

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA



**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA INTERNET DE
LAS COSAS PARA EL MONITOREO AUTOMATIZADO DE
PARÁMETROS DE SUELO EN UN INVERNADERO DEL INIA**

Tesis para obtener el título profesional de Ingeniero Electrónico

AUTOR:

Enzo Nicolas Julio Rivas Rojas

ASESOR:

Donato Andres Flores Espinoza

Lima, 2025


Informe de Similitud


Yo, Donato Andres Flores Espinoza, docente de la Facultad de Ciencias e Ingeniería de la Pontificia Universidad Católica del Perú, asesor de la tesis titulada DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA INTERNET DE LAS COSAS PARA EL MONITOREO AUTOMATIZADO DE PARÁMETROS DE SUELO EN UN INVERNADERO DEL INIA, del autor Enzo Nicolas Julio Rivas Rojas dejo constancia de lo siguiente:

- El mencionado documento tiene un índice de puntuación de similitud de 17%. Así lo consigna el reporte de similitud emitido por el software *Turnitin* el 4/12/2025.
- He revisado con detalle dicho reporte y la Tesis, y no se advierte indicios de plagio.
- Las citas a otros autores y sus respectivas referencias cumplen con las pautas académicas.

Lugar y fecha:

Lima, 4 de diciembre de 2025

| | |
|--|--|
| Apellidos y nombres del asesor: <u>Flores Espinoza, Donato Andres</u> | |
| DNI: 06017817 | Firma  |
| ORCID: https://orcid.org/0000-0003-2092-7666 | |



*Agradezco a mi padre, Julio, y mi madre, Dina,
por enseñarme a perseverar y responsable para la realización es mis objetivos
mi padrino, Mario, por ser mi pilar de conocimientos, mi abuela, Zayda, por el apoyo
emocional y mi hermana, Ariana, que espero que siga mis pasos;*

*a mi asesor de tesis, el profesor Andres Flores,
por acompañarme y aconsejarme durante este episodio de mi vida;*

*a todos mis compañeros y compañeras de la universidad
por todos los buenos momentos que pasamos juntos;*

y a Claudia, por brindarme todo su apoyo y aliento para no rendirme.



Sé como el agua, amigo mío. El agua no resiste, fluye. Cuando viertes agua en una taza, se convierte en la taza. Cuando la viertes en una botella, se convierte en la botella. Cuando la viertes en una tetera, se convierte en la tetera. El agua puede fluir o puede chocar.

Sé agua, amigo mío.

Bruce Lee

Resumen

El avance significativo en el sector agrícola a nivel mundial ha impulsado la búsqueda constante de soluciones innovadoras que mejoren la eficiencia de los procesos agrícolas. Entre las tecnologías emergentes, el Internet de las Cosas (IoT) se presenta como una herramienta prometedora para optimizar la gestión de recursos y monitorear condiciones ambientales en tiempo real.

Esta tesis se enfoca en el desarrollo de un sistema basado en IoT para medir la humedad y la temperatura del suelo en un conjunto de 50 macetas dentro de un invernadero del Instituto Nacional de Innovación Agraria (INIA). La solución propuesta integrará sensores para la medición de variables ambientales, una base de datos para el almacenamiento y procesamiento de la información, y una plataforma web para la visualización interactiva de los datos recolectados.

El objetivo principal de este proyecto es proporcionar una herramienta de monitoreo eficiente que permita a los agricultores, técnicos y gestores del invernadero tomar decisiones más informadas y oportunas, mejorando así la calidad y productividad de los cultivos. Al implementar esta tecnología, se busca optimizar el uso de recursos como el agua y mejorar las condiciones del entorno para un crecimiento vegetal más saludable.

Con esta solución completa, se espera contribuir a la modernización de la agronomía a través de la automatización de procesos y la integración de tecnologías avanzadas, facilitando el trabajo en el campo y promoviendo prácticas agrícolas más sostenibles.

Indice

| | |
|---|-----------|
| Introducción | 1 |
| 1. Importancia de la agricultura y su sostenibilidad en invernaderos | 3 |
| 1.1. Problemática | 3 |
| 1.1.1. Panorama mundial | 3 |
| 1.1.2. Panorama nacional | 5 |
| 1.2. Estado del arte | 6 |
| 1.2.1. Sensor integral de suelo JXBS-3001-TH | 6 |
| 1.2.2. Invernadero Automatizado Inteligente: Humedad del suelo y la temperatura del suelo y sistema automático de suministro de agua (perlita, marga y limosa) [11] | 7 |
| 1.2.3. IoT- Basado en sistemas para la evaluación de los nutrientes del suelo en horticultura | 8 |
| 1.2.4. Sistema inteligente de vigilancia para la agricultura a través de internet de las cosas | 9 |
| 1.2.5. Internet de las cosas para la aplicación de agricultura de precisión | 10 |
| 1.2.6. Sistema IoT para telemetría y control ambiente de invernadero [15] | 11 |
| 1.2.7. Sistema inteligente para la agricultura de precisión [16] | 12 |
| 1.3. Justificación | 13 |
| 1.4. Objetivos | 14 |
| 1.4.1. Objetivo General | 14 |
| 1.4.2. Objetivos Específicos | 14 |
| 2. Sistemas IoT para monitorización de suelos | 15 |

| | | |
|--------|--|----|
| 2.1. | Invernaderos | 15 |
| 2.1.1. | Definición | 15 |
| 2.1.2. | Tipos de invernaderos | 15 |
| 2.1.3. | Retos | 16 |
| 2.1.4. | Ventajas | 16 |
| 2.2. | Internet de las Cosas (IoT) | 17 |
| 2.2.1. | Definición | 17 |
| 2.2.2. | Arquitectura | 17 |
| 2.2.3. | Ventajas en la agricultura | 19 |
| 2.3. | Tecnologías de comunicación | 20 |
| 2.3.1. | Zigbee (IEEE 802.15.4) | 20 |
| 2.3.2. | Wi-Fi (IEEE 802.11) | 21 |
| 2.3.3. | Bluetooth | 21 |
| 2.4. | Protocolos de redes | 22 |
| 2.4.1. | Protocolo COAP | 22 |
| 2.4.2. | Protocolo MQTT | 23 |
| 2.5. | Arquitectura del cliente IoT | 24 |
| 2.5.1. | Arquitectura No-OS | 24 |
| 2.5.2. | Arquitectura Full-OS | 24 |
| 2.5.3. | Arquitectura RTOS | 25 |
| 2.6. | Microcontroladores | 25 |
| 2.6.1. | Microchip Technology | 26 |
| 2.6.2. | Espressif | 26 |
| 2.6.3. | Protocolo de comunicación UART | 26 |
| 2.7. | Sensores | 26 |
| 2.7.1. | Sensor de humedad de suelo | 27 |
| 2.7.2. | Sensor de temperatura de suelo | 28 |
| 2.8. | Red de sensores inalámbricos | 28 |
| 2.8.1. | Definición | 28 |
| 2.8.2. | Topología | 29 |
| 2.8.3. | Nodo de sensor | 30 |

| | |
|--|-----------|
| 2.9. Interfaz de usuario | 31 |
| 2.9.1. Front-end | 32 |
| 2.9.2. Back-end | 32 |
| 2.10. Retos en la implementación | 33 |
| 3. Diseño del sistema IoT para invernadero en el INIA | 34 |
| 3.1. Requerimientos | 34 |
| 3.2. Diagrama de bloques | 35 |
| 3.3. Matriz de funciones | 36 |
| 3.3.1. Dominio mecánico | 37 |
| 3.3.2. Dominio de energía | 38 |
| 3.3.3. Dominio de actuadores | 38 |
| 3.3.4. Dominio de sensores | 39 |
| 3.3.5. Dominio de control | 40 |
| 3.3.6. Dominio de la web | 40 |
| 3.3.6.1. Frontend | 41 |
| 3.3.6.2. Backend | 42 |
| 3.4. Matriz morfológica | 43 |
| 3.4.1. Electrónica | 43 |
| 3.4.2. Nube | 44 |
| 3.5. Selección de componentes | 46 |
| 3.5.1. Sensor de humedad y temperatura de suelo | 47 |
| 3.5.2. Microcontrolador | 48 |
| 3.5.3. Regulador de voltaje | 49 |
| 3.5.4. Alimentación | 49 |
| 3.6. Selección de la nube | 51 |
| 3.6.1. Entorno de programación para front-end | 51 |
| 3.6.2. Entorno de programación para back-end | 52 |
| 3.6.3. Base de datos | 53 |
| 3.7. Diseño electrónico de la tarjeta | 54 |
| 3.7.1. Diagrama esquemático | 54 |

| | |
|--|------------|
| 3.7.2. Circuito impreso | 58 |
| 3.8. Diagramas de flujos | 59 |
| 3.8.1. Diagrama de flujo para la electrónica | 60 |
| 3.8.2. Diagrama de flujo para la página web FrontEnd | 61 |
| 3.8.3. Diagrama de flujo para la página web BackEnd | 62 |
| 3.9. Flujo de datos de la información | 62 |
| 3.10. Consideraciones para la implementación | 63 |
| 3.10.1. Modelado de carcasa | 63 |
| 3.10.2. Bosquejo para la implementación | 64 |
| 3.10.3. Presupuesto total | 67 |
| 4. Pruebas y resultados | 68 |
| 4.1. Validación del nodo sensor | 68 |
| 4.1.1. Validación de los sensores | 68 |
| 4.1.2. Prueba de duración la batería | 70 |
| 4.1.3. Validación de comunicación con el servidor | 70 |
| 4.2. Validación del backend | 72 |
| 4.2.1. Validación de las solicitudes | 72 |
| 4.2.2. Validación con base de datos | 76 |
| 4.2.3. Validación de envío de alerta a usuario | 78 |
| 4.3. Validación del frontend | 81 |
| 4.3.1. Validación de logueo de usuario | 81 |
| 4.3.2. Validación de ingreso de datos | 84 |
| 4.3.3. Validación de visualización de los datos ingresados | 86 |
| 4.3.4. Validación de visualización multiplataforma | 92 |
| Conclusiones | 98 |
| Recomendaciones | 99 |
| Bibliografía | 100 |

Indice de figuras

| | |
|---|----|
| 1.1. PBI a nivel mundial entre 2015 a 2021[5] | 4 |
| 1.2. Valor agregado a nivel mundial entre 2015 a 2021[6] | 4 |
| 1.3. Valor en miles de dólares desde el año 2018 a inicios del año 2023[7] | 5 |
| 1.4. Masa en toneladas desde el año 2018 a inicios del año 2023[7] | 5 |
| 1.5. Sensor integral de suelo JXBS-3001-TH[10] | 7 |
| 1.6. Diagrama de bloques del controlador [11] | 8 |
| 1.7. Diagrama de bloques del nodo de detección inteligente del suelo [12] | 9 |
| 1.8. Diagrama esquemático para las conexiones con el Arduino para un sistema inteligente de vigilancia para la agricultura a través de internet de las cosas [13] | 10 |
| 1.9. Diagrama de bloques general del internet de las cosas para la aplicación de agricultura de precisión[14] | 11 |
| 1.10. Diagrama de bloques de todo el sistema IoT para telemetría y control ambiente de invernadero[15] | 12 |
| 1.11. Diagrama de bloques del sistema IoT para la agricultura de precisión[16] | 13 |
| 2.1. Formas de los distintos invernaderos[18] | 16 |
| 2.2. Arquitectura general de un sistema IoT para la agricultura[20] | 18 |
| 2.3. Arquitectura estandarizada de un sistema IoT para la agricultura[20] | 19 |
| 2.4. Arquitectura de MQTT[27] | 24 |
| 2.5. Topologías [34] | 29 |
| 2.6. Nodo de sensor[34] | 31 |
| 3.1. Diagrama de bloques de solución propuesta[Elaboración propia] | 35 |
| 3.2. Matriz de funciones parte 1[Elaboración propia] | 36 |
| 3.3. Matriz de funciones parte 2[Elaboración propia] | 37 |

| | |
|---|----|
| 3.4. Dominio mecánico[Elaboración propia] | 37 |
| 3.5. Dominio de energía[Elaboración propia] | 38 |
| 3.6. Matriz de funciones[Elaboración propia] | 39 |
| 3.7. Dominio de sensores[Elaboración propia] | 39 |
| 3.8. Dominio de control[Elaboración propia] | 40 |
| 3.9. Dominio de la web[Elaboración propia] | 41 |
| 3.10. Frontend[Elaboración propia] | 42 |
| 3.11. Backend[Elaboración propia] | 42 |
| 3.12. Segmentación base de datos no relacional[Elaboración propia] | 45 |
| 3.13. Estructura de datos para la base de datos[Elaboración propia] | 45 |
| 3.14. Estructura de para la página web[Elaboración propia] | 46 |
| 3.15. Diagrama esquemático[Elaboración propia] | 54 |
| 3.16. Etapa de control de esquemático[Elaboración propia] | 55 |
| 3.17. Conversor USB-UART de esquemático[Elaboración propia] | 55 |
| 3.18. Regulador 5V a 3.3V de esquemático[Elaboración propia] | 56 |
| 3.19. Botones de programación de esquemático[Elaboración propia] | 56 |
| 3.20. Botones de programación de esquemático[Elaboración propia] | 57 |
| 3.21. Alimentación de esquemático[Elaboración propia] | 57 |
| 3.22. Conexión de sensor para esquemático[Elaboración propia] | 57 |
| 3.23. Circuito impreso parte superior[Elaboración propia] | 58 |
| 3.24. Circuito impreso parte inferior[Elaboración propia] | 58 |
| 3.25. Circuito impreso 3D[Elaboración propia] | 59 |
| 3.26. Diagrama de flujo para la electrónica[Elaboración propia] | 60 |
| 3.27. Diagrama de flujo para la página web FrontEnd[Elaboración propia] | 61 |
| 3.28. Diagrama de flujo para la página web BackEnd[Elaboración propia] | 62 |
| 3.29. Flujo de datos de la información[Elaboración propia] | 63 |
| 3.30. Componentes dentro de la caja[Elaboración propia] | 63 |
| 3.31. Ubicación del INIA[Google Maps App 2023] | 64 |
| 3.32. Ubicación del invernadero[Google Maps App 2023] | 64 |
| 3.33. Invernadero parte lateral[Elaboración propia] | 65 |
| 3.34. Invernadero parte frontal[Elaboración propia] | 65 |

| | |
|---|----|
| 3.35. Interior del invernadero[Elaboración propia] | 66 |
| 3.36. Situación de las cajas en el invernadero[Elaboración propia] | 66 |
| 3.37. Solución implementada en una maceta[Elaboración propia] | 67 |
| 4.1. Calibración del sensor de humedad y temperatura en el invernadero[Elaboración propia] | 69 |
| 4.2. Calibración del sensor de humedad y temperatura [Elaboración propia] | 69 |
| 4.3. Nodo sensor en casa.[Elaboración propia] | 70 |
| 4.4. Consola del ESP32.[Elaboración propia] | 71 |
| 4.5. Recepción de datos en el cliente en línea.[66] | 71 |
| 4.6. Interfaz de Postman [64]. | 72 |
| 4.7. Registro de usuario nuevo [64]. | 73 |
| 4.8. Usuario ya existente [64]. | 74 |
| 4.9. Consulta de autenticación exitosa para el servidor [64]. | 74 |
| 4.10. Consulta de autenticación fallida para el servidor.[POSTMAN 2023] | 75 |
| 4.11. Consulta de datos del sensor1 en la base de datos [64]. | 75 |
| 4.12. Recepción de datos por mqtt en la consola de Render [63]. | 76 |
| 4.13. Visualizado de la colección del nodo de sensor 50 en MongoDB [62]. | 77 |
| 4.14. Colección de users en MongoDB.[62] | 78 |
| 4.15. Iniciación del bot de Telegram.[Elaboración propia] | 79 |
| 4.16. Suscripción del bot a un grupo de Telegram.[Elaboración propia] | 79 |
| 4.17. Mensaje de nuevo chat al administrador.[Elaboración propia] | 80 |
| 4.18. Mensaje de alerta a telegram de grupo creado.[Elaboración propia] | 80 |
| 4.19. Página de logeo.[Elaboración propia] | 81 |
| 4.20. Ingreso de credenciales.[Elaboración propia] | 82 |
| 4.21. Ingreso correcto de las credenciales.[Elaboración propia] | 82 |
| 4.22. Ingreso incorrecto de las credenciales.[Elaboración propia] | 83 |
| 4.23. Interfaz de usuario.[Elaboración propia] | 83 |
| 4.24. Barra de navegación.[Elaboración propia] | 84 |
| 4.25. Selector de fechas-Inicio.[Elaboración propia] | 84 |
| 4.26. Selector de fechas-Fin[Elaboración propia] | 85 |

| | |
|---|----|
| 4.27. Selector de nodo de sensor de maceta[Elaboración propia] | 86 |
| 4.28. Datos seleccionados del nodo de sensor 1[Elaboración propia] | 87 |
| 4.29. Primeros datos seleccionados del nodo de sensor 50[Elaboración propia] | 87 |
| 4.30. Segundos datos seleccionados del nodo de sensor 50[Elaboración propia] | 88 |
| 4.31. Envío de datos para la maceta 50[Elaboración propia] | 89 |
| 4.32. Respuesta del servidor y guardado en la base de datos[Elaboración propia] | 90 |
| 4.33. Respuesta del servidor y guardado en la base de datos[Elaboración propia] | 91 |
| 4.34. Solicitud de datos al servidor[Elaboración propia] | 91 |
| 4.35. Datos actualizados para el sensor 50[Elaboración propia] | 92 |
| 4.36. Pantalla de inicio en celular[Elaboración propia] | 93 |
| 4.37. Autenticación exitosa en celular[Elaboración propia] | 93 |
| 4.38. Interfaz de usuario en celular[Elaboración propia] | 94 |
| 4.39. Barra de navegación en celular[Elaboración propia] | 94 |
| 4.40. Selector de nodo de sensor en celular[Elaboración propia] | 95 |
| 4.41. Selector de fecha de inicio en celular[Elaboración propia] | 95 |
| 4.42. Selector de fecha de fin en celular[Elaboración propia] | 96 |
| 4.43. Primeros datos en nodo de sensor 50 en celular[Elaboración propia] | 96 |
| 4.44. Segundos datos en nodo de sensor 50 en celular[Elaboración propia] | 97 |

Indice de tablas

| | | |
|------|---|----|
| 2.1. | Tabla comparativa de tres topologías | 30 |
| 2.2. | Tabla de resultados de topología estrella y arbol | 30 |
| 3.1. | Tabla comparativa de sensores de humedad y temperatura de suelo | 47 |
| 3.2. | Tabla comparativa de microcontroladores | 48 |
| 3.3. | Tabla comparativa de reguladores de voltaje | 49 |
| 3.4. | Tabla comparativa de baterías | 51 |
| 3.5. | Tabla comparativa de servicios para desarrollo de frontend | 52 |
| 3.6. | Tabla comparativa de servicios para desarrollo de backend | 52 |
| 3.7. | Tabla comparativa de servicios para bases de datos | 53 |
| 3.8. | Presupuesto total para 50 macetas[Elaboración propia] | 67 |

Introducción

En la presente tesis se desarrolla el diseño e implementación de un sistema de Internet de las Cosas (en inglés IOT) para el monitoreo sofisticado del suelo para cultivos en el Instituto Nacional de Innovación Agraria.

El sistema IoT puede funcionar como una interfaz clave entre las mediciones meteorológicas y ambientales del entorno donde se desarrollan diversos experimentos orientados a la innovación agrícola. En este contexto, se busca establecer un monitoreo eficiente de los cultivos dentro del INIA (Instituto Nacional de Innovación Agraria), facilitando la implementación de actividades que impulsan la investigación y la transferencia de tecnologías [1].

El INIA se dedica a la producción de semillas, plántones y reproductores de alto valor genético, lo que contribuye significativamente al desarrollo del sector agrícola. Además, se encarga de articular y regular las actividades de investigación, desarrollo e innovación (I+D+i) en colaboración con distintos actores del sector, con el objetivo de fomentar la competitividad, garantizar la seguridad alimentaria y promover la adaptación al cambio climático[2].

En paralelo, el INIA también formula, propone y ejecuta políticas nacionales, así como planes de innovación agraria, alineados con las necesidades del sector y las tendencias globales para fortalecer la sostenibilidad y productividad del agro.

Sin embargo, los procesos que realizan no suelen ser los óptimos, ya que requieren de mano de obra calificada para obtener los datos de monitoreo y el control de los cultivos con los que trabajan. Esto podría conllevar errores por falla humana.

Para poder minimizar el gasto de personal y los errores para el control y monitoreo de los distintos cultivos que yacen en el invernadero, y lograr tener un sistema más eficiente y automatizado en el presente trabajo se plantearon soluciones con distintas clases de controladores para al final optar por el microcontrolador Esp32, debido a sus funcionalidades que beneficia el proceso para la implementación del internet de las cosas. Asimismo, se

brindará un entorno de visualización a través de una página web con un servidor en Internet y un sistema de alertas mediante un servicio de mensajería instantánea.



Capítulo 1

Importancia de la agricultura y su sostenibilidad en invernaderos

1.1. Problemática

1.1.1. Panorama mundial

En todo el mundo, se realiza la actividad agrícola, la cual es muy beneficiosa para todos. Sin embargo, debemos saber cómo está el panorama a nivel mundial de este. Según la FAO, para lograr una alimentación y una agricultura sostenible, se debe basar en la calidad de los alimentos y su acceso para todos y los recursos naturales deben ser gestionados para lograr la preservación de las funciones del ecosistema para suplir las necesidades humanas del ahora y del mañana [3].

Por lo tanto, es necesario adoptar un enfoque integral para la resolución de problemas que permita enfrentar los desafíos científicos, sociales y organizacionales que surgen en la intersección de diversos factores, como el cambio climático, las plagas y enfermedades, los cambios en el uso del suelo, la gestión de recursos hídricos, entre otros. A este conjunto de retos y transformaciones se le conoce como la Cuarta Revolución Industrial o el Mundo 4.0 [4].

Se sabe que el mundo ha mantenido su valor del PIB(Producto Bruto Interno), respecto a sus producciones anuales en el sector agrícola desde el 2015 hasta el 2021, tal cual se puede obtener del gráfico [5].

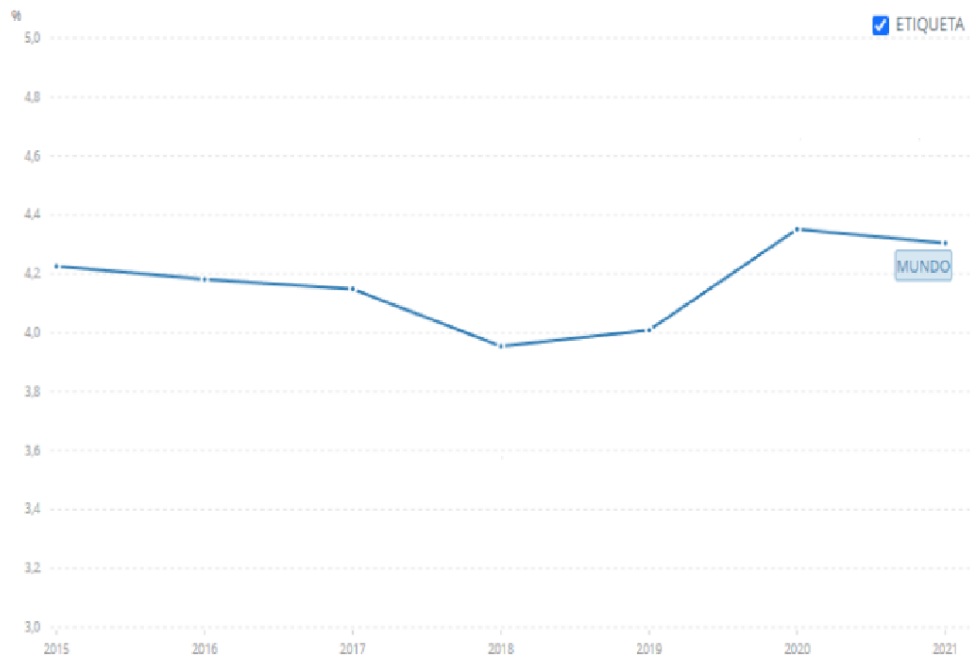


Figura 1.1: PBI a nivel mundial entre 2015 a 2021[5]

Asimismo, se tiene el valor agregado que es la producción neta de un sector después de sumar todos los productos y restar los insumos intermedios. Sin embargo, el valor agregado en Mill. de dólares ha contemplado un crecimiento constante desde el año 2015 hasta el 2021. Esto se puede apreciar en la siguiente gráfica [6].

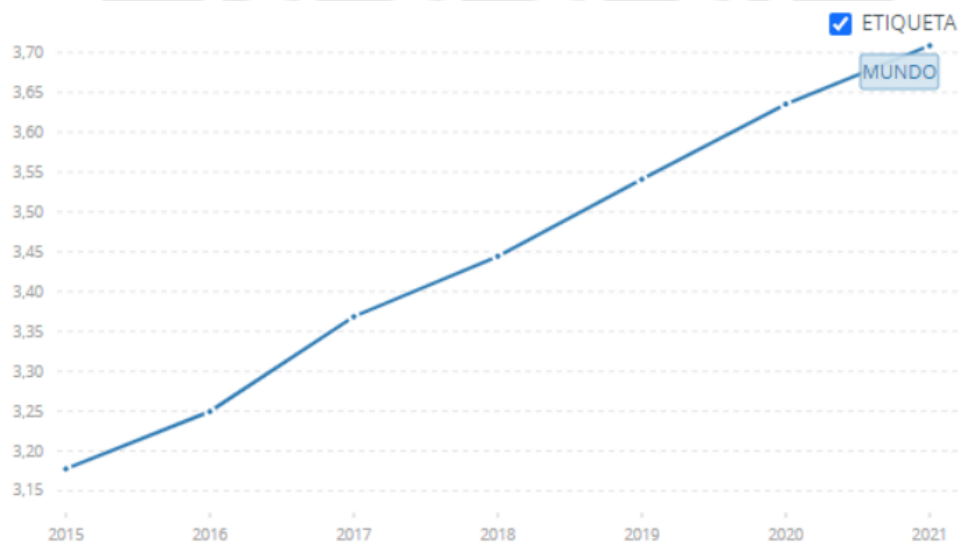


Figura 1.2: Valor agregado a nivel mundial entre 2015 a 2021[6]

Todos estos datos mostrados anteriormente, muestran que se necesita un manejo adecuado de la agricultura de una manera eficiente, óptima y automatizada(de ser posible) para lograr mejores resultados a futuro.

1.1.2. Panorama nacional

En Perú, según los datos del Ministerio de Desarrollo Agrario y Riego [7] se puede observar como el Perú ha tenido un crecimiento notable respecto a los anteriores años. En especial, en el sector de exportaciones en los años 2022 e inicios del 2023. Como en las siguientes graficas:



Figura 1.3: Valor en miles de dólares desde el año 2018 a inicios del año 2023[7]

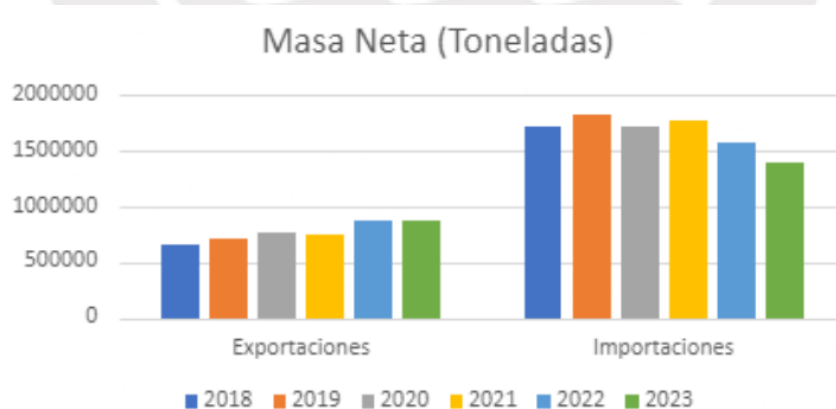


Figura 1.4: Masa en toneladas desde el año 2018 a inicios del año 2023[7]

Dados los datos anteriormente mencionados, se debe contar una entidad que se encargue de encontrar innovaciones para este sector agrícola. Por ello, el Perú posee el Instituto

Nacional de Innovación Agraria, la cual es la encargada de constantemente realizar investigaciones para obtener mejores resultados a futuro.

Existen distintas problemáticas en la agricultura en el aspecto mismo de la agricultura y el tecnológico. Para el aspecto agrícola, llegan a ser la falta de conocimiento correcto de los agricultores sobre la textura del suelo y el cuidado adecuado que deben darles a los cultivos; el uso excesivo de fertilizantes sintéticos, que dañan la textura del suelo; el cambio climático que afecta el crecimiento de los cultivos y la escasez de agua [8]. Para el aspecto tecnológico, se evidencia que los sistemas IoT y los servidores en la nube pueden llevar problemas de retraso en la obtención de resultados y almacenamiento de grandes cantidades de datos generados por los sensores. Sin embargo, existen ciertos inconvenientes cuando se realizan estas nuevas tecnologías como las mencionadas a continuación. Se debe contar con: una infraestructura energéticamente eficiente para IoT; un suministro de energía continuo y protocolos eficientes para una comunicación confiable; la integración de la eficiencia energética en toda la estructura IoT para un rendimiento tolerable; la implementación de algoritmos correctos para contar con una seguridad sólida para el manejo de los datos del sistema IoT; dispositivos IoT con la menor emisión de CO₂ posible; contar con soluciones adecuadas para mejorar los parámetros de calidad de servicio como el ancho de banda, la latencia y el rendimiento; y reducir en lo posible el retraso existente por la comunicación móvil a móvil de alto tráfico [8,9].

1.2. Estado del arte

1.2.1. Sensor integral de suelo JXBS-3001-TH

Este sensor es adecuado para lograr obtener la temperatura del suelo, la humedad, salinidad del suelo (conductividad), el PH del suelo y el nitrógeno, el fósforo y el potasio. El sensor posee alta precisión, respuesta rápida y salida estable; asimismo, logra ser adecuado para distintos tipos de suelo [10]. Puede estar en el suelo durante un largo periodo de tiempo, es resistente a la electrólisis a largo plazo, resistente a la corrosión, encapsulado al vacío y completamente impermeable. Para la instalación de este sensor se debe contar con todas enterradas o todas las sondas insertadas en el medio medido. En la siguiente figura, se muestra el sensor de suelo

explicado anteriormente.



Figura 1.5: Sensor integral de suelo JXBS-3001-TH[10]

1.2.2. Invernadero Automatizado Inteligente: Humedad del suelo y la temperatura del suelo y sistema automático de suministro de agua (perlita, marga y limosa) [11]

El presente artículo académico se centra en el desarrollo de un sistema automático de riego para invernaderos basado en sensores de humedad del suelo y seguimiento inalámbrico. De igual forma, muestra información sobre distintos tipos de suelo y simulación de circuitos.

El sistema utiliza un Arduino Uno conectado a sensores para medir la humedad del suelo, la temperatura del suelo, humedad del ambiente y la cantidad de luz. Asimismo, cuenta con una bomba de agua para lograr el riego automático; además, cuenta con actuadores para controlar la temperatura y humedad del invernadero. Posterior a ello, la comunicación se realiza mediante WIFI para lograr la transmisión de los datos a un servidor.

Finalmente, se presenta la información recolectada mediante una aplicación móvil denominada “BLYNK” para monitorear en tiempo real los niveles de humedad en tres tipos diferentes de suelo (perlita, marga y limosa) y controlar el sistema de riego automático en función de los niveles de humedad previamente recolectados [11].

Este sistema es ideal para investigaciones para evaluar los requisitos de una especie de planta o monitorear cómo se comporta en un ambiente climatizado. En la siguiente figura, se muestra cómo es el diagrama de bloques para el sistema propuesto descrito anteriormente.

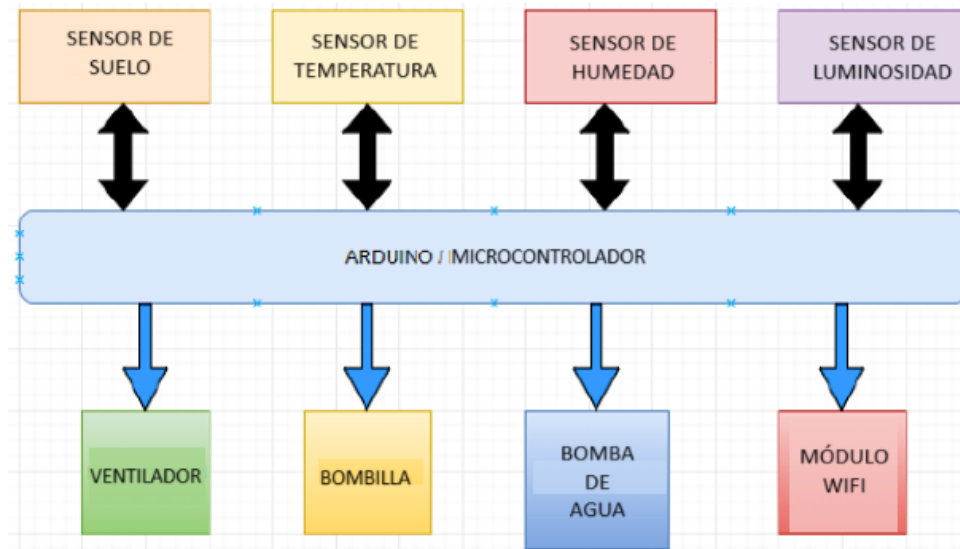


Figura 1.6: Diagrama de bloques del controlador [11]

1.2.3. IoT- Basado en sistemas para la evaluación de los nutrientes del suelo en horticultura

Este artículo académico se centra en la implementación de un sistema basado en IoT para la evaluación de nutrientes existentes en el suelo para la horticultura. Se describen las tecnologías y técnicas para la caracterización de la fertilidad del suelo; asimismo, un sistema de información para la evaluación de los nutrientes del suelo.

