

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR
FACULTAD DE INGENIERÍA
CARRERA DE INGENIERÍA EN TECNOLOGÍA DE LA
INFORMACIÓN



Tema:

Desarrollo de un prototipo de aplicación que permita visualizar el
ecosistema florícola domotizado para el cultivo y cuidado de plantas

AUTOR:

Fabricio Alexander Rojas Fonseca

DIRECTOR:

Juan Francisco Chafla Altamirano

QUITO DM, JULIO DE 2025

CONTENIDOS

ÍNDICES	4
Dedicatoria	5
Resumen	6
1. CAPÍTULO I: INTRODUCCIÓN.....	7
1.1. Justificación.....	8
1.2. Planteamiento del problema	9
1.3. Objetivo	9
1.3.1. Objetivo General	9
1.3.2. Objetivos Específicos.....	9
1.4. Alcance	9
2. CAPÍTULO II: MARCO TEÓRICO Y CONCEPTUAL.....	11
2.1. Antecedentes	11
2.1.1. Domótica en la agricultura	12
2.1.2. Microcontroladores Raspberry pi.....	12
2.1.3. Riego inteligente	13
2.2. Razones para la automatización en el cultivo de plantas.....	13
2.3. Beneficios del sistema propuesto	14
2.4. Preguntas de investigación	14
2.4.1. Uso de Raspberry Pi 5.....	15
2.4.2. Principios	17
2.4.3. Limitaciones.....	17
2.5. Marco Conceptual	18
2.2.1. Automatización.....	18
2.2.2. Internet de las Cosas (IoT).....	19
2.2.3. Sensores y actuadores.....	20
2.3. Red.....	20
2.4. Blynk.cloud IoT	20
3. CAPÍTULO III: DESARROLLO	22
3.1. Planificación técnica	22
3.1.1. Descripción inicial del problema	22
3.1.2. Objetivos técnicos del sistema.....	22
3.1.3. Elección de la arquitectura general.....	22
3.1.4. Listado de componentes de hardware y software	23
3.1.5. Hardware	23

3.1.6.	SOFTWARE	30
3.2	Diseño del Software	32
3.2.1.	Estructura del código	32
3.2.2.	Inicialización de sensores	33
3.2.3.	Lectura de valores y bucle principal	35
3.2.4.	Recepción de comandos manuales.....	36
3.2.5.	Ejecución en la Raspberry Pi y pruebas de consola.....	37
3.2.6.	Automatización y monitoreo.....	39
3.2.7.	Lógica de riego automático.....	40
3.3.	Integración con Blynk IoT	41
3.3.1	Configuración de dashboard	41
3.3.2.	Configuración de datastreams y pines virtuales	43
3.3.2	Interfaz en aplicación móvil	51
3.4.	Proyección futura	52
3.4.1.	Escalabilidad y red	52
3.4.2.	Autonomía Energética Uso de energía solar.....	52
3.4.3.	Mejora de la interfaz Aplicación móvil propia.....	52
3.4.4.	Aplicación social y comercial.....	52
3.4.5.	Inteligencia Artificial	52
4.	CAPÍTULO IV: RESULTADOS	53
4.2.	Análisis de Resultados	53
4.1.1	Pruebas del riego manual	53
4.1.2	Riego automático programado.....	55
4.1.3	Lógica de riego automático por condiciones	56
4.1.3.1	Evaluación del sensor de humedad del suelo.....	56
4.1.3.2	Evaluacion del sensor de temperatura ambiental.....	57
4.1.3.3	Evaluación de sensores de luminosidad	58
4.1.3.4	Evaluación del sensor ultrasónico para el nivel del tanque.....	59
5.	CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES.....	62
5.2.	Conclusiones.....	62
5.3.	Recomendaciones.....	62
6.	BIBLIOGRAFÍA	64

ÍNDICES

Figura 1 Raspberry pi 5.....	16
Figura 2 Sistema operativo.....	16
Figura 3 Prototipo cableado.....	23
Figura 4 Raspberry pi 5.....	24
Figura 5 Sensor de humedad del suelo capacitivo - APM-969.....	24
Figura 6 Sensor de temperatura y humedad DHT22	25
Figura 7 Sensor de luz BH1750.....	25
Figura 8 Sensor ultrasónico HC-SR04	26
Figura 9 Módulo ADS1115.....	27
Figura 10 Módulo relé de 2 canales	27
Figura 11 Bomba de agua de 5V.....	28
Figura 12 Fuente 5V	29
Figura 13 Disipadores de calor	29
Figura 14 Tarjeta microSD de 32 GB.....	30
Figura 15 Instalación de Blynk_env	33
Figura 16 Configuración inicial del sistema y parámetros de riego automatizad	34
Figura 17 Envío de datos a Blynk.....	36
Figura 18 Activadores de bombas normal y nutrientes	36
Figura 19 Control manual de riego por pines virtuales en Blynk	37
Figura 20 Ejecución del sistema y conexión a Blynk Cloud desde Raspberry Pi.....	38
Figura 21 Automatización del inicio del sistema	38
Figura 22 Lógica de automatización del riego basada en condiciones ambientales	39
Figura 23 Lógica de automatización del riego basada en condiciones ambientales	41
Figura 24 Configuración de la aplicación Blynk InT	42
Figura 25 Configuración de Datastreams	43
Figura 26 Configuración del Virtual Pin de Activar bomba normal	44
Figura 27 Configuración del Virtual Pin de Humedad	45
Figura 28 Configuración del Virtual Pin Luz	46
Figura 29 Configuración del Virtual Riego de agua.....	47
Figura 30 Configuración del Virtual Pin Riego de nutrientes	48
Figura 31 Configuración del Virtual Pin de Temperatura	49
Figura 32 Configuración del Virtual Pin Riego de Activar bomba de nutrientes	50
Figura 33 Aplicación móvil	51
Figura 34 Dashboard de los botones de activación de bomba de nutrientes y normas	53
Figura 35 Verificación del funcionamiento del sistema de bombeo	54
Figura 36 Verificación del funcionamiento del riego automático con la primera hora	55
Figura 37 Verificación del funcionamiento del riego automático con la segunda hora	56
Figura 38 Verificación del funcionamiento del sensor de humedad del suelo con su dashboard	56
Figura 39 Verificación del funcionamiento del sensor de temperatura ambiental con su dashboard.....	58
Figura 40 Evaluación del funcionamiento del sensor de luminosidad con su dashboard	59
Figura 41 Evaluación del funcionamiento del ultrasónico para el nivel del tanque con su dashboard.....	60
Figura 42 Evaluación del funcionamiento de las métricas del tanque con su dashboard.....	61

Dedicatoria

A Dios, por ser luz en mis momentos más oscuros y fortaleza cuando sentí que no podía continuar. Gracias por darme paz en medio del caos, por abrir caminos donde todo parecía cerrado y por recordarme que, con fe, incluso lo incierto puede encontrar sentido. Este logro también es testimonio de Tu presencia constante en mi vida.

A mi madre, con todo mi corazón, cada desvelo, cada línea de este trabajo, cada pequeño avance lleva tu nombre. Por tus sacrificios para que este trabajo se haga realidad, por ser ejemplo de lucha, amor y entrega sin condiciones. Por creer en mí incluso en los momentos en que yo mismo no lo hacía. Sin ti, nada de esto habría sido posible. Eres mi fuerza, mi refugio y mi mayor bendición.

A mis profesores, especialmente a quienes acompañaron y guiaron este proyecto, por sus enseñanzas, por exigirme siempre un poco más y por su compromiso con nuestra formación. Gracias por marcar una diferencia.

Y a mí mismo, por no rendirme, por seguir a pesar del cansancio, por aprender de los tropiezos y por levantarme una y otra vez. Hoy, al mirar atrás, sé que todo el esfuerzo valió la pena.

Resumen

Este trabajo presenta el desarrollo de un prototipo enfocado en el monitoreo y automatización del riego en un ecosistema florícola, utilizando tecnologías como Raspberry Pi 5 y sensores ambientales, se busca mejorar la eficiencia del manejo de cultivos ornamentales mediante la recopilación, análisis y visualización en tiempo real de variables como humedad del suelo, temperatura, luminosidad y nivel de agua.

A partir de una arquitectura domotizada, el sistema permite controlar de forma remota y automática los procesos de riego, adaptándose en tiempo real a las condiciones del entorno mediante sensores y actuadores conectados. Para la interfaz de usuario se empleó la plataforma Blynk IoT, lo que permitió desarrollar una solución accesible desde dispositivos móviles o navegadores web, facilitando el monitoreo continuo del estado del cultivo

El prototipo fue evaluado en un entorno controlado, lo que permitió comprobar su funcionamiento tanto en modos de riego manual como automático, se verificó su capacidad para reaccionar ante eventos programados y cambios en las condiciones ambientales. Los resultados obtenidos evidencian una reducción significativa en el uso innecesario de agua, una mejora en la salud de las plantas y una mayor facilidad para los productores florícolas a la hora de tomar decisiones informadas.

Este prototipo demuestra que es viable diseñar soluciones tecnológicas sostenibles, económicas y escalables aplicables al sector agrícola, se concluye que la implementación de sistemas domotizados puede transformar las prácticas tradicionales, abriendo nuevas posibilidades para la innovación y el desarrollo en el ámbito de la floricultura.

1. CAPÍTULO I: INTRODUCCIÓN

El sector florícola se enfrenta a múltiples desafíos, por ejemplo, manejo del recurso hídrico y el control de las condiciones ambientales, las prácticas tradicionales de riego y la supervisión manual derivan en el uso inadecuado del agua, que repercute en la calidad del cultivo, eleva los costos de producción y compromete la sostenibilidad de los procesos.

Frente a esta problemática, este trabajo propone el desarrollo de un prototipo de aplicación basado en tecnologías de bajo costo, como Raspberry Pi y sensores ambientales, con el propósito de monitorear un ecosistema florícola domotizado en tiempo real. A través de la aplicación el usuario podrá visualizar variables como la humedad del suelo, la temperatura, el pH y la luminosidad, lo que facilitará la toma de decisiones en el manejo de cultivos ornamentales.

Cabe destacar que, esta propuesta se enmarca en el paradigma del Internet de las Cosas (IoT), integrando sensores, actuadores y una interfaz de monitoreo remoto mediante la plataforma Blynk. De esta manera, se busca demostrar que es posible desarrollar soluciones sostenibles a productores, contribuyendo así a la eficiencia operativa del sector florícola mediante sistemas domotizados.

La floricultura, presenta dificultades operativas que afectan la eficiencia del cultivo ornamental, uno de los más urgentes y también de los más invisibles es el uso del agua. En muchos cultivos todavía se riegan las plantas sin ninguna medición con sistemas tradicionales y una supervisión manual que, por más experiencia que se tenga, no siempre alcanza para garantizar eficiencia. Y cuando el agua no se gestiona bien, todo se resiente: baja la calidad de las flores, suben los costos y, en última instancia, se vuelve más difícil sostener el cultivo en el tiempo.

Frente a ese escenario, este proyecto plantea una idea bastante concreta: desarrollar una aplicación sencilla pero funcional, basada en tecnologías accesibles como Raspberry Pi y sensores ambientales, que permita tener un control mucho más preciso del entorno donde crecen las flores. La meta de este prototipo es que cualquier persona sin necesidad de ser

experta en tecnología pueda consultar desde su celular o computadora si el suelo está seco, si hace demasiado calor, cómo está el pH o cuánta luz reciben las plantas. Y, con esa información en mano, decidir qué hacer.

Todo esto se enmarca en lo que se conoce como Internet de las Cosas (IoT), en pocas palabras es conectar dispositivos físicos a internet para que “hablen” entre sí y con nosotros presentándonos los datos, en este caso, sensores, actuadores que pueden responder a esos datos, y una interfaz como Blynk IoT que sirve de puente para monitorear todo en tiempo real. Lo que se busca con esta propuesta no es solo demostrar que se puede automatizar un cultivo, sino hacerlo de forma sostenible, práctica y pensada para quienes realmente lo necesitan: los productores. Porque si hay una forma de trabajar mejor y cuidar los recursos al mismo tiempo, vale la pena intentarlo.

1.1. Justificación

El cultivo de plantas ornamentales es, en muchos sentidos, un trabajo delicado. Basta con que una variable ambiental se des controle como la luz, la temperatura o la humedad para que la calidad de las plantas se vea afectada. Y aunque esto es algo bien sabido en el sector, debido a que varias floriculturas todavía dependen de métodos manuales tanto para el riego como para el monitoreo diario. Lo que resulta en el desperdicio de agua, tiempo y los cultivos quedan más expuestos a condiciones desfavorables.

Los microprocesadores Raspberry Pi 5, pueden marcar una diferencia real. Gracias a ellos, es posible optimizar el uso tanto del agua como de la energía, reduciendo la carga de trabajo para los productores y mejorando, al mismo tiempo, la eficiencia en la producción. Al integrar sensores ambientales con plataformas de monitoreo en tiempo real como Blynk.Cloud se puede saber al instante cómo está el cultivo, lo que ayuda a prevenir pérdidas que antes eran inevitables por las limitantes técnicas y cambios ambientales.

Desarrollar un prototipo de domótica no es solo un ejercicio técnico: es una apuesta por soluciones viables que fortalezcan las tareas del día a día en el campo. Este trabajo, en ese sentido, no solo busca hacer más sostenibles los procesos agrícolas actuales, sino también abrir la puerta a mejoras futuras que amplíen su impacto dentro del sector florícola.

1.2. Planteamiento del problema

En la floricultura tradicional, el riego y la falta de monitoreo provocan problemas en el uso del agua, lo que dificulta la detección de cambios que ocurren en el entorno que afectan negativamente el desarrollo de las plantas, la falta de seguimiento dificulta detectar a tiempo los cambios en las condiciones ambientales, lo que puede afectar seriamente el desarrollo de las plantas y, en consecuencia, la salud de los cultivos, no se trata solo de gestionar bien el recurso hídrico, sino también de tener la capacidad de reaccionar con rapidez ante cualquier evento inesperado que pueda poner en riesgo el cultivo.

La falta de automatización y de herramientas de monitoreo para florícolas lleva a pérdidas económicas, disminución de la calidad del producto, mayores costos por uso de recursos hídricos, se cuenta con muchas soluciones tecnológicas ya están disponibles en el mercado, requieren inversiones elevadas o conocimientos avanzados, por lo que, las hacen inaccesibles a los pequeños productores.

Frente a esta problemática, surge la idea de desarrollar un sistema automatizado que permita monitorear variables ambientales y controlar el riego de manera eficiente, utilizando una arquitectura en Raspberry Pi, sensores ambientales y plataformas IoT.

1.3. Objetivo

1.3.1. Objetivo General

Desarrollar un sistema domotizado de monitoreo y riego automatizado para una florícola, utilizando una Raspberry Pi como controlador central, sensores ambientales y una plataforma de visualización remota.

1.3.2. Objetivos Específicos

- Diseñar un sistema de adquisición de datos a través de sensores ambientales.
- Automatizar el riego de acuerdo con valores críticos de humedad y pH.
- Implementar una interfaz móvil para monitoreo y control remoto usando Blynk.
- Evaluar el funcionamiento del sistema mediante pruebas en un prototipo a pequeña escala.

1.4. Alcance

El proyecto se enfoca en una solución de bajo costo, adaptable a pequeñas y medianas florícolas, con una capacidad de monitoreo local y remoto mediante conexión WiFi, no

contempla sistemas de fertilización automatizada ni algoritmos de predicción avanzados, pero existe la posibilidad de una expansión futura.

2. CAPÍTULO II: MARCO TEÓRICO Y CONCEPTUAL

El desarrollo de un ecosistema florícola domotizado abarca a la domótica, a los sistemas embebidos, la automatización y el Internet de las Cosas (IoT). Este capítulo nos introduce en fundamentos teóricos y conceptuales que abordan las bases necesarias para comprender la tecnología como Raspberry Pi, sensores ambientales y plataformas de monitoreo remoto en el sector florícola.

En la primera parte, se describen los antecedentes del uso de la domótica en la agricultura y su impacto en la optimización del riego, resaltando la importancia de estas tecnologías para mejorar la sostenibilidad y productividad en cultivos ornamentales, consiguiente se detallan los principios técnicos del IoT, la funcionalidad de los sensores y actuadores, y las ventajas del uso de microcontroladores como la Raspberry Pi.

Asimismo, se abordan conceptos sobre automatización, redes inalámbricas y control centralizado, esta base teórica servirá para el diseño e implementación del prototipo.

