatizados y las tecnologías relacionadas. Esta información busca servir como base para un análisis detallado de las necesidades y expectativas del público objetivo, contribuyendo al diseño e implementación eficiente del proyecto. La encuesta se dirigió a un grupo diverso que incluyó tanto a estudiantes como a profesionales de la Universidad Laica Eloy Alfaro de Manabí, con el propósito de obtener una visión integral y representativa de la comunidad universitaria.

1. Seleccione su ocupación actual:

Los resultados de la encuesta reflejaron una mayor participación de estudiantes en comparación con docentes de la ULEAM e ingenieros agrónomos o profesionales afines. Este hallazgo evidencia un significativo interés y compromiso por parte de los estudiantes respecto a la implementación de un sistema automatizado de riego en los exteriores del bloque de Informática de la Universidad Laica Eloy Alfaro de Manabí. Además, la mayoría de los encuestados manifestó una actitud favorable hacia la integración de tecnologías automatizadas, lo que destaca la importancia y el potencial impacto de este proyecto dentro de la comunidad universitaria.

Ilustración 15. Interpretación de la ocupación de los encuestados

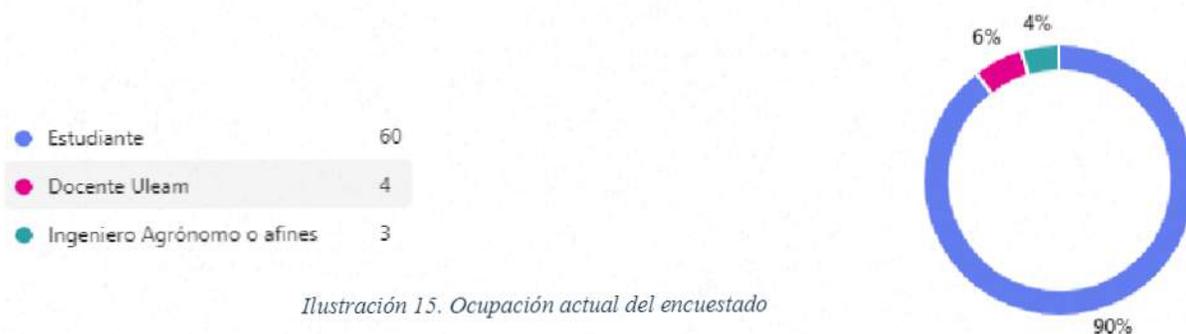


Ilustración 15. Ocupación actual del encuestado

Nota: Imagen sobre la interpretación de la ocupación de los encuestados.

Fuente: Elaboración propia.

Tabla 3: *Tabulación de los datos de la ocupación de los encuestados*

Opciones	Frecuencia	Porcentaje
Estudiantes	60	90%
Docentes Uleam	4	4%
Ingenieros Agrónomo o Afines	3	3%
Total	67	100%

Tabla 3. Tabulación de datos de encuestado primera pregunta

Nota: tabla sobre la tabulación de los datos de la ocupación de los encuestados.

Fuente: Elaboración Propia de los Autores.

2. ¿Cuenta con experiencia previa en la implementación de sistemas automatizados en cualquier sector, ya sea en el ámbito agrícola u otros?

Los resultados obtenidos reflejan que el 42% de los encuestados cuenta con experiencia previa en la implementación de sistemas automatizados, mientras que el 58% indicó no tener experiencia en este ámbito. Este dato evidencia la necesidad de considerar estrategias de capacitación y difusión tecnológica para garantizar una mejor comprensión y adopción del sistema automatizado de riego propuesto en los exteriores del bloque de Informática en la Universidad Laica Eloy Alfaro de Manabí. La participación de quienes poseen experiencia podría ser clave en la ejecución del proyecto.

Ilustración 16. *Interpretación de la experiencia previa con sistemas automatizados*



Ilustración 16. Interpretación de la experiencia de los encuestados

Nota: Imagen sobre la interpretación de la experiencia previa con sistemas automatizados.

Fuente: Elaboración propia de los Autores.

Tabla 4: Tabulación de la interpretación de la experiencia previa con sistemas automatizados

Opciones	Frecuencia	Porcentaje
Si	28	42%
No	39	58%
Total	67	100%

Tabla 4. Tabulación de datos segunda pregunta

Nota: tabla sobre la tabulación de la interpretación de la experiencia previa con sistemas automatizados

Fuente: Elaboración Propia de los Autores.

3. ¿Qué tan importante considera la implementación de sistemas automatizados para el riego en comparación con los métodos tradicionales de riego manual?

Los resultados de la encuesta indican que el 42% de los encuestados considera la implementación de sistemas automatizados para el riego como 'muy importante', mientras que el 45% la califica como 'importante'. Por otro lado, un 13% la percibe como 'moderadamente importante', y ninguna respuesta clasificó esta opción como 'poco importante' o 'nada importante'. Esto refleja una amplia aceptación y valoración positiva hacia la adopción de tecnologías automatizadas en comparación con los métodos tradicionales de riego manual, lo que refuerza la viabilidad del proyecto de implementación de un sistema de riego automatizado en los exteriores del bloque de Informática de la Universidad Laica Eloy Alfaro de Manabí.

Ilustración 17. Interpretación del nivel de interés de la implementación

● Muy importante	28
● Importante	30
● Moderadamente importante	9
● Poco importante	0
● Nada importante	0

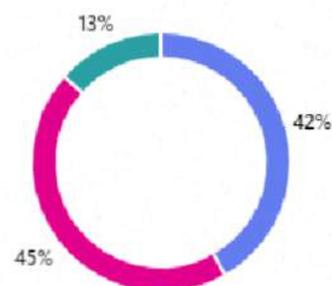


Ilustración 17. Interpretación de nivel de interés

Nota: Imagen sobre la interpretación del nivel de interés de la implementación

Fuente: Elaboración propia de los Autores.

Tabla 5: *Tabulación del nivel de interés en la implementación*

Opciones	Frecuencia	Porcentaje
Muy importante	28	13%
Importante	30	45%
Moderadamente Importante	9	42%
Poco importante	0	0%
Nada importante	0	0%
Total	67	100%

Tabla 5. Tabulación de datos tercera pregunta

Nota: tabla sobre la tabulación del nivel de interés en la implementación.

Fuente: Elaboración Propia de los Autores.

4. ¿Considera que la automatización del riego mediante sensores podría mejorar la eficiencia en el uso del agua en espacios exteriores de la ULEAM?

Los resultados de la encuesta muestran que el 97% de los encuestados considera que la automatización del riego mediante sensores podría mejorar la eficiencia en el uso del agua en los espacios exteriores de la Universidad Laica Eloy Alfaro de Manabí. Solo un 3% manifestó que no lo cree así. Este amplio consenso respalda la implementación del sistema automatizado,

destacando su potencial para optimizar los recursos hídricos y contribuir a una gestión más sostenible del agua.

Ilustración 18. *Percepción sobre los beneficios de la automatización del riego*

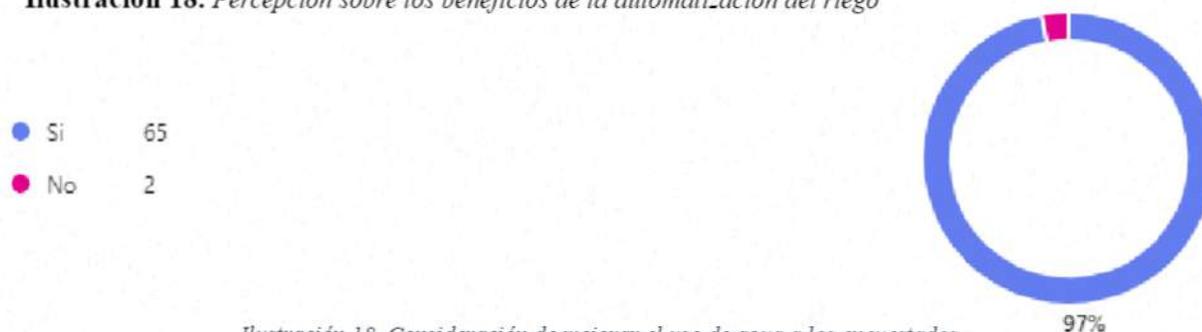


Ilustración 18. Consideración de mejorar el uso de agua a los encuestados

Nota: Imagen de la percepción sobre los beneficios de la automatización del riego

Fuente: Elaboración propia.

Tabla 6: *Tabulación de la percepción sobre los beneficios de la automatización del riego*

Opciones	Frecuencia	Porcentaje
Si	65	97%
No	2	3%
Total	67	100%

Tabla 6. Tabulación de datos cuarta pregunta

Nota: tabla sobre la tabulación de la percepción sobre los beneficios de la automatización del riego.

Fuente: Elaboración Propia de los Autores.

5. ¿Ha tenido alguna experiencia con la integración de tecnologías de comunicación (como redes inalámbricas, LoRa, ¿Wi-Fi) en sistemas automatizados?

Los resultados de la encuesta reflejan que el 59% de los encuestados ha tenido experiencia previa en la integración de tecnologías de comunicación, como redes inalámbricas, LoRa o Wi-Fi, en sistemas automatizados. Por otro lado, el 22% indicó no tener experiencia, mientras que el 19%

no está seguro. Estos datos evidencian que una parte significativa del público objetivo cuenta con conocimientos prácticos en tecnologías asociadas, lo cual podría facilitar la implementación del sistema automatizado de riego propuesto en los exteriores del bloque de Informática en la ULEAM.

Ilustración 19. *Experiencia en la interpretación de Tecnologías de comunicación en sistemas automatizados*



Ilustración 19. Experiencia sobre tecnología de comunicación a los encuestados

Nota: Imagen de experiencia en la interpretación de tecnologías de comunicación en sistemas automatizados.

Fuente: Elaboración propia.

Tabla 7: *Tabulación de la experiencia de tecnologías en sistemas automatizados*

Opciones	Frecuencia	Porcentaje
Si	39	59%
No	15	22%
No esto seguro	13	19%
Total	67	100%

Tabla 7. Tabulación de datos quinta pregunta

Nota: tabla sobre la tabulación de la experiencia de tecnologías en sistemas automatizados.

Fuente: Elaboración Propia de los Autores.

6. ¿Cuál es su nivel de conocimiento sobre los sensores utilizados en sistemas de riego automatizados (como sensores de humedad del suelo, temperatura, etc.)?

Los resultados de la encuesta muestran que el 42% de los encuestados posee un conocimiento básico sobre los sensores utilizados en sistemas de riego automatizados, como sensores de humedad del suelo y temperatura. Un 22% reporta tener un conocimiento moderado, mientras que un 15% indica no tener ningún conocimiento y solo un 21% manifiesta tener un amplio conocimiento. Estos datos destacan la necesidad de ofrecer capacitación y recursos educativos para fortalecer el entendimiento técnico sobre el uso de sensores en el sistema automatizado de riego propuesto en la ULEAM.

Ilustración 20. Nivel de conocimiento sobre sensores en sistemas de riego automatizados



Ilustración 20. Nivel de conocimiento sobre sensores a los encuestados

Nota: Imagen sobre el nivel de conocimiento sobre sensores en sistemas de riego automatizados.

Fuente: Elaboración propia de los Autores.

Tabla 8: Tabulación sobre el nivel de conocimiento sobre sensores en un sistema de riego automatizado

Opciones	Frecuencia	Porcentaje
Amplio conocimiento	14	21%
Conocimiento moderado	15	22%
Conocimiento básico	28	42%
Sin conocimiento	10	15%
Total	67	100%

Tabla 8. Tabulación de datos sexta pregunta

Nota: tabla sobre la tabulación sobre el nivel de conocimiento sobre sensores en un sistema de riego automatizado.

Fuente: Elaboración Propia de los Autores.

7. ¿Qué tan importante considera que es la retroalimentación de los usuarios en la mejora de sistemas de riego automatizados?

Los resultados de la encuesta indican que el 55% de los encuestados considera 'muy importante' la retroalimentación de los usuarios en la mejora de sistemas de riego automatizados, mientras que un 42% la valora como 'moderadamente importante'. Solo un 3% la califica como 'poco importante' y ninguno la considera 'nada importante'. Estos datos destacan la relevancia de involucrar a los usuarios finales en el desarrollo y optimización del sistema, asegurando que este responda a sus necesidades y expectativas.

Ilustración 21. *Importancia de la retroalimentación de usuarios en la mejora de sistemas de riego*



Ilustración 21. Importancia sobre una retroalimentación

Nota: Imagen sobre la importancia de la retroalimentación de usuarios en la mejora de sistemas de riego.

Fuente: Elaboración propia de los Autores.

Tabla 9: *Tabulación de la Importancia de la retroalimentación de usuarios en la mejora de sistemas de riego*

Opciones	Frecuencia	Porcentaje
Muy importante	37	55%
Moderadamente importante	28	42%
Poco importante	2	3%

Nada Importante	0	0%
Total	67	100%

Tabla 9. Tabulación de datos séptima pregunta

Nota: tabla sobre la tabulación de la Importancia de la retroalimentación de usuarios en la mejora de sistemas de riego.

Fuente: Elaboración Propia de los Autores.

8. ¿Cuál es su opinión sobre la aceptación y el apoyo de la comunidad universitaria hacia la implementación de un sistema automatizado de riego y gestión de circuitos eléctricos?

Los resultados de la encuesta reflejan que el 54% de los encuestados tiene una opinión 'muy positiva' respecto a la aceptación y el apoyo de la comunidad universitaria hacia la implementación de un sistema automatizado de riego y gestión de circuitos eléctricos. El 46% restante manifestó una opinión 'positiva', y no se registraron opiniones negativas. Estos datos evidencian un respaldo significativo por parte de la comunidad universitaria, lo que refuerza la viabilidad y el impacto positivo del proyecto dentro de la ULEAM.

Ilustración 22. Opiniones sobre las aceptación y apoyo hacia la implementación del sistema automatizado

● Muy positiva	36
● Positiva	31
● Negativa	0
● Muy negativa	0

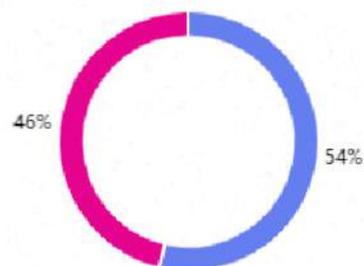


Ilustración 22. Opinión sobre aceptación y apoyo

Nota: Aceptación y Apoyo al Sistema Automatizado.

Fuente: Elaboración propia de los Autores.

Tabla 10: Análisis Tabular de la Aceptación y el Apoyo hacia la Implementación del Sistema Automatizado

Opciones	Frecuencia	Porcentaje
Muy positiva	36	54%
Positiva	31	46%
Negativa	0	0%
Muy negativa	0	0%
Total	67	100%

Tabla 10. Tabulación de datos octava pregunta

Nota: Tabla de la tabulación sobre la aceptación y apoyo hacia la implementación del sistema automatizado.

Fuente: Elaboración Propia de los Autores.

3.7 Análisis y Resultados: Informe Final

Este informe detalla los hallazgos derivados del análisis de datos recopilados en el marco del diseño e implementación de un sistema de riego automatizado basado en sensores, destinado a los exteriores del bloque de Informática de la Universidad Laica Eloy Alfaro de Manabí (ULEAM). El objetivo principal de este proyecto es maximizar la eficiencia en el uso de los recursos hídricos, incorporando tecnologías avanzadas, como sensores de humedad del suelo y temperatura, para promover la sostenibilidad y optimizar la gestión del agua en el entorno universitario.

La investigación se basó en una encuesta estructurada, aplicada a 67 participantes que incluyeron estudiantes, docentes y profesionales del área agrícola, como ingenieros agrónomos. Este instrumento permitió recopilar información clave sobre las necesidades, expectativas y niveles de conocimiento del público objetivo respecto a los sistemas de riego automatizados. Los resultados revelaron un amplio interés por la tecnología propuesta, con el 97% de los encuestados considerando que la automatización del riego podría mejorar significativamente la eficiencia en el uso del agua. Además, se identificaron como beneficios destacados la reducción en el desperdicio de agua, el control preciso de las condiciones ambientales y la facilidad de manejo del sistema.

Entre los encuestados, un 58% demostró tener experiencia previa en el uso de tecnologías relacionadas, como redes inalámbricas (LoRa, Wi-Fi) y sensores de monitoreo, lo que sugiere una base técnica favorable para la implementación del proyecto. Sin embargo, también se evidenció la necesidad de reforzar los conocimientos técnicos, ya que un 42% reportó tener solo un conocimiento básico sobre el funcionamiento de sensores utilizados en sistemas de riego. Esta situación subraya la importancia de incluir programas de capacitación técnica que faciliten la adopción y el uso eficiente del sistema.

Por último, la encuesta también reflejó un respaldo significativo hacia el proyecto, con el 54% de los encuestados calificando su percepción como 'muy positiva' y un 46% como 'positiva'. Estos datos consolidan la viabilidad del sistema y refuerzan su relevancia como una solución innovadora y sostenible para los espacios exteriores del bloque de Informática en la ULEAM.

CAPITULO IV

4. Marco Propositivo

4.1 Introducción

En este capítulo se presenta la propuesta para la solución al problema de riego en los exteriores del bloque de Informáticas de la Universidad Laica Eloy Alfaro de Manabí. El sistema de riego automatizado que se propone tiene como objetivo optimizar el uso del agua, reducir el esfuerzo humano y mejorar la eficiencia operativa a través de la implementación de tecnologías avanzadas. La propuesta está basada en los conocimientos adquiridos a lo largo de la investigación, y responde a las problemáticas planteadas en los capítulos anteriores, tales como el uso excesivo del agua, el esfuerzo físico del personal encargado del riego y la falta de control sobre las condiciones ambientales.

4.2 Descripción de la Propuesta

El sistema de riego automatizado propuesto se basa en la integración de diversos componentes tecnológicos interconectados que permiten automatizar el proceso de riego, adaptándose a las condiciones climáticas y de humedad del suelo en tiempo real. Se instalan sensores de humedad que miden la cantidad de agua en el terreno y, según los datos obtenidos, se activan electroválvulas para liberar la cantidad de agua necesaria, lo que asegura un riego eficiente y adecuado. Además, se emplea una plataforma IoT (Internet de las Cosas) para gestionar y monitorear el sistema de manera remota desde cualquier dispositivo con acceso a internet, brindando la posibilidad de consultar los datos de los sensores y hacer ajustes manuales cuando sea necesario. Esto permite optimizar el control del riego y garantizar una operación precisa y eficiente.

Ilustración 23. *Arquitectura de la solución plantada.*

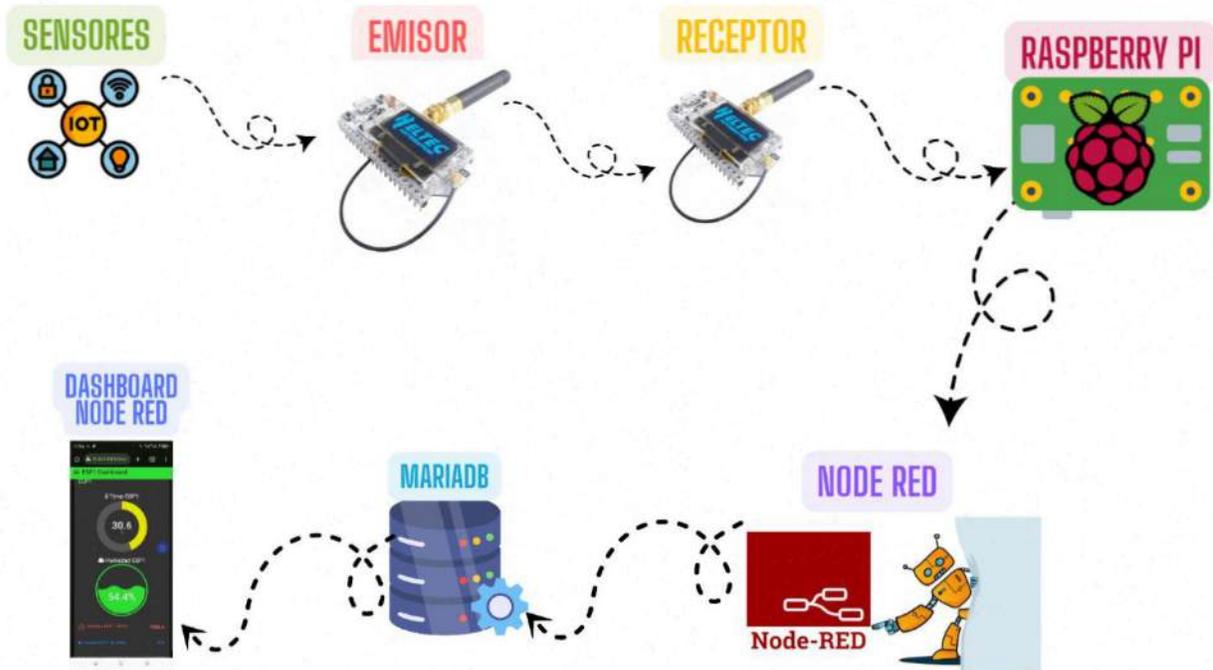


Ilustración 23. Arquitectura de la solución plantada.

Nota: Imagen sobre Arquitectura de la solución plantada.

Fuente: Elaboración propia de los Autores.

El sistema diseñado para el control automatizado de riego se caracteriza por una arquitectura modular y escalable, enfocada en optimizar el uso eficiente del agua en entornos específicos. La solución implementa microcontroladores ESP32 equipados con tecnología LoRa, que funcionan como nodos locales responsables de recopilar información de sensores, tales como la humedad del suelo, el caudal de agua y la temperatura ambiente. Los datos recopilados son transmitidos de manera inalámbrica utilizando LoRa hacia una plataforma centralizada, donde se procesan y supervisan en tiempo real mediante la herramienta Node-RED, garantizando un control efectivo y dinámico del sistema de riego.

La integración de tecnología LoRa permite una comunicación eficiente y de largo alcance entre los nodos y el servidor central, ideal para tecnologías LAN se encuentran en áreas alejadas o dispersas. Los usuarios pueden acceder al sistema desde cualquier dispositivo con conexión a internet, lo que garantiza un monitoreo y control remoto confiable y cómodo.

4.3 Determinación de Recursos

4.3.1 Humanos

En esta fase, se conformó un equipo multidisciplinario encargado de supervisar e implementar las distintas etapas del proyecto. La combinación de experiencia especializada y conocimientos técnicos resultó esencial para llevar a cabo de manera eficiente la implementación de un sistema de riego automatizado basado en sensores, destinado a los exteriores del Bloque de Informática.

Este enfoque interdisciplinario permitió abordar cada etapa del proyecto con precisión, asegurando un sistema funcional y adaptado a las necesidades del entorno.

Tabla 11: Recursos humanos y funciones

RECURSOS	FUNCIÓN
Anchundia Anchundia Jamileth Estela Vera Castro Carol Pamela	Autores sobre el trabajo de titulación sobre el tema ya antes mencionado.
Ingeniero Mero Briones Henry Neurio	Guía de los autores.

Tabla 11. Recursos humanos y funciones

Nota: tabla sobre recursos humanos y funciones.

Fuente: Elaboración propia de los Autores.

4.3.2 Tecnológicos

La siguiente tabla muestra los recursos técnicos utilizados en el desarrollo de este proyecto:

Tabla 12: Recursos utilizados y su función de cada recurso

RECURSOS UTILIZADOS	FUNCIONES DE LOS RECURSOS
---------------------	---------------------------

ESP32 LoRa V2 OLED	Nos ayudó a recolectar información de los distintos parámetros para luego procesarlos y controlar los actuadores
Raspberry pi 4B	Se utilizó como servidor central con Node-RED y Mosquitto MQTT, recibiendo datos del ESP32 LoRa receptor, procesándolos y permitiendo acceso remoto para monitoreo.
Sonda sensor de temperatura DS18B20	Se utilizó para medir la temperatura del riego, proporcionando datos precisos y resistentes a condiciones húmedas gracias a su encapsulado impermeable.
Módulo BME680	Se utilizó para medir la humedad, además de otros parámetros ambientales, proporcionando datos precisos para optimizar el riego y las condiciones del entorno.
Orbit 57280 válvula de rociador 3/4	Se utilizó en el sistema de riego para abrir y cerrar automáticamente el flujo de agua, asegurando un control eficiente y preciso del riego según las necesidades del terreno.
Sensor de flujo de agua 3/4	Se utilizó para medir el caudal del riego, permitiendo registrar el volumen de agua utilizado y optimizar su distribución en el sistema.
Reloj digital	Se utilizó para medir el tiempo de riego y programar horarios específicos, garantizando un riego eficiente y automatizado según las necesidades del sistema.
Transformador 110 a 24V	Se utilizó para suministrar electricidad al cajetín de control, proporcionando la energía necesaria para operar componentes del sistema de riego.