El sistema IoT planteado incluye nodos de sensores inteligentes, los cuales son capaces de realizar la adquisición de datos, el procesamiento de datos primarios y la comunicación de datos utilizando protocolos de comunicación Wi-Fi/LoRa para enviar al servidor en la nube para su almacenamiento y análisis. Estos sensores son sensores multi electrodos basados en el principio del sensor TDR para medir la temperatura, el pH, la humedad del suelo, la conductividad y los contenidos de N, P y K en el suelo [12].

Asimismo, se puede utilizar una red inalámbrica de sensores y los datos se pueden visualizar, posteriormente, en una aplicación móvil. En esta aplicación, se configuran los nodos de sensores y enviar comandos específicos a los nodos de sensores. Al ser un sistema en tiempo real, permite a los agricultores tomar decisiones informadas para el riego y la fertilización. En la siguiente figura, se muestra el diagrama de bloques del nodo de detección inteligente del suelo.

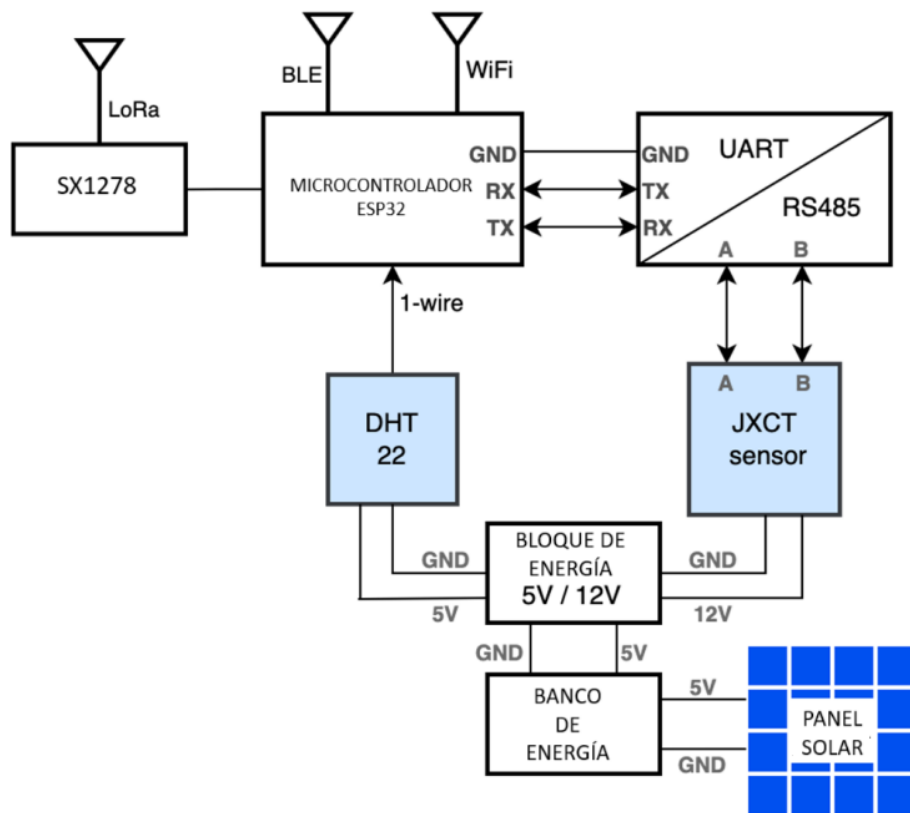


Figura 1.7: Diagrama de bloques del nodo de detección inteligente del suelo [12]

1.2.4. Sistema inteligente de vigilancia para la agricultura a través de internet de las cosas

El artículo académico se centra en un sistema que utiliza sensores para monitorear las condiciones de los cultivos y automatizar tareas que se puedan realizar. El sistema propuesto es un sistema de monitoreo que utiliza sensores conectados a un Arduino para la recopilación de datos sobre las condiciones de los cultivos, humedad de suelo, temperatura y nivel de agua. Los sensores que se utilizan para esta tarea son sensor de humedad de suelo, sensor de temperatura y humedad de suelo, sensor de gotas de lluvia y un sensor PIR. Finalizada la recolección de datos, se envían a un módulo Wi-Fi que se conecta a un Raspberry Pi, que está conectado a Internet [13]. El sistema puede funcionar de manera automática o manual. De manera manual, al usuario se le comunica mediante una notificación a su dispositivo móvil, que debe contar con un sistema operativo Android, con los datos recolectados y este pueda

decidir correctamente. Por el otro lado, el modo automático, el sistema utiliza los datos recolectados para determinar si es necesario activar uno de los actuadores que serían, un ventilador o una bomba de agua para climatizar correctamente. En el siguiente gráfico, se puede visualizar el diagrama esquemático para las conexiones del arduino.

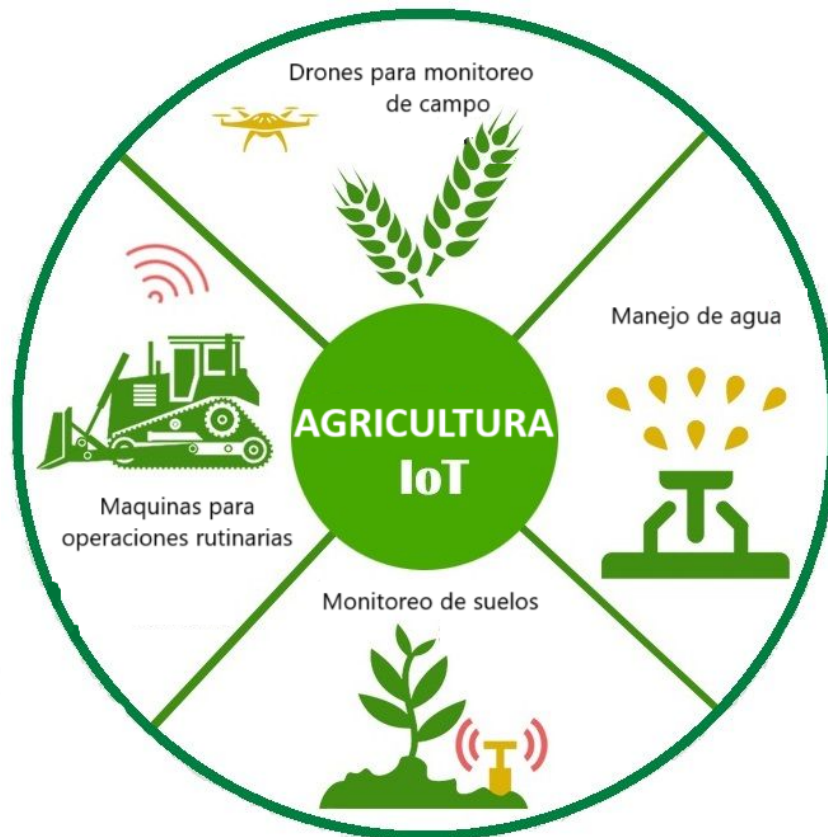


Figura 1.8: Diagrama esquemático para las conexiones con el Arduino para un sistema inteligente de vigilancia para la agricultura a través de internet de las cosas [13]

1.2.5. Internet de las cosas para la aplicación de agricultura de precisión

El artículo académico se centra en el uso de un sistema IoT basado en la nube para la agricultura de precisión. El sistema propuesto cuenta con sensores para la humedad del suelo, la temperatura, la humedad relativa y la intensidad de la luz. Todos estos datos son enviados mediante el microcontrolador NodeMCU hacia la plataforma en la nube, en este caso se usa Thingspeak. Los datos obtenidos se muestran mediante gráficas. Finalmente, se puede acceder mediante una aplicación móvil para la visualización de los gráficos disponibles en tiempo real mediante la nube alojada en Internet [14]. En la siguiente figura se muestra el diagrama de

bloques general del sistema IoT

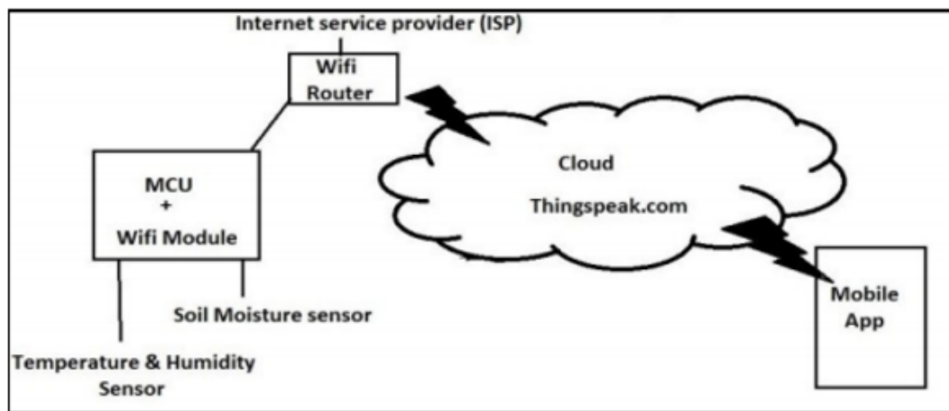


Figura 1.9: Diagrama de bloques general del internet de las cosas para la aplicación de agricultura de precisión[14]

1.2.6. Sistema IoT para telemetría y control ambiente de invernadero [15]

El artículo académico presenta el desarrollo y puesta a prueba de un sistema basado en IoT para el monitoreo y control de ambientes de invernadero. El sistema utiliza nodos de sensores inalámbricos, los cuales constan de sensores, un microcontrolador y un escudo para la comunicación, para monitorear parámetros ambientales en el invernadero. Estos parámetros se obtienen mediante el uso de sensores para la humedad del suelo, intensidad de luz ambiental, calidad de aire. Después de obtener los datos, se transmite a un servidor central de monitoreo, para este caso se utilizó un servidor ThingSpeak, el cual es una plataforma analítica de datos, visualización y análisis en tiempo real. Se realizó la comunicación a través de GSM, el cual es un estándar de comunicaciones móviles de segunda generación (2G) [15]. Asimismo, se cuenta con una aplicación móvil para las notificaciones y control del ambiente del invernadero. El sistema fue exitoso para controlar bombas de riego, los acondicionadores de aire y luces artificiales en el invernadero. Así, se logró una solución efectiva y eficiente de energía para la utilización de este tipo de sistemas en invernaderos. En la siguiente figura, se muestra el funcionamiento del sistema desarrollado.

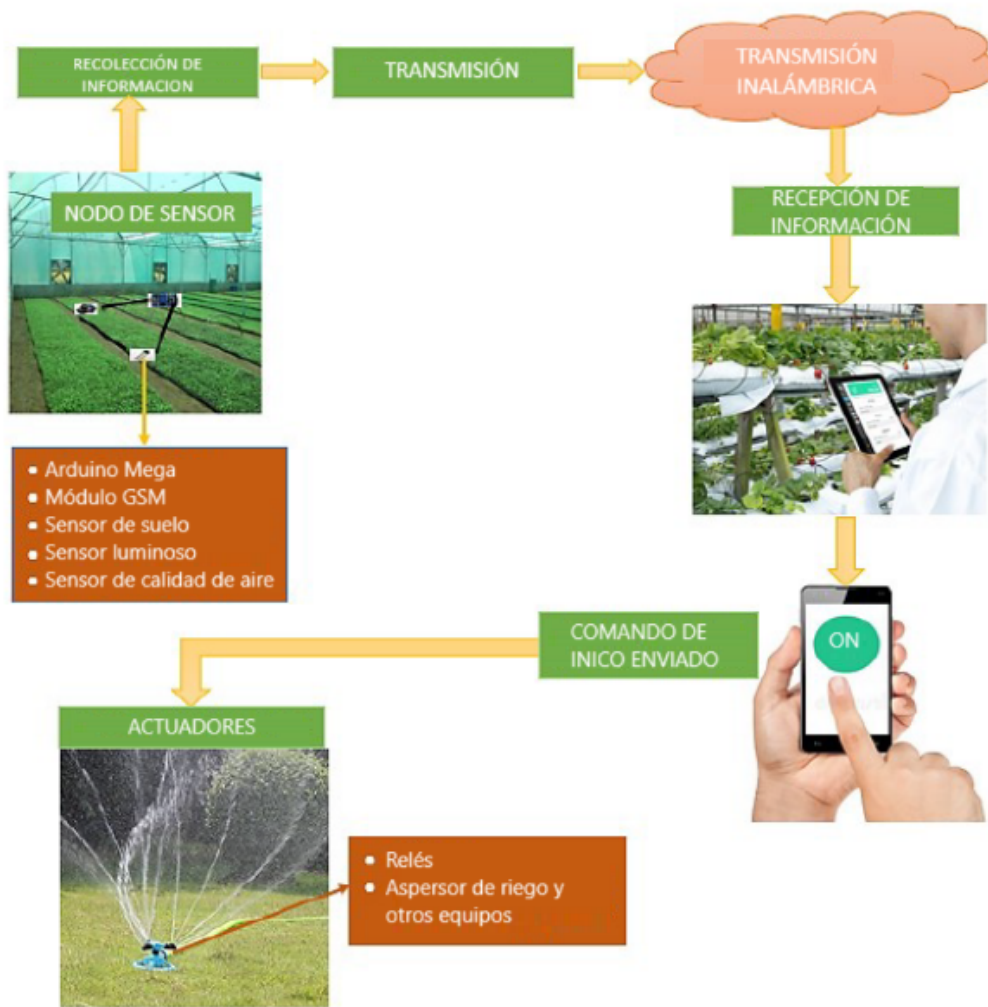


Figura 1.10: Diagrama de bloques de todo el sistema IoT para telemetría y control ambiente de invernadero[15]

1.2.7. Sistema inteligente para la agricultura de precisión [16]

El presente artículo se centra en el desarrollo de un sistema inteligente en un invernadero para la agricultura de precisión utilizando IoT. El sistema propuesto se basa en una comunicación inalámbrica mediante el uso de sensores de temperatura y humedad, nivel de agua y humedad de suelo. Todos estos sensores se conectan a un Raspberry Pi que se encarga del procesamiento y la transmisión de datos mediante el protocolo MQTT.

Posteriormente, estos datos se almacenan en una base de datos denominada InfluxDB, la cual es una base de datos de código abierto para grandes cantidades de datos de series temporales en tiempo real. Asimismo, con Grafana pueden realizar la visualización de datos y la creación de paneles personalizados para monitorear y analizar los datos de la base de datos [16].

Luego de ello, las actividades que se realizan dentro del invernadero se realizan mediante actuadores, los cuales son bombas de agua o sistema de enfriamiento, accionados por el sistema inteligente. Finalmente, los datos se muestran en una interfaz amigable para el usuario mediante una aplicación móvil, la cual posee cuatro indicadores: temperatura y humedad del aire, humedad y temperatura del suelo y presión. En la siguiente figura, se muestra el diagrama de bloques del sistema propuesto.

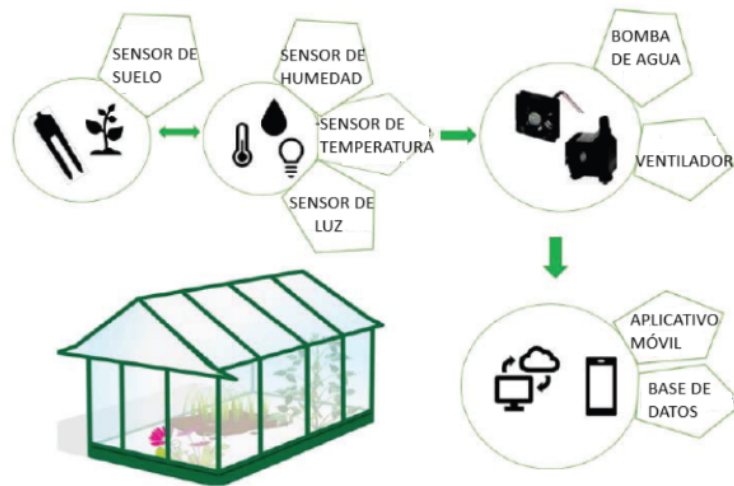


Figura 1.11: Diagrama de bloques del sistema IoT para la agricultura de precisión[16]

1.3. Justificación

Lo que se busca al diseñar e implementar esta solución es obtener un sistema económico y amigable para el productor. Esto logrará que se pueda disminuir la mano de obra que se utiliza en el INIA, ya que realizan esto de manera manual.

Primero, recolectan los datos del sensor para obtener los nutrientes, conductividad eléctrica, humedad de suelo de cada maceta del invernadero para cada cultivo. Esto lo suelen realizar en intervalos semanales, y suele tardar entre 1 a 2 horas, ya que cuentan con 2 personas para realizar la toma de datos. Una persona se encarga de la toma de los datos del sensor de humedad del suelo y la otra los anota. Asimismo, se deben descargar los datos climáticos del invernadero obtenidos por la estación meteorológica instalada dentro. Por lo detallado anteriormente, es un proceso engorroso, el cual es posible lograr una automatización del sistema para que sea más entendible para los agricultores que manejan el invernadero. Luego

de ello, se debe enviar la información mediante ciertas computadoras que posean instalado el programa. Posteriormente, toda esta información la deben subir a un programa, el cual se encargará de procesar la información para realizar la visualización de los datos a los usuarios. Para todo lo descrito anteriormente, se buscará utilizar las tecnologías descritas en el estado del arte, tratando de implementarlas con la red WIFI del mismo INIA, ya que la señal llega hasta el invernadero de manera aceptable. Asimismo, obtener una interfaz para el usuario amigable, en la cual se muestran los datos de manera legible mediante gráficos estadísticos de cada maceta. Finalmente, se tratará de encontrar la solución más adecuada y económica para la realización del sistema.

1.4. Objetivos

1.4.1. Objetivo General

-Diseñar un sistema IOT para el monitoreo automatizado del suelo para un invernadero en el Instituto Nacional de Innovación Agraria.

1.4.2. Objetivos Específicos

- Sensar las variables de interés del suelo, específicamente la humedad y la temperatura de las macetas.
- Determinar las tecnologías o técnicas más viables para la medición precisa de dichas variables.
- Seleccionar el sensor adecuado para las variables de suelo y la red de sensores, definiendo sus características y requerimientos para el sistema.
- Planificar el diseño óptimo de la red de sensores en el invernadero del INIA, considerando la distribución de las 50 macetas.
- Realizar una prueba de concepto del diseño propuesto utilizando una maceta del invernadero.
- Diseñar una interfaz de usuario clara y amigable para la visualización de los datos de los cultivos.
- Contar con un alojamiento web que permita visualizar los datos con acceso restringido únicamente al personal autorizado.

Capítulo 2

Sistemas IoT para monitorización de suelos

2.1. Invernaderos

2.1.1. Definición

Un invernadero es una estructura cerrada, la cual está diseñada para cultivar plantas en un ambiente controlado. Su propósito es crear un ambiente controlado para el crecimiento de las plantas, proporcionando las condiciones óptimas para su desarrollo y protegiéndose de condiciones climáticas adversas. Se pueden controlar factores como la temperatura, la humedad, la luz y la ventilación [17].

2.1.2. Tipos de invernaderos

Para elegir qué invernadero construir se debe tomar en cuenta las condiciones climáticas en las que se piensa construir este. Cada tipo de estructura de invernadero tiene sus ventajas y desventajas en términos de costo, eficiencia de ventilación, transmisión de luz, volumen y resistencia al viento, entre otros factores [18]. Por ello se mencionan a continuación algunas de ellas: a. Invernaderos de techo a dos aguas: también conocidos como invernaderos de techo en forma de sierra, son comunes en climas templados y se caracterizan por su techo inclinado en ambos lados.

b. Invernaderos de arco redondeado con pared lateral vertical: este tipo de invernadero tiene un

techo en forma de arco redondeado y paredes laterales verticales.

c. Invernaderos de túnel con arco redondeado: son los más utilizados en países en desarrollo con climas suaves debido a su construcción relativamente simple y bajo costo.

d. Invernaderos de arco redondeado con pared lateral inclinada: este tipo de invernadero tiene un techo en forma de arco puntiagudo y paredes laterales inclinadas.

e. Invernaderos de arco puntiagudo con pared lateral vertical: este tipo de invernadero tiene un techo en forma de arco puntiagudo y paredes laterales verticales.

f. Invernaderos de arco puntiagudo con pared lateral inclinada: este tipo de invernadero tiene un techo en forma de arco puntiagudo y paredes laterales inclinadas.

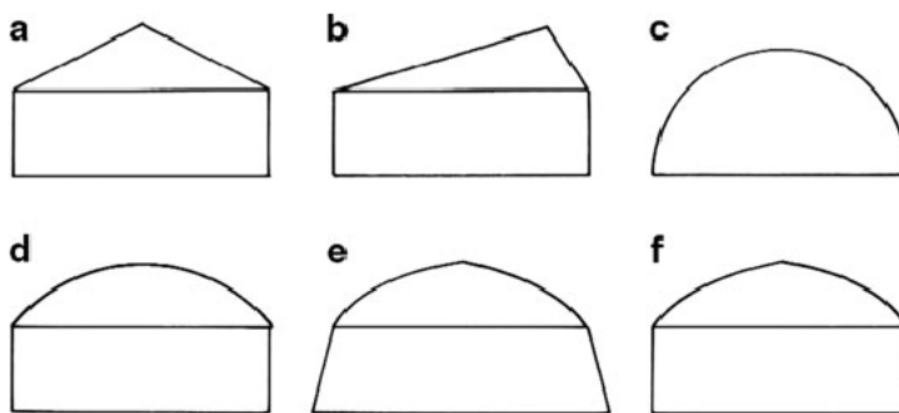


Figura 2.1: Formas de los distintos invernaderos[18]

2.1.3. Retos

Si bien pueden existir muchos factores para el uso de este tipo se mencionan los siguientes: La necesidad de supervisión extensiva para controlar y mantener las condiciones ambientales adecuadas para el crecimiento de las plantas, que requiere de trabajadores capacitados y conocedores de las condiciones ambientales (temperatura, humedad, luminosidad); la toma de decisiones eficientes para proteger los cultivos de plagas u otros problemas. [17,18]

2.1.4. Ventajas

Este tipo de estructura ofrece la siguiente serie de ventajas[19]: 1. Control del clima: Permite controlar el clima en el que se cultivan las plantas, lo que significa que se pueden crear las condiciones ideales para el crecimiento de las plantas, independientemente del clima

exterior.

2. Protección contra plagas y enfermedades: Protege las plantas de plagas y enfermedades, lo que reduce la necesidad de pesticidas y otros productos químicos.

3. Mayor productividad: Al controlar el clima y proteger las plantas de plagas y enfermedades, un invernadero puede aumentar la productividad de los cultivos.

4. Cultivo durante todo el año: Permite cultivar plantas durante todo el año, lo que significa que se pueden producir cultivos fuera de temporada.

5. Ahorro de agua: Permite un uso más eficiente del agua, ya que se puede controlar la cantidad de agua que se utiliza para regar las plantas.

6. Mayor rentabilidad: Al aumentar la productividad y reducir los costos de producción, un invernadero puede aumentar la rentabilidad de la agricultura.

2.2. Internet de las Cosas (IoT)

2.2.1. Definición

El Internet de las cosas (IoT, por sus siglas en inglés) se refiere al proceso mediante el cual objetos cotidianos se les da inteligencia artificial gracias a su conexión a internet. Además, se puede entender como una red de objetos físicos conectados a internet que pueden recopilar y compartir datos.[20] Esto permite que sensores conectados a la internet logren comunicarse y cooperen, lo que permite el monitoreo y control en tiempo real de todo, los cuales pueden ser desde estaciones meteorológicas hasta bombas y granjas. [19][20]

2.2.2. Arquitectura

La arquitectura de un sistema IoT para la agricultura se compone principalmente de tres capas: la capa de la percepción de la información, la capa de transmisión y la capa de aplicación del sistema.[20] A continuación se muestra un diagrama mostrando un ejemplo del sistema.

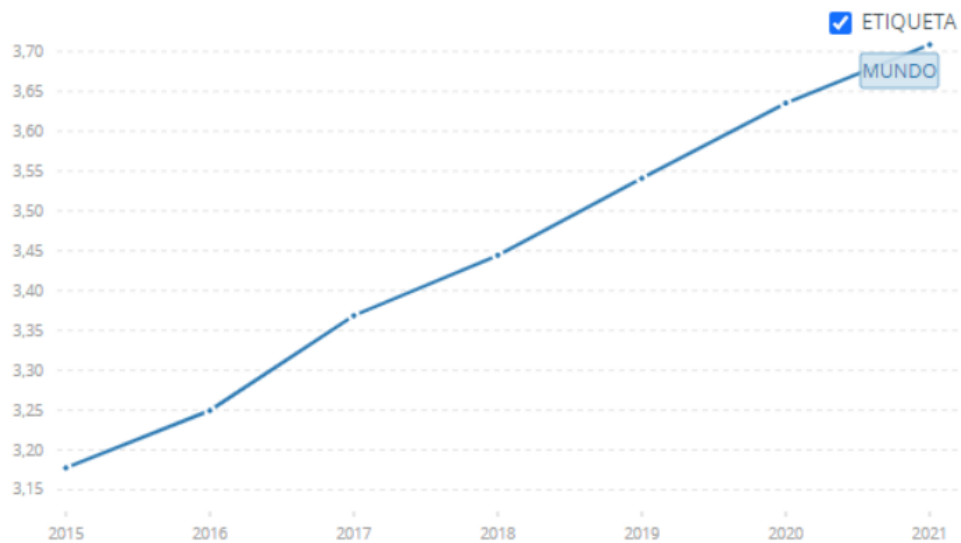


Figura 2.2: Arquitectura general de un sistema IoT para la agricultura[20]

Se debe encontrar una estandarización y esto suele ser complicado ya que debe haber normas adecuadas para todas las redes, aplicaciones o sectores de IoT. Las normas adecuadas suelen crearse en tres niveles a lo largo del proceso de estandarización del IoT agrícola.

- Normalización de la capa de percepción: La capa de percepción consta de componentes de control de la percepción, pasarelas de control de la percepción y redes de sensores que perciben y regulan el entorno físico y transforman la información en un formato de datos definido antes de enviarla a la capa de red. [20] Normalizar esta capa se refiere a las características comunes y los requisitos técnicos en términos de interfaces, comunicaciones y redes, procesamiento colaborativo de la información, soporte de servicios de información, seguridad y privacidad de la red, coherencia y pruebas de interoperabilidad.

- Normalización de la capa de transmisión: la capa de transmisión de IoT se separa en la capa de transmisión de comunicación por cable y la capa de transmisión de comunicación inalámbrica. Una red de área local inalámbrica de larga distancia, una red de área local inalámbrica de corta distancia y una red de área local inalámbrica de distancia ultracorta constituyen la capa de comunicación. [20] La normalización de esta capa crea normas de transmisión de información y protocolos de interfaz comunes para diversos sistemas de transmisión. Establece la normalización de la conexión de la información en diversas modalidades de comunicación para lograr la intercomunicación de la información en diferentes formas de transmisión.

-Normalización de la capa de aplicación: La capa de aplicación procesa la información a través de la plataforma de computación en nube. Puede computar, analizar y extraer datos adquiridos en la capa de percepción para diversos fines. [20]La normalización está estrechamente ligada al proceso de gestión y a la forma de procesar la información.

A continuación se muestra la figura de las capas mencionadas anteriormente.

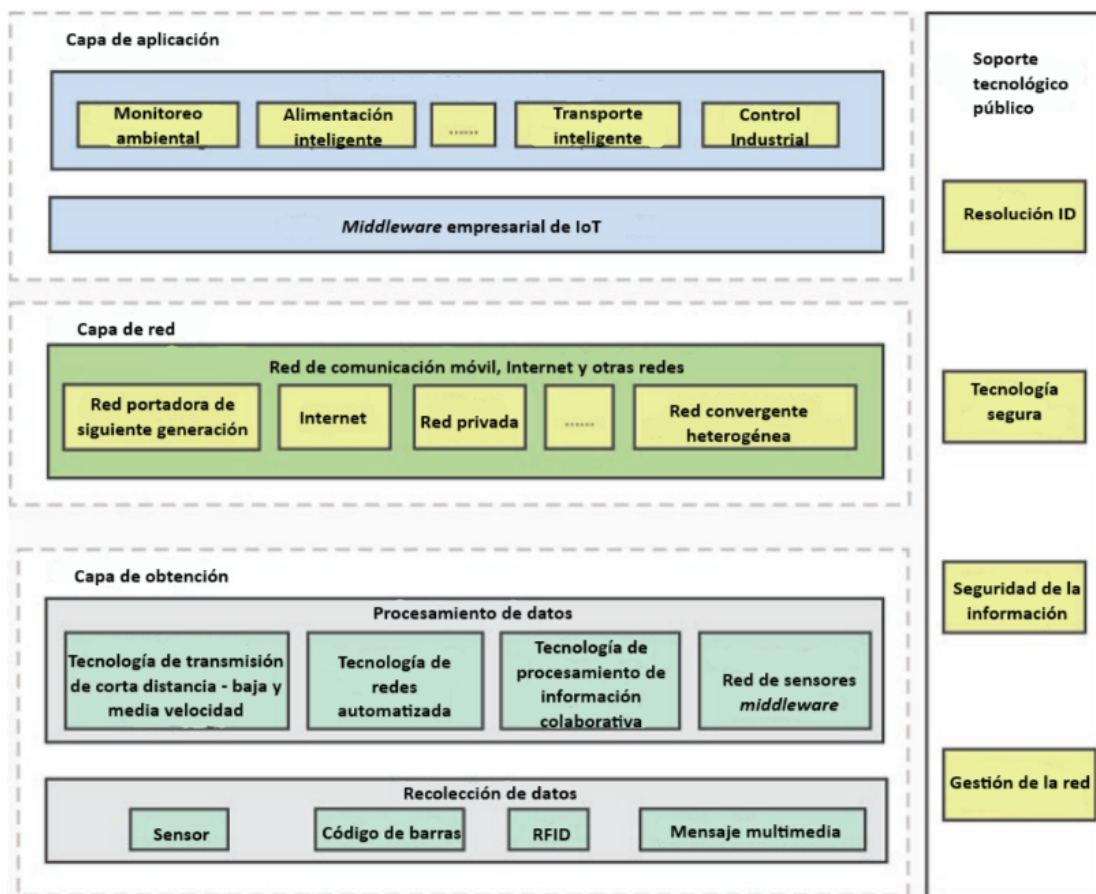


Figura 2.3: Arquitectura estandarizada de un sistema IoT para la agricultura[20]

2.2.3. Ventajas en la agricultura

Al implementar un sistema de Internet de las cosas se pueden obtener los siguientes beneficios [21]:

-Mayor eficiencia: Permite a los agricultores monitorear sus cultivos y el entorno circundante en tiempo real, lo que les permite ser más proactivos y tomar decisiones más informadas. Además, los robots y otros dispositivos IoT pueden realizar tareas como el riego, la fertilización y la cosecha de manera más eficiente que los métodos tradicionales.

-Mayor productividad: Puede ayudar a los agricultores a producir más alimentos de alta calidad al proporcionar información detallada sobre el clima, la calidad del suelo y el crecimiento de los cultivos. Esto les permite ajustar sus prácticas de cultivo para maximizar la producción y la calidad de los cultivos.

-Reducción de costos: Puede ayudar a los agricultores a reducir los costos al optimizar el uso de recursos como el agua, la energía y los fertilizantes. Además, la automatización de tareas puede reducir la necesidad de mano de obra, lo que puede ahorrar dinero a largo plazo.

-Mejora de la sostenibilidad: Puede ayudar a los agricultores a reducir su impacto ambiental al reducir el uso de recursos y minimizar la cantidad de productos químicos utilizados en el cultivo. Además, la tecnología IoT puede ayudar a los agricultores a monitorear y reducir el desperdicio de alimentos.

-Mayor seguridad alimentaria: Puede ayudar a los agricultores a producir alimentos más seguros y saludables al proporcionar información detallada sobre la calidad del suelo, el clima y el crecimiento de los cultivos. Esto les permite tomar medidas para prevenir enfermedades y contaminación en los cultivos.

2.3. Tecnologías de comunicación

Existen distintos tipos de tecnologías, algunas de las más utilizadas en un sistema IoT se mencionan a continuación.

2.3.1. Zigbee (IEEE 802.15.4)

Es un estándar de comunicación inalámbrica de baja potencia y corto alcance, diseñado específicamente para aplicaciones de redes de sensores inalámbricos (WSN) y dispositivos IoT. Este estándar define las especificaciones para la capa física (PHY) y la capa de control de acceso al medio (MAC) de la red. La capa PHY muestra las características de la transmisión de datos, como la frecuencia de operación, la modulación y la tasa de transmisión de datos. La capa MAC presenta cómo los dispositivos acceden al medio de transmisión y cómo se gestionan los conflictos de acceso al medio. Utiliza una topología de red de estrella, donde un dispositivo central (coordinador) actúa como punto de acceso a la red y los dispositivos periféricos (nodos) se comunican directamente con el coordinador. Además, el estándar proporciona mecanismos

de seguridad para proteger la privacidad y la integridad de los datos transmitidos en la red.[17, 22]. Sus características: velocidad de datos de 20-250kb/s, banda de frecuencia de 2,4 GHz, rango de transmisión de 10m-20 m y un consumo de energía bajo [23].

2.3.2. Wi-Fi (IEEE 802.11)

Es definida por el estándar IEEE 802.11 y que la mayoría de los dispositivos soportan los estándares como 802.11n y 802.11ac. Las redes Wi-Fi operan en ambientes sin cables y pueden operar en las frecuencias de 2,4 GHz y 5 GHz. En una red Wi-Fi, los dispositivos se comunican a través de ondas de radio que son transmitidas por un punto de acceso (AP) o un router. Los dispositivos se conectan al AP o router a través de una señal inalámbrica y pueden acceder a Internet o a otros dispositivos en la red. La comunicación Wi-Fi puede ser afectada por la interferencia de otras redes Wi-Fi cercanas, dispositivos electrónicos y otros obstáculos físicos [24]. Sus características: velocidad de datos de 1Mb/s a 7 Gb/s, banda de frecuencia de 5GHz a 60 GHz, rango de transmisión de 20m-100m, consumo de energía alto [23].

