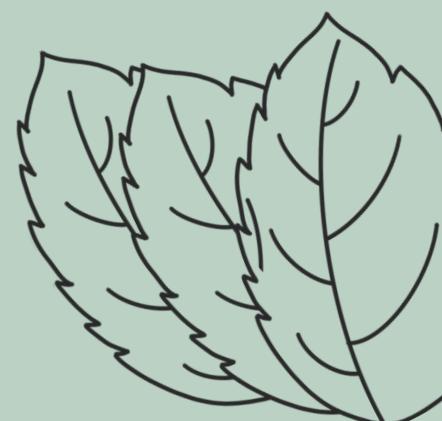


IOT

# PROYECTO MACETA IOT

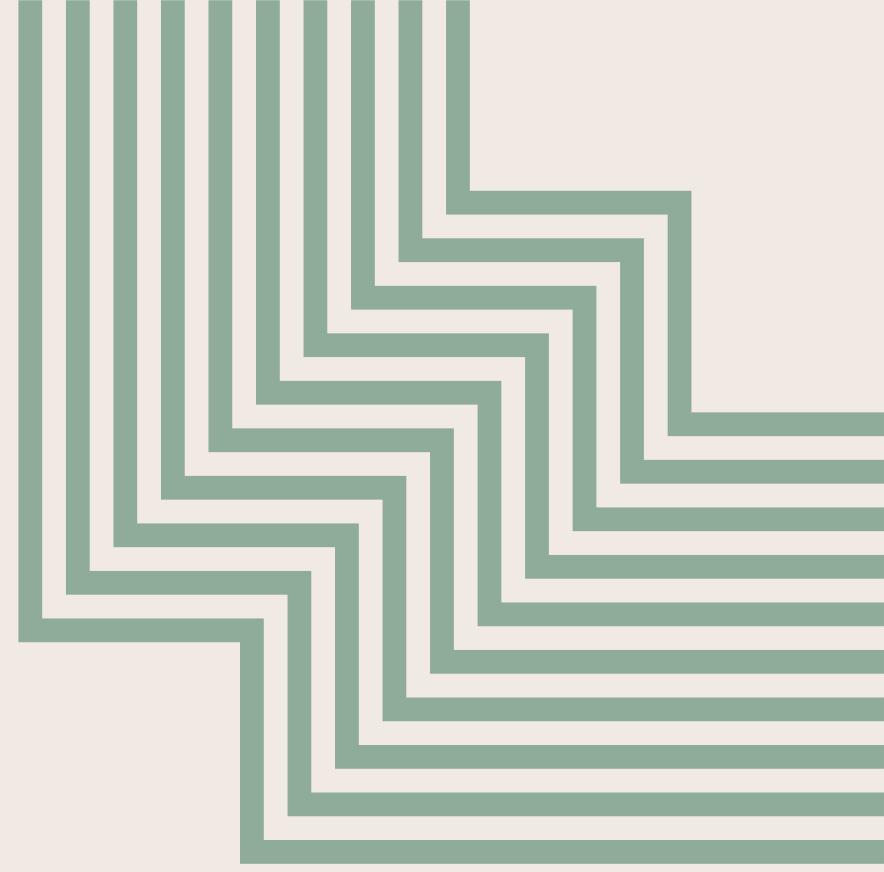
Yessica Lora Vazquez  
Monica Sarahi Flores Lara  
Sebastián Acosta Marín  
Gabriel Esperilla León

# INSTRUMENTOS DE EVALUACIÓN



# ÍNDICE

1. Planteamiento del problema.
2. Alcance del proyecto
3. Planeación
4. Planteamiento económico
5. Diseño
6. Referencias bibliográficas



# INTRODUCCIÓN

Este proyecto tiene como objetivo aplicar tecnologías de Internet de las Cosas (IoT) para monitorear y controlar variables críticas en el crecimiento de plantas.

El uso de IoT en la agricultura de precisión ha demostrado mejorar el rendimiento de los cultivos, reduciendo el desperdicio de agua y energía.





# OBJETIVOS

1

## IOT/ Completo

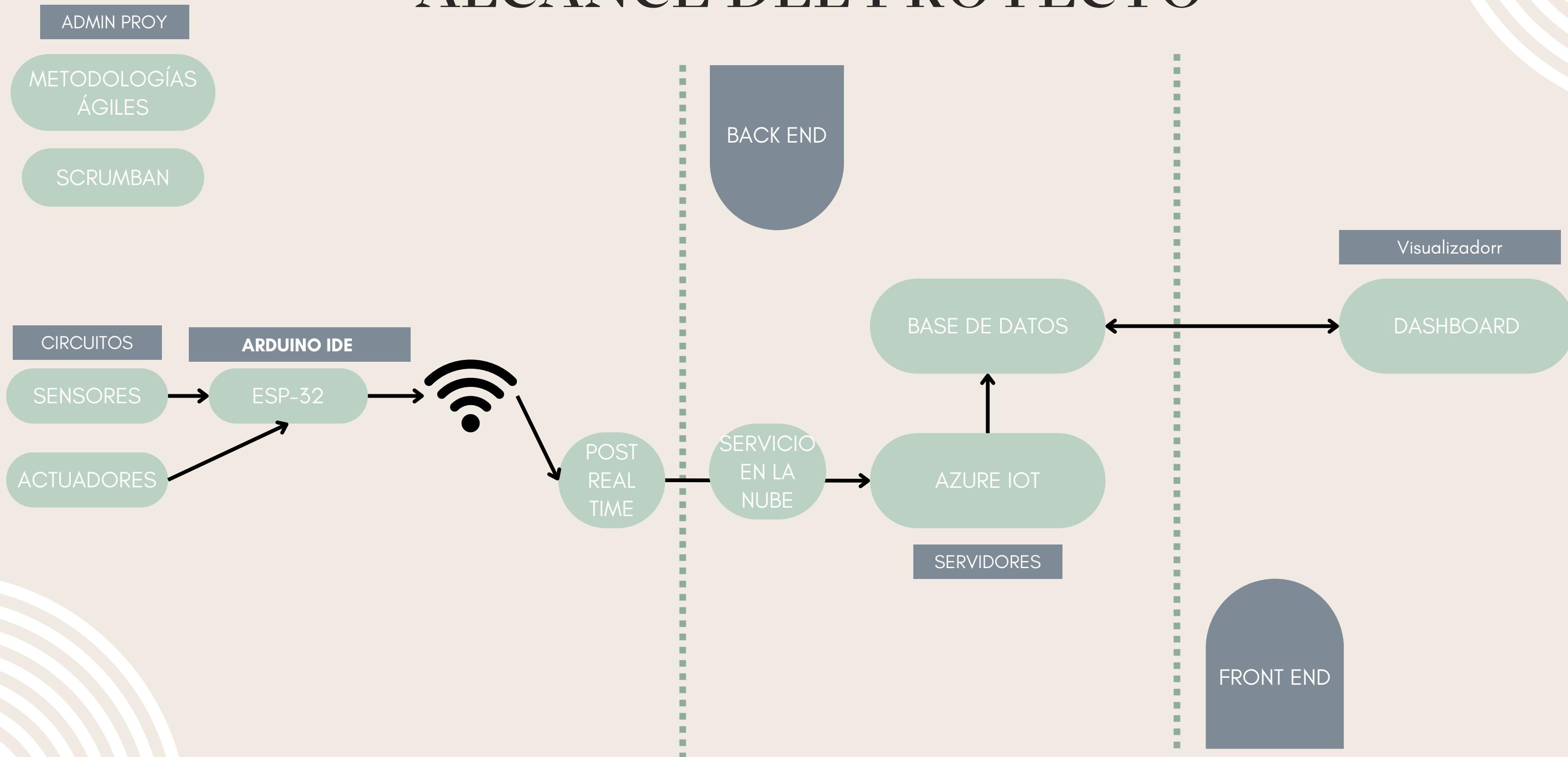
Desarrollar un sistema IoT que monitoree y controle automáticamente las variables ambientales de una planta en maceta así como visualizar los datos en tiempo real mediante dashboards interactivos.