2.1. Antecedentes

La agricultura ha experimentado avances importantes gracias a las tecnologías digitales que permiten optimizar el uso de recursos y mejorar la eficiencia en los procesos de cultivo. Gracias a la automatización del riego y el monitoreo ambiental a través de sensores son áreas con mayor potencial, ya que, permiten recolectar datos en tiempo real sobre variables como la humedad del suelo, la temperatura y la luminosidad, estos sensores son útiles en el cultivo de plantas ornamentales, donde la variabilidad de los factores ambientales impacta en la calidad del producto.

El uso de microprocesador y plataformas accesibles como Raspberry Pi han permitido el desarrollo de prototipos domotizados a bajo costo que están enfocados en la supervisión y automatización de tareas agrícolas. Investigaciones como la de Kalluri y Kalidindi (2020), destacan que los sistemas de riego inteligente no solo reducen el desperdicio de agua, sino que también aumentan la productividad al mantener condiciones óptimas para el desarrollo de las plantas (p.127). Los autores como Madakam, Ramaswamy y Tripathi (2015), subrayan que el Internet de las Cosas (IoT) ha transformado la gestión agrícola, facilitando la conexión entre dispositivos y la toma de decisiones basada en datos (p.165).

Gracias a los estudios de Mahmud, Zahid y Das (2023) destacan que la implementación de tecnologías de automatización, basadas en plataformas IoT, ha mejorado significativamente la eficiencia en el monitoreo y cuidado de cultivos ornamentales, permitiendo optimizar recursos y reducir la intervención humana en tareas repetitivas a la vez permitiendo entender el estado actual de la planta.

Aún existen limitaciones en la adopción de soluciones accesibles y adaptables para pequeña escala, especialmente en el sector florícola, con este trabajo busca aportar en el desarrollo de herramientas tecnológicas sencillas y funcionales que permitan el monitoreo, control del riego y las condiciones ambientales de forma eficiente.

2.1.1. Domótica en la agricultura

La domótica en la agricultura es una herramienta que optimiza el uso de recursos como el agua, la energía y los insumos agrícolas, por medio de la integración de tecnologías como sensores, microcontroladores, actuadores y plataformas digitales es posible automatizar procesos como el riego y de la mano del monitoreo la iluminación, temperatura, esta herramienta se ajusta con sistemas de producción sostenibles y eficientes frente a las variaciones del clima.

En el caso de los cultivos ornamentales que requieren un control más preciso, la domótica permite garantizar condiciones constantes y adecuadas para su correcto crecimiento, de acuerdo con Agroingenia Canarias (2023), la implementación de sistemas de riego automatizado mejora significativamente el uso del agua, aspecto fundamental ante la creciente escasez hídrica y los altos costos asociados al desperdicio de recursos, esta automatización disminuye la necesidad de la intervención humana, reduciendo errores y aumentando la productividad agrícola favoreciendo al mercado.

2.1.2. Microcontroladores Raspberry pi

Raspberry Pi es un microprocesador con costo accesible, de alto rendimiento y perfecto para proyectos de automatización, monitoreo ambiental. Funciona como una minicomputadora, puede ejecutar programas completos y almacenar grandes cantidades de datos. Es ideal para proyectos agrícolas que requieren conectividad, procesamiento local de equipos y control eléctrico de sensores.

Es importante poder conectarse a redes Wi-Fi y plataformas IoT, lo que permite Blynk IoT, implementar soluciones de monitoreo accesibles desde cualquier dispositivo móvil. Cuenta con compatibilidad con sensores analógicos y digitales facilita la recopilación de datos, como la humedad, la temperatura y los niveles de luz. Según SwitchDoc Labs (2018), la Raspberry Pi ha demostrado ser especialmente útil en sistemas de jardinería y agricultura inteligente, ya que permite controlar en tiempo real el estado de las plantas y activar respuestas automáticas ante cambios críticos en el entorno.

2.1.3. Riego inteligente

El riego inteligente es crucial para automatizar la dosis de agua en cultivos mediante el uso de sensores y sistemas de control que responden a las condiciones del entorno y del tipo de planta, se diferencia con el riego manual, el riego inteligente se basa en datos en tiempo real, lo que permite ajustar la cantidad, duración y frecuencia del riego con precisión.

Este tipo de sistemas contribuye a la sostenibilidad del cultivo, ya que evita tanto el exceso como la falta de agua, mejorando la salud de la planta y reduciendo el desperdicio de recursos hídricos. Según Kalluri y Kalidindi (2020, p.128), los sistemas inteligentes de riego, al integrarse con plataformas IoT, representan una solución efectiva para mejorar la eficiencia agrícola, especialmente en cultivos sensibles como los ornamentales, que requieren un cuidado más delicado y constante el uso de sensores de humedad, temperatura, luz permiten mantener condiciones óptimas para el crecimiento vegetal y reduce la dependencia de la supervisión humana.

2.2. Razones para la automatización en el cultivo de plantas

El cultivo de plantas ornamentales tiene varios retos debido a su sensibilidad a los cambios climáticos y a la necesidad de conservar un buen aspecto y estado saludable en las plantas, la automatización es una respuesta efectiva para controlar factores críticos como el riego, la iluminación y el clima sin la intervención humana constante, la implementación de un prototipo domotizado mejora la eficiencia operativa, reduce el margen de error humano y facilita el control del estado de las plantas.

Juntando la Raspberry Pi con sensores de humedad, temperatura, luz, permite desarrollar sistemas inteligentes y adaptables a distintos tipos de plantas y entornos, resulta útil en pequeñas florícolas donde los recursos financieros son limitados, pero la calidad del cultivo

es prioridad. Según SwitchDoc Labs (2018), proyectos de jardinería inteligente basados en Raspberry Pi han demostrado su efectividad en la recolección de datos y en la toma de decisiones autónomas en sistemas agrícolas.

2.3. Beneficios del sistema propuesto

El prototipo propuesto ofrece ventajas, tanto técnicas como económicas. Permite un control preciso y automatizado del proceso de riego, evitando el desperdicio de agua y mejorando la salud de las plantas gracias al riego de nutrientes al garantizar condiciones adecuadas. Además, facilita el monitoreo constante de variables críticas mediante sensores que ayudan a detectar condiciones adversas que afectan el desarrollo del cultivo.

Con Raspberry Pi y Blynk, el sistema se convierte en una solución viable para pequeños productores, facilitando el desarrollo de una agricultura sostenible a un menor costo. Como mencionan Kalluri y Kalidindi (2020), los sistemas inteligentes de riego ayudan a conservar recursos y mejorar el rendimiento del cultivo, lo que se traduce en una mayor productividad y menor impacto ambiental.

2.4. Preguntas de investigación

El cultivo de plantas ornamentales presenta problemas de eficiencia en el manejo de recursos, como el riego y la dosificación de nutrientes, lo que impacta en la calidad y la rentabilidad de los cultivos, por la falta de automatización y monitoreo incrementa los costos de producción y dificulta la administración eficiente de las plantaciones, el problema se enfoca en la ausencia de un sistema automatizado y eficiente para el manejo de huertos florícolas ornamentales.

La falta de comprensión sobre el estado de las plantas influye en su deterioro, y, por medio de la aplicación se entenderá el estado de las plantas, así como sus futuros requisitos. Con ello, surge la pregunta principal para el trabajo.

- ¿Cómo desarrollar un prototipo de aplicación que permita gestionar un ecosistema florícola domotizado, optimizando el manejo de recursos para el cultivo y cuidado de plantas?

Preguntas secundarias.

- ¿Qué tecnologías pueden integrarse para la automatización del riego y la dosificación de nutrientes en un sistema florícola?

- ¿Cómo pueden los sensores monitorear las condiciones del suelo y el entorno para cada tipo de planta ornamental?
- ¿Qué características debe tener la aplicación web para garantizar una gestión eficiente y adaptada a diferentes florícolas?
- ¿Cómo se puede implementar un sistema basado en Raspberry Pi para mantener bajos costos sin comprometer la funcionalidad?

2.4.1. Uso de Raspberry Pi 5

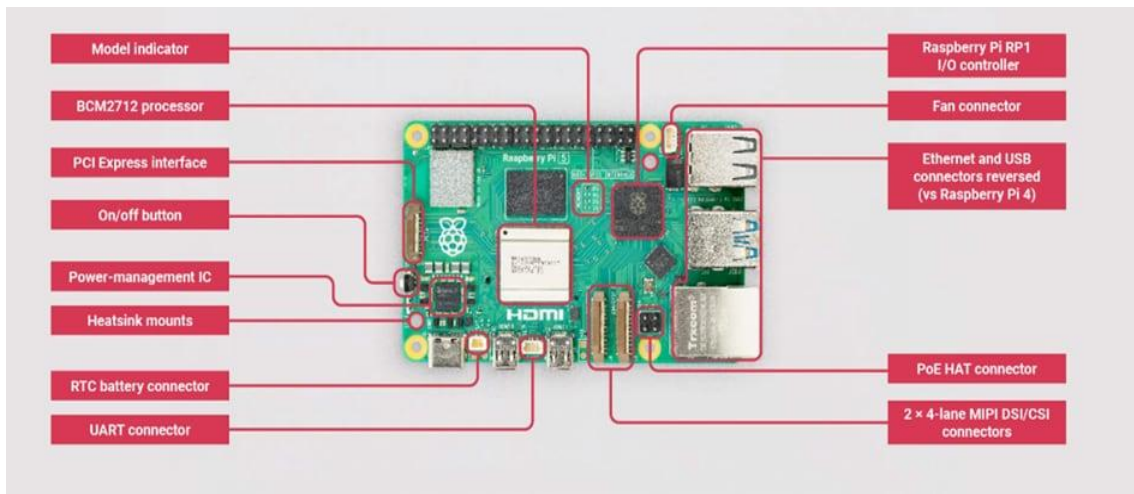
Raspberry Pi 5 es una de las versiones más recientes y potentes de la familia de microcomputadoras Raspberry Pi, ofrece un mayor rendimiento en procesamiento, conectividad y manejo de periféricos.

Cuenta con un procesador de cuatro núcleos ARM Cortex-A76 a 2.4 GHz, su memoria RAM expandible y su conectividad integrada (Wi-Fi, Bluetooth y puertos GPIO), la Raspberry Pi 5 gestiona múltiples sensores y actuadores en tiempo real. Esto resulta útil en sistemas de riego automatizado donde se requiere leer constantemente datos ambientales (como humedad del suelo, temperatura y pH) y accionar dispositivos de forma inmediata y confiable.

Su compatibilidad con sistemas operativos como Raspberry Pi OS y su soporte para lenguajes de programación como Python permiten desarrollar aplicaciones personalizadas de forma sencilla. Buena integración con plataformas IoT como Blynk facilita la visualización remota de datos y el control del sistema desde dispositivos móviles, eliminando la necesidad de interfaces complejas.

La Raspberry Pi 5 representa una solución viable para pequeñas y medianas florícolas que buscan incorporar tecnología en sus procesos sin grandes inversiones. Tal como señala Instructables (2018), esta microcomputadora ha sido utilizada exitosamente en sistemas de jardinería inteligente, permitiendo monitorear condiciones ambientales y activar sistemas de riego de manera eficiente y remota.

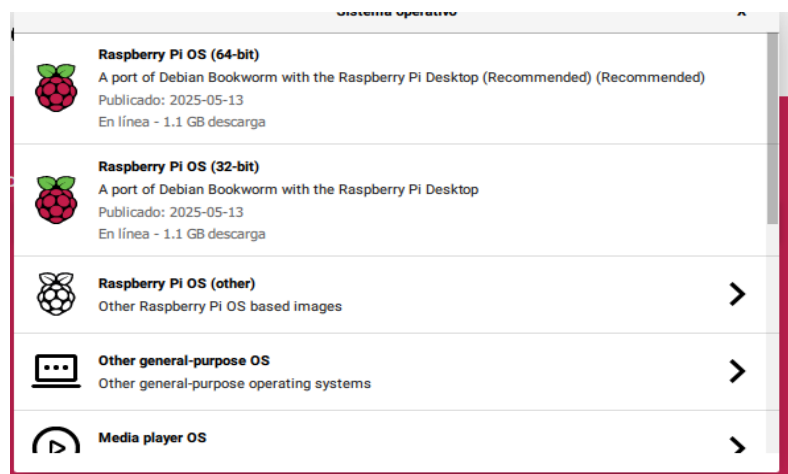
Figura 1 Raspberry pi 5



Nota. El gráfico representa una descripción general del hardware. Tomado de *Raspberry Pi 5 Single Board Computers*. (s. f.). Mouser. <https://www.mouser.ec/new/raspberry-pi/raspberry-pi-5-sbc/>

En este proyecto se optó por utilizar Raspberry Pi OS en su versión de 64 bits, una alternativa estable, ligera y diseñada específicamente para sacar el máximo provecho del hardware de la Raspberry Pi, esta elección no solo simplifica el entorno de desarrollo, sino que también hace más accesible la instalación de herramientas esenciales como Python, Node-RED y distintas librerías para el control de hardware, a esto se suma su buena compatibilidad con interfaces gráficas y el respaldo de una comunidad activa, lo que en conjunto lo convierte en una opción muy adecuada para iniciativas de automatización y monitoreo como la que estamos desarrollando.

Figura 2 Sistema operativo



Nota. El gráfico representa una vista general de los sistemas operativos disponibles para

Raspberry Pi 5. Tomado de *Raspberry Pi 5 Single Board Computers*. (s. f.). Mouser. <https://www.mouser.ec/new/raspberry-pi/raspberry-pi-5-sbc/>

2.4.2.Principios

El prototipo que se plantea tiene como objetivo procesar y compartir información en tiempo real, sin necesidad de que una persona esté interviniendo constantemente. Aplicado al campo, esto permite optimizar el riego, vigilar mejor las condiciones del entorno y tener un control más preciso sobre el cultivo en general.

Madakam, Ramaswamy y Tripathi (2015) explican que el Internet de las Cosas —el IoT es, en realidad, una combinación de varias tecnologías que trabajan juntas: sensores para captar datos del ambiente, conexiones inalámbricas para transmitirlos, espacios donde se almacenan esos datos y sistemas que los analizan. Todo esto mejora la eficiencia, reduce el gasto de recursos y permite reaccionar más rápido cuando el entorno cambia (p.165). En un cultivo de flores, por ejemplo, esta tecnología hace posible monitorear constantemente la humedad del suelo, la temperatura o el pH, y activar el riego solo cuando realmente hace falta. Eso se traduce en menos desperdicio y un manejo mucho más inteligente del cultivo.

Además, el sistema domotizado se apoya en principios de sostenibilidad, simplicidad y accesibilidad tecnológica, que busca demostrar que es posible desarrollar soluciones automatizadas funcionales, económicas y adaptables a distintos escenarios agrícolas, sin necesidad de sistemas complejos o costosos

2.4.3. Limitaciones

Los beneficios de la automatización basada en IoT no se ven restringidos por las limitaciones del prototipo propuesto, las cuales deben tenerse en cuenta durante la implementación, este prototipo se centra exclusivamente en la monitorización y el control ambiental, no considera variables biológicas o fitosanitarias como la detección de plagas, enfermedades o deficiencias nutricionales; estas situaciones, no obstante, requieren intervención humana.

Los componentes electrónicos, como sensores y microcontroladores, son susceptibles a condiciones climáticas extremas (lluvia intensa, calor excesivo, humedad alta o polvo), lo que

podría afectar su funcionamiento o reducir su vida útil si no se aplican medidas de prevención, como encapsulados, carcasas resistentes al agua o ventilación forzada.

Tal como advierte Lal (2020), los sistemas agrícolas basados en tecnología deben considerar la resistencia de sus elementos frente a factores ambientales para garantizar su durabilidad y eficacia operativa.

El sistema depende de una conexión Wi-Fi estable para el envío de datos y visualización remota, lo que limita su funcionalidad en áreas rurales donde el acceso a internet es limitado. el prototipo debe ser considerado como una solución inicial de bajo costo, adaptable que puede ser mejorado o según los requerimientos de cada entorno de cultivo.

2.5. Marco Conceptual

Estos conceptos permiten comprender el funcionamiento del sistema propuesto, así como la interacción entre sus componentes. Por ello, es necesario abordar conceptos como automatización, Internet de las Cosas (IoT), sensores, actuadores y plataformas de visualización remota.

Cada uno de estos elementos posee un rol dentro del sistema: los sensores permiten recopilar datos sobre el estado ambiental, los actuadores ejecutan acciones como activar el riego, y el IoT facilita la comunicación entre dispositivos y la interfaz móvil se considera el uso de la Raspberry Pi 5 como microcontrolador central por su capacidad de integración, procesamiento y conectividad.

