

Sistema de riego inteligente con Bluetooth e IoT para el cuidado de plantas.

Valerio Lucas, Colantonio

Grau de Multimèdia

Arduino

Tutor de TF

Jose, Lopez Vicario

Professor responsable

de l'assignatura

Pere, Tuset Peiró

Universitat Oberta
de Catalunya

Data Lliurament 19/06/23



Aquesta obra està subjecta a una llicència de
[Reconeixement-NoComercial-CompartirIgual 3.0 Espanya de Creative Commons](https://creativecommons.org/licenses/by-nc-sa/3.0/es/)

FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>Sistema de riego inteligente con Bluetooth e IoT para el cuidado de plantas.</i>
Nombre del autor:	<i>Valerio Colantonio</i>
Nombre del consultor/a:	<i>Jose, Lopez Vicario</i>
Nombre del PRA:	<i>Pere, Tuset Peiró</i>
Fecha de entrega:	<i>19/06/2023</i>
Titulación o programa:	<i>Grau de Multimèdia</i>
Área del Trabajo Final:	<i>Arduino</i>
Idioma del trabajo:	<i>Castellano</i>
Palabras clave	<i>Arduino, Esp32, IoT</i>
Resumen del Trabajo	
<p>El creciente interés por el cuidado de plantas ornamentales y huertos urbanos en áreas urbanas contrasta con la falta de habilidad de los habitantes de la ciudad para cuidarlas adecuadamente.</p> <p>Este proyecto tiene como objetivo utilizar la tecnología como una herramienta de apoyo para este cuidado. Consiste en diferentes módulos independientes que se comunican entre sí mediante la tecnología BLE.</p> <p>El módulo de medición de humedad monitorea de forma continua la humedad del sustrato de las plantas, mientras que el módulo de riego suministra el agua necesaria para garantizar su supervivencia.</p> <p>Por otro lado, el módulo principal de IoT se encarga de recibir los datos de los diferentes módulos y determinar qué plantas requieren riego. Además, este último módulo se conecta a través de WiFi a un servicio IoT en la nube que recopila y procesa los datos, poniéndolos a disposición de los usuarios en cualquier parte del mundo.</p>	

Abstract

The growing interest in the care of ornamental plants and urban gardens in urban areas contrasts with the lack of ability of city dwellers to properly care for them.

This project aims to use technology as a tool to assist in this care. It consists of different independent modules that communicate with each other using BLE technology.

The moisture measurement module constantly monitors the moisture level of the plants' substrate, while the irrigation module supplies the necessary water to ensure their survival.

On the other hand, the main IoT module is responsible for receiving data from the different modules and determining which plants need watering. Additionally, this last module is connected via WiFi to an IoT cloud service that collects and processes the data, making it available to users anywhere in the world.

Índice

Índice.....	5
1 Introducción.....	6
1.1 Contexto y motivación del proyecto.....	6
1.2 Objetivos del trabajo.....	7
1.3 Impacto en sostenibilidad, ético-social y de diversidad.....	15
1.4 Enfoque y método seguido.....	16
1.5 Planificación del trabajo.....	17
1.6 Breve resumen de los productos obtenidos.....	21
1.7 Breve descripción de los otros capítulos de la memoria.....	21
2 Investigación.....	22
2.1 Estado del arte.....	22
2.2 Aspectos de ciberseguridad a tener en cuenta.....	33
2.3 Protocolo de comunicación para servicios de IoT en la nube.....	34
2.4 Estudio de viabilidad.....	36
3 Diseño.....	38
3.1 Diagrama de bloques general.....	38
4 Desarrollo.....	40
4.1 Módulo de medición de humedad.....	40
4.2 Módulo de riego.....	45
4.3 Módulo principal de IoT.....	48
5 Resultados.....	52
6 Conclusiones y trabajos futuros.....	53
7 Bibliografía.....	55
8 Anexos.....	58
8.1 Anexo 1: pruebas realizadas.....	58
8.2 Anexo 2: diferencias entre MQTT y CoAP y elección del protocolo a utilizar.....	60
8.3 Módulo principal de IoT.....	61
8.4 Esquema funcional del menú del módulo principal de IoT.....	64

1 Introducció

1.1 Contexto y motivación del proyecto

Desde el comienzo del Neolítico, hace unos 12.000 años, en Oriente Próximo, el cultivo de plantas ha sido una de las actividades más importantes para la humanidad, tanto para la alimentación como para la ornamentación de lugares.

Así, a lo largo del tiempo, se han desarrollado diferentes técnicas y herramientas para el cuidado de las plantas, y es a partir del siglo xx que todas las ramas tecnológicas englobadas en la Agronomía han sufrido un avance espectacular y las potencialidades de producción, conservación y transporte de alimentos, se han incrementado de manera ostensible [1].

En la actualidad, los avances tecnológicos han permitido una gran automatización en el cultivo de plantas, lo que ha transformado esta práctica ancestral en una industria moderna, permitiendo un mayor rendimiento y una reducción de costos [2].

Sin embargo, no todas las noticias son positivas en este ámbito. La industria alimentaria, tiene un impacto ambiental significativo. Según la Organización de las Naciones Unidas para la Agricultura y la Alimentación (FAO), la producción de alimentos es responsable de más de un tercio de las emisiones antropógenas mundiales de gases de efecto invernadero, lo que contribuye al calentamiento global y al cambio climático. Sumado a esto, el transporte de alimentos a largas distancias genera emisiones adicionales de gases de efecto invernadero, lo que agrava la situación [3].

En este contexto, existe una tendencia creciente hacia la autoproducción de alimentos en huertos urbanos que se presentan como una solución sostenible y accesible para producir alimentos frescos y saludables, reducir el impacto ambiental y fomentar la resiliencia comunitaria [4].

No obstante, en las zonas urbanas, muchas personas carecen de experiencia en el cuidado de las plantas y se enfrentan a problemas comunes de plantas marchitas o en mal estado debido a la falta de riego, especialmente cuando se encuentran fuera de casa por un periodo prolongado.

Por lo tanto la motivación de este proyecto es doble: por un lado, se busca mejorar el cuidado de las plantas ornamentales en hogares y apartamentos en zonas urbanas, y por otro, se busca mejorar la producción de plantas comestibles en huertos urbanos.

1.1.1 Descripción general de la propuesta

En general, y teniendo en cuenta el contexto mencionado anteriormente, se propone desarrollar un sistema de riego automático e inteligente para plantas de interior y huertos urbanos que requieren un riego regular y adecuado.

El sistema consta de tres módulos:

- El módulo de medición de la humedad será responsable de medir el nivel de humedad en la tierra de las plantas.
- El módulo de riego será responsable de suministrar el agua necesaria a la planta.
- El módulo principal se encargará de recibir los datos del módulo de medición de humedad, decidir si la planta necesita riego y, en caso afirmativo, ordenar al módulo de riego que comience a regar la planta.

La comunicación entre los diferentes módulos del sistema se realizará a través de la tecnología de comunicación BLE (Bluetooth Low Energy). Esto permitirá a los módulos comunicarse de manera inalámbrica y efectiva entre sí.

Además, el módulo principal estará equipado con conectividad WiFi para enviar los datos recopilados por los módulos a un servicio de IoT en la nube. Esto permitirá a los usuarios monitorear el riego de sus plantas desde cualquier lugar utilizando una aplicación en línea o una aplicación móvil.

En resumen, el sistema de riego automático e inteligente para plantas de interior y huertos urbanos ofrecerá una forma fácil y conveniente para que los usuarios mantengan sus plantas adecuadamente regadas y saludables.

1.2 Objetivos del trabajo

El **objetivo principal** de este proyecto es **diseñar, desarrollar y prototipar un sistema de riego automático e inteligente** que brinde una solución sostenible y eficiente a la problemática del riego en plantas de interior y huertos urbanos. Este sistema estará diseñado para adaptarse a las necesidades específicas de cada planta, evitando el desperdicio de agua causado por un riego excesivo.

Además, este proyecto busca alcanzar **objetivos personales** que están estrechamente relacionados con el desarrollo del sistema de riego automático. Estos objetivos incluyen la **aplicación práctica de los conocimientos adquiridos durante los años de estudio en el grado de Multimedia**, particularmente en el diseño e implementación de sistemas basados en microcontroladores. También se busca desarrollar habilidades en electrónica y programación, aplicar patrones de diseño en la programación y aplicar los principios de usabilidad e interfaz de usuario para crear un sistema intuitivo y fácil de usar.

1.2.1 Objetivos técnicos

Para alcanzar el objetivo principal, se han establecido una serie de objetivos técnicos principales y secundarios.

Objetivos técnicos principales:

- Diseñar, desarrollar y prototipar el módulo de medición de la humedad.
- Diseñar, desarrollar y prototipar el módulo de riego automático.
- Diseñar, desarrollar y prototipar el módulo principal de IoT.
- Establecer la conexión del módulo principal de IoT con un servicio de IoT en la nube para recopilar los datos de los módulos.

Objetivos técnicos secundarios:

- Agregar al servicio en la nube de IoT la capacidad de modificar los parámetros de los módulos.
- Desarrollar una aplicación de IoT para smartphones Android que pueda visualizar los datos de los módulos.

A continuación, se proporcionará una definición detallada de cada objetivo, las etapas correspondientes, las pruebas a realizar y la definición del éxito para cada objetivo.

1.2.2 Objetivo principal 1: Módulo de medición de humedad

Diseñar e implementar un módulo de medición de humedad que se pueda insertar en las plantas y que realice mediciones continuas del nivel de humedad del sustrato. Este módulo deberá ser capaz de medir la humedad de la planta y transmitir esos datos al módulo principal de IoT a través de Bluetooth.

Etapas de este objetivo (en parentesi el peso, en porcentaje, de cada punto en el objetivo total del módulo):

1. Selección de componentes electrónicos adecuados, incluyendo el microcontrolador y el sensor de humedad (10% del objetivo).
2. Diseño del esquema electrónico (10% del objetivo).
3. Conexión de los diferentes componentes utilizando una placa de pruebas y realización de pruebas para verificar el correcto funcionamiento del módulo (20% del objetivo).
4. Programación del microcontrolador (30% del objetivo).
5. Diseño de la caja de prototipado del módulo (10% del objetivo).
6. Ensamblaje de los componentes en la caja de prototipado (10% del objetivo).
7. Realización de pruebas finales para verificar el funcionamiento y la precisión del módulo (10% del objetivo).

Pruebas a realizar.

1. **Comprobación de la repetibilidad de las medidas de humedad:** se realizará una serie de 50 mediciones para verificar la repetibilidad y confiabilidad del módulo. Para considerar esta prueba válida, el **error promedio** de todas las mediciones **no deberá superar el 5%**.
2. **Comprobación de la fiabilidad de la visualización de los datos:** Se llevarán a cabo 50 repeticiones para verificar la repetibilidad y confiabilidad del dispositivo de visualización, el cual debe mostrar de forma visual el porcentaje de humedad leído. En este caso, **el error máximo admisible será del 5%**.
3. **Comprobación del funcionamiento de la comunicación Bluetooth:** Se realizarán 20 conjuntos de pruebas para evaluar el comportamiento de la comunicación Bluetooth del módulo de medición de humedad, tanto en la transmisión como en la recepción de datos, a una distancia máxima de 10 metros. Al finalizar las pruebas, **el error máximo** en la comunicación de los datos **no deberá superar el 1%**.

Definición del éxito del objetivo.

El objetivo se considerará alcanzado cuando el módulo cumpla con los siguientes criterios:

- Estar contenido en una caja de prototipo funcional que proteja adecuadamente los componentes electrónicos.
- Ser capaz de medir de forma precisa la humedad del sustrato de una planta, proporcionando mediciones confiables y consistentes.
- Transmitir las mediciones de humedad de manera efectiva a través de la tecnología Bluetooth, asegurando una comunicación estable y confiable con el módulo principal de IoT.
- Cumplir con los criterios de precisión establecidos durante las pruebas.

1.2.3 Objetivo principal 2: módulo de riego.

Diseñar e implementar un módulo de riego automático que se pueda conectar a una fuente de agua y que regule de manera precisa el caudal y la duración del riego. Este módulo debe tener la capacidad de comunicarse por Bluetooth con el módulo principal de IoT, el cual le indicará la cantidad de agua requerida por la planta y, a su vez, el módulo de riego deberá proveer a la planta.

Etapas de este objetivo (en parentesi el peso, en porcentaje, de cada punto en el objetivo total del módulo):

1. Selección de componentes electrónicos adecuados, incluyendo el microcontrolador y la bomba de agua (10% del objetivo).
2. Diseño del esquema electrónico (10% del objetivo).

3. Conexión de los diferentes componentes utilizando una placa de pruebas y realización de pruebas para verificar el correcto funcionamiento del módulo (20% del objetivo).
4. Programación del microcontrolador (30% del objetivo).
5. Diseño de la caja de prototipado del módulo (10% del objetivo).
6. Ensamblaje de los componentes en la caja de prototipado (10% del objetivo).
7. Realización de pruebas finales para verificar el funcionamiento y la precisión del módulo (10% del objetivo).

Pruebas a realizar.

1. **Comprobación de la repetibilidad del suministro de agua:** En esta prueba se realizarán 40 riegos de diferentes duraciones y se tomarán las correspondientes mediciones para verificar la repetibilidad y confiabilidad del módulo de riego. **El error máximo permitido** en estas pruebas **no deberá ser superior al 10%**.
2. **Comprobación del funcionamiento de la comunicación Bluetooth:** Se realizarán 20 conjuntos de pruebas para evaluar el comportamiento de la comunicación Bluetooth del módulo de riego, tanto en la transmisión como en la recepción de datos, a una distancia máxima de 10 metros. Al finalizar las pruebas, **el error máximo** en la comunicación de los datos **no deberá superar el 1%**.

Definición del éxito del objetivo.

El objetivo se considerará alcanzado cuando el módulo cumpla con los siguientes criterios:

- Estar contenido en una caja de prototipo funcional que proteja adecuadamente los componentes electrónicos.
- Ser capaz de suministrar a la planta la cantidad de agua establecida de manera precisa y confiable.
- Comunicar de manera estable y confiable a través de la tecnología Bluetooth con el módulo principal de IoT.
- Cumplir con los criterios de precisión establecidos durante las pruebas.

1.2.4 Objetivo principal 3: módulo principal de IoT

Diseñar e implementar el módulo principal de IoT que se conecte a través de Bluetooth con el módulo de medición de humedad, permitiendo recibir los datos de humedad de la planta. El módulo principal de IoT deberá analizar estos datos y determinar si la planta requiere riego. En caso afirmativo, se establecerá una conexión Bluetooth con el módulo de riego para enviar la cantidad adecuada de agua que el módulo de riego deberá suministrar a la planta.

Este módulo contará con una interfaz de usuario intuitiva y fácil de usar que permitirá visualizar los datos de humedad de las plantas, así como la cantidad de riego que se ha establecido para la planta. A través de esta interfaz, los usuarios podrán calibrar el módulo de lectura de humedad para garantizar mediciones precisas y ajustar los parámetros de riego según las necesidades de cada planta.

Etapas de este objetivo (en parentesi el peso, en porcentaje, de cada punto en el objetivo total del módulo):

1. Selección de componentes electrónicos adecuados, incluyendo el microcontrolador y el display (10% del objetivo).
2. Diseño del esquema electrónico (10% del objetivo).
3. Conexión de los diferentes componentes utilizando una placa de pruebas y realización de pruebas para verificar el correcto funcionamiento del módulo (20% del objetivo).
4. Programación del microcontrolador (30% del objetivo).
5. Diseño de la caja de prototipado del módulo (10% del objetivo).
6. Ensamblaje de los componentes en la caja de prototipado (10% del objetivo).
7. Realización de pruebas finales para verificar el funcionamiento y la precisión del módulo (10% del objetivo).

Pruebas a realizar.

1. **Comprobación del funcionamiento de la comunicación Bluetooth:** se llevarán a cabo 20 conjuntos de pruebas para evaluar el comportamiento de la comunicación Bluetooth del módulo IoT, tanto en la transmisión como en la recepción de datos desde el dispositivo utilizado. **El error máximo permitido en la transmisión de datos no deberá superar el 1%.**
2. **Comprobación de la comunicación con el módulo de medición de humedad:** se realizarán 20 conjuntos de pruebas para verificar la comunicación Bluetooth entre el módulo IoT y el módulo de lectura de humedad. Al finalizar las pruebas, **el error máximo permitido en la transmisión de datos no deberá superar el 1%.**
3. **Comprobación de la comunicación con el módulo de riego:** se llevarán a cabo 20 conjuntos de pruebas para evaluar la comunicación Bluetooth entre el módulo IoT y el módulo de riego. Al finalizar las pruebas, **el error máximo permitido en la transmisión de datos no deberá superar el 1%.**

Definición del éxito del objetivo.

El objetivo se considerará alcanzado cuando el módulo cumpla con los siguientes criterios:

- Estar contenido en una caja de prototipo funcional que proteja adecuadamente los componentes electrónicos.
- Comunicar de manera estable y confiable a través de la tecnología Bluetooth con el módulo de medición de humedad y el módulo de riego.
- Cumplir con los criterios de precisión establecidos durante las pruebas.

1.2.5 Objetivo principal 4: añadir la conexión con un servicio IoT en la nube al módulo principal de IoT

Añadir al módulo principal de IoT la funcionalidad de comunicación con un servicio de IoT en la nube. El módulo deberá ser capaz de establecer una conexión WiFi con el servicio de IoT, realizar el inicio de sesión en el mismo, y enviar los datos los datos actuales de humedad de la planta.

El servicio de IoT en la nube deberá ser capaz de recibir estos, garantizando su integridad y seguridad. Mediante esta integración, los usuarios podrán monitorear de forma remota el estado de las plantas, acceder a datos en tiempo real sobre la humedad y tomar decisiones informadas para su cuidado.

Etapas de este objetivo (en parentesi el peso, en porcentaje, de cada punto en el objetivo total del módulo):

1. Investigación de los distintos servicios de IoT en la nube disponibles actualmente (25% del objetivo).
2. Selección del servicio más adecuado (25% del objetivo): se evaluarán los servicios de IoT en la nube considerando los siguientes parámetros:
 - a. Facilidad de uso.
 - b. Madurez del servicio (fiabilidad).
 - c. Tiempo necesario para la configuración y puesta en marcha del servicio.
3. Programación del microcontrolador para añadir la funcionalidad de comunicación con el servicio IoT en la nube (50% del objetivo):
 - a. Configuración del módulo WiFi para la conexión con el servicio de IoT en la nube seleccionado.
 - b. Programación de los protocolos de comunicación necesarios para la transmisión de los datos de humedad desde el módulo principal de IoT al servicio de IoT en la nube.
 - c. Pruebas de funcionamiento y seguridad para verificar el correcto funcionamiento del sistema de comunicación con el servicio de IoT en la nube.