2

## Comparación y Documentar / Incompleto

Comparar el rendimiento de una planta en un entorno controlado versus una en un entorno natural

# ALCANCE DEL PROYECTO



# PLANTA SELECCIONADA

La planta que como equipo seleccionamos fue la de menta debido a las siguientes razones:



# VARIABLES A CONTROLAR

## Temperatura

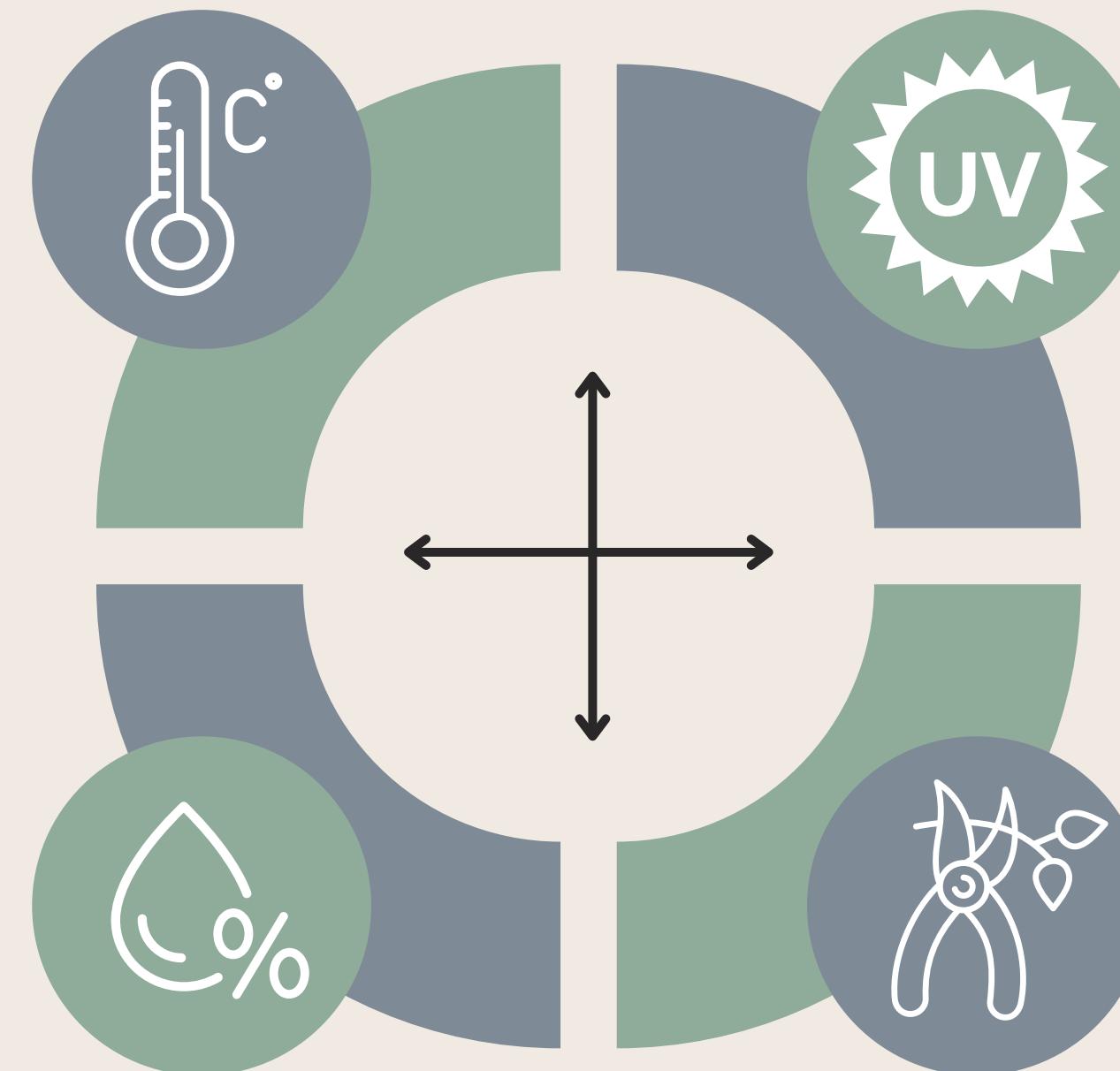
Controla el calor ambiental que puede afectar el crecimiento.

- Siendo una temperatura optima de 15°C a 25°C.

## Humedad

Monitorea la cantidad de agua en el en la tierra.

- Los maceteros pueden secarse más rápidamente que la tierra del jardín.
- Rango optimo del 50% a 70% de humedad



## Radiación solar

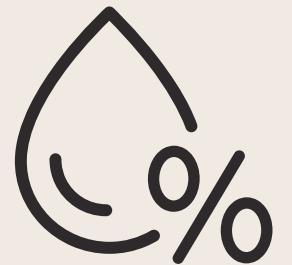
Asegura que las plantas reciban la luz adecuada.

Necesitan sol pleno y directo un rango optimo seria de 10,000 a 25,000 lux

## Poda

Se planeaba controlar esta variable, sin embargo por temas de la implementación del software no pudimos visualizar un cronometro en el dashboard

# SENSORES DE HUMEDAD



## Funcionamiento

Utiliza principios capacitivos para medir la humedad del suelo, evitando la corrosión que afecta a los sensores resistivos. A medida que el contenido de agua en el suelo cambia, también lo hace su capacidad. Este cambio se convierte en una señal analógica, lo que permite monitorear con precisión el nivel de humedad.

## Ventajas

Más duradero: No tiene partes metálicas expuestas, lo que lo hace inmune a la corrosión.

Ofrece valores de voltaje que pueden ser definidos como porcentaje de humedad y no solo cero y uno.

## Calibración

Se realizaron pruebas en dos condiciones: en aire (suelo seco) y sumergido en agua (suelo completamente húmedo). Los valores extremos de estas pruebas se usaron como referencia, asignando un 0% de humedad en el aire y un 100% cuando estaba sumergido, lo que asegura mediciones confiables de la humedad del suelo.

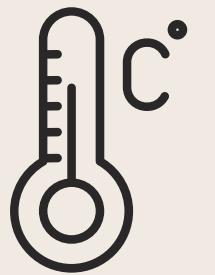
## Rango óptimo

50% - 70% humedad

Sensor Capacitivo de  
Humedad del Suelo  
V2.0 184 MXN



# SENSORES DE TEMPERATURA



## Funcionamiento

Utiliza un diodo de silicio, que convierte la temperatura en una señal digital. Este sensor es capaz de medir temperaturas en un rango amplio, de -55°C a 125°C, y tiene una precisión de ±0.5°C.

