

**GRUPO 7**

**INTEGRANTES:**

Sergio Duarte

Mateo Fux

Felipe Gil

Emanuel Camacho

Gastón Vaccarone

TUP 8 Turno Mañana

Sistema de Procesamiento de Datos

Primer año - Primer cuatrimestre

AÑO: 2023

## Introducción

En este informe, se presenta un proyecto basado en Arduino, un microcontrolador ampliamente utilizado en el ámbito de la electrónica y la programación. A lo largo de este informe, se detallarán los componentes, el diseño del circuito, la programación y los resultados obtenidos. Arduino ofrece una plataforma versátil para la creación de soluciones innovadoras en diversos campos, y este trabajo busca demostrar su aplicabilidad y potencial.

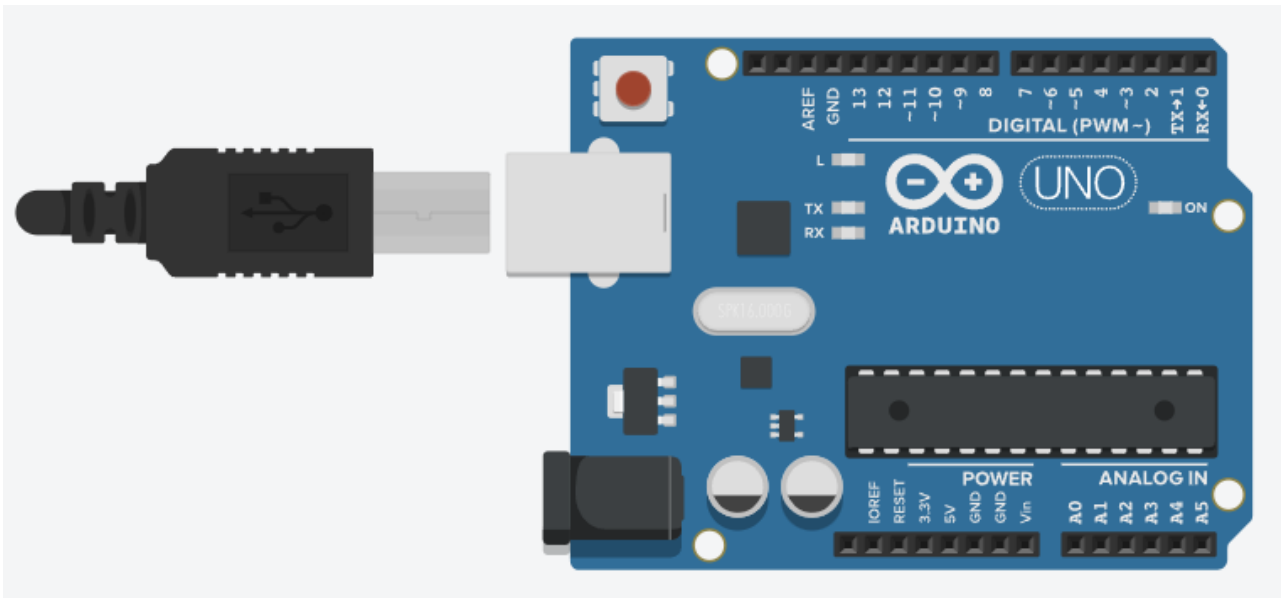
Al final del informe, comentaremos las dificultades del desarrollo y nuestra opinión sobre el mismo.

## Componentes usados

Arduino uno:

¿Qué es?

Arduino Uno es una placa de desarrollo de hardware de código abierto que se utiliza para crear proyectos electrónicos de manera sencilla. Viene con una variedad de pines de entrada/salida (E/S) que permiten conectar sensores, actuadores y otros dispositivos. Arduino Uno se programa utilizando un entorno de desarrollo basado en el lenguaje C/C++, lo que facilita la creación de código personalizado para controlar los componentes electrónicos conectados.



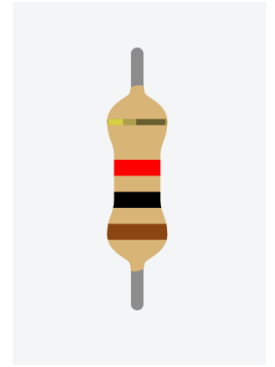
¿Para qué lo usamos?