Pruebas a realizar.

Comprobación del correcto funcionamiento de la comunicación: se llevarán a cabo 20 conjuntos de pruebas para probar el comportamiento de la comunicación entre el módulo IoT y el servicio de IoT en la nube. **El error máximo admisible es**

0%, lo que significa que todos los datos enviados desde el módulo IoT deben llegar correctamente al destino en el servicio de IoT en la nube.

Definición del éxito del objetivo.

El objetivo se considerará alcanzado cuando se cumplan con los siguientes criterios:

- Selección y configuración de un servicio de IoT en la nube apropiado para las necesidades específicas del proyecto.
- Establecimiento de una conexión y comunicación estables y confiables entre el módulo principal de IoT y el servicio de IoT en la nube.
- Cumplimiento de los criterios de precisión establecidos en las pruebas para medir la precisión y exactitud de los datos transmitidos y recibidos entre el módulo principal de IoT y el servicio de IoT en la nube.

1.2.6 Objetivo Secundario 1: Agregar al servicio de IoT en la nube la capacidad de modificar los parámetros de los módulos.

Añadir al servicio de IoT en la nube la capacidad de calibrar el módulo de medición de la humedad y la capacidad de realizar un riego, enviando los comandos adecuados al módulo principal de IoT.

Dado que este objetivo es secundario se entiende que se realizará sólo si todos los objetivos principales serán completados.

Etapas de este objetivo (en parentesi el peso, en porcentaje, de cada punto en el objetivo total del módulo):

1. Investigación sobre cómo enviar los comandos necesarios desde el servicio de IoT en la nube al módulo principal de IoT (20% del objetivo).
2. Modificar el programa del microcontrolador del módulo principal de IoT para añadir las funcionalidades de recepción de los comandos desde el servicio IoT en la nube (80% del objetivo):
 - a. Programar los protocolos de comunicación entre el módulo principal de IoT y el servicio IoT en la nube.
 - b. Programar el servicio de IoT en la nube para implementar los protocolos de escritura hacia el módulo principal de IoT.
 - c. Probar su funcionamiento y seguridad.

Pruebas a realizar.

Comprobación del correcto funcionamiento de la comunicación: se realizarán una serie de 20 pruebas para comprobar el comportamiento de la comunicación entre el módulo IoT y el servicio de IoT en la nube y se cuantificará el número de datos que no han alcanzado el destino. **En este caso el error máximo admisible es 0%**, es decir todos los datos tienen que alcanzar el destino.

Definición del éxito del objetivo.

El objetivo se considerará alcanzado cuando se cumplan los siguientes criterios:

- Se ha seleccionado y configurado el servicio de IoT en la nube con los nuevos requisitos.
- El módulo principal de IoT es capaz de recibir correctamente los comandos desde este servicio.
- Se han alcanzado los criterios de precisión establecidos en las pruebas.

1.2.7 Objetivo Secundario 2: aplicación IoT para smartphones Android

El objetivo es diseñar e implementar una aplicación IoT para smartphones Android que permita visualizar y configurar los parámetros del módulo de lectura de humedad y el módulo de riego. La aplicación debe ser capaz de realizar el inicio de sesión en el servicio de IoT en la nube, obtener todos los datos de humedad y, opcionalmente, permitir la calibración de los parámetros del módulo de medición de humedad y activar el riego para una planta específica.

Es importante tener en cuenta que este objetivo es secundario y solo se llevará a cabo si se han completado todos los objetivos principales. Por lo tanto, contar con una aplicación que proporcione la visualización de datos y la capacidad de enviar comandos al módulo de riego se considera un valor agregado para el sistema en su conjunto, pero no es fundamental, ya que el sistema de riego es automático y autónomo.

Etapas de este objetivo

Este objetivo implica (en parentesi el peso, en porcentaje, de cada punto en el objetivo final):

1. Realizar una investigación sobre las tecnologías software de desarrollo de aplicaciones móviles disponibles (10% del objetivo).
2. Seleccionar la tecnología más adecuada (10% del objetivo), los parámetros a considerar son:
 - a. Fiabilidad de la tecnología.
 - b. Tiempo de desarrollo.
 - c. Madurez de la tecnología en el sector IoT.
 - d. Experiencia previa en el uso de tal tecnología.
3. Definir el alcance de la aplicación (5% del objetivo).
4. Definir los esquemas de funcionamiento (5% del objetivo).
5. Diseñar la UX/UI, el menú y las opciones disponibles para cada dispositivo (10% del objetivo).
6. Programar la lógica (30% del objetivo).
7. Programar la parte visual (20% del objetivo).
8. Probar su funcionamiento y usabilidad (10% del objetivo).

Pruebas a realizar.

Correcto funcionamiento de cada funcionalidad de la aplicación: se realizarán una serie de 20 pruebas para comprobar el correcto comportamiento de la comunicación entre la aplicación y el servicio de IoT en la nube. Al finalizar las pruebas, **el error máximo permitido** en la transmisión de datos **no deberá superar el 5%**.

Dado que este es un objetivo secundario no es fundamental que la aplicación tenga un grado elevado de precisión visto que el entero sistema de riego es automático y autónomo.

Definición del éxito del objetivo.

El objetivo se considerará alcanzado cuando se cumplan los siguientes criterios:

- El prototipo de la aplicación esté instalado en un dispositivo móvil Android.
- La aplicación sea capaz de recibir y visualizar los datos enviados por el módulo IoT.
- Los criterios de precisión establecidos en las pruebas se hayan cumplido.

1.3 Impacto en sostenibilidad, ético-social y de diversidad

La competencia ética y global (CCEG) es la capacidad de actuar de manera honesta, ética, sostenible, socialmente responsable y respetuosa con los derechos humanos y la diversidad, tanto en la práctica académica como en la profesional. La CCEG aborda tres grandes dimensiones: sostenibilidad, comportamiento ético y responsabilidad social.

Aquí seguido se realizará un análisis de la aplicación del CCEG en las distintas etapas de este proyecto.

1.3.1 El CCEG en el momento del diseño del proyecto

El proyecto busca mejorar el cuidado de las plantas ornamentales y la producción de plantas comestibles en entornos urbanos. Se promueve la purificación del aire en interiores [5] y la producción local en huertos urbanos para beneficiar la economía local y reducir la contaminación del transporte. Se considera el uso de baterías recargables y un uso controlado del agua dulce. Sin embargo, se reconoce que la producción de productos electrónicos puede ser contaminante y presenta desafíos en términos de reciclaje [6]. Se destaca la importancia de buscar y comprar productos electrónicos fabricados localmente para promover la sostenibilidad.

En cuanto a la dimensión ética y responsabilidad social, se busca generar impactos positivos, como la creación de empleo y la contribución a los Objetivos de Desarrollo Sostenible. Se espera que el proyecto tenga un efecto beneficioso en la economía local y en la protección del medio ambiente.

En cuanto a la diversidad, género y derechos humanos, el proyecto no tiene un impacto directo en estas áreas. Se tomarán medidas de seguridad para proteger la privacidad, propiedad intelectual y datos personales. Se considerará la mejora de la accesibilidad en futuros desarrollos, aunque el sistema no ha sido diseñado específicamente para personas con discapacidades visuales.

1.3.2 *El CCEG durante el desarrollo del proyecto*

Durante el desarrollo del proyecto, se ha observado un impacto en términos de sostenibilidad. El prototipado de las cajas de los módulos ha generado residuos de material plástico debido a las pruebas realizadas con una impresora 3D. Además, en el proceso de soldadura de los componentes electrónicos se ha utilizado estaño, que produce gases tóxicos al fundirse.

Para mitigar estos impactos, es necesario explorar opciones más sostenibles, como el uso de plásticos biodegradables, el reciclaje de materiales y soldaduras libres de plomo. Tomar conciencia de nuestro impacto ambiental puede impulsar la adopción de prácticas más sostenibles y la búsqueda de alternativas respetuosas con el medioambiente.

1.4 *Enfoque y método seguido*

El proyecto se divide en dos etapas principales: **la etapa conceptual y la etapa realizativa**.

En la **etapa conceptual**, se establecen los objetivos motivacionales y técnicos del proyecto, se realiza la planificación del trabajo, definiendo las diferentes fases, tareas y subtareas que conforman el proyecto en su totalidad, se realiza una investigación exhaustiva del estado del arte tanto en proyectos existentes como en aspectos técnicos, y se evalúa la viabilidad económica y técnica del proyecto.

En la **etapa realizativa**, se procede a la implementación física de todo el sistema. Esta etapa incluye la creación de todos los esquemas técnicos, la conexión y prueba de los componentes electrónicos, la realización de pruebas y, finalmente, el ensamblado final de los prototipos de los módulos en sus respectivas cajas. El desarrollo, modificación y prueba del software desempeñan un papel fundamental en esta fase.

Es importante destacar que, para el **desarrollo del software**, se seguirá un **enfoque clásico o en cascada**, debido a que el alcance del proyecto de software de cada módulo es bastante limitado. Esta metodología consta de diferentes etapas, que incluyen la definición de requisitos (identificación del producto a desarrollar), el análisis y diseño (determinación de cómo debe ser el producto a desarrollar), la implementación (codificación del software), las pruebas (verificación de que el producto cumple con los requisitos) y, por último, el mantenimiento (corrección de errores que puedan surgir).

1.5 Planificación del Trabajo

En este apartado se aborda la organización y distribución del proyecto, estableciendo una planificación detallada que incluye fases, tareas, subtareas, plazos y mecanismos de seguimiento. El objetivo es garantizar una gestión eficiente del tiempo y los recursos, promoviendo un desarrollo ordenado y exitoso.

El enfoque principal es alcanzar los objetivos propuestos de manera estructurada, realizando ajustes y tomando decisiones adecuadas a medida que avanza el proyecto.

1.5.1 Listado de tareas acotadas

A continuación, se describen las diferentes tareas que se llevarán a cabo en el proyecto para cumplir con los objetivos y el plazo de entrega.

Fase 1: Creación del módulo de medición de la humedad

- **Tarea 1:** Seleccionar los componentes y realizar el esquema eléctrico.
 - **Subtarea 1:** Seleccionar los componentes adecuados en base a los objetivos prefijados.
 - **Subtarea 2:** Utilizar un programa en línea para realizar el esquema eléctrico.
- **Tarea 2:** Conectar los componentes y realizar los tests de funcionamiento.
 - **Subtarea 1:** Conectar y probar los componentes del módulo, utilizando una protoboard según el esquema eléctrico.
 - **Subtarea 2:** Realizar los tests necesarios para comprobar el correcto funcionamiento del módulo.
- **Tarea 3:** Desarrollar el software.
 - **Subtarea 1:** Realizar, probar y corregir el programa del módulo.
 - **Subtarea 2:** Añadir y probar la funcionalidad Bluetooth.
- **Tarea 4:** Diseñar la caja contenedora y realizar el ensamblado.
 - **Subtarea 1:** Diseñar la caja contenedora del módulo e imprimirla con una impresora 3D.
 - **Subtarea 2:** Realizar el montaje del módulo (componentes electrónicos) en la caja definitivas.
- **Tarea 5:** Realizar las pruebas finales.
 - **Subtarea 1:** Comprobar con una serie de pruebas la repetibilidad de las medidas de humedad.
 - **Subtarea 2:** Calcular el porcentaje de error y elaborar las conclusiones finales.
- **Tarea 6:** Actualizar la memoria con los datos obtenidos.
 - **Subtarea 1:** Actualizar la memoria con los datos más importantes obtenidos en las tareas de esta fase.

Fase 2: Creación del módulo de riego automático

- **Tarea 1:** Seleccionar los componentes y realizar el esquema eléctrico.
 - **Subtarea 1:** Seleccionar los componentes adecuados en base a los objetivos prefijados.
 - **Subtarea 2:** Utilizar un programa en línea para realizar el esquema eléctrico.
- **Tarea 2:** Conectar los componentes y realizar los tests de funcionamiento.
 - **Subtarea 1:** Conectar y probar los componentes del módulo, utilizando una protoboard según el esquema eléctrico.
 - **Subtarea 2:** Realizar los tests necesarios para comprobar el correcto funcionamiento del módulo.
- **Tarea 3:** Desarrollar el software.
 - **Subtarea 1:** Realizar, probar y corregir el programa del módulo.
 - **Subtarea 2:** Añadir y probar la funcionalidad Bluetooth.
- **Tarea 4:** Diseñar la caja contenedora y realizar el ensamblado.
 - **Subtarea 1:** Diseñar la caja contenedora del módulo e imprimirla con una impresora 3D.
 - **Subtarea 2:** Realizar el montaje del módulo (componentes electrónicos) en la caja definitivas.
- **Tarea 5:** Realizar las pruebas finales.
 - **Subtarea 1:** Comprobar con una serie de pruebas la repetibilidad del riego (cantidad de agua proporcionada).
 - **Subtarea 2:** Calcular el porcentaje de error y elaborar las conclusiones finales.
- **Tarea 6:** Actualizar la memoria con los datos obtenidos.
 - **Subtarea 1:** Actualizar la memoria con los datos más importantes obtenidos en las tareas de esta fase.

Fase 3: Creación del módulo IoT (módulo principal de IoT y servicio en la nube)

- **Tarea 1:** Seleccionar los componentes y realizar el esquema eléctrico.
 - **Subtarea 1:** Seleccionar los componentes adecuados en base a los objetivos prefijados.
 - **Subtarea 2:** Utilizar un programa en línea para realizar el esquema eléctrico.
- **Tarea 2:** Conectar los componentes y realizar los tests de funcionamiento.
 - **Subtarea 1:** Conectar y probar los componentes del módulo, utilizando una protoboard según el esquema eléctrico.
 - **Subtarea 2:** Realizar los tests necesarios para comprobar el correcto funcionamiento del módulo.
- **Tarea 3:** Desarrollar el software.
 - **Subtarea 1:** Realizar, probar y corregir el programa del módulo.
 - **Subtarea 2:** Añadir y probar la funcionalidad Bluetooth.

- **Tarea 4:** Seleccionar el servicio de IoT en la nube y realizar la conexión con el módulo principal de IoT.
 - **Subtarea 1:** Seleccionar el sistema de IoT en la nube, realizar la registración y configuración del sistema.
 - **Subtarea 2:** Probar y corregir el programa del módulo iniciado con la tarea 3, añadiendo la comunicación Wifi y la conexión con el servicio de IoT en la nube.
- **Tarea 5:** Diseñar la caja contenedora y realizar el ensamblado.
 - **Subtarea 1:** Diseñar la caja contenedora del módulo e imprimirla con una impresora 3D.
 - **Subtarea 2:** Realizar el montaje del módulo (componentes electrónicos) en la caja definitivas.
- **Tarea 6:** Realizar las pruebas finales.
 - **Subtarea 1:** Comprobar con una serie de pruebas, la repetibilidad de la comunicación Bluetooth con el módulo de humedad y de riego.
 - **Subtarea 2:** Comprobar con una serie de pruebas la repetibilidad de la comunicación Wifi con el servicio de IoT en la nube.
 - **Subtarea 3:** Calcular el porcentaje de error y elaborar las conclusiones finales.
- **Tarea 7:** Actualizar la memoria con los datos obtenidos.
 - **Subtarea 1:** Actualizar la memoria con los datos más importantes obtenidos en las tareas de esta fase.

Fase 4: Creación de la aplicación IoT para smartphones Android

- **Tarea 1:** Seleccionar la tecnología a utilizar.
 - **Subtarea 1:** Definir los requisitos necesarios (facilidad de uso, librerías existentes para la conexión al servicio de IoT, etc).
 - **Subtarea 2:** Investigar las distintas tecnologías (Android nativo, Xamarin Forms, etc).
 - **Subtarea 3:** Configurar el ambiente de desarrollo (configurar el IDE, instalar las librerías necesarias, etc).
- **Tarea 2:** Desarrollar el proyecto.
 - **Subtarea 1:** Desarrollar la lógica de comunicación al servicio de IoT.
 - **Subtarea 2:** Desarrollar la parte gráfica de visualización de los datos.
- **Tarea 3:** Realizar las pruebas finales.
 - **Subtarea 1:** Realizar las pruebas de lectura y repetibilidad con el servicio de IoT en la nube: comprobar la fiabilidad de la comunicación.
 - **Subtarea 2:** Realizar las pruebas de escritura y repetibilidad con el servicio de IoT en la nube: comprobar la fiabilidad de la comunicación.
- **Tarea 4:** Actualizar la memoria con los datos obtenidos.
 - **Subtarea 1:** Actualizar la memoria con los datos más importantes obtenidos en las tareas de esta fase.

1.5.2 Cronograma o diagrama de Gantt

A continuación se presenta la planificación completa del proyecto a través de un diagrama de Gantt.

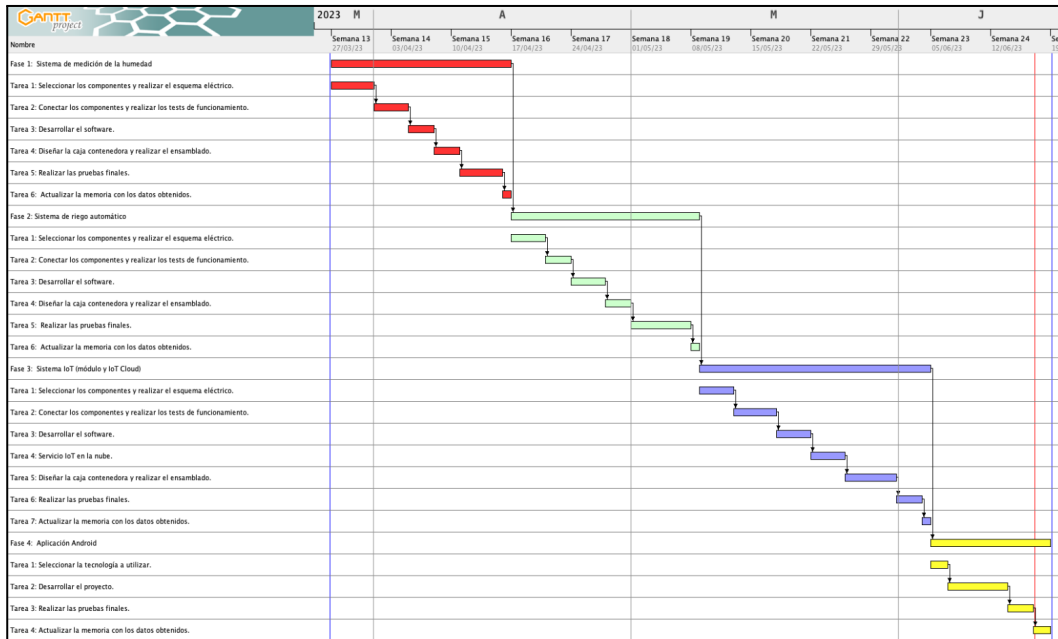


Figura 1: planificación del trabajo a lo largo del semestre.