Mide la temperatura a través de la caída de voltaje a través del diodo.

## Ventajas

- Mayor precisión y capacidad.
- Configuración más compleja.
- Resistencia al agua.
- **Alta precisión y confiabilidad** en entornos extremos, como sistemas de refrigeración o invernaderos.

## Calibración

Se utilizó la temperatura ambiente como referencia, y las mediciones mostraron una variación mínima de medio grado a un grado, lo que indicó que el sensor funcionaba correctamente y no necesitaba ajustes adicionales.

## Rango óptimo

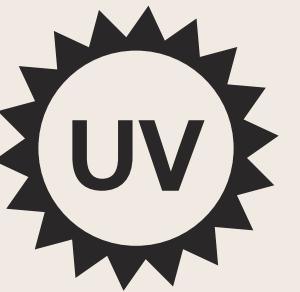
15°C a 25°C Temperatura

DS18B20

79 MXN



# SENSORES DE LUZ



## Funcionamiento

El BH1750 es un sensor digital que mide la intensidad lumínica en lux y transmite los datos mediante el protocolo I2C. Proporciona una medición precisa, y su salida digital permite una integración sencilla con microcontroladores sin necesidad de un ADC.

## Ventajas

- Precisión Alta
- Al ser una señal digital es menos susceptible a interferencias.
- Excelente para proyectos IoT que requieren medición precisa y constante de luz ambiental.

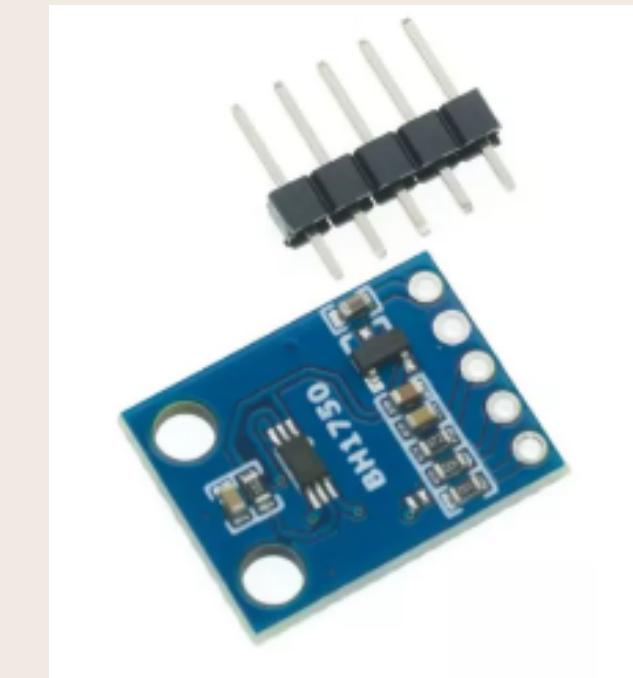
## Calibración

Se usó la linterna de un teléfono móvil como fuente de luz y, tras buscar el valor aproximado de su intensidad, se ajustó el sensor hasta que mostró un valor cercano al esperado, logrando una calibración precisa para medir la intensidad de luz de manera confiable.

## Rango óptimo

10,000 - 25,000 lux

BH1750  
53 MXN



# ESP-32



Microcontrolador/ Características	ESP32	Arduino Uno	Raspberry Pi Pico W
Características principales	<p><b>Conectividad:</b> Wi-Fi y Bluetooth integrados, ideal para proyectos IoT.</p> <p><b>Puertos:</b> Ofrece múltiples pines GPIO y soporte para I2C, UART, SPI (necesario para conectar sensores como el BH1750, DS18B20, y el HC-SR04).</p>	<p><b>Conectividad limitada:</b> No tiene Wi-Fi integrado, pero se puede ampliar con módulos externos.</p> <p>Fácil de usar y programar</p>	<p><b>Conectividad Wi-Fi integrada</b></p> <p>Compatible con múltiples protocolos (I2C, SPI, UART, GPIO)</p>
Diferencias	<p>16 pines ADC para entrada analógica más entradas analógicas. Esto le otorga mayor flexibilidad para conectar múltiples sensores y actuadores</p> <p>Bluetooth incluido</p> <p>Bajo consumo energético</p>	<p>Necesita módulos adicionales para Wi-Fi, aumentando la complejidad</p> <p>Lo cual afecta mucho al desarrollo de proyectos de IOT, ya que el wifi, es un requisito esencial</p>	<p>Solo cuenta con tres pines de entrada analógica.</p> <p>Bluetooth no disponible</p>
Conclusión	<p>El ESP32 es ideal para tu proyecto porque permite conectar varios sensores, enviar datos en tiempo real a la nube y controlar el sistema a distancia mediante Wi-Fi.</p>	<p>Arduino es excelente para prototipos básicos, pero para un sistema avanzado de IOT, no es suficiente sin WIFI</p>	<p>Aunque la Raspberry Pi Pico W es una opción viable, la ESP32 ofrece más flexibilidad, conectividad y recursos, lo que la convierte en una solución más práctica</p>

Aunque ambos microcontroladores ofrecen un número suficiente de pines para la mayoría de los proyectos, la ESP32 tiene una ventaja al ofrecer más GPIO y más entradas analógicas. Esto le otorga mayor flexibilidad para conectar múltiples sensores y actuadores.

# AZURE IOT

Azure IoT (Internet of Things) es un servicio de la nube de Microsoft diseñado para conectar, monitorear y administrar dispositivos IoT de manera segura. Ofrece una infraestructura que permite a dispositivos como la ESP32 enviar datos, recibir comandos y gestionar propiedades, facilitando el monitoreo y control de sistemas remotos.

El Azure IoT Hub es la pieza central del ecosistema IoT en Azure. Funciona como un intermediario entre dispositivos físicos (como sensores y actuadores conectados a una ESP32) y las aplicaciones en la nube



# CÓDIGO PRINCIPAL

Los datos de los sensores conectados a la ESP32 se formatean como un objeto JSON usando la función generate\_telemetry\_payload:

```
static int generate_telemetry_payload(
    uint8_t* payload_buffer,
    size_t payload_buffer_size,
    size_t* payload_buffer_length);
static int generate_device_info_payload(
    const az_iot_hub_client* hub_client,
    uint8_t* payload_buffer,
    size_t payload_buffer_size,
    size_t* payload_buffer_length) {
    az_json_writer jw;
    az_result rc;
    az_span payload_buffer_span = az_span_create(payload_buffer, payload_buffer_size);

    Serial.println("Iniciando generación de payload...");

    rc = az_json_writer_init(&jw, payload_buffer_span, NULL);
    if (az_result_failed(rc)) {
        Serial.println("Error al inicializar el escritor JSON.");
        return RESULT_ERROR;
    }

    rc = az_json_writer_append_begin_object(&jw);
    if (az_result_failed(rc)) {
        Serial.println("Error al abrir el objeto JSON.");
        return RESULT_ERROR;
    }
```

