

Universidad Tecnológica Nacional

Facultad Regional San Francisco



Comunicaciones de datos

Controlador de Hidroponia Automatizado

Autores: Scocco, Andrés Francisco

Viarengo, Nicole Jazmín

Profesor: Valverde, Franco

San Francisco, Córdoba, Argentina - Noviembre 2024

Índice

Introducción.....	2
Problemática.....	3
Objetivo	3
Solución Propuesta	4
1. Sensado.....	4
2. Control.....	4
3. Potencia	4
Comunicación y Conectividad	5
Implementación del Hardware	5
Explicación del Código	7
Comunicación de Datos y Almacenamiento	8
Estructura de la Interfaz en Django	8
1. Pantalla de Resumen	9
2. Pantalla de Cultivos.....	10
Integración del Sistema con MQTT	11
Resultados esperados.....	12
Conclusión.....	13

Informe del Proyecto: Controlador de Hidroponía Automatizado

Materia: Comunicaciones de Datos

Autores: Andrés Francisco Scocco y Nicole Jazmín Viarengo

Introducción

La hidroponía es una técnica de cultivo sin tierra que utiliza soluciones de nutrientes disueltas en agua. Este tipo de cultivo presenta algunas ventajas como un mayor control sobre los nutrientes y un uso más eficiente del agua. A diferencia de los sistemas agrícolas convencionales, los cultivos hidropónicos requieren una supervisión constante de variables como la temperatura, la humedad y el nivel de nutrientes, cualquier desequilibrio puede afectar el desarrollo de las plantas.

En los sistemas hidropónicos tradicionales, estas variables suelen ser controladas manualmente, lo que demanda mucho tiempo y esfuerzo. La falta de automatización no solo limita la eficiencia, sino que también incrementa el riesgo de errores humanos que pueden perjudicar las plantas. Este proyecto busca resolver esta problemática mediante un sistema automatizado basado en tecnología IoT, que controla y monitorea en tiempo real las condiciones del ambiente hidropónico.

Problemática

A continuación se describen los principales desafíos en los sistemas de hidroponía tradicionales:

1. **Control preciso del riego:** la aplicación manual del agua suele ser ineficiente y puede resultar en riegos excesivos o insuficientes, afectando negativamente el crecimiento de las plantas.
2. **Falta de monitoreo constante:** no contar con un sistema automatizado implica que se debe estar presente físicamente para ajustar las condiciones de temperatura y humedad, lo que no siempre es posible.
3. **Ineficiencia en el uso de recursos:** al regar manualmente o con sistemas de riego, se puede generar un uso excesivo de agua, además de que el encendido continuo de sistemas de riego automático puede consumir energía innecesaria.

Objetivo

Desarrollar un sistema automatizado de monitoreo y control para cultivos hidropónicos que optimice el uso de agua, mantenga condiciones de crecimiento estables y reduzca la necesidad de intervención humana constante. Este sistema permitirá a los usuarios acceder a los datos de sus cultivos y realizar ajustes de forma remota, aumentando la eficiencia y confiabilidad del proceso hidropónico.

Solución Propuesta

Para abordar esta problemática, implementamos un sistema IoT para el control automatizado del riego y para conocer las condiciones ambientales de un cultivo hidropónico. Para ello, dividimos nuestro sistema en tres etapas principales: sensado, control y potencia.

1. Sensado

Se utilizan diversos sensores para monitorear las condiciones del cultivo:

- **Sensor DHT11:** este sensor mide la temperatura y la humedad del ambiente. Es económico, confiable y proporciona datos precisos que son cruciales para monitorear el entorno del cultivo.
- **Sensor de Humedad del Suelo:** este sensor mide el nivel de humedad directamente en el sustrato donde se encuentran las raíces de las plantas. Si el nivel de humedad cae por debajo del 50%, el sistema activa automáticamente el riego hasta alcanzar un nivel óptimo del 80%.
- **Sensor de Luz (LDR):** el sistema también incluye un sensor de luz (fotorresistencia) que mide la cantidad de luz disponible para las plantas, prendiendo una lámpara si la planta necesitara luz.

Todos estos sensores están conectados a un microcontrolador, que recopila y procesa los datos de manera continua.