Arduino	Utilización para desarrollar y cargar los códigos del ESP32, configurando tanto el emisor como el receptor para la comunicación y control del sistema de riego automatizado.
Node-Red	Se utilizó para crear el dashboard del sistema, permitiendo visualizar y controlar los datos de riego de manera interactiva y en tiempo real.
MariaDB	Se utilizó como base de datos para almacenar los datos enviados por el ESP32, permitiendo su gestión y análisis en el sistema de riego automatizado.

Tabla 12. Recursos tecnológicos

Nota: tabla sobre recursos tecnológicos.

Fuente: Elaboración propia de los Autores.

4.3.3 Económicos

En la siguiente tabla se pueden observar descripción, detalles de los costos y materiales utilizados para ejecutar este proyecto:

Tabla 13: Recursos humanos y funciones

#	Descripcion	Costo
1	Tecnologicos	372.01
2	Humanos	100
3	Materiales para instalacion	298.83
4	Riel Din	1.36

5	Canaleta 40x40	12.67
6	Breaker para Riel Din	12.94
7	Tablero de metal 60x40x20	57.50
8	Tablero Plastico 20x15x10	21.75
9	Prensa estopa	1.76
10	Cable Flexible 14	7.71
11	Cable Flexible 16	5.41
12	Tomacorritente sobrepuesto	5.06
13	Cinta Aislante	3.83
14	borneras	4.60
15	espadines hembra	4.60
16	Placa perforada 10x10	4.60
17	Broca y tornillos	5.06
18	Pinturas	10.00
19	Sistema de riego	140.00

20	Utilitarios	500.00
21	Resmas de Papel	20.00
22	Impresiones copias	100.00
23	Anillados y Empastados	80.00
24	viaticos	300.00
25	SubTotal (1+2+3+20)	1270.85
26	Gastos adicionales	70.00
27	Total (25+26)	1340.85

Tabla 13. Tabla sobre los económico

Nota: tabla sobre lo económico.

Fuente: Elaboración propia de los Autores.

4.4 Etapas operativas para el desarrollo de la propuesta

4.4.1 Enfoques metodológicos empleados en el desarrollo de la propuesta

La supervisión automatizada de los parámetros en el sistema de riego se desarrolla siguiendo la metodología de cascada, que constituye un enfoque clave para garantizar un proceso organizado y eficiente. Esta metodología se distingue por su planificación minuciosa desde el inicio del proyecto, con la definición precisa de objetivos, requisitos, alcance y entregables. Este enfoque reduce la posibilidad de cambios inesperados durante el desarrollo, favoreciendo un flujo de trabajo estructurado.

El enfoque de cascada divide el proyecto en una serie de fases claramente definidas, donde cada etapa tiene metas específicas y produce resultados concretos. Esta organización secuencial

facilita la gestión del proyecto, al permitir un seguimiento transparente del progreso y una asignación clara de responsabilidades dentro del equipo.

Una de las fortalezas de esta metodología es la incorporación de actividades de control de calidad en cada fase del proyecto. Esto asegura que cada entregable cumpla con los requisitos establecidos, reduciendo el riesgo de errores y garantizando la funcionalidad óptima del sistema. Gracias a este enfoque, se garantiza la calidad general del sistema, previniendo problemas futuros y mejorando su confiabilidad y rendimiento.

La implementación de esta metodología en el desarrollo del sistema de riego automatizado se llevó a cabo de la siguiente manera:

4.4.1.1 Fase Inicio

En la etapa inicial del proyecto, se realizó un análisis exhaustivo de las necesidades específicas del sistema de riego automatizado mediante encuestas y recolección de datos. Estos resultados se utilizaron para definir la visión del proyecto y establecer claramente las funcionalidades clave que debía cumplir el sistema. Entre los aspectos considerados se incluyó la recolección de datos de sensores como el DS18B20 para temperatura, el BME680 para humedad y el sensor de flujo de agua para caudal, así como el diseño de una interfaz de usuario intuitiva para la gestión del riego.

Se definieron los requisitos técnicos del sistema, seleccionando un ESP32 LoRa como emisor de datos hacia un servidor central basado en Raspberry Pi. También se determinaron los elementos de hardware esenciales, como la válvula Orbit 57280 para el control del flujo de agua, el reloj digital para la programación de horarios de riego, y un transformador de 110V a 24V para la alimentación eléctrica del sistema. En cuanto al software, se eligió Node-RED para el diseño del dashboard, MariaDB para la gestión de datos y protocolos de comunicación como MQTT para garantizar la conectividad entre los dispositivos. Todo esto se desarrolló considerando las características específicas de la estructura hidropónica y las necesidades de riego del entorno objetivo.

4.4.1.2 Fase de planificación y estimación

Con base en la información recopilada en el contexto de la automatización del monitoreo como herramienta diagnóstica, se elaboró un diseño detallado para el sistema de riego automatizado. Este diseño incluye un esquema a escala que especifica las dimensiones de los componentes, la disposición estratégica de válvulas y sensores, el recorrido del flujo de agua y los espacios necesarios para su correcta implementación. El diseño también incorporó la integración de componentes clave, como la válvula Orbit 57280, los sensores DS18B20, BME680 y de flujo de agua, junto con un reloj digital para la programación de los horarios de riego, asegurando un sistema eficiente y funcional

Además, se diseñó e integró una base de datos en MariaDB con el sistema web, gestionado desde un servidor Raspberry Pi, lo que permitió almacenar y acceder a datos históricos y en tiempo real. Esto proporcionó a los usuarios la capacidad de visualizar gráficos interactivos y monitorear el funcionamiento del sistema de riego de manera remota y eficiente.

Materiales físicos y componentes electrónicos:

- ESP32 LoRa: Para la comunicación inalámbrica de largo alcance entre los sensores y el sistema central.
- Raspberry Pi: Utilizada como servidor para gestionar la base de datos y el sistema web.
- Sensores:
 - DS18B20: Para medición de temperatura.
 - BME680: Para monitoreo de calidad del aire.
 - Sensor de flujo de agua: Para medir la cantidad de agua utilizada.
- Válvulas y bombas de agua: Para controlar el flujo del riego automatizado.
- Fuente de alimentación: Para garantizar el funcionamiento continuo de los componentes electrónicos.
- Cables, conectores y elementos de instalación: Para la correcta interconexión de los componentes.

Herramientas de desarrollo y software:

- Node-RED: Plataforma de desarrollo visual para automatizar flujos de trabajo.
- Librerías para sensores: Incluyendo las necesarias para la programación del ESP32 Arduino.
- Base de datos MariaDB: Para almacenar datos históricos y en tiempo real.
- Algoritmos de control automatizado: Para gestionar los horarios y parámetros del riego.

4.4.1.3 Fase de implementación

En esta etapa, se realizó la integración física y lógica de los componentes del sistema de riego automatizado. Las actividades principales desarrolladas fueron las siguientes:

Montaje del hardware:

- Se procedió con la instalación de los sensores de humedad (BME680), temperatura (DS18B20) y flujo de agua, ubicándolos estratégicamente según lo definido en la fase de planificación.
- El controlador principal (ESP32 LoRa) fue configurado para enviar los datos capturados al servidor Raspberry Pi mediante una red local (LAN) basada en conectividad Ethernet.
- Las electroválvulas (Orbit 57280) se integraron al sistema de riego, asegurando una instalación robusta, eficiente y completamente operativa.

Desarrollo del software:

- Mediante la plataforma Node-RED, se diseñó e implementó un dashboard interactivo que permite a los usuarios monitorear en tiempo real las variables de temperatura, humedad y caudal. Además, incluye funcionalidades para el control manual de las válvulas.
- Se configuró la base de datos MariaDB para el almacenamiento de datos históricos y en tiempo real, habilitando así el análisis de patrones y facilitando la toma de decisiones informadas.

- El acceso tanto al dashboard como a la base de datos se restringió exclusivamente a dispositivos conectados a la red LAN, garantizando un entorno seguro y controlado.

Configuración de la conectividad:

- Se implementaron protocolos de comunicación basados en MQTT, asegurando una transmisión de datos confiable entre los sensores, el controlador ESP32 LoRa y el servidor central.
- La red LAN, que opera únicamente con conexión Ethernet, fue optimizada para garantizar estabilidad y fiabilidad en la transmisión de datos a lo largo de toda la zona de riego.

Pruebas iniciales:

- Se llevaron a cabo pruebas funcionales en cada componente del sistema, verificando la correcta interacción entre sensores, válvulas y el dashboard.
- Estas pruebas incluyeron la medición de parámetros ambientales, la apertura y cierre automático de válvulas, y la visualización de datos en tiempo real desde dispositivos conectados a la red LAN.

4.4.1.4 Fase de verificación

En esta fase, se validó exhaustivamente la funcionalidad del sistema para garantizar el cumplimiento de los requisitos establecidos en las etapas iniciales. Las actividades desarrolladas fueron las siguientes:

Pruebas de los sensores:

- Se verificó la precisión de los datos generados por los sensores DS18B20 y BME680, comparando sus lecturas con dispositivos de medición estándar para asegurar su exactitud.
- Se evaluó el tiempo de respuesta de los sensores, confirmando la captura ágil y confiable de los datos.

Pruebas del control del flujo de agua:

- Se comprobó el funcionamiento adecuado de las válvulas Orbit 57280, verificando su apertura y cierre en respuesta a los datos proporcionados por los sensores de humedad y temperatura.
- Se midió la cantidad de agua liberada por las válvulas para garantizar un riego eficiente, minimizando el desperdicio.

Pruebas de conectividad:

- Se realizaron pruebas de transmisión de datos entre los sensores, el ESP32 LoRa y el servidor Raspberry Pi a través de la red LAN con conexión Ethernet, asegurando la integridad y ausencia de pérdidas de información.
- Se verificó la cobertura y estabilidad de la red en toda la zona de riego, identificando y documentando posibles mejoras para optimizar la conectividad.

Evaluación del software:

- Se revisó el dashboard diseñado en Node-RED, confirmando la correcta visualización de datos en tiempo real y el funcionamiento adecuado de las opciones de control manual.
- Se evaluaron las consultas realizadas a la base de datos MariaDB, garantizando que los datos históricos se almacenaran y recuperaran con precisión desde dispositivos dentro de la red LAN.

4.4.1.5 Despliegue

El objetivo principal del despliegue es garantizar que el sistema esté completamente operativo y cumpla con los requisitos definidos en el proyecto. Además, asegura que los usuarios tengan las herramientas necesarias para interactuar con el sistema de manera eficiente y que el entorno esté preparado para soportar su uso continuo.

4.4.2 Aplicación de la metodología

Tabla 14. Equipo de Proyecto

Nombre de proyecto:	Implementación de un sistema de riego automatizado a través de sensores en los exteriores del bloque de informática en la Universidad Laica Eloy Alfaro de Manabí
----------------------------	--

Roles	
--------------	--

Desarrolladores	Anchundia Anchundia Jamileth Estela Vera Castro Carol Pamela
------------------------	---

Tutor del trabajo de titulación	Ingeniero Mero Briones Henry Neurio
--	-------------------------------------

Tabla 14. Proyecto en equipo sobre la aplicación de la metodología

Nota: Tabla referente al proyecto en equipo sobre la aplicación de la metodología

Fuente: Elaboración propia de los autores

4.4.2.1 Fase preliminar del proyecto

Tabla 15. Descripción del propósito de la visión del proyecto

Definición de la visión del proyecto

Este proyecto busca el desarrollo de un sistema de riego automatizado que integre tecnologías avanzadas para optimizar el cultivo en espacios específicos. Con un enfoque en la eficiencia y sostenibilidad, el sistema permitirá monitorear y controlar parámetros clave como temperatura, humedad y caudal de agua, asegurando condiciones ideales de crecimiento y minimizando riesgos asociados al uso ineficiente de recursos.

Visión del proyecto: Ofrecer un sistema de riego automatizado mediante ESP32-LoRa y un servidor central con Raspberry Pi, que permita la visualización en tiempo real de los datos a través de Node-RED. Este sistema facilitará el monitoreo eficiente del riego, presentando información gráfica y brindando herramientas para optimizar el uso de recursos en cultivos específicos.

Tabla 15. Descripción de la visión del proyecto

Nota: tabla sobre la descripción de la visión del proyecto.

Fuente: Elaboración propia de los Autores.

Tabla 16. Épicas y Fases: Conceptos Clave en Scrum

FASES	
Fase 1:	Diseño de las funcionalidades principales.
Fase 2:	Desarrollo de la infraestructura hidropónica.
Fase 3:	Automatización con ESP32-LORA y sensores.
Fase 4:	Integración de un sistema web

Tabla 16. Definición de épicas o fases según Scrum

Nota: tabla referente a la definición de épicas o fases según Scrum

Fuente: Elaboración propia de autores

Tabla 17. Definición de los criterios de finalización

Criterios de finalización	
Criterio de finalización 1:	La característica debe ejecutarse completamente y estar operativa, asegurando el cumplimiento de los requisitos establecidos. .
Criterio de finalización 2:	El modelo y la página web deben ser evaluados exhaustivamente, asegurando que cumplen con los objetivos establecidos y funcionan de manera confiable.
Criterio de finalización 3:	La ejecución de las pruebas de aceptación debe ser exitosamente validada antes de dar por finalizado el sprint.
Criterio de finalización 4:	El script debe cumplir con estándares de calidad, ser claro y comprensible, además de alinearse con las buenas prácticas de programación. .
Criterio de finalización 5:	El manual debe mantenerse actualizado y abarcar todos los aspectos relevantes del proyecto.
Criterio de finalización 6:	Cada iteración debe integrarse correctamente con el sistema general y ser desplegada sin inconvenientes en el entorno de producción. .

Tabla 17. Definición de los criterios de finalización

Nota: tabla sobre la definición de los criterios de finalización

Fuente: Elaboración propia.

4.4.2.2 Fase de planificación y estimación

Tabla 18. Desglose de épicas y sus tareas

Épica	Tareas
Definición de funcionalidades	<ul style="list-style-type: none">• Detallar las épicas y descomponerlas en historias de usuario claras y específicas.• Establecer los criterios de aceptación para las historias de usuario.

Desarrollo de la infraestructura del sistema	<ul style="list-style-type: none"> • Crear un esquema de la estructura del sistema basado en los requerimientos y necesidades específicas del proyecto. • Especificar los materiales y componentes necesarios para la construcción del sistema. • Construir la estructura del sistema de riego.
Implementación de automatización mediante ESP32-LORA y sensores	<ul style="list-style-type: none"> • Integrar el módulo ESP32-LORA en el sistema. • Ajustar y calibrar los sensores para la medición precisa de los parámetros ambientales. • Desarrollar e implementar algoritmos de control para regular los parámetros ambientales de temperatura, humedad y pH. • Ejecutar pruebas y realizar ajustes en la automatización básica para asegurar su funcionamiento correcto. • Conexión a la base de datos
Incorporación del sistema web	<ul style="list-style-type: none"> • Integrar el sistema web con la base de datos para permitir la visualización en tiempo real de los datos recopilados • Desarrollar e implementar funcionalidades de usuario que permitan la gestión eficiente del sistema.

Tabla 18. Desglose de épicas y sus tareas

Nota: tabla sobre el desglose de épicas y sus tareas.

Fuente: Elaboración propia.

4.4.2.3 Definición de funcionalidades

Las historias de usuario se aplicarán únicamente a las épicas 3 y 4, mientras que para los sprints 1 y 2 no será necesario su uso.

Tabla 19. Historia de Usuario 1

Código: CA1	Nombre de historia: Diseñar el esquema lógico del circuito
Usuario: Desarrolladores	
Prioridad de negocio: Alta prioridad	Riesgo en desarrollo: Medio de riesgo
Iteración asignada: 3	Desarrolladores: Anchundia Jamileth- Vera Carol

Descripción: El desarrollador se encargará de diseñar el circuito lógico necesario para garantizar la comunicación efectiva entre el microcontrolador ESP32 LoRa y los diferentes sensores del sistema de riego automatizado, tales como el sensor de temperatura DS18B20, el sensor ambiental BME680 y el medidor de flujo de agua, asegurando su correcta integración para el monitoreo y control eficiente del riego.

Observaciones: Este paso es fundamental, ya que sin la correcta configuración e integración del circuito lógico entre el ESP32 LoRa y los sensores, no sería posible realizar las actividades subsiguientes relacionadas con el monitoreo y control del sistema de riego automatizado.

Tabla 19. Historia de usuario1

Nota: Tabla sobre la historia de usuario1

Fuente: Elaboración propia.

Tabla 20. Criterios de aceptación y casos límites CAI

Criterios para aceptar

Criterios para Aceptación 1: El diseño lógico del circuito debe ser representado en un diagrama que ilustre de manera clara y precisa los componentes involucrados, tales como el microcontrolador ESP32 con LoRa, los sensores (DS18B20, BME680, sensor de flujo de agua), y las conexiones necesarias para el monitoreo del sistema de riego. Este diagrama debe detallar cómo se interconectan los componentes y cómo se gestionan los datos para garantizar un funcionamiento eficiente del sistema.

Criterio de Aceptación 2: El diagrama debe ser claro, conciso y fácil de interpretar para garantizar su correcta implementación en el sistema de riego.

Criterio de Aceptación 3: El diseño lógico del circuito debe ajustarse a las especificaciones técnicas del proyecto, garantizando una comunicación eficiente entre los componentes y el servidor Raspberry Pi, así como con la plataforma de Node-RED. Esto asegurará que los datos recopilados por los sensores sean transmitidos correctamente, permitiendo el monitoreo y control del sistema de riego en tiempo real de manera óptima

Caso de límite caso de restricción caso de restricción

Caso Límite 1: Si no existe conexión entre el ESP32 LoRa y los sensores, el sistema de riego no podrá monitorear los parámetros, afectando su funcionamiento.

Caso Límite 2: Si uno de los sensores presenta fallas o mediciones imprecisas, los datos registrados en Node-RED pueden ser incorrectos, afectando la toma de decisiones para el riego.

Caso Límite 3: Si no se realiza un mantenimiento adecuado al sistema, como la limpieza de sensores o válvulas, el flujo de agua puede obstruirse, impactando la capacidad y eficiencia del riego automatizado.

Tabla 20. Criterios de aceptación y casos límites CA1

Nota: tabla de Criterios de aceptación y casos límites CA1

Fuente: Elaboración propia.

Tabla 21. Historia de usuario 2

Código: CA2	Nombre de historia: Configurar WiFi del módulo ESP32 LoRa
Usuario: Desarrolladores	
Prioridad de negocio: Muy Alto	Riesgo en desarrollo: Medio de riesgo
Iteración asignada: 3	Desarrollador: Anchundia Jamileth- Vera Carol
Descripción: El desarrollador configurará la conexión WiFi del módulo ESP32 LoRa utilizando la biblioteca ESP32 Arduino, permitiendo que el sistema de riego automatizado se conecte a la red y envíe los datos recopilados por los sensores al servidor central implementado en el Raspberry Pi. Esto garantizará la comunicación eficiente con Node-RED para el monitoreo en tiempo real y la visualización gráfica de los parámetros.	Observaciones: Es necesario contar con las librerías requeridas para la configuración del módulo y garantizar la estabilidad de la conexión a la red.

Tabla 21. Historia de usuario 2

Nota: tabla sobre la historia de usuario2.

Fuente: Elaboración propia.

Tabla 22. Criterios de aceptación y casos límites CA2

Criterios de Aceptación
Criterio de Aceptación 1: El sistema debe garantizar la validación de la configuración de WiFi introducida para el módulo ESP32 LoRa, asegurando que los datos generados por el sistema de riego sean enviados correctamente al

servidor alojado en el Raspberry Pi. Esto permitirá una comunicación adecuada y el correcto funcionamiento del sistema dentro de su infraestructura.

Criterio de Aceptación: El sistema debe proporcionar al usuario, mediante Node-RED, una indicación clara del estado actual de la conexión WiFi del módulo ESP32 LoRa, permitiendo un monitoreo en tiempo real de la conectividad.

Criterio de Aceptación: El sistema debe ofrecer una guía clara y precisa sobre la configuración de la conexión WiFi del módulo ESP32 LoRa, asegurando una implementación eficiente.

Casos Límite.

Caso Límite: En caso de que el módulo ESP32 LoRa pierda su conexión WiFi, el sistema deberá contar con un mecanismo de reconexión automática, con el fin de minimizar las interrupciones y garantizar que el monitoreo del sistema de riego continúe sin problemas.

Caso Límite: Si se ingresa una contraseña incorrecta para la red WiFi, el módulo ESP32 LoRa debe notificar el fallo, mostrando un mensaje de error que permita corregir la configuración.

Caso Límite: Si los parámetros de la red WiFi cambian, el sistema debe permitir al usuario actualizar la configuración manualmente a través de una interfaz accesible, como Node-RED.

Tabla 22. Criterios de aceptación y casos límites CA2

Nota: tabla de Criterios de aceptación y casos límites CA2.

Fuente: Elaboración propia.

Tabla 23. Historia de usuario 3

Código: CA3

Nombre de historia: Configurar sensores del módulo ESP32 LoRa

Usuario: Desarrollador

Prioridad de negocio: Alta

Riesgo en desarrollo: Media

Iteración asignada:3

Desarrollador: Anhcundia Jamieth- Vera Carol

Descripción: El desarrollador podrá configurar y calibrar los sensores del sistema de riego automatizado, como el DS18B20 para temperatura, el BME680 para humedad y el sensor de flujo de agua,

Observaciones: Es necesario contar con las librerías adecuadas para los sensores seleccionados y verificar que cada uno

utilizando las bibliotecas específicas de cada sensor. Esto permitirá obtener lecturas precisas de los parámetros ambientales y de riego, esenciales para el monitoreo y control automatizado del sistema. Además, se implementarán algoritmos de control que aseguren la confiabilidad de las mediciones.	proporcione datos precisos antes de integrarlos con Node-RED y el sistema web.
--	--

Tabla 23. Historia de usuario 3

Nota: tabla sobre la historia de usuario3.

Fuente: Elaboración propia.

Tabla 24. Criterios de aceptación y casos límites CA3

Criterios de Aceptación
Criterio de Aceptación: Dentro de la configuración, deben utilizarse las librerías adecuadas para cada sensor (DS18B20, BME680, sensor de flujo de agua), garantizando su correcta conexión y funcionamiento con el circuito del ESP32 LoRa.
Criterio de Aceptación: El sistema debe proporcionar, en tiempo real, datos claros y precisos de los sensores del sistema de riego automatizado, como temperatura, humedad y flujo de agua, que serán visualizados a través de la interfaz de Node-RED para su monitoreo y análisis continuo.
Criterio de Aceptación: La información recopilada por los sensores debe actualizarse según el intervalo de tiempo establecido para garantizar datos confiables y actualizados.
Casos Límite.
Caso Límite: Verificación del correcto funcionamiento del ESP32 LoRa para evitar fallos de conexión que interrumpan la transmisión de datos hacia el servidor.
Caso Límite: Gestión adecuada del tráfico de datos recibidos por el ESP32 LoRa para prevenir sobrecargas y garantizar un rendimiento óptimo del sistema.
Caso Límite: Validación continua de los datos obtenidos por los sensores para asegurar su consistencia y evitar errores en la interpretación o visualización en Node-RED.

Tabla 24. Criterios de aceptación y casos límites CA3

Nota: tabla de Criterios de aceptación y casos límites CA3.

Fuente: Elaboración propia.

Tabla 25. Historia de Usuario 4

Código: CA4	Nombre de historia: Crear base de datos para el sistema de riego
Usuario: Desarrollador	
Prioridad de negocio: Alta	Riesgo en desarrollo: Media
Iteración asignada: 4	Desarrollador: Anchundia Jamileth-Vera Carol
<p>Descripción: El desarrollador deberá crear y configurar una base de datos en MariaDB para el almacenamiento eficiente de la información recolectada por los sensores, como temperatura, humedad y flujo de agua. Esta base de datos se integrará con el sistema central en el Raspberry Pi y estará conectada a Node-RED para permitir el monitoreo en tiempo real y la consulta de datos históricos.</p>	<p>Observaciones: Es esencial verificar que las lecturas de los sensores se almacenen correctamente en la base de datos y que estén disponibles para su visualización y análisis a través de la interfaz en Node-RED.</p>

Tabla 25. Historia de usuario 4

Nota: tabla sobre la historia de usuario4.

Fuente: Elaboración propia.

Tabla 26. Criterios de aceptación y casos límites CA4

Criterios de Aceptación
<p>Criterio de Aceptación: El sistema de almacenamiento debe ser capaz de registrar diversos tipos de datos recolectados por los sensores del sistema de riego automatizado, tales como temperatura, humedad y caudal de agua, para su posterior análisis y gestión.</p>
<p>Criterio de Aceptación: Los datos deben ser almacenados de manera estructurada y organizada en la base de datos para facilitar su consulta y análisis desde Node-RED.</p>
<p>Criterio de Aceptación: El sistema debe permitir que los datos almacenados en la base de datos estén disponibles para su monitoreo en tiempo real y para el análisis histórico a través de la interfaz gráfica de Node-RED.</p>
Casos Límite.