# CÓDIGO PRINCIPAL

Estos datos JSON se envían al IoT Hub mediante la función `azure_pnp_send_telemetry`:

```
int azure_pnp_send_telemetry(azure_iot_t* azure_iot) {
    Serial.println("Entrando a azure_pnp_send_telemetry...");
    int result;
    size_t payload_length;

    result = generate_telemetry_payload(data_buffer, DATA_BUFFER_SIZE, &payload_length);
    if (result != RESULT_OK) {
        Serial.println("Error generando payload de telemetría.");
        return RESULT_ERROR;
    }
    Serial.println("Payload de telemetría generado con éxito:");

    // Imprime el payload generado (para verificar qué se enviará)
    Serial.println((char*)data_buffer);

    sendJSONToPHP(data_buffer, payload_length);

    result = azure_iot_send_telemetry(azure_iot, az_span_create(data_buffer, payload_length));
    if (result != 0) {
        Serial.println("Error enviando telemetría.");
        return RESULT_ERROR;
    }
    Serial.println("Telemetría enviada con éxito.");
    return RESULT_OK;
}
```

# DASHBOARD AZURE IOT

- Los datos enviados al IoT Hub se procesan y se muestran automáticamente en el dashboard, cada vez que la ESP32 envía nuevos datos, el dashboard se actualiza.





# ACTUADORES



## Control Automático de Humedad (Riego Automatizado)

### Funcionamiento:

Cuando el sensor detecta niveles bajos de humedad, la Mini Bomba se abre para permitir el flujo de agua. Una vez alcanzado el nivel de humedad óptimo, se apaga automáticamente.

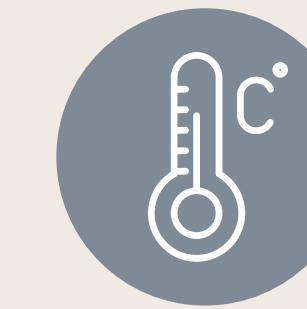
### Componentes involucrados:

- Sensor capacitivo conectado al microcontrolador ESP32 para leer datos y activar la válvula
- Mini Bomba de Agua 3v-6v Sumergible
- Porta pilas de 2 pilas 18650:

**Rango óptimo: 50% - 70%**



# ACTUADORES



## Actuador para el Control de Temperatura (Automatico)

### Funcionamiento:

El control de la temperatura se gestionará de manera asistida por un sensor DS18B20 que activará un ventilador en caso de que la temperatura no sea la óptima

### Componentes involucrados:

- Sensor DS18B20 para monitoreo continuo.
- Ventilador de 5Ve
- Porta pilas de 2 pilas 18650:

Rango óptimo: 15°C a 25°C.



# ACTUADORES



## Actuador para el Control de Luz Solar (Automatico )

### Funcionamiento:

El sistema de luz monitorea la intensidad solar mediante el sensor BH1750. Cuando se detecte un nivel de luz solar excesiva, el sistema activará un servomotor de manera automática, el cual cerrará una compuerta para evitar el paso de la radiación solar, en caso contrario, la compuerta se abrirá para permitir el paso de la luz.

### Componentes involucrados:

- Sensor BH1750 para medición precisa de luz en lux.
- Servomotor SG90 RC 9g
- Porta pilas de 2 pilas 18650:

**Rango óptimo: 10,000 - 25,000 lux**

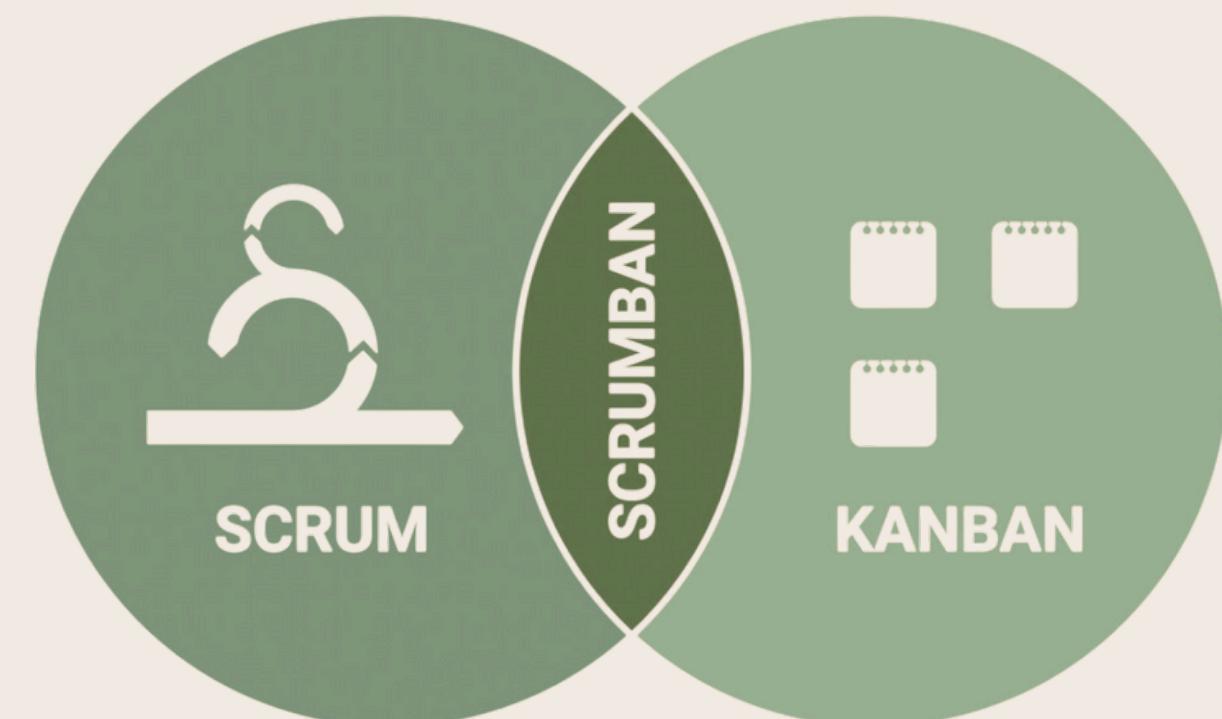


# METODOLOGÍA

El proyecto tiene una duración de 10 semanas y se decidió utilizar la metodología de Scrumban ya que la flexibilidad de Kanban nos permite ajustarnos a los cambios mientras mantenemos la organización y planificación de Scrum

## Scrumban

- Ofrece una estructura para planificar y organizar el trabajo en sprints cortos, con reuniones regulares para evaluar el progreso.
- Proporciona un sistema visual para gestionar tareas y permite un flujo continuo de trabajo, con más flexibilidad que Scrum puro.