2.3.3. Bluetooth

Es una tecnología de comunicación inalámbrica que permite la conexión de dispositivos a corta distancia, y que el *Bluetooth Low Energy* (BLE) es una versión de bajo consumo de energía introducida en la especificación del protocolo Bluetooth 4.0. Los dispositivos se comunican a través de ondas de radio en la banda de frecuencia de 2,4 GHz. Asimismo, estos se emparejan a través de un proceso de descubrimiento y vinculación, y luego pueden intercambiar datos de forma inalámbrica. Esta tecnología se utiliza comúnmente en dispositivos móviles, como teléfonos inteligentes y tabletas, para la transferencia de archivos, la conexión de dispositivos periféricos, como auriculares y altavoces, y para la conexión de dispositivos IoT, como sensores y actuadores. Este puede ser afectado por la interferencia de otras redes Bluetooth cercanas, dispositivos electrónicos y otros obstáculos físicos. Sin embargo, la versión BLE ha sido diseñada para reducir el consumo de energía y mejorar la eficiencia de la conexión, lo que la hace ideal para dispositivos IoT que requieren una larga duración de la batería [17,24]. Sus características: velocidad de datos de 1-24Mb/s, banda de frecuencia de 24 GHz, rango de transmisión de 8m-10m y un consumo de energía muy bajo

[23].

2.4. Protocolos de redes

Se detallarán dos protocolos de mensajería ampliamente utilizados en los sistemas IoT para esta aplicación que se desea realizar.

2.4.1. Protocolo COAP

El protocolo *Constrained Application Protocol* (COAP) es un protocolo de aplicación diseñado para su uso en dispositivos con recursos limitados, como sensores y actuadores en el Internet de las cosas (IoT) [25, 26, 27]. Sus características principales son:

- Protocolo de transferencia de datos basado en UDP (User Datagram Protocol), lo que lo hace adecuado para redes de baja potencia y pérdida de paquetes.
- Soporta métodos de solicitud como GET, POST, PUT y DELETE. - Utiliza un formato de mensaje compacto y eficiente en términos de ancho de banda y energía.
- Proporciona opciones de seguridad como DTLS (Datagram Transport Layer Security) para proteger la comunicación entre dispositivos.
- Permite la suscripción y publicación de recursos, lo que lo hace adecuado para aplicaciones de IoT en tiempo real.
- Proporciona una capa de abstracción para acceder a recursos en dispositivos IoT, lo que facilita el desarrollo de aplicaciones.
- Es compatible con múltiples formatos de datos, como JSON y XML.
- Tiene una baja sobrecarga de red y es adecuado para redes de baja potencia y baja velocidad de transmisión de datos.

Su arquitectura se basa en una comunicación cliente-servidor, donde los dispositivos pueden actuar como clientes o servidores y acceder a recursos a través de direcciones URI. Además, se pueden utilizar componentes adicionales como proxies, observadores y grupos de multidifusión para mejorar la eficiencia y la escalabilidad de la red.

2.4.2. Protocolo MQTT

El protocolo MQTT (*Message Queuing Telemetry Transport*) es un protocolo de comunicación de aplicación ligera diseñado para su uso en dispositivos con recursos limitados, como sensores y actuadores en el Internet de las cosas (IoT) [25, 26, 27]. Sus características principales son:

- Es un protocolo de transferencia de datos basado en TCP (Transmission Control Protocol), lo que lo hace adecuado para redes de alta fiabilidad y baja pérdida de paquetes.
- Utiliza un modelo de publicación/suscripción, lo que permite que los dispositivos se comuniquen entre sí de manera eficiente y escalable.
- Proporciona tres niveles de calidad de servicio (QoS) para garantizar la fiabilidad de la comunicación entre dispositivos.
- Tiene un bajo consumo de energía, lo que lo hace adecuado para dispositivos alimentados por batería.
- Proporciona una alta interoperabilidad, lo que permite la comunicación entre dispositivos de diferentes fabricantes.
- Proporciona una alta flexibilidad, lo que permite la personalización de la red y la configuración de dispositivos según las necesidades de la aplicación.
- Es compatible con múltiples formatos de datos, como JSON y XML.
- Tiene un tamaño de encabezado fijo de dos bytes, lo que lo hace adecuado para dispositivos con recursos limitados.
- Proporciona una alta capacidad de red, lo que permite la conexión de múltiples dispositivos en una red.
- Es adecuado para aplicaciones de IoT en diferentes sectores, como la automatización del hogar, la salud, la seguridad, la agricultura y la industria.

Su arquitectura se basa en un modelo de publicación/suscripción, donde los dispositivos se comunican a través de un intermediario central llamado broker. [26] La arquitectura de MQTT consta de los siguientes componentes: 1. Cliente MQTT: es el dispositivo que publica o suscribe mensajes en el broker MQTT. 2. Broker MQTT: es el intermediario central que recibe los mensajes publicados por los clientes MQTT y los envía a los clientes suscritos a los mismos temas. 3. Tema MQTT: es una cadena de caracteres que identifica el contenido del mensaje publicado o suscrito. Los temas pueden ser jerárquicos y se separan por barras

diagonales (/). 4. Calidad de servicio (QoS): es un nivel de garantía de entrega de mensajes que se puede especificar en MQTT. Hay tres niveles de QoS: QoS 0 (entrega al menos una vez), QoS 1 (entrega al menos una vez y confirmación de entrega) y QoS 2 (entrega exactamente una vez y confirmación de entrega). Esta es adecuada para aplicaciones de IoT en diferentes sectores, como la automatización del hogar, la salud, la seguridad, la agricultura y la industria.

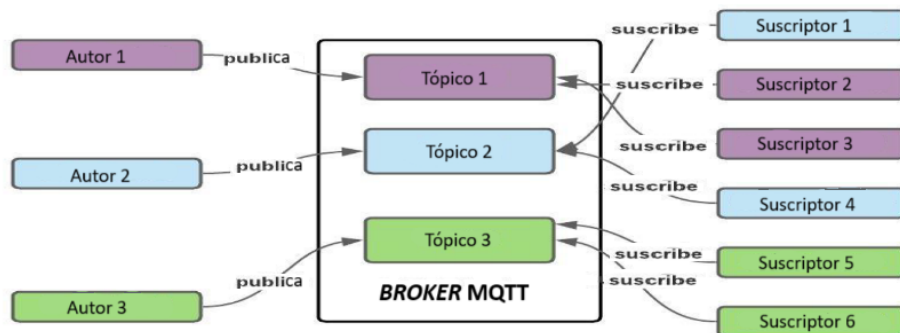


Figura 2.4: Arquitectura de MQTT[27]

2.5. Arquitectura del cliente IoT

2.5.1. Arquitectura No-OS

Se utiliza en dispositivos IoT simples, como sensores, termostatos y bombillas inteligentes, que no requieren un software complejo. Estos dispositivos no tienen un sistema operativo y su software se ejecuta directamente en el hardware. Además, esta arquitectura requiere menos memoria y procesamiento que otras opciones de arquitectura de software para dispositivos IoT. Los trade-offs son las capacidades limitadas y que no pueden ejecutar aplicaciones complejas. Además, no tienen la capacidad de actualización dinámica del software y la gestión de recursos es menos avanzada que en otras arquitecturas de software. También pueden ser menos seguros y menos eficientes en términos de energía que otras opciones de arquitectura de software para dispositivos IoT. [28]

2.5.2. Arquitectura Full-OS

Se utiliza un sistema operativo completo, como Linux, y se utiliza en dispositivos IoT más complejos, como tabletas o portátiles. Estos dispositivos tienen una mayor capacidad de

procesamiento y memoria, lo que les permite ejecutar aplicaciones más complejas. Además, la presencia de un sistema operativo completo trae beneficios como soporte incorporado para transferencias de archivos seguras, cuentas de usuario, gestión de dispositivos, actualizaciones de seguridad, herramientas de desarrollo maduras y numerosas otras características. Asimismo, hace posible ejecutar sin esfuerzo varios tipos de aplicaciones y servicios de terceros, incluidos los lenguajes de programación mencionados anteriormente. Los trade-offs tienen requisitos significativamente más altos de memoria y CPU en comparación con las arquitecturas No-OS o RTOS. Además, los requisitos significativamente más altos de consumo de energía hacen que sea difícil emplear dichos dispositivos en casos de uso que requieren operación con batería, excepto en soluciones del tamaño de una tableta o portátil con una capacidad de batería de al menos unos pocos miles de miliamperios-hora.[28]

2.5.3. Arquitectura RTOS

Se utiliza un sistema operativo en tiempo real (RTOS) que proporciona una gestión de recursos más avanzada que la arquitectura No-OS. Los dispositivos IoT que utilizan esta arquitectura son un poco más complejos y pueden tener más sensores y actuadores. El RTOS permite una gestión más eficiente de los recursos del dispositivo y una mejor capacidad de respuesta en tiempo real. Además, los requisitos de memoria y procesamiento son comparables a los de la arquitectura No-OS. Los trade-offs no tienen la capacidad de actualización dinámica del software y la gestión de recursos es menos avanzada que en otras arquitecturas de software. Además, pueden ser menos seguros y menos eficientes en términos de energía que otras opciones de arquitectura de software para dispositivos IoT. También pueden ser menos adecuados para dispositivos IoT que requieren una mayor capacidad de procesamiento y memoria, como tabletas o portátiles. [28]

2.6. Microcontroladores

A continuación dos de las marcas más conocidas para la producción de microcontroladores con WiFi integrado en sus microcontroladores.

2.6.1. Microchip Technology

Los microcontroladores de esta empresa poseen una amplia gama de microcontroladores. Estos varían según su número de pines de E/S, capacidad de memoria, velocidad del reloj y sus periféricos. Ofrece integrado convertidores analógico-digitales (ADC), convertidores digital-analógicos (DAC), UART, SPI, I2C, temporizadores, PWM, etc. Ofrece una variedad de modos de bajo consumo de energía y características de administración de energía. Su entorno de desarrollo integrado (IDE) es MPLAB X IDE. [29]

2.6.2. Espressif

Los microcontroladores Espressif utilizan Xtensa LX6 de doble núcleo, que proporciona un rendimiento sólido y poder procesar en tiempo real. Posee una amplia gama de periféricos integrados (ADC, DAC, UART, SPI, I2C, temporizadores, PWM entre otros. Su entorno de desarrollo integrado (IDE) ESP-IDF. Son compatibles con varios sistemas y protocolos, como MQTT, CoAP, SSL/TLS, IPv4, IPv6, entre otros.[30]

2.6.3. Protocolo de comunicación UART

La UART (Universal Asynchronous Receiver-Transmitter) es un dispositivo hardware que traduce datos de formato paralelo a formato serie. Puede transmitir y recibir datos en formato serie, y está compuesta por un transmisor y un receptor, así como otros componentes básicos. Tiene diferentes tipos de control de flujo, como el control de flujo por hardware completo, control hardware solo con RTS y CTS, y control de flujo por software (Xon-Xoff). Además, se utiliza en conjunción con estándares de comunicación serie como RS-232, RS-422 y RS-485, cada uno con sus propias características eléctricas y de transmisión. La norma RS-485 se utiliza en sistemas de comunicación de baja velocidad en aviones, sistemas de sonido, automatización de edificios y plantas de producción automatizadas.[31]

2.7. Sensores

Para esta aplicación de IoT específicamente para las macetas que se encuentran en los invernaderos se decidieron analizar los siguientes sensores.

2.7.1. Sensor de humedad de suelo

Existen distintos tipos de sensores de humedad de suelo[32]: capacitancia mide la humedad del suelo midiendo la capacitancia entre dos electrodos enterrados en el suelo; resistencia, que miden la humedad del suelo midiendo la resistencia eléctrica entre dos electrodos enterrados en el suelo; tensión, la humedad del suelo midiendo la tensión eléctrica generada por una celda de carga enterrada en el suelo; TDR (Time Domain Reflectometry), el cual se le dará un mayor enfoque. Su funcionamiento es el siguiente: emite una señal eléctrica que viaja a través de una sonda de medición insertada en el suelo. Cuando la señal eléctrica alcanza la interfaz entre el suelo seco y el suelo húmedo, parte de la señal se refleja hacia la sonda de medición. La sonda de medición mide el tiempo que tarda la señal reflejada en regresar a la sonda, lo que está relacionado con la constante dieléctrica del suelo y, por lo tanto, con la humedad del suelo. -La velocidad de propagación de la señal eléctrica está relacionada con la constante dieléctrica del suelo, que a su vez está relacionada con la humedad del suelo. La fórmula para calcular la humedad del suelo utilizando un sensor TDR es:

$$\theta = \frac{K - Kd}{Kw - Kd} \times 100 \% \quad (2.1)$$

- θ es la humedad del suelo en porcentaje
- K es la constante dieléctrica medida por el sensor TDR
- Kd es la constante dieléctrica del suelo seco
- Kw es la constante dieléctrica del agua

La constante dieléctrica del suelo seco (Kd) se puede medir antes de la instalación del sensor TDR, mientras que la constante dieléctrica del agua (Kw) es una constante conocida.

Posee las siguientes características:

- Rango de medición: se encuentra en el rango de 0 a 100 % de humedad del suelo.
- Precisión: se encuentra en el rango de $\pm 1 \%$ a $\pm 5 \%$ de humedad del suelo.
- Resolución: alta y puede medir cambios de humedad del suelo de hasta 0.1 %
- Velocidad de medición: rápida y puede medir la humedad del suelo en tiempo real.
- Profundidad de medición: Se encuentra en el rango de 10 cm a 1 metro.
- Condiciones ambientales: TDR es resistente a las condiciones ambientales y puede funcionar en suelos con alta conductividad eléctrica, alta salinidad y alta temperatura.

2.7.2. Sensor de temperatura de suelo

Existen distintos tipos de métodos para medir la temperatura del suelo a través de sensores de temperatura sumergibles. A continuación se mencionan los siguientes tipos. [33] RTD (Detector de Temperatura Resistivo): Este sensor está basado en la variación de la resistencia de un conductor con la temperatura. Está compuesto por una aleación de metales como el platino, cobre, níquel y molibdeno. Su funcionamiento se basa en la agitación térmica que causa la dispersión de electrones, lo que a su vez afecta la resistencia del material. Termocuplas/Termopares: Son sensores de temperatura eléctricos que funcionan con dos alambres de distinto material unidos en un extremo. Al aplicar temperatura en la unión de los metales se genera un voltaje muy pequeño que aumenta con la temperatura. Sensor de Temperatura Tipo Infrarrojo: Este sensor mide la temperatura de la superficie mediante la conversión de energía térmica radiada en una señal eléctrica. Cuenta con una incertidumbre de 0,2 °C a 95 % de precisión y una ventana de germanio para minimizar los efectos de bandas de agua. Además, tiene un amplio rango de calibración de temperatura y puede soportar condiciones climáticas extremas. Termistores: Son sensores de temperatura cuyo funcionamiento se basa en la variación de la resistividad que presenta un semiconductor con la temperatura. Existen dos tipos: NTC (Negative Temperature Coefficient) y PTC (Positive Temperature Coefficient).

2.8. Red de sensores inalámbricos

Con parte de lo descrito anteriormente en este capítulo, se debe conformar una solución conjunta.

2.8.1. Definición

Es una red de múltiples dispositivos distribuidos espacialmente, que utilizan sensores para controlar diversas condiciones en distintos puntos, entre ellas la temperatura, sonido, vibración o contaminantes. Los dispositivos deben ser unidades autónomas para esta aplicación, específicamente, se usa un nodo de sensor [34].

2.8.2. Topología

La topología se refiere a cómo se adoptan los componentes de la red tanto del hardware como de la transmisión de la información [34, 35, 36]. A continuación, se detallarán algunas topologías. Topología tipo estrella se basa en cada nodo se conecta al Gateway(dispositivo utilizado en la red); topología tipo árbol, cada nodo se conecta a un nodo de mayor jerarquía en el árbol y ,después, al Gateway, datos enrutados desde un nodo de menor jerarquía hasta el Gateway mediante una ruta establecida; topología tipo malla, cada nodo se puede conectar a múltiples nodos en el sistema y pasar los datos por el camino disponible de mayor confiabilidad. En la siguiente figura, se muestran las topologías descritas anteriormente.

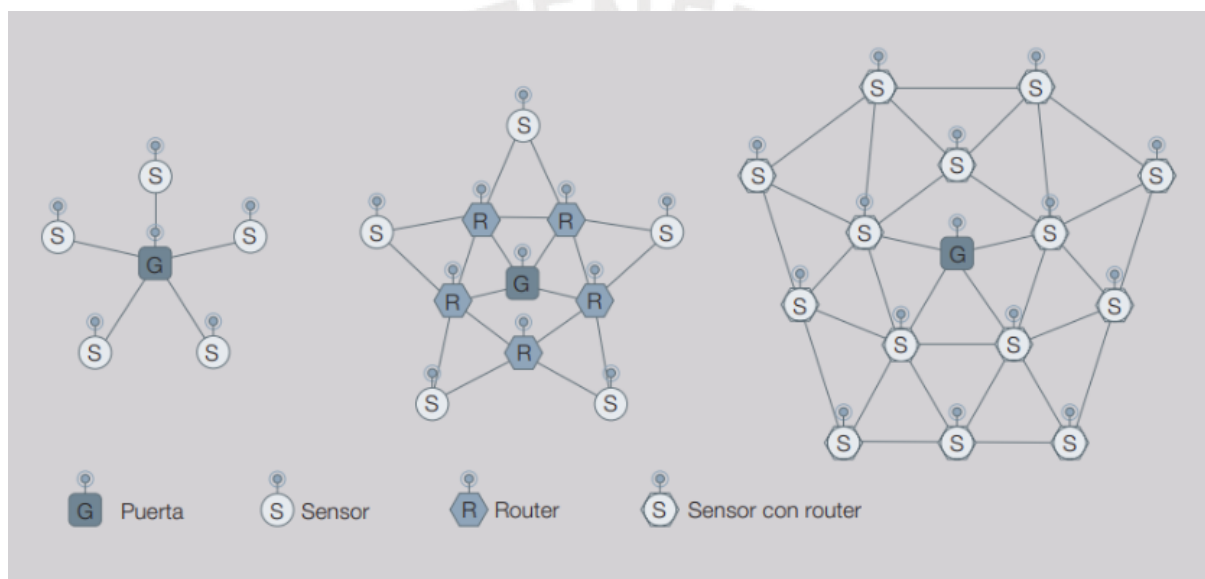


Figura 2.5: Topologías [34]

En la siguiente tabla se muestran las ventajas y desventajas

| Topología | Ventajas | Desventajas |
|-----------|--|---|
| Estrella | <ul style="list-style-type: none"> Gasto energético homogéneo. Sencillez y rapidez en el desarrollo. Fácil de desplegar. Robustez de la red. Baja latencia. | <ul style="list-style-type: none"> De acuerdo a la aplicación no siempre es la más funcional. Escalabilidad baja, posibles colisiones cuando se aumentan el número de nodos Si falla el nodo central cae la red |
| Malla | <ul style="list-style-type: none"> Menor costo de instalación la no necesitar una gran cantidad de nodos routers para alcanzar escalabilidad. Si fallan uno o varios nodos la red sigue funcionando ya que existen rutas alternativas. | <ul style="list-style-type: none"> Complejidad del sistema alto. Alta cantidad de colisiones. Empeora la latencia de la red. Falencia en pruebas de laboratorio e implantación real. |
| Árbol | <ul style="list-style-type: none"> Ventajas de topología tipo árbol. Alta escalabilidad. Menor porcentaje de colisiones. | <ul style="list-style-type: none"> La inclusión de nodos encaminadores puede encarecer significativamente la solución final. Puede caer una parte significativa de la red al caer un nodo router. Costoso y difícil de desarrollar algoritmo de enrutamiento dinámico. Baja fiabilidad de los algoritmos de enrutamiento. |

Cuadro 2.1: Tabla comparativa de las tres topologías [36]

Asimismo se puede notar una tabla comparativa entre la topología estrella y árbol.

| Parámetros | Topología Estrella | Topología Árbol |
|--|--------------------|-----------------|
| PDR (%) | ✓ 97,9 | 43,95 |
| Rendimiento(Kbps) | ✓ 0,7542 | 0,5147 |
| Retado extremo a extremo (ms) | ✓ 0,0162 | 0,3416 |
| Consumo de Energía (J) | 46662,17 | ✓ 46645,58 |
| Área de cobertura de sensado (m ²) | ✓ 12566,4 | 6283,2 |

Cuadro 2.2: Tabla de resultados de topología estrella y arbol [35]

2.8.3. Nodo de sensor

Un nodo de sensor es un dispositivo electrónico que se utiliza para medir y recopilar datos de su entorno. [36] Estos nodos pueden ser pequeños y portátiles, y están equipados con sensores que pueden medir diferentes variables, como temperatura, humedad, presión, luz,

sonido, movimiento, entre otros. Asimismo, pueden ser equipados con dispositivos de comunicación inalámbrica, como Wi-Fi, Bluetooth, que les permiten enviar los datos recopilados a otros dispositivos o a una red de sensores inalámbricos [34]. Se utilizan en una amplia variedad de aplicaciones, como la agricultura de precisión, la monitorización ambiental, la seguridad y vigilancia, la salud y el bienestar, entre otros. En la siguiente figura se detalla lo básico para un nodo de sensor.

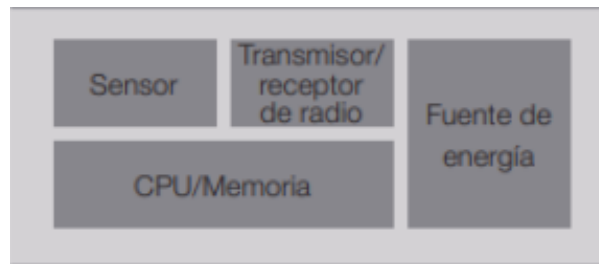


Figura 2.6: Nodo de sensor[34]

2.9. Interfaz de usuario

Con el objetivo de conseguir una interfaz de usuario amigable, es esencial contar con una forma sencilla de presentar la información y que la aplicación sea intuitiva de manejar. La simplicidad de la aplicación se traduce directamente en una experiencia más placentera para el usuario. Además, se debe disponer de un mecanismo eficiente para almacenar la información y acceder a ella cuando sea necesario.[2017] En este sentido, se presentarán a continuación los pilares fundamentales para alcanzar estos objetivos la visualización de información para el usuario (front-end) y el control integral de la información (back-end). Por una parte, el front-end se centra en proporcionar una interfaz gráfica intuitiva que permita al usuario interactuar de manera efectiva con la información presentada. Un diseño limpio y accesible contribuirá significativamente a mejorar la experiencia del usuario. Por otro lado, el back-end desempeña un papel crucial al encargarse de gestionar y controlar toda la información. Esta parte del sistema se asegura de que los datos se almacenen de manera eficiente, estén disponibles cuando se necesiten y se gestionen de manera segura. La colaboración efectiva entre el front-end y el back-end es esencial para lograr una aplicación completa y amigable que satisfaga las necesidades del usuario de manera integral.

2.9.1. Front-end

En esta sección, el usuario puede acceder y visualizar de manera efectiva toda la información relevante. Se define como una aplicación REST, donde las solicitudes del cliente son recibidas a través de las interfaces de aplicaciones empresariales tradicionales basadas en web o clientes robustos, en lugar de depender únicamente de una capa de API. La capa de interfaz de usuario de la aplicación se despliega como una aplicación web independiente que se conecta de manera remota a componentes de servicio desplegados de forma separada (funcionalidad empresarial) mediante interfaces simples basadas en REST [38].

Además, el front-end asume la responsabilidad de desarrollar la interfaz gráfica con la cual el usuario interactúa directamente, ya sea en software, sitios web o aplicaciones. Esto abarca la selección de colores, textos e imágenes que el usuario percibe y con los que interactúa mientras navega por el sistema [39]. El front-end también colabora estrechamente con la gestión de datos para asegurar una experiencia de navegación fluida y una usabilidad óptima del sistema.

2.9.2. Back-end

En esta sección, se destaca la falta de visibilidad del proceso por parte del usuario, ya que no se presenta de manera visible. Para abordar esta limitación, se enfocará en la metodología de una API REST, que constituye una interfaz de programación de aplicaciones siguiendo los principios de la arquitectura REST. Esta metodología utiliza métodos estándar de HTTP, como GET, POST, PUT y DELETE, para facilitar la comunicación entre sistemas.

Las API REST se fundamentan en la representación de recursos, identificados por URLs únicas, y emplean el estado del recurso para llevar a cabo diversas operaciones. Esta estructura posibilita una comunicación eficiente y flexible entre sistemas, lo que ha llevado a su amplia adopción en el desarrollo de aplicaciones web y móviles [38]. Esta no solo permitirá mejorar la transparencia del proceso para el usuario, sino que también ofrecerá una base sólida para la interacción efectiva entre sistemas.

2.10. Retos en la implementación

Como en todo sistema, al implementar una solución basada en IoT se deben considerar los siguientes aspectos:

- La gestión de los recursos de red en IoT puede resultar compleja, ya que los dispositivos a gran escala pueden unirse a la red en cualquier momento [40].
- Los dispositivos inteligentes de baja potencia pueden transmitir datos durante largos periodos antes de que se agote su batería [39].
- El rendimiento de los dispositivos IoT puede verse afectado por problemas de escalabilidad [38].
- La eficiencia en la agregación de datos puede disminuir debido a que los dispositivos IoT intercambian información en plataformas heterogéneas [39].
- La privacidad de los datos no siempre puede garantizarse, dado que muchos dispositivos IoT operan en entornos de red abiertos [39].
- La disponibilidad de servicios de Internet es un factor crítico para la comunicación basada en IoT [40].
- El desarrollo rentable de sensores IoT sigue siendo un desafío importante [40].

Capítulo 3

Diseño del sistema IoT para invernadero en el INIA

Para este capítulo de diseño se realizaron entrevistas al personal del INIA de la subdirección de investigación en estudios especiales (SDIEE) al mando del Dr Enrique Meseth Macchiavello y, en especial, con el ingeniero agrónomo Gabriel Martínez Vidal, quien detalló específicamente los requerimientos para la solución.

3.1. Requerimientos

A continuación, se presentarán los requerimientos del invernadero para la solución a diseñar.

- La resolución del sensor de temperatura de suelo debe ser mínima de 0.1°C y precisión mínima de 5 %.
- La resolución del sensor de humedad de suelo debe ser mínima de 0.1 de humedad y precisión mínima de 5 % de humedad.
- El rango de humedad debe estar entre 0-60 % de humedad del suelo.
- Se debe poder calibrar la solución para poder funcionar en suelos de distinto tipo de composición.
- El sistema debe ser diseñado para un total de 50 macetas.
- Se realizará un prototipo con 2 macetas como prueba de concepto.
- Obtención de los parámetros de suelo. Para este caso se requiere, de humedad del suelo y temperatura de suelo.

- Se requiere de contar con una conexión WiFi en el invernadero.
- El dispositivo debe mandar los datos cada hora.
- La fuente de alimentación debe ser mediante una batería ya que debe ser inalámbrica.
- La página web debe contar con una autenticación solo para personal autorizado.
- La comunicación entre los dispositivos debe estar correctamente cifrada.
- La toma de muestras de los datos debe realizarse una vez por hora.
- La visualización de la temperatura y humedad de cada maceta debe ser mediante gráficas .
- La visualización de los datos solicitados debe ser mediante una página alojada en la web.
- El sistema de alertas debe avisar de valores alarmantes al usuario.
- El dispositivo debe ser totalmente inalámbrico para colocarlo en la maceta.
- La operación del dispositivo debe contar una duración mínima de 1 mes.
- Se debe poder visualizar el estado de la batería.

3.2. Diagrama de bloques

A continuación, se presentará la solución escogida en base a los estados del arte detallados anteriormente. Asimismo, se debe detallar que la solución está siendo planteada para el desarrollo de 50 macetas. A continuación, se mostrará el diagrama de bloques para un nodo sensor para una maceta. Este se debe replicar para todos los nodos de sensores de las otras 49 macetas.

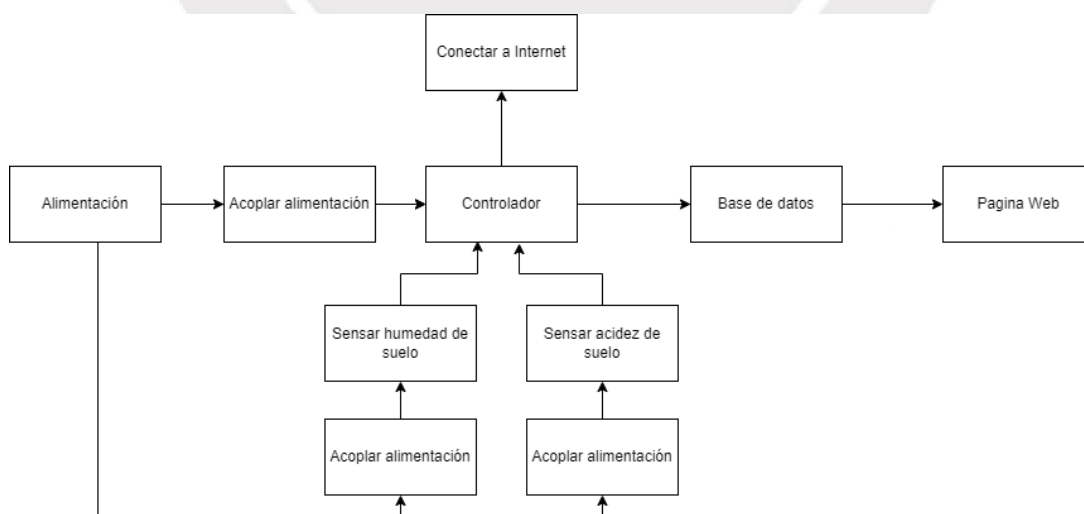


Figura 3.1: Diagrama de bloques de solución propuesta[Elaboración propia]

En la solución planteada anteriormente, se plantea utilizar lo siguiente. La utilización de una fuente alimentación, la cual es sumamente importante aplicarla correctamente para lograr sensor los parámetros del suelo y ,asimismo, el microcontrolador. Posteriormente, la información del suelo debe ser procesada por el microcontrolador. Se debe tener en cuenta, que el microcontrolador debe lograr conectarse a Internet para realizar el envío de la información hacia una base de datos. Se plantea utilizar una página web para realizar de intermediario y no exponer la seguridad del sistema, ya que es una práctica muy utilizada. Finalmente, la página web enviará la información recibida a la base de datos para su posterior almacenamiento y accede a esta información guardada para poder visualizar lo que el usuario desee ver. Con esto se busca lograr la obtención de la interfaz amigable para el usuario.

3.3. Matriz de funciones

En esta sección, se hablará acerca de los distintos dominios que abarcan la matriz de funciones. A continuación se muestra el diagrama de funciones.

En primer lugar, se muestra de manera general la matriz de funciones la primera parte para toda la solución. Esta abarca toda la parte electrónica.

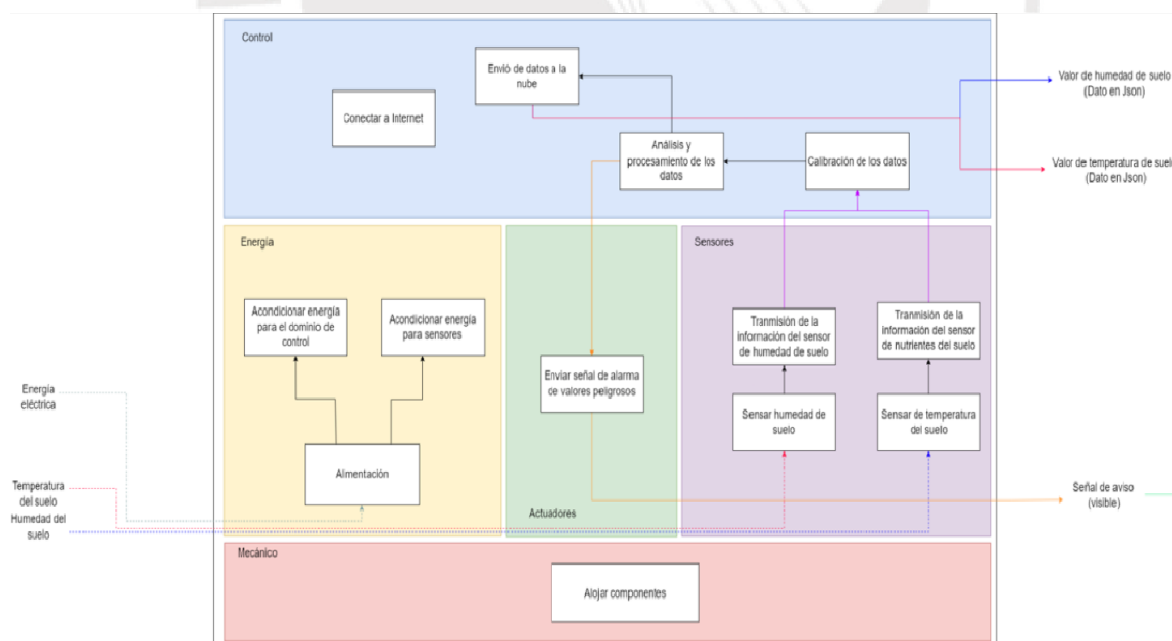


Figura 3.2: Matriz de funciones parte 1 [Elaboración propia]

En segundo lugar, se muestra de manera general la matriz de funciones la primera parte para toda la solución. Esta abarca toda la parte de la página web.