Caso Límite: Si el usuario no tiene acceso a la red local o a internet, no podrá monitorear el sistema en tiempo real a través de Node-RED.

Caso Límite: Si ocurre un fallo en el hardware o software del ESP32 LoRa o del servidor Raspberry Pi, el sistema de riego no podrá recopilar ni almacenar datos correctamente.

Caso Límite: En caso de que los datos guardados en el repositorio de información se corrompan, es posible que la información de los sensores no pueda ser recuperada, afectando el análisis y monitoreo del sistema.

Tabla 26. Criterios de aceptación y casos límites CA4

Nota: tabla de Criterios de aceptación y casos límites CA4.

Fuente: Elaboración propia.

Tabla 27. Historia de usuario 5

Código: CA5	Nombre de historia: Registrar datos en el repositorio de información.
Usuario: Desarrollador	
Prioridad de negocio: Alta	Riesgo en desarrollo: Media
Iteración asignada: 3	Desarrollador: Anchundía Jamileth- Vera Carol
Descripción: Además de recolectar datos de los sensores conectados al ESP32 LoRa (como temperatura, humedad y caudal), estos deberán ser almacenados en una base de datos en MariaDB. Esto permitirá rastrear tendencias y analizar el rendimiento del sistema de riego automatizado, facilitando la toma de decisiones basada en los datos históricos y en tiempo real.	Observaciones: Es fundamental verificar que los datos se estén almacenando correctamente en la base de datos y realizar la creación e integración de esta para asegurar su posterior enlace con Node-RED para la visualización de los parámetros monitoreados.

Tabla 27. Historia de usuario 5

Nota: tabla sobre la historia de usuario5.

Fuente: Elaboración propia.

Tabla 28. Criterios de aceptación y casos límites CA5

Criterios de Aceptación

Criterio de Aceptación: Es necesario almacenar de manera segura y confiable los datos recopilados por los sensores (temperatura, humedad y caudal de agua) en una base de datos MariaDB, asegurando su integridad y disponibilidad para su análisis y monitoreo posterior.

Criterio de Aceptación: La información debe ser almacenados en un formato estructurado y organizado, permitiendo su fácil acceso y análisis para el monitoreo y la toma de decisiones en el sistema de riego automatizado.

Criterio de Aceptación: El sistema debe permitir el acceso a los datos almacenados en la base de datos a través de la interfaz gráfica de Node-RED, asegurando que los usuarios puedan consultar tanto información en tiempo real como registros históricos.

Casos Límite.

Caso Límite: Verificar la conexión entre el módulo ESP32 LoRa y la base de datos en el servidor Raspberry Pi para garantizar la transferencia correcta de datos.

Caso Límite: Validar la consistencia entre los datos enviados por el ESP32 LoRa y los almacenados en la base de datos para evitar discrepancias o errores de registro.

Caso Límite: Implementar mecanismos para evitar la sobrecarga de la base de datos, gestionando eficientemente el almacenamiento de grandes volúmenes de datos provenientes de los sensores.

Tabla 28. Criterios de aceptación y casos límites CA5

Nota: tabla de Criterios de aceptación y casos límites CA5.

Fuente: Elaboración propia.

Tabla 29. Historia de usuario 6

Código: CA6

Nombre de historia: Ingreso al sistema de riego automatizado

Usuario: Cada uno de nosotros

Prioridad de negocio: Alta

Riesgo en desarrollo: Media

Iteración asignada:4

Desarrollador: Anchundia Jamileth- Vera Carol

Descripción: Para acceder al sistema de riego automatizado, los usuarios deben iniciar sesión utilizando un nombre de usuario y una contraseña previamente asignados. Este proceso garantiza que los usuarios autorizados puedan acceder a las funcionalidades del sistema, como la visualización de datos en Node-RED y el monitoreo en tiempo real de los parámetros del riego.

Observaciones: Los usuarios no pueden registrarse por sí mismos; el acceso se gestiona exclusivamente mediante la asignación previa de un nombre de usuario y una contraseña. Es fundamental garantizar que este proceso de autenticación sea seguro y confiable.

Tabla 29. Historia de usuario 6

Nota: tabla sobre la historia de usuario6.

Fuente: Elaboración propia.

Tabla 30. Criterios de aceptación y casos límites CA6

Criterios de Aceptación

Criterio de Aceptación: El sistema deberá validar que las credenciales proporcionadas (usuario y contraseña asignados) sean correctas para garantizar el acceso al monitoreo y control del sistema de riego automatizado.

Criterio de Aceptación: Una vez ingresadas las credenciales correctas, el sistema permitirá al usuario acceder a las funcionalidades de monitoreo a través de Node-RED.

Criterio de Aceptación: El sistema deberá garantizar la seguridad en el acceso, evitando que usuarios no autorizados puedan interactuar con los datos o el control del riego.

Casos Límite.

Caso Límite: Si el usuario proporciona credenciales incorrectas (usuario o contraseña), el sistema denegará el acceso y no permitirá la visualización ni el monitoreo del riego.

Caso Límite: Si no se proporcionan las credenciales asignadas, no será posible acceder al sistema ni visualizar los datos de los sensores.

Caso Límite: Si las credenciales asignadas son comprometidas, se debe prever la capacidad de actualizar la contraseña para garantizar la seguridad del sistema.

Tabla 30. Criterios de aceptación y casos límites CA6

Nota: tabla de Criterios de aceptación y casos límites CA6.

Fuente: Elaboración propia.

Tabla 31. Historia de usuario 7

Código: CA7	Nombre de historia: Visualizar de datos o parámetros en el sistema
Usuario: Todos	
Prioridad de negocio: Alta	Riesgo en desarrollo: Media
Iteración asignada: 4	Desarrollador: Anchundia Jamileth-Vera Carol
Descripción: El usuario podrá visualizar en tiempo real los parámetros monitoreados por el sistema de riego automatizado a través de Node-RED, incluyendo datos de temperatura, humedad y caudal de agua. Esto permite un control constante y eficiente del sistema.	Observaciones: Solo los usuarios con acceso mediante credenciales asignadas podrán visualizar esta información. Es necesario garantizar que la interfaz de Node-RED sea clara y accesible para todos los usuarios autorizados.

Tabla 31. Historia de usuario 7

Nota: tabla sobre la historia de usuario 7.

Fuente: Elaboración propia.

Tabla 32. Criterios de aceptación y casos límites CA7

Criterios de Aceptación

Criterio de Aceptación: Node-RED debe mostrar la información de los sensores en tiempo real, incluyendo la temperatura, la humedad y el flujo de agua.

Criterio de Aceptación: Es necesario que los datos se actualicen de forma automática en periodos establecidos para representar las condiciones presentes del sistema de riego automatizado.

Criterio de Aceptación: La interfaz debe ofrecer al usuario la opción de seleccionar los parámetros que desea visualizar, presentando los datos en formatos gráficos y tabulares para facilitar su comprensión y análisis. Esto permitirá una visualización clara y eficiente del rendimiento del sistema de riego automatizado.

Casos Limite.

Caso Limite: Si existe un error en el diseño o la implementación de la interfaz en Node-RED, los gráficos y tablas podrían no mostrarse correctamente o presentar datos desorganizados.

Caso Limite: Si hay problemas de comunicación entre el ESP32 LoRa y el servidor Raspberry Pi, los datos de los sensores podrían no reflejarse en la interfaz, afectando el monitoreo en tiempo real.

Caso Limite: Si se presentan condiciones extremas (como fallos en los sensores o condiciones ambientales fuera del rango de operación), el sistema podría no registrar datos precisos ni reflejar correctamente los parámetros deseados.

Tabla 32. Criterios de aceptación y casos limites CA7

Nota: tabla de Criterios de aceptación y casos limites CA7.

Fuente: Elaboración propia.

4.4.3 Fase de implementación

4.4.3.1 Control del administrador

El riego automatizado ofrece alternativas eficientes en donde se verifica el total de litros, caudal y la presión por aquello este punto se creó para que el administrador pueda ingresar y si desea poder realizar cambios nuevos para que se visualizan en el Dashboard, a continuación, se detallan los pasos para ingresar:

1. Descargamos un software llamado VNS, ya descargado ejecutamos la app e ingresamos la ip designada en este caso tenemos la **10.200.200.28** (no olvidar de estar con red ethernet).

Ilustración 24. Inicio del VNC.

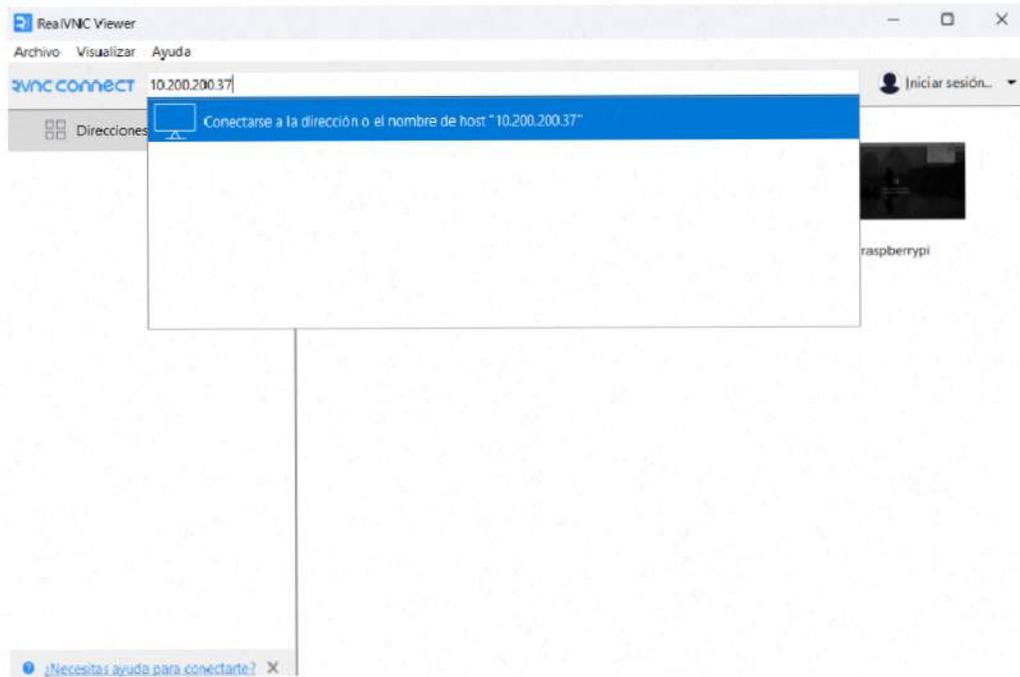


Ilustración 24. Inicio del VNC

Nota: imagen sobre el inicio del VNC

Fuente: Elaboración propia.

2. Al darle enter con la ip que se colocó, nos aparece lo siguiente y lo que se realiza es dar continuar

Ilustración 25. Comprobación de identidad.

Nota: imagen de comprobación de identidad.

Fuente: Elaboración propia.

3. Al darle aceptar nos parece la siguiente pestaña, en este caso ingresamos el nombre de usuario: **tesis** y la contraseña: FCVT_2024

Ilustración 25. Comprobación de identidad.

Ilustración 26. Autenticación en VNC server:

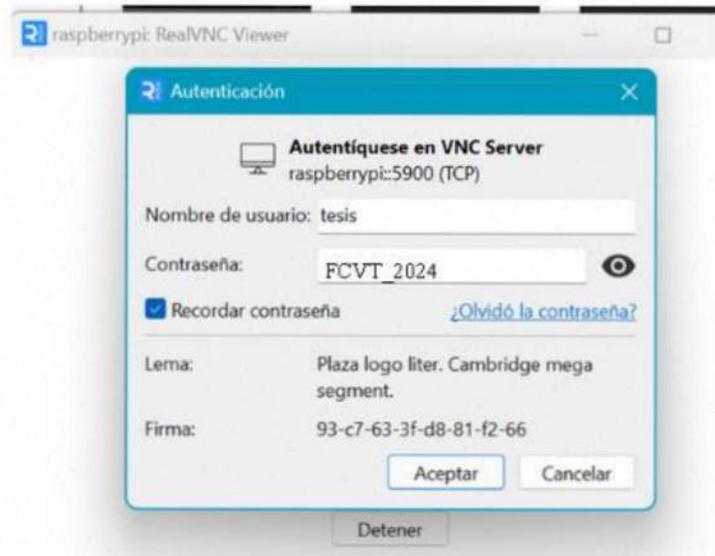


Ilustración 26. Autenticación en VNC server.

Nota: imagen sobre la Autenticación en VNC server:

Fuente: Elaboración propia.

4. Una vez ingresado el nombre del usuario y la contraseña, damos clic en aceptar y nos sale página principal.

Ilustración 27. Página principal

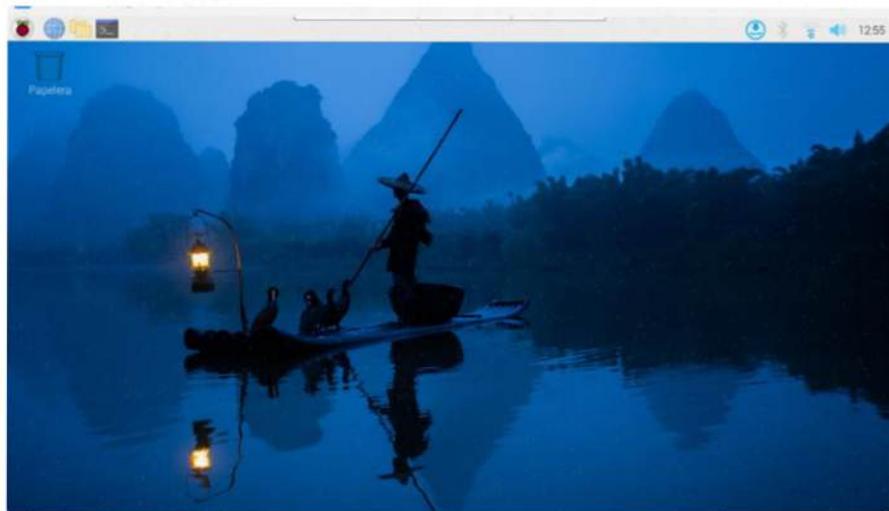


Ilustración 27. Página principal

Nota: imagen sobre la Página principal

Fuente: Elaboración propia.

5. Ya estando en el raspberry , se da clic en el símbolo de raspberry en lo cual nos aparece el Node-Red.

Ilustración 28. *Dirigirse a Node-Red.*



Ilustración 28. Dirigirse a Node-Red.

Nota: imagen al momento de Dirigirse a Node-Red.

Fuente: Elaboración propia.

6. Ingresamos al Node-Red nos pide usuario: tesis contraseña: FCVT_2024

Ilustración 29. *Iniciar las credenciales de Node-RED.*

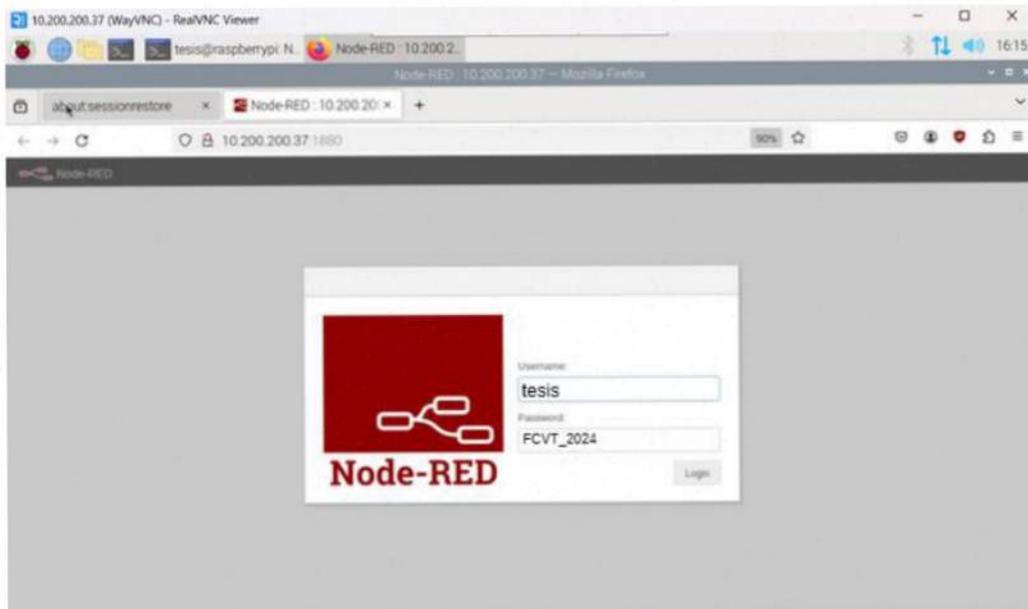


Ilustración 29. Iniciar las credenciales de Node-RED.

Nota: imagen al Iniciar las credenciales de Node-RED.

Fuente: Elaboración propia.

7. Ya estando en Node-Red veremos las configuraciones que tenemos, en este caso tenemos, recepción de Datos LoRa, esto filtra los datos del Esp, en el primero están los que visualizamos en el Dashboard que viene haciendo la temperatura, humedad, presión , caudal y total del libro, en el segundo del mismo Esp están los datos que se guardan en la base de datos MariaDB, también cada 5segundos se van guardado los datos, en ese espacio también podemos cambiar nombre, diseño para poder visualizar en el Dashboard.

Ilustración 30. Dentro de Node-RED.

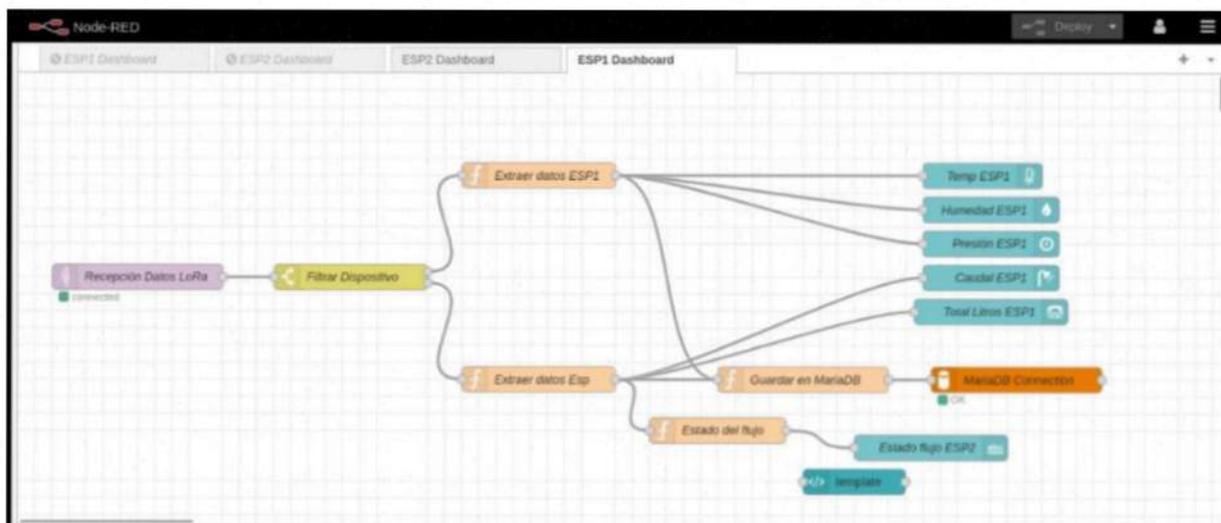


Ilustración 30. Configuración de Node-RED.

Nota: imagen para dentro de Node-RED.

Fuente: Elaboración propia.

Ilustración 31. Configuración de Node-Red

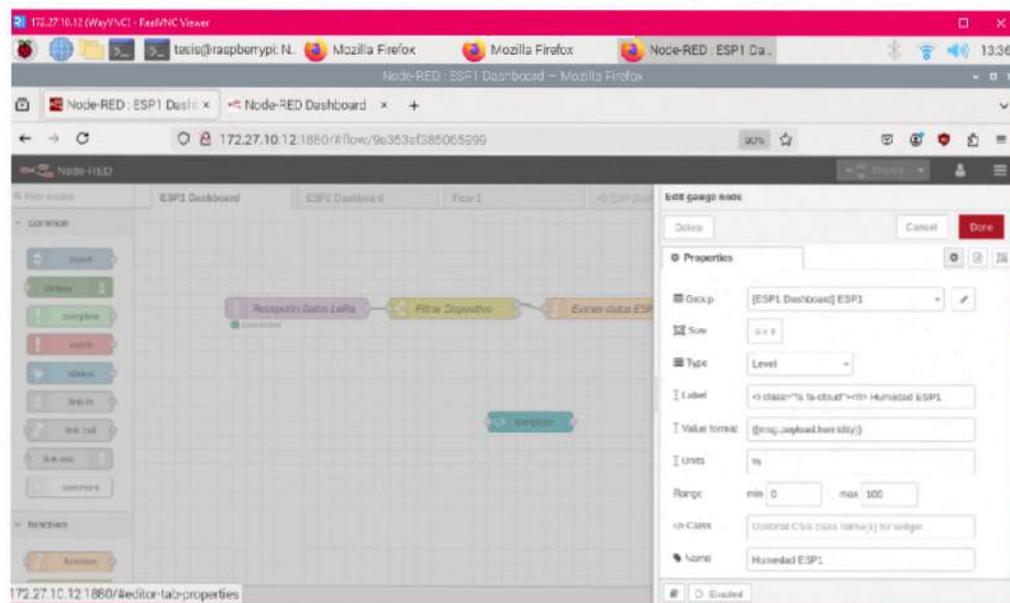


Ilustración 31. Configuración de Node-Red

Nota: imagen la configuración de Node-RED.

Fuente: Elaboración propia.

- De eso los cambios que realizamos lo hacemos con el siguiente link, cabe recalcar que tiene que estar con Red LAN e insertamos el siguiente link para visualizar los cambios que realizamos en Node-Red <http://10.200.200.28:1880/ui/> con este link ya estamos visualizando el Dashboard.

Ilustración 32. Visualización en Dashboard.



Ilustración 32. Visualización en Dashboard.

Nota: imagen sobre la visualización en Dashboard.

Fuente: Elaboración propia.

- Visualización de la base de datos, en este punto se ve como ingresar a la base de datos en este caso utilizamos MARIADB, para ingresar se realiza en consola desde raspberry, se coloca el comando de **sudo mariadb**, ya insertado el comando anterior seguimos con el siguiente **mariadb -u tesis -p** en este caso nos pide el usuario y contraseña, en este caso nuestro usuario es **tesis** y contraseña: **tesis123**.

Ilustración 33. Usuario-Contraseña MARIADB

```
tesis@raspberrypi: ~  
Archivo Editar Pestañas Ayuda  
tesis@raspberrypi:~ $ mariadb -u tesis -p  
Enter password:  
Welcome to the MariaDB monitor.  Commands end with ; or \g.  
Your MariaDB connection id is 33  
Server version: 10.11.6-MariaDB-0+deb12u1 Debian 12  
  
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
MariaDB [(none)]> █
```

Ilustración 33. Usuario-Contraseña MARIADB

Nota: imagen sobre el Usuario-Contraseña MARIADB

Fuente: Elaboración propia.