# PLANEACIÓN

[Link a Microsoft Planner](#)

Por Hacer	En Proceso	Hecho	Reuniones Semanales
<a href="#">+ Agregar tarea</a>	<a href="#">+ Agregar tarea</a>	<a href="#">+ Agregar tarea</a>	<a href="#">+ Agregar tarea</a>
	<p>Evaluación oral y escrita 1 FINAL</p> <p>Vencimiento YY GL M +1</p> <p>Tareas completadas 1</p>	<p>Tareas completadas 9</p> <p>Investigación y Pruebas de Sensores Durante estas semanas se realizaron pruebas iniciales con los sensores seleccionados para garantizar su correcto funcionamiento</p> <p>Reunión Final-8-Serum Completada por Yessica Lora Váz...</p> <p>Desarrollo del Código e Integración de Sensores Se trabajó en la programación del sistema para que los sensores recojan y almacenen datos de manera automática</p> <p>Reunión 7-Serum Completada por Yessica Lora Váz...</p> <p>Reunión 6-Serum Completada por Yessica Lora Váz...</p> <p>Reunión 5-Serum Completada por Yessica Lora Váz...</p> <p>Reunión 4-Serum Completada por Yessica Lora Váz...</p> <p>Reunión 3-Serum Completada por Yessica Lora Váz...</p> <p>Reunión 2-Serum Completada por Yessica Lora Váz...</p> <p>Reunión 1-Serum Completada por Yessica Lora Váz...</p>	<p>Tareas completadas 8</p> <p>Reunión Final-8-Serum Completada por Yessica Lora Váz...</p> <p>Reunión 7-Serum Completada por Yessica Lora Váz...</p> <p>Reunión 6-Serum Completada por Yessica Lora Váz...</p> <p>Reunión 5-Serum Completada por Yessica Lora Váz...</p> <p>Reunión 4-Serum Completada por Yessica Lora Váz...</p> <p>Reunión 3-Serum Completada por Yessica Lora Váz...</p> <p>Reunión 2-Serum Completada por Yessica Lora Váz...</p> <p>Reunión 1-Serum Completada por Yessica Lora Váz...</p>

# PLANTEAMIENTO ECONÓMICO

## Hardware

Protoboard, placa de desarrollo, material de cableado: **150 MXN**

### Sensores

- Sensor Capacitivo de Humedad del Suelo V2.0 **184 MXN**
- DS18B20 (temperatura: **79 MXN**
- BH1750 (radiación solar): **53 MXN.**

### Actuadores

- Mini Bomba de Agua 3V-6V Sumergible: **80 - 120 mxn**
- Servomotor SG90 RC 9g: **40 - 80 mxn**
- ventilador 5V para Raspberry Pi (30x30mm): **60 - 120 mxn**

## Software

- Azure IOT **Proporcionado por el profesor**
- IDE de Arduino: **Gratis**

## Extras

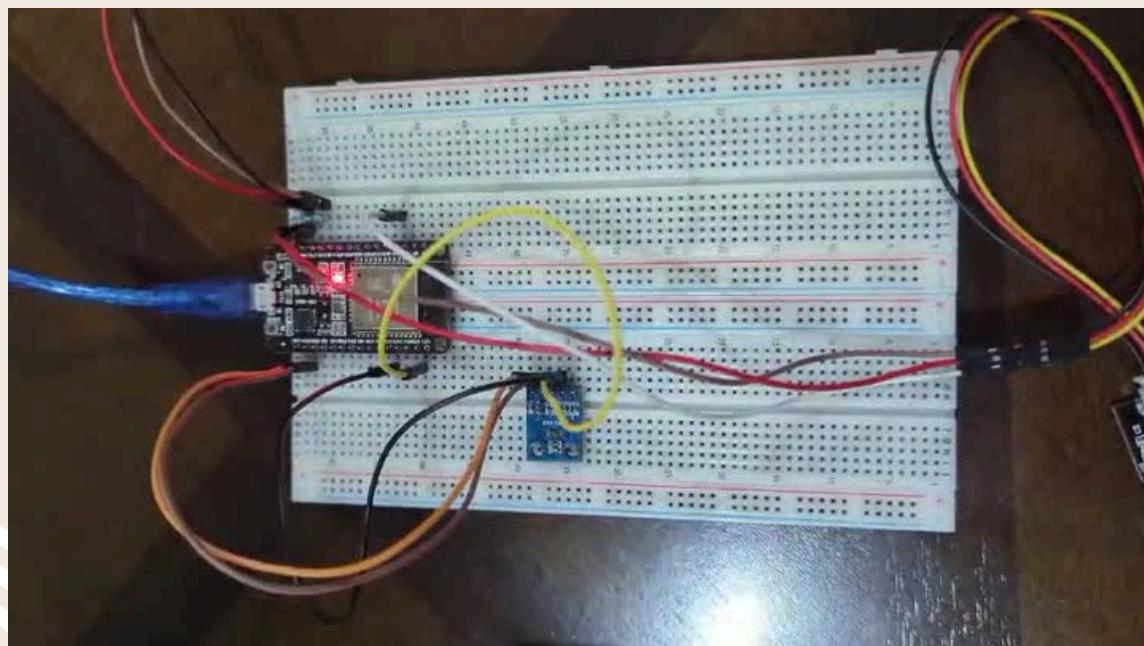
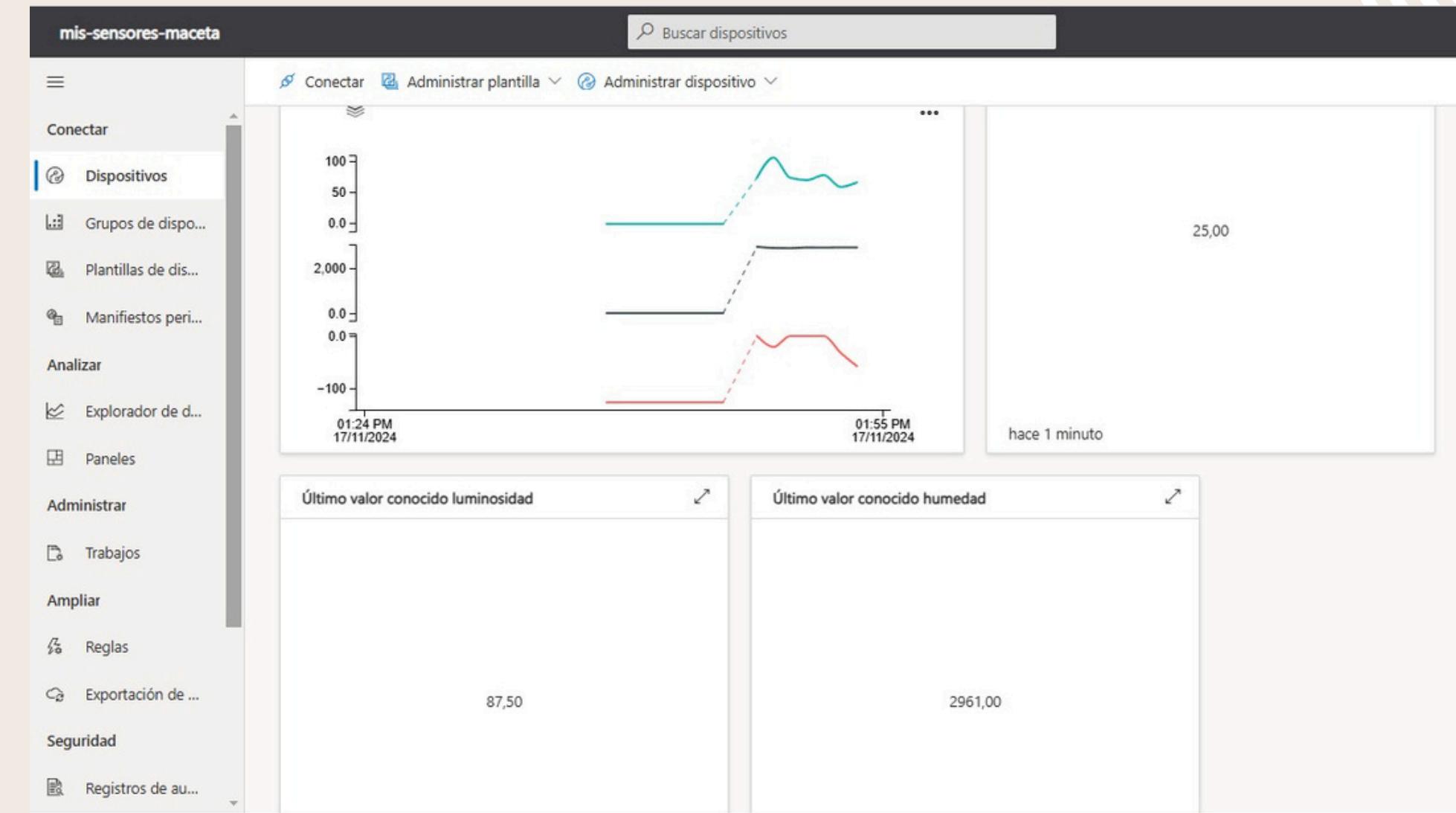
- Brote de planta: **100 mxn**
- Dos Placas de MDF de 1.22\* 61 CM **180 mxn**