2. Control

El microcontrolador procesa la información recibida de los sensores y utiliza algoritmos de control para decidir cuándo deben activarse los sistemas de riego y ventilación. Además, se emplea una tabla de parámetros en la base de datos que define los valores óptimos de temperatura, humedad, y luz para cada tipo de cultivo. De este modo, el sistema adapta automáticamente las condiciones de cultivo según los requisitos específicos de cada planta.

En esta etapa también se integró la API de OpenWeatherMap, que permite obtener datos en tiempo real sobre la temperatura y humedad externa. Esto resulta especialmente útil para ajustar los ciclos de riego en función de las condiciones climáticas actuales, lo que aporta un nivel adicional de precisión y adaptabilidad al sistema.

3. Potencia

Para la activación de los actuadores, utilizamos relés que permiten manejar dispositivos de distintos voltajes, de 5V, convirtiendo los 220V de línea a través de una fuente. Los relés garantizan que los dispositivos de alta potencia (lampara y electroválvula) se activen solo cuando sea necesario, reduciendo así el consumo de energía y optimizando el uso de recursos.

Comunicación y Conectividad

La transmisión de datos entre los dispositivos se realiza mediante el protocolo MQTT, un protocolo de comunicación ligera que permite la publicación y suscripción de datos en tiempo real, de manera eficiente y con baja latencia. Empleamos MQTTX como broker para gestionar esta comunicación, permitiendo una conexión constante y eficiente entre el sistema de control en Arduino y el backend.

La arquitectura del sistema permite que los datos capturados por los sensores se almacenen en una base de datos MySQL cada hora, lo que permite un historial detallado de las condiciones del cultivo. Un script en Python facilita la conexión entre el broker MQTT y la base de datos, asegurando que todas las medidas de los sensores se registren y estructuren adecuadamente para su posterior análisis y visualización.

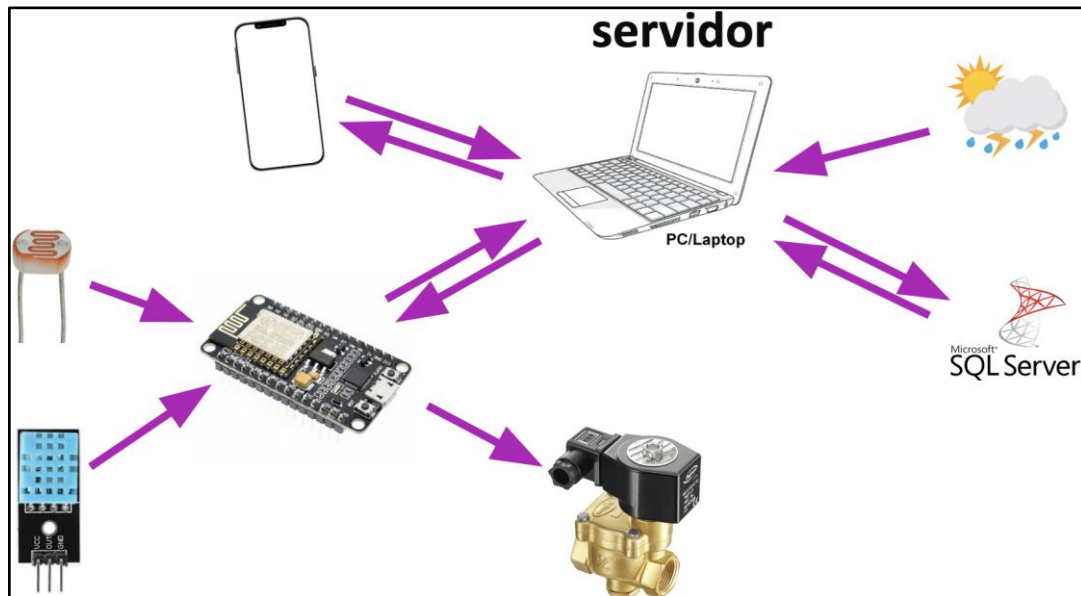
Además, integramos un sitio web integrando el Frontend y Backend donde se pueden visualizar gráficamente los valores actuales de temperatura, humedad y estado del riego. Desde esta interfaz también es posible ajustar los umbrales de control (parámetros) de forma remota, lo que agrega un nivel de flexibilidad al sistema.

