**BBDD:**

CREATE DATABASE IF NOT EXISTS sistema\_riego;

USE sistema\_riego;

-- Tabla para almacenar los datos de los sensores

CREATE TABLE IF NOT EXISTS datos\_sensores (

id INT AUTO\_INCREMENT PRIMARY KEY,

humedad FLOAT,

temperatura FLOAT,

humedad\_suelo INT,

fecha\_hora TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

**ARDUINO:**

#include <ESP8266WiFi.h>

#include <WebSocketsClient.h>

#include <DHT.h>

#include <MySQL\_Connection.h>

#include <MySQL\_Cursor.h>

#define DHTPIN 2 // Pin digital al que está conectado el sensor DHT

#define DHTTYPE DHT11 // Tipo de sensor DHT que estás utilizando

#define SOIL\_MOISTURE\_PIN A0 // Pin analógico al que está conectado el sensor de humedad del suelo

#define PUMP\_PIN 3 // Pin digital al que está conectada la bomba de agua

const char\* ssid = "TU\_SSID";

const char\* password = "TU\_CLAVE\_WIFI";

const char\* webSocketServerAddress = "IP\_DEL\_SERVIDOR\_WS";

const char\* dbServer = "IP\_DEL\_SERVIDOR\_MYSQL";

const char\* dbUser = "USUARIO\_MYSQL";

const char\* dbPassword = "CONTRASEÑA\_MYSQL";

const char\* dbName = "sistema\_riego"; // Nombre de la base de datos

WebSocketsClient webSocket;

DHT dht(DHTPIN, DHTTYPE);

WiFiClient client;

MySQL\_Connection conn((Client \*)&client);

void setup() {

Serial.begin(115200);

pinMode(PUMP\_PIN, OUTPUT);

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED) {

delay(1000);

Serial.println("Conectando a la red WiFi...");

}

Serial.println("Conexión WiFi establecida.");

dht.begin();

if (conn.connect(dbServer, 3306, dbUser, dbPassword, dbName)) {

Serial.println("Conexión a la base de datos exitosa.");

} else {

Serial.println("Error al conectar a la base de datos.");

}

webSocket.begin(webSocketServerAddress);

webSocket.onEvent(webSocketEvent);

}

void loop() {

webSocket.loop();

// Leer la humedad y temperatura

float humedad = dht.readHumidity();

float temperatura = dht.readTemperature();

// Leer la humedad del suelo

int humedadSuelo = analogRead(SOIL\_MOISTURE\_PIN);

// Guardar los datos en la base de datos

if (conn.connected()) {

char query[128];

sprintf(query, "INSERT INTO datos\_sensores (humedad, temperatura, humedad\_suelo) VALUES (%f, %f, %d)", humedad, temperatura, humedadSuelo);

MySQL\_Cursor \*cursor = new MySQL\_Cursor(&conn);

cursor->execute(query);

delete cursor;

}

// Enviar los datos al servidor WebSocket

if (webSocket.isConnected()) {

String data = "{\"humedad\": " + String(humedad) + ", \"temperatura\": " + String(temperatura) + ", \"humedadSuelo\": " + String(humedadSuelo) + "}";

webSocket.sendTXT(data);

}

delay(5000); // Esperar 5 segundos antes de enviar nuevos datos

}

void webSocketEvent(WStype\_t type, uint8\_t \* payload, size\_t length) {

// Manejar eventos WebSocket si es necesario

}

// Función para activar la bomba de agua

void activarRiego() {

digitalWrite(PUMP\_PIN, HIGH); // Encender la bomba de agua

delay(5000); // Mantener la bomba activada durante 5 segundos

digitalWrite(PUMP\_PIN, LOW); // Apagar la bomba de agua

}