1.6 Breve resumen de los productos obtenidos

Al finalizar las distintas etapas del proyecto, se obtuvieron los siguientes productos:

- Módulo de medición de la humedad de la tierra de las plantas: dispositivo completamente operativo que permite medir con precisión la humedad del suelo.
- Módulo de riego de plantas: dispositivo completamente operativo que se encarga de activar la bomba de agua para regar las plantas cuando sea necesario.
- Módulo principal IoT: dispositivo completamente operativo que puede conectarse tanto al módulo de medición de humedad como al módulo de riego de plantas, además de establecer una conexión con un broker MQTT. Permite controlar y monitorear el sistema de riego de manera remota.
- Memoria del trabajo: documento detallado que expone todo el proceso de desarrollo del proyecto, incluyendo su argumentación, metodología utilizada, resultados obtenidos y conclusiones.
- Presentación del proyecto: material diseñado para comunicar los aspectos más destacados del proyecto, presentando de manera concisa y visual los resultados, objetivos y características principales.
- Código fuente: incluye el desarrollo del software de los módulos, proporcionando el conjunto de instrucciones y algoritmos utilizados para su funcionamiento.

1.7 Breve descripción de los otros capítulos de la memoria

Los otros capítulos de la memoria son los siguientes:

- **Investigación:** en este capítulo se realiza una investigación exhaustiva del estado del arte relacionado con el proyecto. Se analizan las soluciones existentes, evaluando sus ventajas y desventajas en comparación con la solución propuesta. También se investigan los componentes electrónicos y software disponibles en el mercado que se utilizarán en este proyecto. Se aborda la viabilidad técnica y económica del proyecto y se llega a una conclusión sobre su factibilidad.
- **Diseño:** en este capítulo se describe el diseño del proyecto utilizando diagramas y representaciones gráficas. Se detallan los diferentes módulos y componentes del sistema, mostrando su interconexión y funcionamiento.
- **Desarrollo:** en este capítulo se presenta el desarrollo tanto del hardware como del software del proyecto. Se describe el proceso de implementación de cada módulo, incluyendo los pasos y decisiones clave tomadas durante el desarrollo.

- **Resultados:** en este capítulo se realiza un análisis de los resultados obtenidos. Se evalúa el rendimiento y la eficacia de los módulos y se comparan con los objetivos establecidos inicialmente.
- **Conclusiones y trabajos futuros:** en este capítulo se presentan las conclusiones finales del proyecto. Se evalúa el éxito del proyecto y se destacan los aspectos positivos y las áreas de mejora. Además, se proponen posibles trabajos futuros o mejoras que podrían realizarse en el proyecto para su continuación o expansión.

2 Investigación

2.1 Estado del arte

2.1.1 Soluciones comerciales existen actualmente.

Existen diferentes tipos de sistemas de riego automático en el mercado para plantas de interior, para pequeñas macetas de exterior o huertos urbanos, cada uno con sus ventajas e inconvenientes.

Hay que considerar que para sistemas más amplios como riego urbano de amplias superficies, invernaderos de cultivo o agricultura efectivamente existen sistemas inteligentes IoT, mientras que para sistemas de hogar (interior o exterior) o para huertos urbanos no existen soluciones en el mercado que no sean DIY (Do It Yourself, hágalo usted mismo en inglés) [25][28][29].

Algunos de los sistemas automatizados para hogar o huerto urbano más comunes son:

Riego automático por goteo: consiste en colocar un gotero en cada planta que dispensa el agua directamente a la tierra. Se puede programar el riego mediante un temporizador y regular el caudal de los goteros. Es un sistema versátil y eficiente, pero requiere una instalación previa y una fuente de alimentación eléctrica [26][28][31].

Riego automático por exudación: consiste en usar una goma porosa que permite que la humedad traspase desde el interior y llegue a la planta. Se crea un ambiente permanente de humedad que le sirve a la planta para disponer del agua que necesita en cada momento. Es un sistema sencillo y ecológico, pero puede provocar encharcamientos o fugas si no se controla bien el caudal. Al igual que el sistema de riego por goteo se puede automatizar mediante un temporizador y un regulador del caudal [26][33][34][35].

Riego automático utilizando programadores de electroválvulas: Los programadores de electroválvulas son dispositivos que se usan en sistemas de riego automático de jardines que tienen varias zonas de riego diferentes. Los programadores se encargan de abrir y cerrar las electroválvulas, que son las que regulan el paso del agua a cada zona. Los

programadores de grifo son los más sencillos y manejables para jardines con pocas zonas de riego[28][36][37].

Ventajas e inconvenientes respecto a la propuesta de este proyecto.

Estos sistemas presentan algunas ventajas respecto al proyecto aquí propuesto:

- Todos los sistemas enumerados arriba son más simples a nivel de número de componentes y complejidad. Esto implica que tienen un menor coste de fabricación y mantenimiento, así como una mayor facilidad de uso e instalación. Y sobre todo, al tener menos elementos que pueden fallar, son más robustos y duraderos.
- Las soluciones existentes en el mercado ya tienen una cierta antigüedad y por eso son muy fiables. Esto significa que han sido probadas y validadas por muchos usuarios y que cuentan con el respaldo de marcas reconocidas en el sector del riego automático. También disponen de garantías, servicios técnicos y repuestos en caso de necesidad.
- Fácil de montar. Esto se refiere a que los sistemas existentes no requieren de herramientas especiales ni conocimientos avanzados para su montaje. Basta con seguir las instrucciones del fabricante o del vendedor para conectar los componentes entre sí y al suministro de agua. En algunos casos, incluso se pueden comprar kits o packs que incluyen todo lo necesario para el riego automático.

Mientras que las limitaciones respecto a la propuesta de este proyecto son las siguientes:

- No se adaptan a las condiciones ambientales ni a las características específicas de cada planta. Esto significa que el riego se realiza de forma uniforme y fija, sin tener en cuenta factores como la temperatura, la humedad, la luz o el tipo de planta. Esto puede provocar un riego insuficiente o excesivo, lo que afecta al crecimiento y la salud de las plantas.
- No permiten recoger ni visualizar los datos sobre la humedad del suelo o el consumo de agua. Esto impide conocer el estado real de las plantas y el impacto del riego en el medio ambiente. Sumado a lo anterior, no se puede optimizar el uso del agua ni detectar posibles problemas como fugas o roturas.
- No permiten enviar órdenes al sistema de manera remota ni recibir alertas en caso de anomalías. Esto limita la flexibilidad y el control del usuario sobre el sistema de riego. El usuario no puede modificar los parámetros del riego desde su dispositivo móvil ni recibir notificaciones si hay algún fallo o incidencia en el sistema.

Limitaciones de cara a solucionar el problema planteado en este documento.

La principal limitación de los sistemas actualmente en el mercado para solucionar el problema planteado en este documento es su falta de inteligencia. Es decir, no se puede controlar con exactitud la cantidad de agua que se entrega a cada planta, y no ofrecen la posibilidad de medir la humedad de la tierra, lo que impide saber si la planta tiene demasiada o poca agua. Asimismo, muchas veces, no se puede adaptar el riego a cada tipo de planta, ya que un solo tipo de dispositivo, como por ejemplo las electroválvulas programadas, se utiliza para regar distintos tipos de plantas.

2.1.2 Trabajos académicos: trabajos de Final de Grado

Se llevó a cabo una búsqueda bibliográfica de TFGs publicados relacionados con el tema de estudio, seguido de un resumen.

Agricultura de precisión amb sistemes de sensors agraris [38]:

El proyecto tiene como objetivo diseñar un sistema de agricultura de precisión mediante el uso de sensores (humedad ambiental, humedad del suelo, temperatura, pH, presión) conectados a un controlador Arduino que centralizará y transmitirá los datos a través del módulo Xbee a un servidor central. Luego, el servidor recibirá los datos a través del puerto serial con la ayuda de código de programación Python para almacenarlos en una base de datos. Todo esto en un contexto de sostenibilidad económica y ambiental, con sistemas escalables y exportables a países subdesarrollados.

Regduino [39]:

Regduino es un sistema de riego autónomo que utiliza sensores de humedad de la tierra, temperatura, luz, entre otros, para que una planta regule la cantidad de agua que necesita para sobrevivir. El agua puede ser obtenida de un suministro de casa, un bidón de lluvia o de forma manual. La información proporcionada por los sensores se utiliza para controlar el suministro de agua a la planta, asegurando que solo reciba lo que necesita [39].

Arduino + módulo GSM/GPRS: monitorización, automatización y gestión remota en un viñedo [40]:

El presente proyecto consiste en el diseño de un sistema de monitorización de parámetros ambientales, automatización de procesos y gestión remota mediante la red GSM/GPRS. Se ha contextualizado para ser empleado en un viñedo por lo que se ha presentado la solución al problema con los argumentos pertinentes. Aparte, se ha recurrido al universo Arduino para implementar la solución, debido a su potencia y escalabilidad.

2.1.3 Trabajos académicos: otras fuentes

En Internet existen muchos ejemplos de sistemas de “**smart agriculture**” similares al que se propone en este proyecto. Uno de ellos es un estudio realizado por estudiantes universitarios de la India, donde se desarrolló un sistema de agricultura automatizada utilizando microcontroladores y sensores. El objetivo principal de ese proyecto era mejorar el crecimiento y la producción de los cultivos mediante el uso de tecnología electrónica avanzada.

Para monitorear las condiciones ambientales, como la temperatura, la luz, la humedad y la humedad del suelo, se utilizó tecnología electrónica avanzada y un microcontrolador. El sistema completo se conectó remotamente a través de una línea telefónica GSM para controlarlo. Para transmitir la información de los sensores al microcontrolador central que controla los sistemas automatizados, se utilizó Zigbee, una tecnología inalámbrica de baja potencia y bajo costo que se utiliza para la comunicación entre dispositivos electrónicos.

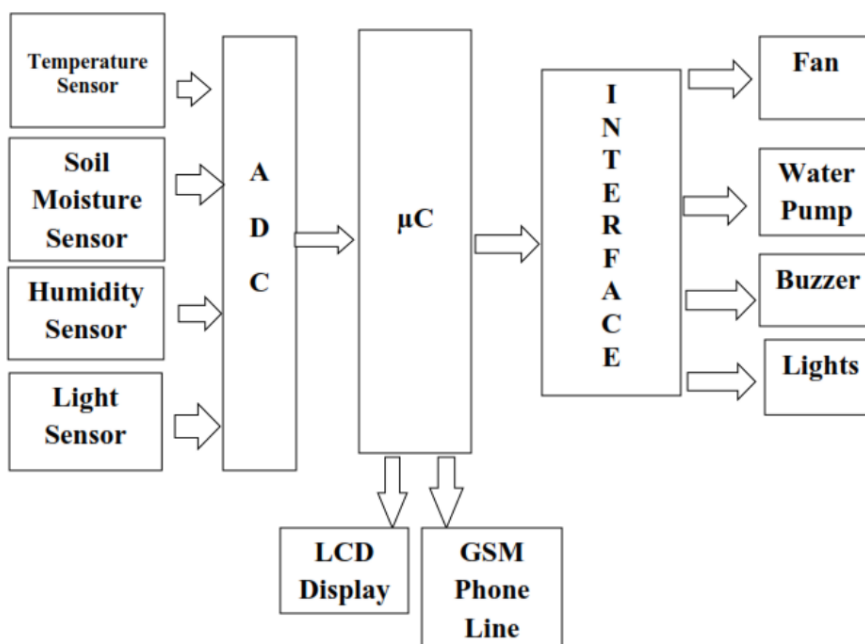


Figura 2: esquema del sistemas de “smart agriculture”.

El sistema de agricultura automatizada utilizando microcontroladores y sensores consta de varias partes. En primer lugar, los sensores detectan la temperatura, la humedad, la intensidad de luz y la humedad del suelo. A continuación, envían la información al convertidor analógico a digital, que la convierte en una señal digital procesable por el microcontrolador. El microcontrolador es la parte central del sistema que procesa la información recibida de los sensores y toma decisiones para controlar los sistemas automatizados, como el riego y la iluminación.

En conclusión, este tipo de sistemas demuestra que la agricultura automatizada utilizando microcontroladores y sensores puede mejorar significativamente el crecimiento y la producción de los cultivos. Al monitorear constantemente las condiciones ambientales,

como la temperatura, la luz, la humedad y la humedad del suelo, se pueden tomar medidas para optimizar el crecimiento de los cultivos [7].

Otro ejemplo es el sistema creado por otros estudiantes del Departamento de Ingeniería Electrónica y Comunicación en SACET, India.

El sistema surge como solución a problemas que suelen aparecer en el campo durante situaciones críticas y que no pueden ser manejados oportunamente, tales como cambios repentinos en el clima, fuertes lluvias, variaciones extremas de temperatura o la presencia de animales dañinos.

Para solucionar este tipo de problemas los autores diseñaron un sistema de monitoreo agrícola basado en IoT que puede ser monitoreado desde cualquier lugar y es fácilmente accesible a través de alertas SMS. Este sistema utiliza sensores de temperatura y humedad colocados en lugares adecuados para monitorear los cultivos. Se desarrolló un algoritmo con valores umbral de temperatura y humedad del suelo que se pueden programar en una puerta de enlace basada en microcontrolador para controlar la cantidad de agua. El sistema puede ser alimentado por paneles fotovoltaicos y puede tener un enlace de comunicación dúplex basado en una interfaz celular a través de una página web. El sistema también envía alertas SMS a los agricultores para informarles sobre cualquier cambio importante en las condiciones del campo.

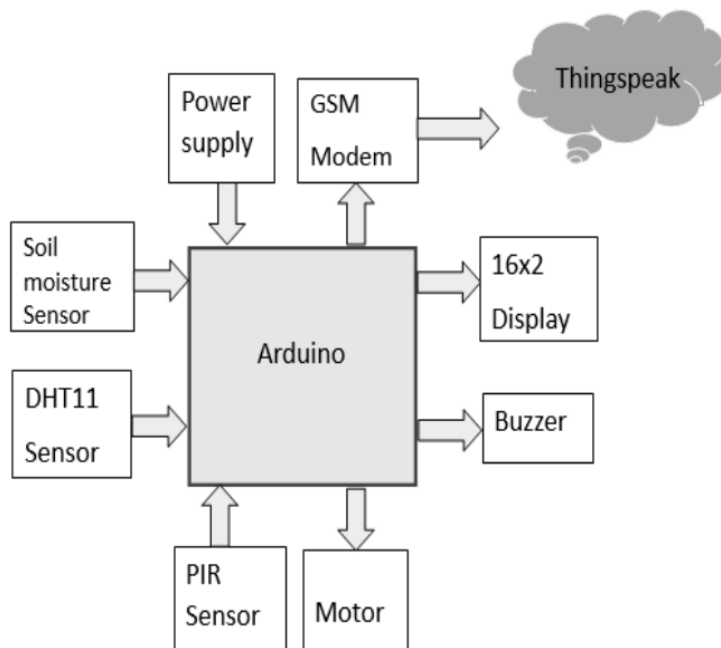


Figura 3: sistema creado por el Departamento de Ingeniería Electrónica y Comunicación en SACET, India.

El sistema utiliza varios sensores para recopilar datos sobre las condiciones del campo. Estos sensores incluyen un sensor de temperatura, un sensor de humedad, un sensor de nivel de agua, un sensor PIR (sensor infrarrojo pasivo) y un módulo GSM. El sensor de temperatura y el sensor de humedad se colocan en lugares adecuados para monitorear los cultivos, mientras que el sensor de nivel de agua se utiliza para verificar el nivel del agua en el campo. El sensor PIR se utiliza para detectar la presencia de animales en el campo dado que es capaz de detectar la radiación infrarroja emitida por los cuerpos calientes y enviar una señal al microcontrolador para que se active una alerta SMS.

Del mismo modo, el módulo GSM se utiliza para enviar alertas SMS a los agricultores sobre cualquier cambio importante en las condiciones del campo.

El sistema utiliza el microcontrolador Arduino Uno que utiliza el chip ATmega328p, un microcontrolador de 8 bits de alta velocidad y bajo consumo de energía. La placa está equipada con un conjunto de 14 pines digitales y 8 pines analógicos de entrada/salida que se han utilizado para conectar los diferentes sensores y periféricos.

A nivel de comunicación, además de los mensajes SMS para las alertas, el sistema utiliza como servicio en la nube ThingSpeak, una plataforma de análisis de IoT que permite agregar, visualizar y analizar flujos de datos en vivo en la nube. De este modo los usuarios pueden ver las visualizaciones instantáneas de datos en vivo, recibir alertas y enviar comandos directamente desde sus dispositivos.

En conclusión, los autores han verificado que el sistema propuesto es muy útil para la gestión durante situaciones críticas (tales como cambios repentinos en el clima, fuertes lluvias, variaciones extremas de temperatura o la presencia de animales dañinos) dando a los agricultores una herramienta para poder gestionar estos eventos de manera oportuna.[8]

Ventajas e inconvenientes de los trabajos académicos respecto a la propuesta de este proyecto.

Los sistemas estudiados presentan algunas ventajas respecto al proyecto aquí propuesto:

- Algunos proyectos incorporan múltiples sensores adicionales para adquirir datos, como la medición de luz, temperatura y humedad ambiental, además de la humedad del suelo.
- Algunos proyectos incluyen protocolos de comunicación como XBee o Zigbee, diseñados especialmente para la interconexión entre dispositivos remotos.
- Algunos proyectos cuentan con sistemas de base de datos para almacenar la información recolectada.
- Algunos proyectos utilizan comunicación GSM/GPRS, lo que permite su instalación en zonas remotas.