2.2.1. Automatización

La automatización en la agricultura ha hecho posible controlar y monitorear diferentes procesos del campo con muy poca intervención humana. La idea es simple pero potente: aprovechar mejor los recursos, hacer que las tareas sean más eficientes y lograr una mayor precisión en actividades clave como el riego, la fertilización o el control de las condiciones del entorno. Todo esto no solo aligera el trabajo diario, sino que también mejora los resultados a largo plazo.

En la floricultura, la automatización importancia por lo delicadas que pueden ser las plantas ornamentales. Son muy sensibles con pequeñas variaciones en temperatura, humedad

de la tierra o luz llegan a afectar su crecimiento. Implementar sistemas automatizados permite mantener condiciones más controladas para su desarrollo, reduciendo los riesgos por errores humanos.

Beneficios clave de la automatización agrícola:

- **Eficiencia en el uso de recursos:** La automatización permite aplicar insumos como agua y fertilizantes de manera precisa, evitando desperdicios y reduciendo costos operativos
- **Monitoreo en tiempo real:** Los sistemas automatizados recopilan y analizan datos constantemente, facilitando la toma de decisiones informadas y oportunas.
- **Reducción de la carga laboral:** Al automatizar tareas repetitivas, se libera tiempo para que los agricultores se enfoquen en actividades estratégicas.
- **Mejora en la calidad del cultivo:** Mantener condiciones ambientales estables contribuye al desarrollo saludable de las plantas, aumentando su calidad y valor en el mercado.

2.2.2. Internet de las Cosas (IoT)

El Internet de las Cosas permite la conexión de dispositivos físicos a través de redes digitales mediante Internet, tiene el objetivo de recolectar, transmitir y procesar datos de manera autónoma, estos dispositivos están diseñados para interactuar entre sí y con sistemas en la nube, facilitando el control y la supervisión de procesos.

En el contexto agrícola, el IoT ha permitido transformar la forma en que se monitorean y gestionan las actividades productivas, por medio de la implementación de sensores distribuidos en los campos o invernaderos, es posible obtener información en tiempo real sobre variables críticas como la humedad del suelo, la temperatura ambiental, la luminosidad. Esta información es visualizada por el productor desde una aplicación móvil o una plataforma web, permitiendo tomar decisiones rápidas y precisas para el cuidado del cultivo.

Según Madakam, Ramaswamy y Tripathi (2015), el IoT tiene el potencial de revolucionar el sector agrícola, ya que permite una gestión más eficiente de los recursos y reduce la dependencia de la supervisión manual. La capacidad de generar alertas automáticas, activar sistemas como el riego o la ventilación, y almacenar datos históricos para su análisis,

abre las puertas al uso de inteligencia artificial y modelos predictivos que optimicen aún más los procesos productivos.

En el caso del proyecto, el IoT se implementa mediante una Raspberry Pi conectada a sensores ambientales y una interfaz de monitoreo remoto a través de la plataforma Blynk. Esta arquitectura permite automatizar el riego y visualizar el estado del ecosistema florícola en tiempo real, brindando al usuario un control completo sobre el entorno desde cualquier lugar con acceso a internet.

2.2.3. Sensores y actuadores.

En cualquier sistema de automatización, los sensores y actuadores son esenciales. Por qué, los sensores facilitan la recopilación de datos basados en variables como la humedad, la temperatura, la luz o el pH. Para activar el sistema de agua, se utilizan botones para activado manual y la automatización según los valores de sensores de humedad y temperatura. Tomando en cuenta a ResearchGate (2019), los sensores de humedad detectan los niveles de agua en el suelo y regulan automáticamente el riego para un suministro óptimo. Los actuadores, como las válvulas de riego, se activan basándose en los datos recopilados por los sensores, mejorando así la precisión del sistema y reduciendo el desperdicio de recursos.

2.3. Red

La red empleada se fundamenta en una estructura centralizada con conexión WiFi, en la que la Raspberry Pi desempeña el papel de nodo maestro, dado que, recolecta datos directamente desde sensores conectados por cable, y controla actuadores mediante sus pines GPIO. Esta arquitectura facilita el mantenimiento, disminuye costos y permite una implementación efectiva en áreas agrícolas confinadas.

Como señala Upton y Halfacree (2019), la Raspberry Pi cuenta con interfaces integradas de red que permiten establecer comunicación inalámbrica efectiva en aplicaciones domóticas. Además, la conectividad WiFi garantiza que los datos sean enviados a la nube o a dispositivos móviles para su visualización y análisis en tiempo real.

2.4. Blynk.cloud IoT

Para el prototipo se utilizó la plataforma Blynk.cloud IoT para la visualización remota y el control del sistema de automatización de riego y los diferentes datos de los sensores. Blynk.cloud es una plataforma perfecta para IoT que permite crear de manera rápida interfaces

gráficas personalizadas para aplicaciones móviles y web, facilitando así la interacción con dispositivos conectados en nuestro caso la Raspberry Pi.

Blynk ofrece ventajas gracias a su gran capacidad en integrar sensores y actuadores mediante un sistema de Virtual Pins, lo que facilita el envío y la recepción de datos en tiempo real, cuenta con una arquitectura en nube y permiten a los usuarios monitorear y controlar el sistema desde cualquier lugar, sin necesidad de configurar redes ni de mantener servidores propios.

Blynk cuenta con una buena compatibilidad con la Raspberry Pi 5 es directa y eficiente, lo que facilita la integración con scripts en Python, como los que se emplean en este proyecto para la lectura de sensores y la gestión del sistema de riego. A través del su entorno intuitivo de Blynk.cloud, es posible diseñar dashboards personalizados según nuestra necesidad como humedad del suelo, la temperatura, la humedad ambiental, la luminosidad y el nivel de agua. Además, se habilita el control manual y automático de los actuadores, como las bombas de agua y las electroválvulas.

El uso de Blynk IoT aporta al sistema un enfoque moderno y accesible, eliminando la necesidad de desarrollar aplicaciones móviles desde cero y ofreciendo una experiencia de usuario completa tanto en dispositivos móviles como en navegadores web.

3. CAPÍTULO III: DESARROLLO

En este capítulo se detalla el proceso de construcción del prototipo del sistema de riego automatizado y monitoreo ambiental desarrollado, se describe la arquitectura general del sistema, los elementos físicos y lógicos utilizados, y cómo se integraron para cumplir con los objetivos planteados, se presentan las capacidades de automatización, control remoto y monitoreo en tiempo real, junto con una visión de las posibilidades de mejora o expansión a futuro.

3.1. Planificación técnica

3.1.1. Descripción inicial del problema

En la floricultura tradicional, el riego y la falta de monitoreo provocan problemas en el uso del agua, lo que dificulta la detección de cambios que afectan el desarrollo de las plantas

3.1.2. Objetivos técnicos del sistema

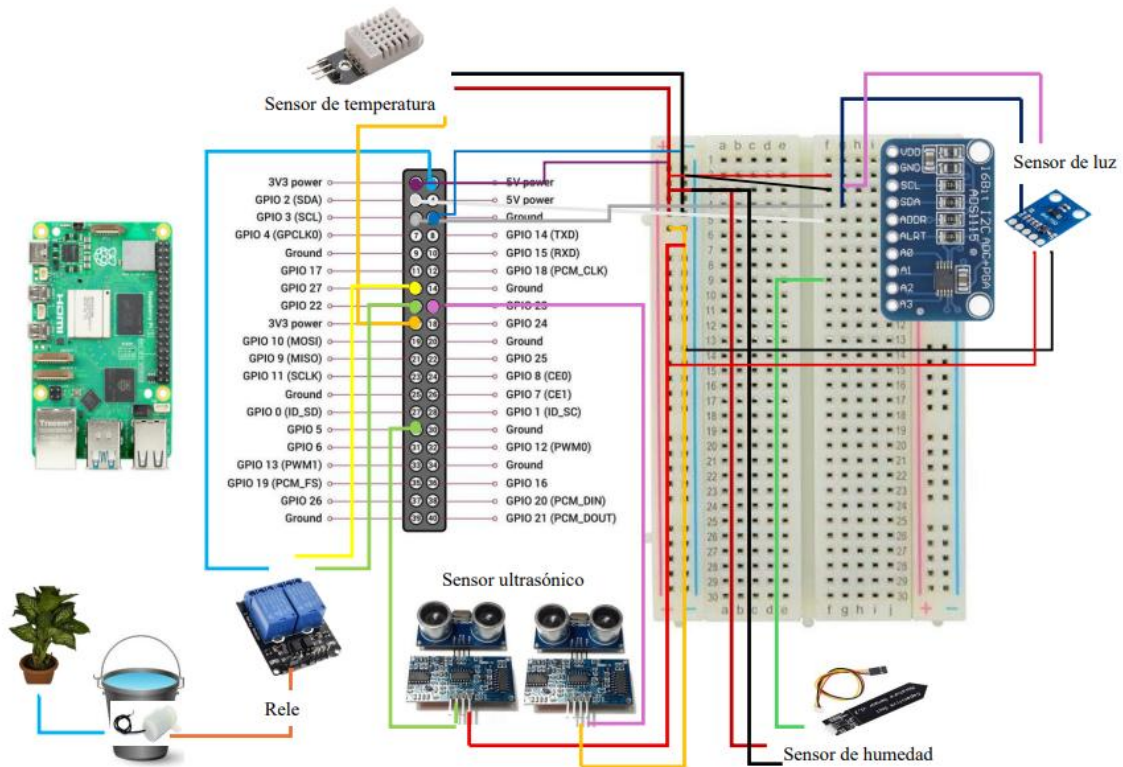
El desarrollo del sistema técnico se basó en tres objetivos fundamentales que orientaron las decisiones de diseño e implementación del prototipo son los siguientes:

- Diseñar un sistema de adquisición de datos a través de sensores ambientales.
- Automatizar el riego de acuerdo con valores críticos de humedad y pH.
- Implementar una interfaz móvil para monitoreo y control remoto usando Blynk IoT.

3.1.3. Elección de la arquitectura general

El diseño del prototipo está basado en la necesidad de automatizar el riego de un espacio cultivado mediante un sistema capaz de tomar decisiones en función de datos ambientales, se utilizó una Raspberry Pi 5 como unidad central de procesamiento, aprovechando su capacidad para manejar múltiples sensores y módulos a través de sus pines GPIO y puertos digitales, el sistema no solo ejecuta acciones como el encendido de una bomba de agua, sino que también permite al usuario monitorear las condiciones de su cultivo desde una aplicación móvil o interfaz web gracias a la plataforma Blynk IoT.

Figura 3 Prototipo cableado



Nota. El gráfico muestra la arquitectura de automatización, donde se integran sensores, actuadores y una unidad central de procesamiento (Raspberry Pi), con el objetivo de controlar procesos automáticamente. Tomado de Rojas, F. (2025). *Arquitectura de automatización*

3.1.4. Listado de componentes de hardware y software

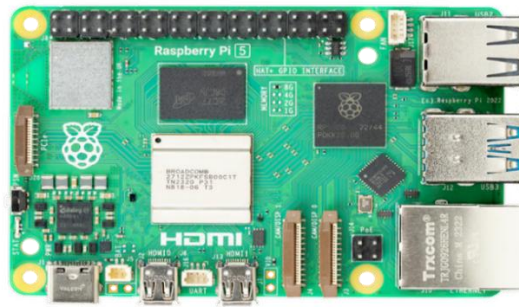
El sistema fue construido a partir de una selección estratégica de componentes electrónicos que permiten la lectura de parámetros ambientales, el control de riego y la conectividad con una plataforma de monitoreo.

3.1.5. Hardware

3.1.5.1. Raspberry Pi 5

La unidad de control central del sistema es una Raspberry Pi 5 (versión de 4 GB de RAM), una minicomputadora de placa única que integra un procesador ARM Cortex-A76 de 64 bits, conectividad Wifi y Bluetooth, con pines GPIO, por tal motivo fue elegida por su capacidad de procesamiento suficiente para ejecutar lógicas complejas y gestionar múltiples sensores.

Figura 4 Raspberry pi 5



Nota. El gráfico representa el diseño superior del circuito impreso (PCB Top Layout) de la Raspberry Pi 5. Tomado de *Raspberry Pi 5 Single Board Computers*. (s. f.). Mouser. <https://www.mouser.ec/new/raspberry-pi/raspberry-pi-5-sbc/>

3.1.5.2. Sensor de humedad del suelo capacitivo - APM-969

El sensor de humedad del suelo, que proporciona una señal analógica proporcional al nivel de humedad del sustrato, este tipo de sensor se seleccionó por su resistencia a la corrosión, lo que lo hace adecuado para entornos agrícolas y garantiza una mayor durabilidad frente a sensores tradicionales, cuenta con una buena integración permite implementar estrategias de riego basadas en el nivel real de humedad detectado en el suelo.

Figura 5 Sensor de humedad del suelo capacitivo - APM-969

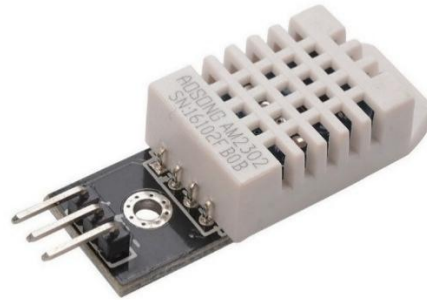


Nota. El gráfico representa una vista general del sensor de humedad del suelo capacitivo modelo APM-969, utilizado para medir la humedad del sustrato mediante cambios en la capacitancia. Tomado de *Humedad de Suelo Capacitivo – Apm*. (s. f.). APM Electronics. <https://apmelectronica.com/producto/humedad-de-suelo-capacitivo/>

3.1.5.3. Sensor de temperatura y humedad DHT22

Para la monitorización de variables climáticas, se incluye un sensor DHT22, capaz de medir la temperatura y la humedad ambiental con una buena precisión y un bajo coste. Este sensor aporta información valiosa sobre las condiciones ambientales, permitiendo correlacionar los patrones de riego con el clima y optimizar así el uso de recursos hídricos.

Figura 6 Sensor de temperatura y humedad DHT22

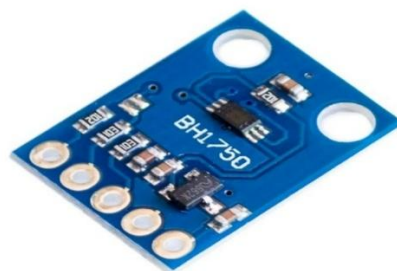


Nota. El gráfico representa un sensor DHT22, utilizado para medir la temperatura y la humedad relativa del entorno. Tomado de *Modulo DHT22 Temperatura y Humedad.* (s. f.). TIENDATEC.ES. <https://www.tiendatec.es/maker-zone/modulos/789-modulo-dht22-temperatura-y-humedad-8472496009805.html>

3.1.5.4. Sensor de luz BH1750

El sistema también incorpora un sensor BH1750, un sensor digital de luz ambiental que comunica sus lecturas en unidades de lux mediante la interfaz I2C. Su inclusión permite conocer el nivel de luminosidad, una variable importante para determinados tipos de plantas y que puede influir en las necesidades hídricas del cultivo.

Figura 7 Sensor de luz BH1750



Nota. El gráfico muestra el diseño del sensor BH1750, utilizado para medir la intensidad de luz

ambiental en unidades de lux. Tomado de *Sensor de intensidad de luz GY302 BH1750*. (s. f.). Novatronic. <https://novatronicec.com/index.php/product/sensor-de-intensidad-de-luz-gy302-bh1750/>

3.1.5.5. ¿Por qué no se utilizó el sensor de pH?

No se utilizó un sensor de pH de tierra por su alto costo, si es importante medir el pH del suelo puede aportar información valiosa para el manejo agrícola, este sensor es más costosos, delicados y requieren un mantenimiento específico para que funcione correctamente, el objetivo del proyecto es ofrecer una solución automatizada accesible se priorizó el uso de sensores económicos, como los de humedad y temperatura, que permiten cumplir con las funciones principales de riego automatizado sin elevar demasiado el presupuesto.

3.1.5.6. Sensor ultrasónico HC-SR04

Se han implementado dos sensores ultrasónicos HC-SR04, estos utilizan pulsos de ultrasonido para medir distancia, lo que permite determinar con precisión el nivel de agua este presente en los depósitos, es importante para evitar el funcionamiento en caso de que se seque las reservas de agua.