Implementación del Hardware

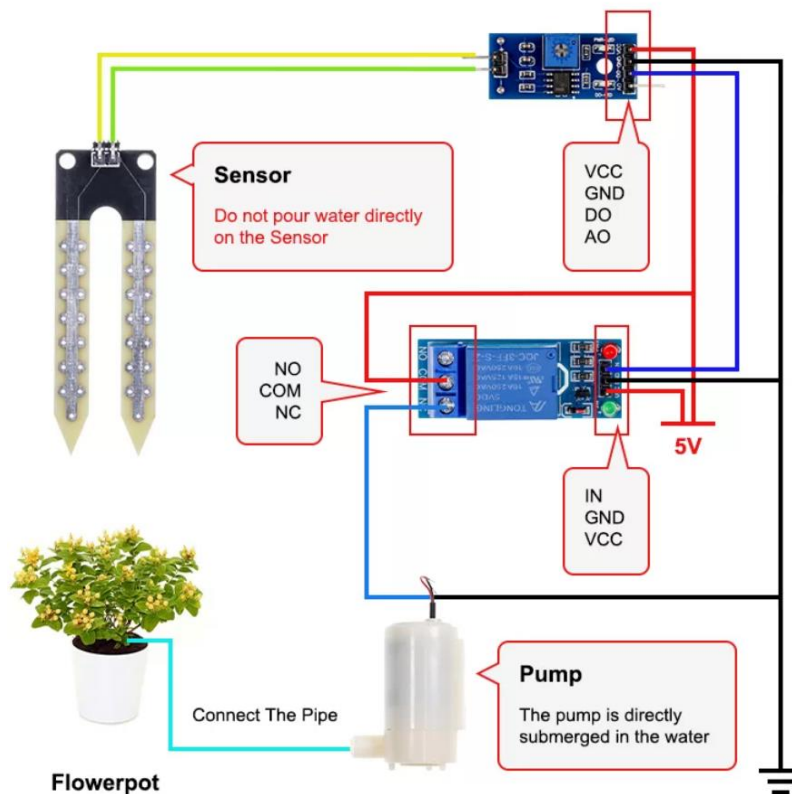
El sistema está compuesto por los siguientes elementos de hardware:

- **Placa universal:** de diseño propio. pensada para conectar la ESP32 con los diferentes sensores utilizados y demás cosas que se le quieran agregar en el futuro. La placa fue diseñada en KiCad y construida con la empresa Pcboard, especializada en diseño de hardware. El instagram de la empresa es: @pcboard_ingenieria
- **Sensores DHT11:** utilizado para medir la temperatura y la humedad del aire.

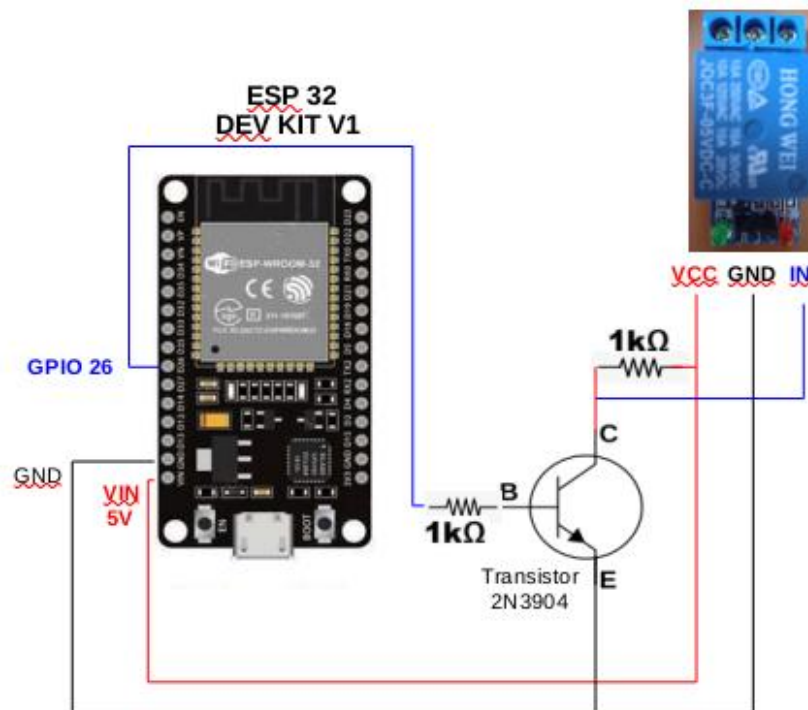
- **Sensor LDR:** fotoresistencia para medir la luminiscencia.
- **Sensores de humedad del suelo:** permiten detectar el nivel de agua en el sustrato.
- **Relés de 5V y Fuente de 220V:** usados para controlar el encendido de bombas de agua.
- **Microcontrolador ESP32 (devkit V1):** centraliza la información de los sensores y ejecuta el código de control.
- **Conexión MQTT:** permite la comunicación entre el hardware y el servidor.