Mientras que las limitaciones respecto a la propuesta de este proyecto son las siguientes:

- Algunos proyectos incluyen elementos complejos, como módulos de comunicación XBee o servidores con base de datos, lo que aumenta la probabilidad de fallos en el sistema.
- Algunos sistemas combinan diferentes IDEs, bases de datos (SQL, MySQL, etc.) y lenguajes de programación (C++, Python, Javascript, etc.), lo cual complica la implementación.
- Algunos proyectos utilizan cables en lugar de BLE para la comunicación entre los módulos en la misma área, lo cual puede resultar problemático si el número de dispositivos aumenta.
- La mayoría de los proyectos utilizan Arduino Uno, lo que implica que el procesamiento de datos sea más lento en comparación con el ESP32, y también que tenga menos capacidad de almacenamiento, lo que requiere la incorporación de módulos externos para la comunicación.

2.1.4 Componentes electrónicos

Microcontroladores

El cerebro de cada módulo del sistema propuesto en este proyecto es el microcontrolador, que recibe las señales de los sensores, las procesa, toma decisiones y se comunica con otros sistemas. Los microcontroladores son dispositivos electrónicos programables capaces de ejecutar instrucciones de forma autónoma.

En este proyecto se utilizará el microcontrolador **Esp32**, conocido por sus características y funcionalidades que lo hacen adecuado para el sistema de riego automático IoT.

Esp32

El Esp32 WROOM-32 es un módulo Wi-Fi producido por Espressif Systems [10] y diseñado especialmente para proyectos de IoT. Este módulo ofrece numerosos GPIO y soporte para varios protocolos de comunicación. Su característica distintiva es la conectividad inalámbrica incorporada, lo que permite controlar y monitorear dispositivos de forma remota a través de Wi-Fi y Bluetooth. Es un sistema en chip (SoC) que combina un microcontrolador Tensilica de 32 bits con interfaces periféricas, amplificadores de potencia y gestión de energía. Ofrece capacidades de Wi-Fi de 2.4 GHz, Bluetooth LE y clásico, y una variedad de interfaces de E/S. El procesador principal, LX6, funciona a una frecuencia de 80-240 MHz y tiene memoria ROM y SRAM integradas, además de soportar memoria flash externa [11].

Las placas que incorporan el Esp32 utilizadas en este proyecto son las fabricadas por Az-Delivery, empresa alemana especializada en la fabricación de componentes electrónicos [12].

La primera es la **Az-Delivery ESP32 Dev Kit C V2**, una placa de desarrollo creada en torno al chip ESP32 WROOM 32, que contiene un regulador de voltaje y un circuito programador USB para el chip ESP32, y otras características [13].

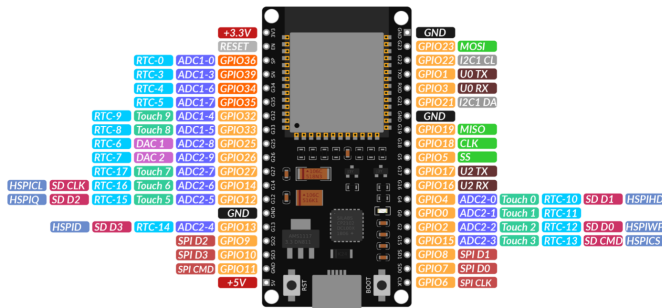


Figura 4: pinout Esp32 Dev Kit C V2.

El **D1 Mini ESP32 de AZ-Delivery** es una placa de desarrollo compacta basada en el chip Esp32 WROOM 32. Ofrece características similares al Az-Delivery ESP32 Dev Kit C V2, pero en un tamaño más reducido. Esta placa está diseñada para ser utilizada en una protoboard y cuenta con un regulador de voltaje y un circuito programador USB para el chip ESP32. Los pines de entrada/salida operan a 3,3V y la comunicación USB a serie se realiza a través del chip CP2104 [14].

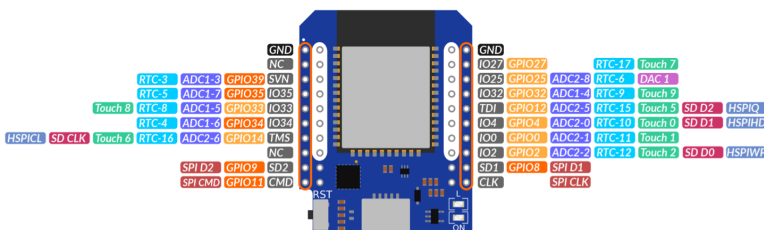


Figura 5: pinout D1 Mini ESP32.

Sensores

Los sensores son dispositivos que detectan estímulos del entorno y los convierten en señales eléctricas. En este proyecto se utilizará el sensor de humedad del suelo, que utiliza la capacitancia para medir la permitividad dieléctrica del suelo, relacionada con su contenido de agua. El sensor genera un voltaje proporcional a la humedad del suelo y realiza un promedio a lo largo de su longitud [16].

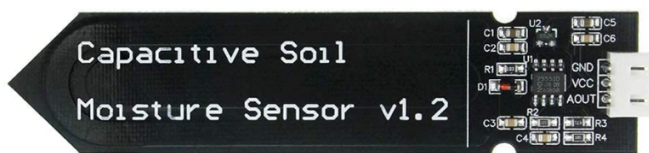


Figura 6: sensor de humedad del suelo.

Actuadores

Los actuadores son componentes que convierten señales eléctricas en movimiento mecánico. Pueden ser motores eléctricos, solenoides, actuadores lineales, piezoeléctricos y otros tipos de transductores. Su función es controlar y manipular sistemas mecánicos mediante señales eléctricas.

En este proyecto se utilizarán distintos actuadores, uno de ellos es el **servomotor** que se empleará en el módulo de medición de humedad del suelo. El servomotor es un tipo de motor eléctrico que se caracteriza por su capacidad de definir un ángulo de giro preciso y controlar la velocidad. Está compuesto por un motor DC, engranajes reductores de velocidad y un circuito de control de posición [17]. El servomotor utilizado en este proyecto tiene un ángulo de giro de 180 grados y se conecta a través de tres cables: tierra, alimentación (5V) y un pin PWM. La tensión de alimentación se encuentra entre 4.8V y 7.2V y puede ser controlada mediante la placa Esp32.



Figura 7: servomotor.

El **relé** es otro de los actuadores utilizados en el proyecto. Se puede definir como un interruptor electromagnético que permite el paso de una corriente de mayor magnitud, en este caso, se utiliza para controlar la alimentación de las Mini Bombas de Agua. El relé está compuesto por una bobina de alambre que actúa como un electroimán temporal cuando una corriente eléctrica atraviesa por ella. [18].

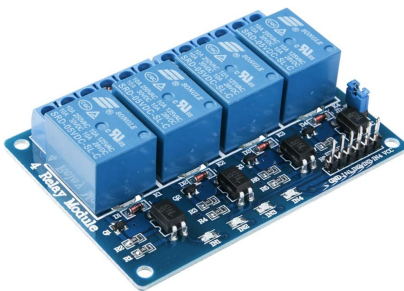


Figura 8: relés de 4 canales a DC 5V..

En el proyecto se emplea un tercer actuador, que es una mini bomba de agua. Esta bomba contiene un motor Brushless de 12V DC, el cual funciona con corriente continua y no utiliza escobillas, lo que le proporciona mayor eficiencia y menor necesidad de mantenimiento. La energía eléctrica se puede transformar en energía mecánica utilizando un motor. Luego, para convertir la energía mecánica en trabajo realizado, se utiliza una bomba [19]. En este caso, se emplea en la bomba para suministrar agua al suelo.



Figura 9: mini bomba de agua

Pantalla OLED LCD

En este proyecto se empleará la pantalla SSH1106 OLED LCD de 1.3 pulgadas. Con una resolución de 128x64 píxeles, ofrece una calidad de imagen nítida y un alto contraste gracias a su tecnología OLED. Su comunicación a través de la interfaz I2C/IIC (Inter-Integrated Circuit) simplifica su integración en proyectos electrónicos y sistemas embebidos. Además, destaca por su eficiencia energética, lo que contribuye a un bajo consumo de energía.

I2C (Inter-Integrated Circuit) es un protocolo de comunicación serie que permite la interconexión de múltiples dispositivos en un sistema electrónico. Se caracteriza por su simplicidad y eficiencia, utilizando solo dos cables para la transmisión de datos: uno para la señal de reloj (SCL) y otro para la transferencia de datos (SDA). El protocolo I2C facilita la comunicación entre componentes electrónicos, como sensores, pantallas y microcontroladores, permitiendo el intercambio de información de manera rápida y confiable.



Figura 10: pantalla OLED LCD de 1.3 pulgadas.

Bluetooth Low Energy (BLE)

La comunicación entre los diferentes módulos del sistema se llevará a cabo a través de Bluetooth Low Energy (BLE) que es una tecnología de comunicación inalámbrica de corto alcance y bajo consumo de energía. Esta tecnología, también conocida como Bluetooth Inteligente, fue introducida como parte de la Especificación Central de Bluetooth versión 4.0 y fue adoptada en 2010.

El propósito principal de esta tecnología es permitir el uso de transceptores con menor consumo de energía, baja complejidad y menor costo que los posibles con la tecnología

clásica de Bluetooth. Aunque BLE tiene un rango de transmisión relativamente corto (entre 10 y 50 metros), los dispositivos que utilizan BLE pueden funcionar durante años gracias al modo de espera de ultra bajo consumo de energía. De hecho, según un estudio realizado en un módulo Bluetooth Low Energy System-on-a-Chip con una batería estándar de 230 mAh, la batería puede durar hasta 14 años en condiciones ideales[20].

La velocidad de transferencia máxima teórica en BLE 4.0 es de 1 Mbps (megabit por segundo), mientras que en BLE 5.0 se incrementa a 2 Mbps[42].

El microcontrolador ESP32 incluye la capacidad de comunicación Bluetooth de baja energía (BLE) lo que le brinda la capacidad de transmitir datos de manera inalámbrica y establecer conexiones de manera rápida y confiable.

GATT

El perfil de atributo genérico (GATT, por sus siglas en inglés) es un componente clave del protocolo de atributo (ATT) en dispositivos Bluetooth de baja energía (BLE). En BLE, el protocolo ATT es un protocolo sin estado Cliente/Servidor basado en atributos, donde los dispositivos pueden asumir cualquier rol (Periférico o Central) independientemente de su rol BLE.

El GATT profile es una estructura jerárquica de atributos que permite la transferencia de información entre un cliente y un servidor. Dentro de un perfil GATT, los atributos pueden ser servicios o características y se identifican mediante un identificador único universal (UUID). Las características, además de su UUID, están compuestas por un identificador de atributo, un conjunto de propiedades y un valor. El identificador de atributo especifica la posición de la característica en el perfil, mientras que el valor contiene los datos reales de la característica. Las propiedades son metadatos que especifican qué operaciones ATT (lectura, escritura, etc.) se pueden ejecutar en cada atributo en particular y con qué requisitos de seguridad específicos (encriptación, autenticación) [42].

Software

El software es esencial en los proyectos de IoT para gestionar y procesar datos de manera efectiva. Permite la comunicación constante, la automatización de procesos y la toma de decisiones en tiempo real.

Ambiente de desarrollo

En este proyecto, se utiliza **Visual Studio Code (VSC)** como herramienta de desarrollo de software. VSC es un editor de código fuente ampliamente utilizado en proyectos de IoT debido a su interfaz intuitiva y a las extensiones disponibles. Una extensión popular es PlatformIO, una plataforma de desarrollo de código abierto que facilita la programación de placas y microcontroladores como el ESP32. PlatformIO ofrece características como la gestión de librerías y la depuración de código, lo que lo hace muy útil en el desarrollo de proyectos IoT.

Diseño 3D

Para la creación de las cajas de los módulos se utilizarán programas de generación 3D como Autodesk Tinkercad y Fusion 360 para diseñar cajas personalizadas que se adapten a las necesidades del proyecto. Tinkercad es un software en línea fácil de usar que ofrece una variedad de herramientas para la creación de modelos precisos. Es especialmente útil para diseñadores sin experiencia previa en diseño 3D, ya que su interfaz intuitiva permite crear prototipos rápidamente. Por otro lado, Fusion 360 es una herramienta más avanzada que ofrece capacidades adicionales para el diseño, simulación y fabricación de productos. Ambos programas son útiles en la creación de cajas para dispositivos IoT.

2.2 Aspectos de ciberseguridad a tener en cuenta

El rápido crecimiento del Internet de las cosas (IoT) plantea desafíos de seguridad. La arquitectura del IoT consta de tres capas: percepción, red y aplicación, cada una con sus propias funciones y dispositivos. La capa de percepción es la más vulnerable, ya que recopila datos a través de sensores y dispositivos físicos. La capa de red utiliza diferentes protocolos con características de seguridad específicas. La capa de aplicación procesa los datos y utiliza plataformas IoT en la nube, como Google Cloud IoT y Amazon Web Services IoT. También existen opciones de código abierto, como Thingsboard y Mozilla WebThings, que permiten personalizar el sistema. La seguridad en el IoT es crucial para evitar riesgos y proteger la información [9].

2.2.1 Problemas de Seguridad en IoT

El Internet de las cosas (IoT) presenta varios problemas de seguridad, incluyendo malware y vulnerabilidades. Para garantizar la ciberseguridad en IoT, se recomiendan diversas estrategias y buenas prácticas. Estas incluyen la autenticación y autorización adecuadas, mantener actualizados los dispositivos y software con parches de seguridad, monitorear y analizar datos para identificar amenazas, utilizar encriptación de datos, establecer políticas de control de acceso, evaluar cuidadosamente a los proveedores de dispositivos y servicios, contar con un plan de gestión de incidentes y realizar pruebas de penetración regularmente. Estas medidas ayudan a proteger la privacidad y la integridad de los sistemas IoT y a prevenir posibles ataques y violaciones de seguridad [15].

2.3 Protocolo de comunicación para servicios de IoT en la nube

Una de las claves para el éxito de un sistema IoT es la selección de un protocolo apropiado para la comunicación. Después de una investigación exhaustiva para este proyecto, se ha decidido utilizar el protocolo MQTT (Message Queue Telemetry Transport).

2.3.1 MQTT (Message Queue Telemetry Transport)

MQTT es un protocolo de conectividad máquina a máquina (M2M) / Internet de las cosas (IoT) diseñado para ser extremadamente liviano y eficiente en redes con recursos limitados. Es un protocolo de mensajería basado en publicación / suscripción que utiliza una arquitectura cliente-servidor. Los clientes se suscriben a temas específicos y reciben mensajes publicados por otros clientes o servidores. MQTT es adecuado para aplicaciones que requieren baja latencia, ancho de banda limitado y baja sobrecarga de red. MQTT admite diferentes niveles de calidad de servicio (QoS) para garantizar la entrega confiable de mensajes [21][22].

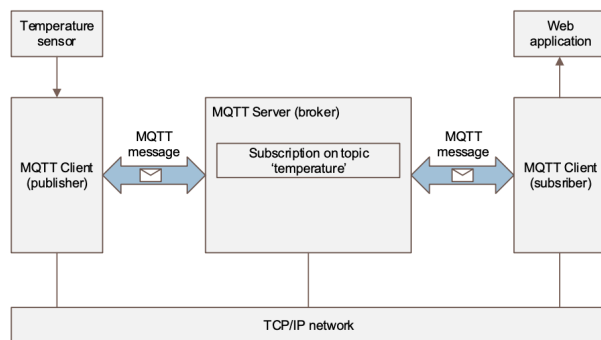


Figura 11: MQTT: esquema funcional.

Un **cliente MQTT** es una aplicación que se conecta a un **broker MQTT** para enviar y recibir mensajes. Los clientes MQTT pueden ser desde pequeños dispositivos como Arduino o Raspberry Pi hasta potentes ordenadores que ejecutan programas gráficos. La implementación del protocolo MQTT en el lado del cliente es simple y directa, y existen bibliotecas de clientes disponibles para diferentes lenguajes de programación y plataformas. Para recibir datos a través de MQTT, un cliente debe suscribirse a un tema específico en el broker [23].

Un **broker MQTT** es un servidor que actúa como intermediario entre los clientes MQTT. Los clientes publican mensajes en el broker, y el broker se encarga de enrutar los mensajes a los clientes suscritos a los temas correspondientes. El broker también puede almacenar mensajes para los clientes que no están disponibles en el momento de la publicación. Los brokers MQTT pueden ser escalados horizontalmente para manejar grandes volúmenes de tráfico y pueden ser configurados para garantizar diferentes

niveles de calidad de servicio (QoS) para la entrega confiable de mensajes. Además, los brokers MQTT pueden ser configurados con diferentes políticas de seguridad para garantizar la privacidad y la autenticación de los clientes [23].

2.3.2 Selección de un broker MQTT como servicio IoT en la nube

¿Porqué se ha elegido un broker MQTT como servicio IoT en la nube para este proyecto?

Como mencionado anteriormente el protocolo MQTT es ampliamente utilizado en aplicaciones de IoT debido a su eficiencia, bajo consumo de ancho de banda y capacidad de manejar una gran cantidad de dispositivos conectados.

Al utilizar un broker MQTT en la nube, se puede lograr una mayor flexibilidad y escalabilidad en el proyecto. Esto significa que el sistema de riego puede crecer y adaptarse fácilmente a medida que se agregan más sensores, actuadores y funcionalidades en el futuro. El broker MQTT en la nube proporciona una infraestructura confiable y segura para gestionar la comunicación entre los dispositivos y la plataforma de control.

En cuanto a la elección específica de **MyQttHub.com** como broker MQTT, esta decisión se tomó después de una exhaustiva investigación. Se evaluaron diferentes proveedores de servicios MQTT en la nube y se tuvieron en cuenta factores como la confiabilidad, seguridad, facilidad de uso y costo (en este caso el servicio es gratuito). MyQttHub.com se destacó como una opción confiable y bien establecida en el mercado, ofreciendo características avanzadas y una interfaz amigable para la gestión de dispositivos y la configuración de canales de comunicación.

La elección de MyQttHub.com también se basó en las buenas referencias y comentarios positivos de otros usuarios de IoT que han utilizado sus servicios. La plataforma ofrece un alto nivel de seguridad en la comunicación y la posibilidad de implementar diferentes niveles de autenticación y control de acceso, lo que es fundamental para proteger los datos y garantizar la integridad del sistema de riego automatizado.