2. En eso ya estamos en nuestra base de datos para ellos se coloca el comando **use iot_data**.

Ilustración 34. Ingreso de la base de datos

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
MariaDB [(none)]> use iot_data
```

Ilustración 34. Ingreso de la base de datos

Nota: imagen como se ingresa a MARIADB

Fuente: Elaboración propia.

3. Después de ese comando se escribe este **SELECT * FROM esp1_data;** para leer los datos que se almacenan.

Ilustración 35. Verificación Base de datos.

id	esp1	esp2	esp3	esp4	esp5	esp6	esp7	esp8	esp9
70007	31.1	47.0	1000.0	0	0	2024-12-00 12:00:02			
70008	30.3	47	1000.0	27.9	2219	2024-12-00 12:00:03			
70009	31.1	47.0	1000.0	0	0	2024-12-00 12:00:04			
70010	30.3	47	1000.0	23.7	1847	2024-12-00 12:00:05			
70011	31.1	48	1000.0	0	0	2024-12-00 12:00:07			
70012	30.3	47.1	1000.0	21.5	2004	2024-12-00 12:00:08			
70013	31.2	46	1000.0	0	0	2024-12-00 12:00:10			
70014	30.3	47	1000.0	20	1837	2024-12-00 12:00:11			
70015	31.1	46	1000.0	0	0	2024-12-00 12:00:13			
70016	30.3	47	1000.0	24.2	2002	2024-12-00 12:00:14			
70017	31.2	46	1000.0	0	0	2024-12-00 12:00:15			
70018	30.3	47	1000.0	20	1808	2024-12-00 12:00:17			
70019	31.1	47.0	1000.0	0	0	2024-12-00 12:00:18			
70020	30.3	47	1000.0	20.4	1831	2024-12-00 12:00:19			
70021	31.1	46	1000.0	0	0	2024-12-00 12:00:21			
70022	30.3	47	1000.0	25.7	1859	2024-12-00 12:00:22			
70023	31.1	46	1000.0	0	0	2024-12-00 12:00:24			
70024	30.3	47	1000.0	20.9	1890	2024-12-00 12:00:25			
70025	31.1	46	1000.0	0	0	2024-12-00 12:00:27			
70026	30.3	47	1000.0	25.7	1717	2024-12-00 12:00:28			
70027	31.1	46	1000.0	0	0	2024-12-00 12:00:29			
70028	30.3	47	1000.0	23.9	1932	2024-12-00 12:00:31			
70029	31.1	47.0	1000.0	0	0	2024-12-00 12:00:32			
70030	30.3	47	1000.0	20.7	1858	2024-12-00 12:00:33			
70031	31.1	46	1000.0	0	0	2024-12-00 12:00:35			
70032	30.3	47	1000.0	19.5	1909	2024-12-00 12:00:36			
70033	31.1	47.0	1000.0	0	0	2024-12-00 12:00:38			
70034	30.3	47	1000.0	17	1847	2024-12-00 12:00:39			
70035	31.1	47.0	1000.0	0	0	2024-12-00 12:00:40			
70036	30.3	47	1000.0	25.4	1803	2024-12-00 12:00:42			
70037	31.1	47.0	1000.0	0	0	2024-12-00 12:00:43			
70038	30.3	47	1000.0	14.2	1748	2024-12-00 12:00:44			
70039	31.1	47.0	1000.0	0	0	2024-12-00 12:00:46			
70040	30.3	46.0	1000.0	15.1	1791	2024-12-00 12:00:47			
70041	31.1	47.0	1000.0	0	0	2024-12-00 12:00:49			
70042	30.3	46.0	1000.0	17.1	1876	2024-12-00 12:00:50			

Ilustración 35. Verificación Base de datos.

Nota: imagen sobre la verificación de la Base de datos.

Fuente: Elaboración propia.

4.4.3.2 Control del usuario

1. El usuario tiene que estar conectado a la red LAN de cualquier parte de la Universidad Laica Eloy Alfaro de Manabí insertando el siguiente link <http://10.200.200.28:1880/ui/> , el usuario solo tendrá acceso al dashboard, tendrá la visualización de la siguiente imagen:



Ilustración 36. Verificación Dashboard

Nota: imagen sobre la verificación en el Dashboard

Fuente: Elaboración propia.

Ilustración 36. Verificación Dashboard

4.4.3.3 Desarrollo y construcción

Ilustración 37. Plano de la estructura

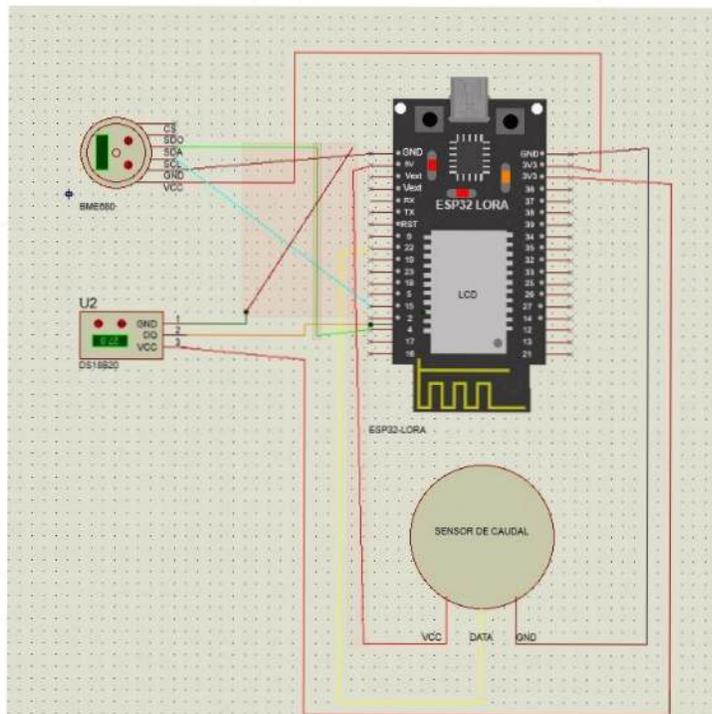


Ilustración 37. Plano de la estructura

Nota: imagen del plano de la estructura

Fuente: Elaboración propia.

Se diseñó un sistema de riego automatizado utilizando sensores y tecnología de IoT. El primer paso consistió en preparar la estructura física y electrónica para el monitoreo y control del flujo de agua y las condiciones ambientales.

El sistema utiliza un ESP32 LORA como microcontrolador principal, que recopila datos de varios sensores. En la imagen, se observa el diseño del circuito, el cual incluye:

- **BME680:** Sensor de calidad del aire y condiciones ambientales (temperatura, humedad, presión y compuestos volátiles). Este sensor está conectado al ESP32 para proporcionar datos en tiempo real sobre el ambiente.

- **DS18B20:** Sensor de temperatura digital, utilizado para monitorear la temperatura del agua en el sistema de riego.
- **Sensor de flujo:** Encargado de medir el caudal de agua en el sistema, asegurando que los nutrientes circulen de manera eficiente.

Se desarrolló un sistema de riego automatizado que utiliza un cajetín de control equipado con un ESP32 LORA encargado de recibir datos de sensores y controlar dispositivos conectados como una válvula rociadora de 3/4 que regula el flujo de agua hacia las mangueras el cajetín incluye un temporizador y relés que permiten gestionar el paso del agua además de una fuente de alimentación para los componentes electrónicos el agua se distribuye mediante mangueras conectadas a la válvula asegurando una irrigación eficiente hacia las áreas designadas para mangos y vetiver el sistema está integrado con una interfaz en Node-RED que muestra en tiempo real datos como la temperatura y humedad ambiental el estado del flujo de agua y el volumen total suministrado al sistema se realizaron pruebas para verificar el correcto funcionamiento asegurando que no existan fugas y que la válvula opere correctamente transmitiendo datos precisos desde los sensores hacia la plataforma de monitoreo

4.4.3.4 Diseño del sistema de riego

Herramientas necesarias:

Arduino:

- Placa de desarrollo ESP32-Lora V2 OLED

Sensores:

- BME680
- DS18B20
- Sensor de flujo de agua 3/4

Otros componentes:

- Orbit 57280 Válvula de rociador 3/4

- Transformador 110 a 24V
- Reloj Digital
- Breaker para riel din
- Placa perforadora 10x10

4.4.3.5 Descripción del diseño del circuito

Se diseñó un sistema de riego automatizado utilizando la herramienta tecnológica avanzada Proteus, una plataforma de simulación electrónica que facilita la simulación y evaluación de proyectos electrónicos y de programación de microcontroladores previo a su puesta en marcha física. En este diseño, se emplearon diversos sensores vinculados a una tarjeta de desarrollo ESP32, instalada en una protoboard para facilitar la organización y conexión de los componentes.

Entre los sensores implementados se encuentra el DHT11, utilizado para medir la temperatura y la humedad ambiental. Este sensor necesita tres conexiones fundamentales: energía, conexión eléctrica y una salida de datos. Adicionalmente, se implementó un sensor DS18B20 como segunda fuente de medición de temperatura, que incorpora una resistencia de soporte en la línea de datos para la medición de temperatura. También se integró un sensor de flujo de caudal, diseñado para medir el volumen de agua que circula en el sistema. Este sensor se conecta a la placa ESP32 mediante líneas de alimentación, tierra y datos, y permite calcular en tiempo real la cantidad de agua suministrada, asegurando un uso eficiente de los recursos hídricos.

Las conexiones se organizaron cuidadosamente en la protoboard para asegurar una comunicación eficiente entre los sensores y la placa de desarrollo, facilitando así la integración y la lectura de los datos producidos.

En Proteus, el diseño del circuito se centra en la obtención de datos fundamentales como temperatura, humedad y flujo, fusionando todos los sensores en una sola unidad de procesamiento. Esta unión permite monitorear las variables en tiempo real, optimizando el riego para cultivos de mango y vetiver. Además, el sistema es escalable y flexible, permitiendo adaptaciones para cubrir necesidades específicas de diferentes aplicaciones agrícolas y ambientales. Este enfoque asegura un diseño robusto y funcional que facilita tanto el análisis de datos como la mejora continua del sistema.

Ilustración 38. Circuito integrado



Ilustración 38. Circuito integrado

Nota: imagen sobre el circuito integrado

Fuente: Elaboración propia.

4.4.3.6 Creación y configuración del código

El esquema para el ESP32 está concebido para leer información de diversos sensores y enviarla a través de LoRa y MQTT para su seguimiento y estudio. Se emplearon varias librerías fundamentales: **WiFi.h** para la vinculación con el WiFi, **PubSubClient.h** para la comunicación mediante el protocolo MQTT, **LoRa.h** para la transmisión inalámbrica de datos, **Wire.h** para manejar la comunicación I2C, y **Adafruit_GFX.h** junto con **Adafruit_SSD1306.h** para la visualización de información en una pantalla OLED. También se usaron las librerías **Adafruit_Sensor.h**, **Adafruit_BME680.h**, **OneWire.h** y **DallasTemperature.h** para la lectura de los sensores ambientales.

En la configuración inicial, se definieron los pines de conexión para cada componente. El sensor BME680 está conectado mediante los pines I2C SDA 4 y SCL 15, mientras que el DS18B20 utiliza el pin 2 configurado como OneWire. El sensor de flujo de agua está conectado al pin 22 mediante una interrupción para contar pulsos, y la pantalla OLED usa los pines SDA 4, SCL 15 y RESET 16. Además, LoRa se configuró en los pines SS 18, RST 14 y DI0 26.

En el bucle principal del sketch `loop()`, se realizan lecturas periódicas de los sensores. El BME680 proporciona datos de temperatura, humedad y presión atmosférica, mientras que el DS18B20 mide la temperatura de manera precisa. El sensor de flujo calcula la tasa de flujo de agua en litros por minuto, el volumen instantáneo en mililitros y el flujo total acumulado. Estas lecturas se procesan y se formatean en un mensaje que se envía mediante LoRa al receptor y a través de MQTT al servidor. La pantalla OLED muestra las lecturas en tiempo real, facilitando un monitoreo local de las variables.

Para la transmisión de datos, el sketch utiliza LoRa para enviar mensajes en formato JSON. Cada mensaje incluye identificadores como "ESP1" o "ESP2" para distinguir entre dispositivos y los datos correspondientes a cada sensor. La función `reconnect()` asegura la reconexión al servidor MQTT en caso de pérdida de conexión, y la función de interrupción para el sensor de flujo cuenta los pulsos generados por el paso del agua para calcular el caudal.

Este diseño combina tecnologías avanzadas como LoRa, MQTT e I2C, integrando sensores de ambiente y flujo de agua en un sistema IoT robusto y eficiente. Su implementación permite monitorear en tiempo real las variables críticas de un sistema de riego automatizado, optimizando recursos y facilitando el análisis de datos tanto a nivel local como remoto.

4.4.4 Sistema web en la administración y herramienta de la base de datos

4.4.4.1 Sistema de Infraestructura:

Se creó una base de datos destinada a la administración y conservación de información proveniente de los sensores incorporados en el sistema de riego automático. La base de datos fue implementada en MariaDB e integrada con Node-RED, permitiendo el almacenamiento de las lecturas de los sensores en tiempo real de manera eficiente. El diseño de la base de datos asegura la integridad y consistencia de los datos, facilitando su monitoreo y análisis continuo.

La estructura de la base de datos incluye una tabla principal llamada **esp1_data**, la cual se encarga de almacenar las lecturas generadas por el dispositivo ESP32 y los sensores conectados, permitiendo una organización clara de los datos recopilados para su posterior acceso y análisis. Esta tabla incluye un identificador único (id), un campo para identificar el dispositivo (device_id), y valores que registran las lecturas de temperatura, humedad, presión atmosférica, tasa de flujo de agua (flow_rate) y el volumen total acumulado en mililitros (total_milliliters). Además, cuenta con una marca temporal (timestamp) que permite identificar el momento exacto en que se registraron las lecturas. Este diseño asegura una gestión eficiente de los datos y proporciona una solución robusta y adaptable para el sistema de riego automatizado.

Modelo de la base de datos:

Ilustración 39. Modelo conceptual de la base de datos

esp1_data	
<u>id</u>	<pi> Integer <M>
device_id	Variable characters (50)
temperature	Float
humidity	Float
pressure	Float
flow_rate	Float
total_milliliters	Float
timestamp	Date & Time
Identifier_1	<pi>

Ilustración 39. Modelo conceptual de la base de datos

Nota: imagen del modelo conceptual de la base de datos

Fuente: Elaboración propia.

4.4.4.2 Herramientas y Librerías Utilizadas:

Lenguajes de Programación:

- **Node-RED:** Fue utilizado para el diseño y procesamiento de los flujos de datos provenientes de los sensores, permitiendo el control lógico del sistema y la visualización dinámica en el dashboard.

Librerías:

- **Node-RED Dashboard:** Librería empleada para crear la interfaz gráfica que muestra las lecturas de los sensores en tiempo real, facilitando el monitoreo de variables como temperatura, humedad, presión y caudal de agua.
- **JSON:** Formato utilizado para estructurar y transmitir los datos de los sensores dentro de Node-RED, garantizando una integración fluida entre los componentes del sistema.

Base de Datos:

- **MariaDB:** Implementada para almacenar y gestionar las lecturas de los sensores, asegurando la integridad de los datos y permitiendo su consulta en tiempo real para análisis y monitoreo.

4.4.4.3 Entorno de Desarrollo

Node-RED: Plataforma utilizada para el desarrollo del sistema.

El sistema creado incluye múltiples secciones y flujos en Node-RED, cada uno con una función particular para la administración y representación de los datos de los sensores en tiempo real. Seguidamente, se describen las características de cada flujo:

a) Json dashboard:

Este código JSON simboliza la configuración de un tablero de seguimiento (dashboard) desarrollado en Node-RED para administrar y visualizar información de sensores vinculados a dispositivos ESP (como ESP32). El propósito principal de este tablero es

obtener datos de un canal MQTT, procesarlos y exhibirlos visualmente, además de almacenarlos en una base de datos MariaDB. Está concebido para laborar.

Ilustración 40. Código de Json dashboard.

```
{
  "id": "7b9abb04b9c70bc",
  "type": "tab",
  "label": "199 Dashboard",
  "disabled": false,
  "info": ""
},
{
  "id": "1e7b24b0e1aca2db",
  "type": "mqtt_in",
  "z": "2b00b0c43b0c70bc",
  "name": "Recepción datos Lora",
  "topic": "canal",
  "qos": "0",
  "datatype": "auto",
  "broker": "mqtt_broker",
  "nl": false,
  "trap": false,
  "inputs": 0,
  "x": 140,
  "y": 240,
  "wires": [
    [
      "4451ab0909090909"
    ]
  ]
},
{
  "id": "4451ab0909090909",
  "type": "switch",
  "z": "2b00b0c43b0c70bc",
  "name": "Filtros Dispositivo",
  "property": "payload",
  "propertyType": "msg",
  "rules": [
    [
      "cont"
    ]
  ]
}
```

Ilustración 40.código Json dashboard.

Nota: imagen sobre el código Json dashboard

Fuente: Elaboración propia.

Ilustración 41. Código de Json dashboard.

```

36      "t": "cont",
37      "v": "ESP1",
38      "vt": "str"
39    },
40  ],
41  "t": "cont",
42  "v": "ESP2",
43  "vt": "str"
44  },
45  ],
46  "checkbox": "true",
47  "repair": false,
48  "outputs": 2,
49  "x": 399,
50  "y": 240,
51  "wires": [
52  ],
53  [
54  ],
55  [
56  ],
57  ],
58  ],
59  ],
60  ],
61  "id": "092002f6b01ff04",
62  "type": "mysql",
63  "z": "2bbabed3bc708c",
64  "xyb": "85,bf0b3f61e171",
65  "name": "mysqli connection",
66  "x": 1170,
67  "y": 360,
68  "wires": [
69  ],
70  ],
71  ],
72  ],
73  ],
74  ],
75  ],
76  ],
77  ],
78  ],
79  ],
80  ],
81  ],
82  ],
83  ],
84  ],
85  ],
86  ],
87  ],
88  ],
89  ],
90  ],
91  ],
92  ],
93  ],
94  ],
95  ],
96  ],
97  ],
98  ],
99  ],
100  ],
101  ],

```

Ilustración 41. código Json dashboard.

Nota: imagen sobre el código Json dashboard

Fuente: Elaboración propia.

Ilustración 42. Código de Json dashboard.

```

68  "wires": [
69  ],
70  ],
71  ],
72  ],
73  "id": "2916dd3f440012",
74  "type": "ui_text",
75  "z": "2bbabed3bc708c",
76  "group": "123a2bfa6d0d3d",
77  "order": 4,
78  "width": 6,
79  "height": 1,
80  "name": "Caudal ESP1",
81  "label": "<div class='font-family: monospace;'>caudal ESP1 (L/min):",
82  "format": "({{log.payload.Flow}})",
83  "layout": "row-spread",
84  "className": "",
85  "style": true,
86  "font": "",
87  "fontSize": "",
88  "color": "#3385c5",
89  "x": 1140,
90  "y": 260,
91  "wires": [
92  ],
93  "icon": "font-awesome/fa-shower"
94  ],
95  "id": "7f1f3206f63aac5",
96  "type": "ui_gauge",
97  "z": "2bbabed3bc708c",
98  "name": "Temp ESP1",
99  "group": "02204e9648039850",
100  "order": 1,
101  "width": "6",

```

Ilustración 42. código Json dashboard.

Nota: imagen sobre el código Json dashboard

Fuente: Elaboración propia.

Ilustración 43. Código de Json dashboard.



Ilustración 43. código Json dashboard.

Nota: imagen sobre el código Json dashboard

Fuente: Elaboración propia.

Ilustración 44. Código de Json dashboard.



Ilustración 44. código Json dashboard.

Nota: imagen sobre el código Json dashboard

Fuente: Elaboración propia.

Ilustración 45. Código de Json dashboard.


```

238     "id": "0bda90ca19119db",
239     "type": "function",
240     "z": "2bcabbe49bc708c",
241     "name": "Guardar en MariaDB",
242     "func": "guardar_en_mariadb",
243     "outputs": 1,
244     "timeout": "0",
245     "nodes": 0,
246     "initialize": "",
247     "finalize": "",
248     "lib": [
249     ],
250     "x": 920,
251     "y": 380,
252     "wires": [
253     ],
254     "id": "d02a2f6e6a0ff0a",
255     "type": "ui_template",
256     "z": "2bcabbe49bc708c",
257     "group": "c23a2bfacfd4d13d",
258     "name": "",
259     "order": 0,
260     "width": 0,
261     "height": 0,
262     "id": "b33aac0b0a0610",
263     "type": "ui_template",
264     "z": "2bcabbe49bc708c",
265     "group": "c23a2bfacfd4d13d",
266     "name": "",
267     "order": 0,
268     "width": 0,
269     "height": 0,
270     "id": "70ca11155d0cf2be",
271     "type": "ui_text",
272     "z": "2bcabbe49bc708c",
273     "group": "c23a2bfacfd4d13d",
274     "order": 0,
275     "width": "b",
276     "height": "1",
277     "name": "Estado Flujo ESP2",
278     "label": "Estado del Flujo",
279     "format": "{{msg.payload.status}}",
280     "layout": "row spread",
281     "classmate": "",
282     "style": true,
283     "font": "",
284     "fontSize": "",
285     "color": "ff5722",
286     "x": 1070,
287     "y": 440,
288     "wires": [
289     ]

```

Ilustración 47. código Json dashboard.

Nota: imagen sobre el código Json dashboard

Fuente: Elaboración propia.

Ilustración 48. Código de Json dashboard.

```

270     "height": 0,
271     "format": "style>ui body (N background-image: url(\\\"https://media.istockphoto.com/id/130978893/es/vector/Invernad
272     \"storeOutMessages\": true,
273     \"fadeInMessages\": true,
274     \"resendOnRefresh\": true,
275     \"simulateScope\": \"local\",
276     \"classmate\": \"\",
277     \"x\": 980,
278     \"y\": 480,
279     \"wires\": [
280     ]
281     ],
282     \"id\": \"70ca11155d0cf2be\",
283     \"type\": \"ui_text\",
284     \"z\": \"2bcabbe49bc708c\",
285     \"group\": \"c23a2bfacfd4d13d\",
286     \"order\": 0,
287     \"width\": \"b\",
288     \"height\": \"1\",
289     \"name\": \"Estado Flujo ESP2\",
290     \"label\": \"Estado del Flujo\",
291     \"format\": \"{{msg.payload.status}}\",
292     \"layout\": \"row spread\",
293     \"classmate\": \"\",
294     \"style\": true,
295     \"font\": \"\",
296     \"fontSize\": \"\",
297     \"color\": \"ff5722\",
298     \"x\": 1070,
299     \"y\": 440,
300     \"wires\": [
301     ]
302     ],
303     \"id\": \"b33aac0b0a0610\",
304     \"type\": \"ui_template\",
305     \"z\": \"2bcabbe49bc708c\",
306     \"group\": \"c23a2bfacfd4d13d\",
307     \"name\": \"\",
308     \"order\": 0,
309     \"width\": 0,
310     \"height\": 0,
311     \"id\": \"d02a2f6e6a0ff0a\",
312     \"type\": \"ui_template\",
313     \"z\": \"2bcabbe49bc708c\",
314     \"group\": \"c23a2bfacfd4d13d\",
315     \"name\": \"\",
316     \"order\": 0,
317     \"width\": 0,
318     \"height\": 0,
319     \"id\": \"0bda90ca19119db\",
320     \"type\": \"function\",
321     \"z\": \"2bcabbe49bc708c\",
322     \"name\": \"Guardar en MariaDB\",
323     \"func\": \"guardar_en_mariadb\",
324     \"outputs\": 1,
325     \"timeout\": \"0\",
326     \"nodes\": 0,
327     \"initialize\": \"\",
328     \"finalize\": \"\",
329     \"lib\": [
330     ],
331     \"x\": 920,
332     \"y\": 380,
333     \"wires\": [
334     ]

```

Ilustración 48. código Json dashboard.

Nota: imagen sobre el código Json dashboard

Fuente: Elaboración propia.

Ilustración 49. Código de Json dashboard.

```
303 },
304
305 "id": "888e908f93190b",
306 "type": "function",
307 "i": "78b8bb44b9c708c",
308 "name": "Estado del flujo",
309 "func": "if (msg.payload.flow === 0) {\n  msg.payload.status = 'no hay flujo de agua';\n} else {\n  msg.payload.status = '
310 "outputs": 1,
311 "timeout": "",
312 "noerr": 0,
313 "initialize": "",
314 "finalize": "",
315 "libs": [],
316 "x": 820,
317 "y": 420,
318 "wires": [
319   [
320     "76ca1155d8cf2be"
321   ]
322 ]
323 },
324
325 "id": "mqtt_broker",
326 "type": "mqtt-broker",
327 "name": "mqtt broker",
328 "broker": "10.42.5.58",
329 "port": "1883",
330 "clientId": "",
331 "autoconnect": true,
332 "usetls": false,
333 "protocolVersion": "4",
334 "keepalive": "60",
335 "cleansession": true,
336 "autounsubscribe": true,
337 "birthTopic": ""
```

Ilustración 49. código Json dashboard.

Nota: imagen sobre el código Json dashboard

Fuente: Elaboración propia.

Ilustración 50. Código de Json dashboard.

```

337     "birthTopic": "",
338     "birthQos": "0",
339     "birthPayload": "",
340     "birthMsg": {},
341     "closeTopic": "",
342     "closeQos": "0",
343     "closePayload": "",
344     "closeMsg": {},
345     "willTopic": "",
346     "willQos": "0",
347     "willPayload": "",
348     "willMsg": {},
349     "userProps": "",
350     "sessionExpiry": ""
351   },
352   {
353     "id": "85cbf0b33fe1e173",
354     "type": "MySQL database",
355     "name": "iot_data",
356     "host": "localhost",
357     "port": "3306",
358     "db": "iot_data",
359     "tz": "local",
360     "charset": "UTF-8"
361   },
362   {
363     "id": "c23a2bfacddd13d",
364     "type": "ui group",
365     "name": "ACUA",
366     "tab": "5da5b7184717f46f",
367     "order": 1,
368     "disp": true,
369     "width": "6",
370     "collapse": false,
371     "iconName": ""

```

Ilustración 50. código Json dashboard.

Nota: imagen sobre el código Json dashboard

Fuente: Elaboración propia.

Ilustración 51. Código de Json dashboard.