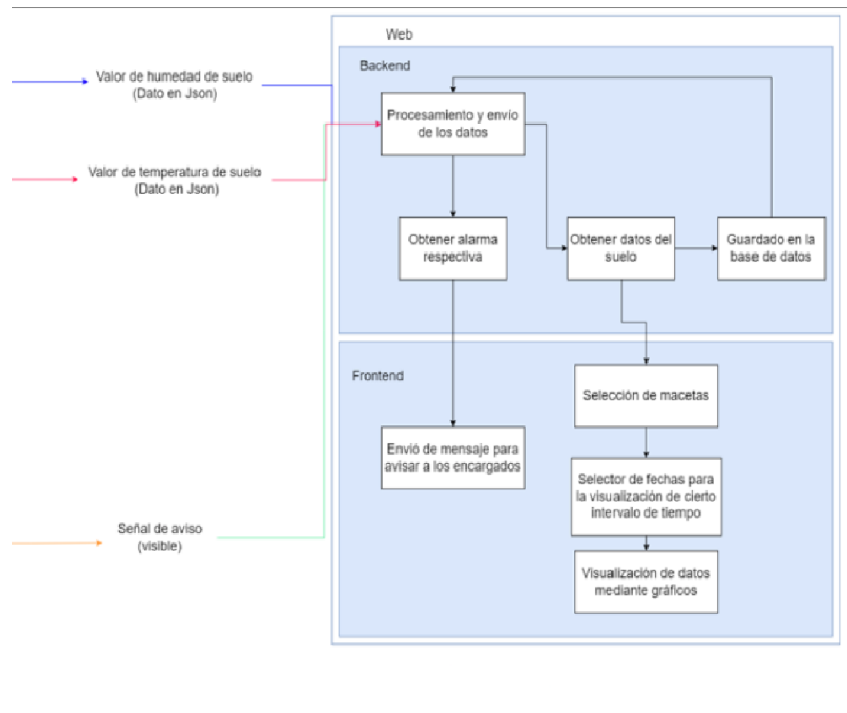


Figura 3.3: Matriz de funciones parte 2[Elaboración propia]

A continuación empezaremos con los dominios.

3.3.1. Dominio mecánico

Para el dominio mecánico, se está tomando en consideración una función: alojar componentes, la cual se centra en cómo va a ser la disposición de todos los componentes que tendrá la solución. Asimismo, contará con una estructura que cubrirá los componentes del ambiente externo, mas no debe ser aislado totalmente, ya que debe incluirse la salida de los sensores para el exterior.



Figura 3.4: Dominio mecánico[Elaboración propia]

3.3.2. Dominio de energía

Para el dominio de energía se está considerando tres funciones. Se mencionan a continuación:

Almacenamiento para la alimentación del sistema: En esta función, se debe contar con la existencia de una forma de poder almacenar la energía eléctrica. Acondicionar energía para el

dominio de control: En esta función, se debe contar con el acondicionamiento correcto para obtener la energía suficiente para la parte de control del sistema. Acondicionar energía para el

dominio de control: En esta función, se debe contar con el acondicionamiento correcto para obtener la energía suficiente para los sensores del sistema.

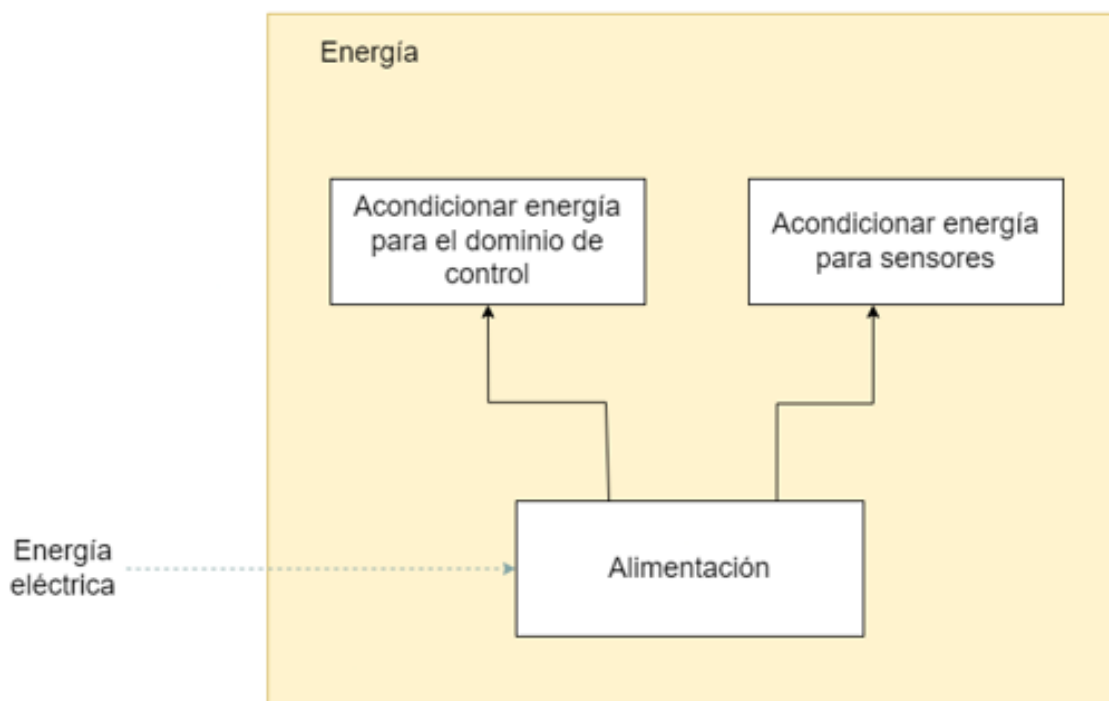


Figura 3.5: Dominio de energía[Elaboración propia]

3.3.3. Dominio de actuadores

Para el dominio de actuadores se está considerando solo una función: Enviar señal de alarma de valores peligrosos. Esta se encargará de poder recibir la señal de alarma y enviarla para que pueda ser visible al usuario.

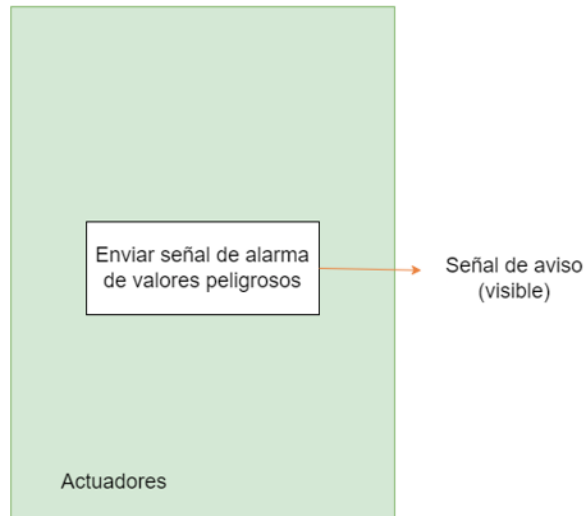


Figura 3.6: Matriz de funciones[Elaboración propia]

3.3.4. Dominio de sensores

Para el dominio de sensores, se está considerando cuatro funciones. Se mencionan a continuación:

Sensar humedad de suelo: Esta función debe poder recibir el parámetro de la humedad del suelo para poder posteriormente transmitirla. Sensar temperatura de suelo. Esta función debe poder recibir el parámetro de la temperatura del suelo para poder posteriormente transmitirla.

Transmisión de la información del sensor de humedad de suelo: Esta función debe permitir la transmisión de los datos recaudados de la humedad del suelo para su posterior procesamiento.

Transmisión de la información del sensor de temperatura de suelo: Esta función debe permitir la transmisión de los datos recaudados de la temperatura del suelo para su posterior procesamiento.

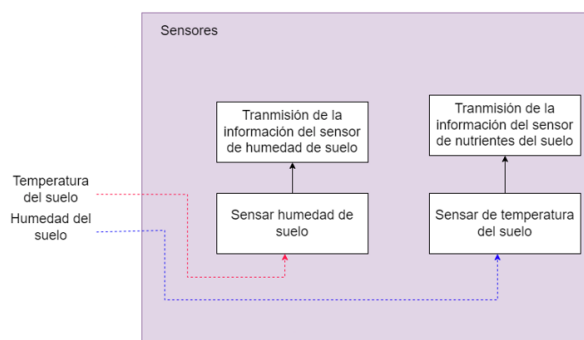


Figura 3.7: Dominio de sensores[Elaboración propia]

3.3.5. Dominio de control

Para el dominio de control se está considerando cuatro funciones. Se mencionan a continuación:

Conectarse a Internet: En esta función, se encargará de entablar la conexión a Internet.

Calibración de datos: En esta función, se debe contar con la calibración correspondiente para los sensores que se utilizarán. Esto es muy importante, para obtener mediciones reales y sea efectiva la solución.

Análisis y procesamiento de los datos: En esta función, se encargará el procesamiento correspondiente de los datos de los sensores para poder empaquetar todos esos datos a través de un arreglo JSON para su posterior envío. Asimismo, se encargará de analizar esa información para poder determinar si hay valores alarmantes para enviar una señal de alarma valores peligrosos. Finalmente, también debe encargarse de obtener un valor de referencia para poder hacer el envío de los datos cada cierto tiempo.

Envío de datos a la nube: En esta función, se debe tener en cuenta la implementación necesaria para poder enviarla a una nube, en la cual se depositará toda la información recaudada.

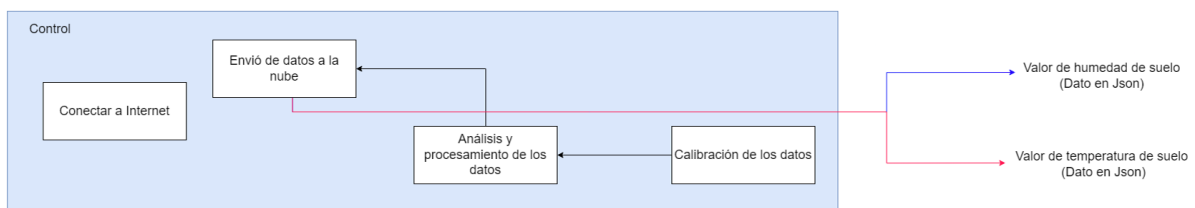


Figura 3.8: Dominio de control[Elaboración propia]

3.3.6. Dominio de la web

Para este dominio, el cual se encargará de realizar todo el apartado de la nube. Para esta sección se utilizarán dos subsecciones Frontend, la cual es la parte, la cual el usuario verá y podrá interactuar y el Backend, la cual se encarga de la realización de las funciones de la aplicación web.

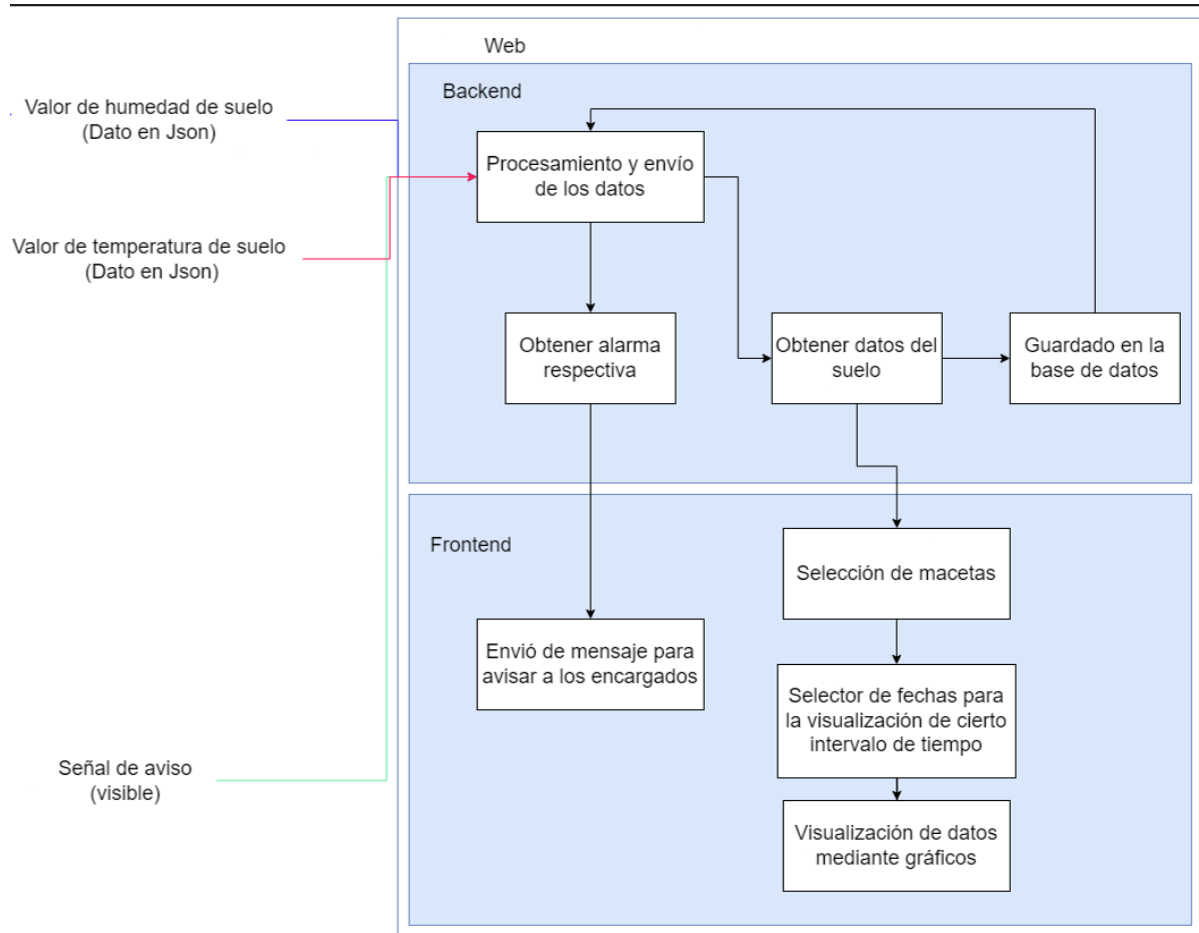


Figura 3.9: Dominio de la web[Elaboración propia]

3.3.6.1. Frontend

Para este apartado se utilizarán cuatro funciones mencionadas a continuación:

Envío de un mensaje a los encargados: En esta función, debe encargarse de mostrar la información recibida mediante un mensaje acerca de qué maceta proviene la alarma y sus datos respectivos de humedad de suelo y temperatura de suelo a los encargados. **Selección de maceta:** Se debe poder seleccionar los datos de una maceta para poder recibir todos los datos de esa información. **Selector de fechas para la visualización de cierto intervalo de tiempo:** En esta función, debe ser capaz de seleccionar en un intervalo de fechas para poder realizar la visualización de los datos. **Visualización de datos mediante gráficos:** En esta función, se debe poder mostrar los datos recibidos del selector de fechas mediante gráficos estadísticos para los valores de temperatura y humedad del suelo. Esto debe ser implementado para cada maceta

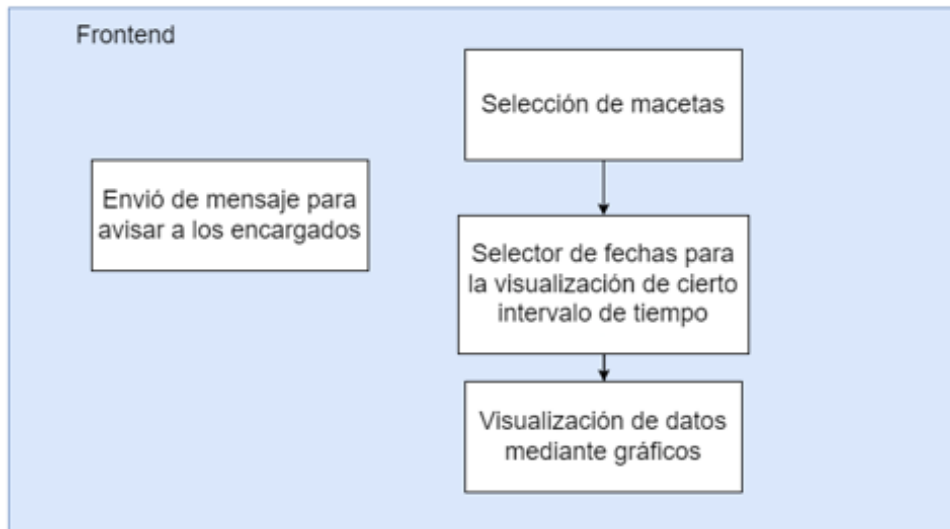


Figura 3.10: Frontend[Elaboración propia]

3.3.6.2. Backend

Para este apartado se utilizarán tres funciones mencionadas a continuación:

Procesamiento y envío de los datos: En esta función, se debe poder realizar toda la lógica de gestiones y controles para poder distribuir la información.

Obtener alarma respectiva: En esta función, se debe poder obtener los datos de la alarma como los datos relevantes de esta para su identificación correspondiente.

Obtener datos del suelo: En esta función, se debe poder obtener los datos respectivos del suelo.

Guardado en la base de datos: En esta función, se debe guardar los datos en una base de datos.

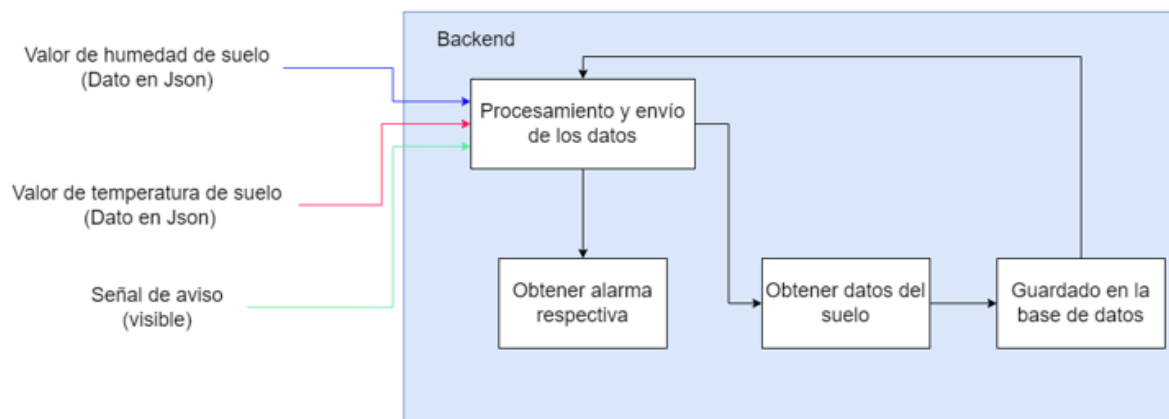


Figura 3.11: Backend[Elaboración propia]

3.4. Matriz morfológica

En esta sección se tendrá detalladamente la selección de las tecnologías correspondientes para la realización y viabilidad de la solución. Se ha tomado en cuenta el estado del arte y su fundamento teórico para la selección correspondiente. A continuación se realizará una comparación para cada función analizada previamente.

3.4.1. Electrónica

A continuación, se presentarán las selecciones realizadas para todo el nodo de sensor.

-Análisis y procesamiento de los datos: Para poder realizar estas tareas se plantea utilizar un microcontrolador, debido a que sus características la harían portable y de menor consumo que la utilización de otro tipos de procesamientos. Asimismo, se incluirá la comunicación serial necesaria para realizar la programación de este mediante los pines respectivos de programación.

-Sensor humedad de suelo y temperatura de suelo: Para poder realizar la medición respectiva de la humedad y temperatura del suelo se plantea utilizar un sensor integrado, el cual pueda brindar estos dos parámetros. Asimismo, se toma esta consideración, ya que un sensor integrado puede ampliarse para obtener más datos del suelo en un futuro.

-Transmisión de la información de los sensores: Se debe analizar y tomar en cuenta la salida que posee el sensor. Para este caso, nos inclinamos por la parte de comunicación I2C.

- Envío de datos a la nube: Debido a la restricción de tecnología disponible a utilizar, ya que únicamente se puede utilizar WiFi en las instalaciones del INIA, se debe basar la solución en el protocolo de comunicación de MQTT para poder realizar la conexión entre el microcontrolador y el Internet. Este protocolo es el más adecuado en condiciones de practicidad y costos energéticos.

-Calibración de los datos: Para realizar la calibración de los datos se utilizarán equipos brindados por el INIA para poder asemejar sus resultados que obtienen con estos sensores con los valores que se buscan obtener con los sensores. Se utilizará el sensor laquatwin para la calibración de estos.

- Enviar señal de alarma de valores peligrosos: Para realizar esta función, según el

requerimiento por parte del INIA, desean que pueda ser mediante un servicio de mensajería instantánea.

- Alimentación del sistema: Para poder realizar esta tarea se buscará utilizar una batería que permita brindarle al sistema ser totalmente inalámbrico para la colocación de esta en las macetas.

- Acondicionar energía para el dominio de control y sensores. Se usará un regulador de voltaje para brindar el voltaje de 3.3V para la operación del sistema.

3.4.2. Nube

A continuación, se presentarán las selecciones realizadas para la parte de la nube del sistema. En primer lugar, se mencionan para la parte del Backend:

-Procesamiento y envío de los datos: En este caso, se utilizará el protocolo Http, debido a su practicidad y eficiencia. Asimismo, se usarán formatos JSON para manejar toda la información. Esto ayudará para la realización de los procedimientos respectivos y enlazamientos de toda la información mediante una API, para poder gestionar y procesar la información.

-Base de datos: Para poder seleccionar la base de datos correspondiente se debe tener en cuenta la estructura de toda la información. Como toda la información está parcialmente entendida y por motivos de practicidad y ordenamiento, se utilizará una base de datos no relacional, debido a que no debemos conocer completamente la información con la que vamos a trabajar para poder guardarla correctamente y acceder a ella. A continuación se muestra cómo esta segmentado:

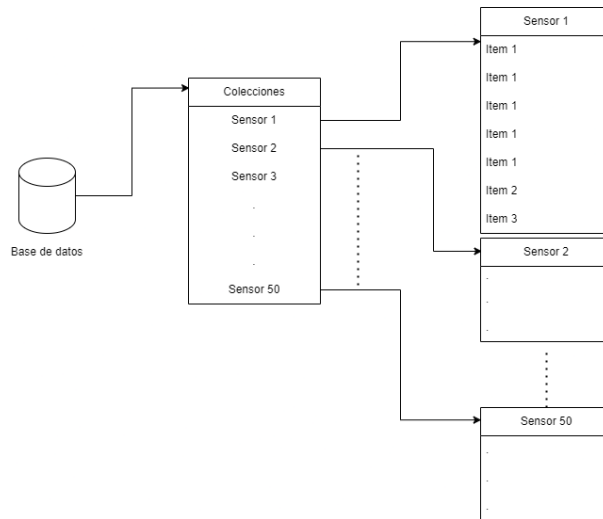


Figura 3.12: Segmentación base de datos no relacional[Elaboración propia]

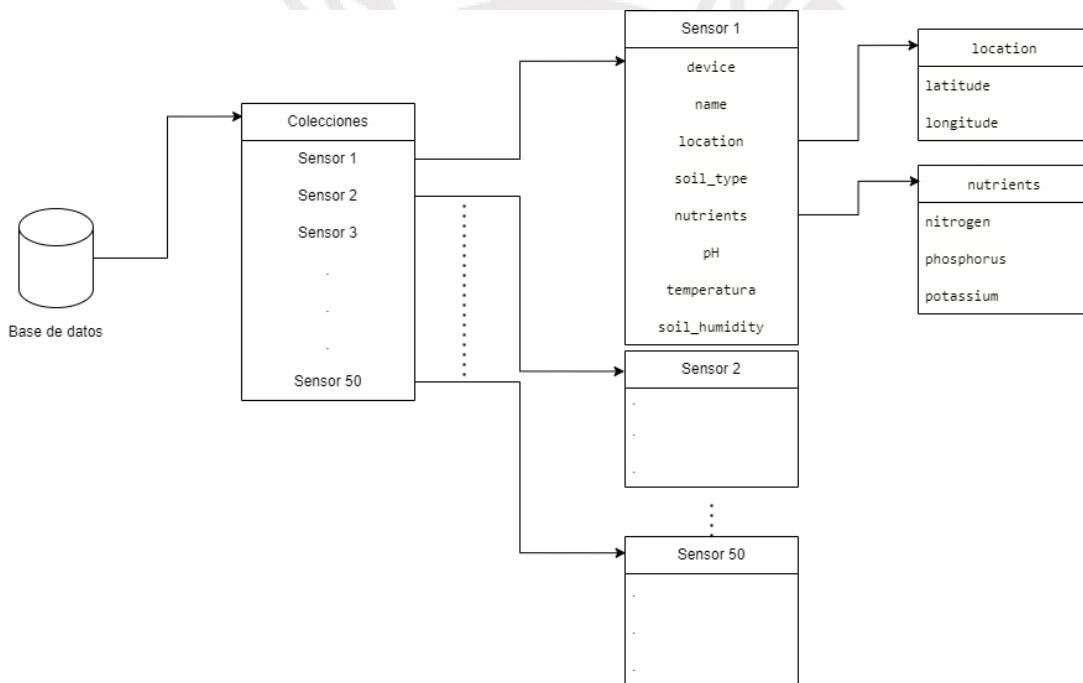


Figura 3.13: Estructura de datos para la base de datos[Elaboración propia]

-Obtener datos del suelo: Haremos una solicitud a nuestra base de datos, mediante una API para obtener los datos del suelo.

-Obtener datos del suelo: Haremos una solicitud a nuestra base de datos, mediante una API para obtener los datos del suelo.

En segundo lugar, se mencionan para la parte del Frontend:

- Selección de macetas: Haremos una solicitud a nuestra base de datos, mediante una API para

poder visualizar los datos del suelo de una maceta en específico.

- Selector de fechas para la visualización de cierto intervalo de tiempo: Para visualización de los datos en un intervalo de tiempo especificado.

- Visualización de datos mediante gráficos: Para la visualización de los gráficos mediante gráficos estadísticos se mostrarán a través de una pantalla, que será netamente codificada con código. A continuación se muestra la estructura para la página.

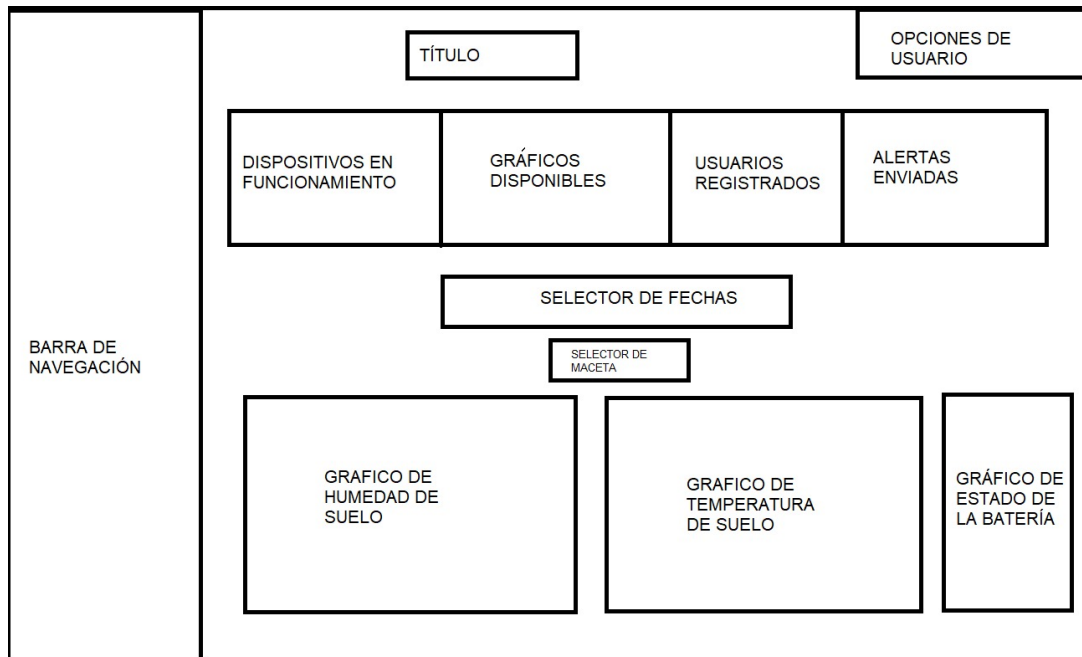


Figura 3.14: Estructura de para la página web[Elaboración propia]





- Envío de un mensaje a los encargados: Para realizar el envío de la alarma recibida optamos por utilizar una API de Telegram, a través de un bot para realizar el envío de mensajes de alarma hacia los encargados correspondientes.

3.5. Selección de componentes

Partiendo de todo lo descrito en la matriz morfológica, se procederá a realizar comparación de opciones para encontrar la mejor alternativa de solución. Se determinaron las siguientes secciones para poder realizar la implementación del sistema electrónico. Debemos basarnos en

el requerimiento de que la solución debe ser la más barata, eficiente y duradera posible.




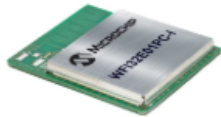
3.5.1. Sensor de humedad y temperatura de suelo

| | Sensor temperatura y humedad para exterior (SHT31) [41] | Bi-Sensor de temperatura y humedad de suelo [42] | Sensor combinado de temperatura y humedad del suelo Truebner SMT100 [43] | Sensor de suelo aramet (Teros-12) [44] |
|--------------------------------------|---|---|---|---|
| Imagen |  |  |  |  |
| Voltaje alimentación(V) | 2.4 - 5.5V | 3.3-5 | 4-24 | 3.3 - 5 |
| Contenido volumétrico de agua (VWC) | Rango: 0 a 100% Resolución: 0.0001 m3/m3 Exactitud: Calibración genérica: ±0.02 m3/m3 en solución <8 dS/m Calibración específica media: ±0.02 m3/m3 | Rango: Calibración de suelos minerales: 0.00-0.70 m3/m3 Calibración hidropónico: 0.0-1.0 m3/m3 Permisividad dieléctrica aparente (εa): 1 (aire) a 80 (agua) Resolución: 0.001 m3/m3 Exactitud: Calibración genérica: ±0.03 m3/m3 en solución <8 dS/m Calibración específica media: ±0.01-0.02 m3/m3 en cualquier medio poroso | Rango: 0-60% Calibración de suelos minerales: 0.00-0.70 m3/m3 Calibración hidropónico: 0.0-1.0 m3/m3 Permisividad dieléctrica aparente (εa): 1 (aire) a 80 (agua) Resolución: 0.001 m3/m3 Exactitud: Calibración genérica: ±0.03 m3/m3, en solución <5 dS/m Calibración específica media: ±0.01 m3/m3 | Rango: 0 - 100 % Permisividad dieléctrica aparente (εa): 1 (aire) a 80 (agua) Resolución: 0.001 m3/m3 Exactitud: ±0.03 m3/m3 Calibración específica media: ±0.01-0.02 m3/m3 en cualquier medio poroso |
| Temperatura | Rango: -40° a-125°C Resolución: 0.015°C Exactitud: ±0.2 °C | Rango: -40 - 60 °C Resolución: 0.1 °C Exactitud: ±1 °C desde -40 - 0 °C ±0.5 °C a 0 - +60 °C | Rango: -40 - 80 °C (versión digital) -40 - 60 °C (versión analógica) Resolución: 0.01°C (versión digital), 0.2°C (versión analógica) Exactitud: ± 0.2°C, max. ± 0.4°C (versión digital), ± 0.8°C(versión analógica) | Rango: -40 - 60 °C Resolución: 0.1 °C Exactitud: ±0,5 °C |
| Comunicación con Uc | I2C | DDI o SDI-12 | RS485 con TBUS RS485 con ASCII SDI-12 RS485 con Modbus Analógica: 0 - 10 V | DDI o SDI-12 |
| Consumo(mA) | <1.5 | 35 | <40 | <10 |
| Conectividad | - | - | - | Aranet Cloud |
| Sensible a la sal | No | No | No | No |
| Tiempo de respuesta | >2s | Inmediato | Inmediato | Inmediato |
| Regulador de Voltaje Interno | Si | Si | Si | Si |
| Precio en soles (sin costo de envío) | 105.21 | 250 | 668.76 | 176.32 |

Cuadro 3.1: Tabla comparativa de sensores de humedad y temperatura de suelo[Elaboración propia]

En este caso, se opta por utilizar el sensor integrado, para humedad y temperatura de suelo. Debido a este usa el protocolo I2C para la comunicación con el microcontrolador. Asimismo, cuenta con la resolución de 0.1°C y precisión de +-2°C, con los cuales cubrimos los requerimientos respectivos descritos anteriormente.




3.5.2. Microcontrolador

| | ESP32S2WROVER(4MB) [45] | ESP32WROOM32E(4MB) [46] | DA16200 [47] | WFI32E01PC [48] |
|------------------------------------|---|--|--|---|
| Imagen |  |  |  |  |
| Alimentación(V) | 3-3.6 | 3-3.6 | 2.1-3.6 | 3.0-3.6 |
| Frecuencia de reloj(Mhz) | 240 | 240 | 160 | 200 |
| Microcontrolador | ESP32-S2 integrado, microprocesador Xtensa® single-core LX7 de 32 bits | ESP32-D0WD-V3 o ESP32-D0WDR2-V3 integrado, microprocesador Xtensa dual-core 32-bit LX6 | Núcleo Arm® Cortex®-M4F | Núcleo de microprocesador MIPS32® Clase M |
| Consumo típico(mA) | 20 | 20 | 7 | 20 |
| Consumo ultra ahorro(uA) | 20 | 5 | 0.2 | 0.71 |
| Memoria Ram(KB) | 320 | 520 | 512 | 320 |
| Memoria Flash(MB) | 4 | 4 | 4 | 1.024 |
| Empresa | Espressif | Espressif | Renesas | Microchip |
| Connectividad | 802.11 b/g/n, 802.11n hasta 150 Mbps | 802.11 b/g/n, 802.11n hasta 150 Mbps y Bluetooth V4.2 | IEEE 802.11b/g/n, IEEE 802.11s, Bluetooth | IEEE® 802.11 b/g/n, |
| Periféricos | GPIO, SPI, LCD, UART, I2C, I2S, interfaz IR, contador de impulsos, LED PWM, USB OTG de alta velocidad, ADC, DAC, sensor táctil, sensor de temperatura | Tarjeta SD, UART, SPI, SDIO, I2C, LED PWM, PWM de motor, I2S, IR, contador de pulsos, GPIO, sensor táctil capacitivo, ADC, DAC, TWAI | Memoria ampliada eMMC/SD, SDIO, ADC UART, SPI, I2C, PWM, JTAG y SWD | UART, SPI, SDIO, I2C, PWM de motor, I2S, IR, contador de pulsos, GPIO, sensor táctil capacitivo, ADC, DAC, TWAI |
| Potencia de salida TX(dBm) | 19,5 | 19 | 19 | 21,5 |
| Consumo normal de envío.nico(mA) | 220 | 276 | 240 | 287 |
| Consumo máximo de envío(mA) | 310 | 379 | 320 | 390 |
| Entorno de desarrollo | Arduino IDE, PlatformIO, SDK de Espressif | Arduino IDE, PlatformIO, SDK de Espressif | Arduino IDE 1.6+ | Arduino IDE, Microchip Studio |
| Precio(dólares) sin costo de envío | 4 | 2,5 | 9,5 | 14 |

Cuadro 3.2: Tabla comparativa de microcontroladores[Elaboración propia]

En la presente tabla, se visualizan distintas alternativas que poseen similares características y corrientes de operación similares. Sin embargo, existen 2 opciones que cumplen el requisito de entrar en un bajo consumo muy pequeño, por ello, el factor determinante para la solución es el precio ya que serán implementadas para 50 nodos de sensores en total.