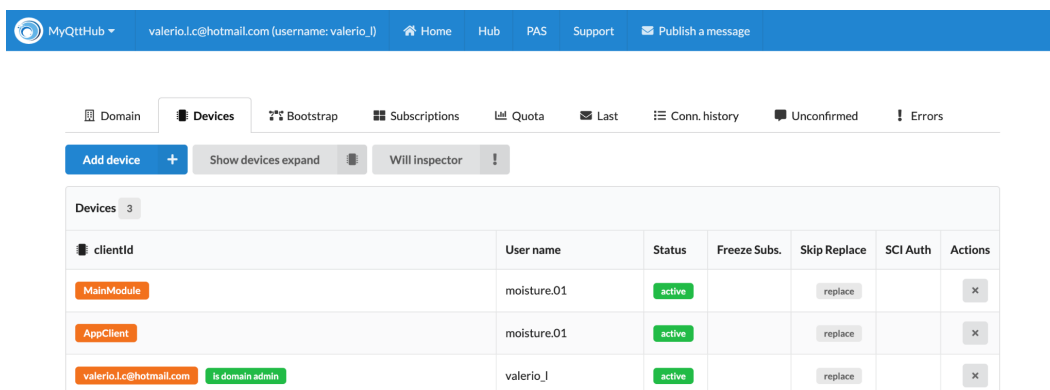


Figura 12: MyQttHub.com: página de gestión de dispositivos.

2.4 Estudio de viabilidad

2.4.1 Viabilidad técnica

Como se ha podido comprobar existe una amplia referencia en proyectos comerciales y trabajos académicos relacionados con el tema. Además, hay una gran variedad de componentes electrónicos disponibles en el mercado y literatura especializada que se puede utilizar en el proyecto.

A continuación, se analizarán los elementos de software y hardware utilizados en este proyecto.

Objetivo principal 1, sistema de medición de la humedad, hardware:

- **Sensor capacitivo de humedad:** es el componente que medirá la humedad del sustrato de la planta.
- **Microcontrolador ESP32 Mini (con Bluetooth BLE incorporado):** se ocupará de toda la lógica del módulo, recibirá los datos del sensor capacitivo e interpretará esos datos, a su vez se ocupará de la comunicación Bluetooth.
- **Servomotor:** este se empleará como indicador visual de la humedad leída.
- **Batería 9V recargable:** se ocupará de la alimentación del módulo.
- **Módulo de regulador de 5V:** dado que el Esp32 tiene que ser alimentado con 5V, se necesita este módulo para poder regular el voltaje que llega de la batería.
- **Otros elementos:** condensador para eliminar el ruido eléctrico del servomotor.

Objetivo principal 2, sistema de riego automático, hardware

- **Microcontrolador ESP32 (con Bluetooth BLE incorporado):** se ocupará de toda la lógica del módulo, la gestión de la activación de los relés y de la cantidad de agua que tienen que suministrar, a su vez se ocupará de la comunicación Bluetooth.
- **Powerbank solar 5V:** se ocupará de la alimentación del módulo y puede ser recargada a través de un panel solar incorporado.
- **Relés de 4 Canales a DC 5V:** estos dispositivos electromecánicos permiten el control de las bombas de agua mediante la apertura o cierre de contactos eléctricos.
- **Bomba de Agua DC 3V:** será utilizada para proveer el agua necesaria a la planta.

Objetivo principal 3 y 4, módulo principal de IoT, hardware:

- **Microcontrolador Esp32 (con Bluetooth BLE y Wifi incorporados):** el componente que se encargará de gestionar la lógica y la comunicación con los módulos (a través del Bluetooth) y con el servicio IoT en la nube (a través del WiFi).
- **Powerbank solar 5V:** se ocupará de la alimentación del módulo y puede ser recargada a través de un panel solar incorporado.

- **Display I2C LCD OLED, 1.3 pulgadas, 128x64 píxeles:** para la visualización de los datos y la navegación en los distintos menús.
- **Otros elementos:** pulsadores y resistencias de 10K utilizados para la navegación en los menús.

Objetivo secundario 1, añadir funcionalidades al servicio IoT para la interacción con los distintos sistemas:

Este objetivo no requiere ningún componente hardware adicional. Se utilizará el software de desarrollo elegido para poder añadir las funcionalidades.

Objetivo secundario 2, aplicación IoT para smartphones Android

Este objetivo no requiere ningún componente hardware adicional. Se utilizará el software de desarrollo elegido para poder desarrollar la aplicación Android.

Software de desarrollo

Como ya se ha considerado en la sección de estado del arte, el software de desarrollo que se ha seleccionado para la programación de los microcontroladores Esp32 y Esp32 Mini es Microsoft Visual Studio Code con framework Arduino y plataforma PlatformIO.

2.4.2 Viabilidad económica

La siguiente tabla resume los costes del proyecto.

	Precio
Hardware (microcontroladores, pantalla, servomotor, etc)	138,92 €
Software (Visual Studio Code, Tinkercad, librerías, etc)	0 €
Horas totales de investigación, diseño e implementación*	1224,00 €
Total:	1362,92 €

Tabla 1: viabilidad económica: cálculo total.

*Precio medio de un desarrollador de software/hardware IoT junior dada mi experiencia en el sector IoT [41].

Un resumen más detallado de los costes se encuentra en el anexo 3.

Conclusión.

Es importante destacar que, según se pudo observar en el estado del arte previamente presentado, existen soluciones similares a la propuesta en este proyecto, lo que indica su

viabilidad. Además, los avances en la tecnología de IoT y los microcontroladores han facilitado y abaratado la implementación de sistemas de este tipo.

Por lo tanto, el desarrollo del proyecto es factible y está respaldado por la tecnología actual. No obstante, es importante tener en cuenta que el costo principal se relaciona con las horas dedicadas a la investigación, diseño e implementación del proyecto en su totalidad.

3 Diseño

3.1 Diagrama de bloques general

El siguiente diagrama ilustra el sistema completo de riego automático. El componente central es el módulo principal de IoT, el cual se conecta mediante BLE al módulo de riego y recopila los datos de la última medición de humedad. Si la medición de humedad está por debajo del umbral preestablecido, el módulo de IoT envía al módulo de riego una orden de suministrar el agua a la planta.

Además, cada vez que se recibe una medición de humedad, se envía el dato al Broker MQTT, encargado de distribuirlo a todos los clientes MQTT que se han suscrito al servicio.

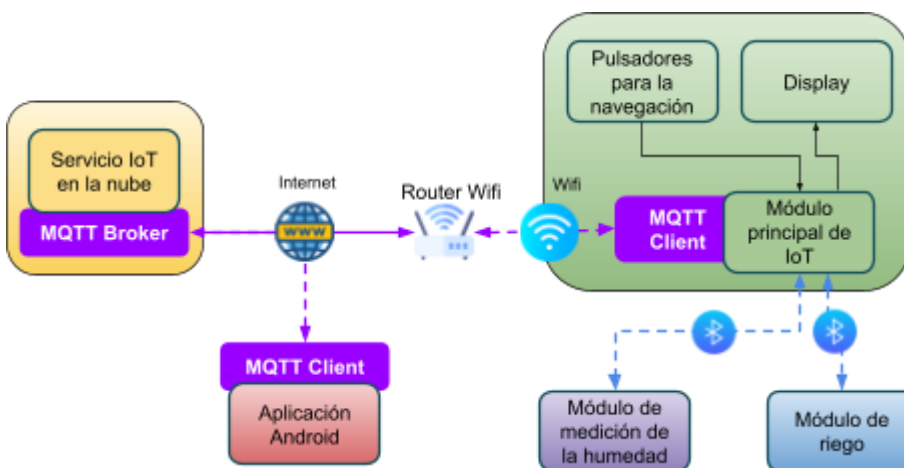


Figura 13: Diagrama de bloques general.

3.1.1 Diagrama de bloques del módulo de medición de la humedad

El siguiente diagrama muestra las conexiones de los componentes del módulo de medición de humedad, cuya función principal es medir la humedad del suelo de la planta, transformar el valor en porcentaje, visualizarlo a través del servomotor y comunicarlo al módulo principal IoT.

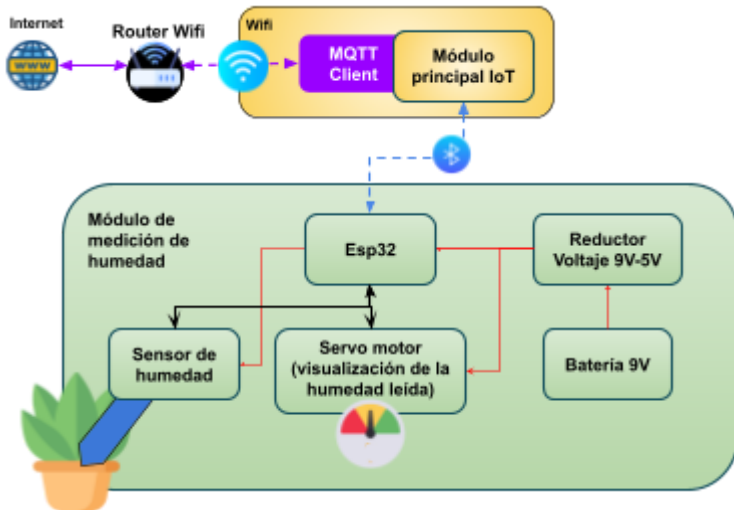


Figura 14: módulo de medición de la humedad: diagrama de bloques.

3.1.2 Diagrama de bloques del módulo de riego

El siguiente diagrama muestra las conexiones de los componentes del módulo de riego. El microcontrolador Esp32 es el componente principal encargado de recibir la orden de riego del módulo principal de IoT a través del BLE. Una vez recibida la orden, el microcontrolador activa el relé, que a su vez activa la mini bomba de agua para regar la planta. Todos los componentes del módulo son alimentados por un Powerbank que proporciona una alimentación de 5V.

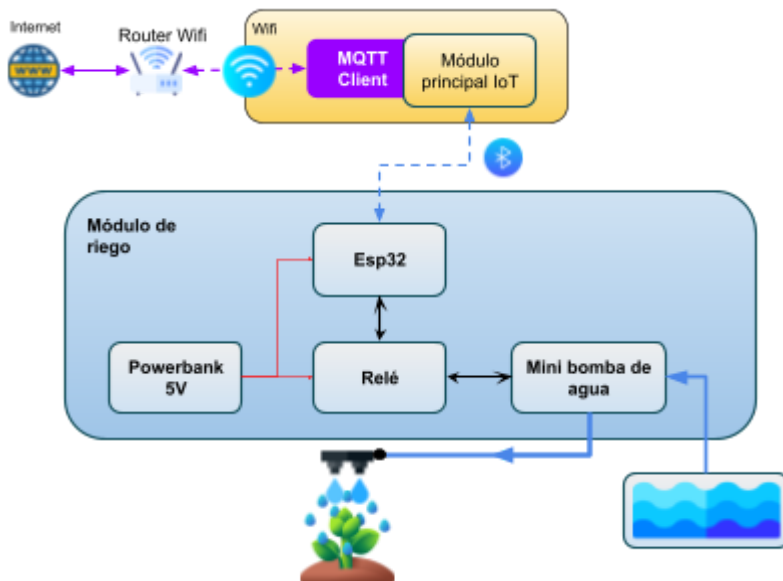


Figura 15: módulo de riego: diagrama de bloques.

4 Desarrollo

4.1 Módulo de medición de humedad

4.1.1 Diseño de conexión

El circuito de este módulo está compuesto por un microcontrolador Esp32 Mini, que contiene la lógica de funcionamiento y la comunicación BLE. También incluye un sensor de medición de humedad para medir la humedad del suelo de la planta, y un servomotor que indica la humedad leída según su ángulo.

Para asegurar el correcto funcionamiento del microcontrolador al iniciar el movimiento del servomotor, se ha agregado un condensador de 100 μ F.

La alimentación del módulo se realiza mediante una batería de 9V. Sin embargo, dado que el microcontrolador requiere una alimentación de 5V, se utiliza un circuito reductor de tensión.

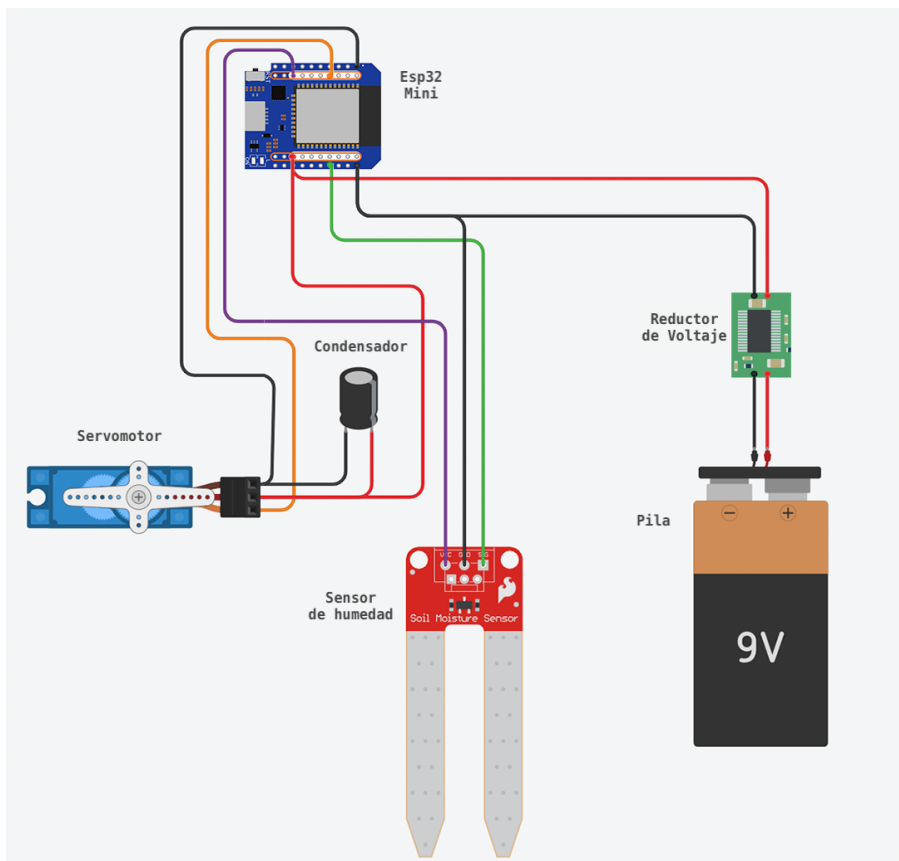


Figura 16: conexión de los componentes utilizados en el módulo de medición de humedad.

4.1.2 Esquema eléctrico

El siguiente esquema eléctrico indica las conexiones entre los componentes del sistema de humedad.

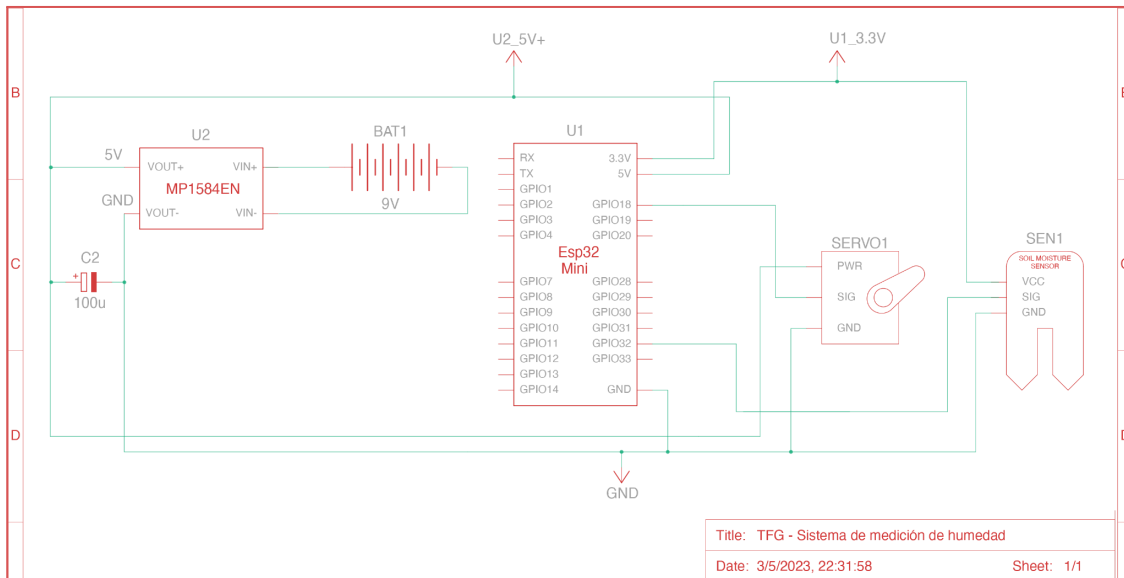


Figura 17: módulo de medición de humedad: esquema eléctrico.

El elenco de los componentes está elencado en la siguiente tabla:

Código	Componente	Código	Componente
U1	Esp32 Mini	SERVO1	Servomotor
U2	MP1584EN Reductor de Voltaje	SEN1	Sensor de humedad
BAT1	Pila 9V		

Tabla 2: módulo de medición de humedad: elenco de componentes.

4.1.3 Prototipado

Como se ha explicado anteriormente, el objetivo de este primer módulo es diseñar e implementar un sistema de medición continua de la humedad del sustrato en las plantas. Para ello, se ha utilizado un sensor de humedad de tipo capacitivo que se encarga de medir la humedad del suelo. Este sensor se alimenta a 3,3V y proporciona una salida en voltios proporcional a la humedad medida.

El sensor de humedad está conectado a una de las entradas analógicas del Esp32 Mini, que es el componente principal del sistema. El Esp32 Mini convierte la señal analógica del sensor a señal digital con una resolución de 12 bits.

Dado que el sistema es portátil, se ha optado por utilizar un servomotor para indicar visualmente el porcentaje de humedad, en lugar de una pantalla que requeriría

alimentación constante y consumiría más energía de la batería. El Esp32 Mini envía una señal al servomotor indicando el ángulo de movimiento correspondiente a la lectura de humedad. Entre una medición y otra, el motor queda en estado de reposo.

El sistema completo se alimenta con una batería de 9V. El módulo Esp32 Mini de AZDelibery no tiene un convertor interno de voltaje y solo acepta 5V como alimentación. Por lo tanto, se requiere el módulo regulador de Voltaje de Fuente de alimentación MP1584EN para regular la alimentación.

Por último, el mismo módulo del Esp32 Mini, incorpora la comunicación BLE (Bluetooth Low Energy) para la comunicación con el módulo principal de IoT y el envío y recepción de datos.

El primer paso para poner en marcha el sistema es conectar todos los componentes en una Breadboard (placa de prototipado) para comprobar el funcionamiento y realizar la programación.