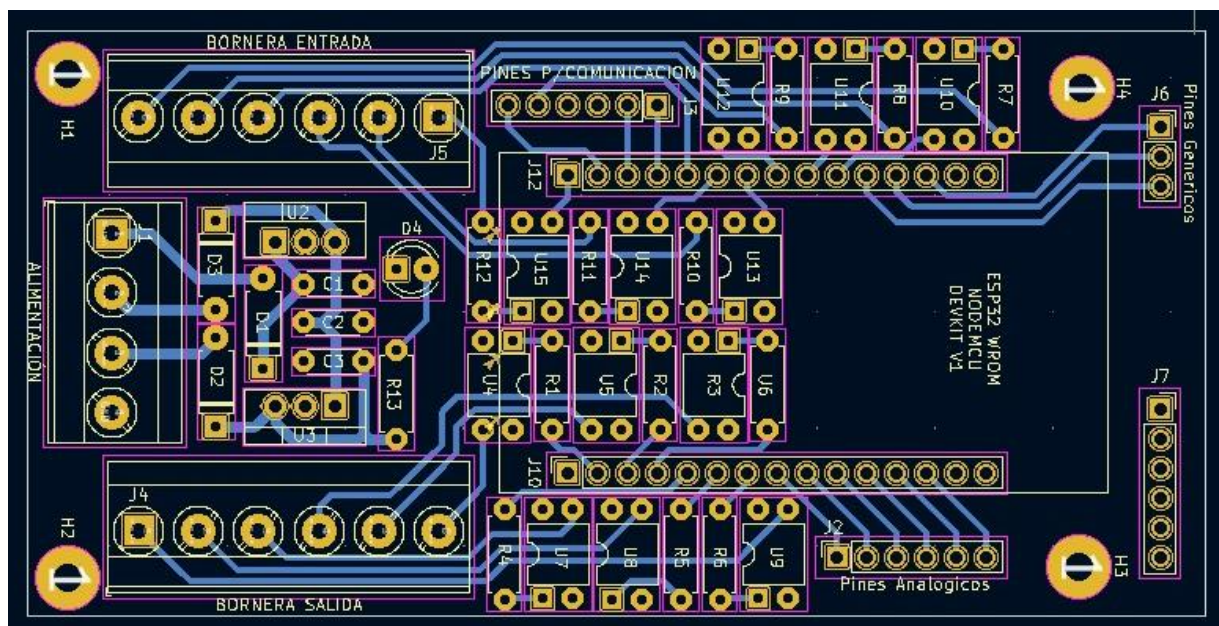
1. Diagrama del proyecto



2.1 Diagrama de conexiones de hardware



2.2 Diagrama de conexiones del hardware



usuario. Empleamos varios sensores para monitorear diferentes variables ambientales que son esenciales para el cultivo hidropónico.

A lo que llamamos sensor es físicamente es un transductor, cuyo uso es el de transformar una unidad física en una eléctrica. Los sensores aquí utilizados trabajan de manera que cuando varia a magnitud física que miden, a sea humedad o luz entre otros valores, varia su resistencia y al aplicarles una determinada tensión (5 V) podemos medir una tensión de salida que significa el cambio de la magnitud física medida a través del sensor.

Otros sensores ya ejecutan todo el procesamiento de la información que miden internamente y entregan, una vez alimentados, información mediante algún tipo de protocolo. Como es con el DHT, que mide temperatura y humedad de ambiente.