Como medio de conexión entre todos los componentes.

## Resistencia:

¿Qué es?

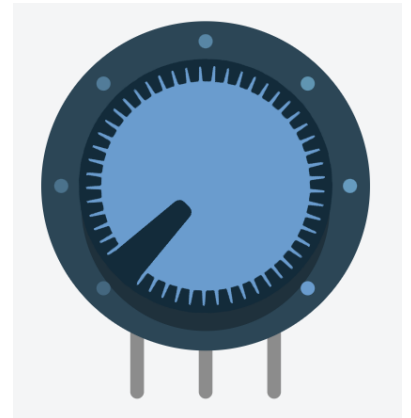
Una resistencia es un componente electrónico que limita el flujo de corriente eléctrica en un circuito. Se utiliza para controlar la intensidad de la corriente, proteger otros componentes y establecer niveles de voltaje deseados.



## Potenciómetro:

¿Qué es?

Un potenciómetro en Arduino es un componente ajustable que se utiliza para variar la resistencia eléctrica de un circuito. Consiste en un control giratorio o deslizante que permite modificar la cantidad de resistencia en función de su posición. Cuando se conecta a una placa Arduino permite regular el ajuste de un valor en una pantalla.



## Pantalla LCD:

¿Qué es?

Una pantalla LCD en Arduino es un dispositivo que se utiliza para mostrar información de manera visual en proyectos electrónicos. Estas pantallas, que pueden variar en tamaño y capacidad, se conectan a una placa Arduino y permiten mostrar texto, números y gráficos. La comunicación con la pantalla LCD se realiza a través de un conjunto de pines específicos, y se utiliza una biblioteca (`#include <LiquidCrystal.h>`) de software para controlar la visualización.



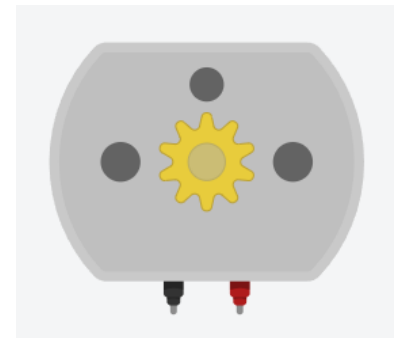
¿Para qué lo usamos?

Para mostrar los valores de la temperatura en grados centígrados y un mensaje dependiendo de los valores de humedad que se registran en el sensor.

## Motor de CC:

¿Qué es?

Un motor de corriente continua (CC) en Arduino es un dispositivo eléctrico que convierte la energía eléctrica en movimiento mecánico. Los motores de CC en Arduino son componentes fundamentales para proyectos que involucren movimiento.



¿Para qué lo usamos?

Simula el sistema de sistema de ventilación que se activa cuando la temperatura es superior a 30°C.

## Micro servo motor:

¿Qué es?

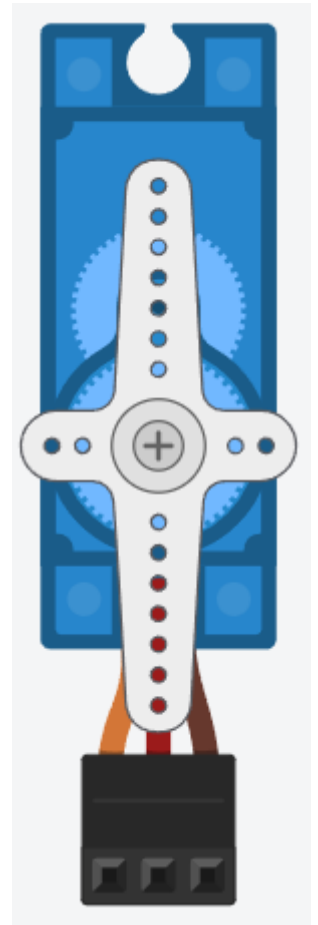
Un micro servo motor en Arduino es un pequeño dispositivo que se utiliza para controlar el movimiento en proyectos electrónicos. Está diseñado para realizar movimientos precisos y se controla mediante pulsos eléctricos generados por una placa Arduino. Los micros servos motores son ideales para aplicaciones donde se necesita un control de posición o movimiento.