3.5.3. Regulador de voltaje

| | SGM2212[49] | MP2322 [50] | RP118 [51] | RP170N331B-TR-FE [52] |
|------------------------------------|---|---|--|---|
| Imagen |  |  |  |  |
| Voltaje de entrada(V) | 2,7-20 | 3-22 | 1,7-5,5 | 2,6-10 |
| Salida de corriente(mA) | 800 | 1000 | 100 | 300 |
| Corriente de reposo(uA) | - | 5 | 0,002 | 0,1 |
| Voltaje de salida(V) | 1,25 - 15 | 0,6 - 12 | 3,3 | 3,3 |
| Precisión de salida (+-%) | 1 | 20 | 0.8 | 1 |
| Temperatura de operación(°C) | -40 - 125 | -40 - 125 | -40 - 80 | -40 - 85 |
| Precio(dólares) sin costo de envío | 8,9 | 1,5 | 0,8 | 0,64 |

Cuadro 3.3: Tabla comparativa de microcontroladores[Elaboración propia]

En la presente tabla, se visualizan distintas alternativas que poseen similares características y corrientes de bajo consumo similares. Sin embargo, existen 2 opciones que cumplen el requisito de entregar el amperaje necesario. Esto lo realizamos tomando en cuenta el consumo normal de envío acerca del amperaje del microcontrolador cuando realiza el envío por Wifi añadiendo lo que utiliza el sensor de suelo elegido. Esto nos da un resultado de mínimo 300mA. Por ello, el regulador elegido es el que cumple el requisito de entregar a la salida esa cantidad de amperaje y ser de muy bajo consumo y ,en especial, un costo económico.

3.5.4. Alimentación

Para la elección de la batería elegida se tomo en cuenta lo siguiente: El equipo dure por lo menos 1 mes sin que tenga que cambiarse la batería. Por ello, tenemos que tener en cuenta los consumos de energía respectivos.

Asimismo, se utilizará el modo de bajo consumo del microcontrolador. Se debe tomar en cuenta el requisito de poder enviar los datos cada 30 minutos. Entonces debemos realizar las siguientes consideraciones: El tiempo que el controlador estará en funcionamiento con los sensores, cada cuánto tiempo el ESP32 obtendrá la referencia para comparar si salir del modo ahorrador de energía para la realización del sensado, las veces que se activará en el día y los picos máximos al utilizar el ESP32 WIFI. Para esto nos basaremos en la siguiente fórmula:

$$((T_{uso})(A_{uso}) + (T_{sueno})(A_{sueno}))(N_{veces}) = Bat_{Ah} \quad (3.1)$$

Tuso es el tiempo en segundos cuando está realizando las medidas.

Tsueño es el tiempo en segundos cuando se mantiene en modo dormido o ahorro de energía.

Auso la cantidad de amperaje que usa mientras está en operación. En esta se incluye corriente de reposo del regulador, del microcontrolador en modo sueño profundo y el sensor sin que utilice energía.

Asueño la cantidad de amperaje que usa mientras está dormido.

Nveces la cantidad de veces que se debe repetir en el intervalo de cada hora para llegar a una duración específica.

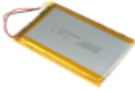

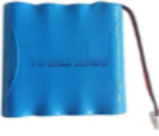
Usando nuestros datos obtenidos, la ecuación queda de la siguiente manera:

$$(5)(300)(10^{(-3)}) + (3595)(5,1)(10^{(-6)}))720/3600 = Bat_{Ah} \quad (3.2)$$

En la cual se ha asignado al número de veces 720 debido que se considera al mes de 30 días y el día tiene 24 horas. Asimismo, se divide en 3600 para cambiar de segundos a horas.

Finalmente, como resultado de esa operación se obtiene 0.3036Ah lo que equivale a 3036mAh.

El cual sería el resultado para cubrir 1 mes. Asimismo, el protocolo wifi al realizar la conexión no suele ser rápida y mientras espera ese lapso de tiempo consume una cantidad considerable de amperaje. Por todo lo anterior, se le considera poner un factor de seguridad de 2 para poder obtener como mínimo un uso de 1 mes. Por ello, se elige la batería más económica y con 6000mAh.

| | BATERÍA PLANA 3.7V 6000MAH 806090 [53] | BATERÍA PLANA 3.7V 1400MAH 454261 [54] | Lifepo4 batería 18650 6000mAh 3,3 V [55] |
|------------------------------------|---|---|--|
| Imagen |  |  |  |
| Voltaje alimentación(V) | 3,7 | 3,7 | 3,3 |
| Capacidad(mAh) | 6000 | 1400 | 6000 |
| Voltaje de carga máxima(V) | 4,2 | 4,2 | 3,3 |
| Temperatura de funcionamiento (°C) | -10 - 45 | -10 - 45 | -10 - 45 |
| Ciclo de vida | 300-500 veces | 300-500 veces | 1000 veces |
| Precio | 70 soles | 30 soles | 20 soles (sin envío) |

Cuadro 3.4: Tabla comparativa de baterías[Elaboración propia]

3.6. Selección de la nube

Para la nube se determinaron las siguientes secciones para poder realizar la implementación de la nube.

3.6.1. Entorno de programación para front-end



Presentamos las siguientes opciones para el entorno de programación para front-end.

| | Next [56] | Vite [57] |
|---------------------------|--|--|
| Imagen |  |  |
| Experiencia de desarrollo | Solución completa para construir aplicaciones React | Entorno de desarrollo eficiente y rápido para cualquier problema |
| Renderizado | Opciones de renderizado 'Estático' y 'Dinámico' | Configurar aplicaciones para diferentes frameworks |
| Rendimiento | Renderizado del lado del servidor, que pre-renderiza las páginas, mejorando así los tiempos de carga iniciales | Proceso de construcción rápido y un servidor de desarrollo |
| Flexibilidad | Diseñado como una solución completa para aplicaciones React | Diseñado para ser ligero pero proporciona un entorno de desarrollo rápido |
| Despliegue | Soporta despliegues estáticos, de servidor y sin servidor. | Soporta alojamiento estático y sin servidor |
| Comunidad | Gran comunidad y un amplio ecosistema de bibliotecas y herramientas | Pequeña |

Cuadro 3.5: Tabla comparativa de servicios para desarrollo de frontend[Elaboración propia]

3.6.2. Entorno de programación para back-end

Presentamos las siguientes opciones para el entorno de programación para back-end.



| | Nodejs [58] | SpringBoot [59] |
|--------------------------|--|---|
| Imagen |  |  |
| Lenguaje de programación | Javascript | Java |
| Modelo | E/S sin bloqueo(un thread trabaje en otra tarea mientras espera que se complete una tarea diferente.) | Consumo del thread principal, y otros se usan simultáneamente. |
| Soporta computo pesado | No | Si |
| Utilización de memoria | Media | Alta |
| Velocidad | Normal, debido a que no se rige que variables se utilizarán | Rápido debido a las variables se debe implementar |
| Comunidad | En crecimiento | Establecida y próspera |
| Empresas | - Medium- Netflix- Uber- LinkedIn- Paypal- Ebay- NASA | - Google- Microsoft- Accenture- Amazon- Intuit- JPMorgan Chase & Co.- Capital One |

Cuadro 3.6: Tabla comparativa de servicios para desarrollo de backend[Elaboración propia]

Nos inclinamos por utilizar el framework de nodejs debido a su practicidad y multiplataforma para la codificación.

3.6.3. Base de datos

Presentamos las siguientes opciones la base de datos

| Característica | Google Cloud [60] | Amazon AWS [61] | MongoDB [62] |
|----------------------------------|---|---|--|
| Imagen |  |  |  |
| Lenguaje de implementación | Múltiples | Múltiples | Múltiples |
| Sistemas operativos del servidor | Servidores virtuales en la nube | Servidores virtuales en la nube | Puede ser instalado en varios sistemas operativos |
| Esquema de datos | SQL y NoSQL | NoSQL (DynamoDB) y SQL (RDS) | NoSQL |
| Soporte XML | Sí | Sí | Sí |
| Índices secundarios | Varies según el servicio | Sí (en bases de datos relacionales) | Sí (en colecciones NoSQL) |
| Base de datos | Relacional y NoSQL | Relacional y NoSQL | NoSQL |
| Método de | API y consola | API y consola | API y consola |
| Escalabilidad | Escalabilidad a pedido y | Escalabilidad a pedido y | Escalabilidad a pedido y |
| Regulador de Voltaje Interno | No aplica | No aplica | No aplica |
| Precio | Créditos por consumo | Créditos por consumo | Gratuito hasta cierto punto |

Cuadro 3.7: Tabla comparativa de servicios para bases de datos[Elaboración propia]

Nos inclinamos por utilizar MongoDB, debido a que la utilización y creación de este es factible para poder realizar el guardado de los datos.

3.7. Diseño electrónico de la tarjeta

A continuación, se mostrarán el circuito esquemático diseñado y el circuito en 3D. Para la realización del circuito, se utilizó la plataforma en línea de diseño de circuitos electrónicos de Easyeda [Easyeda 2023].

3.7.1. Diagrama esquemático

En esta sección, se mostrará el circuito esquemático realizado.

NODO SENSOR IOT

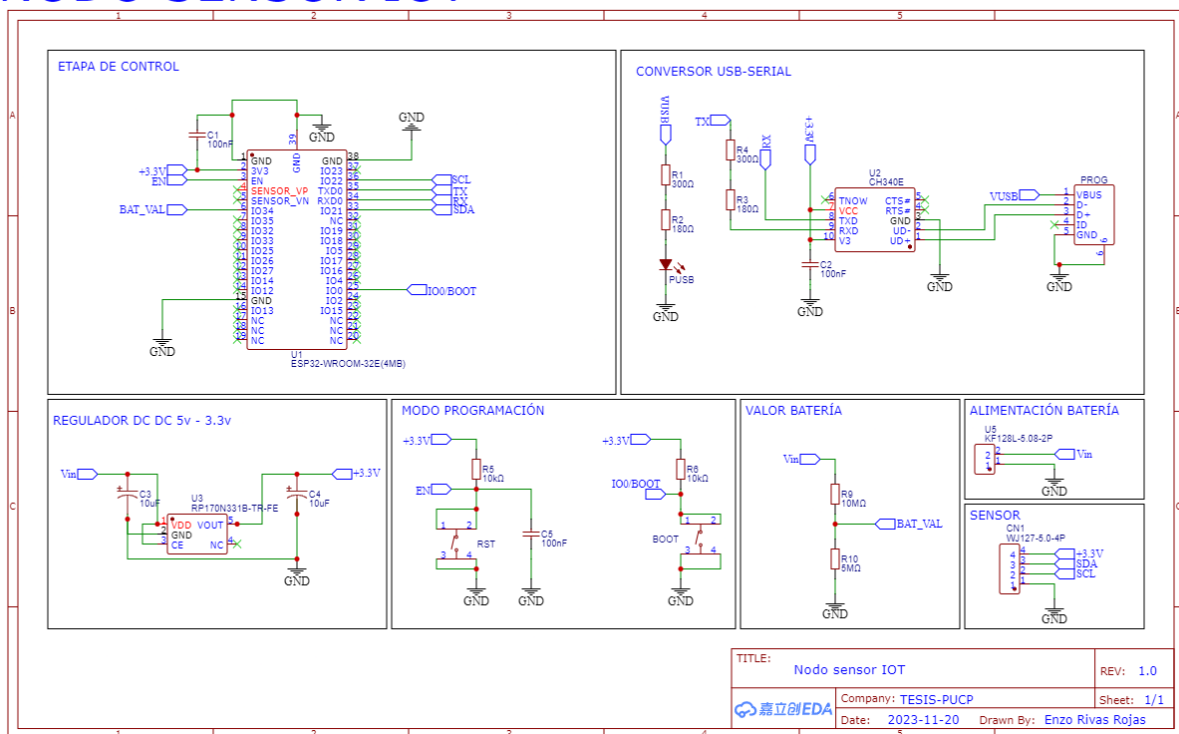


Figura 3.15: Diagrama esquemático[Elaboración propia]

En la etapa de control, se encuentra por el microcontrolador elegido. Asimismo, se puede visualizar que se hace uso del protocolo I2C para la comunicación del ESP32 con el sensor. Por otro lado, el microcontrolador se programa mediante conexión serial a UART. Se utiliza en este el CH340E para realizar esta acción.

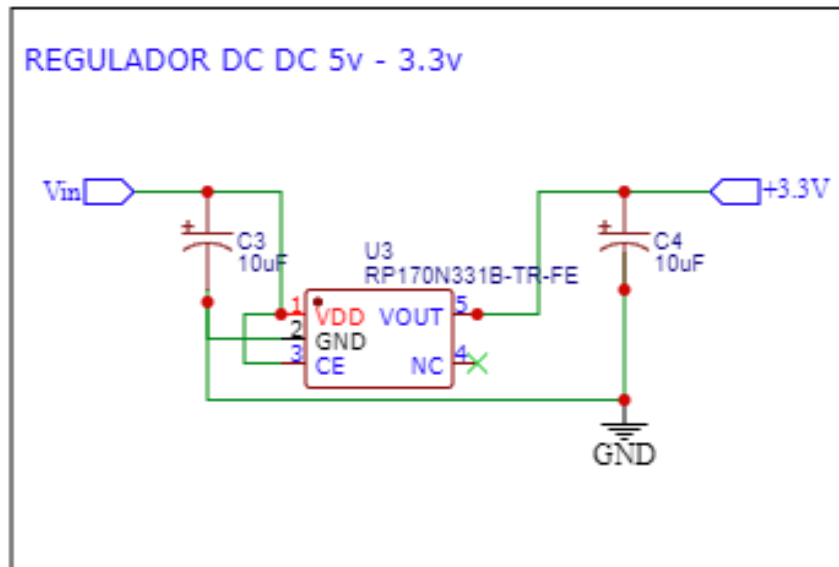


Figura 3.18: Regulador 5V a 3.3V de esquemático[Elaboración propia]

En la parte de modo programación, se encuentran dos botones colocados en modo pull up para el uso adecuado. En este caso, el botón de RST para resetear el circuito y el de BOOT para inicializar la programación.

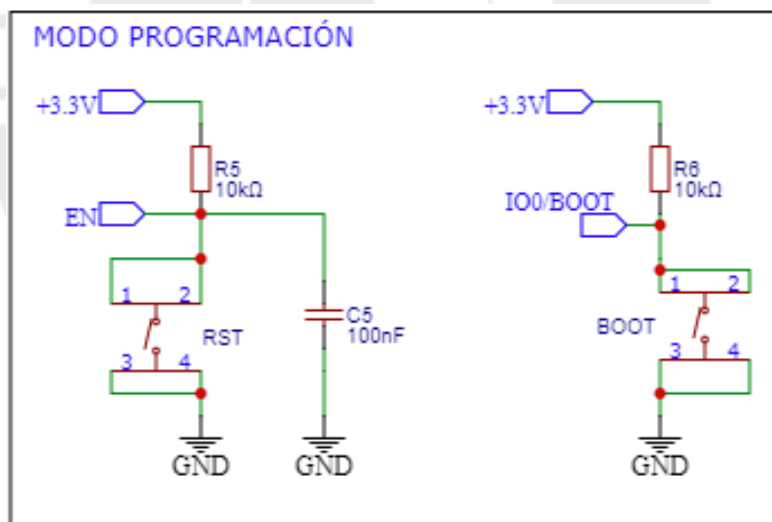


Figura 3.19: Botones de programación de esquemático[Elaboración propia]

En la parte de valor batería, se encuentran dos resistencias en megaohmios para realizar un arreglo resistivo para poder disminuir el voltaje de la entrada de la batería, que variará según su estado de carga gracias a esto podemos obtener el valor de estado de la batería.

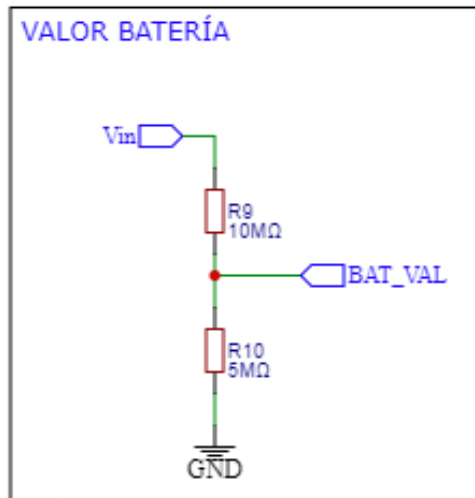


Figura 3.20: Botones de programación de esquemático[Elaboración propia]

En la parte de alimentación de batería, se encuentran únicamente una bornera para realizar la conexión necesaria entre la batería externa y el circuito de alimentación.

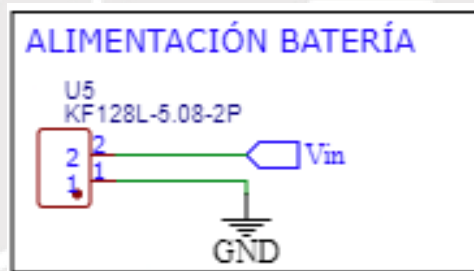


Figura 3.21: Alimentación de esquemático[Elaboración propia]

En la parte de sensor, se encuentran únicamente una bornera para realizar la conexión necesaria entre el sensor y la tarjeta diseñada.

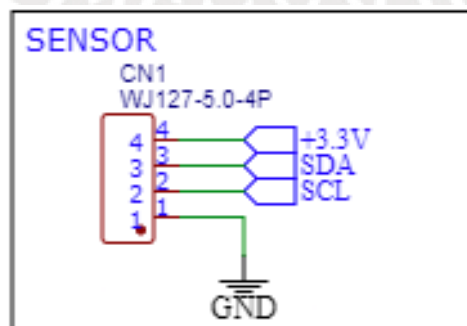


Figura 3.22: Conexión de sensor para esquemático[Elaboración propia]

3.7.2. Circuito impreso

A continuación, se mostrará el circuito impreso diseñado con las siguientes reglas: espaciado de 0,254mm entre pistas y grosor de pista de 0,3mm.

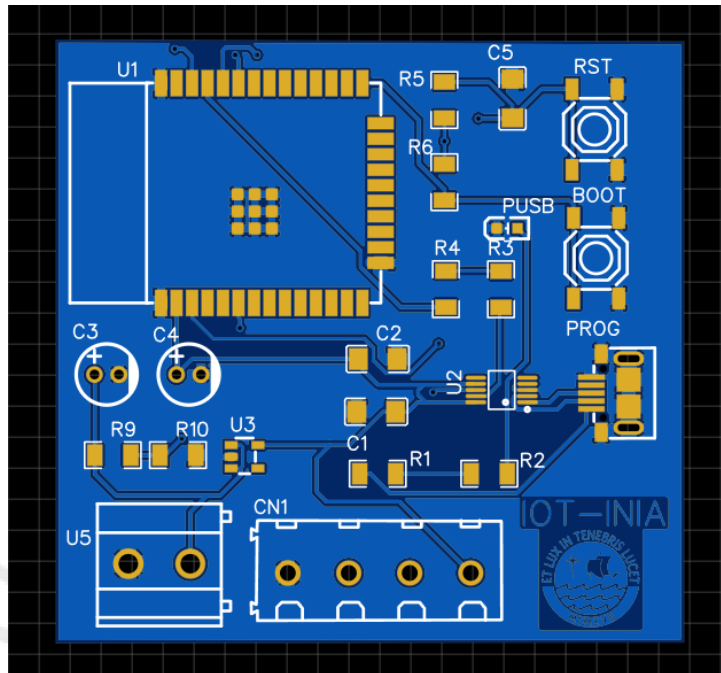


Figura 3.23: Circuito impreso parte superior[Elaboración propia]

En esta parte superior, se puede apreciar la realización de la acomodación de los componentes de tal manera de no generarle ruido al microcontrolador. Asimismo, las borneras, micro-usb y botones colocados en los bordes de la pista.

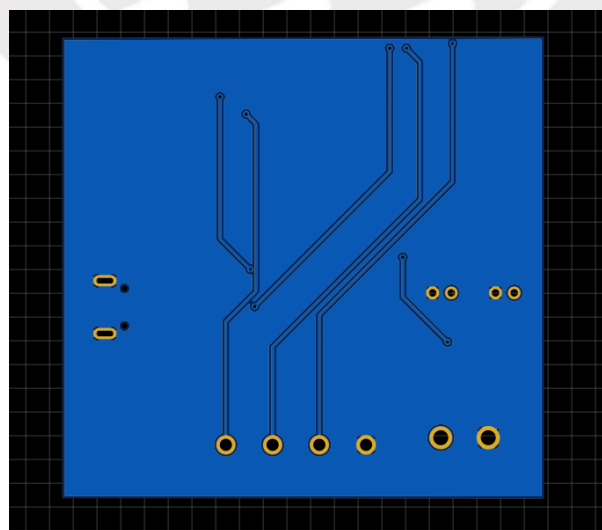


Figura 3.24: Circuito impreso parte inferior[Elaboración propia]

En esta parte inferior, solo se ha realizado conexiones de apoyo para conectar satisfactoriamente los pines utilizados para la comunicación del sensor con el microcontrolador.

Finalmente, se muestra el modelo 3D final para el nodo sensor a implementar.

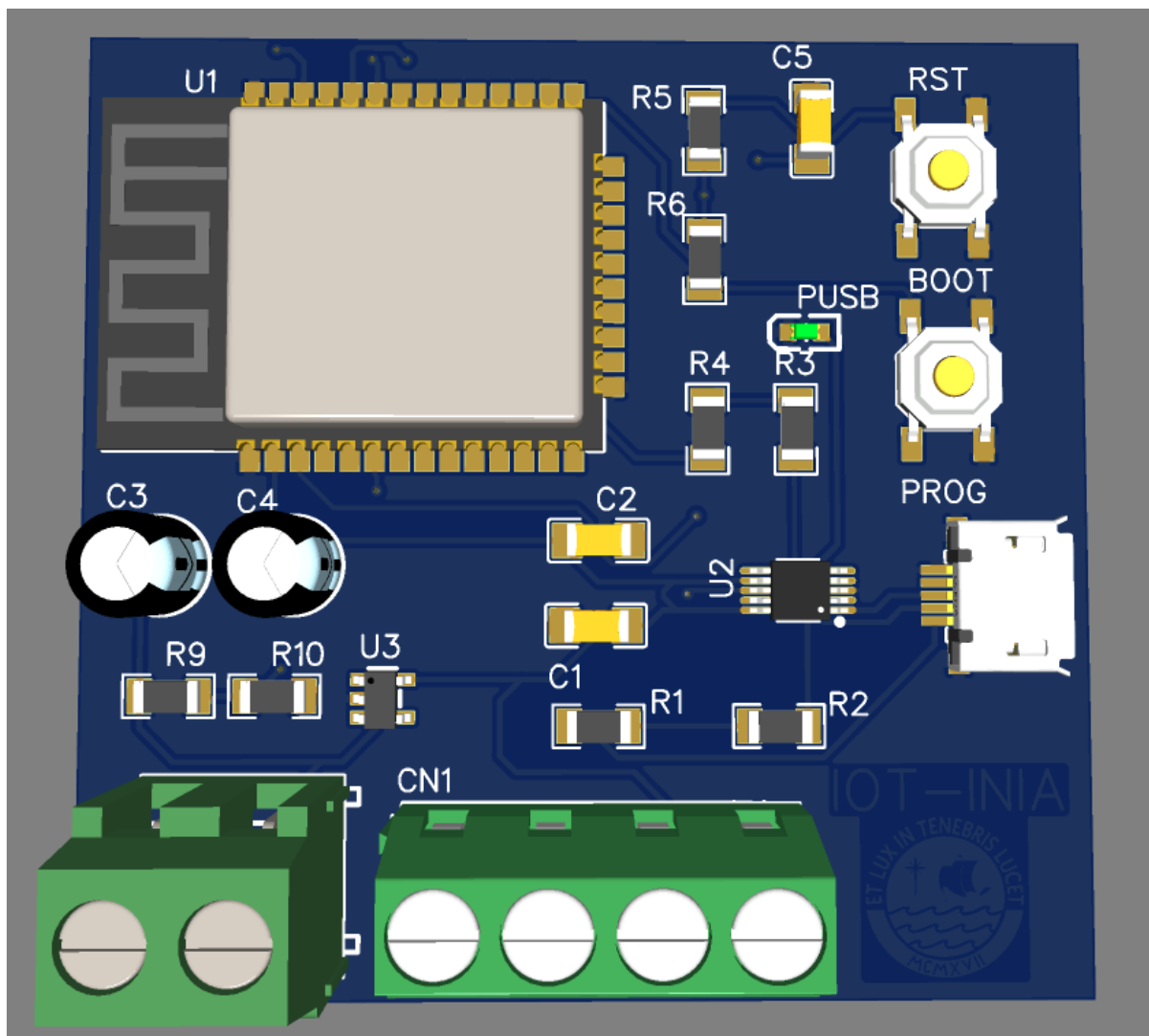


Figura 3.25: Circuito impreso 3D[Elaboración propia]

3.8. Diagramas de flujos

A continuación, se mostrarán diagramas de flujo que se realizaron para la solución propuesta. Se mostrarán los diagramas para la parte electrónica, la página web FrontEnd y la página web Backend.

3.8.1. Diagrama de flujo para la electrónica

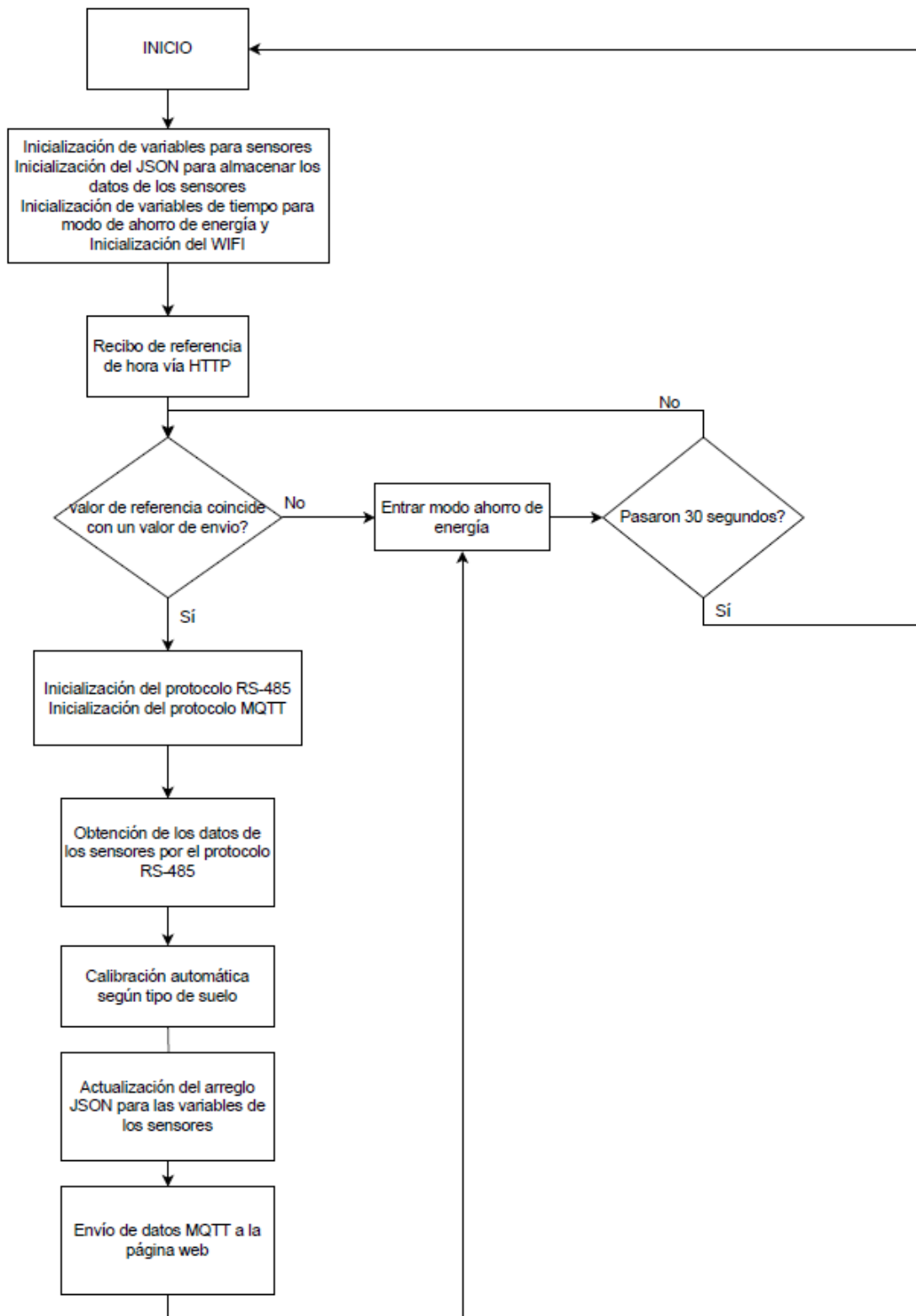


Figura 3.26: Diagrama de flujo para la electrónica[Elaboración propia]

3.8.2. Diagrama de flujo para la página web FrontEnd

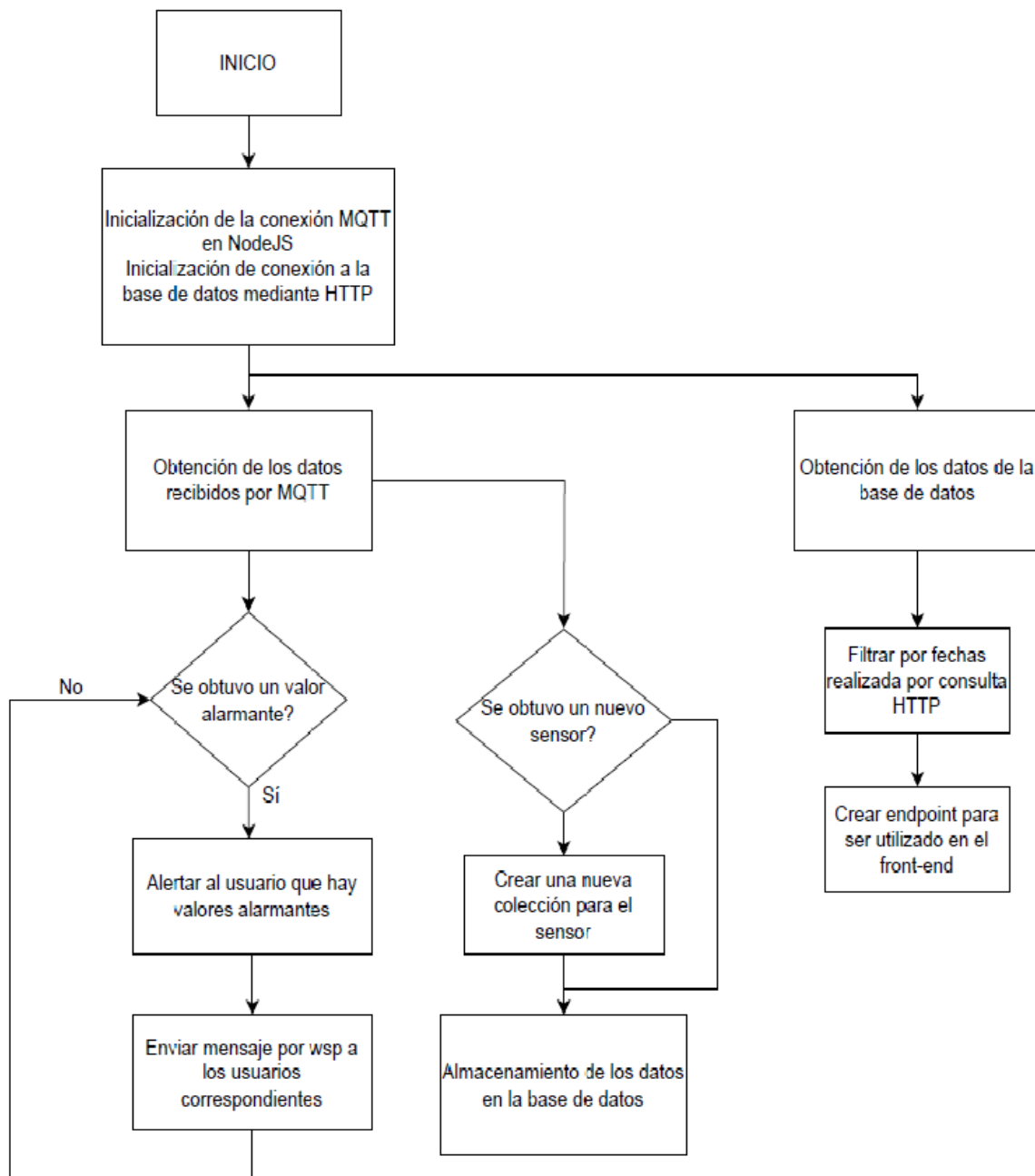


Figura 3.27: Diagrama de flujo para la página web FrontEnd[Elaboración propia]

3.8.3. Diagrama de flujo para la página web BackEnd

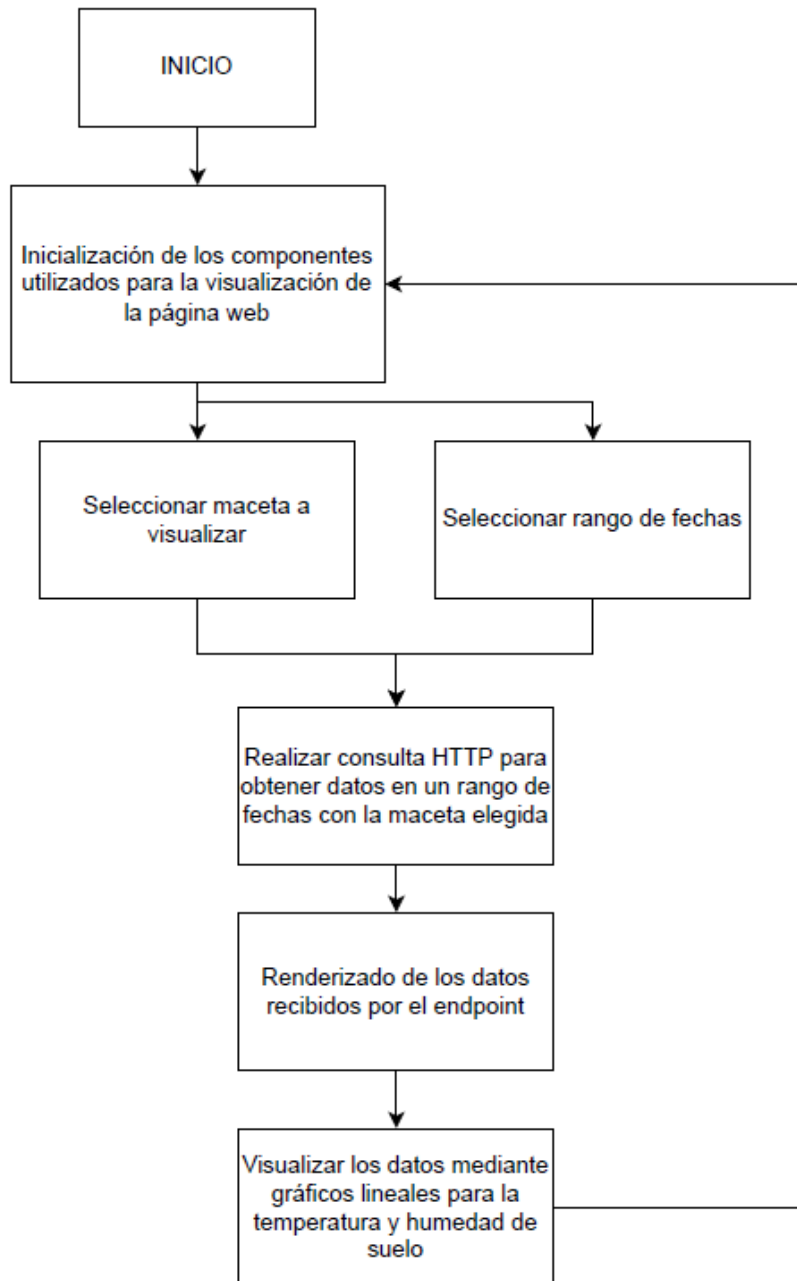


Figura 3.28: Diagrama de flujo para la página web BackEnd[Elaboración propia]

3.9. Flujo de datos de la información

A continuación se mostrará el flujo de datos de toda la información.