```
362 {
363   "id": "r23a2bfa6d113d",
364   "type": "ui_group",
365   "name": "AGUA",
366   "lab": "5da5b7104717f46f",
367   "order": 1,
368   "disp": true,
369   "width": "6",
370   "collapse": false,
371   "className": ""
372 },
373 {
374   "id": "e2204e9648639850",
375   "type": "ui_group",
376   "name": "RICO EXTERIOR",
377   "lab": "5da5b7104717f46f",
378   "order": 1,
379   "disp": true,
380   "width": "6",
381   "collapse": true,
382   "className": ""
383 },
384 {
385   "id": "5da5b7104717f46f",
386   "type": "ui_lab",
387   "name": "TESTS-SISTEMA ØVC±ØVØ",
388   "icon": "dashboard",
389   "order": 1,
390   "disabled": false,
391   "hidden": false
392 }
393 }
```

Ilustración 51. código Json dashboard.

Nota: imagen sobre el código Json dashboard

Fuente: Elaboración propia.

b) ESP32 Emisor

Este programa establece un sistema de transmisión de datos sensoriales mediante un microcontrolador ESP32, sensores y comunicación inalámbrica LoRa (Long Range). Su principal objetivo es cuantificar diversos parámetros físicos y enviarlos de manera remota a través de LoRa para su análisis o seguimiento.

Ilustración 52. Código del ESP32 sobre el EMISON

```
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 #include <LoRa.h>
4 #include <Wire.h>
5 #include <Adafruit_GFX.h>
6 #include <Adafruit_SSD1306.h>
7
8 #define SS 18
9 #define RS1 14
10 #define DTR 26
11 #define BMM180 8
12
13
14 #define SCREEN_WIDTH 128
15 #define SCREEN_HEIGHT 64
16 #define OLED_RESET 16
17 #define I2C_SDA 4
18 #define I2C_SCL 15
19
20 Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);
21
22 // wifi y mqtt
23 #include <espclient.h>
24 #include <pubsubclient.h>
25
26 char msg[128]; // Buffer para manejar el mensaje en formato JSON
27
28 String esp1_data = ""; // Para almacenar los datos de ESP1
29 String esp2_data = ""; // Para almacenar los datos de ESP2
30
31 void setup() {
32   Serial.begin(9600);
33 }
```

Ilustración 52. Código del ESP32 sobre el EMISON

Nota: imagen relacionada con el código del ESP32 sobre el EMISON

Fuente: Elaboración propia.

Ilustración 53. Código del ESP32 sobre el EMISON

```
34 #include <Wire.h>
35 #include <Adafruit_Sensor.h>
36 #include <Adafruit_BME680.h>
37 #include <OneWire.h>
38 #include <DallasTemperature.h>
39 #include <Adafruit_GFX.h>
40 #include <Adafruit_SSD1306.h>
41 #include <LoRa.h>
42
43 // definiciones del sensor BME680 y pantalla OLED
44 #define I2C_SDA 4 // Pin SDA para BME680
45 #define I2C_SCL 15 // Pin SCL para BME680
46 Adafruit_BME680 bme;
47
48 // definiciones del sensor DS18B20
49 #define ONE_WIRE_BUS 2 // Pin para DS18B20
50 OneWire oneWire(ONE_WIRE_BUS);
51 DallasTemperature sensors(&oneWire);
52
53 // definiciones de la pantalla OLED
54 #define SCREEN_WIDTH 128
55 #define SCREEN_HEIGHT 64
56 #define OLED_RESET 16 // GPIO 16 para reset
57 #define SCREEN_ADDR 0x3C
58 Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);
59
60 #define SS 18
61 #define RS1 14
62 #define DTR 26
63 #define BMM180 8
```

Ilustración 53. Código del ESP32 sobre el EMISON

Nota: imagen relacionada con el código del ESP32 sobre el EMISON

Fuente: Elaboración propia.

Ilustración 54. Código del ESP32 sobre el EMISON.

```
96 display.setTextColor(SSD1306_WHITE);
97 display.setCursor(0, 0);
98 display.setCursor(0, 0);
99 display.display();
100
101 // inicializar el sensor BME680
102 if (!bme.begin(I2C_ADDR) || !bme.begin(I2C_ADDR)) {
103     Serial.println("No se encontró un sensor BME680 válido, ¡verifique la conexión!");
104     while (1);
105 }
106 bme.setTemperatureOversampling(BME680_OS_8X);
107 bme.setHumidityOversampling(BME680_OS_2X);
108 bme.setPressureOversampling(BME680_OS_4X);
109 bme.setIIRFilterSize(BME680_IIR_FILTER_1);
110 bme.setGasHeater(320, 150); // 320°C por 150 ms
111
112 // Inicializar el sensor DS18B20
113 pinMode(ONE_WIRE_BUS, INPUT_PULLUP);
114 sensors.begin();
115
116 // Inicializar el sensor de flujo de agua
117 pinMode(FLOW_SENSOR_PIN, INPUT_PULLUP);
118 attachInterrupt(digitalPinToInterrupt(FLOW_SENSOR_PIN), pulseCounter, FALLING);
119 }
120
121 void loop() {
122     // Leer el sensor BME680
123     unsigned long endTime = bme.beginReading();
124     if (endTime == 0) {
125         Serial.println("Fallo al comenzar la lectura del BME680 :(");
126         return;
127     }
128     delay(endTime - millis());
```

Ilustración 54. Código del ESP32 sobre el EMISON

Nota: imagen relacionada con el código del ESP32 sobre el EMISON

Fuente: Elaboración propia.

Ilustración 55. Código del ESP32 sobre el EMISON

```
64
65 #define FLOW_SENSOR_PIN 22 // Cambia este pin si es necesario y no está en conflicto
66 volatile int pulseCount = 0;
67 float flowRate; // Tasa de flujo en litros por minuto
68 float flowMillilitres; // Flujo instantáneo en mililitros
69 unsigned long totalMillilitres = 0; // Flujo total acumulado en mililitros
70 unsigned long oldTime = 0;
71
72 void setup() {
73     Serial.begin(115200);
74
75     SPI.begin(19, 22, 5); // pines para SPI
76     LoRa.setPins(SS, RST, DIO);
77     if (!LoRa.begin(1000)) {
78         Serial.println("Error al inicializar LoRa");
79         while (1);
80     }
81     Serial.println("LoRa inicializado");
82
83     // Inicializar I2C y pantalla OLED
84     Wire.begin(I2C_SDA, I2C_SCL);
85     pinMode(OLED_RESET, OUTPUT);
86     digitalWrite(OLED_RESET, LOW); // Reset de la pantalla
87     delay(20);
88     digitalWrite(OLED_RESET, HIGH); // Final del reset
89
90     if (!display.begin(SSD1306_SWITCHCAPVCC, SCREEN_WIDTH)) {
91         Serial.println("No se pudo encontrar una pantalla OLED.");
92         while (true);
93     }
94
95     display.clearDisplay();
96     display.setTextColor(SSD1306_WHITE);
```

Ilustración 55. Código del ESP32 sobre el EMISON

Nota: imagen relacionada con el código del ESP32 sobre el EMISON

Fuente: Elaboración propia.

Ilustración 56. Código del ESP32 sobre el EMISON

```
128 delay(endTime - millis());
129
130 if (!bme.endHeading()) {
131   Serial.println("Fallo al completar la lectura del BME680 :(");
132   return;
133 }
134
135 float humidity = bme.humidity;
136 float pressure = bme.pressure / 100.0; // conversión a hPa
137
138 // leer el sensor DS18B20
139 sensors.requestTemperatures();
140 float ds18b20Temperature = sensors.getTempCByIndex(0);
141
142 // Calcular el tiempo transcurrido en milisegundos para el sensor de flujo
143 unsigned long currentTime = millis();
144 unsigned long elapsedTime = currentTime - oldTime;
145
146 // Cada segundo (1000 ms), calcular la tasa de flujo
147 if (elapsedTime > 1000) {
148   // Desactivar la interrupción temporalmente para evitar errores durante el cálculo
149   detachInterrupt(digitalPinToInterrupt(FLOW_SENSOR_PIN));
150
151   // Calcular la tasa de flujo en litros por minuto
152   flowRate = ((1000.0 / elapsedTime) * pulseCount) / 7.5; // Ajustar la constante según el modelo del sensor
153
154   // Convertir flujo a mililitros por segundo
155   flowMillilitres = (flowRate / 60) * 1000;
156
157   // Sumar al total de mililitros
158   totalMillilitres += flowMillilitres;
159
160   // Mostrar los resultados en el Monitor Serial
```

Ilustración 56. Código del ESP32 sobre el EMISON

Nota: imagen relacionada con el código del ESP32 sobre el EMISON

Fuente: Elaboración propia.

Ilustración 57. Código del ESP32 sobre el EMISON

```
160 // Mostrar los resultados en el Monitor Serial
161 Serial.print("Tasa de flujo (L/min): ");
162 Serial.println(flowRate);
163 Serial.print("Flujo instantáneo (ml): ");
164 Serial.println(flowMillilitres);
165 Serial.print("Flujo total (mL): ");
166 Serial.println(totalMillilitres);
167 Serial.println();
168
169 // Reiniciar el contador y actualizar el tiempo
170 pulseCount = 0;
171 oldTime = currentTime;
172
173 // Reactivar la interrupción
174 attachInterrupt(digitalPinToInterrupt(FLOW_SENSOR_PIN), pulseCounter, FALLING);
175 }
176
177 // Encender la pantalla OLED
178 display.clearDisplay();
179 display.setCursor(0, 0);
180
181 // Mostrar lectura del DS18B20
182 if (ds18b20Temperature != -127.00) { // Verificar si la temperatura es válida
183   Serial.print("Temperatura DS18B20: ");
184   Serial.print(ds18b20Temperature);
185   Serial.println(" C");
186
187   display.print("Temp DS18B20: ");
188   display.print(ds18b20Temperature);
189   display.println(" C");
190 } else {
191   Serial.println("Error: No se pudo leer el sensor DS18B20.");
```

Ilustración 57. Código del ESP32 sobre el EMISON

Nota: imagen relacionada con el código del ESP32 sobre el SERVIDOR

Fuente: Elaboración propia.

Ilustración 58. Código del ESP32 sobre el EMISON

```
101 | | Serial.println("Error: No se pudo leer el sensor DS18B20.");
102 | | display.print("Sensor DS18B20 error");
103 | | }
104 |
105 | Serial.print("Hum: ");
106 | Serial.print(humidity);
107 | Serial.println(" %");
108 |
109 | Serial.print("Pres: ");
110 | Serial.print(pressure);
111 | Serial.println(" hPa");
112 |
113 | display.setCursor(0,0);
114 | display.print(humidity);
115 | display.println(" %");
116 | display.print("Pres: ");
117 | display.print(pressure);
118 | display.println(" hPa");
119 |
120 |
121 | display.print("Caudal (L/min): ");
122 | display.println(limitate);
123 |
124 |
125 |
126 | String message = "ESP1," + String(dS18B20Temperature, 1) + "," + String(humidity, 1) + "," + String(pressure, 1) + "," + String(flowRate, 1) + "," + String(totalWaterLitres);
127 |
128 | // Enviar mensaje mediante I2C
129 | LoRa.beginPacket();
130 | LoRa.print(message);
131 | LoRa.endPacket();
132 |
133 | display.display();
```

Ilustración 58. Código del ESP32 sobre el EMISON

Nota: imagen relacionada con el código del ESP32 sobre el EMISON

Fuente: Elaboración propia.

Ilustración 59. Código del ESP32 sobre el EMISON

```
222 | | display.display();
223 | |
224 | | delay(2000);
225 | | }
226 |
227 | void pulsetcounter() {
228 | | pulsetcount++;
229 | | }
230 |
```

Ilustración 59. Código del ESP32 sobre el EMISON

Nota: imagen relacionada con el código del ESP32 sobre el EMISON

Fuente: Elaboración propia.

c) ESP32 Servidor

Este código nos permite establecer un ESP32 como servidor para recibir mensajes vía LoRa, procesarlos y publicarlos en un servidor MQTT. Además, emplea una pantalla OLED para exhibir datos en tiempo real.

Ilustración 51. Código del ESP32 sobre el SERVIDOR

```
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 #include <LoRa.h>
4 #include <Wire.h>
5 #include <Adafruit_GFX.h>
6 #include <Adafruit_SSD1306.h>
7
8 // ||||| #define SS 10
9 #define NGT 14
10 #define DTG 26
11 #define DMD 0x80FF
12
13 // ||||| #define SCREEN_WIDTH 128
14 #define SCREEN_HEIGHT 64
15 #define OLED_RESET 16
16 #define OLED_SDA 4
17 #define OLED_SCL 15
18
19 Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);
20
21 WiFiClient espClient;
22 PubSubClient client(espClient);
23
24 char msg[128]; // buffer para manejar el mensaje en formato json
25
26 string esp1_data = "";
27 string esp2_data = "";
28
29 void setup() {
30   serial.begin(9600);
31   pinMode(OLED_RESET, OUTPUT);
32   digitalWrite(OLED_RESET, LOW);
33   delay(20);
```

Ilustración 60. Código del ESP32 sobre el SERVIDOR

Nota: imagen relacionada con el código del ESP32 sobre el SERVIDOR

Fuente: Elaboración propia.

Ilustración 51. Código del ESP32 sobre el SERVIDOR.

```
34   digitalWrite(OLED_RESET, HIGH);
35
36   Wire.begin(O_SDA, O_SCL);
37
38   if (!display.begin(SSD1306_SWITCHCAPWCC, 0x3C)) {
39     Serial.println("Falha iniciando SSD1306");
40     while (true);
41   }
42
43   display.clearDisplay();
44   display.setTextColor(WHITE);
45   display.setTextSize(1);
46   display.setCursor(0, 0);
47   display.display();
48
49   SPI.begin(5, 18, 27, SS); // Pines de tabla
50   LoRa.begin(125, 767, DTG);
51   if (!LoRa.begin(800)) {
52     Serial.println("Error al inicializar LoRa");
53     display.clearDisplay();
54     display.setCursor(0, 0);
55     display.print("error al inicializar LoRa");
56     display.display();
57     while (1);
58   }
59   Serial.println("LoRa iniciado");
60   display.clearDisplay();
61   display.setCursor(0, 0);
62   display.print("LoRa iniciado");
63   display.display();
64
65   setup_wifi();
```

Ilustración 61. Código del ESP32 sobre el SERVIDOR

Nota: imagen relacionada con el código del ESP32 sobre el SERVIDOR

Fuente: Elaboración propia.

Ilustración 51. Código del ESP32 sobre el SERVIDOR

```
65 | setup_wifi();
66 | client.setServer("192.168.1.104", 8080);
67 | client.setCallback(callback);
68 |
69 |
70 | void loop() {
71 |
72 |   if (!client.connected()) {
73 |     reconnect();
74 |   }
75 |   client.loop();
76 |
77 |   int packetSize = LoRa.parsePacket();
78 |   if (packetSize) {
79 |     String received = "";
80 |     while (LoRa.available()) {
81 |       received += (char)LoRa.read();
82 |     }
83 |
84 |     Serial.println("Mensaje recibido via LoRa: " + received);
85 |
86 |     String identifier = "";
87 |     String data = "";
88 |     if (received.startsWith("ESP1,") {
89 |       identifier = "ESP1";
90 |       esp1_data = received.substring(5); // Remove "ESP1,"
91 |       data = esp1_data;
92 |     } else if (received.startsWith("ESP2,") {
93 |       identifier = "ESP2";
94 |       esp2_data = received.substring(5); // Remove "ESP2,"
95 |       data = esp2_data;
96 |     } else {
```

Ilustración 62. Código del ESP32 sobre el SERVIDOR

Nota: imagen relacionada con el código del ESP32 sobre el SERVIDOR

Fuente: Elaboración propia.

Ilustración 51. Código del ESP32 sobre el SERVIDOR

```
95 |   } else {
96 |     Serial.println("Identificador no reconocido");
97 |     display.clearDisplay();
98 |     display.setCursor(0, 0);
99 |     display.print("Identificador no reconocido");
100 |    display.display();
101 |    return;
102 |   }
103 |
104 |   snprintf(msg, sizeof(msg), "%s,%s", identifier.c_str(), data.c_str());
105 |
106 |   client.publish("channel", msg);
107 |
108 |   Serial.println("Mensaje enviado: " + String(msg));
109 |   display.clearDisplay();
110 |   display.setCursor(0, 0);
111 |   display.print("ESP1: ");
112 |   display.print(esp1_data);
113 |   display.setCursor(0, 20);
114 |   display.print("ESP2: ");
115 |   display.print(esp2_data);
116 |   if (esp2_data.indexOf(",") != -1) {
117 |     String flowRate = esp2_data.substring(esp2_data.lastIndexOf(",") + 1);
118 |     display.setCursor(0, 40);
119 |     display.print("Caudal (L/min): ");
120 |     display.print(flowRate);
121 |     Serial.println("Flujo total (L): " + flowRate);
122 |   }
123 |
124 |   display.display();
125 | }
126 |
127 | }
```

Ilustración 63. Código del ESP32 sobre el SERVIDOR

Nota: imagen relacionada con el código del ESP32 sobre el SERVIDOR

Fuente: Elaboración propia.

Ilustración 51. Código del ESP32 sobre el SERVIDOR

```
127
128 void setup_wifi() {
129   display.clearDisplay();
130   display.setCursor(0, 0);
131   display.print("Conectando a WiFi...");
132   display.display();
133
134   WiFi.begin("", "");
135   while (WiFi.status() != WL_CONNECTED) {
136     delay(500);
137     Serial.print(".");
138     display.print(".");
139     display.display();
140   }
141   Serial.println("");
142   Serial.println("WiFi conectado");
143   Serial.println("Dirección IP: ");
144   Serial.println(WiFi.localIP());
145
146   display.clearDisplay();
147   display.setCursor(0, 0);
148   display.print("WiFi conectado");
149   display.setCursor(0, 10);
150   display.print("IP:");
151   display.setCursor(10, 10);
152   display.print(WiFi.localIP());
153   display.display();
154 }
155
156 void reconnect() {
157   while (!client.connected()) {
158     Serial.print("Intentando conexión MQTT...");
159     display.clearDisplay();
```

Ilustración 64. Código del ESP32 sobre el SERVIDOR

Nota: imagen relacionada con el código del ESP32 sobre el SERVIDOR

Fuente: Elaboración propia.

Ilustración 51. Código del ESP32 sobre el SERVIDOR