Estos motores suelen tener un rango de movimiento de aproximadamente 180 grados y son conocidos por su facilidad de uso y precisión. Se conectan directamente a la placa Arduino y se controlan mediante la modulación de ancho de pulso (PWM) a través de un pin específico.



¿Para qué lo usamos?

Simula el sistema de riego que funciona cuando la humedad es menor al 80%.



## Led RGB:

¿Qué es?

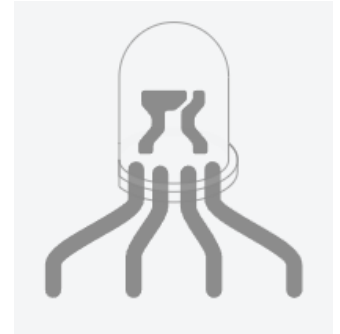
Un LED RGB (Light Emitting Diode) en Arduino es un dispositivo luminoso que puede emitir una variedad de colores, incluyendo rojo, verde y azul, al combinar diferentes intensidades de luz en esos colores primarios.

Con una placa Arduino, es posible controlar un LED RGB ajustando la intensidad de los canales rojo, verde y azul. Al modular la intensidad de cada canal mediante señales PWM



¿Para qué lo usamos?

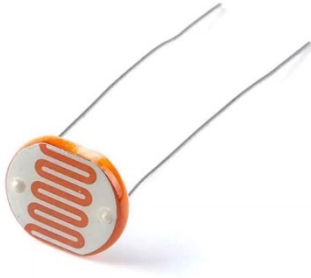
Se enciende cuando el jardín detecte menor intensidad de luz.



## Fotorresistencia:

¿Qué es?

Una fotorresistencia en Arduino es un componente sensor de luz que cambia su resistencia eléctrica en función de la cantidad de luz a la que está expuesta. Al conectar una fotorresistencia a una placa Arduino y configurar un circuito adecuado, es posible leer los cambios en la resistencia de la fotorresistencia a través de un pin analógico del Arduino. Con este dispositivo, se pueden crear proyectos que respondan a la intensidad lumínica,



¿Para qué lo usamos?

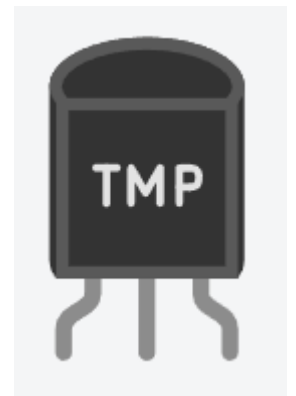
Detecta la cantidad de luz en el ambiente y trabaja en conjunto con el led RGB.

## Sensor de Temperatura (TMP):

¿Qué es?

Un sensor de temperatura TMP en Arduino es un dispositivo diseñado para medir la temperatura del entorno y proporcionar esa información a una placa Arduino. El sensor TMP36 convierte la temperatura en una señal eléctrica que se puede leer y procesar digitalmente a través de la placa Arduino.

Los sensores de temperatura TMP son populares en proyectos de monitoreo de temperatura, control de climatización.



¿Para qué lo usamos?

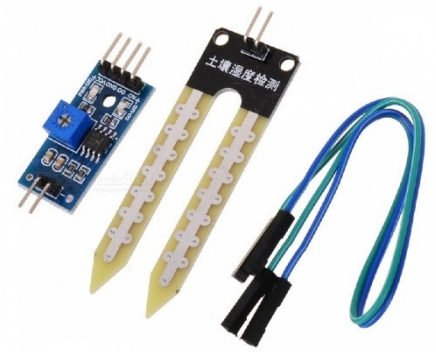
Detecta la temperatura del ambiente y trabaja en conjunto con el motor de CC (sistema de ventilación).

## Sensor de humedad:

¿Qué es?