Presupuesto estimado: Entre **1000** con un límite de **1500 MXN**  
Gastado **1066 mx**

# DISEÑO

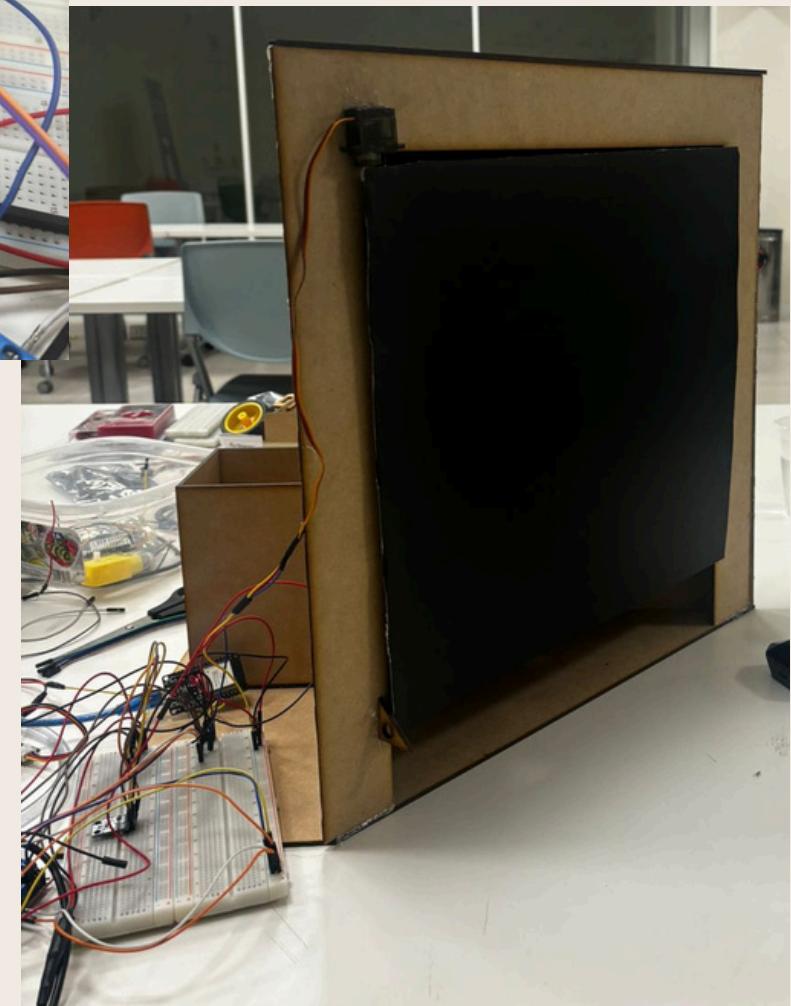
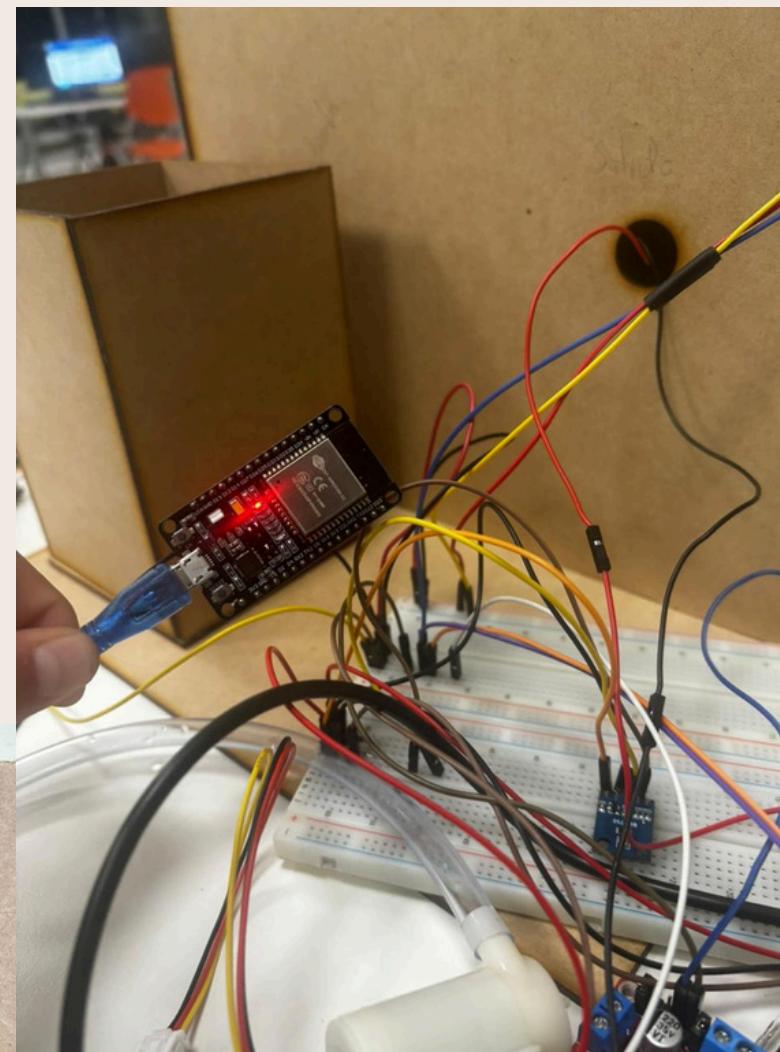
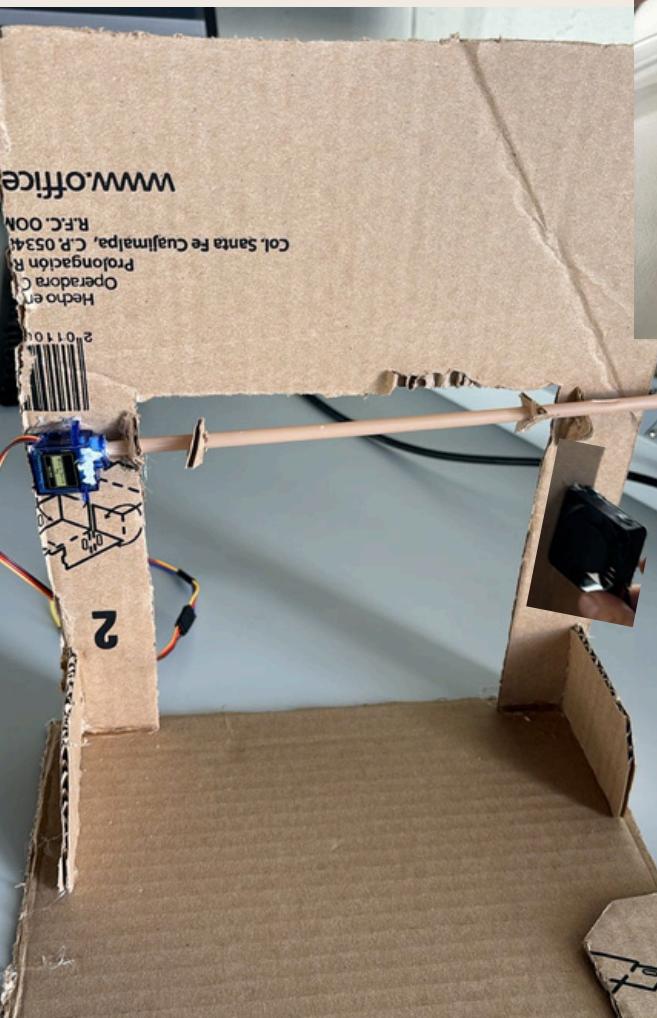
## Software