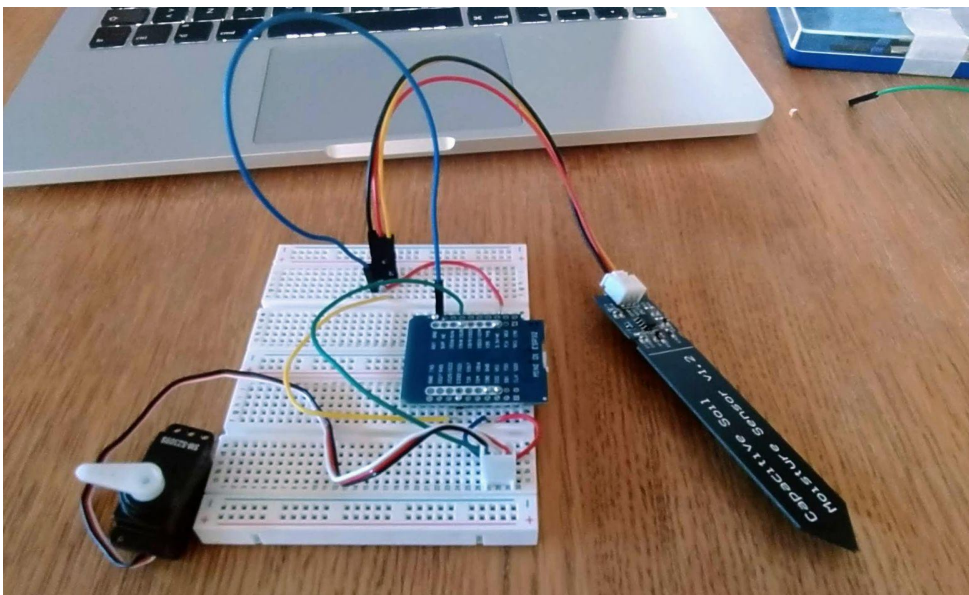


Figura 18: módulo de medición de humedad: prototipado.

La programación se realiza utilizando el ambiente de desarrollo de Microsoft Visual Studio Code con el framework Arduino y la plataforma PlatformIO.

En esta fase se crearán cuatro programas uno es el de funcionamiento del módulo mientras que tres son de test para las distintas partes de este: un programa se encargará del test del sensor de humedad, uno del test del servomotor y el último de la comunicación BLE del Esp32 Mini.

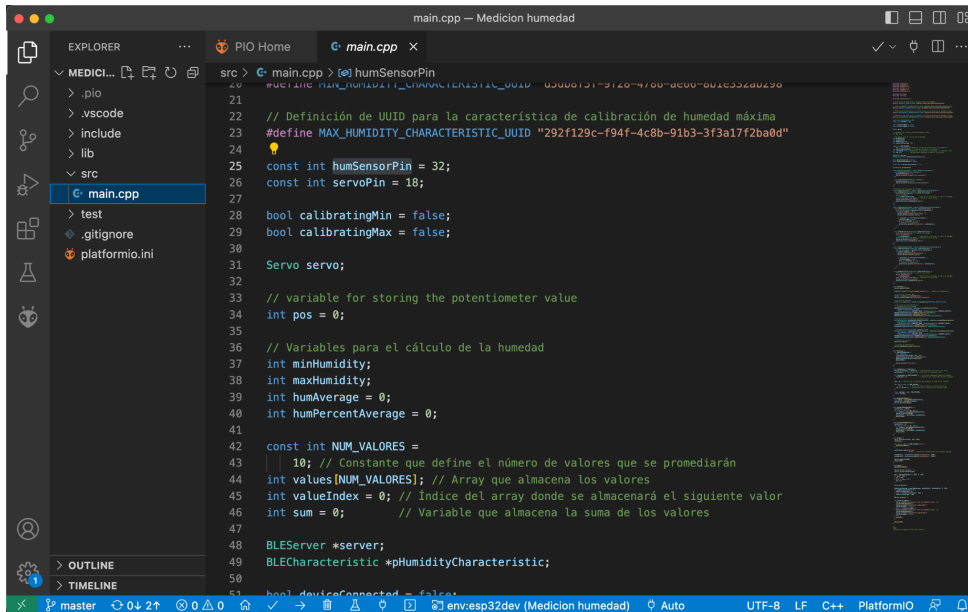


Figura 19: módulo de medición de humedad: ambiente de desarrollo de Visual Studio Code.

Una vez testadas las distintas partes del módulo y realizado el programa de funcionamiento del módulo se realizará el ensamblado del sistema.

Por último se realiza la caja que contiene el módulo con todos sus componentes.

Esto se realiza en el programa de diseño 3D, Tinkercad, con el cual se debe de modelar la entera caja partiendo y combinando formas básicas. El paso sucesivo es la impresión del modelo con una impresora 3D.

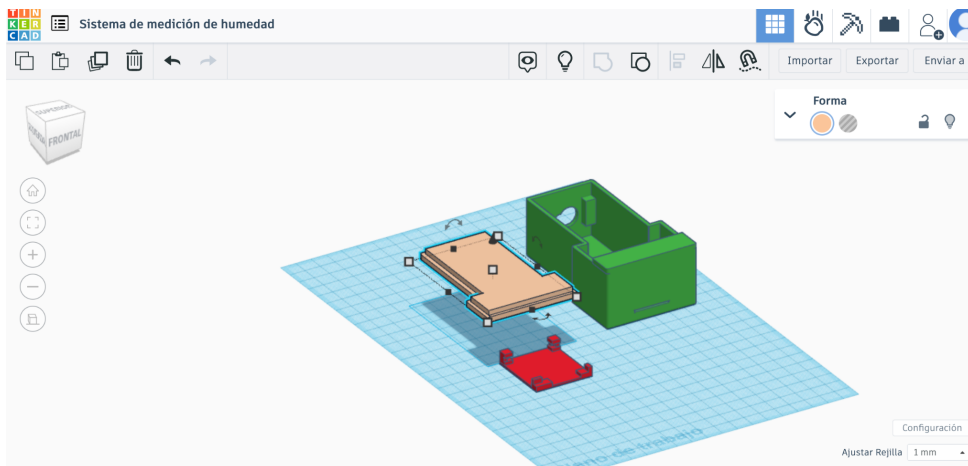


Figura 20: Tinkercad.

4.1.4 Software

El siguiente diagrama UML de actividad describe el funcionamiento del programa del módulo de medición de humedad de forma clara y organizada. El módulo puede encontrarse en tres estados distintos: RUNNING, MAX_HUM_CALIB y MAX_DRYNESS_CALIB, los cuales son activados mediante comandos enviados por BLE desde el módulo principal de IoT.

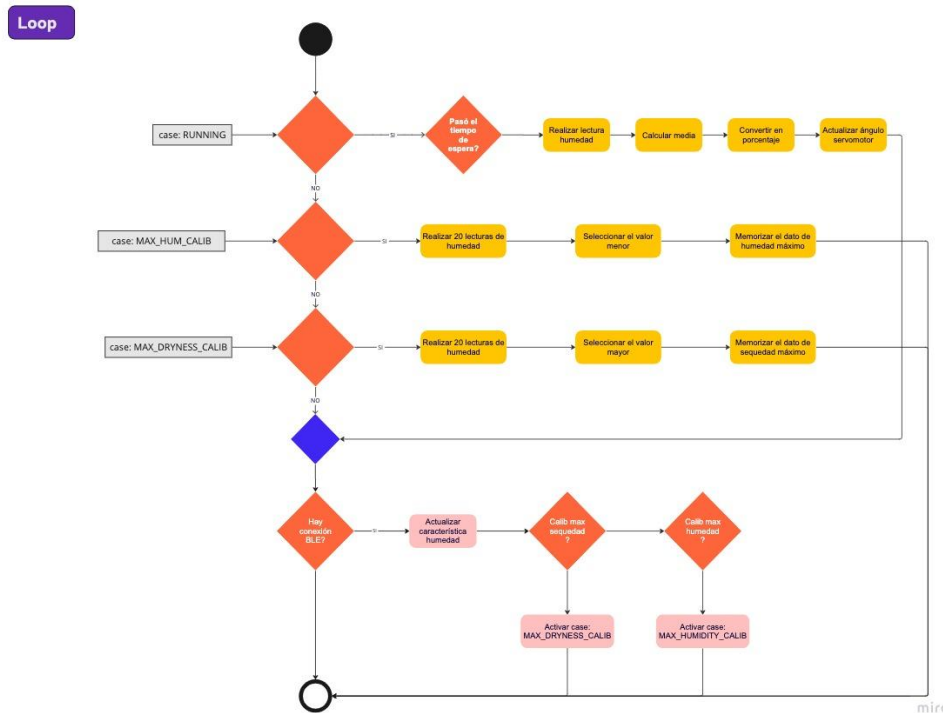


Figura 21: módulo de medición de humedad: diagrama UML de actividad.

En el estado RUNNING, el módulo de medición de humedad realiza lecturas periódicas del sensor de humedad. Una vez obtenida la lectura, se convierte en un valor de porcentaje que representa la humedad actual del suelo. Además, se calcula el ángulo correspondiente para indicar al servomotor la posición adecuada. En este caso, el módulo actúa como un servidor BLE y el módulo principal de IoT como cliente que solicita y recibe el último dato de humedad cuando lo necesita.

En el estado MAX_HUM_CALIB (calibración de máxima humedad), se lleva a cabo la determinación y memorización de la humedad máxima que el sensor de humedad puede alcanzar. Por ejemplo, esto podría lograrse sumergiendo completamente el sensor en agua.

En el estado MAX_DRYNESS_CALIB (calibración de máxima sequedad), se realiza la determinación y memorización de la humedad mínima que el sensor puede detectar. Por ejemplo, esto podría lograrse cuando el sensor se encuentra completamente seco.

4.2 Módulo de riego

4.2.1 Diseño de conexión

El circuito de este módulo consiste en un microcontrolador Esp32, que es el que contiene toda la lógica de funcionamiento y de comunicación BLE, una bomba de agua encargada del riego y un relé que está conectado a la bomba de agua para proveer la corriente necesaria para su activación.

El microcontrolador no está directamente conectado a la bomba de agua dado que esta al activarse requiere una cantidad de corriente elevada. Por este motivo una salida del microcontrolador se conecta al relé para controlar a través de este la activación de la mini bomba.

El entero sistema está alimentado por un powerbank de 8000 mAh que puede recargar a través de un panel solar integrado y que se conecta al microcontrolador a través de un cable micro USB.

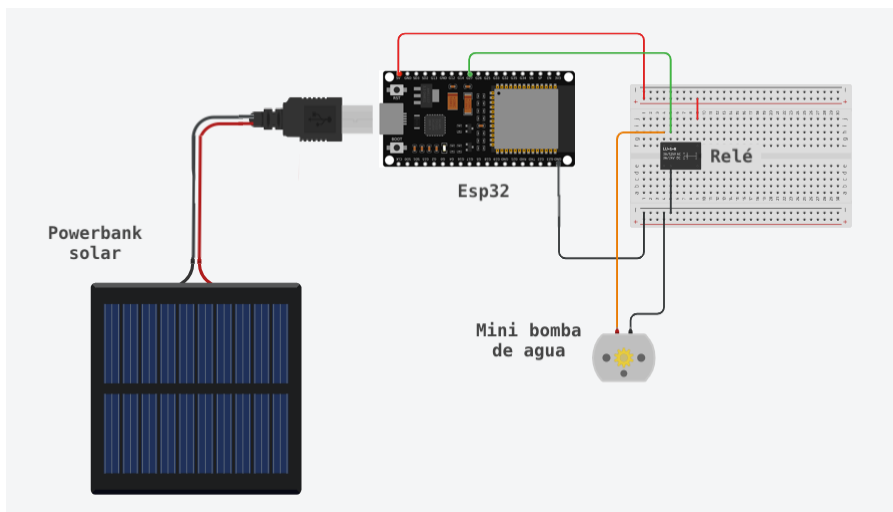


Figura 22: conexión de los componentes utilizados en el módulo de riego.

4.2.2 Esquema eléctrico

El siguiente esquema eléctrico indica las conexiones entre los componentes del sistema de riego.

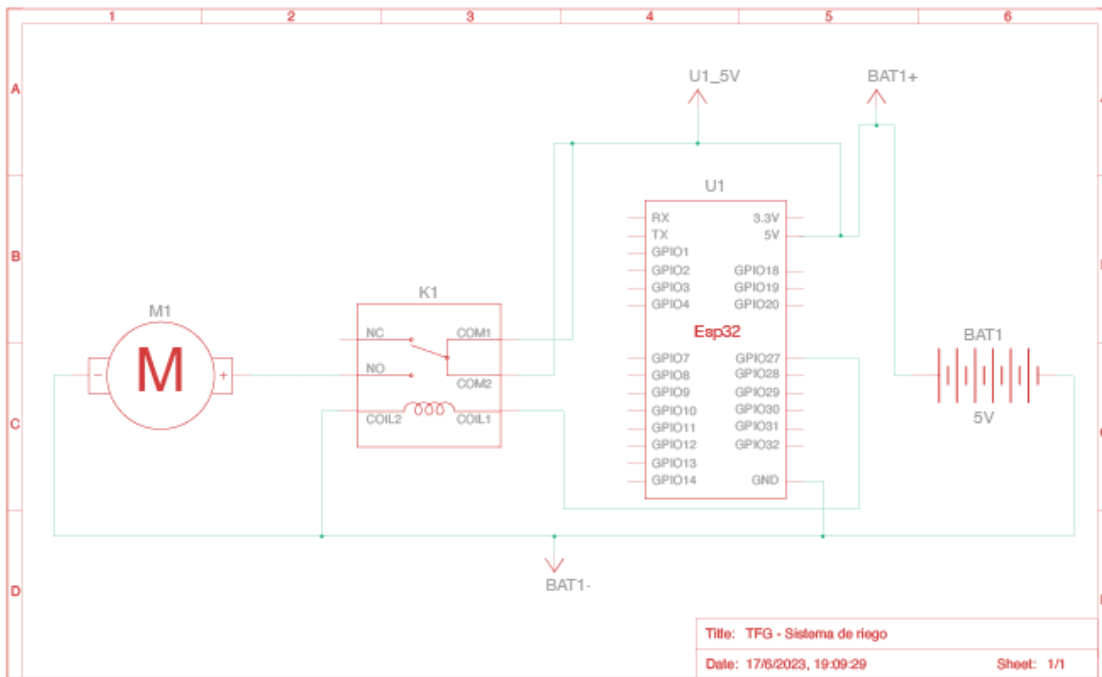


Figura 23: módulo de riego: esquema eléctrico.

Los componentes está elencado en la siguiente tabla:

Código	Componente	Código	Componente
U1	Esp32	K1	Relé 5VDC
M1	Mini bomba de agua DC 3V 5V	BAT1	Powerbank de 5V 8000 mAh

Tabla 3: módulo de medición de humedad: elenco de componentes.

4.2.3 Prototipado

La fase de prototipado implica probar el módulo inicialmente en una breadboard, donde se conecta el Esp32 al módulo de relés siguiendo el esquema eléctrico. La salida del relé se conectará a la mini bomba de agua, que será la encargada de suministrar agua a la planta.

Una vez se ha verificado la correcta conexión entre los componentes, se procede a desarrollar los programas de prueba y el programa final utilizando Visual Studio

Code. Esto permite realizar pruebas exhaustivas y asegurar el correcto funcionamiento del sistema antes de su implementación final.

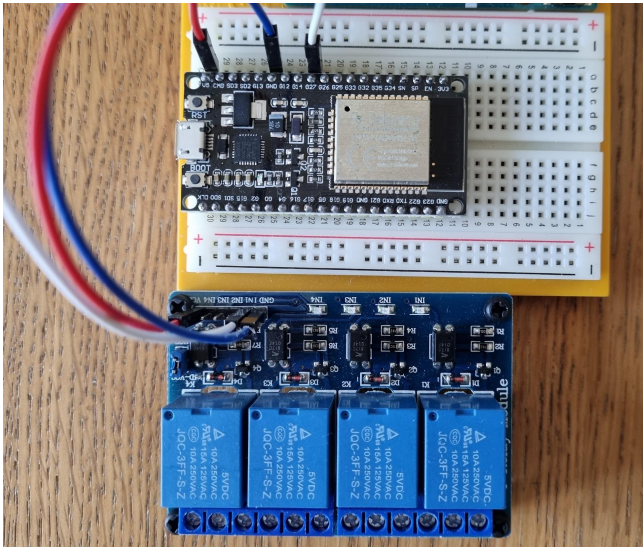


Figura 24: módulo de riego: prototipado.

Después de verificar el correcto funcionamiento de los programas, se procede a diseñar la caja que albergará el sistema utilizando el software de diseño 3D Fusion 360. Una vez finalizado el diseño, se imprime la caja utilizando una impresora 3D. Este proceso permite obtener una carcasa personalizada y adaptada a las dimensiones y necesidades del sistema, brindando protección y organización a los componentes internos.

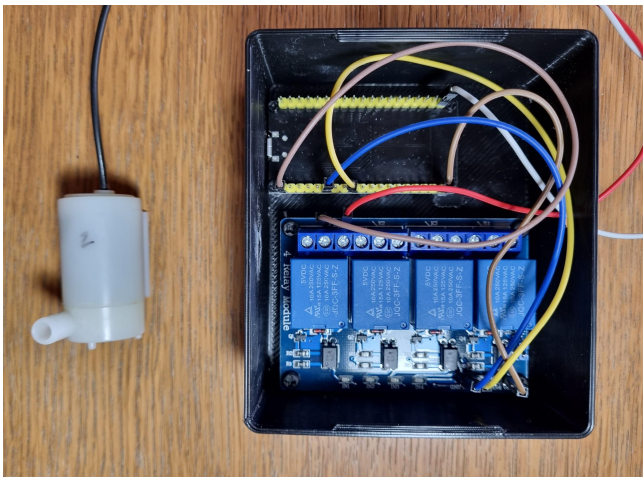


Figura 25: módulo de riego: prototipado.

4.2.4 Software

El siguiente diagrama UML de actividad describe el funcionamiento del programa del módulo de riego.

En cada ciclo, el programa verifica si ha recibido datos del módulo principal de IoT. En caso afirmativo, decodifica el mensaje recibido que contiene el número de salida que debe activarse y la duración en la que la bomba de agua debe permanecer activa. En este caso, el módulo actúa como un servidor BLE y el módulo principal de IoT como cliente que envía el dato de riego solo si es necesario.

El módulo no necesita calibración dado que la memorización del tiempo de actividad de cada salida se realiza en el módulo principal de IoT.