Figura 8 Sensor ultrasónico HC-SR04



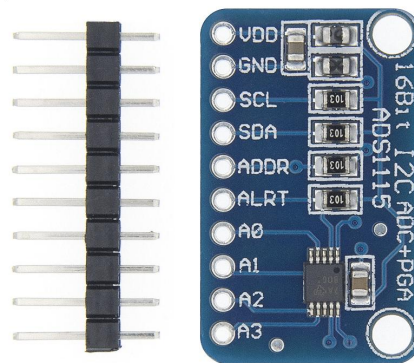
Nota. El gráfico representa un sensor ultrasónico HC-SR04, utilizado para medir la distancia entre el sensor y un objeto mediante ondas de sonido. Tomado de Diosdado, R. (2014). *Sensor de ultrasonidos HC-SR04*. Zona Maker. <https://www.zonamaker.com/ultrasonido-hc-sr04>

3.1.5.7. Módulo ADS1115

La Raspberry Pi 5 no cuenta con entradas analógicas nativas, por ente se ha incorporado un módulo ADS1115, un conversor analógico-digital de 16 bits que permite la lectura precisa

de los sensores analógicos (humedad del suelo, temperatura y luz), permitiendo la comunicación mediante I2C facilita la integración y reduce el número de cables requeridos.

Figura 9 Módulo ADS1115

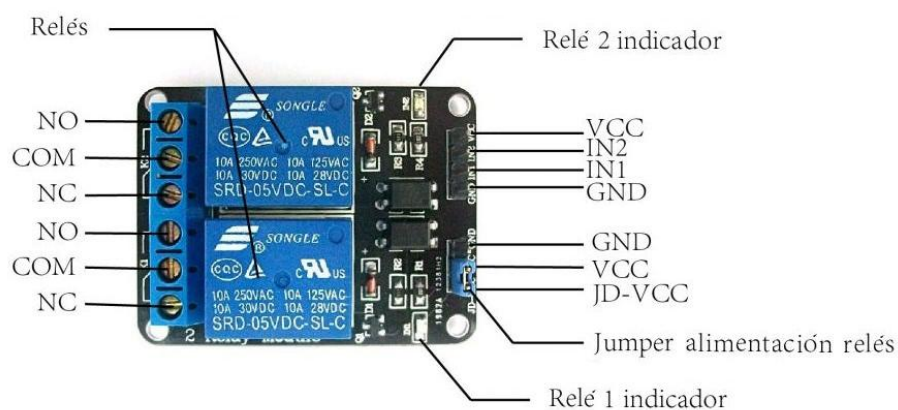


Nota. El gráfico representa un módulo ADS1115, utilizado para convertir señales analógicas a digitales con alta precisión. Tomado de *MODULO ADC 4 CANALES ADS1115*. (2024). Tecnoconciencia. <https://tecnoconciencia.com/producto/modulo-adc-4-canales-ads1115-16bit-i2c/>

3.1.5.8. Módulo relé de 2 canales

El sistema de actuadores está compuesto por un módulo de relé de 4 canales, que permite controlar dispositivos de mayor carga, como bombas de agua o electroválvulas, mediante señales digitales de la Raspberry Pi. En este proyecto, el relé permite la activación de una bomba de agua de 5V y de una electroválvula, responsables de gestionar el flujo de agua hacia las plantas, tanto para riego normal como para riego con nutrientes.

Figura 10 Módulo relé de 2 canales



Nota. El gráfico representa un módulo relé de 2 canales, el cual permite controlar dispositivos eléctricos mediante señales de baja potencia. Tomado de *Módulo relé de 2 canales*. (s. f). Teslatronica Sumador. <https://sumador.com/products/modulo-rele-de-2-canales>

3.1.5.9. Bomba de agua de 5V

El sistema incluye dos bombas de agua de 5V, ambos actuadores controlados a través del módulo de relé. Estos elementos permiten automatizar el proceso de riego, activándose de forma automática según las condiciones del suelo y el entorno, o manualmente a través de la aplicación Blynk.

Figura 11 Bomba de agua de 5V



Nota. El gráfico representa una bomba de agua de 5V, diseñada para el movimiento de líquidos en pequeños proyectos. Tomado de *MINI BOMBA DE AGUA SUMERGIBLE 5V*. (s. f.). Grupo Electrostore. <https://grupoelectrostore.com/shop/motores/bombas-para-agua/mini-bomba-de-agua-sumergible-5v/>

3.1.5.10. Fuente 5V

Para garantizar la portabilidad y operatividad del sistema utiliza la fuente de 5V. para el correcto funcionamiento de las Raspberry Pi 5

Figura 12 Fuente 5V

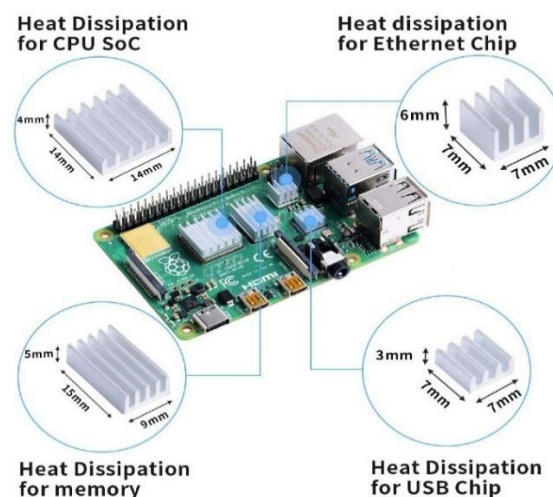


Nota. El gráfico representa una fuente de 5V. Tomado de *AV Electronics*. (s. f.). *Fuente Raspberry Pi*. AV Electronics. <https://avelectronics.cc/producto/fuente-raspberry-pi/>

3.1.5.11. Disipadores

Con disipadores instalados en la Raspberry Pi, con el fin de mantener una temperatura adecuada durante su funcionamiento continuo, previniendo el sobrecalentamiento del procesador y mejorando la fiabilidad del sistema.

Figura 13 Disipadores de calor



Nota. El gráfico representa disipadores, componentes que ayudan a reducir la temperatura de dispositivos electrónicos. Tomado de *Disipadores de calor para Raspberry*. (s. f.). Amazon. <https://www.amazon.com.mx/GeekPi-Raspberry-Model-alimentaci%C3%B3n->

disipadores/dp/B07XDXMN8F

3.1.5.12. Tarjeta microSD de 32 GB

Finalmente, se emplea una tarjeta microSD de 32 GB que aloja el sistema operativo, las bibliotecas y los scripts del sistema de automatización, proporcionando almacenamiento suficiente para el correcto funcionamiento de la solución.

Figura 14 Tarjeta microSD de 32 GB



Nota. El gráfico representa una tarjeta microSD de 32 GB para almacenamiento de datos. Tomado de *Micro SD de 32GB Kingston*. (2025). LA VICTORIA. <https://lavictoria.ec/producto/tarjeta-de-memoria-micro-sd-de-32gb-kingston/>

3.1.6. SOFTWARE

3.1.6.1. Raspberry Pi OS (64-bit)

Como sistema operativo se utiliza Raspberry Pi OS (64-bit), una distribución basada en Debian, que ofrece un entorno estable y bien documentado para el desarrollo en Raspberry Pi. Se optó por la versión con escritorio para facilitar las tareas de desarrollo y pruebas.

3.1.6.2. Python 3.11

El lenguaje de programación principal es Python 3.11, elegido por su amplia compatibilidad con bibliotecas de hardware, su legibilidad y su gran comunidad de soporte. Python permite controlar los sensores, gestionar las lógicas de riego y establecer la comunicación con plataformas externas como Blynk.

3.1.6.3. Librerías de Python

Entre las bibliotecas de Python empleadas destacan blynklib (para la interacción con la plataforma Blynk), gpiozero y RPi.GPIO (para el control de pines GPIO), smbus2 (para la comunicación I2C) y otras específicas para cada sensor.

3.1.6.4. Editores de código: VS Code y Nano

El desarrollo de los scripts se realizó utilizando Visual Studio Code (VS Code) y Nano, editores de texto ligeros y flexibles que permiten trabajar directamente en la Raspberry Pi o de forma remota.

3.1.6.5. Terminal y SSH

Finalmente, se utilizaron herramientas de terminal como SSH, que permitieron el acceso remoto a la Raspberry Pi durante el desarrollo y pruebas, facilitando la gestión y actualización del sistema sin necesidad de conexión directa de monitor o teclado.

3.1.7. Justificación de uso de Blynk Iot

La plataforma Blynk IoT fue seleccionada debido a la capacidad de solución de monitoreo y control remoto para nuestro prototipo debido a que ofrece facilidad de integración con la Raspberry Pi, debido a su enfoque con dispositivos IoT, y su compatibilidad multiplataforma (web y móvil). Esta herramienta permite la creación de interfaces gráficas intuitivas sin necesidad de conocimientos avanzados en desarrollo móvil, lo cual agiliza el diseño de aplicaciones funcionales en entornos de automatización.

Con un dashboard intuitivo podemos visualizar variables críticas como la humedad del suelo, temperatura, luminosidad ambiental y nivel del tanque, además de habilitar el control manual y automático de actuadores. Estas interacciones se gestionan a través de pines virtuales, lo que permite una comunicación eficiente entre el sistema físico y la interfaz digital.

Blynk IoT ofrece una modalidad gratuita (*Free Tier*), adecuada para pruebas y proyectos educativos que permite un dispositivo, hasta 10 datastreams, y dashboards limitados. Para aplicaciones más avanzadas, la plataforma cuenta con planes Plus, Pro y Enterprise, los cuales incluyen funcionalidades como:

- Gestión de múltiples dispositivos y usuarios.

- Historial de datos y visualización de tendencias.
- Automatización de eventos complejos (basados en condiciones o tiempo).
- Integración con APIs externas (REST, Webhooks, Node-RED, etc.).
- Control de acceso por roles (admin, técnico, operador).
- Reportes automatizados y gestión remota a escala.

Estas capacidades convierten a Blynk en una herramienta escalable, capaz de crecer junto con el sistema, desde una fase prototipo hasta una eventual implementación en producción o comercial. Además, al estar basado en la nube, elimina la necesidad de configurar servidores locales, lo que reduce costos de infraestructura y mantenimiento.

3.2 Diseño del Software

3.2.1. Estructura del código

El sistema fue desarrollado en Python 3.11, trabajando con librerías que permiten la interacción con los sensores, la gestión de los pines GPIO y la comunicación con la plataforma Blynk IoT.

El script denominado `riego_blynk.py` constituye el núcleo funcional del sistema, el propósito es gestionar de manera integral los datos desde los sensores, enviar esta información en tiempo real a la plataforma Blynk.cloud, recibir comandos de control manual desde la aplicación Blynk.cloud y ejecutar la lógica de automatización para el sistema de riego.

El proyecto hizo uso de diversas librerías especializadas, como la librería `blynklib` permite la comunicación fluida con la plataforma Blynk.cloud, facilitando tanto el envío de datos como la recepción de comandos, la librería `RPi.GPIO` se empleó para el control directo de los pines GPIO de la Raspberry Pi, permitiendo gestionar los actuadores del sistema.

Para la interacción con sensores `gpiozero` permite la comunicación como los HC-SR04, utilizados para medir distancias mediante ultrasonido para la lectura de sensores analógicos, como los de humedad del suelo, se utiliza la librería `adafruit-circuitpython-ads1x15`, que opera en conjunto con el módulo convertidor ADS1115.

La medición de la luminosidad ambiental se realiza mediante el sensor BH1750, utilizando la librería `adafruit-circuitpython-bh1750`, se realizó lo mismo con la lectura de

temperatura y humedad ambiental proveniente del sensor DHT22 se gestiona a través de la librería `adafruit-circuitpython-dht`.

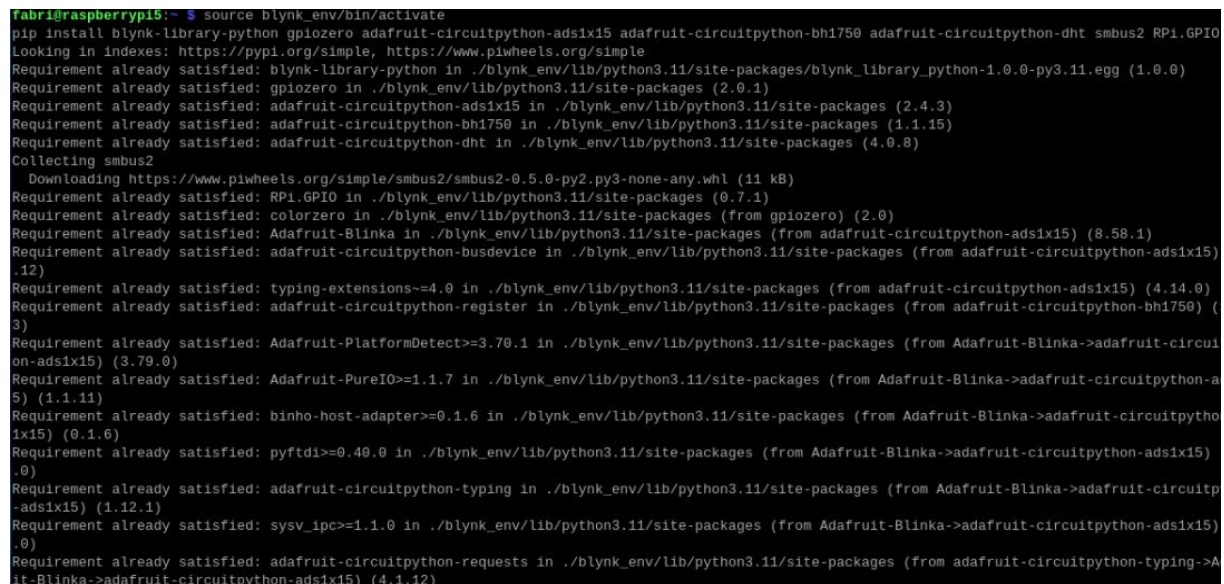
La librería estándar `time` es utilizada para controlar los tiempos de espera en las lecturas de los sensores y para organizar el ciclo de ejecución del sistema, asegurando un funcionamiento estable y sincronizado.

Estas librerías se instalaron previamente mediante el gestor de paquetes `pip` en un entorno virtual de Python:

`source blynk_env/bin/activate`

`pip install blynk-library-python gpiozero adafruit-circuitpython-ads1x15 adafruit-circuitpython-bh1750 adafruit-circuitpython-dht smbus2 RPi.GPIO`

Figura 15 Instalación de Blynk_env



```
fabri@raspberrypi5:~$ source blynk_env/bin/activate
pip install blynk-library-python gpiozero adafruit-circuitpython-ads1x15 adafruit-circuitpython-bh1750 adafruit-circuitpython-dht smbus2 RPi.GPIO
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Requirement already satisfied: blynk-library-python in ./blynk_env/lib/python3.11/site-packages/blynk_library_python-1.0.0-py3.11.egg (1.0.0)
Requirement already satisfied: gpiozero in ./blynk_env/lib/python3.11/site-packages (2.0.1)
Requirement already satisfied: adafruit-circuitpython-ads1x15 in ./blynk_env/lib/python3.11/site-packages (2.4.3)
Requirement already satisfied: adafruit-circuitpython-bh1750 in ./blynk_env/lib/python3.11/site-packages (1.1.15)
Requirement already satisfied: adafruit-circuitpython-dht in ./blynk_env/lib/python3.11/site-packages (4.0.8)
Collecting smbus2
  Downloading https://www.piwheels.org/simple/smbus2/smbus2-0.5.0-py2.py3-none-any.whl (11 kB)
Requirement already satisfied: RPi.GPIO in ./blynk_env/lib/python3.11/site-packages (0.7.1)
Requirement already satisfied: colorzero in ./blynk_env/lib/python3.11/site-packages (from gpiozero) (2.0)
Requirement already satisfied: Adafruit-Blinka in ./blynk_env/lib/python3.11/site-packages (from adafruit-circuitpython-ads1x15) (8.58.1)
Requirement already satisfied: adafruit-circuitpython-busdevice in ./blynk_env/lib/python3.11/site-packages (from adafruit-circuitpython-ads1x15) (0.12)
Requirement already satisfied: typing-extensions==4.0 in ./blynk_env/lib/python3.11/site-packages (from adafruit-circuitpython-ads1x15) (4.14.0)
Requirement already satisfied: adafruit-circuitpython-register in ./blynk_env/lib/python3.11/site-packages (from adafruit-circuitpython-bh1750) (3)
Requirement already satisfied: Adafruit-PlatformDetect>=3.70.1 in ./blynk_env/lib/python3.11/site-packages (from Adafruit-Blinka->adafruit-circuitpython-ads1x15) (3.79.0)
Requirement already satisfied: Adafruit-PureIO>=1.1.7 in ./blynk_env/lib/python3.11/site-packages (from Adafruit-Blinka->adafruit-circuitpython-ads1x15) (1.1.11)
Requirement already satisfied: binho-host-adapter>=0.1.6 in ./blynk_env/lib/python3.11/site-packages (from Adafruit-Blinka->adafruit-circuitpython-ads1x15) (0.1.6)
Requirement already satisfied: pyftdi>=0.40.0 in ./blynk_env/lib/python3.11/site-packages (from Adafruit-Blinka->adafruit-circuitpython-ads1x15) (0.40.0)
Requirement already satisfied: adafruit-circuitpython-typing in ./blynk_env/lib/python3.11/site-packages (from Adafruit-Blinka->adafruit-circuitpython-ads1x15) (1.12.1)
Requirement already satisfied: sysv_ipc>=1.1.0 in ./blynk_env/lib/python3.11/site-packages (from Adafruit-Blinka->adafruit-circuitpython-ads1x15) (1.0.2)
Requirement already satisfied: adafruit-circuitpython-requests in ./blynk_env/lib/python3.11/site-packages (from adafruit-circuitpython-typing->Adafruit-Blinka->adafruit-circuitpython-ads1x15) (4.1.12)
```

Nota. Instalación y verificación de dependencias en entorno virtual para la ejecución del sistema de monitoreo y riego automatizado. Tomado de Rojas, F. (2025). *Arquitectura de automatización*

3.2.2. Inicialización de sensores

El código comienza con la inicialización de los diferentes sensores conectados a la Raspberry Pi 5. Los sensores de humedad del suelo, que generan señales analógicas, son leídos a través del módulo conversor ADS1115, el cual se comunica mediante el bus I2C. Esta conversión es esencial, ya que la Raspberry Pi no cuenta con entradas analógicas nativas.

