[[1]](#footnote-1)

Intelligent Plant Irrigation System **Abstract** — This document presents an innovative project that seeks to improve the care and health of plants through a smart device that monitors environmental conditions and automatically activates an irrigation system. The project combines sensor technology with an automatic irrigation system to optimize plant growth and development..



**Key words:**

Planta Irrigation System

ESP32 Microcontroller

Artificial Intelligence (AI)

Sensors

**Device components**

The smart plant care device is made up of several essential components that work together to monitor environmental conditions and activate the automatic watering system. These components include:

**Soil Moisture Sensors**: These sensors measure soil moisture levels around plants and provide accurate information about the amount of water available to the roots.

**Temperature sensors:** Temperature sensors monitor the temperature of the environment and soil, allowing the device to adjust the frequency and duration of watering according to the needs of the plants.

**Light Sensor:** The light sensor detects the intensity of sunlight that plants receive, which is crucial for the photosynthesis process and healthy growth.

Control unit: The control unit is the brain of the device, it processes information from the sensors and makes decisions about activating the irrigation system.

Actuator: The actuator is a device that controls the flow of water to the plants. It can be an electromagnetic valve that opens and closes according to the instructions of the control unit.

Irrigation System: The irrigation system can be a drip system, sprinkler system or any other method appropriate to the type of plants..

**Automatic irrigation system**

The device's automatic watering system works intelligently and efficiently, adjusting to the needs of the plants.

The control unit receives data from humidity, temperature and light sensors and decides when and fur how long to activate irrigation.

Upon detecting that soil moisture levels fall below a preset threshold, the control unit activates the actuator, opening the valve and allowing water to flow to the plants.

The irrigation duration is set to depending on temperature and light intensity, ensuring optimal water supply for plants.

The system also has the ability to program regular watering schedules, allowing you to maintain a humidity Constant soil humidity, even in periods of drought or when plants need specific watering..

Condition Monitoring

The device's sensors continuously measure humidity, temperature and light in the plants' environment.

Data analysis

The control unit analyzes the data from the sensors and compares them with preset parameters.

Irrigation activation

If the soil moisture drops below the threshold, the control unit activates the irrigation system.

Irrigation adjustment

The duration and frequency of watering are adjusted depending on the temperature and light intensity.

**connections**

Soil Moisture Sensors (SHS):

These sensors measure soil moisture and help determine when watering is necessary. Connection: Connect the SHS sensor to the microcontroller (such as Arduino) using the appropriate pins (example, A0). Calibration: Adjust the sensor according to the characteristics of your soil and plant

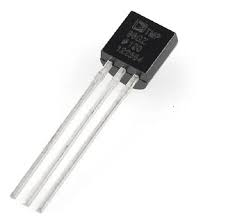
**Solenoid valve:**

Provides water to plants. It connects to the irrigation system and the microcontroller.



**Temperature sensor:**

Connect the temperature sensor to the microcontroller (such as Arduino) using the appropriate pins. If a DHT11 or DHT22 is used, follow the specific connections for each model. If an AM2301A is used, connect it via I2C or following its pin.



**Humidity sensor**

The humidity sensor is a device used in indoor spaces to monitor air humidity and sometimes also ambient temperature.

The values measured by the sensor are transformed into an electrical signal (with an intensity between 4 and 20 mA).



**manufacturing process**

**Required components**:

Arduino board: An Arduino MKR1000 IoT board or any other Arduino board with Wi-Fi connectivity can be used.

Sensors: Soil moisture sensor: Measures soil moisture. Temperature and humidity sensor: Provides data on ambient temperature and humidity. Light sensor: Detects the intensity of light.

Irrigation pump or motor: Provides water to plants.

**Connections**

Connections: The sensors are connected to the Arduino board according to the specifications of each sensor.

Use a relay or transistor to control the irrigation pump.

Programming: Read the sensor values (temperature, soil humidity and light).

Thresholds are defined to activate irrigation (for example, if soil moisture is low or light is insufficient).

If the sensor values are below the thresholds, activate the irrigation pump.

**Recommendations**

Keep in mind that the sensors must be calibrated correctly and the thresholds adjusted according to the needs of each situation.

This system can be expanded by adding more sensors or integrating it with a cloud platform to monitor and control irrigation from anywhere, aI can also be added to improve the efficiency of the system.

**Distribution**

Global Connectivity: An Arduino board with Wi-Fi or GSM connectivity (such as the Arduino MKR GSM 1400) can be used to send data over the Internet.

The board is configured to connect to a Wi-Fi network or a SIM card with Internet access.