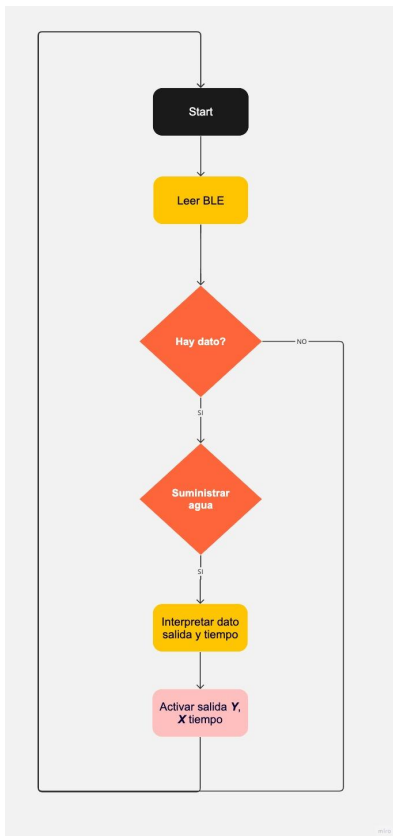


Figura 26: módulo de riego: diagrama UML de actividad.

4.3 Módulo principal de IoT

4.3.1 Diseño de conexión

El circuito de este módulo consiste en un microcontrolador Esp32, cuatro pulsantes con sus respectivas resistencias y una pantalla LCD OLED I2C de 1,3 pulgadas.

Los pulsadores vienen utilizados para la navegación en el menú creado para la configuración y prueba del sistema.

El entero sistema está alimentado por un powerbank de 8000 mAh que puede recargar a través de un panel solar integrado y que se conecta al microcontrolador a través de un cable micro USB.

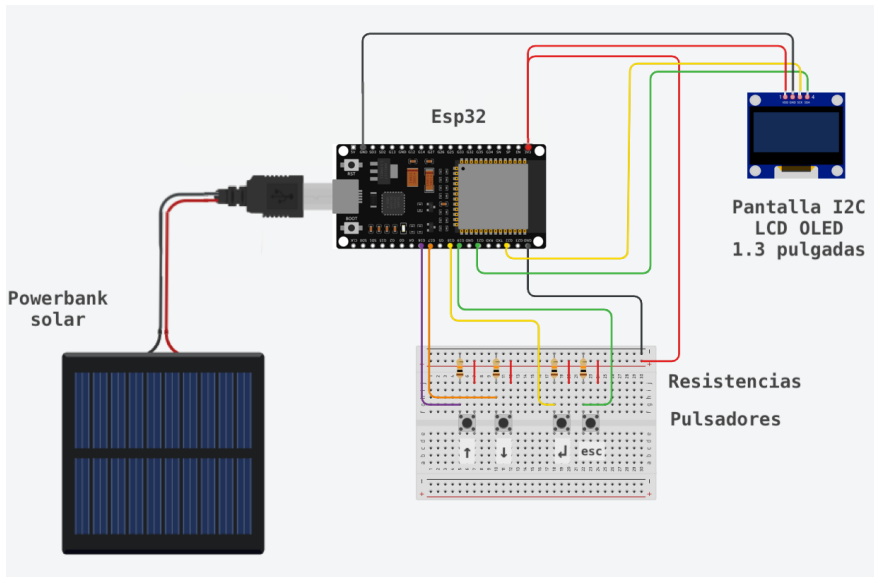


Figura 27: conexión de los componentes utilizados en el módulo principal de IoT.

4.3.2 Esquema eléctrico

El siguiente esquema eléctrico indica las conexiones entre los componentes del sistema de riego.

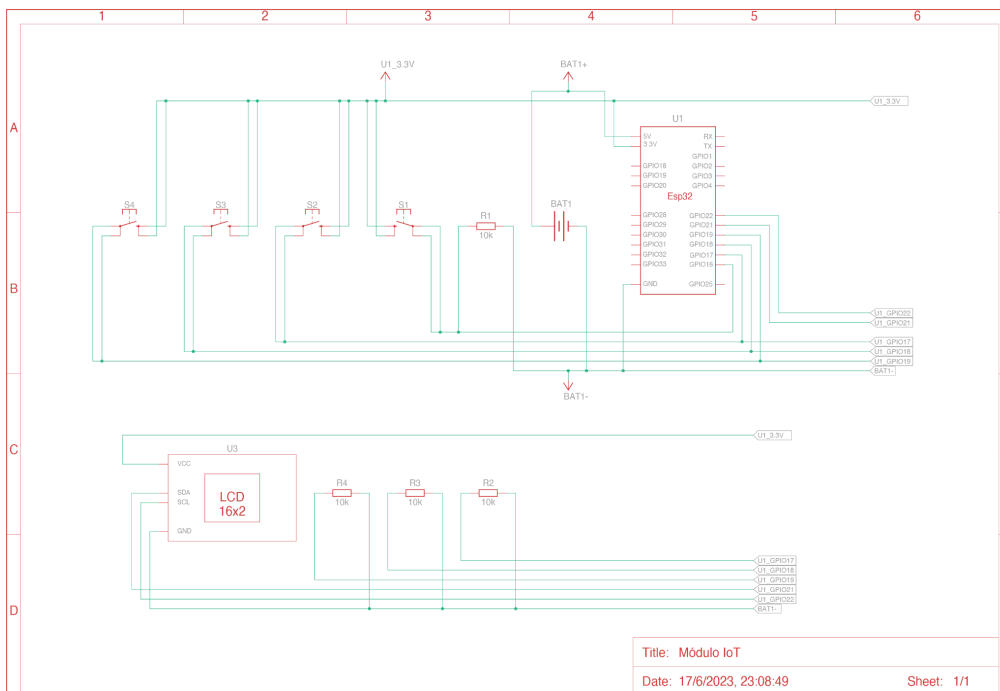


Figura 28: módulo principal de IoT: esquema eléctrico.

Los componentes está elencado en la siguiente tabla:

Código	Componente	Código	Componente
U1	Esp32	S1,S2,S3,S4	Pulsadores
BAT1	Powerbank de 5V 8000 mAh	R1, R2, R3, R4	Resistencias 10KΩ
U3	Pantalla LCD OLED I2C de 1,3 pulgadas y 128x64 pixeles		

Tabla 4: módulo principal de IoT: elenco de componentes.

4.3.3 Prototipado

Para el prototipado de este módulo, al igual que en los otros módulos, se realizó una prueba inicial de todos los componentes en una breadboard y se desarrollaron los programas de prueba correspondientes y el programa definitivo. Una vez confirmado el correcto funcionamiento del módulo, se procedió a soldar los pulsadores y las resistencias correspondientes en una PCB, estableciendo la conexión adecuada con el Esp32. Esta etapa de soldadura garantiza una conexión más estable y duradera, brindando un mejor rendimiento y facilitando el posterior ensamblaje del módulo en el sistema final.

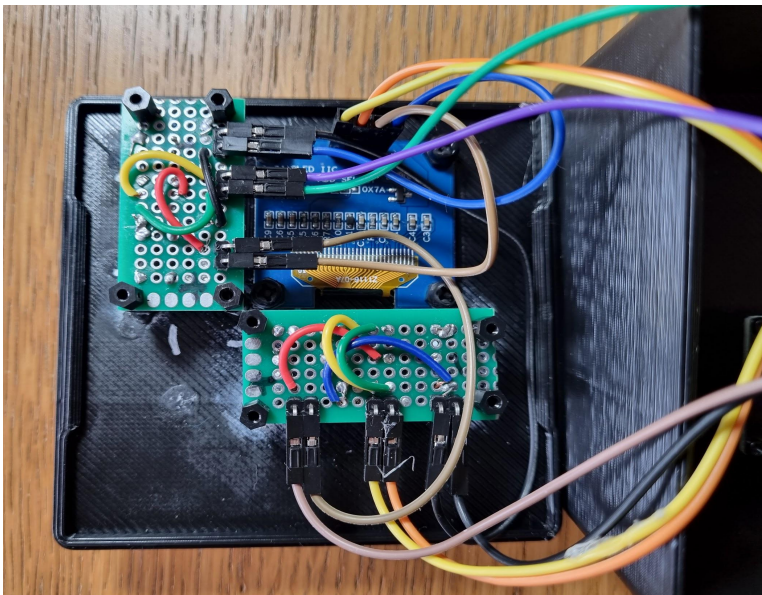


Figura 29: módulo principal de IoT: prototipado.

Luego de completar las etapas anteriores, se procedió a diseñar y crear la caja contenedora utilizando el software de diseño 3D Fusion 360. Posteriormente, se imprimió la caja utilizando una impresora 3D, asegurando un acabado preciso y ajustado a las dimensiones requeridas. Finalmente, se realizó el ensamblaje

definitivo del módulo, asegurando una correcta instalación de los componentes dentro de la caja impresa en 3D.



Figura 30: módulo principal de IoT:
pantalla principal

4.3.4 Software

El siguiente diagrama UML de actividad describe el funcionamiento del programa del módulo principal de IoT cuando está activada la función de control de humedad.

El programa verifica si se encuentra en la página principal (condición necesaria para el control de humedad) y si ha transcurrido el tiempo de espera entre ciclos. Si ambas condiciones se cumplen, se solicita la medición de humedad al módulo de medición correspondiente. A continuación, se verifica que el valor de humedad no supere el umbral preestablecido. Si la humedad excede el umbral, se lee el tiempo de riego previamente almacenado en la memoria del módulo para la planta, se establece una conexión con el módulo de riego y se envía el comando para activar el riego con la duración preestablecida. Por último, se muestra un mensaje de error en el display.

En cada lectura de humedad recibida, el módulo principal IoT envía ese dato al broker MQTT para que los pueda distribuir a sus suscriptores.

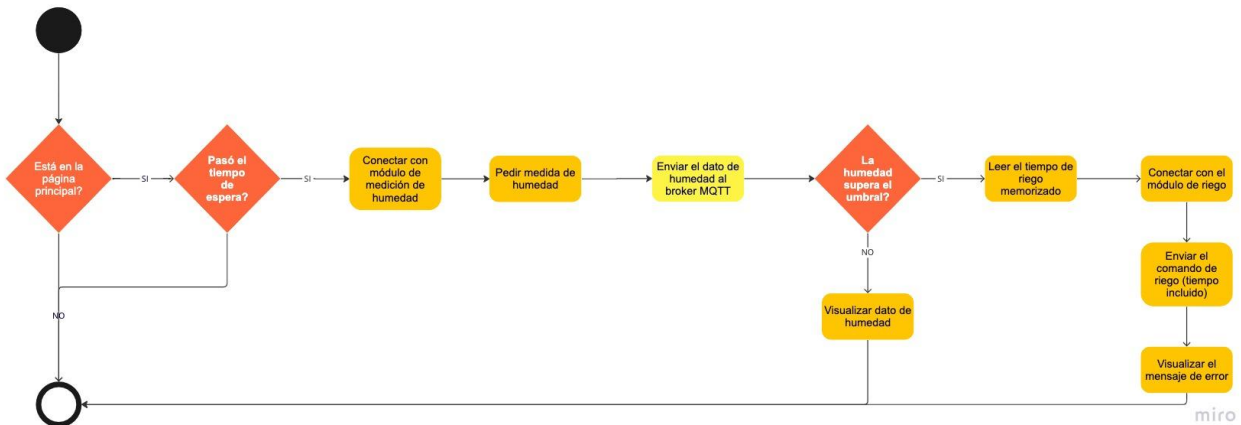


Figura 31: módulo principal de IoT: diagrama UML de actividad.

5 Resultados

El resultado final fue la creación exitosa de los tres módulos que componen el sistema: el módulo de medición de la humedad del suelo, el módulo de riego y el módulo principal IoT.

Estos tres módulos se comunican entre sí utilizando la tecnología BLE (Bluetooth Low Energy), mediante la lectura y escritura de características GATT. El módulo principal IoT se encarga de enviar los datos de humedad a un broker MQTT, el cual puede distribuir dichos datos a los clientes suscritos.

Asimismo, el sistema principal IoT cuenta con un menú interactivo que permite al usuario gestionar manualmente algunas funcionalidades del sistema, como el riego manual, la calibración del módulo de humedad y el control de la conexión con el broker MQTT.

En conclusión, el sistema de riego automático está completamente operativo y es capaz de gestionar automáticamente el riego de la planta, brindando una solución eficiente y funcional.



Figura 32: sistema automático de riego.

6 Conclusiones y trabajos futuros

Este proyecto ha representado un desafío significativo en varios aspectos, siendo el principal de ellos la integración exitosa de diversas tecnologías, protocolos, librerías y aplicaciones, logrando que funcionen de manera coordinada.

Si bien se logró el resultado final esperado en su totalidad, se realizaron cambios durante el desarrollo, especialmente a nivel de software: la lógica se encuentra completamente en el módulo principal IoT, mientras que los otros dos módulos actúan como componentes pasivos. Además, se trasladó la funcionalidad de memorización del tiempo de riego al módulo principal IoT en lugar del módulo de riego, al igual que el umbral de humedad con el módulo de medición de la humedad.

Si bien se lograron los objetivos principales del proyecto, los objetivos secundarios no pudieron ser alcanzados debido a la falta de tiempo. Algunas tareas no se completaron en el plazo previsto, lo que llevó a utilizar el tiempo asignado para los objetivos secundarios con el fin de lograr los objetivos principales.

El retraso en la planificación se debió principalmente a la falta de experiencia en el uso de la tecnología BLE con GATT y en la implementación de la comunicación MQTT. Además, la integración del servicio IoT en la nube presentó un desafío adicional. Aunque inicialmente se planteó como un servicio que permitía almacenar datos y visualizar un historial a través de una interfaz gráfica, al elegir el protocolo

MQTT se descubrió que los servicios gratuitos se limitan a un broker MQTT que distribuye los datos a los clientes suscritos.

La creación del menú del sistema principal IoT también generó retrasos en el proyecto. Inicialmente se utilizó una librería para gestionar las diferentes páginas del menú, pero después de varias pruebas se determinó que no era posible adaptarla a las necesidades del proyecto. Por lo tanto, se tomó la decisión de implementar el menú utilizando el patrón de programación State, lo cual supuso un verdadero reto.

En cuanto a los impactos ético-sociales, de sostenibilidad y diversidad previstos, el sistema de riego automático ha demostrado ser exitoso en algunos aspectos, como la eficiencia en el uso del agua y la mejora en el crecimiento de las plantas. Además, se abordaron de manera efectiva los impactos no previstos, optimizando el consumo energético mediante el uso de powerbank con recarga solar. Sin embargo, aún es necesario realizar un mayor estudio en áreas como el reciclaje del plástico de las cajas, el uso de materiales electrónicos más respetuosos con el medio ambiente y la protección de datos de los usuarios.

Las habilidades adquiridas durante este proyecto incluyen una mayor experiencia en tecnologías de hardware IoT, especialmente en la programación del microcontrolador ESP32, así como en tecnologías y protocolos de comunicación BLE, GATT y MQTT. También se ha ganado experiencia en programación en lenguaje C++, uso del IDE Microsoft Visual Studio Code, PlatformIO y programas de diseño 3D como Tinkercad y Fusion 360.

Las áreas de mejora para futuros desarrollos incluyen la gestión de errores en la comunicación BLE y MQTT, la mejora de la comunicación entre los módulos, la implementación de estadísticas de riego realizado y la implementación de los objetivos secundarios del proyecto, como el envío de comandos de riego a través de MQTT y el desarrollo de una aplicación cliente MQTT para Android.

En conclusión, este proyecto fue un desafío que presentó muchas dificultades en el camino, pero que al final brindó una gran satisfacción por los resultados obtenidos.

7 Bibliografía

- [1] MAROTO BORREGO, José Vicente. Historia de la Agronomía. Ediciones Mundi-Prensa, 2014.
- [2] ANDRADE, Fernando H. La tecnología y la producción agrícola. Ediciones INTA, 2011.
- [3] Food and Agriculture Organization of the United Nations (FAO). FAO - News Article: Los sistemas alimentarios representan más de un tercio de las emisiones mundiales de gases de efecto invernadero. 9 de marzo de 2023. Disponible en: <https://www.fao.org/news/story/es/item/1379490/icode/>. [Consulta: 22 de abril de 2023].
- [4] Ecologistas en Acción. Huertos urbanos, una década de crecimiento. [en línea]. 1 de diciembre de 2018. Disponible en: <https://www.ecologistasenaccion.org/114822/huertos-urbanos-una-decada-de-crecimiento/>. [Consultado el 22 de abril de 2023].
- [5] CURRIE, Beth Anne; BASS, Brad. Estimates of air pollution mitigation with green plants and green roofs using the UFORE model. Urban ecosystems, 2008, vol. 11, p. 409-422.
- [6] AGUILERA, Luis Hidalgo. La basura electrónica y la contaminación ambiental. Enfoque UTE, 2010, vol. 1, no 1, p. 46-61.
- [7] TANVEER, Abdullah, et al. Automated farming using microcontroller and sensors. International Journal of Scientific Research and Management Studies (IJSRMS), 2015, vol. 2, no 1, p. 21-30.
- [8] RAJESH, T., et al. IoT based smart agriculture monitoring system. International Journal of Scientific Engineering and Research (IJSER), 2020.
- [9] DAS, Ashok Kumar; ZEADALLY, Sherali; HE, Debiao. Taxonomy and analysis of security protocols for Internet of Things. Future Generation Computer Systems, 2018, vol. 89, p. 110-125.
- [10] Espressif Systems. (s.f.). ESP32 - Secure Wi-Fi & Bluetooth MCU. Recuperado 27 de abril de 2023, de <https://www.espressif.com/en/products/socs/esp32>.
- [11] Espressif Systems. (2023). ESP32WROOM32E/ESP32WROOM32UE Datasheet. Recuperado el 27 de abril de 2023, de https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e_esp32-wroom-32ue_datasheet_en.pdf
- [12] AZ-Delivery Vertriebs GmbH. (s.f.). AZ-Delivery Tienda Online para Arduino y Raspberry Pi en España. Recuperado el 27 de abril de 2023, de <https://www.az-delivery.de/es>

- [13] AZ-Delivery Vertriebs GmbH. (s.f.). ESP32 Dev Kit C V2 - Guía de inicio rápido. Recuperado el 27 de abril de 2023, de <https://www.az-delivery.de/es/products/esp32-nodemcu-kostenfreies-e-book?variant=8687877750880>
- [14] AZ-Delivery Vertriebs GmbH. (s.f.). ESP32 D1 Mini NodeMCU WiFi Modul + Bluetooth Internet Entwicklungsboard. Recuperado el 27 de abril de 2023, de <https://www.az-delivery.de/es/products/esp32-d1-mini-nodemcu-wifi-modul?variant=32763537817696>
- [15] CHAHID, Yassine; BENABDELLAH, Mohamed; AZIZI, Abdelmalek. Internet of things protocols comparison, architecture, vulnerabilities and security: State of the art. En Proceedings of the 2nd International Conference on Computing and Wireless Communication Systems. 2017. p. 1-6.
- [16] SINGH, Pushkar; SAIKIA, Sanghamitra. Arduino-based smart irrigation using water flow sensor, soil moisture sensor, temperature sensor and ESP8266 WiFi module. En 2016 IEEE Region 10 Humanitarian Technology Conference (R10-HTC). IEEE, 2016. p. 1-4.
- [17] CÓRCOLES CÓRCOLES, S. y MORENO MUÑOZ, A. Arduino. Madrid: RA-MA Editorial, 2018, p. 174-175. ISBN 84-9964-782-0.
- [18] PVS, Divya Dhatri, et al. A low-cost Arduino based automatic irrigation system using soil moisture sensor: design and analysis. En 2019 2nd International Conference on Signal Processing and Communication (ICSPC). IEEE, 2019. p. 105.
- [19] PVS, Divya Dhatri, et al. A low-cost Arduino based automatic irrigation system using soil moisture sensor: design and analysis. En 2019 2nd International Conference on Signal Processing and Communication (ICSPC). IEEE, 2019. p. 106.
- [20] Boualouache, A. E., Nouali, O., Moussaoui, S. y Derder, A., "A BLE-based data collection system for IoT," 2015 First International Conference on New Technologies of Information and Communication (NTIC), Mila, Algeria, 2015, pp. 1-5, doi: 10.1109/NTIC.2015.7368748.
- [21] V. Gazis, M. Görtz, M. Huber, A. Leonardi, K. Mathioudakis, A. Wiesmaier, F. Zeiger and E. Vasilomanolakis. "A Survey of Technologies for the Internet of Things", International Wireless Communications and Mobile Computing Conference, pp. 1090-1095, August 2015.
- [22] Benchmark of MQTT servers, version 1.1, January 2015.
- [23] HEDI, Ivan; ŠPEH, I.; ŠARABOK, A. IoT network protocols comparison for the purpose of IoT constrained networks. En 2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). IEEE, 2017. p. 501-505.