El sensor de luz ambiental BH1750 también utiliza la interfaz I2C para transmitir datos, lo que facilita su integración con el sistema y permite obtener mediciones precisas de la luminosidad en el entorno.

El sensor DHT22, encargado de medir la temperatura y la humedad ambiental, se conecta directamente a un pin GPIO configurado para recibir datos digitales, para obtener información confiable de la temperatura del ambiente del cultivo.

Finalmente, los sensores ultrasónicos HC-SR04, utilizados para medir el nivel de agua en los tanques, operan mediante pines GPIO configurados tanto como entradas como salidas. A través de impulsos de ultrasonido y su correspondiente medición de tiempo, el sistema puede determinar con exactitud la cantidad de agua disponible.

Esta configuración integral garantiza una lectura precisa y confiable de las variables ambientales y de proceso, lo que es clave para el correcto funcionamiento del sistema de automatización del riego.

Figura 16 Configuración inicial del sistema y parámetros de riego automatizad

a)

```
import time
import board
import busio
import digitalio
import adafruit_bh1750
import adafruit_ads1x15.ads1115 as ADS
from adafruit_ads1x15.analog_in import AnalogIn
import adafruit_dht
import BlynkLib
from gpiozero import DistanceSensor
from digitalio import DigitalInOut, Direction

# --- CONFIGURACIÓN ---

BLYNK_AUTH = "d7I3IKaeiMpzaCZargnrN1X4tFPViike" # Token de tu SmartGarden1
blynk = BlynkLib.Blynk(BLYNK_AUTH)

# Umbrales
UMBRAL_HUMEDAD = 20000
UMBRAL_PH_MIN = 5.5
UMBRAL_PH_MAX = 7.5
UMBRAL_LUZ = 50 # Lux mínimo para permitir riego
UMBRAL_TEMP_MAX = 30 # Temperatura mínima para activar riego automático

# Bus I2C
i2c = busio.I2C(board.SCL, board.SDA)

# Sensor luz BH1750
sensor_luz = adafruit_bh1750.BH1750(i2c)

# ADS1115
ads = ADS.ADS1115(i2c)
chan_humedad = AnalogIn(ads, ADS.P0)
chan_ph = AnalogIn(ads, ADS.P1)

# DHT22
dht_device = adafruit_dht.DHT22(board.D4)
```

b)

```
# RELÉS
relay_bomba_normal = DigitalInOut(board.D17)
relay_bomba_normal.direction = Direction.OUTPUT
relay_bomba_normal.value = True

relay_bomba_nutrientes = DigitalInOut(board.D27)
relay_bomba_nutrientes.direction = Direction.OUTPUT
relay_bomba_nutrientes.value = True

relay_valvula = DigitalInOut(board.D22)
relay_valvula.direction = Direction.OUTPUT
relay_valvula.value = True

# Modo manual
modo_manual_normal = False
modo_manual_nutrientes = False

# Sensores ultrasónicos
sensor1 = DistanceSensor(echo=6, trigger=5)
sensor2 = DistanceSensor(echo=24, trigger=23)
```

Nota. El gráfico a estructura del código de automatización implementado en Raspberry Pi, incluyendo la inicialización de sensores de humedad, luz, temperatura y nivel de agua, así como la configuración de los relés para el control de bombas y válvulas. b umbrales clave para activar el riego de forma automática. Tomado de Rojas, F. (2025). *Arquitectura de automatización*

3.2.3. Lectura de valores y bucle principal

Durante la ejecución continua del sistema, el código entra en un bucle principal donde se realiza la lectura periódica de todos los sensores conectados. Esta rutina permite obtener un conjunto actualizado de variables clave que describen el estado actual del entorno y del sistema de riego.

Los valores obtenidos del sensor de la humedad del suelo, medida como un valor analógico leído a través del conversor ADS1115, de la misma forma, la luminosidad ambiental se obtiene en unidades de lux mediante el sensor BH1750, proporcionando información útil sobre las condiciones de luz que afectan a las plantas.

Por su parte, la temperatura y la humedad relativa del ambiente son medidas por el sensor DHT22, cuyos datos digitales son procesados por el sistema para su monitoreo en tiempo real.

El sistema calcula el nivel de agua disponible en los tanques mediante la medición se basa en los datos de distancia capturados por los sensores ultrasónicos HC-SR04, lo que nos permitió conocer con precisión la cantidad de agua restante, el siguiente código ilustra cómo se lleva a cabo la lectura de estos valores y su posterior envío a la plataforma Blynk.cloud, donde los usuarios pueden visualizar la información de forma remota y en tiempo real.

Figura 17 Envío de datos a Blynk

```
# Enviar datos a Blynk
blynk.virtual_write(0, humedad)
blynk.virtual_write(1, luz)
blynk.virtual_write(2, ph_value)
if temp is not None:
    blynk.virtual_write(3, temp)
if hum_amb is not None:
    blynk.virtual_write(6, hum_amb)
blynk.virtual_write(7, distancia1)
blynk.virtual_write(8, distancia2)
```

Nota. El gráfico muestra el fragmento del código utilizado para enviar datos de los sensores hacia la plataforma Blynk, incluyendo variables como humedad, luz, pH, temperatura, humedad ambiental y niveles de agua. Esta interacción permite el monitoreo remoto en tiempo real del sistema de riego domótico. Tomado de Rojas, F. (2025). *Arquitectura de automatización*

3.2.4. Recepción de comandos manuales

El sistema permite, además, el control manual del riego desde la aplicación Blynk, mediante botones virtuales configurados en los Virtual Pins V4 y V5.

Figura 18 Activadores de bombas normal y nutrientes



Nota. El gráfico muestra la interfaz de control manual implementada en la plataforma Blynk, donde el usuario puede activar o desactivar remotamente las bombas de riego normal y de nutrientes. Esta función permite supervisión y acción directa sobre el sistema automatizado. Tomado de Rojas, F. (2025). *Arquitectura de automatización*

El siguiente fragmento de código muestra cómo se gestiona la recepción de estos comandos:

Figura 19 Control manual de riego por pines virtuales en Blynk

```
@blynk.on("V4")
def v4_write_handler(value):
    global modo_manual_normal
    if int(value[0]) == 1:
        modo_manual_normal = True
        print("Modo manual: bomba normal activada")
    else:
        modo_manual_normal = False
        relay_bomba_normal.value = True
        relay_valvula.value = True
        print("Modo manual: bomba normal desactivada")

@blynk.on("V5")
def v5_write_handler(value):
    global modo_manual_nutrientes
    if int(value[0]) == 1:
        modo_manual_nutrientes = True
        print("Modo manual: bomba nutrientes activada")
    else:
        modo_manual_nutrientes = False
        relay_bomba_nutrientes.value = True
        relay_valvula.value = True
        print("Modo manual: bomba nutrientes desactivada")
```

Nota. El gráfico muestra el manejo de comandos manuales recibidos desde la aplicación Blynk, mediante los pines virtuales V4 y V5. Estas funciones permiten activar o desactivar las bombas de riego normal y de nutrientes según las acciones del usuario, integrando el control manual dentro del sistema automatizado. Tomado de Rojas, F. (2025). *Arquitectura de automatización*

3.2.5. Ejecución en la Raspberry Pi y pruebas de consola

La ejecución del sistema de riego en la Raspberry Pi 5 se realiza dentro de un entorno virtual de Python, creado específicamente para el proyecto, este entorno nos permite gestionar de manera ordenada las dependencias y versiones de librerías necesarias, evitando conflictos con otros proyectos o con el sistema base de la Raspberry Pi 5.

Para poner en marcha el sistema, se siguen unos pasos básicos: primero se activa el entorno virtual, y luego se ejecuta el script principal. A continuación, se presentan los

Figura 20 Ejecución del sistema y conexión a Blynk Cloud desde Raspberry Pi

Nota. El gráfico muestra la ejecución del sistema automatizado desde el entorno virtual blynk_env, así como la conexión correcta con el servidor de Blynk Cloud, paso esencial para el monitoreo y control remoto del riego automatizado. Tomado de Rojas, F. (2025).

Arquitectura de automatización

Se configuró un proceso de ejecución automática utilizando cron, de modo que el sistema se inicia de forma automática cada vez que la Raspberry Pi 5 es encendida o reiniciada. Con el objetivo de garantizar que el sistema opere de manera autónoma, sin requerir intervención manual después de cada arranque como se muestra en la imagen anterior.

Figura 21 Automatización del inicio del sistema

38

```
# In the following command
@reboot sleep 20 && /home/fabri/blynk_env/bin/python /home/fabri/riego_blynk.py >> /home/fabri/riego.log 2>&1
```

De esta forma, el sistema de riego automatizado se ejecuta de manera continua y autónoma, sin necesidad de intervención manual. Los registros de ejecución se almacenan en el archivo riego.log, lo que permite al usuario realizar un seguimiento del comportamiento del sistema.

b)

```
(blynk_env) fabri@raspberrypi5:~ $ cat /home/fabri/riego.log
```

Nota. El gráfico a) muestra la arquitectura de automatización, que integra dispositivos, y el gráfico b) Uso del comando cat en la terminal de la Raspberry Pi para acceder al archivo riego.log, donde se almacenan los mensajes de salida y errores del script riego_blynk.py. Tomado de Rojas, F. (2025). *Arquitectura de automatización*

3.2.6. Automatización y monitoreo

En esta sección se describe cómo el sistema lleva a cabo el riego de manera automática, tomando decisiones en función de los datos captados por los sensores ambientales, también se detalla el proceso de monitoreo en tiempo real mediante la plataforma Blynk IoT, que permite visualizar desde una interfaz remota el estado del entorno y el funcionamiento del sistema.

Figura 22 Lógica de automatización del riego basada en condiciones ambientales

```
# --- Automatización del riego ---
tanque_lleno = distancia1 < 15

if not tanque_lleno:
    blynk.virtual_write(9, "⚠ Nivel bajo de agua")
else:
    blynk.virtual_write(9, "✅ Nivel de agua adecuado")

if not modo_manual_normal and not modo_manual_nutrientes:
    if (
        humedad > UMBRAL_HUMEDAD and
        luz >= UMBRAL_LUZ and
        temp is not None and temp >= TEMP_MINIMA and
        tanque_lleno
    ):
        relay_valvula.value = False
        relay_bomba_normal.value = False
        blynk.virtual_write(10, "🔴 Riego automático activado")
        print("🔴 Riego automático activado")
    else:
        relay_bomba_normal.value = True
        relay_valvula.value = True
        blynk.virtual_write(10, "🟢 Riego automático desactivado")
        print("🟢 Riego automático desactivado")
```

a)

b)

```
# --- Automatización por hora específica
ahora = datetime.now()
dia_semana = ahora.weekday() # lunes=0, martes=1, ..., domingo=6
hora_actual = ahora.strftime("%H:%M")

if dia_semana == 1 and hora_actual in ["19:00", "19:10"]:
    print("🔴 Riego programado activado (martes)")
    relay_valvula.value = False
    relay_bomba_normal.value = False
    blynk.virtual_write(10, "🔴 Riego programado activado (martes)")
    time.sleep(60) # evita que se repita en el mismo minuto
```

Nota. a) muestra la lógica de automatización por condiciones ambientales, donde se evalúan variables como humedad, luz, temperatura y nivel de agua para activar el sistema de riego automáticamente; el gráfico b) representa la automatización por horario específico, programada para ejecutarse los martes a las 19:00 y 19:10. Tomado de Rojas, F. (2025). *Configuración de la aplicación Blynk InT*

3.2.7. Lógica de riego automático

El sistema tiene una lógica de riego automático basada en las condiciones ambientales detectadas por los sensores, cuando la humedad del suelo cae por debajo de lo predefinido, y siempre que la luminosidad se mantenga dentro de un rango, el sistema activa de forma automática tanto la bomba de riego como la electroválvula para iniciar el proceso de irrigación.

Si las condiciones no son óptimas o si el nivel de humedad del suelo es suficiente, el sistema mantiene los actuadores desactivados, evitando así un consumo innecesario de agua.

Esta lógica se evalúa de manera continua dentro del bucle principal del código, donde en cada iteración se comprueban las condiciones de riego para tomar las decisiones correspondientes.

Este enfoque híbrido, que combina un control automático con la posibilidad de intervención manual, dota al sistema de una gran flexibilidad y autonomía, permite al usuario supervisar y controlar el riego de forma remota desde cualquier lugar mediante la plataforma Blynk.cloud.

Figura 23 Lógica de automatización del riego basada en condiciones ambientales

```
# Lógica automática si no hay modo manual
if not modo_manual_normal and not modo_manual_nutrientes:
    if (humedad > UMBRAL_HUMEDAD and
        UMBRAL_PH_MIN <= ph_value <= UMBRAL_PH_MAX and
        luz >= UMBRAL_LUZ and
        temp is not None and temp >= UMBRAL_TEMP_MAX):
        relay_valvula.value = False
        relay_bomba_normal.value = False
        print("Riego automático activado")
    else:
        relay_bomba_normal.value = True
        relay_valvula.value = True
        print("Riego automático desactivado")

# Modo manual
if modo_manual_normal:
    relay_valvula.value = False
    relay_bomba_normal.value = False

if modo_manual_nutrientes:
    relay_valvula.value = False
    relay_bomba_nutrientes.value = False

time.sleep(5)
```

Nota. El gráfico muestra la lógica de automatización implementada en el sistema, donde se integran sensores y actuadores mediante código Python, permitiendo controlar procesos de riego de forma automática y manual. Tomado de Rojas, F. (2025). *Arquitectura de automatización*

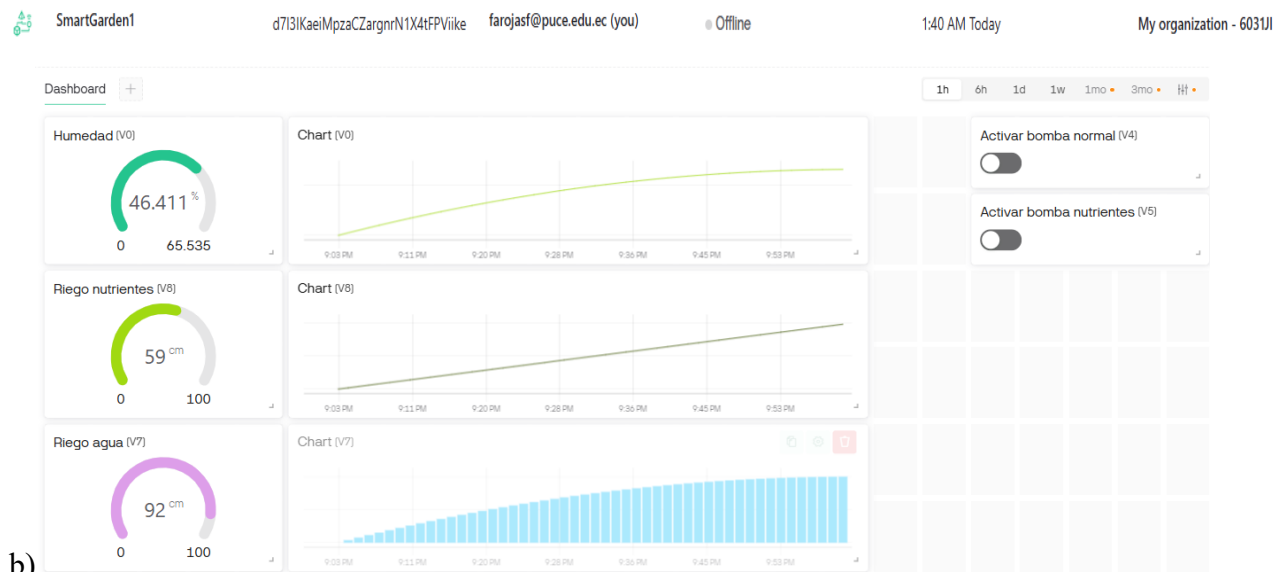
3.3. Integración con Blynk IoT

La plataforma Blynk IoT fue seleccionada como herramienta para la visualización remota y el control del sistema de automatización de riego desarrollado. Esta plataforma ofrece una solución flexible y multiplataforma que permite tanto el monitoreo en tiempo real de las variables ambientales como la gestión manual o automática de los actuadores, a través de una interfaz accesible desde dispositivos móviles y navegadores web.