- Azure IOT



## Hardware

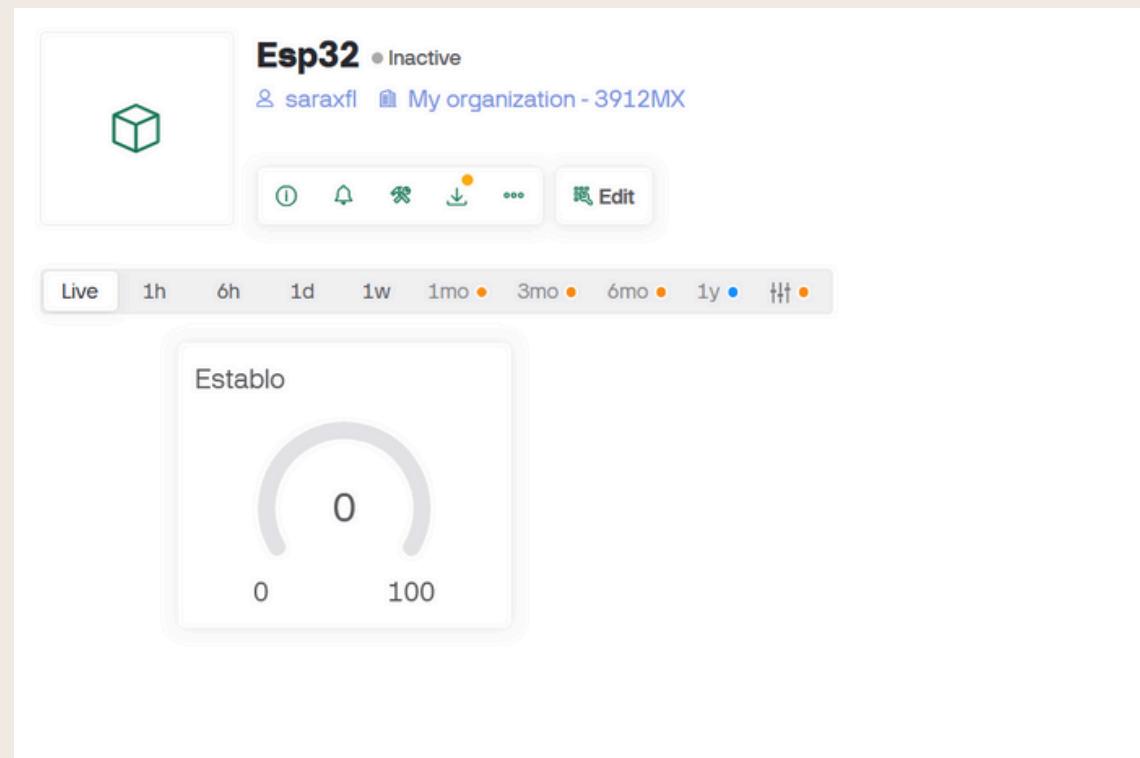
Diseñamos e imprimimos en MDF una base para colocar la maceta junto con los sensores y actuadores, garantizando un montaje estable y funcional.



# BLYNK

Es una plataforma de IoT que permite a los usuarios crear interfaces gráficas personalizadas para interactuar con dispositivos conectados de manera sencilla. Es compatible con una amplia gama de microcontroladores y placas de desarrollo, incluyendo el ESP32.

Tratamos de conectarnos configurando blynk y usando un código en Arduino , pero a pesar de seguir todos los pasos, al final no se logró generar una conexión correcta por lo cual los datos no podían ser visualizados en el dashboard.



Devices				
<input type="text"/> Start typing				
<input type="button"/> Add Filter				
All	1	...	My devices	
<input type="checkbox"/>	Name	Auth Token	Device Owner	Status
	esp32	fEz7d3vikqxBw3vxCbgxNDMkrvt...	a07045944@tec.mx (you)	Inactive

# BASE DE DATOS

Dentro de Azure generamos un servicio de mysql para tener un registro histórico de los datos organizados por fecha de registro. Al principio habíamos logrado generar con éxito la base de datos y generar una conexión con el equipo, pero al agregar una función al código para mandar el json a un php que se encargara de leerlos y mandarlas a la base de datos no se logró generar una conexión.