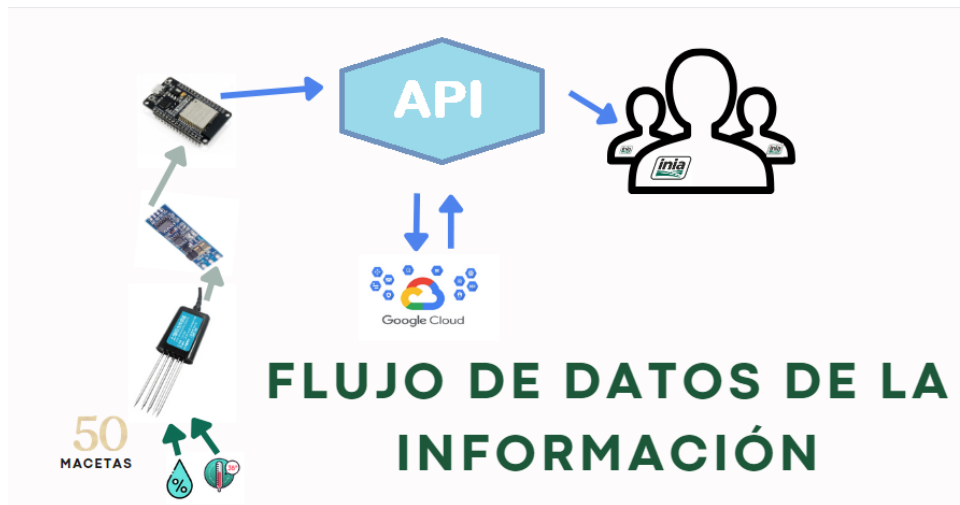


Figura 3.29: Flujo de datos de la información[Elaboración propia]

3.10. Consideraciones para la implementación

Se darán los detalles más importantes para tener en cuenta en la implementación.

3.10.1. Modelado de carcasa

A continuación, se muestra cómo está la estructura dentro de la caja que contendría el Esp32 y la batería. Existen dos orificios que serán las salidas de los sensores, para este caso como utilizamos un sensor integrado lo utilizaremos para la salida de los cables de alimentación e información del sensor. Asimismo, se incluirá una presiostapa para evitar la penetración del polvo y agua al equipo.

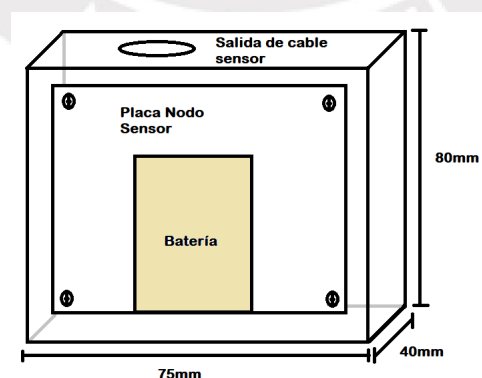


Figura 3.30: Componentes dentro de la caja[Elaboración propia]

3.10.2. Bosquejo para la implementación

A continuación, se muestra un bosquejo de la implementación cómo se colocarían los equipos en el invernadero del INIA.

En primer lugar, se visualiza la ubicación del recinto a través de Google Maps [Google Maps App 2023]

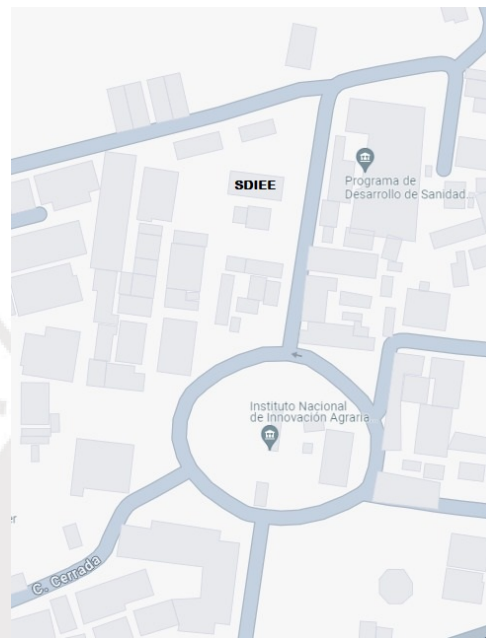


Figura 3.31: Ubicación del INIA[Google Maps App 2023]

A continuación se detalla el área de SDIEE en la cual se está desarrollando la colaboración para el desarrollo de esta solución.



Figura 3.32: Ubicación del invernadero[Google Maps App 2023]

Asimismo, el área nos brinda acceso al repetidor-02 que tiene cobertura hasta el invernadero mostrado en la imagen.

Por otra parte, se mostrarán imágenes del invernadero.



Figura 3.33: Invernadero parte lateral[Elaboración propia]



Figura 3.34: Invernadero parte frontal[Elaboración propia]



Figura 3.35: Interior del invernadero[Elaboración propia]

A continuación, se realizó un bosquejo para la ubicación de las macetas con sus respectivos nodos de sensores.

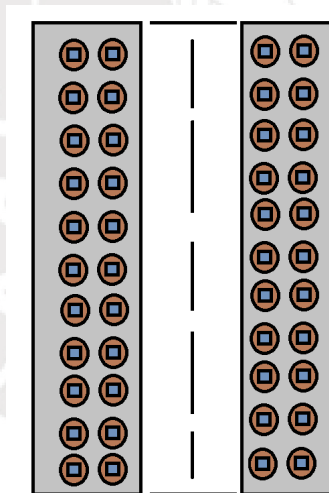


Figura 3.36: Situación de las cajas en el invernadero[Elaboración propia]

Finalmente, se muestra cómo irá colocado el dispositivo, que contiene el nodo de sensor dentro de las macetas. A continuación, se muestra detalladamente.

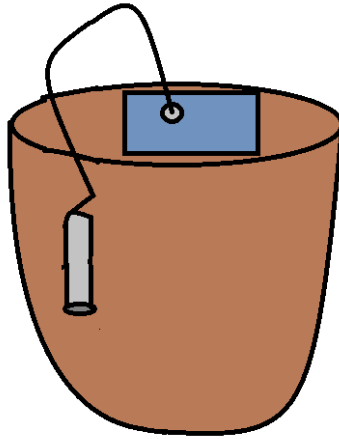


Figura 3.37: Solución implementada en una maceta[Elaboración propia]

3.10.3. Presupuesto total

A continuación, se muestra el presupuesto que debe ser utilizado para la realización de la solución completa con los sensores adecuados.

| Presupuesto | | | | | | |
|-------------|--|--------------------|----------|--------------|----------------|-------------|
| ID | Nombre | Costo unitario(S/) | Unidades | Precio total | Costo de envío | Costo total |
| 1 | Sensor temperatura y humedad para exterior (SHT31) | S/ 105,21 | 50 | S/ 5.260,50 | S/ 50,00 | S/ 5.310,50 |
| 2 | ESP32WROOM32E | S/ 9,50 | 50 | S/ 475,00 | S/ 100,00 | S/ 575,00 |
| 3 | Componentes SMD variadas | S/ 3,00 | 50 | S/ 150,00 | S/ 100,00 | S/ 250,00 |
| 4 | Regulador de voltaje | S/ 2,43 | 50 | S/ 121,60 | S/ 100,00 | S/ 221,60 |
| 5 | Lifepo4 2000mAh 3,7 V | S/ 4,00 | 100 | S/ 400,00 | S/ 100,00 | S/ 500,00 |
| 6 | Circuito electrónico impreso | S/ 15,00 | 50 | S/ 750,00 | S/ 200,00 | S/ 950,00 |
| 7 | Carcasa de dispositivo | S/ 5,00 | 50 | S/ 250,00 | S/ 0,00 | S/ 250,00 |
| 8 | Costo de ingeniero | S/ 6.000,00 | 1 | S/ 6.000,00 | S/ 0,00 | S/ 6.000,00 |
| 9 | Costo de instalación | S/ 2.000,00 | 1 | S/ 2.000,00 | S/ 0,00 | S/ 2.000,00 |

| |
|---------------------|
| Costo total |
| S/ 16.057,10 |

| |
|---------------------|
| Cambio dólar |
| S/ 3,80 |

Cuadro 3.8: Presupuesto total para 50 macetas[Elaboración propia]

Para la realización de este presupuesto. Se tomó en cuenta, la solución elegida y sus componentes respectivos. Asimismo, se realizaron las conversiones respectivas para la moneda nacional. Para esto, se considero un cambio de dólar a nuevos soles de 3.80. Asimismo, los costos del ingeniero y la instalación respectiva para la correcta implementación y funcionamiento de la solución.

Capítulo 4

Pruebas y resultados

En este capítulo, se describirá las pruebas realizadas y las validaciones respectivas para el nodo de sensor, el servidor y la página web. Cabe recalcar que tanto la página web como el servidor se encuentran desplegados en Internet. Asimismo, se desarrollará una prueba de concepto para validar la solución implementada. Para esta prueba de concepto se usarán equipos de menor costo: Esp32kit [46], sensor de humedad de suelo YL67, sensor de temperatura de suelo DS18B20 y una batería de 1700mAh.

4.1. Validación del nodo sensor

En esta sección, se planteará realizar las respectivas validaciones para el nodo de sensor diseñado. Se tomará en cuenta los sensores, la batería y la comunicación que realiza este.

4.1.1. Validación de los sensores

Para esta validación, se realizó la implementación respectiva para poder obtener humedad de suelo y temperatura de suelo. Para una próxima entrega, se realizará la validación de los valores obtenidos de los sensores con el sensor LAquatwin para comparar los datos de humedad y temperatura del suelo obtenidos. Para esta ocasión, se realizaron las calibraciones en el invernadero brindado por el INIA. Como se muestra en la imagen a continuación.



Figura 4.1: Calibración del sensor de humedad y temperatura en el invernadero[Elaboración propia]

Para el caso del sensor de humedad de suelo, se colocó en una maceta del invernadero y se le colocó un valor inicial que depende específicamente de las condiciones del suelo para obtener el porcentaje máximo: 60 %, que era cuando se encontraba totalmente mojado el sensor. Por el contrario, cuando estaba totalmente seco se utilizaba el valor que se obtenía como porcentaje mínimo: 0 %. Para esta prueba, se utilizó un suelo normal sin ningún tipo de tratamiento previo.

Para el caso del sensor de temperatura de suelo, se colocó en una maceta del invernadero y se le colocó un offset para validarlo correctamente con el entorno. Sin embargo, se optó por no colocarle uno debido a su precisión del sensor utilizado.



Figura 4.2: Calibración del sensor de humedad y temperatura [Elaboración propia]

4.1.2. Prueba de duración la batería

En esta prueba, se verificó la duración de la batería. En este caso, se optó por implementar un algoritmo para que entre a un modo sueño profundo[DEEPSLEEP 2022] con el cual se puede obtener un consumo de energía mínimo del microcontrolador.

Se optó por realizar la prueba de duración de batería. Colocando el nodo sensor con la batería y que envíe los datos cada hora.

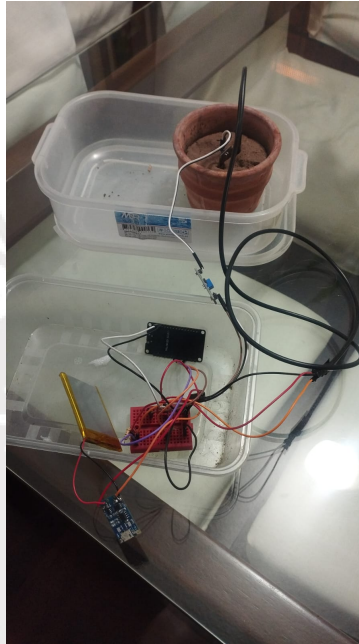


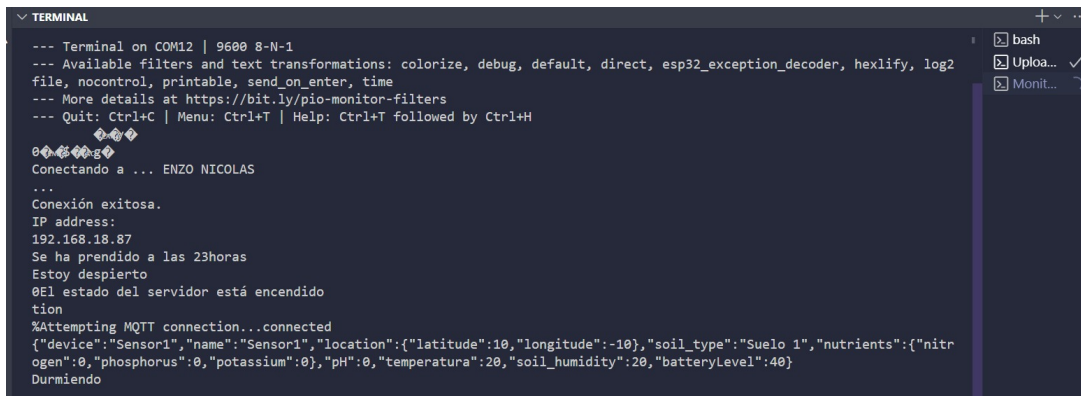
Figura 4.3: Nodo sensor en casa.[Elaboración propia]

La duración de la batería fue de aproximadamente de 3 horas. Se realizó múltiples veces las pruebas y los resultados fueron similares. Esto se debe a que el dispositivo consume más cuando envía debido a que se utiliza WIFI. Asimismo, al utilizar una tarjeta no optimizada para la utilización de este, influye bastante en la duración. Asimismo, los sensores están consumiendo electricidad todo el tiempo por lo que la batería no duraría lo esperado.

4.1.3. Validación de comunicación con el servidor

En esta validación, se verificó la comunicación mediante MQTT, que posteriormente el servidor se encargaría de procesar. En este caso, se optó por utilizar para pruebas el broker gratuito EMQX. Asimismo, cuenta con un cliente en línea para realizar las pruebas correspondientes. A continuación se muestra la prueba de envío y recepción en el EMQX.

Para el caso de toda la implementación, se utilizará el tópico Tesis/Sensores/Envios. Asimismo, se mostrará el envío de los datos desde el ESP32 mediante el protocolo MQTT dentro del tópico anterior.



```
--- Terminal on COM12 | 9600 8-N-1
--- Available filters and text transformations: colorize, debug, default, direct, esp32_exception_decoder, hexlify, log2
file, nocontrol, printable, send_on_enter, time
--- More details at https://bit.ly/pio-monitor-filters
--- Quit: Ctrl+C | Menu: Ctrl+T | Help: Ctrl+T followed by Ctrl+H

00000000
Conectando a ... ENZO NICOLAS
...
Conexión exitosa.
IP address:
192.168.18.87
Se ha prendido a las 23horas
Estoy despierto
0El estado del servidor está encendido
tion
%Attempting MQTT connection...connected
{"device": "Sensor1", "name": "Sensor1", "location": {"latitude": 10, "longitude": -10}, "soil_type": "Suelo 1", "nutrients": {"nitrogen": 0, "phosphorus": 0, "potassium": 0}, "pH": 0, "temperatura": 20, "soil_humidity": 20, "batteryLevel": 40}
Durmiendo
```

Figura 4.4: Consola del ESP32.[Elaboración propia]

Se puede apreciar que se envía adecuadamente los datos de los sensores cuando este se despierta para realizar el sensado y posteriormente el envío de los datos. A continuación se mostrará el dato enviado recepcionado en el cliente.



```
Topic: Tesis/Sensores/Envios QoS: 0
{"device": "Sensor1", "name": "Sensor1", "location": {"latitude": 10, "longitude": -10}, "soil_type": "Suelo 1", "nutrients": {"nitrogen": 0, "phosphorus": 0, "potassium": 0}, "pH": 0, "temperatura": 20, "soil_humidity": 20, "batteryLevel": 40}
2023-11-19 23:41:58.051
```

Figura 4.5: Recepción de datos en el cliente en línea.[66]

Se logrará apreciar como los datos enviados desde el ESP32 se transmiten correctamente por WIFI mediante MQTT hacia el broker gratuito. La respuesta de este es inmediata gracias al servicio del broker a pesar de que el servidor está alojado fuera de Sudamérica[66].

4.2. Validación del backend

En esta sección, se planteará realizar las respectivas validaciones sobre la API rest diseñada para esta solución. En este caso, se realizó una codificación para todo el entorno de la programación mediante Nodejs y sus librerías y Javascript. Asimismo se utilizaron herramientas como Render[61] para el despliegue del servidor en la nube, Postman[62] para la verificación de los enlaces al servidor y el cliente en línea de EMQX [66] para la verificación de la comunicación MQTT en el servidor.

4.2.1. Validación de las solicitudes

Para realizar la validación de las solicitudes mediante enlaces webs se usará Postman[POSTMAN 2023] para poder verificar que el enlace funciona correctamente.

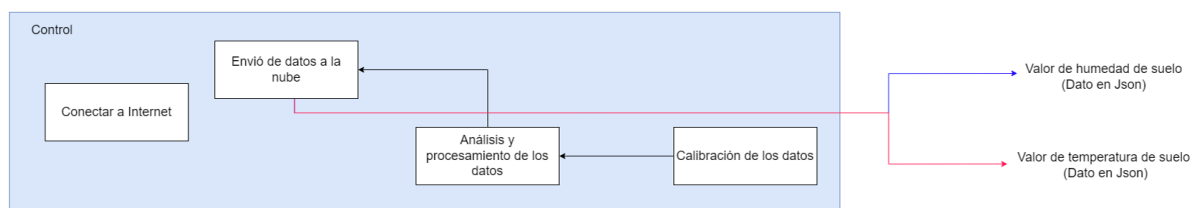


Figura 4.6: Interfaz de Postman [64].

Asimismo, la api rest ha sido colocada en la nube de Render.com exitosamente para que el servidor esté corriendo de manera continua. El enlace de este es: <https://iot-inia.onrender.com>, que será utilizado para verificar los enlaces.

El primer enlace que se validará es el de registro de nuevos usuarios para la página web. Para este caso enviamos el usuario y la contraseña deseados mediante el cuerpo de la solicitud.

De caso de lo contrario, se obtendrá la siguiente respuesta en caso el usuario o la contraseña sean incorrectos.

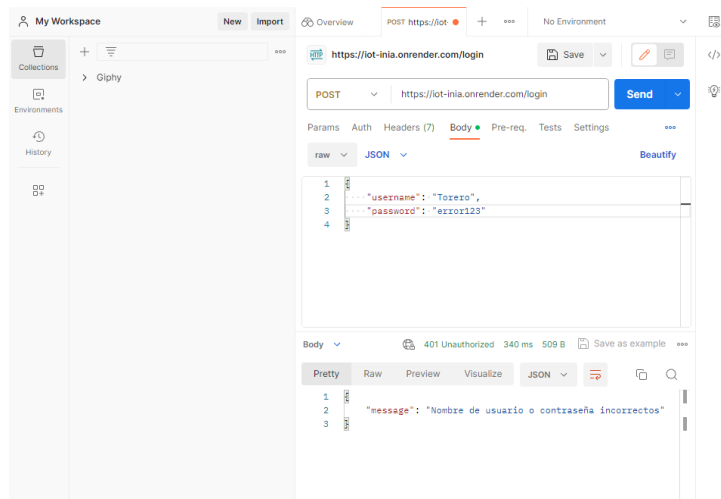


Figura 4.10: Consulta de autenticación fallida para el servidor.[POSTMAN 2023]

El último enlace que se validará será el de obtener todos los datos del nodo de sensor a consultar en la base de datos.

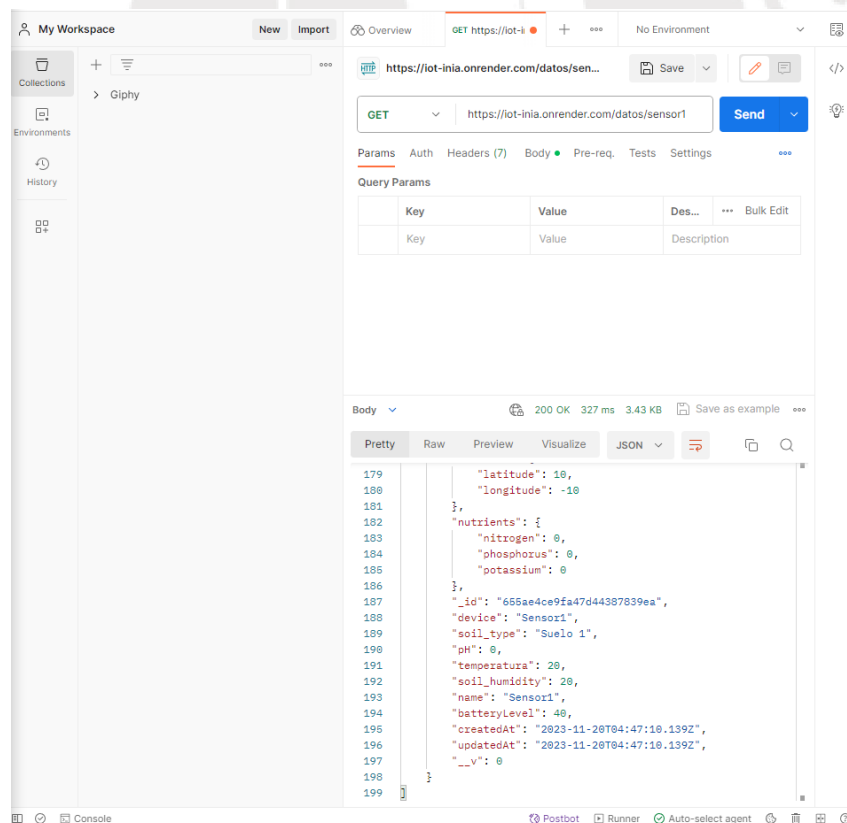


Figura 4.11: Consulta de datos del sensor1 en la base de datos [64].

Según los datos que se envíen en el enlace de la consulta se logra el filtrado de únicamente los datos elegidos. Esto se verá más adelante en el frontend de la página web.

4.2.2. Validación con base de datos

En esta validación se utilizará la consola de Render.com y la consola de MongoDB para visualizar los guardados de los datos correspondientes.

Primero, se realizará la validación de la recepción del dato enviado por el sensor y guardado correspondiente. Para este caso, se utilizará el nodo de sensor 50, ya que no tiene datos registrados hasta el momento. A continuación, se visualiza la consola de Render para el visualizado de los datos.

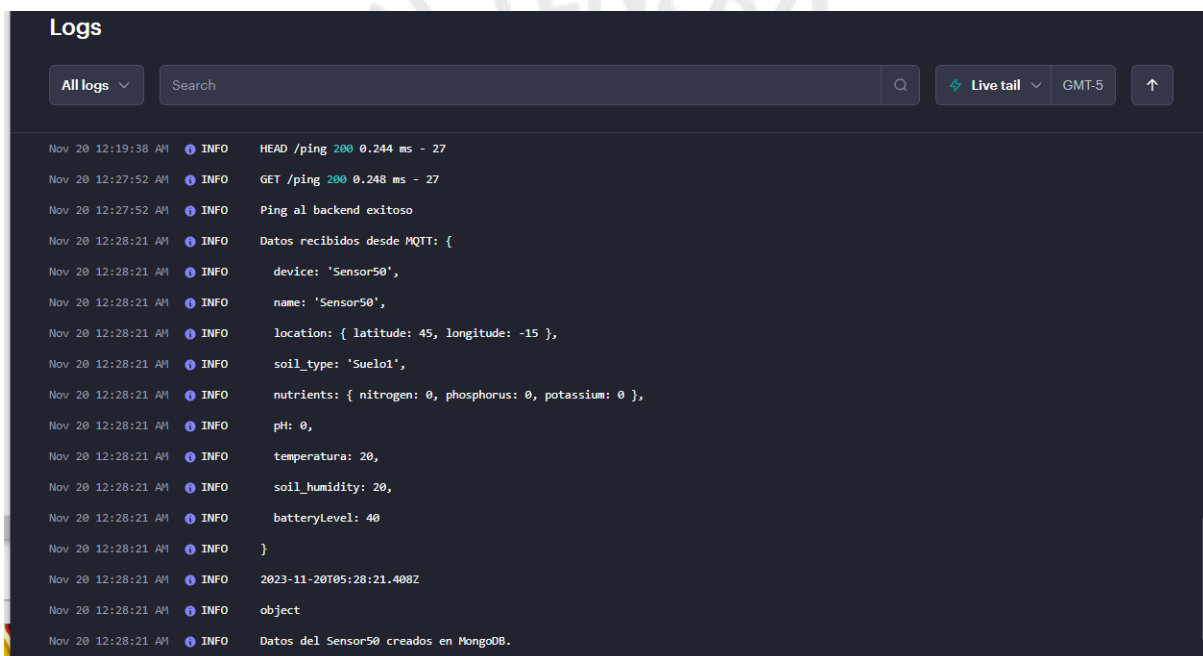


Figura 4.12: Recepción de datos por mqtt en la consola de Render [63].

En la siguiente figura, el dato enviado desde el ESP32 mediante MQTT hacia el servidor es exitosa. Asimismo, el servidor se encarga de guardarla en la base de datos de manera que la información no se pierde. Se visualizará la colección del nodo de sensor 50 en MongoDB.

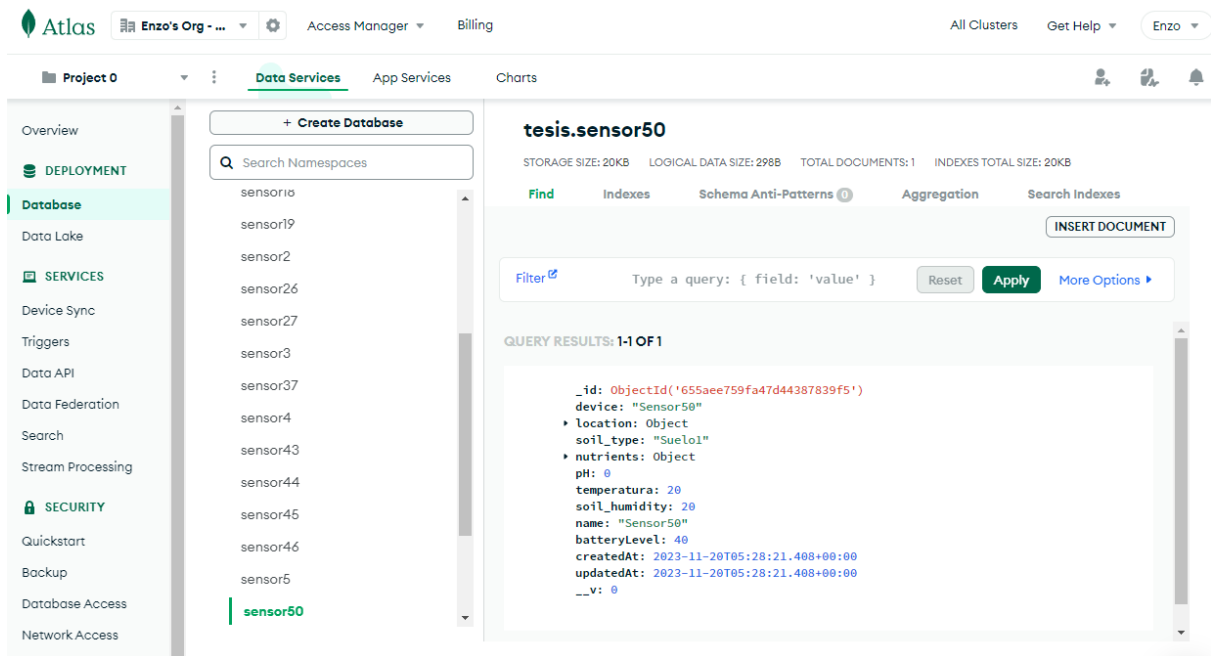


Figura 4.13: Visualizado de la colección del nodo de sensor 50 en MongoDB [62].

La ventaja de trabajar con bases de datos no relaciones radica en que se pueden expandir fácilmente las colecciones para cada sensor. En caso haya una colección para un sensor no creada, esta se generará automáticamente gracias a la codificación realizada.

Ahora se procede a la validación del guardado de los nuevos usuarios registrados en la base de datos. Para este caso, se utiliza la colección de users para guardar los usuarios y contraseñas de todos en la base de datos. A continuación se visualizará la colección.

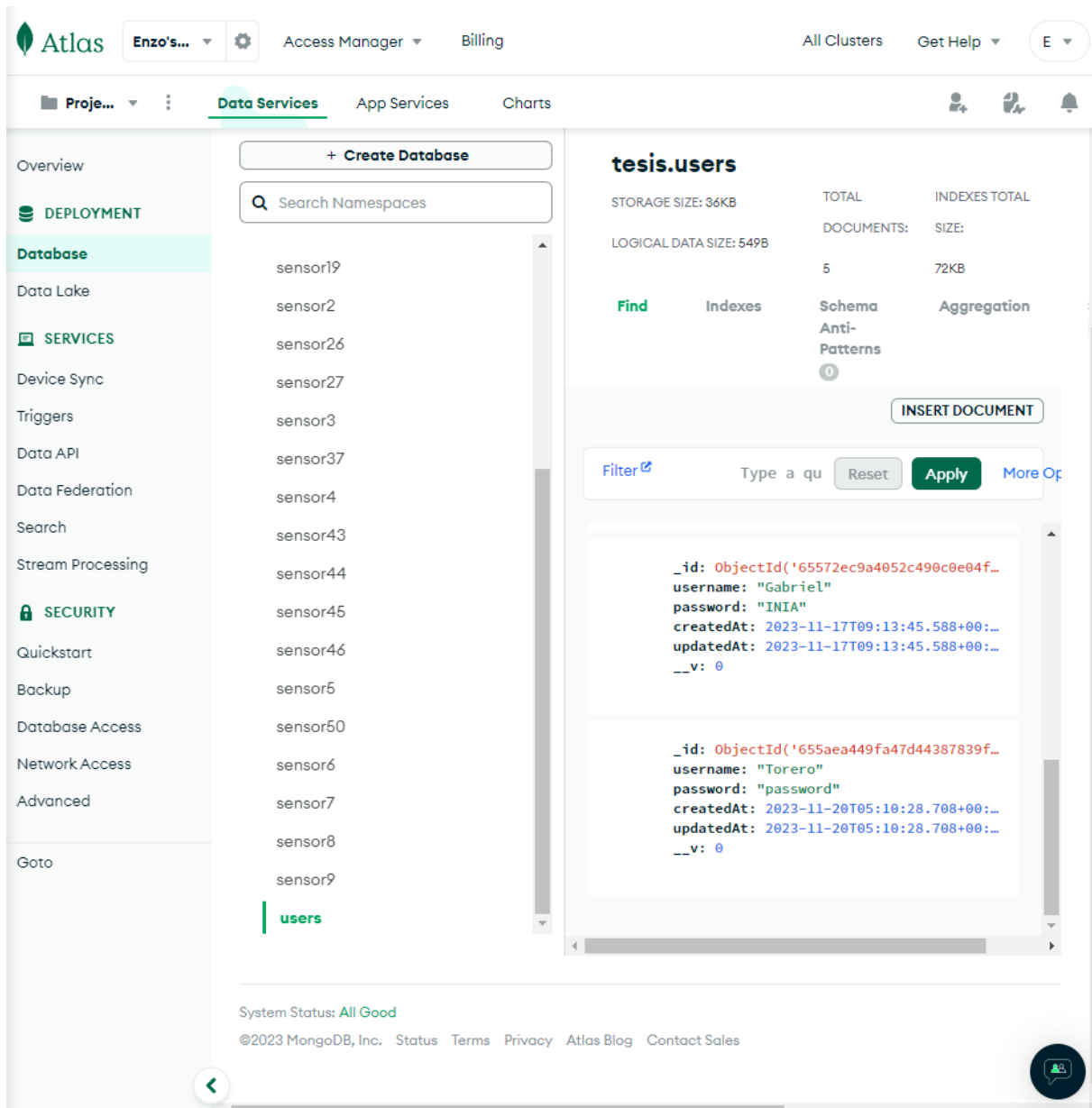


Figura 4.14: Colección de users en MongoDB.[62]

Como se muestra anteriormente, queda validado como se guardan los datos en el servidor.