- [24] ELHADI, Sakina, et al. Comparative study of IoT protocols. Smart Application and Data Analysis for Smart Cities (SADASC'18), 2018.
- [25] Jardinerianomeolvides.es. "Sistemas de riego automático | Que es, tipos, ventajas y desventajas." [En línea]. Disponible en: <https://www.jardinerianomeolvides.es/sistemas-de-riego-automatico/>. [Accedido el 26/03/2023].
- [26] Guiadejardin.com. "Tres sistemas de riego automático para macetas." [En línea]. Disponible en: <https://www.guiadejardin.com/tres-sistemas-de-riego-automatico-para-macetas/>. [Accedido el 26/03/2023].
- [28] Libelium. "Soluciones IoT para una agricultura inteligente." [En línea]. Disponible en: <https://www.libelium.com/soluciones-iot-agricultura-inteligente/>. [Accedido el 26/03/2023].
- [29] Viagua.es. "Kit de riego." [En línea]. Disponible en: <https://viagua.es/kit-riego/#tab-con-13>. [Accedido el 26/03/2023].
- [31] Mundoriego.es. "Mundoriego - Sistemas de riego automático." [En línea]. Disponible en: <https://mundoriego.es/>. [Accedido el 26/03/2023].
- [33] Ventasinternacionales.com. "Riego por exudación." [En línea]. Disponible en: <https://ventasinternacionales.com/riego-por-exudacion/#:~:text=El%20riego%20por%20exudaci%C3%B3n%20es.continua%20en%20toda%20su%20longitud..> [Accedido el 26/03/2023].
- [34] Maherelectronica.com. "Programador de riego con electroválvulas para agricultura." [En línea]. Disponible en: <https://www.maherelectronica.com/programador-riego-electrovalvulas-agricultura/>. [Accedido el 26/03/2023].
- [35] Riegopro.com. "Programador de riego." [En línea]. Disponible en: <https://riegopro.com/programador-de-riego/>. [Accedido el 26/03/2023].
- [36] YouTube. (2023, mayo 7). CÓMO FUNCIONA UN RIEGO AUTOMÁTICO DE ELECTROVÁLVULAS. [Archivo de video]. Disponible en: <https://www.youtube.com/watch?v=H4CHDNvPyWY>
- [37] Pool Fun Store. (s.f.). Cómo funciona un programador de riego automático. [Blog post]. Disponible en: <https://www.poolfunstore.com/blog/como-funciona-un-programador-de-riego-automatico-51f/>.
- [38] VELASCO RODRÍGUEZ, Sergio. Agricultura de precisión amb sistemes de sensors agraris. 2020.

[39] MOLINA SUÁREZ, Javier. Regduino. 2019.

[40] ADRIÁN DE LA CÁMARA, Rubén. Arduino+ módulo GSM/GPRS: monitorización, automatización y gestión remota en un viñedo. 2017.

[41] Indeed. ¿Cuánto se gana como uno Desarrollador/a de software en España? [En línea]. Disponible en: [Suelo de Desarrollador/a de software en España \(indeed.com\)](https://www.indeed.com)

[42] CELOSIA, Guillaume; CUNCHE, Mathieu. Fingerprinting bluetooth-low-energy devices based on the generic attribute profile. En Proceedings of the 2nd International ACM Workshop on Security and Privacy for the Internet-of-Things. 2019. p. 24-31.

8 Anexos

8.1 Anexo 1: pruebas realizadas

Objetivo principal 1, módulo de medición de la humedad.

Prueba 1: Comprobación de la repetibilidad de las medidas de humedad.

Antes de iniciar las pruebas, se realizaron un total de 40 mediciones: 20 con el sensor en estado seco y 20 con el sensor sumergido en agua. Esto se hizo para encontrar los valores extremos y calibrar adecuadamente el sensor.

Posteriormente, se realizaron 50 mediciones con el sensor implantado en la tierra. De estas mediciones, se obtuvo un error máximo del 1,41%.

Dado que el error máximo permitido era del 5%, se puede concluir que esta prueba ha sido superada exitosamente.

Prueba 2: Comprobación de la fiabilidad de la visualización de los datos.

En esta prueba se realizaron 10 ciclos para comprobar la repetibilidad de la visualización de los datos utilizando un servomotor como indicador.

Para cada ciclo se envía el dato en grados empezando desde 0 grados (correspondiente a 0% en la medición de la humedad) e incrementando el valor en 45 grados (25% de humedad) hasta llegar a los 180 grados (100% de humedad).

Dado que el sistema de visualización no es altamente preciso, no se ha podido verificar las mediciones con una precisión elevada. Sin embargo, el margen de error nunca ha sido mayor a 5 grados, lo que representa un 2,7% y está por debajo del límite establecido del 5%.

Prueba 3: Comprobación del funcionamiento de la comunicación Bluetooth.

Se llevaron a cabo un total de 80 pruebas para verificar la conexión Bluetooth del Esp32. Estas pruebas evaluaron su funcionamiento a distancias de 1, 2, 5 y 10 metros, tanto en presencia como en ausencia de obstáculos como paredes y puertas.

Para realizar estas pruebas, se utilizó la aplicación nRF Connect en dispositivo Android. Esta aplicación permitió recibir y enviar datos a través de una característica configurada para la lectura y escritura en un programa de pruebas del Esp32.

Al concluir todas las pruebas, se pudo comprobar que en ningún caso se perdieron los datos enviados o recibidos. Por lo tanto, se considera que la prueba fue exitosa en su totalidad.

Objetivo principal 2, módulo de riego.

Prueba 1: Comprobación de la repetibilidad del suministro de agua:

Se realizaron un total de 40 riegos con diferentes duraciones: 1, 2, 5 y 10 segundos, con el objetivo de verificar la repetibilidad del sistema de riego. A partir de los datos obtenidos, se calculó la media del agua suministrada y se determinó la desviación máxima (error de suministro).

Los cálculos revelaron que la desviación máxima fue del 3,15%. Dado que el error máximo permitido era del 10%, se puede concluir que esta prueba fue superada exitosamente.

Prueba 2: Comprobación del funcionamiento de la comunicación Bluetooth.

Se han realizado las mismas pruebas que con el módulo de medición de la humedad y se ha obtenido el mismo resultado. Esto era de esperar, ya que el módulo bajo prueba es el Esp32 en ambos casos.

Objetivo principal 3, módulo principal IoT

Prueba 1: Comprobación del funcionamiento de la comunicación Bluetooth

Se han realizado las mismas pruebas que con el módulo de medición de la humedad y se ha obtenido el mismo resultado. Esto era de esperar, ya que el módulo bajo prueba es el Esp32 en ambos casos.

Prueba 2: Comprobación de la comunicación con el módulo de medición de humedad.

Se llevaron a cabo un total de 80 pruebas para verificar la conexión Bluetooth entre los dos módulos. Estas pruebas evaluaron su funcionamiento a distancias de 1, 2, 5 y 10 metros, tanto en presencia como en ausencia de obstáculos como paredes y puertas.

Al concluir todas las pruebas, se pudo comprobar que en ningún caso se perdieron los datos enviados o recibidos. Por lo tanto, se considera que la prueba fue exitosa en su totalidad.

Prueba 3: Comprobación de la comunicación con el módulo de riego

Se llevaron a cabo un total de 80 pruebas para verificar la conexión Bluetooth entre los dos módulos. Estas pruebas evaluaron su funcionamiento a distancias de 1, 2, 5 y 10 metros, tanto en presencia como en ausencia de obstáculos como paredes y puertas.

Al concluir todas las pruebas, se pudo comprobar que en ningún caso se perdieron los datos enviados o recibidos. Por lo tanto, se considera que la prueba fue exitosa en su totalidad.

8.2 Anexo 2: diferencias entre MQTT y CoAP y elección del protocolo a utilizar

La siguiente tabla enumera las características principales de los dos protocolos y destaca las diferencias sustanciales entre ellos [24].

	CoAP	MQTT
Standard	IETF	IBMs
Tecnología	XML	JS
Transporte	UDP	TCP
Arquitectura	response/request	Subscribe
2G, 3G, 4G Adecuación (noeuds 1000S)	Excelente	Excelente
Aptitude IIn (noeuds 1000s)	Excelente	Justo
Recursos calculados	10Ks RAM / Flash	10Ks RAM / Flash
Application	Utility Area Networks	Expandir mensajes empresariales a aplicaciones IoT
Seguridad y QoS	Ambos	Ambos

Factor de calidad	Autenticación de confiabilidad, integridad, confidencialidad	Fiabilidad
Ventajas	<ul style="list-style-type: none"> • Soporte de multidifusión • Gastos indirectos bajos • Complejidad de mapeo con http minimizada • Modelos de comunicación flexibles • Baja latencia 	<ul style="list-style-type: none"> • Tiempo real (Real Time) • Baja latencia • Fácilmente comprensible • Fácilmente ampliable • Cualquier servidor XMPP puede ser aislado
Desventajas	<ul style="list-style-type: none"> • No habilita el nivel de comunicación seguro • Pocas bibliotecas y soluciones de ejemplo existentes 	<ul style="list-style-type: none"> • Sobrecarga pesada de datos • No es adecuado para IoT incrustado aplicaciones

Tabla 5: comparativa entre MQTT y CoAP.

Conclusión y elección del protocolo a utilizar

Analizadas las principales características de los dos protocolos, el seleccionado para este proyecto es el **MQTT**.

Las razones principales por la elección de este protocolo son las siguientes:

- MQTT es un protocolo liviano y eficiente que permite la transferencia de datos en redes con recursos limitados, como las redes de sensores.
- MQTT es muy escalable y puede manejar grandes cantidades de dispositivos IoT. Es un protocolo de mensajería basado en el modelo de publicación-suscripción, lo que significa que puede enviar mensajes a múltiples suscriptores simultáneamente, en este caso se podría implementar el Objetivo Secundario 2: aplicación IoT para smartphones Android con un esfuerzo relativamente pequeño.
- MQTT es compatible con una amplia gama de plataformas y dispositivos, lo que lo convierte en una opción flexible para proyectos IoT como el presente. Igualmente, es compatible con múltiples lenguajes de programación y sistemas operativos, lo que lo hace fácilmente integrable con el framework de Arduino y Esp32, teniendo a disposición numerosa literatura.

8.3 Anexo 3: detalles de la viabilidad económica

Objetivo principal 1, módulo de medición de la humedad:

El módulo necesita de los siguientes elementos hardware:

Hardware	Precio
Sensor capacitivo de humedad.	8,54€
Microcontrolador ESP32 mini con módulo de comunicación Bluetooth incorporado.	9,96€
Servomotor.	3€
Batería 9V recargable.	10,94€
Módulo de regulador de 5V.	1,5€
Total:	33,94€

Tabla 6: objetivo principal 1: elenco de elementos.

Objetivo principal 2, módulo de riego automático:

El módulo necesita de los siguientes elementos hardware:

Hardware	Precio
Microcontrolador Esp32 (con Bluetooth BLE incorporado).	11,99€
Powerbank solar 5V 8000mA	27,88€
Relés de 4 Canales a DC 5V.	9,99€
Bomba de Agua DC 3V.	7,25€
Total:	57,11€

Tabla 7: objetivo principal 2: elenco de elementos.

Objetivo principal 3 y 4, módulo principal de IoT:

El sistema necesita de los siguientes elementos hardware:

Hardware y servicio IoT	Precio
Microcontrolador Esp32 (con Bluetooth BLE y Wifi incorporados).	11,99€

Powerbank solar 5V 8000mA	27,88€
Pantalla OLED LCD I2C, 1.3 pulgadas, 128x64 píxeles	8,00€
Total:	47,87€

Tabla 8: objetivo principal 3: elenco de elementos.

Objetivo secundario 1, añadir funcionalidades al servicio IoT para la interacción con los distintos sistemas:

En esta fase se considera el software y las horas de trabajo.

Objetivo secundario 2, aplicación IoT para smartphones Android:

En esta fase se considera el software y las horas de trabajo.

Software

Software	
Ambiente de desarrollo, Framework y librerías compatibles con Esp32.	Gratis
Servicio de IoT en la nube.	Gratis
Total:	Gratis

Tabla 9: elenco de software.

Horas de trabajo

Horas totales de investigación, diseño e implementación	Precio
90 horas x 13,6 €/hora*	1224 €

Tabla 10: horas totales trabajadas.

*Precio medio de un desarrollador de software/hardware IoT junior [41] .

8.4 Esquema funcional del menú del módulo principal de IoT

El siguiente esquema describe la navegación del menú del módulo principal de IoT.

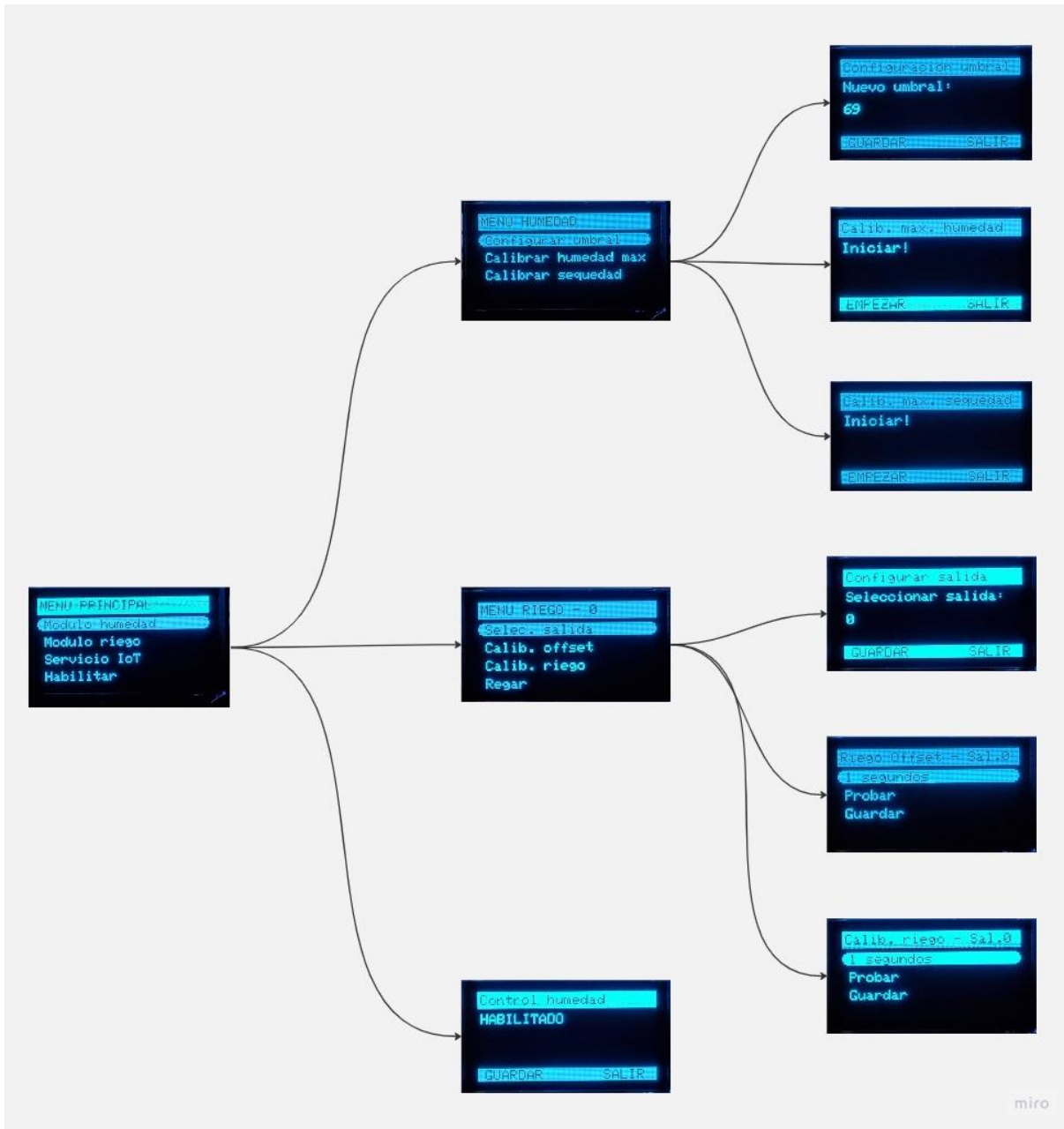


Figura 33: esquema de la navegación del menú del módulo principal de IoT.