3.3.1 Configuración de dashboard

Figura 24 Configuración de la aplicación Blybk InT

a)



b)



c)

Nota. La figura a) Detalles del dispositivo en Blynk IoT b) muestra la interfaz principal de la aplicación Blynk IoT, configurada para el monitoreo y control del sistema de riego automático; el gráfico c) muestra visualizaciones específicas de variables como luz y temperatura en tiempo real. Tomado de Rojas, F. (2025). *Configuración de la aplicación Blybk InT*

Estas imágenes muestran la interfaz principal de la aplicación Blynk IoT configurada para el sistema de riego, la dashboard permite visualizar en tiempo real los datos del ecosistema florícola, como la humedad del suelo, la temperatura ambiente, la humedad relativa, la luminosidad y el nivel de agua en los tanques, con el objetivo de facilitar el monitoreo remoto desde cualquier dispositivo móvil o navegador web.

3.3.2. Configuración de datastreams y pines virtuales

Figura 25 Configuración de Datastreams

ID	Name	Pin	Color	Data Type	Units	Is Raw	Min	Max	Decimals	Default Value	Automation Type	Condition	Action	Actions
8	Activar bomba normal	V4	Purple	Integer		false	0	1	--	0	Switch	On	Off	
1	Humedad	V0	Green	Integer	%	false	0	65535	--	0	Switch	On	Off	
3	Luz	V1	Yellow	Integer	lx	false	0	1000	--	0	Switch	On	Off	
4	PH	V2	Black	Integer		false	0	14	--	0	Switch	On	Off	
2	Riego	V6	Blue	Integer		false	0	1	--	0	Switch	On	Off	
6	Riego agua	V7	Cyan	Integer	cm	false	0	100	--	0	Switch	On	Off	
7	Riego nutrientes	V8	Dark Green	Integer	cm	false	0	100	--	0	Switch	On	Off	
5	Temperatura	V3	Brown	Integer		false	0	50	--	0	Switch	On	Off	+
9	Activar bomba nutrientes	V5	Light Blue	Integer		false	0	1	--	0	Switch	On	Off	

Nota. El gráfico muestra la tabla de configuración de *datastreams* en la aplicación **Blynk IoT**, donde se definen los pines virtuales, tipos de datos, rangos y controles asociados al monitoreo y automatización del sistema de riego inteligente. Tomado de Rojas, F. (2025). *Configuración de la aplicación Blynk IoT*

En la plataforma Blynk se configuraron los datastreams que permiten la interacción entre los sensores, actuadores y la interfaz de usuario del sistema Smart Garden. Cada datastream corresponde a una variable monitoreada o controlada, como la humedad del suelo, la temperatura, la luz ambiental o el pH. Asimismo, se establecieron datastreams para la activación de bombas y válvulas de riego, tanto para agua como para nutrientes. Estos están asociados a pines virtuales (V0 a V8) y configurados en modo *Switch*, lo que permite activar o desactivar componentes del sistema de forma manual o automática, según los valores detectados por los sensores.

Figura 26 Configuración del Virtual Pin de Activar bomba normal

The image shows the 'Virtual Pin Datastream' configuration screen in the Blynk IoT platform. It has two tabs: 'General' (selected) and 'Expose to Automations'. The 'General' tab contains the following fields:

- NAME:** A text input field containing 'Activar bomba normal'.
- ALIAS:** A text input field containing 'Activar bomba normal'.
- PIN:** A dropdown menu showing 'V4'.
- DATA TYPE:** A dropdown menu showing 'Integer'.
- UNITS:** A dropdown menu showing 'None'.
- MIN:** A text input field containing '0'.
- MAX:** A text input field containing '1'.
- DEFAULT VALUE:** A text input field containing '0'.
- Enable history data:** A toggle switch that is currently turned off.

At the bottom right of the form, there are two buttons: 'Cancel' (light green) and 'Save' (dark green).

La imagen muestra la configuración de un Virtual Pin Datastream en la plataforma Blynk IoT, utilizado para el control manual de la bomba de riego normal del sistema. En este caso, se ha asignado el Virtual Pin V4, configurado como un dato de tipo entero (Integer), con un rango de valores de 0 a 1, donde:

- 0 indica que la bomba está desactivada.
- 1 indica que la bomba está activada.

Los campos Name y Alias se han definido como “*Activar bomba normal*”, lo que facilita su identificación en el dashboard de la aplicación. La unidad está configurada como None (sin unidades), ya que este control funciona de manera binaria (encendido/apagado).

Por otro lado, la opción Enable history data no se encuentra activada, dado que en este caso no es necesario almacenar un historial de estados para este control específico.

Este Virtual Pin permite al usuario activar o desactivar manualmente la bomba de riego normal desde la aplicación móvil o la interfaz web de Blynk. Al cambiar el estado del botón virtual, se envía el comando correspondiente al sistema en la Raspberry Pi, que gestiona la activación o desactivación del relé que controla la bomba.

Figura 27 Configuración del Virtual Pin de Humedad

The screenshot shows the 'Virtual Pin Datastream' configuration window in the Blynk IoT application. The 'General' tab is selected, and the 'Expose to Automations' option is visible. The configuration fields are as follows:

NAME	ALIAS	PIN	DATA TYPE	UNITS	MIN	MAX	DEFAULT VALUE
Humedad	Humedad	V0	Integer	Percentage, %	0	65535	0

Below the fields, there is a toggle switch for 'Enable history data' which is currently turned off. At the bottom right, there are 'Cancel' and 'Save' buttons.

Nota. El gráfico muestra la ventana de configuración individual de un *datastream* en la aplicación **Blynk IoT**, donde se especifican parámetros como el nombre, tipo de dato, pin virtual, rango de valores y opciones de historial para un componente específico del sistema de riego inteligente. Tomado de Rojas, F. (2025). *Configuración de la aplicación Blynk IoT*

Se configuró un Virtual Pin Datastream en la plataforma Blynk IoT para visualizar la humedad del suelo que mide el sistema. En este caso, se utiliza el Virtual Pin V0, configurado como un dato de tipo Integer y con las unidades expresadas en porcentaje (%).

El rango de valores va desde 0 (que indica un suelo completamente seco) hasta 65.535, lo que cubre todo el rango de valores que puede entregar el sensor capacitivo de humedad del suelo, el cual está conectado a la Raspberry Pi a través del conversor analógico-digital ADS1115.

Con esta configuración, el nivel de humedad del suelo se muestra en tiempo real en el panel de control de la aplicación Blynk, permitiendo al usuario monitorear a distancia esta variable clave para el manejo eficiente y automatizado del sistema de riego.

Figura 28 Configuración del Virtual Pin Luz

Virtual Pin Datastream

General Expose to Automations



NAME

Luz

ALIAS

Luz



PIN

V1

DATA TYPE

Integer

UNITS

Lux, lx

MIN

0

MAX

1000

DEFAULT VALUE

0

☐ Enable history data

Cancel

Save

La configuración de un *Virtual Pin Datastream* en la plataforma Blynk IoT, utilizada para visualizar el nivel de luminosidad ambiental en el sistema. En este caso, se ha asignado el Virtual Pin V1, configurado como un dato de tipo *Integer*, con las unidades expresadas en lux (lx).

El rango de valores definido va de 0 a 1000 lux, lo que permite representar el nivel de luz ambiental que mide el sensor BH1750, el cual se comunica con la Raspberry Pi a través del bus I2C. El valor inicial se establece en 0, y la opción *Enable history data* no está activada para esta medición.

Gracias a esta configuración, la intensidad de luz captada por el sensor se visualiza en tiempo real en el panel de control de la aplicación Blynk. Esta información es útil tanto para complementar la lógica de riego automatizado como para el monitoreo general de las condiciones ambientales del entorno controlado.

Figura 29 Configuración del Virtual Riego de agua

Virtual Pin Datastream

General Expose to Automations



NAME

Riego agua

ALIAS

Riego agua



PIN

V7

▼

DATA TYPE

Integer

▼

UNITS

Centimeter, cm

▼

MIN

0

MAX

100

DEFAULT VALUE

0

☐ Enable history data

Cancel

Save

La configuración de un Virtual Pin Datastream en la plataforma Blynk IoT, utilizada para visualizar el nivel de agua en el sistema de riego. En este caso, se ha configurado el Virtual Pin V7, definido como un dato de tipo Integer y con unidades de medida en centímetros (cm).

El rango de valores establecido va de 0 a 100 cm, lo que permite representar de forma visual el nivel de agua medido por el sensor ultrasónico HC-SR04, instalado en el tanque del sistema. Este sensor calcula la distancia entre el propio sensor y la superficie del agua, y a partir de esa información se determina el nivel de agua disponible en el tanque. El valor predeterminado está configurado en 0, y la opción Enable history data permanece desactivada, ya que para esta variable no se requiere almacenar un historial.

Al incluir este Virtual Pin en el panel de control de Blynk IoT, el usuario puede consultar en tiempo real el nivel de agua del sistema, lo que facilita la supervisión del suministro y permite anticiparse a posibles necesidades de recarga o mantenimiento.

Figura 30 Configuración del Virtual Pin Riego de nutrientes

Virtual Pin Datastream

General Expose to Automations



NAME

Alias

Riego nutrientes

Riego nutrientes



PIN

DATA TYPE

V8

Integer

UNITS

Centimeter, cm

MIN

MAX

DEFAULT VALUE

0

100

0

☐ Enable history data

Cancel

Save

Nota. Los gráficos muestran la configuración individual de los pines virtuales en la aplicación **Blynk IoT**, utilizada para monitorear y controlar el sistema de riego inteligente. En a) se configura el pin V1 para la variable de luz ambiental en lux, en b) el pin V7 para el nivel del agua en centímetros, y en c) el pin V8 para el control del riego de nutrientes. Cada configuración especifica el rango, la unidad de medida y el tipo de dato correspondiente Tomado de Rojas, F. (2025). *Configuración de la aplicación Blynk IoT*

La configuración de un *Virtual Pin Datastream* en la plataforma Blynk IoT, destinado a visualizar el nivel de agua del tanque de nutrientes en el sistema de riego. Para esta variable se ha asignado el Virtual Pin V8, configurado como un dato de tipo *Integer*, con unidades de medida en centímetros (cm).

El rango de valores establecido va de 0 a 100 cm, lo que permite representar el nivel del líquido medido por un sensor ultrasónico HC-SR04, instalado en el tanque de solución nutritiva. Este sensor calcula la distancia hasta la superficie del líquido, y el sistema procesa esta información para enviarla a Blynk como un valor en centímetros. El valor predeterminado está configurado en 0, y la opción *Enable history data* permanece desactivada.

Gracias a esta configuración, el usuario puede supervisar en tiempo real el nivel del tanque de nutrientes desde el panel de control de Blynk, lo que facilita el seguimiento y el mantenimiento oportuno de la solución nutritiva dentro del sistema de riego automatizado.

Figura 31 Configuración del Virtual Pin de Temperatura

The screenshot shows the 'Virtual Pin Datastream' configuration page in the Blynk IoT platform. The 'General' tab is selected, and the 'Expose to Automations' option is also visible. The configuration fields are as follows:

- NAME:** A text input field containing 'Temperatura'.
- ALIAS:** A text input field containing 'Temperatura'.
- PIN:** A dropdown menu set to 'V3'.
- DATA TYPE:** A dropdown menu set to 'Integer'.
- UNITS:** A dropdown menu set to 'None'.
- MIN:** A text input field containing '0'.
- MAX:** A text input field containing '50'.
- DEFAULT VALUE:** A text input field containing '0'.
- Enable history data:** A toggle switch that is currently turned off.

At the bottom right of the configuration area, there are two buttons: 'Cancel' (light green) and 'Save' (dark green).

La configuración de un Virtual Pin Datastream en la plataforma Blynk IoT, utilizado para visualizar la temperatura ambiental en el sistema. Para esta variable se ha configurado el Virtual Pin V3, definido como un dato de tipo Integer.

Aunque en la configuración el campo Units aparece vacío (None), en la práctica la temperatura se mide en grados Celsius (°C), que es la unidad proporcionada por el sensor DHT22, conectado a la Raspberry Pi a través de un pin GPIO. El rango de valores establecido va de 0 a 50 °C, lo que permite cubrir las condiciones ambientales típicas esperadas en el entorno de operación del sistema. El valor predeterminado está configurado en 0, y la opción Enable history data permanece desactivada.

Con esta configuración, el usuario puede monitorear en tiempo real la temperatura del ambiente en el que opera el sistema de riego. Esta información complementa la lógica de control y facilita la toma de decisiones para una gestión hídrica más eficiente.

Figura 32 Configuración del Virtual Pin Riego de Activar bomba de nutrientes

The screenshot shows the 'Virtual Pin Datastream' configuration page in the Blynk IoT platform. The 'General' tab is selected. The configuration includes:

- NAME:** Activar bomba nutrientes
- ALIAS:** Activar bomba nutrientes
- PIN:** V5
- DATA TYPE:** Integer
- UNITS:** None
- MIN:** 0
- MAX:** 1
- DEFAULT VALUE:** 0
- Enable history data:** Disabled (toggle switch)

At the bottom right, there are 'Cancel' and 'Save' buttons.

Nota. El gráfico a) muestra la configuración del pin virtual para la variable de temperatura, mientras que el gráfico b) muestra la configuración del control manual para activar la bomba de nutrientes. Ambos se configuran en la plataforma Blynk IoT para su monitoreo y control remoto dentro del sistema de riego automatizado. Tomado de Rojas, F. (2025). *Configuración de la aplicación Blybk InT*

La configuración de un *Virtual Pin Datastream* en la plataforma Blynk IoT, utilizado para el control manual de la bomba de nutrientes en el sistema de riego automatizado. Para esta función se ha configurado el Virtual Pin V5, definido como un dato de tipo *Integer*, con un rango de valores de 0 a 1, donde:

- **0** indica que la bomba está desactivada.
- **1** indica que la bomba está activada.

El campo *Units* está configurado como *None*, ya que se trata de un control binario (encendido/apagado) que no requiere unidades de medida, el valor predeterminado está establecido en 0, y la opción *Enable history data* permanece desactivada, ya que no es necesario almacenar un historial de los estados de este control.

Con esta configuración, el usuario puede activar o desactivar manualmente la bomba de nutrientes desde la interfaz de la aplicación Blynk.cloud, ya sea desde la versión móvil o

web. Al hacerlo, se envía el comando correspondiente a la Raspberry Pi, que gestiona la activación del relé asociado a esta bomba.