4.2.3. Validación de envío de alerta a usuario

Para esta validación se utiliza la aplicación de Telegram, así como el API de telegram con su bot creado para la solución desarrollada. [APITELEGRAM 2023] Primero se mostrará la creación del bot y la obtención del chatID, que es fundamental para iniciar la recepción de mensajes desde el bot, que está enlazado al servidor.

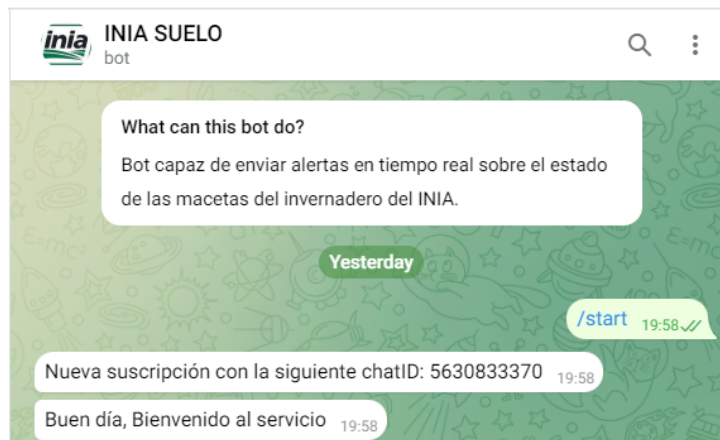


Figura 4.15: Iniciación del bot de Telegram.[Elaboración propia]

A continuación se muestra cómo un usuario nuevo puede suscribirse al bot creado. Para este caso, se creó un grupo de telegram en el cual está agregado el bot creado para que envíe las alertas correspondientes.

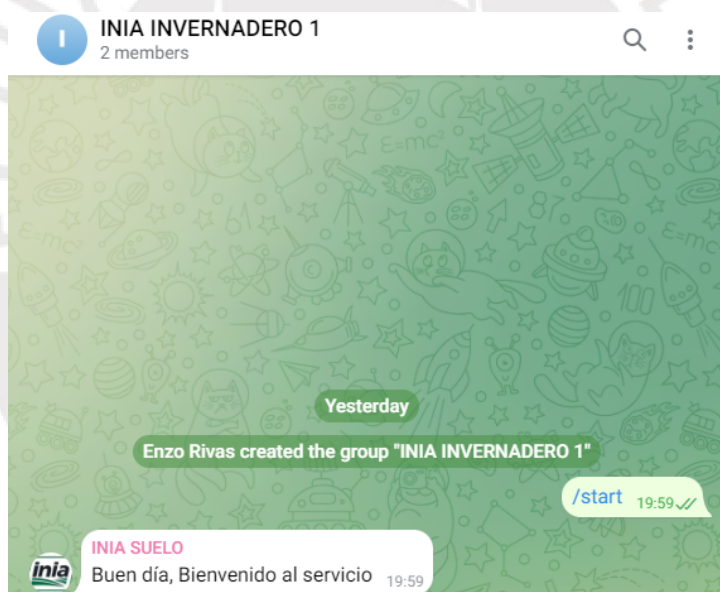


Figura 4.16: Suscripción del bot a un grupo de Telegram.[Elaboración propia]

Asimismo, un usuario nuevo al suscribirse le llegará un mensaje a los administradores correspondientes del nuevo id del chat para poder suscribirlo al servidor para los envíos correspondientes.

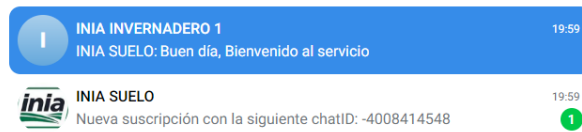


Figura 4.17: Mensaje de nuevo chat al administrador.[Elaboración propia]

Ahora se procederá a mostrar la recepción de la alerta y cuando el servidor se encuentra en línea mandará un mensaje de Servidor en línea. Asimismo, se simulará que llegan valores alarmantes, para este caso será de humedad de 6 %, temperatura de 10°C y batería de 3 % del nodo de sensor 50.



agregar a los usuarios correspondientes que manejarán el invernadero correspondiente.

4.3. Validación del frontend

En esta sección, se planteará realizar las respectivas validaciones sobre la página diseñada para esta solución. En este caso, se realizó una codificación para todo el entorno de la programación mediante Nodejs y sus librerías, Vite y React. Asimismo, cabe recalcar que la página web ha sido desplegada en Internet mediante Netlify [63]. El enlace de la página web es el siguiente: <https://luminous-kashata-7a8734.netlify.app/>

4.3.1. Validación de logeo de usuario

Para esta validación, se posee la página inicial de la página web, que es la página de logeo. A continuación se mostrará.

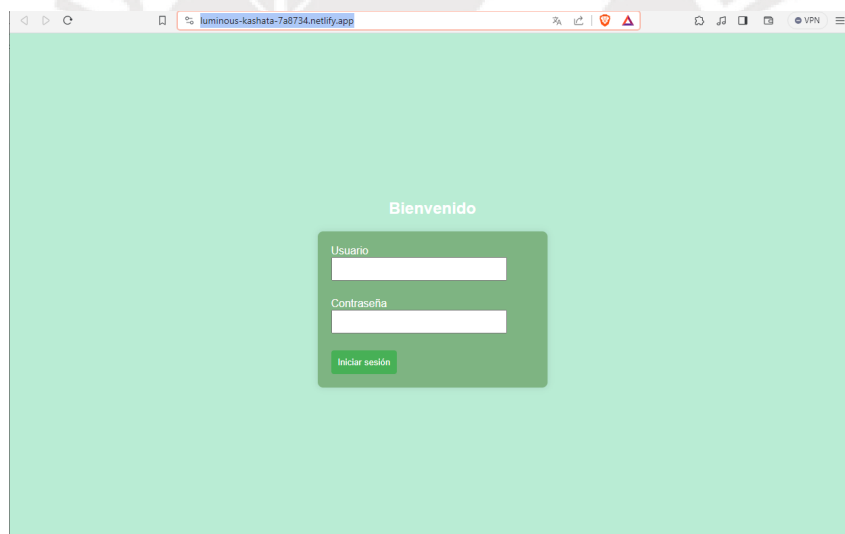


Figura 4.19: Página de logeo.[Elaboración propia]

Ahora se procede a ingresar las credenciales correspondientes de un usuario previamente registrado.



Figura 4.20: Ingreso de credenciales.[Elaboración propia]

En caso las credenciales ingresadas sean las correctas se mostrará la siguiente alerta mediante el navegador y permitirá el acceso a la página web.

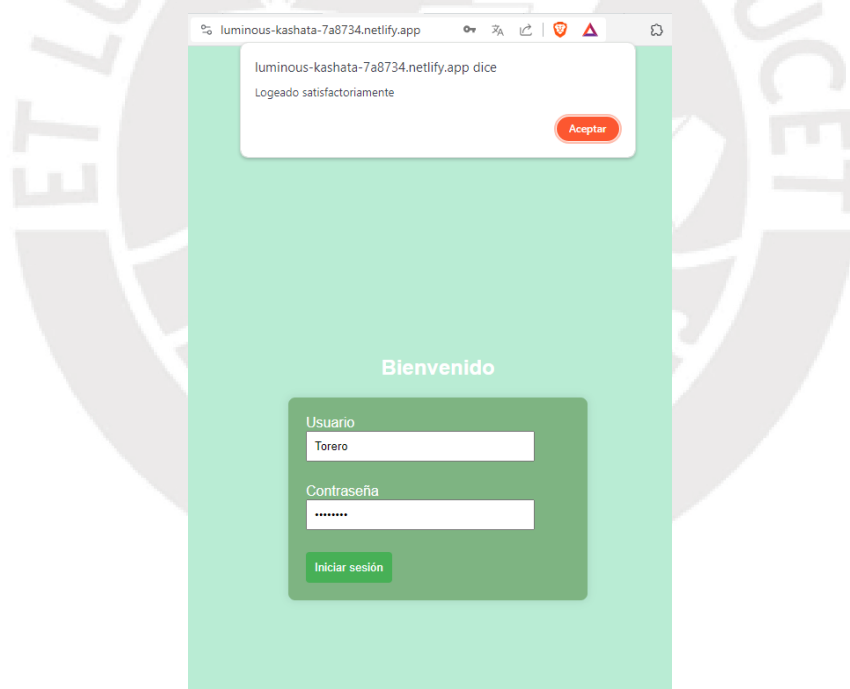


Figura 4.21: Ingreso correcto de las credenciales.[Elaboración propia]

En caso contrario, si las credenciales ingresadas no sean las correctas se mostrará la siguiente alerta mediante el navegador.

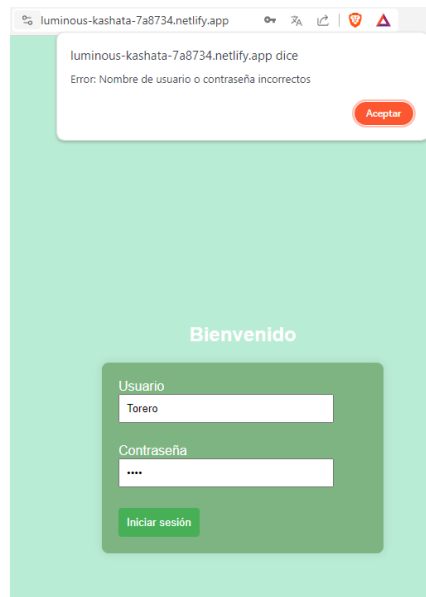


Figura 4.22: Ingreso incorrecto de las credenciales.[Elaboración propia]

Finalmente, se muestra la interfaz de usuario desarrollada.

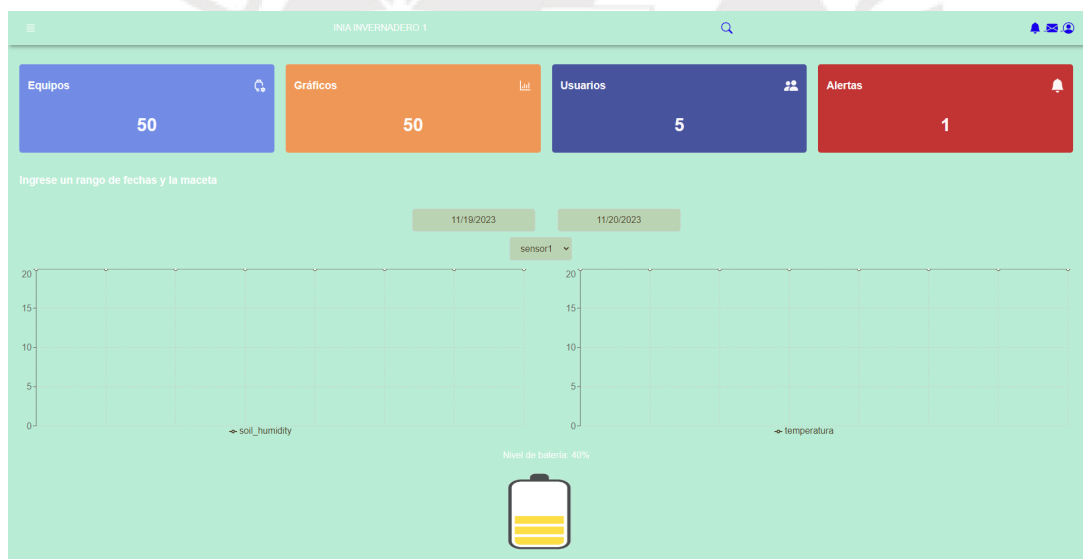


Figura 4.23: Interfaz de usuario.[Elaboración propia]

En esta se puede visualizar, la cantidad de equipos desplegados, la cantidad de equipos con sus gráficos respectivos. Asimismo, la cantidad de usuarios registrados en la base de datos y las alertas emitidas. Asimismo, se cuenta con una barra de navegación.



Figura 4.24: Barra de navegación.[Elaboración propia]

4.3.2. Validación de ingreso de datos

Para esta validación, nos centraremos en el rango de fechas y la maceta seleccionada. Usaremos los valores del nodo de sensor 50 para poder realizar las respectivas validaciones. Primero empezaremos con el seleccionamiento de rango de fechas. Al presionar el selector de rango de fechas se muestra un calendario. A continuación, se mostrará cuando se selecciona el rango de inicio.

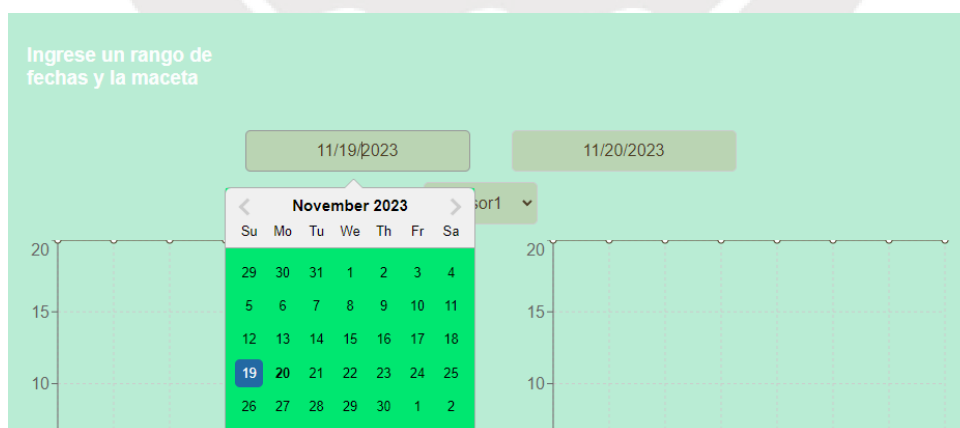


Figura 4.25: Selector de fechas-Inicio.[Elaboración propia]

Ahora se visualiza cuando se selecciona el rango de fin.

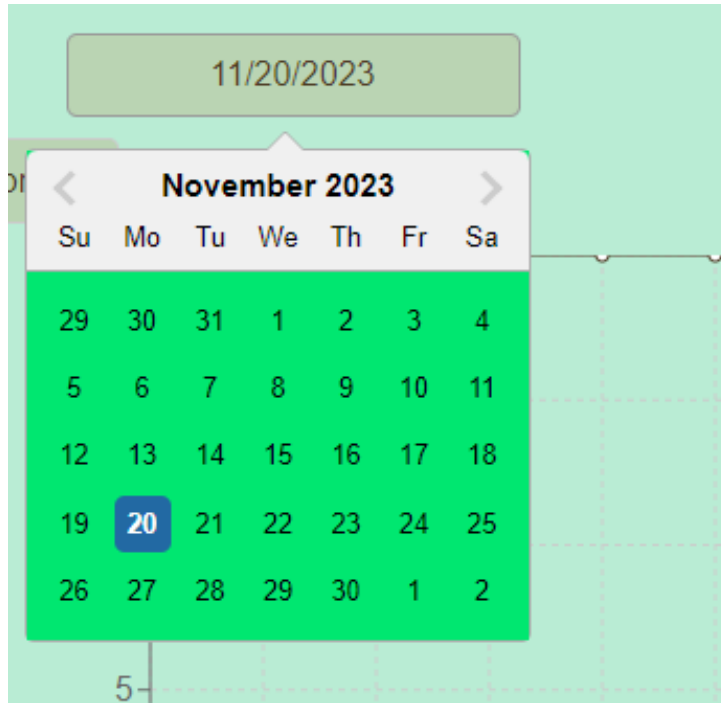


Figura 4.26: Selector de fechas-Fin[Elaboración propia]

Por otra parte, para el seleccionamiento de la maceta a visualizar se diseñó una barra selectora desplazable para poder elegir entre las 50 macetas disponibles como se ve a continuación.

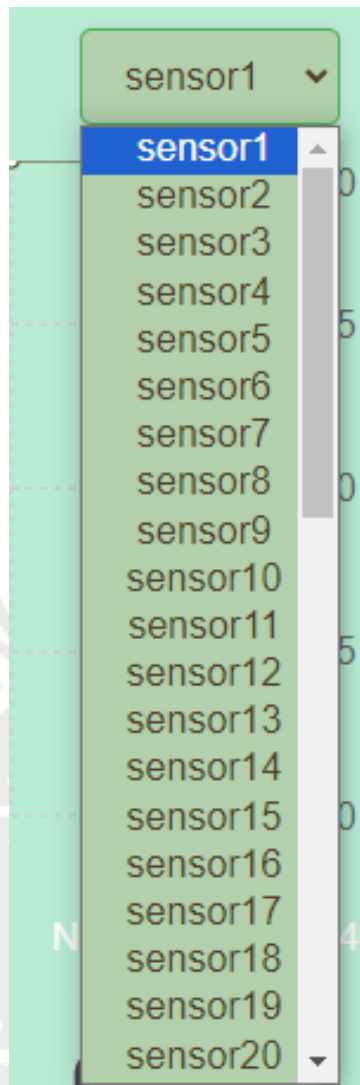


Figura 4.27: Selector de nodo de sensor de maceta[Elaboración propia]

4.3.3. Validación de visualización de los datos ingresados

Para esta validación, se tomará en cuenta que realizaremos la visualización entre los días 19 y 20 de noviembre del 2023 para el nodo de sensor de la maceta 1. A continuación se muestra los resultados obtenidos. Para este caso, tenemos un valor de humedad de 20 %, temperatura de 20°C y batería en 40 %. Se debe tomar en cuenta, el último valor que se tiene de batería es el que se muestra para tener el valor más actualizado de la maceta.

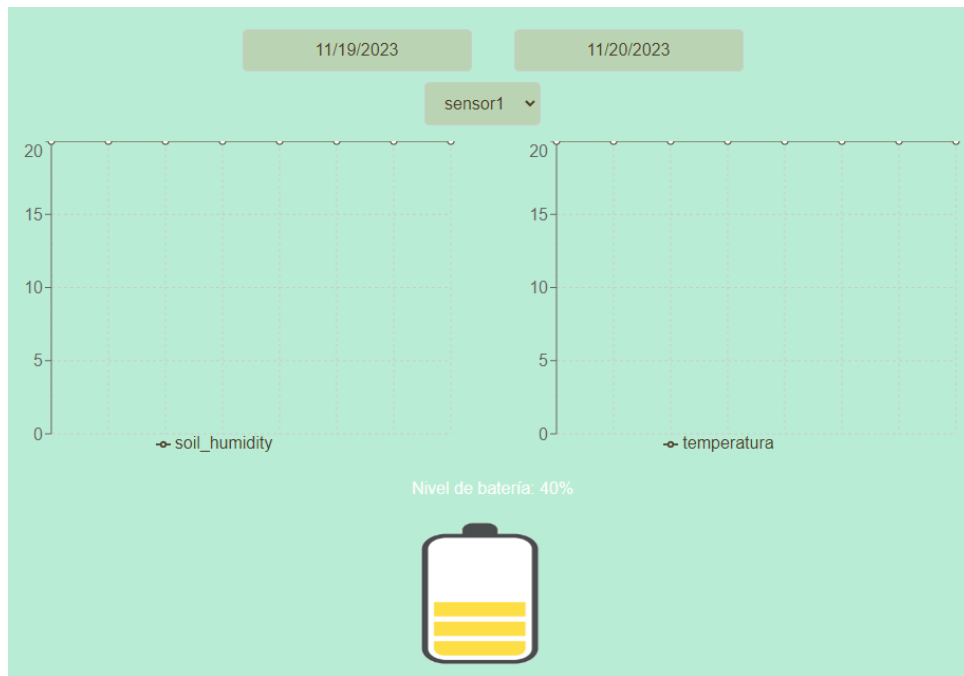


Figura 4.28: Datos seleccionados del nodo de sensor 1[Elaboración propia]

Ahora se procederá a realizar el seleccionamiento de la maceta 50 entre los días 19 y 20 de noviembre. Se debe obtener los dos únicos datos ingresados.

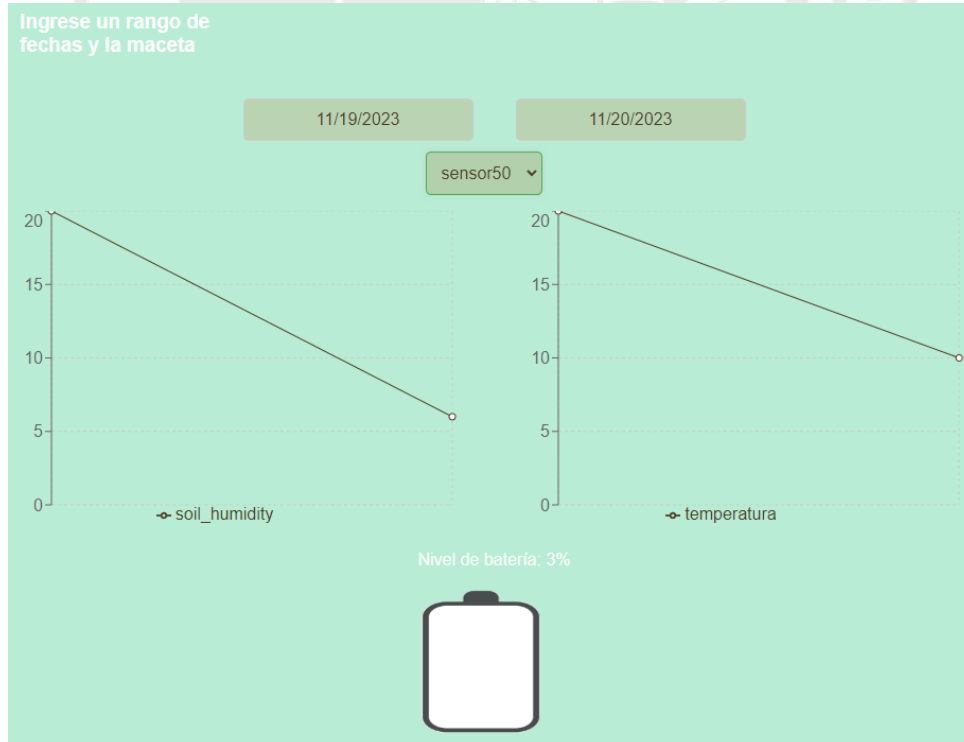


Figura 4.29: Primeros datos seleccionados del nodo de sensor 50[Elaboración propia]

Se puede visualizar exitosamente esos datos de humedad de 6 %, temperatura de 10°C y el 3 % de batería.

Ahora se procederá a realizar el seleccionamiento de la maceta 50 entre los días 17 y 18 de noviembre. No se deben obtener datos ya que no hay datos. Cabe recalcar que todos los datos se actualizan en tiempo real según la selección. Sin embargo, si se entran nuevos datos, se debe actualizar la página web.

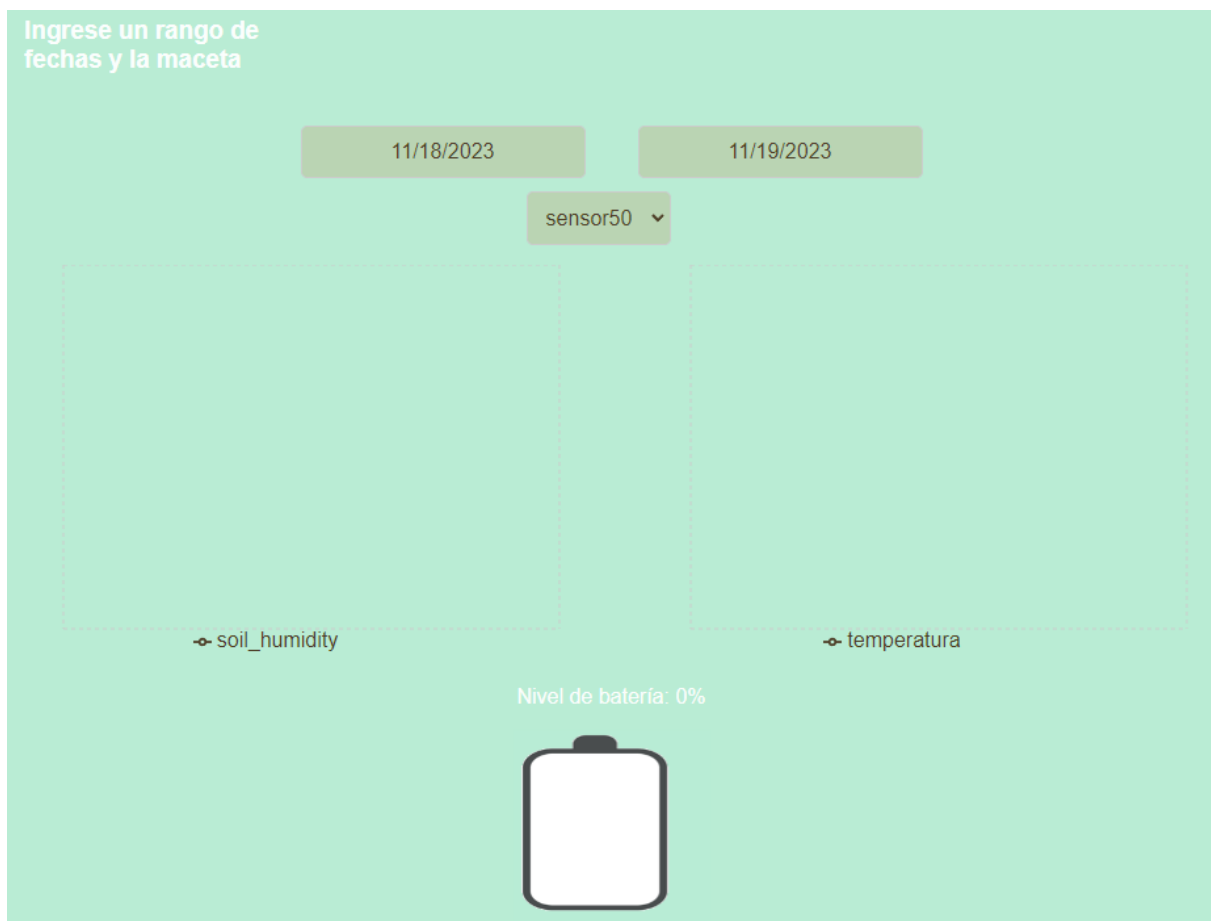


Figura 4.30: Segundos datos seleccionados del nodo de sensor 50[Elaboración propia]

Para finalizar se enviará un nuevo valor al sensor 50 que serán los siguientes: 55 % humedad, 30°C de temperatura y 100 % de batería.

Primero se realizará la simulación de envío mediante el envío del cliente de MQTT.A continuación se muestra.

Pruebas_Tesis

Tesis/Sensores/Envios

Tesis/Sensores/En... QoS 0

JSON All Received Published

batteryLevel: 100

2023-11-20 02:06:58:866

Topic: Tesis/Sensores/Envios QoS: 0

```
{
  "device": "Sensor50",
  "name": "Sensor50",
  "location": {
    "latitude": 45,
    "longitude": -15
  },
  "soil_type": "Suelo1",
  "nutrients": {
    "nitrogen": 0,
    "phosphorus": 0,
    "potassium": 0
  },
  "pH": 0,
  "temperatura": 30,
  "soil_humidity": 55,
  "batteryLevel": 100
}
```

2023-11-20 02:06:58:976

JSON QoS 0 Retain Meta

Tesis/Sensores/Envios

```
{
  "device": "Sensor50",
  "name": "Sensor50",
  "location": {
    "latitude": 45,
    "longitude": -15
  },
  "soil_type": "suelo1",
  "nutrients": {
    "nitrogen": 0,
    "phosphorus": 0,
    "potassium": 0
  },
  "pH": 0,
  "temperatura": 30,
  "soil_humidity": 55,
  "batteryLevel": 100
}
```

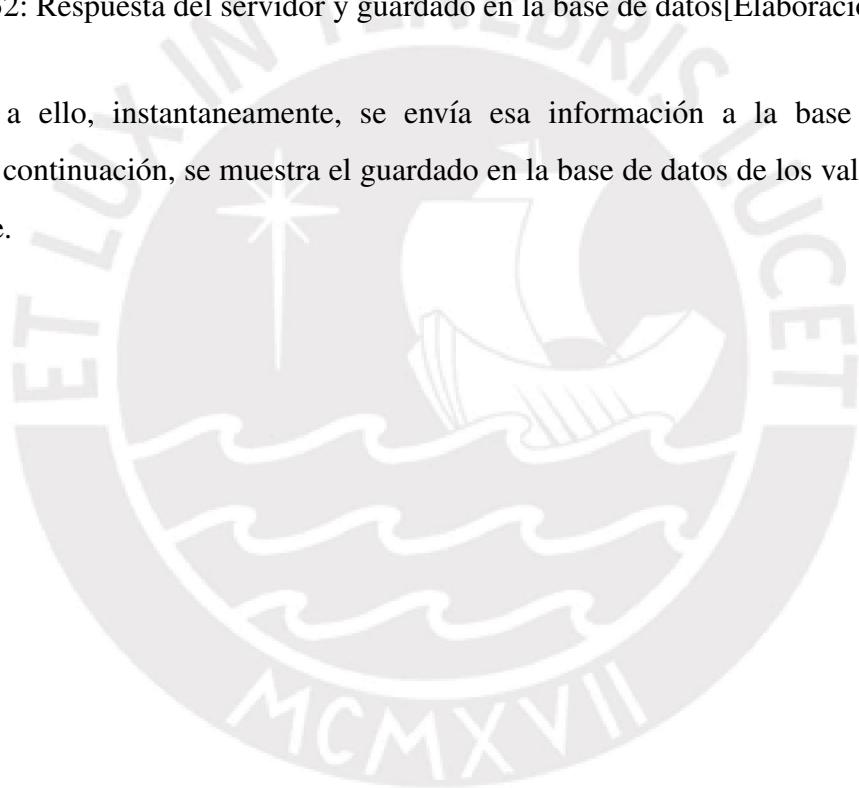
Figura 4.31: Envío de datos para la maceta 50[Elaboración propia]

Asimismo, se mostrará la respuesta del servidor que es inmediata.

```
Nov 20 02:06:59 AM INFO Datos recibidos desde MQTT: {
Nov 20 02:06:59 AM INFO   device: 'Sensor50',
Nov 20 02:06:59 AM INFO   name: 'Sensor50',
Nov 20 02:06:59 AM INFO   location: { latitude: 45, longitude: -15 },
Nov 20 02:06:59 AM INFO   soil_type: 'Suelo1',
Nov 20 02:06:59 AM INFO   nutrients: { nitrogen: 0, phosphorus: 0, potassium: 0 },
Nov 20 02:06:59 AM INFO   pH: 0,
Nov 20 02:06:59 AM INFO   temperatura: 30,
Nov 20 02:06:59 AM INFO   soil_humidity: 55,
Nov 20 02:06:59 AM INFO   batteryLevel: 100
Nov 20 02:06:59 AM INFO }
Nov 20 02:06:59 AM INFO 2023-11-20T07:06:59.475Z
Nov 20 02:06:59 AM INFO object
Nov 20 02:06:59 AM INFO Datos del Sensor50 creados en MongoDB.
```

Figura 4.32: Respuesta del servidor y guardado en la base de datos[Elaboración propia]

Posterior a ello, instantaneamente, se envía esa información a la base de datos en MondoDB.A continuación, se muestra el guardado en la base de datos de los valores enviados anteriormente.