Se elige medir estos 4 valores ya que son de las variables más significativas, junto con valores de la tierra como pH o minerales, a la hora de tener ecosistemas controlados (viveros)

La toma de decisiones con estos datos se determina por parte del usuario.

El código del microcontrolador ejecuta dos tareas en paralelo, por un lado, se comunica constantemente con wifi consultando la hora actual, una vez cumplido cierto plazo de tiempo elegido como parámetro realiza las mediciones antes descriptas y toma las decisiones habilitadas por el usuario para después enviar los datos a la segunda tarea la cual se encarga de publicar los datos leídos por la primer tarea y suscribir los parámetros definidos por el usuario.

Ambas tareas se ejecutan de forma asíncrona y para evitar colisión de datos entre ambas tareas se utilizan colas (pequeñas pausas) para comunicar información entre una y otra.

- **Sensor de Humedad del Suelo:** mide el nivel de humedad en el sustrato de las plantas. Cuando el nivel de humedad cae por debajo del 50%, el sistema de riego se activa automáticamente hasta alcanzar el 80%.
- **Sensor de Luz:** controla la cantidad de luz disponible para el cultivo.
- **Sensor de Temperatura y Humedad Ambiental:** mide tanto la temperatura como la humedad del aire, asegurando que se mantengan en niveles óptimos para el crecimiento del cultivo.

Los valores de referencia para cada una de estas variables (luz, humedad del suelo y ambiente, y temperatura) se gestionan mediante una tabla de parámetros en la base de datos.

Esta tabla envía al programa en Arduino los valores máximos y mínimos específicos para cada tipo de cultivo, de modo que el sistema pueda ajustar sus acciones en función de las necesidades de cada planta.

Comunicación de Datos y Almacenamiento

La transmisión y gestión de los datos recolectados se realiza a través de MQTTX como broker, empleando el protocolo MQTT. Este protocolo permite una comunicación eficiente entre el Arduino y el sistema en Django. A través de MQTT, los datos de los sensores se publican y suscriben en tiempo real, lo que permite una actualización constante de las variables monitoreadas.

Para almacenar estos datos, implementamos una base de datos MySQL, donde se registran las lecturas cada hora, lo que permite llevar un historial detallado de las condiciones de cada cultivo. Un script en Python conecta el broker MQTT con la base de datos, asegurando que todas las medidas recolectadas se almacenen de manera continua y estructurada.

Estructura de la Interfaz en Django

La interfaz de usuario, desarrollada en Django, permite monitorear y controlar el sistema desde cualquier dispositivo con acceso a internet. La estructura de la aplicación en Django está organizada de la siguiente manera:

- **Views:** aquí se encuentra toda la lógica del programa, comunicación con la API, funcionalidad de los botones, etc.
- **Models:** contiene las definiciones de las tablas en la base de datos, como los cultivos y los parámetros, lo que permite gestionar las configuraciones y los datos de cada planta de manera estructurada.
- **URL:** define todas las rutas del programa, permitiendo al usuario navegar entre las diferentes secciones de la aplicación.
- **Front-End:** se desarrolló utilizando HTML, CSS y JavaScript para crear una interfaz intuitiva y accesible para el usuario.