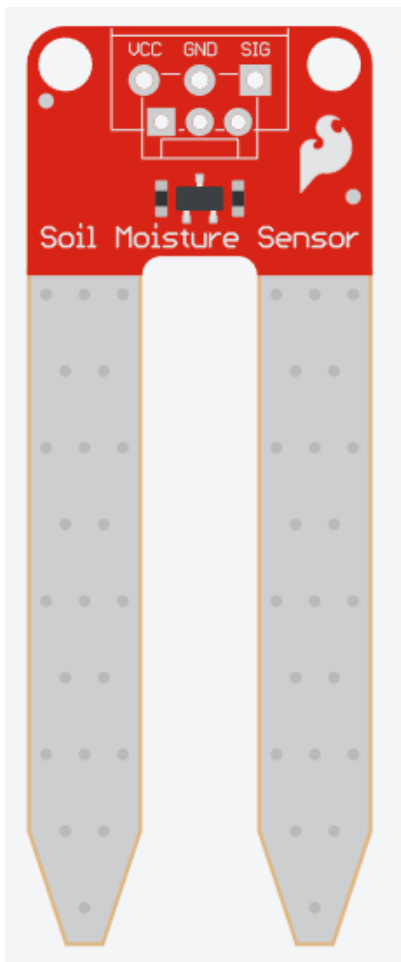
Un sensor de humedad en Arduino es un dispositivo diseñado para medir y registrar el nivel de humedad en el entorno. Al conectarse a una placa Arduino, el sensor de humedad proporciona información en forma de voltaje o datos digitales, que se pueden utilizar para monitorear la humedad en tiempo real.

Estos sensores de humedad son ampliamente utilizados en proyectos de control de riego, sistemas de climatización, monitorización de invernaderos y en muchas otras aplicaciones donde es importante conocer y controlar los niveles de humedad.

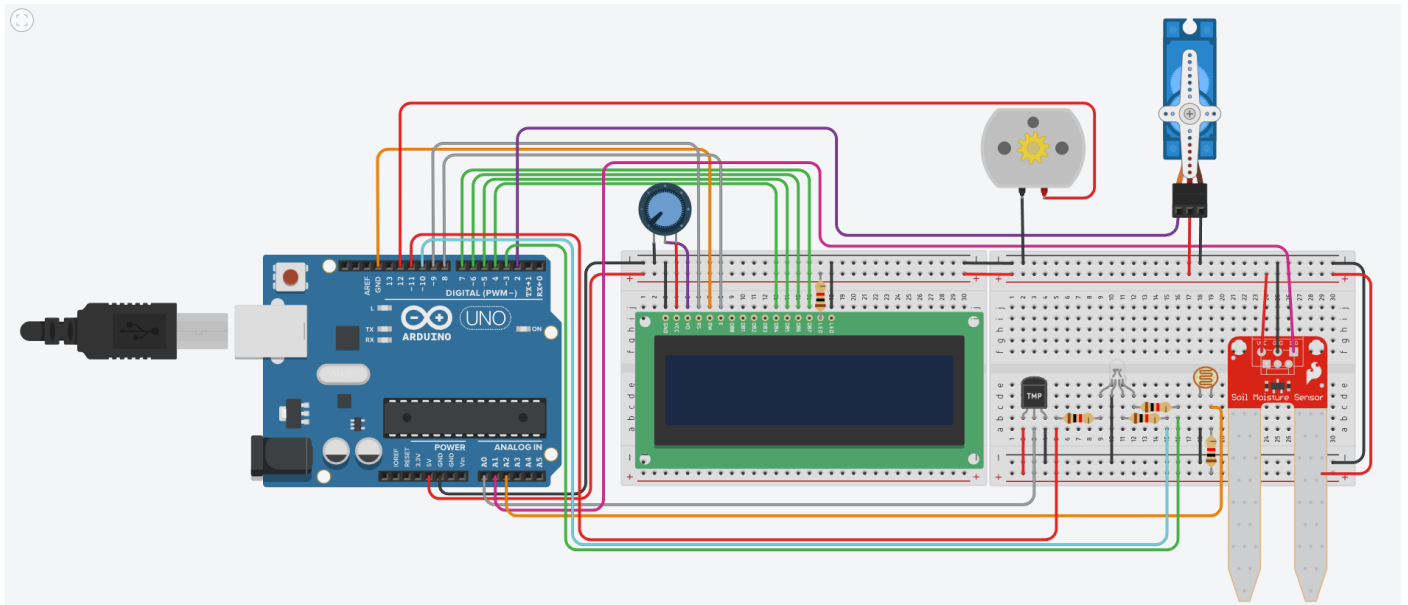


¿Para qué lo usamos?

Detecta la humedad del ambiente y trabaja en conjunto con el micro servo motor (sistema de riego).