```
156 void reconnect() {
157   while (!client.connected()) {
158     Serial.print("Intentando conexión MQTT...");
159     display.clearDisplay();
160     display.setCursor(0, 0);
161     display.print("Conectando a MQTT...");
162     display.display();
163
164     if (client.connect("CONASCLIENT")) { // nombre del cliente MQTT
165       Serial.println("Conectado");
166       display.clearDisplay();
167       display.setCursor(0, 0);
168       display.print("Conexión MQTT OK");
169       display.display();
170     } else {
171       Serial.print("Falló, rca");
172       Serial.print(client.state());
173       Serial.println(" Intentando de nuevo en 5 segundos");
174       display.setCursor(0, 10);
175       display.print("Intentando...");
176       display.display();
177       delay(5000);
178     }
179   }
180 }
181
182 void callback(char* topic, byte* payload, unsigned int length) {
183   // Manejo de mensajes recibidos en el servidor MQTT (no implementado en este ejemplo)
184 }
185
```

Ilustración 65. Código del ESP32 sobre el SERVIDOR

Nota: imagen relacionada con el código del ESP32 sobre el SERVIDOR

Fuente: Elaboración propia.

4.4.4.4 Revisión y retrospectiva

Su diseño eficaz, sencillo de manejar y escalable lo hace una elección sobresaliente para varios usuarios. El sistema de riego automatizado creado emplea elementos electrónicos y tecnologías avanzadas como Node-RED y MariaDB para asegurar un control exacto de parámetros tales como la humedad ambiental, la temperatura y la circulación del agua. Estas habilidades brindan múltiples beneficios importantes en la administración de cultivos como el mango y el vetiver, permitiendo un monitoreo constante que favorece la toma de decisiones informadas por parte del encargado para optimizar el uso de recursos hídricos y mejorar la productividad. El diseño intuitivo basado en herramientas de monitoreo en tiempo real, como el dashboard de Node-RED, facilita su operación incluso para usuarios sin experiencia técnica previa. Además, la escalabilidad del sistema posibilita su adaptación flexible a diversos espacios y necesidades agrícolas. La integración de sensores precisos y un flujo continuo de datos garantiza una gestión eficiente y moderna del riego.

Sin embargo, plantea retos como un alto costo que podría restringir su puesta en marcha inicial, la demanda de habilidades técnicas especializadas para la instalación y el mantenimiento, y la potencial susceptibilidad a errores eléctricos o de software. Estos elementos necesitan un cuidado meticuloso para garantizar la máxima eficacia y longevidad del sistema en diferentes contextos agrícolas. Es esencial planificar estrategias para mitigar estos riesgos, como proporcionar capacitación técnica y asegurar redundancias en los componentes críticos.

El desarrollo de este proyecto representó un proceso de aprendizaje y desarrollo enriquecedor para el equipo participante. A lo largo del proceso, se adquirieron valiosos conocimientos sobre el diseño, la integración y el mantenimiento de sistemas automatizados de riego, además de un entendimiento más detallado de los retos y posibilidades en la administración sostenible del agua en la agricultura.

Las vivencias durante la realización del proyecto han aportado enseñanzas significativas:

- La planificación y el diseño minucioso desde las fases iniciales son cruciales para garantizar el éxito del proyecto, permitiendo una integración eficiente entre los sensores, la base de datos y la interfaz de monitoreo.

- Es importante mantener una actitud flexible y adaptable ante posibles imprevistos, como la necesidad de modificar componentes o resolver problemas relacionados con el flujo de datos, adoptando enfoques creativos y ajustando las estrategias según las circunstancias.
- La colaboración interdisciplinaria con especialistas en áreas como la agronomía y la informática desempeñó un papel fundamental al ampliar las perspectivas y contribuir al desarrollo de soluciones completas que combinan tecnología avanzada con las demandas específicas del sector agrícola.

CAPITULO V

5. Evaluación de Resultados

5.1 Introducción

Este capítulo expone los resultados alcanzados, que se encuentran en consonancia con los objetivos establecidos al comienzo de esta investigación. El objetivo principal fue desarrollar y analizar cada aspecto del sistema de riego automatizado, comenzando con el levantamiento de información sobre las necesidades hídricas del área de estudio, y culminando con la implementación y evaluación del sistema de monitoreo y control en tiempo real.

El propósito de este capítulo es ofrecer una visión clara y comprensiva de los resultados logrados, resaltando tanto los avances obtenidos como las áreas de mejora identificadas durante el proceso. Se destacan los elementos clave que permitieron abordar el problema de investigación y evaluar la eficacia del sistema propuesto.

A lo largo de este capítulo se presentan gráficos, tablas y diagramas que facilitan la comprensión de los datos obtenidos, apoyando un análisis riguroso de los resultados. Estos recursos son esenciales para evidenciar la efectividad del sistema en la optimización del uso del agua y en la automatización del riego.

Finalmente, este capítulo proporciona la base para las conclusiones y recomendaciones que se desarrollan en el siguiente capítulo. Los hallazgos obtenidos se conectan con aplicaciones prácticas y con posibles líneas de investigación futuras que puedan expandir y perfeccionar las funcionalidades del sistema.

5.2 Presentación de resultados

En esta parte se muestran los logros alcanzados durante la creación e implementación del sistema de riego automatizado. Los resultados se presentan de acuerdo con los objetivos específicos establecidos en la fase inicial de la investigación, con el fin de evaluar el cumplimiento de cada uno de ellos y proporcionar una visión clara sobre el desempeño y eficacia del sistema.

Los resultados incluyen datos obtenidos de las pruebas realizadas, análisis de rendimiento y observaciones clave sobre la implementación y operación del sistema de riego automatizado. A través de la presentación de estos resultados, se busca validar la efectividad de la solución propuesta y proporcionar una base para las conclusiones y futuras mejoras.

5.2.1 Resultados según objetivos planteados

A continuación, se presentan los resultados obtenidos en función de los objetivos específicos planteados para esta investigación, en el contexto del desarrollo e implementación del sistema de riego automatizado en los exteriores del bloque de Informática de la UNIVERSIDAD LAICA ELOY ALFARO DE MANABÍ.

Objetivo Específico 1: Realizar el levantamiento de información y análisis de los procesos necesarios para la automatización del sistema de riego.

- **Resultado:** Durante la fase de investigación inicial, se recopiló información relevante sobre las necesidades hídricas del área seleccionada para el sistema de riego. Se realizaron visitas al sitio y mediciones de humedad del suelo en diferentes puntos. Esta información permitió determinar las condiciones óptimas para el riego y seleccionar los sensores adecuados para monitorear la humedad y temperatura del suelo.
- **Evaluación:** El levantamiento de información fue crucial para garantizar que el sistema automatizado operara en función de las condiciones específicas del entorno y los cultivos en el área. El análisis permitió definir los rangos de humedad que activarían el sistema de riego.

-

Ilustración 66. Levantamiento de información



Ilustración 66. Levantamiento de información

Nota: imagen del levantamiento de información

Fuente: Elaboración propia.

Objetivo Específico 2: Investigar y seleccionar las tecnologías y sensores más adecuados para monitorear las necesidades hídricas del área.

- **Resultado:** Se seleccionaron los siguientes sensores para el monitoreo en tiempo real:
 - **DS18B20:** Sensor de temperatura que permitió medir la temperatura ambiente en tiempo real.
 - **BME680:** Sensor de calidad del aire y humedad, utilizado para monitorear la humedad del suelo de manera precisa.
 - **Sensor de flujo de agua:** Para medir la cantidad de agua utilizada por el sistema de riego.

- **Evaluación:** La elección de estos sensores fue adecuada, ya que proporcionaron lecturas confiables y permitieron una integración fluida con el microcontrolador ESP32 LoRa para la transmisión de datos.

Ilustración 67. Selección e investigación



Ilustración 67. Selección e investigación

Nota: imagen de la selección e investigación.

Fuente: Elaboración propia.

Objetivo Específico 3: Diseñar un sistema de riego automatizado utilizando microcontroladores ESP32 con tecnología LoRa, sensores y actuadores para su óptimo funcionamiento.

- **Resultado:** El diseño del sistema implicó la integración de microcontroladores ESP32 LoRa, que se encargaron de la recolección y transmisión de datos. Los sensores de humedad y temperatura se conectaron al ESP32 para monitorear en tiempo real las condiciones del suelo y activar las válvulas del sistema de riego en función de los valores de humedad.
- **Evaluación:** El sistema de control basado en ESP32 LoRa funcionó correctamente, permitiendo la automatización del riego según los parámetros de humedad previamente establecidos. La tecnología LoRa garantizó una comunicación eficiente y de largo alcance entre los sensores y el servidor central.

Ilustración 68. Diseño del sistema

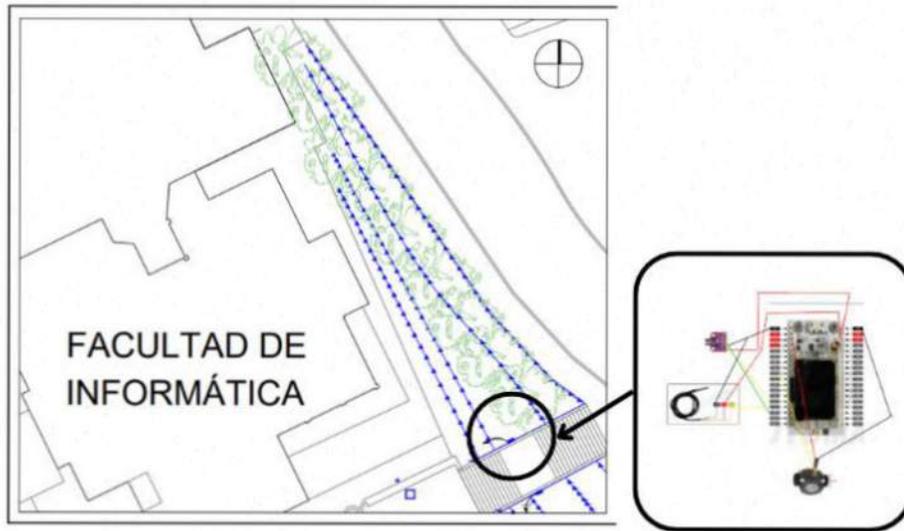


Ilustración 68. Diseño del sistema

Nota: imagen del modelo conceptual de la base de datos

Fuente: Elaboración propia.

Objetivo Específico 4: Programar el ESP32 para gestionar los datos de los sensores mediante la tecnología LoRa, y configurar el Raspberry Pi para procesar y almacenar esta información, garantizando la supervisión remota y eficiente del sistema de riego.

- **Resultado:** Se programó el ESP32 para gestionar los datos de los sensores mediante la tecnología LoRa, y se configuró el Raspberry Pi para procesar y almacenar esta información. Esto permitió establecer una comunicación eficiente entre los dispositivos, asegurando la supervisión remota y precisa del sistema de riego automatizado.
- **Evaluación:** La programación del ESP32 y la configuración del Raspberry Pi fueron exitosas, logrando una comunicación eficiente mediante la tecnología LoRa. Los datos de los sensores se gestionaron y procesaron correctamente, permitiendo un almacenamiento confiable y una supervisión remota precisa del sistema de riego automatizado.

Ilustración 69. Programación del ESP32 para gestionar los datos de los sensores mediante la tecnología LoRa, y configurar el Raspberry Pi para procesar y almacenar esta información, garantizando la supervisión remota y eficiente del sistema de riego.

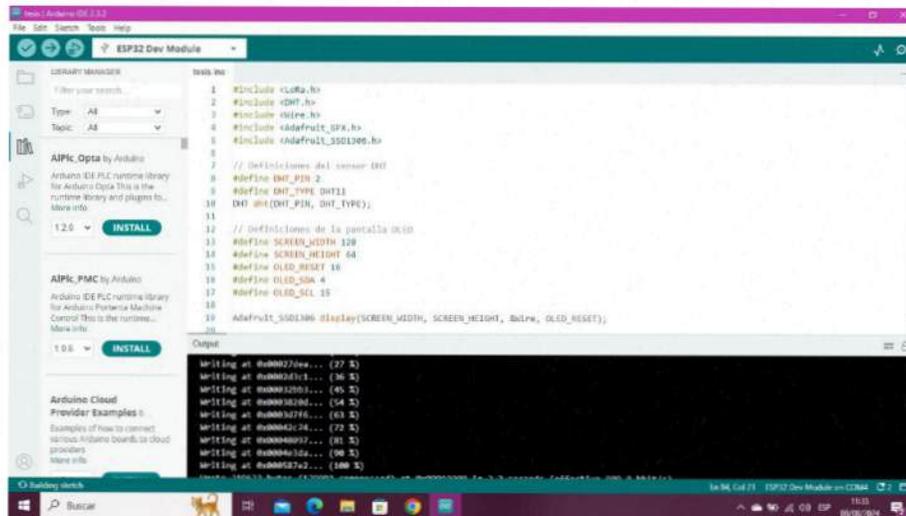


Ilustración 69. Programación del ESP32

Nota: imagen de la programación realizada

Fuente: Elaboración propia.

Objetivo Específico 5: Implementar y ensamblar los componentes del sistema, incluyendo las partes eléctricas y electrónicas, para garantizar el monitoreo preciso de las condiciones de riego.

- **Resultado:** Se llevó a cabo la instalación y ensamblaje de todos los componentes electrónicos, incluyendo los sensores, microcontroladores, válvulas y actuadores. Además, se realizó la configuración de la red de comunicación, utilizando conexiones Ethernet y el protocolo MQTT para la transmisión de datos.
- **Evaluación:** El ensamblaje fue exitoso, y todos los componentes se integraron de manera eficiente, lo que permitió un monitoreo preciso de las condiciones de riego. Las válvulas de control de agua se activaron correctamente de acuerdo con las lecturas de los sensores.

Ilustración 70. Implementación y ensamblar componentes del sistema



Ilustración 70. Implementación y ensamblar componentes del sistema

Nota: imagen de la implementación y ensamblar.

Fuente: Elaboración propia.

Ilustración 71. Implementación y ensamblar componentes del sistema 2

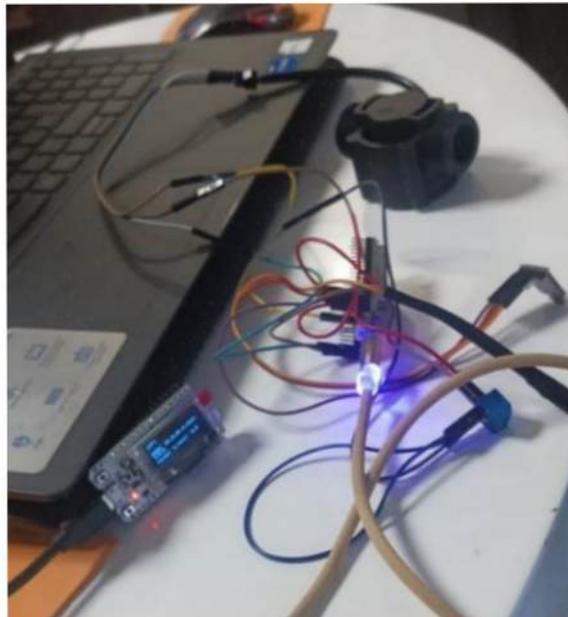


Ilustración 71. Implementación y ensamblar componentes del sistema2

Nota: imagen de la implementación y ensamblar componentes del sistema 2.

Fuente: Elaboración propia.

Ilustración 72. *Implementación y ensamblar componentes del sistema 3*



Ilustración 72. Implementación y ensamblar componentes del sistema 3

Nota: imagen sobre la implementación y ensamblar componentes del sistema 3.

Fuente: Elaboración propia.

Objetivo Específico 6: Realizar pruebas de funcionamiento del sistema de riego, asegurando la conectividad a internet y el monitoreo preciso de los niveles de humedad del suelo.

- **Resultado:** Se realizaron pruebas exhaustivas de conectividad y funcionalidad del sistema, que incluyeron la verificación de la conexión de los dispositivos a la red LAN y la

estabilidad de la comunicación entre los sensores, el ESP32 LoRa y el servidor Raspberry Pi. También se probaron los mecanismos de control de las válvulas y el sistema de riego.

- **Evaluación:** Las pruebas confirmaron que el sistema era funcional y estable, con una correcta transmisión de datos a través de la red. Las válvulas de riego se activaron y desactivaron según los valores de humedad predefinidos, y la visualización de datos en el dashboard fue precisa y en tiempo real.

Ilustración 73. Prueba de funcionamiento



Ilustración 73. Prueba de funcionamiento

Nota: imagen sobre prueba de funcionamiento

Fuente: Elaboración propia.

Ilustración 74. Prueba de funcionamiento del lugar asignado



Ilustración 74. Prueba de funcionamiento del lugar asignado

Nota: imagen sobre prueba de funcionamiento del lugar asignado

Fuente: Elaboración propia.

Ilustración 75. Prueba de funcionamiento del lugar asignado 2.



Ilustración 75. Prueba de funcionamiento del lugar asignado 2

Nota: imagen sobre la prueba de funcionamiento del lugar asignado 2.

Fuente: Elaboración propia.

5.2.3 Selección de sensores y librerías

En primer lugar, se seleccionaron los sensores adecuados para medir los parámetros críticos del entorno en el sistema de riego destinado al cultivo de mangos y vetiver. Para medir la temperatura, humedad y presión atmosférica, se eligió el sensor BME680, debido a su capacidad para proporcionar mediciones precisas de estos parámetros, todo en un solo dispositivo. Esta elección permitió optimizar los recursos y garantizar la fiabilidad de los datos recolectados. Para monitorear la temperatura del agua, se eligió el sensor DS18B20, conocido por su durabilidad y precisión en entornos húmedos. Asimismo, se implementó un sensor de flujo de agua para calcular tanto la tasa como el volumen total de agua utilizada, lo que permite una gestión eficiente de los recursos hídricos. Se importaron las librerías requeridas para interactuar con estos sensores:

Ilustración 76. Librerías necesarias

```
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BME680.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <LoRa.h>
```

Ilustración 76. Librerías necesarias

Nota: imagen sobre las librerías necesarias

Fuente: Elaboración propia.

5.2.4 Establecimiento de sensores

Se establecieron los pines que permitieron la conexión de los sensores en el ESP32-LORA:

Ilustración 77. Definición de sensores

```
#define I2C_SDA 4
#define I2C_SCL 15
#define ONE_WIRE_BUS 2
#define FLOW_SENSOR_PIN 22

Adafruit_BME680 bme;
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);

volatile int pulseCount = 0;
float flowRate;
float flowMilliLitres;
unsigned long totalMilliLitres = 0;
unsigned long oldTime = 0;
```

Ilustración 77. Definición de sensores

Nota: imagen de la definición de sensores.

Fuente: Elaboración propia.

5.2.5 Configuración Lora

Fue imprescindible establecer la conexión LoRa para facilitar la transferencia de los datos obtenidos a un servidor a distancia. La sección siguiente del código tiene como objetivo inicializar el módulo LoRa y establecer la comunicación con el servidor:

Ilustración 78. Configuración lora

```
#define SS 18
#define RST 14
#define DI0 26
#define BAND 868E6

void setupLoRa() {
  Serial.println("Inicializando LoRa...");
  SPI.begin(5, 19, 27, SS);
  LoRa.setPins(SS, RST, DI0);
  if (!LoRa.begin(BAND)) {
    Serial.println("Error al inicializar LoRa. Verifique la configuración.");
    while (1);
  }
  Serial.println("LoRa inicializado correctamente.");
}

void sendDataToServer(String message) {
  Serial.println("Enviando datos al servidor...");
  LoRa.beginPacket();
  LoRa.print(message);
  LoRa.endPacket();
  Serial.println("Datos enviados: " + message);
}
```

Ilustración 78. Configuración lora

Nota: imagen sobre las configuraciones lora

Fuente: Elaboración propia.

5.2.6 Inicialización y Lectura de Sensores

En la función setup(), los sensores se inicializan y se configura la comunicación con el módulo LoRa para la transmisión de datos recolectados. En la función loop(), se realizan las lecturas periódicas de los sensores y se calculan los valores medidos:

Ilustración 79. Inicio y lectura

```
void setup() {  
  Serial.begin(115200);  
  
  sensors.begin();  
  if (!bme.begin(0x76) && !bme.begin(0x77)) {  
    Serial.println("Error al inicializar el sensor BME680.");  
    while (1);  
  }  
  
  setupLoRa();  
}  
  
void loop() {  
  unsigned long endTime = bme.beginReading();  
  if (endTime == 0) {  
    Serial.println("Error al comenzar la lectura del BME680.");  
    return;  
  }  
  delay(endTime - millis());  
  if (!bme.endReading()) {  
    Serial.println("Error al completar la lectura del BME680.");  
    return;  
  }  
  float humidity = bme.humidity;  
  float pressure = bme.pressure / 100.0;  
  
  sensors.requestTemperatures();  
  float temperatureDS18B20 = sensors.getTempCByIndex(0);  
  
  unsigned long currentTime = millis();  
  unsigned long elapsedTime = currentTime - oldTime;  
  if (elapsedTime > 1000) {  
    detachInterrupt(digitalPinToInterrupt(FLOW_SENSOR_PIN));  
    flowRate = ((1000.0 / elapsedTime) * pulseCount) / 7.5;  
    flowMillilitres = (flowRate / 60) * 1000;  
    totalMillilitres += flowMillilitres;  
    pulseCount = 0;  
    oldTime = currentTime;  
    attachInterrupt(digitalPinToInterrupt(FLOW_SENSOR_PIN), pulseCounter, FALLING);  
  }  
  
  String message = "Temp:" + String(temperatureDS18B20, 1) +  
    ", Hum:" + String(humidity, 1) +  
    ", Pres:" + String(pressure, 1) +  
    ", Flow:" + String(flowRate, 1) +  
    ", Total:" + String(totalMillilitres);  
  sendDataToServer(message);  
  
  delay(2000);  
}
```

Ilustración 79. Inicio y lectura

Nota: imagen sobre el inicio y lectura

Fuente: Elaboración propia.

5.2.7 Transferencia de información al servidor

La información recolectada de los sensores se transmite a un servidor a distancia mediante comunicación LoRa. La función `sendDataToServer` tiene como tarea estructurar los datos y transmitirlos al servidor mediante el protocolo LoRa:

Ilustración 80. Envío de datos

```
void sendDataToServer(String message) {  
    Serial.println("Enviando datos al servidor...");  
    LoRa.beginPacket();  
    LoRa.print(message);  
    LoRa.endPacket();  
  
    Serial.println("Datos enviados: " + message);  
}
```

Ilustración 80. Envío de datos

Nota: imagen sobre envió de datos

Fuente: Elaboración propia.

Las evaluaciones llevadas a cabo se basan en las pautas y sugerencias ofrecidas para el microcontrolador ESP32 LoRa, empleado en el sistema de riego automatizado. Durante estas pruebas, se verificó la interacción del ESP32 LoRa con los sensores BME680, DS18B20 y el sensor de flujo de agua, asegurando una comunicación estable y precisa con cada componente. También se realizó la calibración de las lecturas obtenidas para garantizar la exactitud de los valores recolectados, fundamentales para el monitoreo eficiente del sistema.

La transmisión de los datos recolectados se llevó a cabo hacia un servidor basado en una Raspberry Pi conectada a otro ESP32 LoRa, que actúa como receptor. Este proceso permitió evaluar la confiabilidad y eficiencia del protocolo LoRa en la transferencia de datos, incluso en entornos con baja conectividad WiFi. Estas pruebas confirmaron la funcionalidad integral del sistema y su capacidad para operar de manera eficiente en las condiciones específicas del riego automatizado de cultivos como mangos y vetiver.

A continuación, se describen las pruebas efectuadas, diseñadas para asegurar la confiabilidad y exactitud del sistema en general:

5.2.8 Comprobación del enlace y la comunicación

Para realizar la primera prueba y conseguir resultados coherentes en la comprobación de la comunicación entre el ESP32 LoRa, los sensores vinculados y la pantalla OLED, se empleó un software sencillo creado en Arduino IDE.

Esto nos permitió obtener los siguientes resultados:

- La conexión del microcontrolador ESP32 LoRa con los sensores (BME680, DS18B20 y la pantalla OLED) se verificó como funcional, lo que permitió visualizar los datos recolectados en tiempo real a través del Monitor Serial y la pantalla.
- Se verificó que los sensores estuvieran correctamente configurados y listos para enviar los datos al ESP32 LoRa, asegurando que las lecturas, como la temperatura, fueran consistentes y precisas. Esto permitió garantizar que los datos recolectados pudieran ser procesados adecuadamente y transmitidos al servidor remoto a través de LoRa sin errores.

Ilustración 81. Comprobación de conexión y comunicación entre el ESP32 y los sensores.

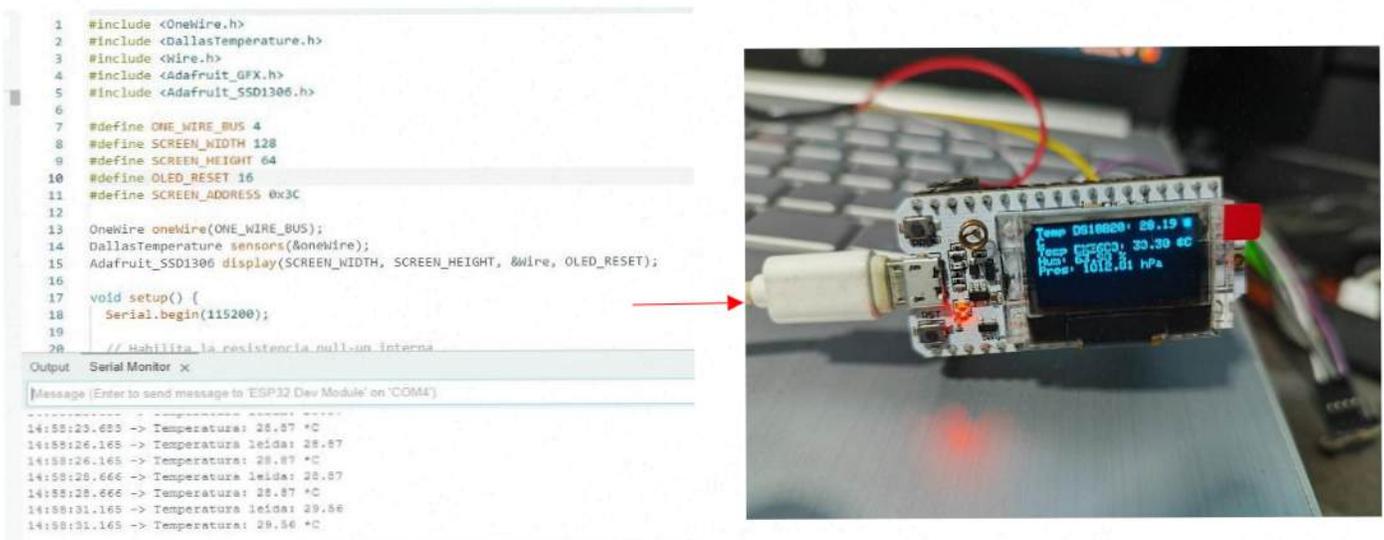


Ilustración 81. Comprobación de conexión y comunicación entre el ESP32 y los sensores.

Nota: imagen sobre la prueba de verificación de conexión y comunicación ESP32 y sensores.

Fuente: Elaboración propia.

5.2.9 Pruebas de lectura de datos

Una vez comprobados los primeros aspectos de configuración, se procedió a la segunda prueba, que abarcó la verificación de la lectura de los datos enviados por los sensores al microcontrolador ESP32 LoRa.

Para esta prueba, se desarrolló un programa que lee continuamente los valores obtenidos de los sensores (BME680, DS18B20 y el sensor de flujo de agua) y los muestra en el Monitor Serial del IDE de Arduino. Esta implementación permitió comprobar la funcionalidad de cada sensor y verificar la recepción correcta de los datos por parte del ESP32 LoRa.

Los resultados obtenidos fueron los siguientes:

- En el Monitor Serial se mostraron mediciones de temperatura ambiental y del agua, humedad relativa y presión atmosférica. Los valores observados fueron consistentes, indicando que los sensores están enviando datos válidos al ESP32 LoRa.
- Adicionalmente, se verificó que las mediciones relacionadas con el flujo de agua (tasa de flujo y volumen total) se procesan correctamente, aunque en la prueba inicial no se detectaron pulsos debido a la ausencia de flujo real en ese momento.

Estas observaciones confirmaron que los sensores están operativos y que el ESP32 LoRa es capaz de procesar y mostrar los datos de manera continua y precisa, lo que garantiza la correcta funcionalidad de esta etapa del sistema de riego automatizado.

Ilustración 82. Examen de lectura de información de los sensores y su procesamiento por el ESP32 -LORA para su representación visual