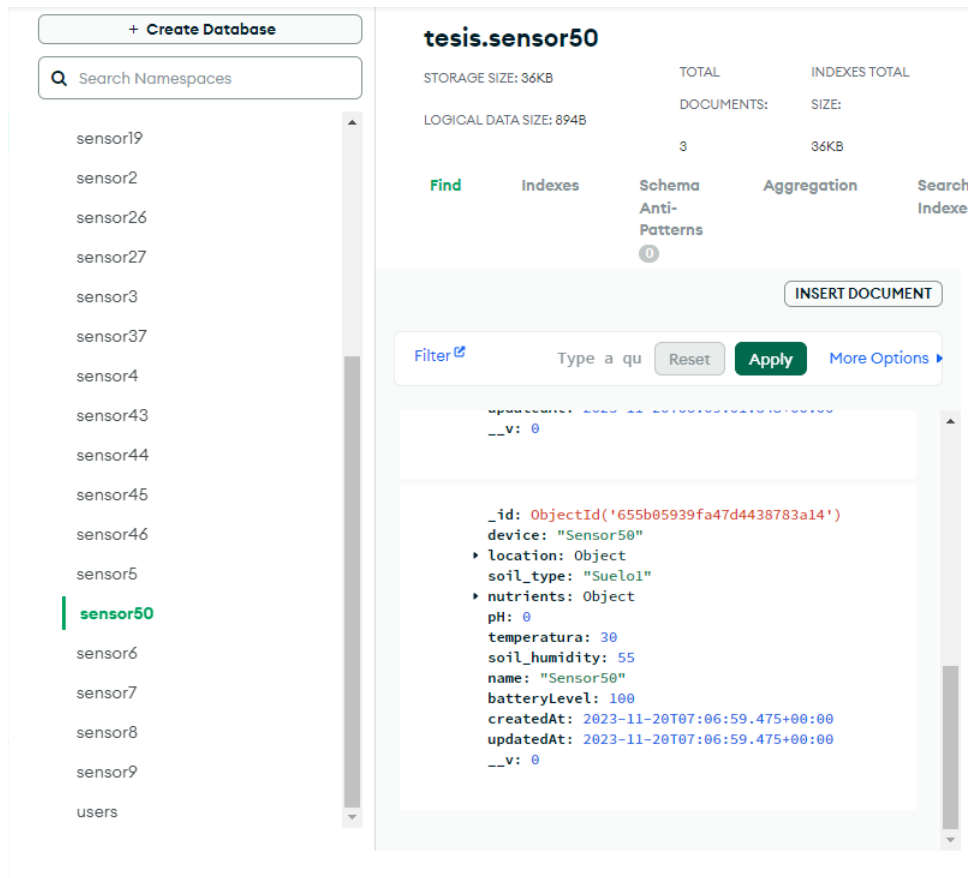


Figura 4.33: Respuesta del servidor y guardado en la base de datos[Elaboración propia]

Por otra parte, en la página web se realiza una solicitud para el servidor con los datos ingresados por el usuario en el enlace correspondiente para obtener los datos y renderizarlos. A continuación se muestra la solicitud en el servidor.

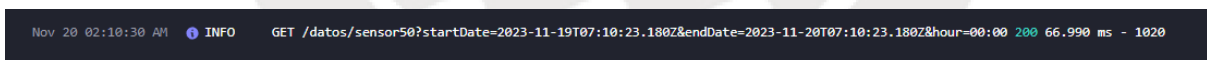


Figura 4.34: Solicitud de datos al servidor[Elaboración propia]

A continuación, se muestra la obtención de los valores de la base de datos por la página web para visualizar esos valores. Previamente, se tuvo que actualizar la página web.

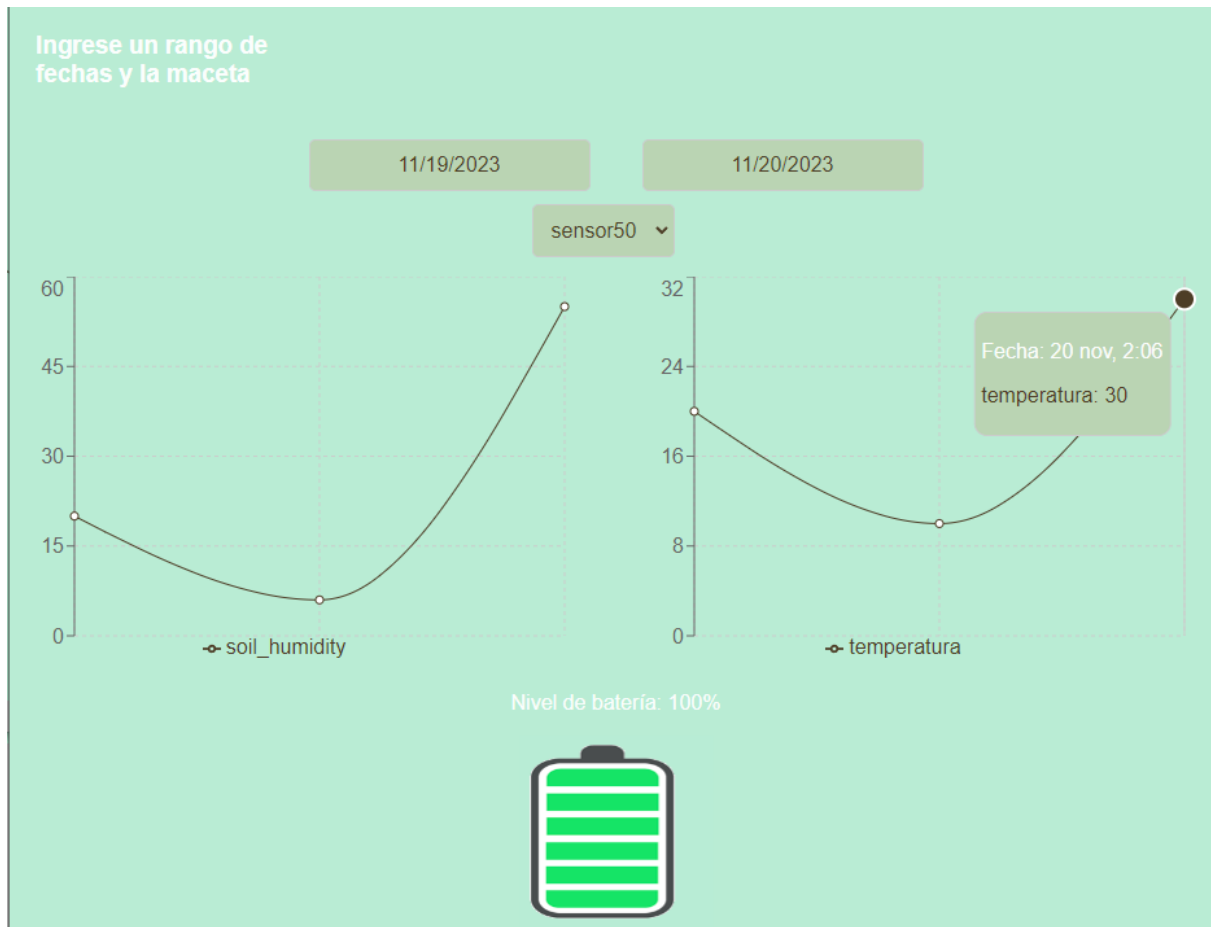


Figura 4.35: Datos actualizados para el sensor 50[Elaboración propia]

4.3.4. Validación de visualización multiplataforma

Para esta validación, se mostrará toda la interfaz desarrollada en un celular Xiaomi Redmi Note 11.

Primero empezamos con la pantalla de inicio.



Figura 4.36: Pantalla de inicio en celular[Elaboración propia]

Asimismo, su respectiva alerta de autenticación exitosa.

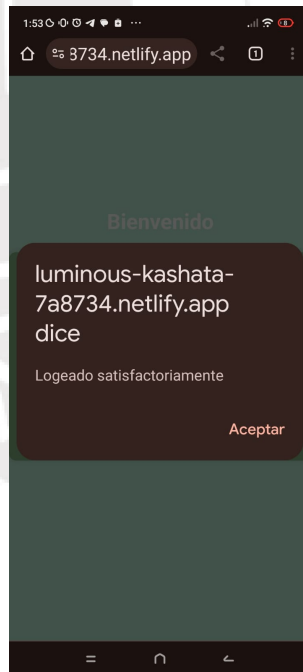


Figura 4.37: Autenticación exitosa en celular[Elaboración propia]

Ahora se visualiza la interfaz de usuario correspondiente.



Figura 4.38: Interfaz de usuario en celular[Elaboración propia]

Asimismo se visualiza la barra de navegación.

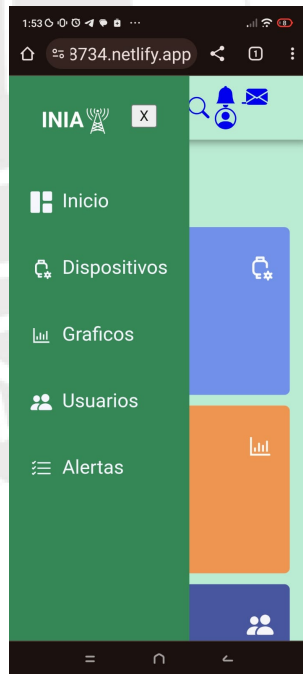


Figura 4.39: Barra de navegación en celular[Elaboración propia]

Por otra parte se muestran los distintos selectores para visualizar los datos elegidos en el celular.

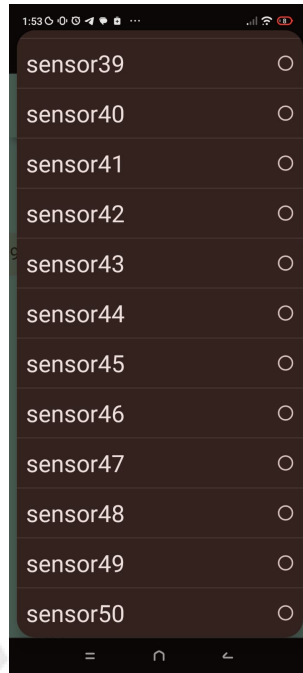


Figura 4.40: Selector de nodo de sensor en celular[Elaboración propia]

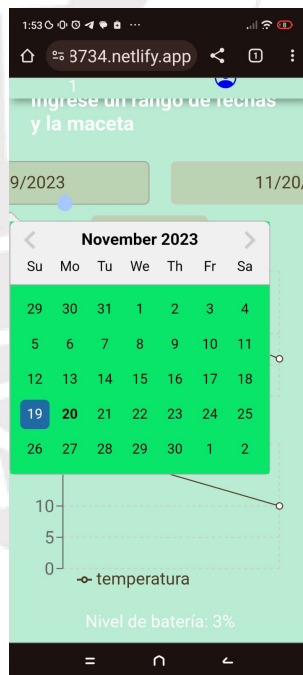


Figura 4.41: Selector de fecha de inicio en celular[Elaboración propia]

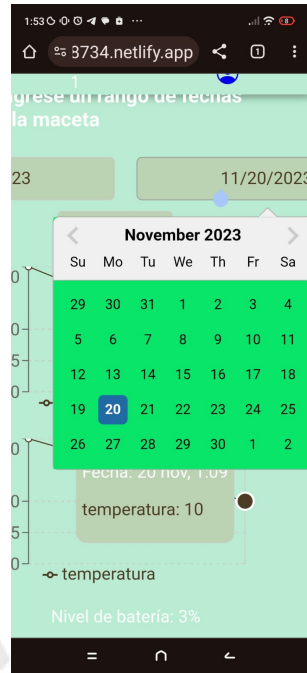


Figura 4.42: Selector de fecha de fin en celular[Elaboración propia]

Ahora se visualizará la selección elegida en el celular. En este caso, se tomó la visualización hasta antes de ingresar nuevos datos para el nodo de sensor 50.

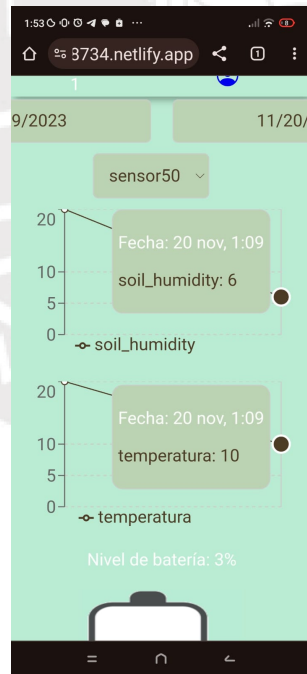


Figura 4.43: Primeros datos en nodo de sensor 50 en celular[Elaboración propia]

Ahora se visualizará la visualización de los datos después de ingresar nuevos datos para el

nodo de sensor 50.

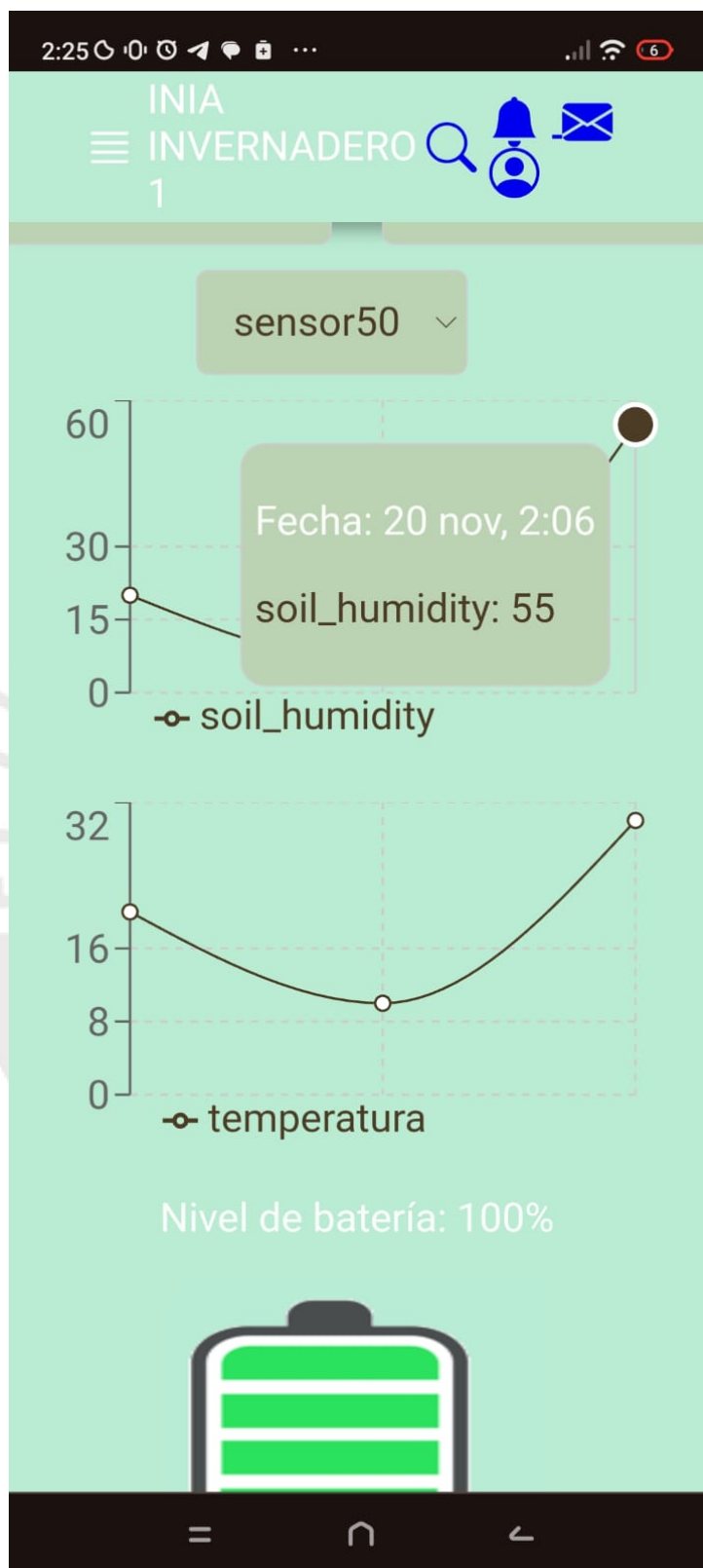


Figura 4.44: Segundos datos en nodo de sensor 50 en celular[Elaboración propia]

Conclusiones

A continuación, se presentarán las conclusiones acerca de toda la tesis.

- Se logró el proceso de obtención de los datos de humedad de suelo y temperatura de suelo realizando el protocolo MQTT para la comunicación entre el nodo de sensor con la página web.
- Se logró diseñar el nodo de sensor que cuenta con un sensor integrado para medir variables de suelo, específicamente la humedad y temperatura de suelo.
- Se logró establecer cómo se colocarán los nodos de sensores para obtener una red de sensores para las 50 macetas necesarias.
- Se diseñó para la implementación de una red de sensores para las 50 macetas del invernadero en el INIA, a través de un circuito impreso para la conexión de los sensores a la batería.
- Se desarrolló de un entorno amigable mediante la realización de un página web para que el usuario visualice mediante gráficos estadísticos los datos de temperatura y humedad de suelo correspondientes a la maceta que seleccione.
- Se logró realizar un alojamiento web, mediante herramientas de desarrollo gratuito para el visualizado de la información de los datos a través de Internet.
- Se logró brindar un acceso restringido hacia la página web mediante el uso de una sistema de autenticación para que solo el personal autorizado pueda acceder a la página y obtener la información correspondiente.

Recomendaciones

A continuación, se presentarán las recomendaciones para la tesis.

-Utilizar sensores comerciales y comparar los resultados con los valores obtenidos. Mejoraría la precisión de estos ayudando a realizar modificaciones extras en la calibración de los sensores utilizados.

-Se recomienda usar otros protocolos de comunicación de red para poder obtener menor consumo para el envío de datos por parte del microcontrolador. Así reduciendo más el uso de la batería.

-Utilizar módulos de desarrollo para el entorno de pruebas es aceptable para conseguir resultados; sin embargo, se recomienda hacer una tarjeta de desarrollo para evitar usar componentes innecesarios que se utilizan en los módulos de desarrollo ya que suele haber componentes como el regulador de tensión, que suelen consumir demasiado y drena la batería.

-Al utilizar baterías debe colocarse un circuito extra para estabilizar la energía necesaria para obtener un valor estable para alimentar el microcontrolador y el sensor utilizado.

-Para realizar un servidor o backend es necesario contar con un framework determinado para poder realizar las conexiones correspondientes.

-Se logró realizar la prueba de concepto para la solución de las 50 macetas para una maceta del invernadero del INIA.

-Para obtener una actualización de los datos en la página en tiempo real, se deben utilizar Websockets para lograr un actualizado en tiempo real de los datos.

-Para poder realizar la visualización de los resultados al usuario, sería recomendable brindarles un aplicativo móvil para mejorar la experiencia al usuario que utilizará la app constantemente.

Bibliografía

- [1] Dorsemayne, B., Gaulier, J. P., Wary, J. P., Kheir, N., & Urien, P. (2016). *Internet of Things: A Definition and Taxonomy. Proceedings - NGMAST 2015: The 9th International Conference on Next Generation Mobile Applications, Services and Technologies*, 72–77. <https://doi.org/10.1109/NGMAST.2015.71>
- [2] Instituto Nacional de Innovación Agraria - INIA. Plataforma del Estado Peruano. Recuperado el 7 de abril de 2023, de <https://www.gob.pe/inia>.
- [3] Food and Agriculture Organization of the United Nations. Recuperado el 12 de mayo de 2023, de <https://www.fao.org/sustainability/background/es/>.
- [4] J. Pérez. (2022). *Agricultura 4.0: Desafíos y Oportunidades para los Institutos Nacionales de Investigación Agropecuaria de América Latina y el Caribe*. *Revista de Investigación Agropecuaria*, 45(2), 67-75. doi:<http://dx.doi.org/10.18235/0003584>.
- [5] World Bank Open Data (2023) 1. *Agricultura, valor agregado del PIB*. Recuperado el 12 de mayo de 2023, de <https://datos.bancomundial.org/indicador/NV.AGR.TOTL.ZS?end=2021&start=2015>.
- [6] World Bank Open Data (2023) 2. *Agricultura, valor agregado en miles de dólares*. World Bank Open Data 2023. Recuperado el 12 de mayo de 2023, de <https://datos.bancomundial.org/indicador/NV.AGR.TOTL.KD?end=2021&start=2015&view=chart>
- [7] MIDAGRI (Febrero 2023). Boletín Mensual *El Agro en Cifras*. pp.112. Recuperado de <https://www.gob.pe/institucion/midagri/informes-publicaciones/4024332-boletin-estadistico-mensual-el-agro-en-cifras-2023>.

- [8] Paul, S., Sunil, B., & Mohammed Ali, A. A. A. (2022). *Green IoT and sustainability in the U.A.E. post pandemic for an eco-friendly and sustainable smart agriculture*. 8th International Conference on Information Technology Trends: Industry 4.0: Technology Trends and Solutions, ITT 2022, 124–129. <https://doi.org/10.1109/ITT56123.2022.9863936>.
- [9] Mahajan, P. (2021, October 1). *Internet of things revolutionizing Agriculture to Smart Agriculture*. 2021 2nd Global Conference for Advancement in Technology, GCAT 2021. <https://doi.org/10.1109/GCAT52182.2021.9587896>.
- [10] jxctdzkj (2023). *Sensor integral de suelo JXBS-3001-TH.*. Recuperado el 13 de mayo de 2023, de <http://jxctdzkj.com/category/productDetail/3969.html>.
- [11] Al-Humairi, S. N. S., Manimaran, P., Abdullah, M. I., & Daud, J. (2019). *A Smart Automated Greenhouse: Soil Moisture, Temperature Monitoring and Automatic Water Supply System (Peaty, Loam and Silty)*. En 2019 IEEE Conference on Sustainable Utilization and Development in Engineering and Technologies, CSUDET 2019 (pp. 111-115). <https://doi.org/10.1109/CSUDET47057.2019.9214661>.
- [12] Postolache, S., Sebastião, P., Viegas, V., Postolache, O., & Cercas, F. (2023). *IoT-Based Systems for Soil Nutrients Assessment in Horticulture*. *Sensors*, 23(1). <https://doi.org/10.3390/s23010403>.
- [13] Manglani, T., Vaishnav, A., Solanki, A. S., & Kaushik, R. (2022). *Smart Agriculture Monitoring System Using Internet of Things (IoT)*. Proceedings of the International Conference on Electronics and Renewable Systems, ICEARS 2022, 501–505. <https://doi.org/10.1109/ICEARS53579.2022.9752446>.
- [14] Dholu, M., & Ghodinde, K. A. (2018). *Internet of Things (IoT) for Precision Agriculture Application*. In 2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI) (pp. 339-342). Tirunelveli, India. <http://doi.org/10.1109/ICOEI.2018.8553720>
- [15] Islam, M. F., Hossain, M. T., & Amin, M. R. (2018). *An IoT based system for telemetry and control of Greenhouse environment*. In 2018 International Conference

- on Networking, Systems and Security (NSysS) (pp. 1-6). Dhaka, Bangladesh. doi: 10.1109/NSYSS.2018.8665518.
- [16] Anghelof, M. M., Suciu, G., Craciunescu, R., & Marghescu, C. (2020). *Intelligent System for Precision Agriculture*. In 2020 13th International Conference on Communications (COMM) (pp. 407-410). Bucharest, Romania. doi: 10.1109/COMM48946.2020.9141981.
- [17] Rayhana, R., Xiao, G., & Liu, Z. (2020). *Internet of Things Empowered Smart Greenhouse Farming*. IEEE Journal of Radio Frequency Identification, 4(3), 195-211. doi: 10.1109/JRFID.2020.2984391.
- [18] Von Zabeltitz, C. (2011). *Greenhouse Structures*. In Integrated Greenhouse Systems for Mild Climates. Springer, Berlin, Heidelberg. doi:10.1007/978-3-642-14582-7
- [19] Lloret, J., Tomas, J., Canovas, A., & Parra, L. (2016). *An Integrated IoT Architecture for Smart Metering*. IEEE Communications Magazine, 54(12), 50-57. doi: 10.1109/MCOM.2016.1600647CM
- [20] He, Y., Nie, P., Zhang, Q., & Liu, F. (2021). *Agricultural Internet of Things* (Vol. 1, pp. 1-21). Springer Nature Switzerland AG. doi: 10.1007/978-3-030-65702-4.
- [21] A. Anand, N. K. Trivedi, V. Gautam, R. G. Tiwari, D. Witarasyah, & A. Misra. (2022). *Applications of Internet of Things (IoT) in Agriculture: The Need and Implementation*. In 2022 International Conference Advancement in Data Science, E-learning and Information Systems (ICADEIS) (pp. 01-05). Bandung, Indonesia. doi: 10.1109/ICADEIS56544.2022.10037505.
- [22] F. A. Urbano-Molano. (2013). *Redes de Sensores Inalámbricos Aplicadas a Optimización en Agricultura de Precisión para Cultivos de Café en Colombia*. Journal de Ciencia e Ingeniería, Vol.5, No.1, Agosto de 2013, pp. 46-52. ISSN 2145-2628. Obtenido de <https://jci.uniautonoma.edu.co/2013/2013-8.pdf>
- [23] Bersani, C., Ruggiero, C., Sacile, R., Soussi, A., & Zero, E. (2022). *Internet of Things Approaches for Monitoring and Control of Smart Greenhouses in Industry 4.0*. Energies, 15(10), 3834. <https://doi.org/10.3390/en15103834>

- [24] Dos Santos, R. P., Leithardt, V. R. Q., & Beko, M. (2022). *Analysis of MQTT-SN and LWM2M communication protocols for precision agriculture IoT devices*. In 2022 17th Iberian Conference on Information Systems and Technologies (CISTI) (pp. 1-6). Madrid, Spain. doi: 10.23919/CISTI54924.2022.9820048
- [25] Ramirez, J., & Pedraza, C. (2017). *Performance analysis of communication protocols for Internet of things platforms*. In 2017 IEEE Colombian Conference on Communications and Computing (COLCOM) (pp. 1-7). Cartagena, Colombia. doi: 10.1109/ColComCon.2017.8088198
- [26] Tisza, J., & Ortega, D. (2022). *Architecture proposal for real-time sensor monitoring using IoT and Augmented Reality*. In 2022 IEEE Engineering International Research Conference (EIRCON) (pp. 1-4). Lima, Peru. doi: 10.1109/EIRCON56026.2022.9934808
- [27] Al Enany, M. O., Harb, H. M., & Attiya, G. (2021). *A Comparative analysis of MQTT and IoT application protocols*. In 2021 International Conference on Electronic Engineering (ICEEM) (pp. 1-6). Menouf, Egypt. doi: 10.1109/ICEEM52022.2021.9480384
- [28] Taivalsaari, A., & Mikkonen, T. (2018). *A Taxonomy of IoT Client Architectures*. IEEE Software, 35(3), 83-88. doi: 10.1109/MS.2018.2141019.
- [29] Microchip (2023). *Microchip*. Recuperado el 10 de junio de 2023, de <https://www.microchip.com/>
- [30] Espressif (2023). *Espressif*. Recuperado el 13 de junio de 2023, de <https://www.espressif.com/>
- [31] Megibar Pozo, L. (2015). *UART: Universal Asynchronous Receiver-Transmitter* (Ingeniería en Tecnologías Industriales). Universidad Carlos III de Madrid. Departamento de Tecnología Electrónica. Recuperado el 15 de junio de 2023, de <http://hdl.handle.net/10016/23730>.
- [32] Webster, J. G. (1998) *2. pH Measurement*. En Measurement, Instrumentation, and Sensors Handbook. CRC Press.

- [33] Gutiérrez & Iturralde, 2017. *Fundamentos Básicos de Instrumentación y Control*. Serie de textos de la Facultad de Ciencias de la Ingeniería. Pág 28-33. Universidad Estatal Península de Santa Elena.
- [34] Aakvaag, N., & Frey, J.-E. (2006). *Redes de sensores inalámbricos* (Vol. 2). ABB.
- [35] Saigua Carvajal, S. F., Villafuerte Haro, M. V., Ávila-Pesantez, D. F., & Arellano Aucancela, A. L. (2016). *Evaluación de las topologías físicas de redes WSN para la medición de variables ambientales*. Universidad Estatal Península de Santa Elena, 3(1), 159–165. Recuperado de <https://repositorio.upse.edu.ec/handle/46000/7380>
- [36] Vela Velasco, A. M. (2016). *Estudio de la agricultura de precisión enfocado en la implementación de una red de sensores inalámbricos (WSN) para el monitoreo de humedad y temperatura en cultivos - caso de estudio hacienda Cabalinus ubicada en la provincia de Los Ríos* (p. 1–17). Recuperado de <http://repositorio.puce.edu.ec/handle/22000/11112>
- [37] Zhang, R., Chen, L., Guo, J., Meng, Z., & Xu, G. (2010). *An energy-efficient wireless sensor network used for farmland soil moisture monitoring*. En IET International Conference on Wireless Sensor Network 2010 (IET-WSN 2010) (pp. 2-6). Beijing. doi: 10.1049/cp.2010.1017.
- [38] Lopez D. & Maya E. (2017). *Arquitectura de Software basada en Microservicios para Desarrollo de Aplicaciones Web* En Séptima Conferencia de Directores de Tecnología de Información, TICAL 2017 Gestión de las TICs para la Investigación y la Colaboración, San José, del XX al XX de julio de 2017. Imbabura, Ecuador.
- [39] Colhyer de Paul Pereira, M. J. (2022). *Analizador de energía utilizando o chip ATM90E36 e Modulo ESP8266 con arquitectura restful* (Obtención do título de Engenheiro Eletricista). Universidade Federal Do Amazonas Faculdade de Tecnologia Engenharia Electrica- Electronica.
- [40] Gupta, A., Gulati, T., & Bindal, A. K. (2022). *WSN based IoT applications: A Review*. En 2022 10th International Conference on Emerging Trends in Engineering and Technology - Signal and Information Processing (ICETET-SIP-22) (pp. 1-6). Nagpur, India. doi: 10.1109/ICETET-SIP-2254415.2022.9791495.

- [41] Altronics 2023. *Sensor de Temperatura y Humedad SHT31*. Altronics.cl. Recuperado el 12 de octubre de 2023, de <https://altronics.cl/sensor-sht31>
- [42] Electrónica, M. (2020, marzo 13). *Sensor de Temperatura de Suelo + Humedad. ¡Optimiza el uso de agua!* Maher Electrónica. <https://www.maherelectronica.com/sensores-agricolas-suelo/bi-sensor-de-temperatura-y-humedad-de-suelo/>
- [43] Ambimet 2021. *Sensor combinado de temperatura y humedad del suelo Truebner SMT100*. Ambimet Instrumentación. Recuperado el 11 de octubre de <https://www.ambimet-instrumentacion.cl/producto/sensor-combinado-de-temperatura-y-humedad-del-suelo-truebner-smt100/>
- [44] Aranet, *Aranet Soil Sensor Teros 12* Aranet, 2025. [En línea]. Disponible: <https://aranet.com/en/pro/products/aranet-soil-sensor-teros-12>. [Accedido: 9 de febrero de 2025].
- [45] ESPS2 2023. *ESP32S2WROVER(4MB)*. Espressif. Recuperado el 13 de octubre de 2023, de https://www.espressif.com/sites/default/files/documentation/esp32-s2-wrover_esp32-s2-wrover-i_datasheet_en.pdf
- [46] ESPWROOM32E 2023. *ESP32WROOM32E(4MB)*. Espressif. Recuperado el 13 de octubre de 2023, de https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e_esp32-wroom-32ue_datasheet_en.pdf
- [47] DA16200 2023. *DA16200*. Renesas. Recuperado el 14 de octubre de 2023, de <https://www.renesas.com/us/en/document/dst/da16200mod-datasheet?r=1599971>
- [48] WFI32E01PC 2023. *WFI32E01PC*. Microchip. Recuperado el 15 de octubre de 2023, de <https://www.microchip.com/en-us/product/WFI32E01PC>
- [49] SMG 2023. *SGM221*. SG-micro. Recuperado el 17 de octubre de 2023, de <https://www.sg-micro.com/uploads/soft/20220506/1651829970.pdf>
- [50] MP 2023. *MP2322*. Mouser. Recuperado el 18 de octubre de 2023, de <https://www.mouser.es/ProductDetail/Monolithic-Power-Systems-MPS/MP2322GQH-Z?qs=byeeYqUIh0PNRfOYikm40g%3D%3D>

- [51] RP118 2023. *RP118*. Mouser. Recuperado el 18 de octubre de 2023, de https://www.mouser.es/datasheet/2/294/rp118_ea-3219867.pdf
- [52] RP170 2023. *RP170N331B-TR-FE*. SG-micro. Recuperado el 18 de octubre de 2023, de https://www.mouser.es/datasheet/2/294/rp170_ea-3219762.pdf
- [53] Batería1. *PLANA 3.7V 6000MAH 806090* Naylamp Mechatronics - Perú. Recuperado el 15 de octubre de 2023, de <https://naylampmechatronics.com/baterias/1069-bateria-plana-37v-6000mah-806090.html>
- [54] Batería2. *Plana 3.7V 1400mAh 806090* Naylamp Mechatronics - Perú. Recuperado el 14 de octubre de 2023, de <https://naylampmechatronics.com/baterias/1067-bateria-plana-37v-1400mah-454261.html>
- [55] Batería3. *Lifepo4 Batería 18650 6000mah 3,3 V* Alibaba.com. Recuperado el 14 de octubre de 2023, de <https://spanish.alibaba.com/product-detail/Lifepo4-Battery-18650-6000mAh-3-3V-60644850491.html>
- [56] Next 2023. *Nextjs*. Recuperado el 13 de octubre de 2023, de <https://nextjs.org/docs>
- [57] Vite 2023. *Vitejs*. Recuperado el 14 de octubre de 2023, de <https://vitejs.dev/guide/features.html>
- [58] Node 2023. *Node* Recuperado el 16 de octubre de 2023, de <https://nodejs.org/en/docs>
- [59] SpringBoot. *SpringBoot* Recuperado el 16 de octubre de 2023, de <https://spring.io/projects/spring-boot>
- [60] Google Cloud 2023. *Google Cloud 2023* Recuperado el 14 de octubre de 2023, de <https://cloud.google.com/docs?hl=es>
- [61] AWS 2023. *Información general sobre Amazon Web Services*. Amazon.com. Recuperado el 13 de octubre de 2023, de <https://docs.aws.amazon.com/whitepapers/latest/aws-overview/aws-overview.pdf>
- [62] MongoDB 2023. *Documentación MongoDB*. Recuperado el 13 de octubre de 2023, de <https://www.mongodb.com/collateral/mongodb-atlas-datasheet>

[63] Render 2023. *Documentación de Render*. Render. Recuperado el 15 de noviembre de 2023, de <https://render.com/docs>

[64] Postman 2023. *Documentación de Postman* Postman. Recuperado el 17 de noviembre de 2023, de <https://learning.postman.com/docs/introduction/overview/>

[65] Netlify 2023. *Documentación de Netlify*. Recuperado el 16 de noviembre de 2023, de <https://docs.netlify.com/>

[66] EMQX 2023. *Documentación de EMQX*. Recuperado el 16 de noviembre de 2023, de [/https://www.emqx.com/en](https://www.emqx.com/en)