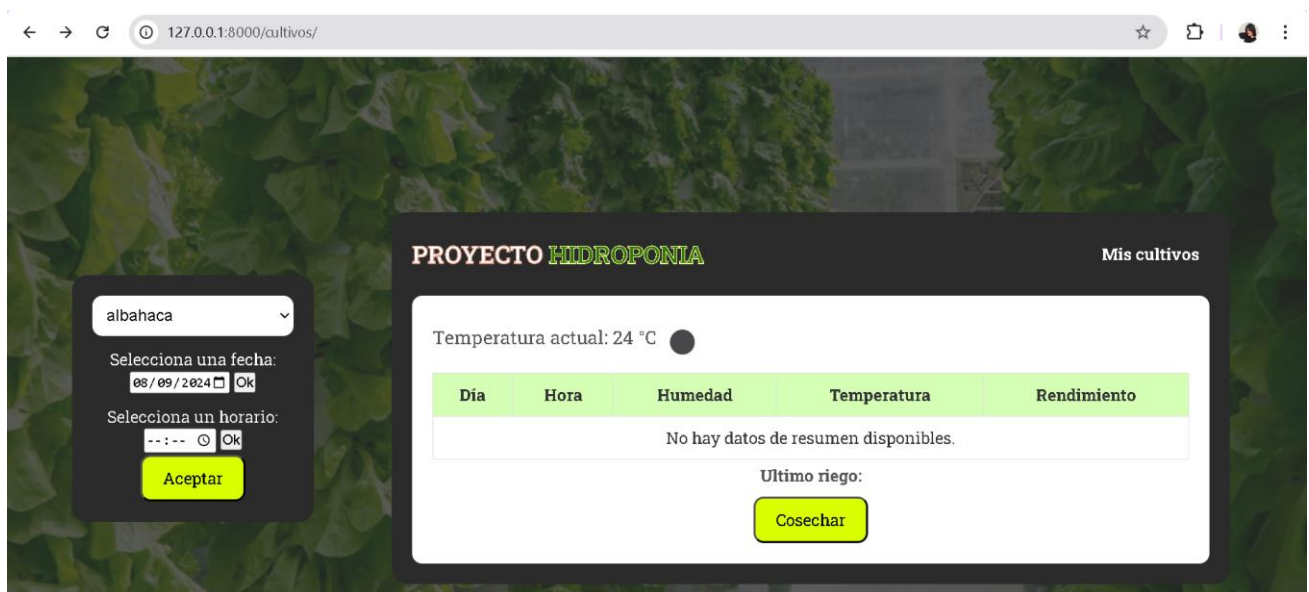
La aplicación tiene dos pantallas principales, cada una con funciones específicas:

1. Pantalla de Resumen

Se pueden filtrar los datos según la fecha y hora registrados y además una nueva funcionalidad que se implementara es poder seleccionar el funcionamiento del riego y lampara para la luz (Manual | Desactivado | Automático).

En esta pantalla, el usuario puede visualizar los datos de un cultivo específico en una fecha y hora determinadas. Además, para complementar la información, integramos la API de OpenWeatherMap que permite consultar el clima en tiempo real. Los datos mostrados en esta pantalla incluyen:

- Día y Hora
- Humedad del Suelo
- Temperatura y Humedad Ambiental
- Nivel de Luz
- Rendimiento del Cultivo
- Botón de “Cosechar”: Permite registrar la recolección de un cultivo cuando esté listo.