```
11:58:18.152 -> Pres: 1008.30 hPa
11:00:20.002 -> Tasa de flujo (L/min): 0.00
11:58:20.887 -> Flujo instantáneo (mL): 0.00
11:58:20.887 -> Flujo total (mL): 0
11:58:20.887 ->
11:00:20.007 -> Temperatura DS10D20: 20.75 C
11:58:20.887 -> Hum: 56.66 %
11:58:20.887 -> Pres: 1008.32 hPa
11:58:23.611 -> Tasa de flujo (L/min): 0.00
11:00:20.611 -> Flujo instantáneo (mL): 0.00
11:58:23.611 -> Flujo total (mL): 0
11:58:23.611 ->
11:58:23.611 -> Temperatura DS18B20: 28.81 C
11:00:20.707 -> Hum: 56.70 %
11:58:23.707 -> Pres: 1008.32 hPa
11:58:26.383 -> Tasa de flujo (L/min): 0.00
11:58:26.383 -> Flujo instantáneo (mL): 0.00
11:00:26.303 -> Flujo total (mL): 0
11:58:26.383 ->
11:58:26.383 -> Temperatura DS18A20: 28.75 C
11:58:26.383 -> Hum: 56.57 %
11:00:26.303 -> Pres: 1008.30 hPa
```

Ilustración 82. Examen de lectura de información de los sensores y su procesamiento por el ESP32 -LORA para su representación visual

Nota: imagen sobre examen de lectura de información de los sensores y su procesamiento por el ESP32 -LORA para su representación visual

Fuente: Elaboración propia.

5.2.10 Transmisión y documentación de datos

Es esencial establecer un sistema de archivo de datos para recolectar datos a largo plazo y conservar un registro que facilite la evaluación de tendencias y patrones en las mediciones efectuadas por los sensores.

Se llevó a cabo un examen de comprobación para verificar el funcionamiento adecuado del sistema de envío de datos al servidor y su registro en la base de datos MariaDB, además de su visualización en Node-RED.

Los resultados obtenidos en esta prueba fueron los siguientes:

- Se verificó que los datos procesados por el ESP32 LoRa están siendo enviados al servidor correctamente mediante comunicación LoRa. Esta transmisión fue confirmada por la recepción exitosa de los datos en el servidor, visualizándose de manera inmediata en el Monitor Serial del IDE de Arduino.

- En Node-RED, se observó la representación de los datos en tiempo real, permitiendo una visualización clara y organizada de las mediciones obtenidas, como la temperatura, humedad, presión y flujo de agua.
- Adicionalmente, se comprobó que los datos están siendo almacenados de forma continua y consistente en la base de datos MariaDB alojada en el servidor. Esto garantiza la integridad de la información y su disponibilidad para análisis posteriores.

La integración del envío de datos al servidor, su representación en Node-RED y el almacenamiento en MariaDB validaron la funcionalidad completa del sistema, demostrando su capacidad para registrar y visualizar datos de manera confiable y eficiente.

Ilustración 83. Comprobación de respuesta en Node-Red



Ilustración 83. Comprobación de respuesta en Node-Red

Nota: imagen sobre la comprobación de respuesta en Node-Red

Fuente: Elaboración propia.

Ilustración 84. Comprobación en la base de datos en MariaDB

id	device_id	temperature	humidity	pressure	flow_rate	total_milliliters	timestamp
95300	ESP1	37.8	30.3	1001.4	16.4	560997	2024-12-09 15:58:17
95301	ESP1	39.5	32.4	1001.9	0	0	2024-12-09 15:58:19
95302	ESP1	37.9	30.2	1001.4	15.9	561262	2024-12-09 15:58:19
95303	ESP1	39.6	32.4	1001.9	0	0	2024-12-09 15:58:21
95304	ESP1	37.8	30.2	1001.4	15.4	561518	2024-12-09 15:58:23
95305	ESP1	39.6	32.4	1001.9	0	0	2024-12-09 15:58:24
95306	ESP1	37.8	30.2	1001.4	15	561767	2024-12-09 15:58:25
95307	ESP1	39.6	32.4	1001.9	0	0	2024-12-09 15:58:26
95308	ESP1	37.8	30.2	1001.4	14.7	562011	2024-12-09 15:58:28
95309	ESP1	39.6	32.4	1002	0	0	2024-12-09 15:58:29
95370	ESP1	37.9	30.2	1001.4	14.1	562246	2024-12-09 15:58:30
95371	ESP1	39.6	32.4	1002	0	0	2024-12-09 15:58:32
95372	ESP1	37.8	30.2	1001.4	13.6	562472	2024-12-09 15:58:33
95373	ESP1	39.6	32.4	1002	0	0	2024-12-09 15:58:35
95374	ESP1	37.8	30.2	1001.4	13.5	562696	2024-12-09 15:58:36
95375	ESP1	39.7	32.4	1002	0	0	2024-12-09 15:58:38
95376	ESP1	37.9	30.2	1001.4	13.1	562914	2024-12-09 15:58:39
95377	ESP1	39.6	32.4	1002	0	0	2024-12-09 15:58:40
95378	ESP1	37.8	30.2	1001.4	12.8	563126	2024-12-09 15:58:42
95379	ESP1	39.6	32.4	1002	0	0	2024-12-09 15:58:43
95380	ESP1	37.8	30.2	1001.4	12.6	563336	2024-12-09 15:58:44
95381	ESP1	39.7	32.4	1002	0	0	2024-12-09 15:58:46
95382	ESP1	37.9	30.2	1001.4	12.4	563543	2024-12-09 15:58:47

Ilustración 84. Comprobación en la base de datos en MariaDB

Nota: imagen sobre la Comprobación en la base de datos en MariaDB

Fuente: Elaboración propia.

5.2.11 Prueba de integración con la plataforma de node-red

En este apartado se documenta la integración del sistema de riego automatizado con la plataforma Node-RED, utilizada para gestionar y visualizar los datos obtenidos de los sensores conectados al sistema. La prueba tuvo como objetivo validar que la comunicación y el envío de datos funcionan correctamente en un entorno de red local (LAN), garantizando la interacción eficiente entre los componentes del sistema.

Resultados obtenidos de esta prueba son:

- **Recepción y procesamiento de datos:** Se verificó que los datos capturados por los sensores, como los niveles de humedad del suelo, temperatura y flujo de agua, fueron enviados desde el ESP32 LoRa y correctamente procesados por los flujos configurados en Node-RED.
- **Visualización de estado del riego:** En los dashboards de Node-RED se pudo visualizar un mensaje claro e intuitivo que indica si el sistema de riego está activado ("Electroválvula funcionando") y cuando está desactivado ("No hay flujo de agua"). Esto permitió al usuario conocer el estado del sistema en tiempo real.
- **Control remoto dentro de la LAN:** Se validó que los comandos enviados desde la interfaz de Node-RED, como la activación o desactivación del riego, se ejecutaron con precisión y mostraron el estado actualizado del sistema en la interfaz gráfica.
- **Limitación de acceso:** Durante las pruebas, se confirmó que el sistema opera exclusivamente dentro de la red local (LAN), lo cual garantiza la seguridad de los datos y del control del sistema, pero limita su alcance para usuarios fuera de dicha red.

Este apartado demuestra que la integración con Node-RED en un entorno de red local cumple con los requisitos de monitoreo y control planteados en los objetivos del sistema. Aunque la funcionalidad está restringida a la red LAN, el sistema ofrece una solución confiable y práctica para la gestión del riego automatizado.

Ilustración 85. Datos de temperatura



Ilustración 85. Datos de temperatura

Nota: imagen sobre los datos de temperatura

Fuente: Elaboración propia.

Ilustración 86. Datos de humedad

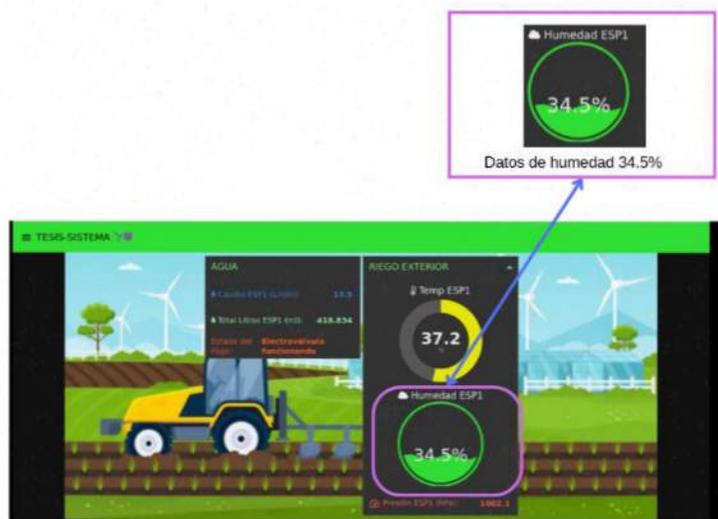


Ilustración 86. Datos de humedad

Nota: imagen sobre los datos de la humedad

Fuente: Elaboración propia.

Ilustración 87. Datos presión

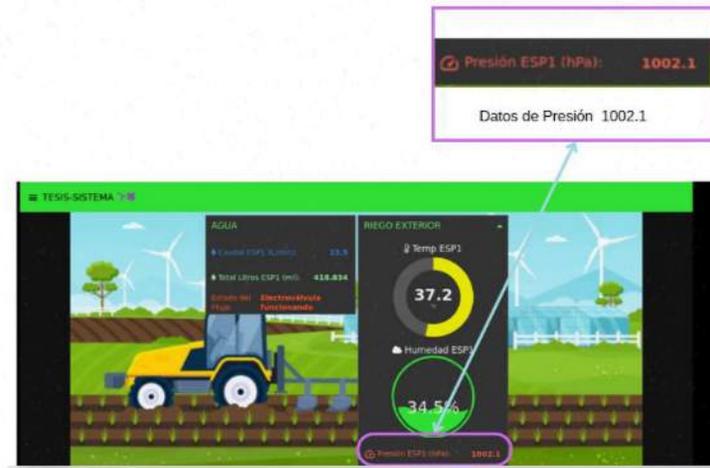


Ilustración 87. Datos presión

Nota: imagen sobre los datos de presión.

Fuente: Elaboración propia.

Ilustración 88. Datos del caudal

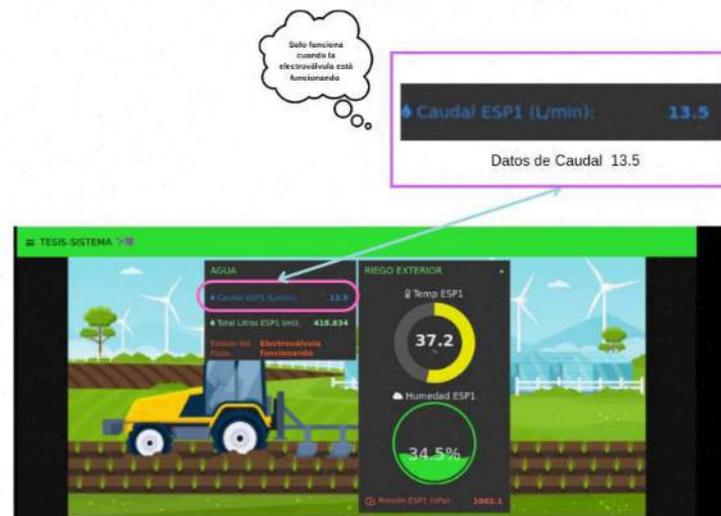


Ilustración 88. Datos del caudal

Nota: imagen sobre la comprobación de respuesta en Node-Red

Fuente: Elaboración propia.

Ilustración 89. Total, de litros



Ilustración 89. Total, de litros

Nota: imagen sobre el total de litros

Fuente: Elaboración propia.

Ilustración 90. Estado del flujo tenemos los dos casos encendido y apagado



Ilustración 90. Estado del flujo tenemos los dos casos encendido y apagado

Nota: imagen sobre la comprobación de respuesta en Node-Red

Fuente: Elaboración propia.

5.2.12 Pruebas de Estabilidad y Robustez

En esta sección se documentan los resultados obtenidos al evaluar la estabilidad y robustez del sistema de riego automatizado bajo diferentes condiciones operativas. Estas pruebas se realizaron durante el mes de diciembre de 2024, con el propósito de garantizar que el sistema funcione de manera confiable y consistente en un entorno controlado.

5.2.12.1 Procedimiento y Resultados

El sistema fue sometido a pruebas que incluyeron la recopilación y almacenamiento de datos esenciales en la base de datos MariaDB. Estos datos incluyen humedad, temperatura, presión, caudal, el total de litros suministrados, junto con la fecha y hora correspondientes.

5.3 Interpretación objetiva

Los resultados obtenidos demuestran que el funcionamiento de cada componente del sistema de riego automatizado es correcto y cumple con los objetivos planteados. Este sistema, diseñado específicamente para optimizar el riego en los exteriores del bloque de Informática de la Universidad Laica Eloy Alfaro de Manabí, ha sido evaluado bajo distintos escenarios operativos. Las pruebas realizadas se enfocaron en la verificación de la conexión y comunicación entre los sensores y el microcontrolador ESP32, el envío de datos recopilados por los sensores, y la integración con la plataforma basada en Node-RED.

Adicionalmente, se ha valorado el sistema basándose en su funcionalidad y desempeño. La plataforma utilizada en la gestión del sistema simplifica la recolección, análisis y exhibición de datos en tiempo real, mejorando de esta manera la precisión y eficiencia en el monitoreo de las condiciones ambientales del área de irrigación. La habilidad de la plataforma para acoplarse a varios sensores y dispositivos garantiza su adaptabilidad y escalabilidad, lo que la convierte en idónea para su uso en diversos contextos, ya sea en proyectos de educación.

Además, la implementación del sistema en un entorno educativo como los exteriores del bloque de Informática ofrece una herramienta práctica y tangible para los estudiantes de tecnología. Este proyecto no solo contribuye a la formación académica, sino que también sirve como un modelo demostrativo para aplicaciones futuras en el ámbito agrícola, evidenciando cómo

la tecnología puede ser aplicada de manera efectiva para resolver problemas cotidianos como el manejo eficiente del agua.

En conclusión, el sistema de riego automatizado ha mostrado ser una solución robusta, eficiente y adaptable, capaz de cumplir con los requerimientos planteados tanto a nivel técnico como práctico, lo que respalda su utilidad para su implementación en escenarios reales y educativos.

CAPÍTULO VI

6. Conclusiones y Recomendaciones

6.1 Conclusiones

En este capítulo se presentan las conclusiones obtenidas a partir de la implementación y evaluación del sistema de riego automatizado diseñado para optimizar el riego en los exteriores del bloque de Informática de la Universidad Laica Eloy Alfaro de Manabí.

El sistema de riego automatizado implementado ha logrado regar únicamente cuando el suelo lo requiere. Gracias a los sensores de humedad, el sistema activa el riego únicamente si los niveles de humedad del suelo son bajos, lo que contribuye a evitar el desperdicio de agua y garantiza que las plantas reciban la cantidad precisa de agua necesaria para su crecimiento.

El uso de tecnología LoRa con el microcontrolador ESP32 permitió que los sensores de humedad transmitieran datos de forma eficiente a larga distancia sin necesidad de cables. Esto fue crucial para cubrir toda el área de riego sin problemas de conectividad.

El ESP32 fue programado para recolectar los datos de los sensores y decidir cuándo activar o desactivar el sistema de riego, según los niveles de humedad. Este control automático hizo que el sistema funcionara sin necesidad de intervención manual, lo que lo hace más conveniente y eficiente.

Se configuró un Raspberry Pi para recibir y almacenar los datos del sistema, lo que permitió monitorear los niveles de humedad y el estado del riego desde cualquier lugar a través de internet. Esto facilita la gestión del sistema de riego sin necesidad de estar físicamente en el lugar.

Durante las pruebas, se comprobó que el sistema de riego funcionaba correctamente. Los sensores transmitieron datos sin interrupciones, y el sistema de riego activó y desactivó las válvulas correctamente según los niveles de humedad del suelo. Todo funcionó de acuerdo con lo esperado.

El sistema demostró ser confiable y capaz de operar de manera continua sin fallos. Esto significa que se puede implementar en otras áreas del campus para ahorrar agua y mejorar el cuidado de las plantas en todo el lugar.

El diseño y la implementación del sistema de riego automatizado fueron ejecutados de forma eficiente, integrando sensores, actuadores y microcontroladores para garantizar un funcionamiento óptimo. Las pruebas realizadas confirmaron la estabilidad y fiabilidad del sistema, el cual es capaz de operar de manera autónoma, adaptándose a las condiciones variables del entorno.

Las pruebas de funcionamiento realizadas al sistema demostraron que este es capaz de mantener un control adecuado del riego, basándose en las condiciones de humedad del suelo. Se verificó la estabilidad del sistema bajo diversas condiciones operativas, asegurando su correcto desempeño a lo largo del tiempo y la correcta conectividad a la plataforma IoT.

Para asegurar un acceso constante al *dashboard* del sistema de riego, se asignó una IP fija a los dispositivos conectados por cable Ethernet. Esto evitó que las direcciones IP cambiaran y permitió un acceso estable y seguro al sistema. Al usar una IP estática solo para los dispositivos esenciales, como el servidor del *dashboard*, se mejoró la conectividad y facilitó la gestión del sistema, asegurando que los responsables pudieran monitorear el riego sin interrupciones.

6.2 Recomendaciones

A continuación, se presentan una serie de recomendaciones basadas en los hallazgos de la investigación y los resultados obtenidos durante la implementación y pruebas del sistema de riego automatizado.

Optimización del Consumo Energético:

- Aunque el sistema está funcionando correctamente, se recomienda investigar soluciones energéticas alternativas, como la integración de paneles solares o fuentes de energía renovable. Esto contribuiría a hacer el sistema más sostenible y autónomo, especialmente en entornos rurales o zonas donde el acceso a la energía eléctrica es limitado.

Expansión del Sistema de Sensores:

- Para mejorar la precisión y eficiencia del riego, se recomienda incorporar más sensores, tales como sensores de luz o sensores de calidad del agua. Estos sensores permitirían un control más detallado y personalizado del riego, adaptado a las necesidades específicas de cada tipo de cultivo.

Escalabilidad del Sistema:

- Se sugiere realizar pruebas en otras áreas de la universidad o en diferentes tipos de cultivo para evaluar la escalabilidad del sistema. El sistema ha mostrado un buen desempeño en un entorno controlado, por lo que sería interesante comprobar su efectividad en un rango más amplio de condiciones ambientales.

Mejoras en la Interfaz de Usuario:

- Aunque la interfaz de usuario desarrollada es funcional, se recomienda mejorarla para que sea más intuitiva y dinámica. Se podrían incluir gráficos interactivos que muestren las tendencias de humedad a lo largo del tiempo, así como alertas

personalizadas que notifiquen cualquier anomalía en el funcionamiento del sistema o en los niveles de humedad del suelo.

Monitoreo Remoto en Tiempo Real:

- Para facilitar la gestión del sistema desde cualquier lugar, se recomienda la implementación de una plataforma de monitoreo remoto en tiempo real, a través de una aplicación móvil o una interfaz web. Esto permitiría a los usuarios tener acceso continuo a los datos y, en caso necesario, ajustar el sistema de riego desde cualquier dispositivo conectado.

Bibliografía

- Agronet. (28 de noviembre de 2022). *El vetiver, la planta milagrosa de la agricultura*. Obtenido de <https://www.agronet.gov.co/Noticias/Paginas/El-vetiver,-la-planta-milagrosa-de-la-agricultura.aspx>
- Artículo. (25 de agosto de 2022). *Qué es LoRa, cómo funciona y características principales*. Obtenido de Qué es LoRa, cómo funciona y características principales: <https://www.vencoel.com/que-es-lora-como-funciona-y-caracteristicas-principales/>
- AWS. (08 de noviembre de 2024). *¿Qué es MQTT?* Obtenido de ¿Qué es MQTT?: <https://aws.amazon.com/es/what-is/mqtt/>
- Buerkert. (2024). *¿Qué es una electroválvula y cómo funciona?* Obtenido de <https://www.burkert.es/es/empresa-y-carrera/actualidades/prensa/mediateca/informes-tecnicos/que-es-una-electrovalvula-y-como-funciona#:~:text=Las%20electrov%C3%A1lvulas%20son%20los%20elementos,en%20muchos%20%C3%A1mbitos%20de%20aplicaci%C3%B3n.>
- Carrizo, V., & Manuel, J. (2019). Obtenido de <https://ebuah.uah.es/dspace/handle/10017/35420#:~:text=Se%20trata%20de%20un%20microcontrolador,con%20internet%20u%20otros%20dispositivos.>
- Domodesk, *Domótica, IoT, WIFI6*. (30 de agosto de 2019). Obtenido de Monthly Archives: <https://www.domodesk.com/blog/wp-content/uploads/2019/08/wifi6domotica2-300x200.jpg>
- Eclipse Mosquito*. (2024). Obtenido de <https://pbs.twimg.com/media/CgZ3XzbWwAACVae.jpg>
- Ecuador, N. (1 de agosto de 2022). *La evolución de la técnica del riego en el tiempo*. Obtenido de <https://es.linkedin.com/pulse/la-evoluci%C3%B3n-de-t%C3%A9cnica-del-riego-en-el-tiempo-netafim-ecuador>

- Editorial Prensa Alicantina, S. (1 de septiembre de 2022). *La forma más fácil de cultivar un mango en una maceta*. Obtenido de https://estaticos-cdn.prensaiberica.es/clip/22feec76-cc68-4873-9a29-aa75d5a91c69_16-9-discover-aspect-ratio_default_0_x485y899.webp
- Eicos. (2024). *¿Qué es un Sensor de Flujo?* Obtenido de <https://www.eicos.com/datos-tecnicos/que-es-un-sensor-de-flujo/>
- Electrónicas, D. (08 de agosto de 2024). *Sensor de humedad, temperatura, presión y gas. BME680*. Obtenido de https://didacticaselectronicas.com/index.php/view/productdetails/virtuemart_product_id/13675/virtuemart_category_id/42
- electrovalvula*. (9 de diciembre de 2024). Obtenido de <https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcS4icekcgCsnUpfWL55SmO-5-ACtp69RkDsUkE9m0mJcUYlQAtp>
- Enciso, J. M., Porter, D., & Périès, X. (2007). *Uso de sensores de humedad del suelo para eficientizar el riego*. Obtenido de <https://oaktrust.library.tamu.edu/server/api/core/bitstreams/7892b0fc-93e6-489d-8bc6-c13303a87c2b/content>
- Estévez, C. (4 de julio de 2019). *El vetiver: una olorosa planta que crece en las orillas de los ríos*. Obtenido de <https://abanuc.com/blogs/aromas-de-autor/el-vetiver-la-planta-que-crece-en-las-orillas-de-los-rios>
- Ferrovial. (2024). *Fuente de alimentación*. Obtenido de <https://www.ferrovial.com/es/stem/fuente-de-alimentacion/#:~:text=%C2%BFQu%C3%A9%20se%20entiende%20por%20fuente,un%20dispositivo%20o%20sistema%20electr%C3%B3nico.>
- fuelle de poder*. (9 de diciembre de 2024). Obtenido de https://daenet.wordpress.com/wp-content/uploads/2009/07/fuelle_poder.jpeg?w=584
- García, D. D. (9 de enero de 2025). *Mango*. Obtenido de <https://www.lechepuleva.es/nutricion-y-bienestar/el-mango>