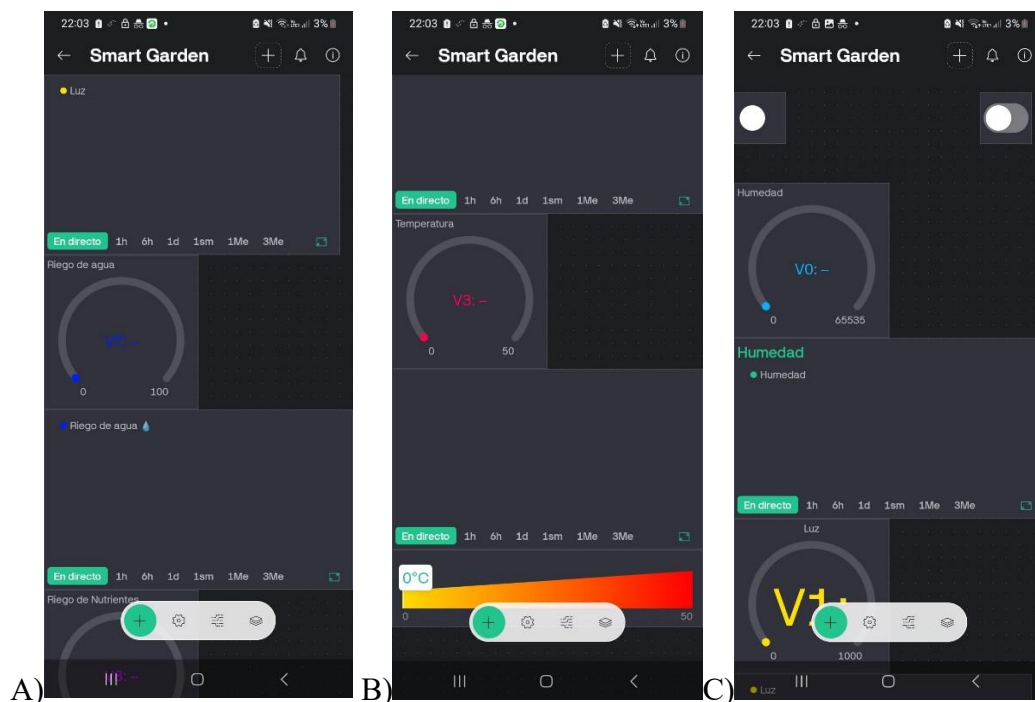
3.3.2 Interfaz en aplicación móvil

La aplicación móvil cuenta con la misma funcionalidad que la versión web, ya que están sincronizadas a través de la plataforma Blynk Cloud. Esta conexión permite que cualquier cambio en una se refleje inmediatamente en la otra, lo que permite la facilidad del diseño y visualización de la interfaz en tiempo real.

La construcción de la interfaz resulta rápida e intuitiva, permitiendo al usuario controlar y monitorear variables críticas del sistema domótico, como la humedad del suelo, la temperatura ambiental, la intensidad lumínica y el estado de los sistemas de riego.

A continuación, se presentan capturas de pantalla de la aplicación móvil:

Figura 33 Aplicación móvil



Nota. El gráfico a) muestra la interfaz principal de la aplicación **Blynk IoT**, configurada para el sistema de riego automatizado desde un dispositivo móvil; el gráfico b) presenta la visualización de diferentes pines virtuales utilizados para el monitoreo de variables como humedad y activación de bombas. Tomado de Rojas, F. (2025). *Configuración de la aplicación Blynk IoT*

3.4. Proyección futura

El prototipo desarrollado ha cumplido con su propósito de automatizar el riego y monitorear variables clave en cultivos ornamentales, el prototipo tiene gran potencial para seguir mejorando.

3.4.1. Escalabilidad y red

Una de las ideas más interesantes a futuro es permitir el control de varias zonas de cultivo desde una sola interfaz. Para conseguirlo se puede conectar varias Raspberry Pi o microcontroladores en red, de forma que cada uno cubra un sector distinto, también se podría aplicar una red en malla, lo que mejoraría la comunicación entre dispositivos y permitiría cubrir áreas más extensas sin perder eficiencia.

3.4.2. Autonomía Energética Uso de energía solar

Integrar energías limpias para operar el sistema en zonas sin acceso a la red eléctrica, garantizando su autonomía energética.

3.4.3. Mejora de la interfaz Aplicación móvil propia

Se podría desarrollar una aplicación específica, con mayor grado de personalización y funcionalidades avanzadas como accesos por roles, la correcta implementación de niveles de usuario (administrador, técnico, operario) facilitaría la gestión del sistema.

3.4.4. Aplicación social y comercial

Este sistema podría tener un impacto positivo más allá de lo técnico, sería útil implementarlo en comunidades rurales o capacitar a pequeños floricultores para que lo usen y lo adapten a sus propios cultivos y para comercializar, lo cual facilitaría su adopción por parte de quienes buscan soluciones prácticas y accesibles.

3.4.5. Inteligencia Artificial

Riego adaptativo con IA, se proyecta el uso de modelos predictivos que ajusten los parámetros de riego automáticamente en función del clima y las necesidades de las plantas con análisis de imágenes, con herramientas de visión computacional sería posible identificar anomalías, enfermedades y plagas.

4. CAPÍTULO IV: RESULTADOS

4.2. Análisis de Resultados

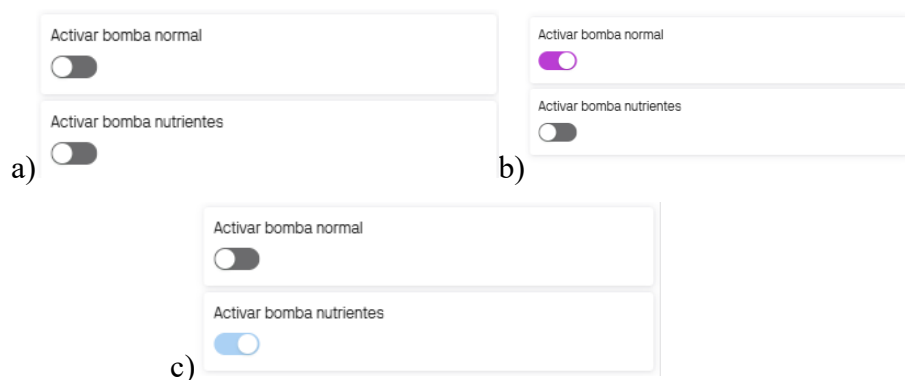
El desarrollo del prototipo sirvió como una prueba clara de que es posible automatizar el riego y el monitoreo ambiental en un entorno florícola utilizando un sistema domotizado. Lo interesante fue ver cómo, al integrar sensores de humedad, temperatura, luz y nivel de agua, se pudo lograr un riego más eficiente, sin necesidad de intervención constante por parte de una persona. Esto no solo simplificó las tareas, sino que también ayudó a usar mejor el agua, algo clave en cualquier cultivo.

Durante las pruebas se comprobó que, al tomar decisiones de riego basadas en las condiciones reales del entorno, se mantiene un equilibrio más estable en el ecosistema, lo que se traduce en un mejor cuidado para las plantas ornamentales. Además, gracias a una interfaz remota basada en Blynk.io, fue posible seguir en tiempo real las variables más importantes del cultivo. Esta capacidad de monitoreo demuestra que el sistema tiene potencial para adaptarse a otros contextos agrícolas con características similares.

4.1.1 Pruebas del riego manual

Se validó la funcionalidad de los botones virtuales configurados para el riego manual, permitiendo activar y desactivar las bombas de agua normal y de nutrientes. El sistema reconoció correctamente cada comando enviado desde la aplicación, y se confirmó la activación independiente de cada relé, sin interferencias.

Figura 34 Dashboard de los botones de activación de bomba de nutrientes y normas



Nota. La figura a) Tenemos los activadores para las bombas de agua b) Activador prendido de la bomba de agua c) Activador prendido de la bomba de agua con nutrientes. Tomado de Rojas, F. (2025). *Configuración de la aplicación Blybk InT*

Tenemos los actuadore para las bombas de aguas de manera manual para poder regar en caso de que sea necesario, utilizamos 2 para una bomba norma y otra para los nutrientes.

Figura 35 Verificación del funcionamiento del sistema de bombeo



Nota. El gráfico a) muestra el sistema de bombeo dentro del balde principal, donde se encuentran sumergidas las bombas encargadas de distribuir el agua y los nutrientes. El gráfico b) evidencia el montaje completo del prototipo, incluyendo la planta ornamental, los sensores, relés, mangueras y la Raspberry Pi como unidad de control. Tomado de Rojas, F. (2025). *Configuración de la aplicación Blybk InT*

Las imágenes muestran el prototipo funcional del sistema de riego automatizado que se desarrolló. En el balde amarillo se alojan las bombas sumergibles, que tienen la tarea de extraer tanto el agua como los nutrientes necesarios para la planta.

4.1.2 Riego automático programado

Se probó el riego automático según una programación horaria específica en nuestro caso fue el miércoles a las 7:35 y 7:40, cuando llega a la hora configurada, el sistema activó la bomba automáticamente durante un minuto.

Figura 36 Verificación del funcionamiento del riego automático con la primera hora



Nota. La imagen a) nos indica la hora actual b) Se evidencia que el sistema está funcionando, activando la bomba de agua. Tomado de Rojas, F. (2025). *Configuración de la aplicación Blybk InT*

Durante la prueba del sistema de riego automático, se validó que el prototipo al activarse solo, según un horario previamente programado se comprobó que el sistema funciona a la perfección cumpliendo uno de los objetivos que es que funcione el sistema automatizado de riego.

Figura 37 Verificación del funcionamiento del riego automático con la segunda hora



Nota. La imagen a nos indica la hora del segundo riego para nutrientes b) Se evidencia que el sistema está funcionando, activando la bomba de agua. Tomado de Rojas, F. (2025). *Configuración de la aplicación Blybk InT*

Según la hora que se programó el riego automático para el riego con nutrientes funciona de la misma manera que la anterior explicación con el riego de agua y se validó que el sistema funciona a la perfección.

Esto demuestra que el sistema responde correctamente a eventos temporales, lo que resulta especialmente útil cuando se busca mantener una rutina de riego regular.

4.1.3 Lógica de riego automático por condiciones

El sistema evaluó las condiciones de riego según cuatro variables clave: humedad del suelo, temperatura ambiente, luz ambiental y nivel de agua en el tanque. Los resultados fueron los siguientes:

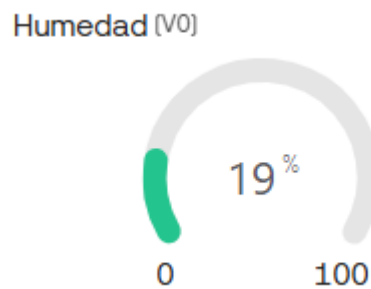
4.1.3.1 Evaluación del sensor de humedad del suelo

En este apartado se evalúa el comportamiento del sensor de humedad del suelo ante variaciones en la cantidad de agua en nuestra planta.

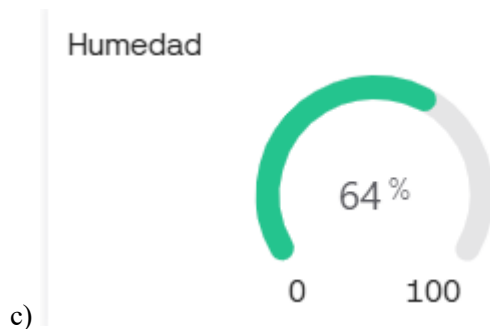
Figura 38 Verificación del funcionamiento del sensor de humedad del suelo con su



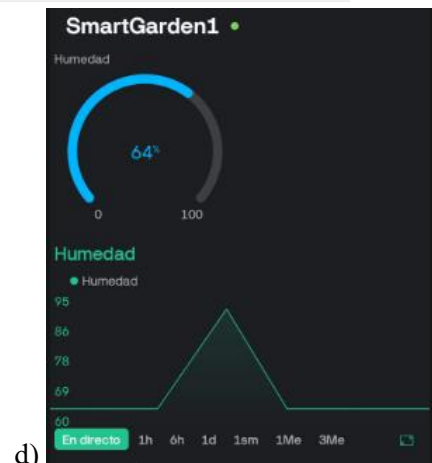
a)



b)



c)



d)

Nota. La imagen a) Sensor de humedad de tierra en la planta b) Monitoreo de humedad antes de ser regado, c) La humedad aumenta según se riega la planta, d) monitore de humedad en celular Tomado de Rojas, F. (2025). *Configuración de la aplicación Blybk InT*

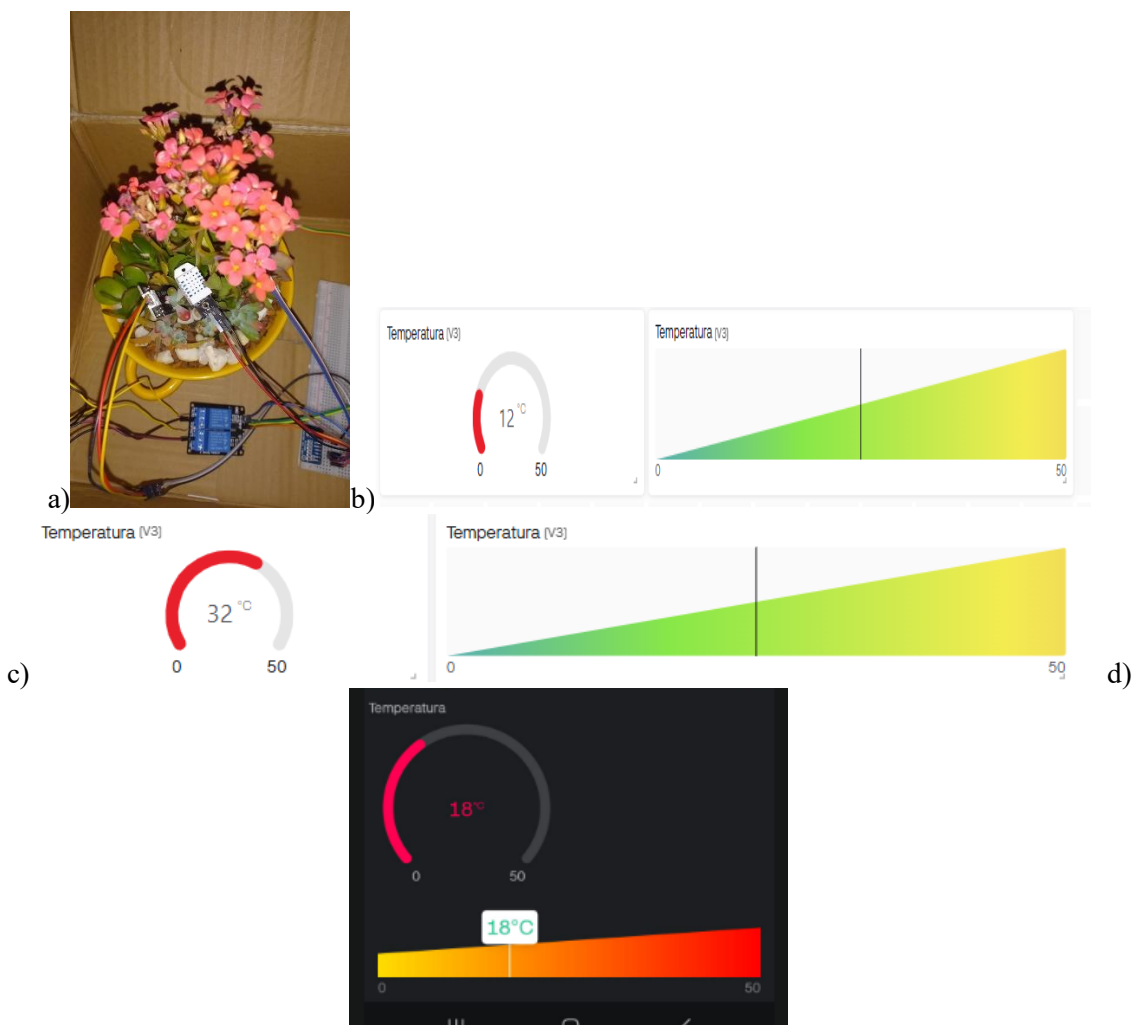
Se realizo la prueba de campo del sensor de humedad de tierra en el cual se monitorea el funcionamiento de este donde indica si la humedad decae bastante esta se debe ser regada evidenciándose en la última imagen con el aumento de la humedad de a tierra una vez la planta fue sometida al riego de agua por el sistema, de igual manera en la app móvil se reflejan los datos de igual manera que la página web.

4.1.3.2 Evaluacion del sensor de temperatura ambiental

En esta sección se analiza la respuesta del sistema frente a valores elevados de

temperatura ambiental, detectados mediante el sensor DHT22.