The screenshot shows the Azure portal interface for a MySQL server named 'macera-server-r'. The left sidebar has a tree view with 'Información general' selected. The main panel displays essential information about the server, including its subscription (Azure subscription 1), resource group (Maceta\_IOT), location (Canada Central), and creation date (2024-11-14). It also shows the server's name (macera-server-r.mysql.database.azure.com), session ID (kkwgcbcne), configuration (Con capacidad de ráfaga, B1ms, 1 núcleos virtuales, 2 GiB de RAM 2...), MySQL version (8.0), and availability zone (1).

```
<?php
// Configuración de la base de datos
$host = 'macera-server-r.mysql.database.azure.com';
$dbname = 'macera-database'; // Cambia esto por el nombre de tu base de datos
$username = 'kkwgcbcne'; // Cambia esto por tu usuario de la base de datos
$password = 'MJASP$3BfgTmuaePg'; // Cambia esto por tu contraseña

// Conexión a la base de datos
$conn = new mysqli($host, $username, $password, $dbname);
if ($conn->connect_error) {
    die("Error de conexión: " . $conn->connect_error);
}

// Leer el contenido del cuerpo de la solicitud (JSON)
$json = file_get_contents('php://input');
$data = json_decode($json, true); // Decodifica el JSON recibido en un arreglo asociativo

// Mapear datos desde el JSON recibido
$temperatura = isset($data['temperature']) ? $data['temperature'] : null; // Mapear "temperature" a "Temperatura"
$humedad = isset($data['soilMoisture']) ? $data['soilMoisture'] : null; // Mapear "soilMoisture" a "Humedad"
$luz_solar = isset($data['lightLevel']) ? $data['lightLevel'] : null; // Mapear "lightLevel" a "Luz_solar"
$altura_poda = 0.0; // Valor por defecto, ya que no está incluido en tu JSON original
$fecha = date('Y-m-d H:i:s'); // Asignar fecha actual del servidor

// Verificar que los campos requeridos no estén vacíos
if ($temperatura === null && $humedad === null && $luz_solar === null) {
    $stmt = $conn->prepare(
        "INSERT INTO monitoreo (Fecha, Humedad, Temperatura, Luz_solar, Altura_poda)
        VALUES (?, ?, ?, ?, ?)"
    );
    $stmt->bind_param("sddd", $fecha, $humedad, $temperatura, $luz_solar, $altura_poda);

    if ($stmt->execute()) {
        echo "Datos guardados correctamente.";
    } else {
        echo "Error al guardar los datos: " . $stmt->error;
    }
}
```

```
void sendJSONToPHP(const uint8_t * payload, size_t payload_length) {
    HTTPClient http;

    http.begin(serverUrl); // Configura la URL del servidor
    http.addHeader("Content-Type", "application/json"); // Establece la cabecera HTTP

    // Envía el JSON como POST
    int httpResponseCode = http.POST((uint8_t*)payload, payload_length);

    if (httpResponseCode > 0) {
        Serial.print("Datos enviados al servidor PHP. Respuesta HTTP: ");
        Serial.println(httpResponseCode);
        String response = http.getString();
        Serial.println("Respuesta del servidor:");
        Serial.println(response);
    } else {
        Serial.print("Error al enviar datos al servidor PHP. Código HTTP: ");
        Serial.println(httpResponseCode);
    }

    http.end(); // Cierra la conexión HTTP
}
```

# CONCLUSIONES

Los resultados obtenidos fueron satisfactorios, ya que logramos completar la maceta inteligente dentro del plazo establecido. Este éxito fue el resultado de una planificación detallada, trabajo colaborativo y un enfoque claro en cada fase del proyecto. Desde la selección de componentes adecuados, como sensores de temperatura, humedad del suelo y radiación solar, hasta la integración con el sistema IoT.

La implementación de los sensores permitió monitorear las variables críticas de la planta de menta de manera precisa, lo que facilitó ajustes automáticos al sistema, como el control del riego y la apertura de la puerta para la luz solar. El uso del microcontrolador ESP32 y la recolección de datos en tiempo real fueron claves para la toma de decisiones automáticas.



# REFERENCIAS BIBLIOGRÁFICAS

Jecrespom. (2024, 20 marzo). Proyectos IoT. Aprendiendo Arduino.  
<https://aprendiendoarduino.wordpress.com/2024/03/20/proyectos-iot/>

Proyectos con Arduino. (2024, 27 agosto).  
 Sensores Arduino / Proyectos con Arduino. Proyectos Con Arduino.  
<https://proyectosconarduino.com/sensores/>

9 consejos para cultivar tomates cherry orgánicos caseros todo el año. (2024, 5 julio). Ecoinventos.  
<https://ecoinventos.com/consejos-cultivar-tomates-cherry-organicos TODO-el-ano/>

Geek Factory. (s. f.). Sensores para proyectos con Arduino y microcontroladores - Geek Factory.  
<https://www.geekfactory.mx/categoría-de-producto/sensores/>

Roser Castelló (2020)  
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec nec efficitur lacus, quis blandit turpis. Vestibulum dictum varius ipsum. Duis fermentum rutrum pulvinar.

Lola Sánchez (2019)  
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec nec efficitur lacus, quis blandit turpis. Vestibulum dictum varius ipsum. Duis fermentum rutrum pulvinar.

¿Qué es MySQL? (s. f.). Oracle México.  
<https://www.oracle.com/mx/mysql/what-is-mysql/>

LeGeek Factory. (s. f.). Sensores para proyectos con Arduino y microcontroladores - Geek Factory. <https://www.geekfactory.mx/categoría-de-producto/sensores/>

Humanizada, T. (2023, 31 agosto). Blynk, plataforma de internet de las cosas en la red.  
Tecnología Humanizada. <https://humanizationoftechnology.com/blynk-plataforma-de-internet-de-las-cosas-en-la-red/revista/2018/volumen-4-2018/11/2018/>

Jesús. (2024, 31 marzo). MySQL: características, ventajas y desventajas.  
Tutoriales Dongee. <https://www.dongee.com/tutoriales/mysql-características-ventajas-y-desventajas/>

De Souza, I. (2021, 12 febrero). Descubre qué es el lenguaje de programación PHP y en qué situaciones se hace útil. Rock Content - ES.  
<https://rockcontent.com/es/blog/php/>

¿Qué permite el Internet de las cosas en el agro? | BASF México. (s. f.).  
<https://agriculture.bASF.com.mx/es/contenidos-de-agricultura/digitalizacion-iot>

Telcel. (s. f.). IoT en la agricultura de precisión | Telcel Empresas. Telcel.  
<https://www.telcel.com/empresas/tendencias/notas/como-funciona-iot-en-agricultura-deprecision#:~:text=La%20aplicaci%C3%B3n%20de%20IoT%20est%C3%A1%20regida%20por%20tecnolog%C3%ADas%20inteligentes%20automatizadas.>

Tutorial sensor de temperatura y humedad DHT11 y DHT22. (s. f.). Naylamp Mechatronics – Perú.  
[https://naylampmechatronics.com/blog/40\\_tutorialsensor-de-temperatura-y-humedad-dht11-y-dht22.html](https://naylampmechatronics.com/blog/40_tutorialsensor-de-temperatura-y-humedad-dht11-y-dht22.html)

Módulo sensor de luz LM393 de 3 pines. (2021, abril 28). Tesla Electronic EIRL.  
<https://www.teslaelectronic.com.pe/producto/modulo-sensor-de-luz-lm393-de-3-pines/>

Display Oled Azul y Blanco 128x64 0.96" I2C SSD1306. (s/f). UNIT Electronics. Recuperado el 3 de octubre de 2024, de <https://uelectronics.com/producto/display-oled-azul-y-blanco-128x64-0-96-i2c-ssd1306/?srsltid=AfmBOoqk-2pH3UrKmWnBsCKD86i57l85aYLtgsgju1RmsbnMcEQovstl>

Display Oled Azul y Blanco 128x64 0.96" I2C SSD1306. (s/f). UNIT Electronics. Recuperado el 3 de octubre de 2024, de <https://uelectronics.com/producto/display-oled-azul-y-blanco-128x64-0-96-i2c-ssd1306/?srsltid=AfmBOoqk-2pH3UrKmWnBsCKD86i57l85aYLtgsgju1RmsbnMcEQovstl>



MUCHAS  
GRACIAS

IOT