- Gómez, R. F., Milla, M. M., Alabarces, R. Á., Herrera, J. B., Zafra, P. G., & Gutiérrez, N. A. (2010). *juntadeandalucia*. Obtenido de *juntadeandalucia*: https://www.juntadeandalucia.es/export/drupaljda/1337160940Riego_por_superficie_baja.pdf
- Grunbaum, S. (03 de Octubre de 2023). *TIPOS DE INVESTIGACION TODOS - EXPLICADOS DETALLADAMENTE - Diseños - Dr. Sergio Grunbaum*. Obtenido de TIPOS DE INVESTIGACION TODOS - EXPLICADOS DETALLADAMENTE - Diseños - Dr. Sergio Grunbaum: <https://publicacoes.even3.com.br/preprint/tipos-de-investigacion-todos--explicados-detalladamente-dr-sergio-grunbaum-3181867>
- heltec. (08 de noviembre de 2024). *Wi-Fi LoRa 32(V2)*. Obtenido de *Wi-Fi LoRa 32(V2)*: <https://heltec.org/project/wifi-lora-32v2/>
- Hodges, B., Tagert, M. L., & Paz, J. O. (11 de julio de 2022). *Uso de un modelo de cultivo y sensores de humedad del suelo para estimar la humedad del suelo y las aplicaciones de riego en un campo de producción de soja*. Obtenido de <https://doi.org/10.1007/s00271-022-00802-1>
- Innovatione. (8 de julio de 2019). *SISTEMAS DE RIEGO I*. Obtenido de <https://innovatione.eu/2019/07/08/riego-agricultura/>
- Lorawan-lora*. (9 de diciembre de 2024). Obtenido de <https://www.vencoel.com/wp-content/uploads/2022/08/Capa-fisica-y-MAC-de-LoRa.png>
- mar, M. d. (4 de octubre de 2023). *El riego automatizado en parques y jardines como solución ante la crisis climática*. Obtenido de <https://es.linkedin.com/pulse/el-riego-automatizado-en-parques-y>
- Millán, p. S., Casadesús, J., C. C., Moñino, M. J., & Prieto1, M. H. (2 de octubre de 2019). *Uso de sensores de humedad del suelo para la programación automatizada del riego en un cultivo de ciruelas*. Obtenido de <https://www.mdpi.com/2073-4441/11/10/2061>
- Mosquito. (08 de noviembre de 2024). *Eclipse Mosquito*. Obtenido de <https://mosquito.org/>

- Naylamp. (9 de DICIEMBRE de 2024). *Naylamp Mechatronics*. Obtenido de https://naylampmechatronics.com/2763-superlarge_default/sensor-bme680-presion-temperatura-humedad-y-gas.jpg
- Olmos, L. M. (9 de enero de 2025). *El riego automatizado, principios y herramientas actuales*. Obtenido de Tecnología Hortícola: <https://www.tecnologiahorticola.com/ritec-riego-automatizado/>
- Pérez, J. (25 de Mayo de 2022). *Definicion de*. Obtenido de <https://definicion.de/automatizacion/>
- Proofpoint. (08 de noviembre de 2024). *¿Qué es el wifi?* Obtenido de *¿Qué es el wifi?:* <https://www.proofpoint.com/es/threat-reference/wifi#:~:text=a%20las%20redes.-,%C2%BFQu%C3%A9%20significa%3F,red%20mediante%20frecuencias%20de%20radio.>
- protocolo MQTT*. (2024). Obtenido de <https://i.ytimg.com/vi/urLGRqbaqqo/maxresdefault.jpg>
- raspberry pi*. (9 de diciembre de 2024). Obtenido de https://encrypted-tbn1.gstatic.com/images?q=tbn:ANd9GcQ0MRpTSpXlQk4m-Xfrb7ECUKWZDqRkQtSA-AC3Q5pwO_KCsxev
- Raúl, B. F. (2021). *matematicasfesc*. Obtenido de *matematicasfesc:* <https://matematicasfesc.cuautitlan.unam.mx/Memorias/Congreso15/images/prototipos/Sistema%20de%20riego%20automatico.pptx.pdf>
- S&P. (21 de junio de 2024). *Sensor de temperatura: tipos y normativa*. Obtenido de <https://www.solerpalau.com/es-es/blog/sensor-temperatura/#:~:text=Su%20funci%C3%B3n%20principal%20es%20medir,conocen%20como%20sondas%20de%20temperatura.>
- Salazar, J., & Santiago, S. (s.f.). INTERNET DE LAS COSAS. (*REVISTA CIENTIFICA*). TechPedia, Technicka.
- SDI. (8 de agosto de 2024). *Fuente de poder*. Obtenido de <https://sdindustrial.com.mx/blog/fuentes-de-poder/>

sensor flujo de agua. (9 de diciembre de 2024). Obtenido de https://ventas.ardunel.com.bo/392-large_default/sensor-flujo-de-agua-yf-s201g-1-2-pulg-1-30l-min.jpg

Serigio. (2021). *Mundo de Riego.* Obtenido de Mundo de Riego : <https://mundoriego.es/automatizacion-de-sistemas-de-riego/>

Solé, R. (08 de noviembre de 2024). *Qué es Raspberry Pi.* Obtenido de Qué es Raspberry Pi: <https://www.profesionalreview.com/2021/07/18/que-es-raspberry-pi/>

Sonda De Sensor Interfaz I2c Temperatura Del Suelo Sens... (9 de diciembre de 2024). Obtenido de https://http2.mlstatic.com/D_NQ_NP_690266-MLC76959468120_062024-O.webp

Spectrum, I. (2024). *Uso de sondas y sensores con digitalizadores modulares.* Obtenido de https://spectrum-instrumentation.com/applications/product_notes/PN_Using_Probes_and_Sensors_with_Modular_Digitizers.php

Tagert, M. A., & Paz, J. O. (11 de julio de 2022). *Uso de un modelo de cultivo y sensores de humedad del suelo para estimar la humedad del suelo y las aplicaciones de riego en un campo de producción de soja.* Obtenido de <https://link.springer.com/article/10.1007/s00271-022-00802-1>

UNIT. (2024). *BME680 Sensor de Presión, Temperatura y Humedad.* Obtenido de <https://uelectronics.com/producto/bme680-sensor-de-presion-temperatura-y-humedad/#:~:text=BME680%20Sensor%20es%20un%20peque%C3%B1o,comunicaci%C3%B3n%20por%20I2C%20y%20SPI.>

Urdaneta, E., Pérez, J., & Custodio, Á. (2014). *ve.scielo.* Obtenido de [ve.scielo: https://ve.scielo.org/pdf/uct/v18n70/art02.pdf](https://ve.scielo.org/pdf/uct/v18n70/art02.pdf)

Valdivieso, A. (2024). *iagua.es.* Obtenido de [iagua.es: https://www.iagua.es/respuestas/que-consiste-riego-goteo](https://www.iagua.es/respuestas/que-consiste-riego-goteo)

Zúñiga, P. I. (27 de septiembre de 2023). *Metodología de la investigación científica: guía práctica*.

Obtenido de Metodología de la investigación científica: guía práctica:

<https://ciencialatina.org/index.php/cienciala/article/view/7658>

Anexos

Ilustración 91. Instalación en diversos entornos



Ilustración 91. Instalación en diversos entornos

Nota: imagen sobre la instalación en diversos entornos

Fuente: Elaboración propia.

Ilustración 92. Instalación del riego



Ilustración 92. Instalación del riego

Nota: imagen sobre la instalación del riego

Fuente: Elaboración propia.

Ilustración 93. Instalación del riego y detalles del riego



Ilustración 93. Instalación del riego y detalles del riego

Nota: imagen relacionada con instalación del riego y detalles del riego

Fuente: Elaboración propia.

Ilustración 94. Tablero y servidor del riego



Ilustración 94. Tablero y servidor del riego

Nota: imagen sobre el tablero y servidor del riego

Fuente: Elaboración propia.

Ilustración 95. Datos recogidos visualizados en la base de datos visualización del 4 al 5 de diciembre del 2024

Archivo	Editar	Pestañas	Ayuda
95347	ESP1	39.4	32.4 1992 0 0 2024-12-09 15:57:59
95348	ESP1	37.8	30.2 1991.4 15.8 0 2024-12-09 15:58:00
95349	ESP1	39.5	32.4 1992 0 0 2024-12-09 15:58:01
95350	ESP1	37.9	30.2 1991.4 17.8 559548 2024-12-09 15:58:03
95351	ESP1	39.5	32.4 1992 0 0 2024-12-09 15:58:04
95352	ESP1	37.9	30.2 1991.4 19.1 559426 2024-12-09 15:58:05
95353	ESP1	39.5	32.4 1992 0 0 2024-12-09 15:58:07
95354	ESP1	37.8	30.3 1991.4 19.2 560146 2024-12-09 15:58:14
95355	ESP1	39.5	32.4 1991.9 0 0 2024-12-09 15:58:14
95356	ESP1	39.6	32.4 1991.9 0 0 2024-12-09 15:58:14
95357	ESP1	37.8	30.3 1991.4 17.0 560442 2024-12-09 15:58:14
95358	ESP1	37.8	30.3 1991.4 17 560724 2024-12-09 15:58:14
95359	ESP1	39.5	32.4 1991.9 0 0 2024-12-09 15:58:16
95360	ESP1	37.8	30.3 1991.4 18.4 560997 2024-12-09 15:58:17
95361	ESP1	39.5	32.4 1991.9 0 0 2024-12-09 15:58:19
95362	ESP1	37.9	30.2 1991.4 15.9 561262 2024-12-09 15:58:19
95363	ESP1	39.6	32.4 1991.9 0 0 2024-12-09 15:58:21
95364	ESP1	37.8	30.2 1991.4 15.4 561518 2024-12-09 15:58:23
95365	ESP1	39.6	32.4 1991.9 0 0 2024-12-09 15:58:24
95366	ESP1	37.8	30.2 1991.4 15 561767 2024-12-09 15:58:25
95367	ESP1	39.6	32.4 1991.9 0 0 2024-12-09 15:58:26
95368	ESP1	37.8	30.2 1991.4 14.7 562013 2024-12-09 15:58:28
95369	ESP1	39.6	32.4 1992 0 0 2024-12-09 15:58:28
95370	ESP1	37.9	30.2 1991.4 14.1 562246 2024-12-09 15:58:30
95371	ESP1	39.6	32.4 1992 0 0 2024-12-09 15:58:32
95372	ESP1	37.8	30.2 1991.4 13.6 562472 2024-12-09 15:58:33
95373	ESP1	39.6	32.4 1992 0 0 2024-12-09 15:58:35
95374	ESP1	37.8	30.2 1991.4 13.5 562696 2024-12-09 15:58:36
95375	ESP1	39.7	32.4 1992 0 0 2024-12-09 15:58:38
95376	ESP1	37.9	30.2 1991.4 13.1 562914 2024-12-09 15:58:39
95377	ESP1	39.6	32.4 1992 0 0 2024-12-09 15:58:40
95378	ESP1	37.8	30.2 1991.4 12.8 563126 2024-12-09 15:58:42
95379	ESP1	39.6	32.4 1992 0 0 2024-12-09 15:58:43
95380	ESP1	37.8	30.2 1991.4 12.6 563336 2024-12-09 15:58:44

2804	ESP2	37.4	26.9	1992.1	0	341011	2024-12-11 03:40:50
2805	ESP2	37.4	26.8	1992.1	0	341011	2024-12-11 03:45:50
2806	ESP2	37.4	26.8	1992.1	0	341011	2024-12-11 03:50:50
2807	ESP2	37.4	26.7	1992.1	0	341011	2024-12-11 03:55:50
2808	ESP2	37.4	26.7	1992.1	0	341011	2024-12-11 04:00:50
2809	ESP2	37.4	26.7	1992.1	0	341011	2024-12-11 04:05:50
2810	ESP2	37.4	26.7	1992.1	0	341011	2024-12-11 04:10:50
2811	ESP2	37.4	26.7	1992	0	341011	2024-12-11 04:15:50
2812	ESP2	37.4	26.7	1992	0	341011	2024-12-11 04:20:50
2813	ESP2	37.4	26.7	1992	0	341011	2024-12-11 04:25:50
2814	ESP2	37.4	26.7	1992	0	341011	2024-12-11 04:30:50
2815	ESP2	37.4	26.7	1992.1	0	341011	2024-12-11 04:35:50
2816	ESP2	37.4	26.7	1992	0	341011	2024-12-11 04:40:50
2817	ESP2	37.3	26.7	1992	0	341011	2024-12-11 04:45:50
2818	ESP2	37.3	26.7	1992	0	341011	2024-12-11 04:50:50
2819	ESP2	37.3	26.8	1992	0	341011	2024-12-11 04:55:50
2820	ESP2	37.3	26.7	1992	0	341011	2024-12-11 05:00:50
2821	ESP2	37.3	26.7	1992.1	0	341011	2024-12-11 05:05:50
2822	ESP2	37.3	26.7	1992.1	0	341011	2024-12-11 05:10:50
2823	ESP2	37.3	26.7	1992.1	0	341011	2024-12-11 05:15:50
2824	ESP2	37.3	26.7	1992	0	341011	2024-12-11 05:20:50
2825	ESP2	37.3	26.7	1992	0	341011	2024-12-11 05:30:50
2826	ESP2	37.3	26.7	1992	0	341011	2024-12-11 05:35:50
2827	ESP2	37.4	26.7	1992	0	341011	2024-12-11 05:40:50

Ilustración 95. Datos recogidos visualizados en la base de datos visualización del 4 al 5 de diciembre del 2024

Nota: imagen relacionada con datos recogidos visualizados en la base de datos visualización del 4 al 5 de diciembre del 2024

Fuente: Elaboración propia.

Ilustración 96. Datos recogidos visualizados en la base de datos visualización del 6 al 7 de diciembre del 2024

Archivo	editar	Pestañas	Ayuda						
20723	ESP1	31.4	34.2	1007.7	0	0	2024-12-06	11:00:56	
20724	ESP1	29.8	47.7	1000.3	0	0	2024-12-06	11:00:57	
20725	ESP1	31.3	34.2	1007.7	0	0	2024-12-06	11:00:58	
20726	ESP1	29.8	47.7	1000.3	0	0	2024-12-06	11:01:00	
20727	ESP1	31.4	34.2	1007.7	0	0	2024-12-06	11:01:01	
20728	ESP1	29.8	47.7	1000.3	0	0	2024-12-06	11:01:03	
20729	ESP1	31.4	34.2	1007.7	0	0	2024-12-06	11:01:04	
20730	ESP1	29.8	47.7	1000.3	0	0	2024-12-06	11:01:06	
20731	ESP1	31.4	34.2	1007.7	0	0	2024-12-06	11:01:08	
20732	ESP1	29.8	47.7	1000.3	0	0	2024-12-06	11:01:08	
20733	ESP1	31.4	34.2	1007.7	0	0	2024-12-06	11:01:14	
20734	ESP1	29.8	47.7	1000.3	0	0	2024-12-06	11:01:14	
20735	ESP1	31.4	34.2	1007.7	0	0	2024-12-06	11:01:14	
20736	ESP1	29.8	47.7	1000.3	0	0	2024-12-06	11:01:14	
20737	ESP1	31.4	34.2	1007.7	0	0	2024-12-06	11:01:15	
20738	ESP1	29.8	47.7	1000.3	0	0	2024-12-06	11:01:22	
20739	ESP1	31.3	34.2	1007.7	0	0	2024-12-06	11:01:22	
20740	ESP1	29.8	47.6	1000.3	0	0	2024-12-06	11:01:22	
20741	ESP1	31.3	34.2	1007.7	0	0	2024-12-06	11:01:22	
20742	ESP1	29.8	47.6	1000.3	0	0	2024-12-06	11:01:22	
20743	ESP1	31.4	34.2	1007.7	0	0	2024-12-06	11:01:25	
20744	ESP1	29.8	47.6	1000.3	0	0	2024-12-06	11:01:25	
20745	ESP1	31.4	34.2	1007.7	0	0	2024-12-06	11:01:26	
20746	ESP1	29.8	47.5	1000.3	0	0	2024-12-06	11:01:28	
20747	ESP1	31.4	34.2	1007.7	0	0	2024-12-06	11:01:29	
20748	ESP1	29.8	47.5	1000.3	0	0	2024-12-06	11:01:31	
20749	ESP1	31.4	34.2	1007.7	0	0	2024-12-06	11:01:31	
20750	ESP1	29.8	47.5	1000.3	0	0	2024-12-06	11:01:33	
20751	ESP1	31.4	34.2	1007.7	0	0	2024-12-06	11:01:34	
20752	ESP1	29.8	47.5	1000.3	0	0	2024-12-06	11:01:36	
20753	ESP1	31.4	34.2	1007.7	0	0	2024-12-06	11:01:37	
20754	ESP1	29.8	47.5	1000.3	0	0	2024-12-06	11:01:39	
20755	ESP1	31.4	34.2	1007.7	0	0	2024-12-06	11:01:40	
20756	ESP1	29.8	47.4	1000.3	0	0	2024-12-06	11:01:42	

Archivo	editar	Pestañas	Ayuda						
70957	ESP1	31.1	47.9	1000.3	0	0	2024-12-06	12:00:02	
70958	ESP1	30.3	47	1005.0	17.8	2119	2024-12-06	12:00:03	
70959	ESP1	31.1	47.9	1000.3	0	0	2024-12-06	12:00:04	
70960	ESP1	30.3	47	1005.0	19.7	2447	2024-12-06	12:00:05	
70961	ESP1	31.1	48	1000.3	0	0	2024-12-06	12:00:07	
70962	ESP1	30.3	47.1	1005.0	21.5	2904	2024-12-06	12:00:08	
70963	ESP1	31.2	48	1000.3	0	0	2024-12-06	12:00:10	
70964	ESP1	30.3	47	1005.0	23	3187	2024-12-06	12:00:11	
70965	ESP1	31.1	48	1000.3	0	0	2024-12-06	12:00:13	
70966	ESP1	30.3	47	1005.0	24.3	3592	2024-12-06	12:00:14	
70967	ESP1	31.2	48	1000.3	0	0	2024-12-06	12:00:15	
70968	ESP1	30.3	47	1005.0	25	4000	2024-12-06	12:00:17	
70969	ESP1	31.1	47.9	1000.3	0	0	2024-12-06	12:00:18	
70970	ESP1	30.4	47	1005.0	25.4	4431	2024-12-06	12:00:19	
70971	ESP1	31.1	48	1000.3	0	0	2024-12-06	12:00:21	
70972	ESP1	30.4	47	1005.0	25.7	4859	2024-12-06	12:00:22	
70973	ESP1	31.1	48	1000.3	0	0	2024-12-06	12:00:24	
70974	ESP1	30.4	47	1005.0	25.9	5290	2024-12-06	12:00:25	
70975	ESP1	31.1	48	1000.3	0	0	2024-12-06	12:00:27	
70976	ESP1	30.4	47	1005.0	25.7	5727	2024-12-06	12:00:28	
70977	ESP1	31.1	48	1000.3	0	0	2024-12-06	12:00:29	
70978	ESP1	30.3	47	1005.0	23.8	6132	2024-12-06	12:00:31	
70979	ESP1	31.1	47.9	1000.3	0	0	2024-12-06	12:00:32	
70980	ESP1	30.3	47	1005.0	20.7	6456	2024-12-06	12:00:33	
70981	ESP1	31.1	48	1000.3	0	0	2024-12-06	12:00:35	
70982	ESP1	30.4	47	1005.0	18.5	6764	2024-12-06	12:00:36	
70983	ESP1	31.1	47.9	1000.3	0	0	2024-12-06	12:00:38	
70984	ESP1	30.3	47	1005.0	17	7047	2024-12-06	12:00:39	
70985	ESP1	31.1	47.9	1000.3	0	0	2024-12-06	12:00:40	
70986	ESP1	30.4	47	1005.0	15.4	7383	2024-12-06	12:00:42	
70987	ESP1	31.1	47.9	1000.3	0	0	2024-12-06	12:00:43	
70988	ESP1	30.3	47	1005.0	14.2	7540	2024-12-06	12:00:44	
70989	ESP1	31.1	47.9	1000.3	0.1	0	2024-12-06	12:00:46	
70990	ESP1	30.3	46.9	1005.0	15.1	7791	2024-12-06	12:00:47	

Ilustración 96. Datos recogidos visualizados en la base de datos visualización del 6 al 7 de diciembre del 2024

Nota: imagen es relacionadas con datos recogidos visualizados en la base de datos visualización del 6 al 7 de diciembre del 2024

Fuente: Elaboración propia.

Ilustración 97. Datos recogidos visualizados en la base de datos visualización del 7 al 8 de diciembre del 2024

Archivo	Editar	Pestañas	Ayuda						
20723	ESPI	31.4	34.2	1007.7	0	0	2024-12-06	11:00:56	
20724	ESPI	29.8	47.7	1006.3	0	0	2024-12-06	11:00:57	
20725	ESPI	31.3	34.2	1007.7	0	0	2024-12-06	11:00:58	
20726	ESPI	29.8	47.7	1006.3	0	0	2024-12-06	11:01:00	
20727	ESPI	31.4	34.2	1007.7	0	0	2024-12-06	11:01:01	
20728	ESPI	29.8	47.7	1006.3	0	0	2024-12-06	11:01:03	
20729	ESPI	31.4	34.2	1007.7	0	0	2024-12-06	11:01:04	
20730	ESPI	29.8	47.7	1006.3	0	0	2024-12-06	11:01:06	
20731	ESPI	31.4	34.2	1007.7	0	0	2024-12-06	11:01:08	
20732	ESPI	29.8	47.7	1006.3	0	0	2024-12-06	11:01:08	
20733	ESPI	31.4	34.2	1007.7	0	0	2024-12-06	11:01:14	
20734	ESPI	29.8	47.7	1006.3	0	0	2024-12-06	11:01:14	
20735	ESPI	31.4	34.2	1007.7	0	0	2024-12-06	11:01:14	
20736	ESPI	29.9	47.7	1006.3	0	0	2024-12-06	11:01:14	
20737	ESPI	31.4	34.2	1007.7	0	0	2024-12-06	11:01:15	
20738	ESPI	29.8	47.7	1006.3	0	0	2024-12-06	11:01:22	
20739	ESPI	31.3	34.2	1007.7	0	0	2024-12-06	11:01:22	
20740	ESPI	29.8	47.6	1006.3	0	0	2024-12-06	11:01:22	
20741	ESPI	31.3	34.2	1007.7	0	0	2024-12-06	11:01:22	
20742	ESPI	29.8	47.6	1006.3	0	0	2024-12-06	11:01:25	
20743	ESPI	31.4	34.2	1007.7	0	0	2024-12-06	11:01:25	
20744	ESPI	29.8	47.6	1006.3	0	0	2024-12-06	11:01:25	
20745	ESPI	31.4	34.2	1007.7	0	0	2024-12-06	11:01:26	
20746	ESPI	29.8	47.5	1006.3	0	0	2024-12-06	11:01:28	
20747	ESPI	31.4	34.2	1007.7	0	0	2024-12-06	11:01:29	
20748	ESPI	29.8	47.5	1006.3	0	0	2024-12-06	11:01:31	
20749	ESPI	31.4	34.2	1007.7	0	0	2024-12-06	11:01:31	
20750	ESPI	29.8	47.5	1006.3	0	0	2024-12-06	11:01:33	
20751	ESPI	31.4	34.2	1007.7	0	0	2024-12-06	11:01:34	
20752	ESPI	29.8	47.5	1006.3	0	0	2024-12-06	11:01:36	
20753	ESPI	31.4	34.2	1007.7	0	0	2024-12-06	11:01:37	
20754	ESPI	29.8	47.5	1006.3	0	0	2024-12-06	11:01:39	
20755	ESPI	31.4	34.2	1007.7	0	0	2024-12-06	11:01:40	
20756	ESPI	29.9	47.4	1006.3	0	0	2024-12-06	11:01:42	

Archivo	Editar	Pestañas	Ayuda						
70957	ESPI	31.1	47.9	1006.3	0	0	2024-12-08	12:00:02	
70958	ESPI	29.3	47	1005.0	17.8	2119	2024-12-08	12:00:03	
70959	ESPI	31.1	47.9	1006.3	0	0	2024-12-08	12:00:04	
70960	ESPI	29.3	47	1005.0	19.7	2447	2024-12-08	12:00:05	
70961	ESPI	31.1	48	1006.3	0	0	2024-12-08	12:00:07	
70962	ESPI	29.3	47.1	1005.0	21.5	2804	2024-12-08	12:00:08	
70963	ESPI	31.2	48	1006.3	0	0	2024-12-08	12:00:10	
70964	ESPI	29.3	47	1005.0	23	3187	2024-12-08	12:00:11	
70965	ESPI	31.1	48	1006.3	0	0	2024-12-08	12:00:13	
70966	ESPI	29.3	47	1005.0	24.3	3582	2024-12-08	12:00:14	
70967	ESPI	31.2	48	1006.3	0	0	2024-12-08	12:00:15	
70968	ESPI	29.3	47	1005.0	25	4008	2024-12-08	12:00:17	
70969	ESPI	31.1	47.9	1006.3	0	0	2024-12-08	12:00:18	
70970	ESPI	29.4	47	1005.0	25.4	4431	2024-12-08	12:00:19	
70971	ESPI	31.1	48	1006.3	0	0	2024-12-08	12:00:21	
70972	ESPI	29.4	47	1005.0	25.7	4859	2024-12-08	12:00:22	
70973	ESPI	31.1	48	1006.3	0	0	2024-12-08	12:00:24	
70974	ESPI	29.4	47	1005.0	25.9	5290	2024-12-08	12:00:25	
70975	ESPI	31.1	48	1006.3	0	0	2024-12-08	12:00:27	
70976	ESPI	29.4	47	1005.0	25.7	5717	2024-12-08	12:00:28	
70977	ESPI	31.1	48	1006.3	0	0	2024-12-08	12:00:29	
70978	ESPI	29.3	47	1005.0	23.8	6132	2024-12-08	12:00:31	
70979	ESPI	31.1	47.9	1006.3	0	0	2024-12-08	12:00:32	
70980	ESPI	29.3	47	1005.0	26.7	6456	2024-12-08	12:00:33	
70981	ESPI	31.1	48	1006.3	0	0	2024-12-08	12:00:35	
70982	ESPI	29.4	47	1005.0	18.5	6764	2024-12-08	12:00:36	
70983	ESPI	31.1	47.9	1006.3	0	0	2024-12-08	12:00:38	
70984	ESPI	29.3	47	1005.0	17	7047	2024-12-08	12:00:39	
70985	ESPI	31.1	47.9	1006.3	0	0	2024-12-08	12:00:40	
70986	ESPI	29.4	47	1005.0	15.4	7303	2024-12-08	12:00:42	
70987	ESPI	31.1	47.9	1006.3	0	0	2024-12-08	12:00:43	
70988	ESPI	29.3	47	1005.0	14.2	7548	2024-12-08	12:00:44	
70989	ESPI	31.1	47.9	1006.3	0.1	0	2024-12-08	12:00:46	
70990	ESPI	29.3	46.9	1005.0	15.1	7791	2024-12-08	12:00:47	

Ilustración 97. Datos recogidos visualizados en la base de datos visualización del 7 al 8 de diciembre del 2024

Nota: imagen relacionada con datos recogidos visualizados en la base de datos visualización del 7 al 8 de diciembre del 2024

Fuente: Elaboración propia.

Ilustración 98. Datos recogidos visualizados en la base de datos visualización del 9 de diciembre del 2024



Ilustración 99. El antes y después del terreno que se eligió para la automatización



Ilustración 99. El antes y después del terreno que se eligió para la automatización

Nota: imagen del antes y después del terreno que se eligió para la automatización

Fuente: Elaboración propia.

Ilustración 100. Automatización del riego con sus respectivas plantas



Ilustración 100. Automatización del riego con sus respectivas plantas

Nota: imagen de la automatización del riego con sus respectivas plantas

Fuente: Elaboración propia.

Ilustración 101. Automatización del riego en la actualidad.



Ilustración 101. Automatización del riego en la actualidad.

Nota: imagen la automatización del riego en la actualidad.

Fuente: Elaboración propia.