Figura 39 Verificación del funcionamiento del sensor de temperatura ambiental con su dashboard



Nota. La imagen a) Sensor de temperatura en la planta b) Monitoreo de temperatura de nuestra planta, c) Aumento según le aplica calor con una secadora, d) Monitoreo de temperatura en la app móvil. Tomado de Rojas, F. (2025). *Configuración de la aplicación Blybk InT*

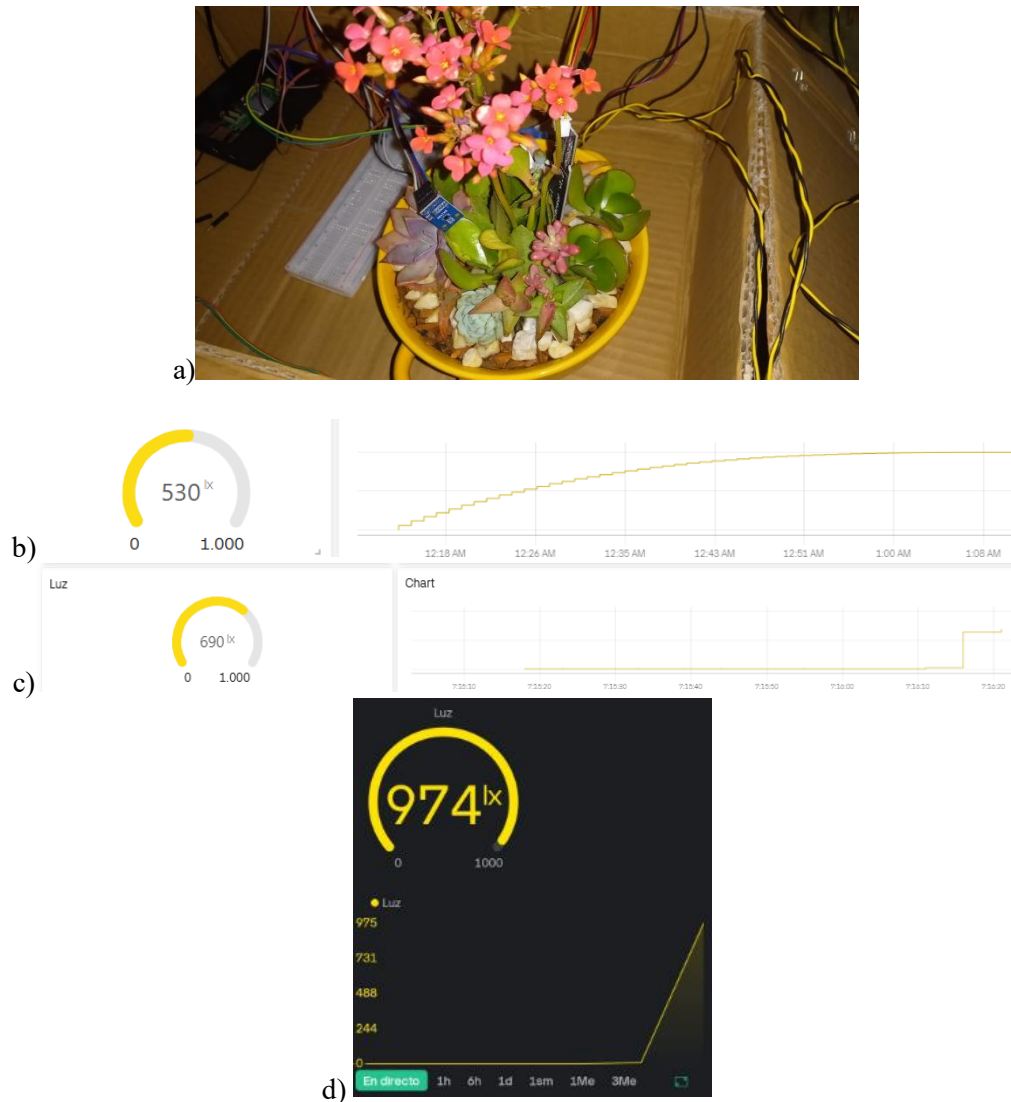
Se realizó la comprobación del funcionamiento del sensor de temperatura para verificar su operatividad en nuestro prototipo. Se evidenció que el sensor funciona correctamente, ya que se aplicaron las condiciones necesarias para su validación. En el caso específico de la temperatura, esta fue incrementada intencionalmente con el fin de observar el comportamiento del sensor y del sistema de riego. Una vez que la temperatura aumentó, el sistema activó automáticamente el riego, demostrando así su adecuada respuesta.

4.1.3.3 Evaluación de sensores de luminosidad

Este subapartado presenta los resultados obtenidos al probar el sensor de luminosidad

BH1750, cuya función es detectar la cantidad de luz presente en el entorno.

Figura 40 Evaluación del funcionamiento del sensor de luminosidad con su dashboard



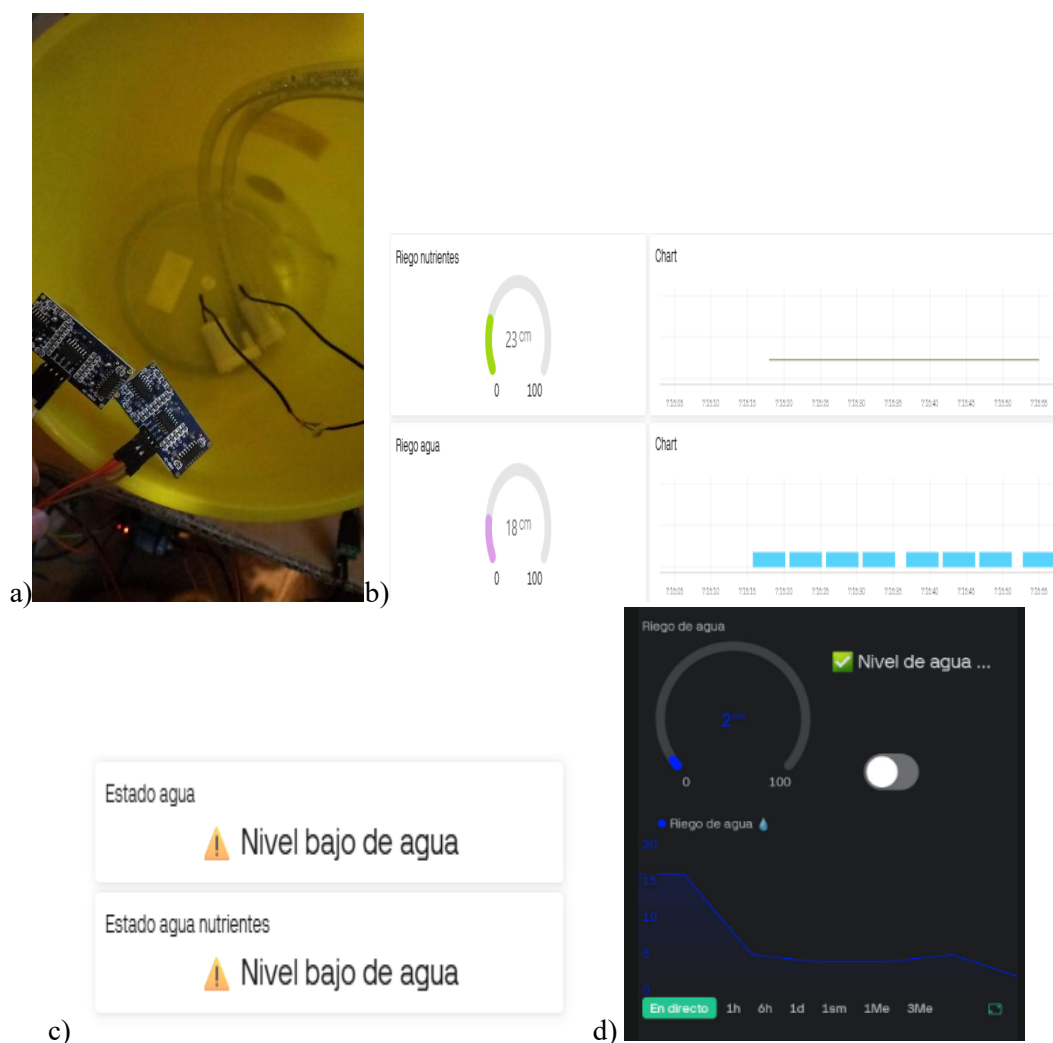
Nota. La imagen a) Sensor de Luz ya en la planta b) Monitoreo de luz de nuestra planta, c) La luz aumenta según se le aplica una fuente de luz con una linterna, d) Monitoreo de luz en la app móvil. Tomado de Rojas, F. (2025). *Configuración de la aplicación Blybk InT*

En el caso de el sensor de luz su función es darnos a conocer cuanta luz recibe nuestra planta al estar en un lugar cerrado el sensor de la imagen b no refleja tanta luz, pero una vez que se le acerca una linterna esta crece como se muestra en la imagen c y así mismo se activa el relé para el sistema de riego.

4.1.3.4 Evaluación del sensor ultrasónico para el nivel del tanque

En este segmento se examina el funcionamiento del sistema de control de nivel de agua en el tanque, basado en un sensor ultrasónico.

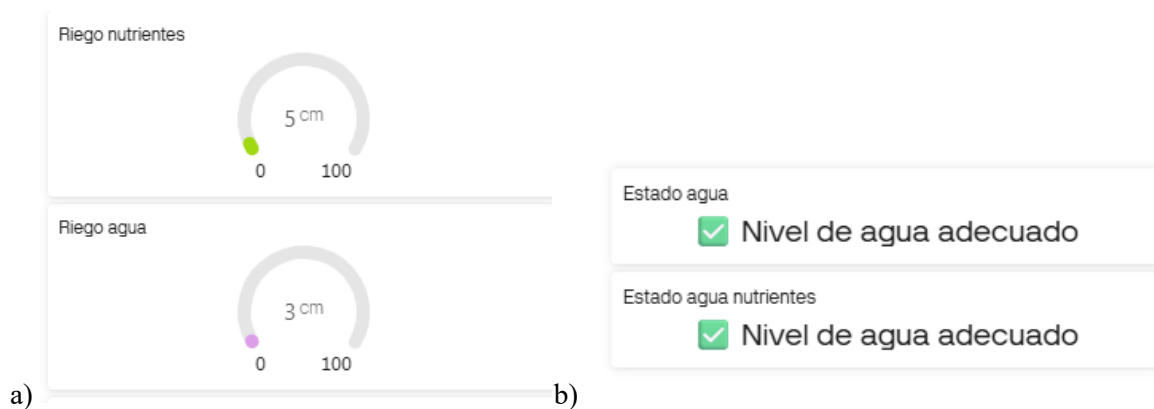
Figura 41 Evaluación del funcionamiento del ultrasónico para el nivel del tanque con su dashboard



Nota. La imagen a) Sensores de medición de distancia ya en la planta b) Estado del agua de nuestra en un ciclo de tiempo de las fuentes de agua, c) Se presenta el mensaje de poca agua ya que el valde esta con bajo nivel de agua, d) Monitoreo del sistema de riego de agua en la app móvil. Tomado de Rojas, F. (2025). *Configuración de la aplicación Blybk InT*

Para poder tener controlado el sistema de agua los sensores de medición nos indican la cantidad de agua que esta posee dándonos la información es cm entre más grande sea la distancia indica que las reservas de agua se están agotando lo cual se refleja en las alertas del estado de agua.

Figura 42 Evaluación del funcionamiento de las métricas del tanque con su dashboard



Nota. La imagen a) Se evidencia el monitoreo de la distancia del agua de nuestra planta b), c) Se presenta el mensaje de nivel adecuado de agua ya que está lleno de agua. Tomado de Rojas, F. (2025). *Configuración de la aplicación Blybk InT*

Como se ve en la imagen a al ser la distancia menor como 3cm a 5cm nos indica que la cantidad de agua de las reservas son las óptimas, complementando esto los mensajes de estado de las reservas de agua nos indica que el nivel de agua es el indicado.

5. CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES

5.2. Conclusiones

- El sistema de automatización implementado demostró ser funcional bajo condiciones controladas, se verificó que es capaz de activar el riego en momentos específicos definidos por el programador, sin requerir intervención humana. Esto confirma que la lógica basada en tiempo funciona de forma fiable, lo cual es especialmente útil para ejecutar riegos en horarios estratégicos.
- El sistema no solo se limita a seguir una programación horaria, sino que incorpora sensores que permiten tomar decisiones de riego basadas en las condiciones reales del entorno. Esta capacidad de adaptación evita tanto el riego innecesario como la falta de agua, haciendo que el funcionamiento sea más eficiente y acorde a las necesidades reales del cultivo.
- La integración con la plataforma Blynk (IoT) permite al usuario monitorear en tiempo real los datos recogidos por los sensores, visualizar históricos mediante gráficas e incluso activar o desactivar el sistema desde una aplicación o navegador, con el objetivo de que sea intuitiva para una buena experiencia de uso.
- A pesar de tratarse de un prototipo a escala reducida, el prototipo logró replicar de forma convincente las condiciones de un entorno florícola, activando bombas y distribuyendo agua de forma efectiva.
- No se implementó la medición de pH de tierra por cuestiones de presupuesto del prototipo, sumando a ello el sensor de pH de tierra existe fuera del país con precios elevados por su uso industrial.

5.3. Recomendaciones

- Es recomendable incorporar sensores de nivel de agua más precisos. Actualmente, el sistema emplea un sensor ultrasónico para estimar el nivel del tanque, pero si cualquier objeto es mal colocado ya sea el mismo motor para bombear el agua o las mangueras del sistema de riego puede llegar a alterar la fiabilidad de las mediciones para solucionarlo se debería evaluar alternativas como los sensores de flotador o de presión, que suelen ofrecer lecturas más estables y consistentes en entornos húmedos o con presencia de condensación.

- Realizar pruebas en entornos reales para implementar el sistema en contextos agrícolas reales, como invernaderos o parcelas productivas, para evaluar su comportamiento bajo condiciones ambientales cambiantes, para poder identificar posibles vulnerabilidades frente variaciones climáticas, factores que no siempre pueden simularse completamente en laboratorio.

- Para trabajos futuros se recomienda explorar el uso de algoritmos de machine learning para ajustar el riego de forma dinámica en función del comportamiento pasado del sistema y las condiciones climáticas. Esto aumentaría la autonomía y precisión del sistema en escenarios cambiantes.

- Para futuros trabajos se recomienda adaptar el sistema para gestionar varias zonas de riego de forma independiente, mediante la distribución de sensores y actuadores en distintos sectores del cultivo y utilizar nodos controladores que trabajen de manera coordinada, con el objetivo de no solo enfocarse en un solo tipo de planta sino en una mayor variedad.

6. BIBLIOGRAFÍA

Agroingenia Canarias. (2023, 23 de febrero). *¿Qué es el riego inteligente?*
<https://agroingeniacanarias.com/que-es-el-riego-inteligente/>

Casadesús, J., Biel, C., & Montesinos, P. (2019). Using soil moisture sensors for automated irrigation scheduling in a plum orchard: Water-balance-based irrigation with feedback adjustment. *Water*, 11(10), 2061. <https://doi.org/10.3390/w11102061>

Diosdado, R. (2014). *Sensor de ultrasonidos HC-SR04*. Zona Maker.
<https://www.zonamaker.com/ultrasonido-hc-sr04>

García, E., & Sánchez, M. (2018). Cuidado de plantas ornamentales mediante sistemas automatizados. *Revista Agrícola Digital*, 8(2), 45–55.

Instructables. (2018). *Raspberry Pi powered IoT garden*.
<https://www.instructables.com/Raspberry-Pi-Powered-IoT-Garden/>

Kalluri, S. K., & Kalidindi, D. (2020). Smart irrigation systems using IoT for sustainable water management. *Journal of Environmental Management*, 265, 110501.
<https://doi.org/10.1016/j.jenvman.2020.110501>

Kim, Y., Evans, R. G., & Iversen, W. M. (2008). Remote sensing and control for precision agriculture: A wireless application. *Computers and Electronics in Agriculture*, 61(1), 32–39.
<https://doi.org/10.1016/j.compag.2007.05.001>

Lal, R. (2020). Soil health and climate resilience: Sowing the seeds of sustainable agriculture. *Journal of Soil and Water Conservation*, 75(5), 98–104.
<https://doi.org/10.2489/jswc.2020.0620A>

Madakam, S., Ramaswamy, R., & Tripathi, S. (2015). Internet of Things (IoT): A literature review. *Journal of Computer and Communications*, 3(5), 164–173.
<https://doi.org/10.4236/jcc.2015.35021>

Mahmud, M. S., Zahid, A., & Das, A. K. (2023). Sensing and automation technologies for ornamental nursery crop production: Current status and future prospects. *Sensors*, 23(4), 1818. <https://doi.org/10.3390/s23041818>

Projects Raspberry. (2018). *Smart home gardening system using Raspberry Pi*.
<https://projects-raspberry.com/smart-home-gardening-system-using-raspberry-pi/>

SwitchDoc Labs. (2018, 9 de noviembre). *New product smart garden system – Raspberry Pi*.
<https://www.switchdoc.com/2018/11/new-smart-garden-system-raspberry-pi/>

SwitchDoc Labs. (2018). *Tutorial: Using Blynk with the Raspberry Pi smart garden system in Python*. <https://www.switchdoc.com/2018/11/tutorial-using-blynk-with-the-raspberry-pi-smart-garden-system-in-python/>

Upton, E., & Halfacree, G. (2019). *Raspberry Pi user guide* (4ª ed.). Wiley.