## Diseño del circuito



## Dificultades del desarrollo:

- Mantener un orden de los componentes.
- Coincidir los tiempos entre los integrantes.

## Conclusión

En conclusión, este proyecto de sistema de riego en Arduino ha sido una valiosa oportunidad de aprendizaje. Hemos aplicado nuestros conocimientos en electrónica y programación para crear un sistema eficiente y hemos comprendido la relevancia de la tecnología para facilitar la realización de diferentes tareas.



## CODIGO

[https://www.tinkercad.com/things/d0pt7ar4MIE?sharecode=WGGYHiJwv\\_Ss9Ksu-IWfzpGBNcJZSGZ\\_eJbAKX9GaSg](https://www.tinkercad.com/things/d0pt7ar4MIE?sharecode=WGGYHiJwv_Ss9Ksu-IWfzpGBNcJZSGZ_eJbAKX9GaSg)

```
#include <LiquidCrystal.h>
#include <Servo.h>
// LiquidCrystal LCD(rs, enable, d4, d5, d6, d7);

LiquidCrystal LCD(9, 8, 4, 5, 6, 7);
Servo Servomotor;

float temp1 = 0.00;
float temp2 = 0.00;
float temp3 = 0.00;
float temp4 = 0.00;
float valor_leido = 0.00;
float voltaje = 0.00;

int angulo = 0;
int ventilacion = 12;

int rojo = 11;
int azul = 10;
int verde = 3;

void setup()
{
  Serial.begin(9600);
  LCD.begin(16, 2);
  LCD.setCursor(0, 0);
  pinMode(3, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(11, OUTPUT);
  pinMode(ventilacion, OUTPUT);
  Servomotor.attach(2);
}

void loop()
{
  // TMP
  int valor_leido = analogRead(A0);

  voltaje = map(valor_leido, 20, 358, 0, 5);

  temp1 = 5 * valor_leido;
  temp2 = temp1 * 100;
  temp3 = temp2 / 1024;
  temp4 = temp3 - 50;

  LCD.setCursor(0, 0);
  LCD.print("TMP: ");
  LCD.print(temp4);
  LCD.print(" C");

  // SENSOR DE HUMEDAD
  int sensor_hum = analogRead(A1);
  int humedad = map(sensor_hum, 0, 876, 0, 100);
```

```

LCD.setCursor(0, 1);
LCD.print("Hum: ");
if (humedad <= 20)
{
    LCD.print("Muy Seco");
}
else if (humedad > 20 && humedad <= 70)
{
    LCD.print("Seco");
}
else if (humedad > 70)
{
    LCD.print("Humedo");
}

```

## // FOTORRESISTENCIA

```

float luminosidad = analogRead(A2);
float luz = map(luminosidad, 1017, 344, 0, 100);
if (luz >= 78)
{
    digitalWrite(rojo, 254);
    digitalWrite(azul, 0);
    digitalWrite(verde, 254);
}
else if (luz < 78)
{
    digitalWrite(rojo, 0);
    digitalWrite(azul, 254);
    digitalWrite(verde, 0);
}
Serial.println("LUZ: ");
Serial.println(luz);

```

## // SISTEMA DE RIEGO

```

if (temp4 > 30)
{
    digitalWrite(ventilacion, HIGH);
}
else
{
    digitalWrite(ventilacion, LOW);
}

if (humedad < 69)
{
    if (angulo == 0)
    {
        delay(400);
        Servomotor.write(180);
        delay(600);
        angulo = 180;
    }
    else if (angulo == 180)
    {
        delay(400);
        Servomotor.write(0);
        delay(600);
        angulo = 0;
    }
}

```

```
}  
  delay(1000);  
  LCD.clear();  
}
```

## **INDICE**

Introducción.....	2
Componentes usados.....	2
Arduino uno:.....	2
Resistencia:.....	3
Potenciómetro:.....	3
Pantalla LCD:.....	4
Motor de CC:.....	4
Micro servo motor:.....	5
Led RGB:.....	5
Fotorresistencia:.....	6
Sensor de Temperatura (TMP):.....	6
Sensor de humedad:.....	7
Diseño del circuito.....	8
Dificultades del desarrollo:.....	8
Conclusión.....	8
CODIGO.....	9