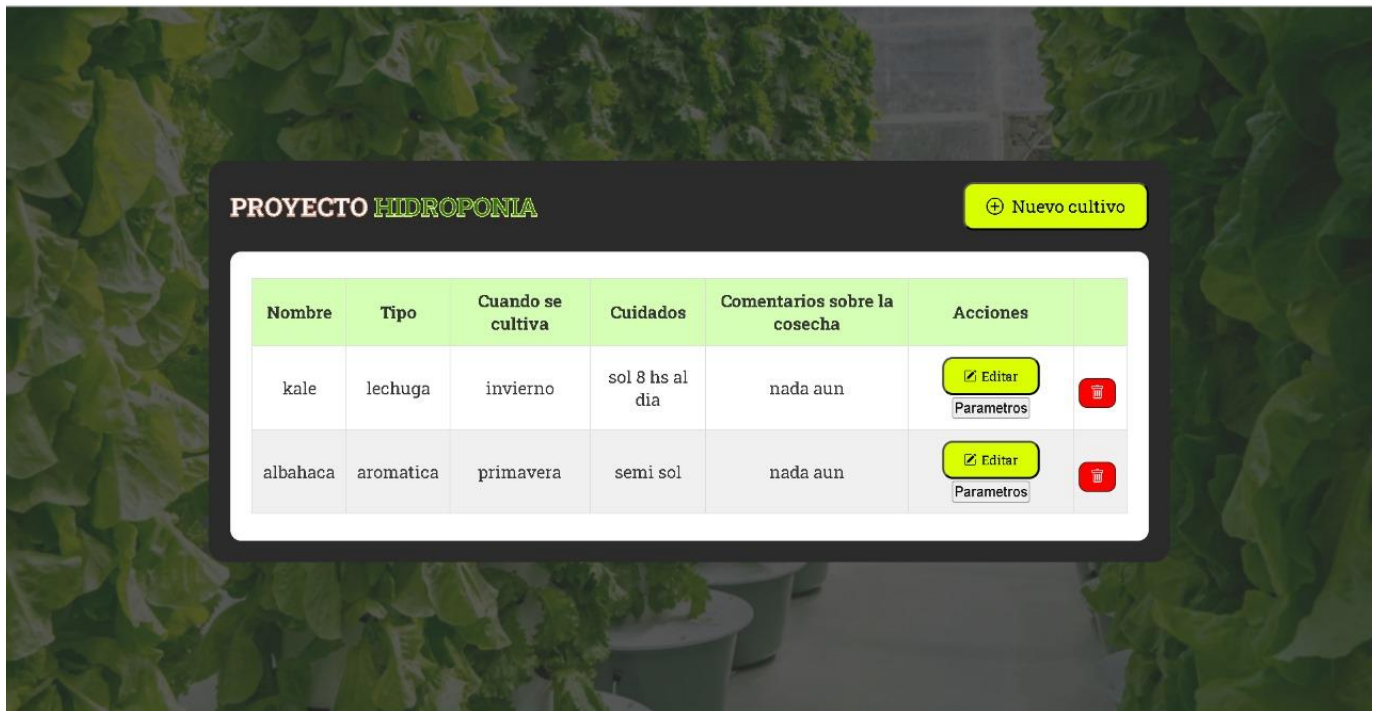
4. Pantalla de Resumen

2. Pantalla de Cultivos

Esta pantalla muestra los diferentes tipos de cultivos registrados, con información detallada de cada uno, incluyendo:

- Nombre del Cultivo
- Tipo de Cultivo

- Época de Siembra
- Cuidados Específicos
- Comentarios: Observaciones adicionales que el usuario puede agregar sobre cada cultivo.
- Botón de “Editar”: Permite modificar la información de cada cultivo registrado.
- Botón de “Nuevo Cultivo”: Permite agregar un nuevo cultivo a la lista, registrando sus necesidades específicas.



5.1 Pantalla de Cultivos

← → ↻ ⓘ 127.0.0.1:8000/cultivos/miscultivos/ 🔍 ☆ 📄 👤 ⋮

Agregar Nuevo Cultivo

Nombre:

Tipo:

Cuando se cultiva:

Cuidados:

Comentario sobre la cosecha:

Guardar

5.2 Pantalla de Agregar nuevo Cultivo

Integración del Sistema con MQTT

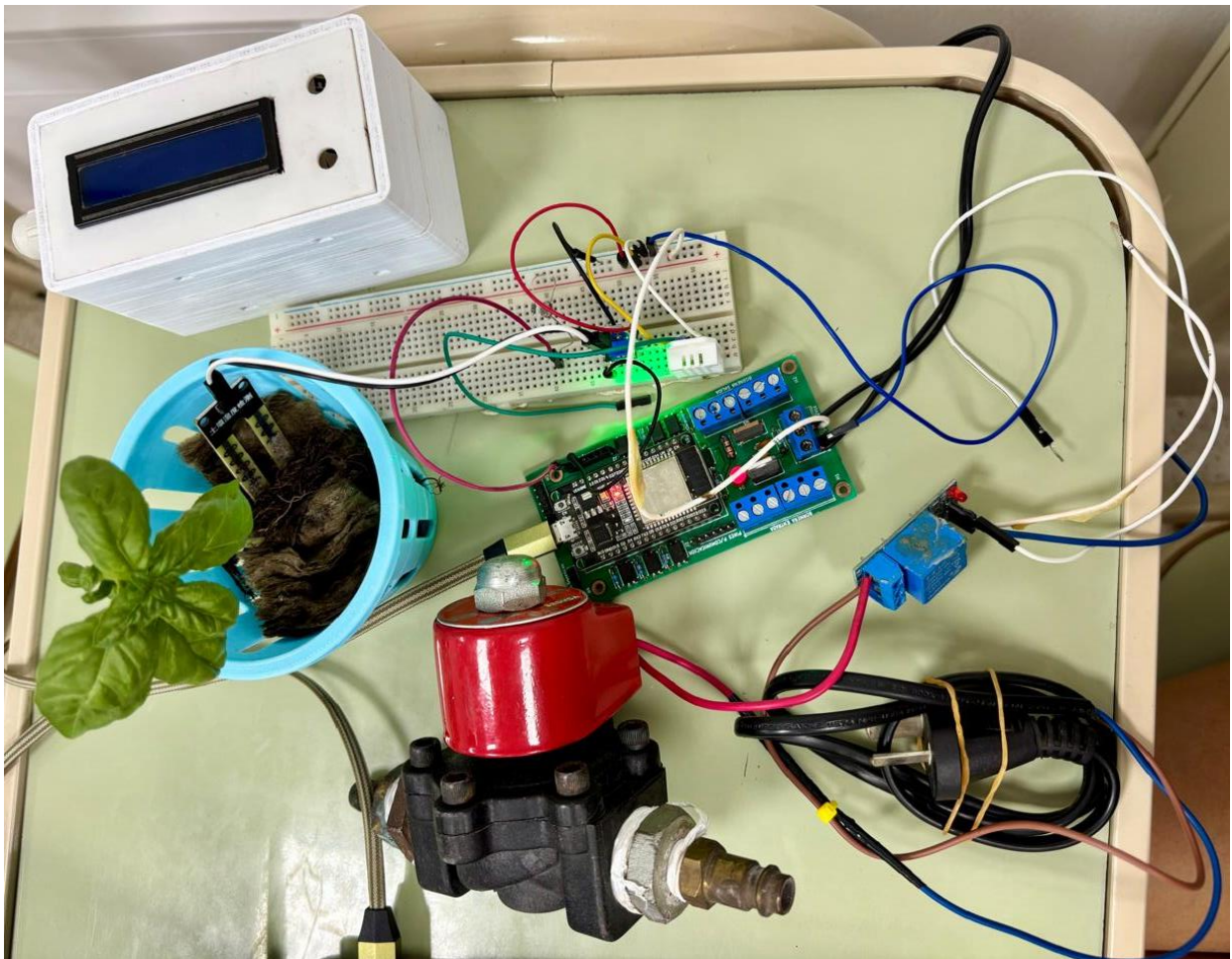
La integración entre el programa en Arduino y la aplicación en Django se gestiona a través del protocolo MQTT. Este protocolo permite conectar ambos sistemas de forma bidireccional, permitiendo que los valores de los sensores se envíen y suscriban en tiempo real. Esto significa que cualquier cambio en las lecturas de los sensores se actualiza automáticamente en la base de datos y se refleja en la interfaz, facilitando un monitoreo constante y preciso.

Esta estructura completa facilita el monitoreo y la administración de cultivos hidropónicos, permitiendo al usuario realizar ajustes y seguir de cerca el crecimiento de las plantas mediante una interfaz intuitiva y accesible. Con el diseño modular del sistema, se pueden realizar futuras ampliaciones e integrar nuevos sensores o funcionalidades de manera eficiente.

Resultados esperados

El sistema diseñado permitirá la automatización completa del riego, lo que mejora la eficiencia en el uso del agua y reduce la necesidad de supervisión constante. A continuación, se destacan los resultados que se esperan obtener:

- **Ahorro de agua:** el sistema automatizado permitirá una reducción de aproximadamente un 20% en el consumo de agua en comparación con un sistema de riego manual.
- **Mayor control:** al tener un monitoreo constante de las variables climáticas y del suelo, lograremos mantener las condiciones óptimas de crecimiento en las plantas durante todo el cultivo.
- **Conectividad remota:** gracias a la integración con MQTT y la API de OpenWeatherMap, el sistema puede ser controlado y monitoreado desde cualquier lugar, lo que agrega comodidad y flexibilidad al usuario.



6. Fotografía de la primera versión hardware

Conclusión

El proyecto de controlador de hidroponía automatizado para la materia "Comunicaciones de Datos" nos permitió aplicar conceptos de IoT en un contexto real y resolver una problemática de agricultura moderna. Mediante el uso de sensores, actuadores y un sistema de control eficiente, logramos desarrollar una solución que optimiza el uso de recursos, mejora las condiciones de cultivo y reduce el tiempo de supervisión requerido por el usuario.

El proyecto no solo cumplió con los objetivos iniciales, sino que también deja espacio para futuras mejoras, como la integración de modelos predictivos que, basados en los datos históricos, puedan anticipar las necesidades de las plantas y optimizar aún más el sistema.