Cloud platform: A cloud platform (such as AWS, GoogleCloud, or Microsoft Azure) is used to receive and process sensor data.

Data is sent from theArduino board to the cloud using MQTT, HTTP or some other protocol.

Servers and databases: Cloud servers are configured to receive and store the data.

A database (such as MySQL, PostgreSQL or NoSQL) is used to store the sensor information.

APIs and applications: An API can be created so that the Arduino board can send data to the cloud.

A web or mobile application is developed so that users can monitor and control the irrigation system from anywhere.

Security: Implement security measures, such as authentication and encryption, to protect data and communication between the board and the cloud.

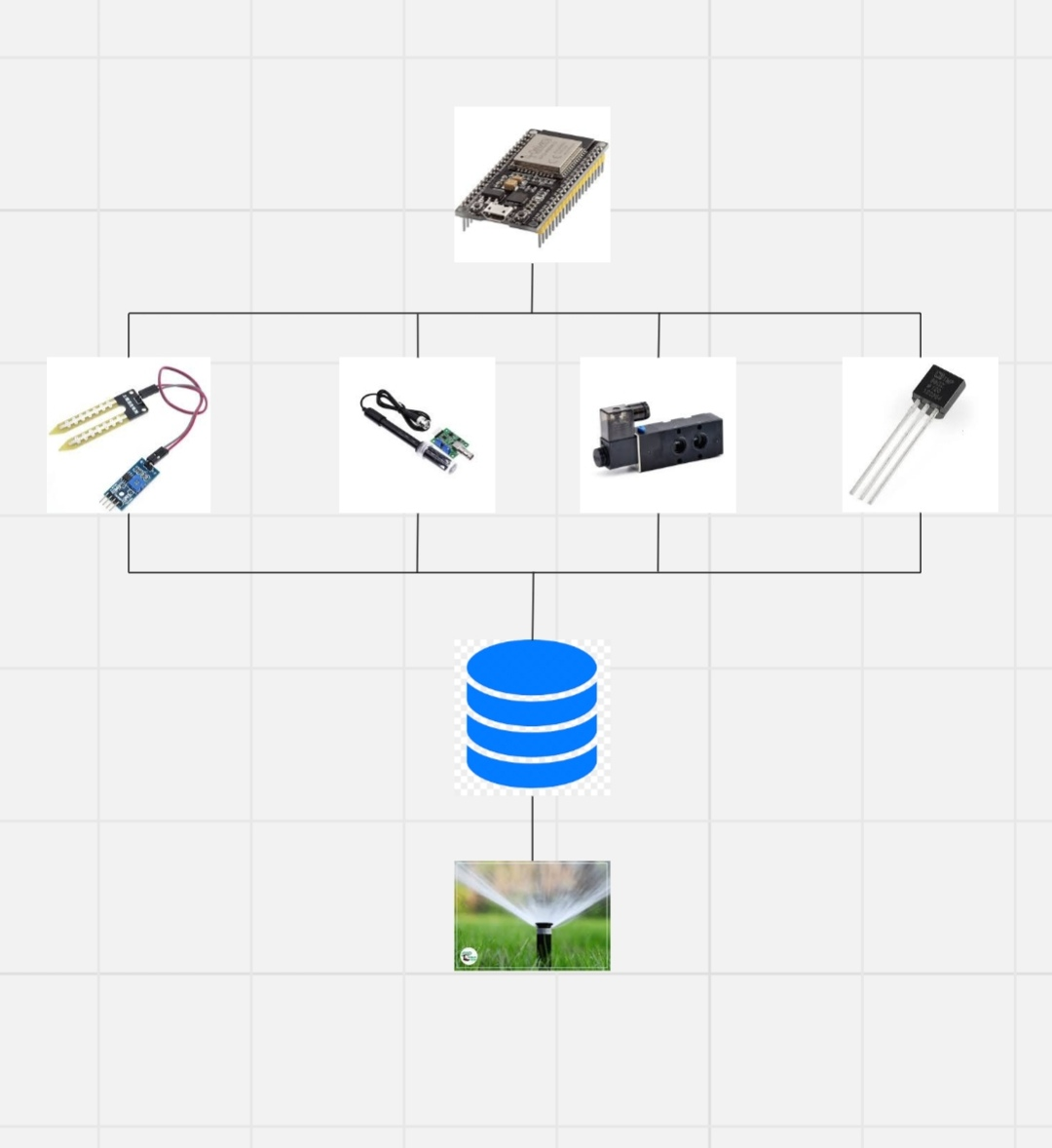
Architecture example: The Arduino board sends data via MQTT to a cloud server.

The server processes the data and stores the information in a database.

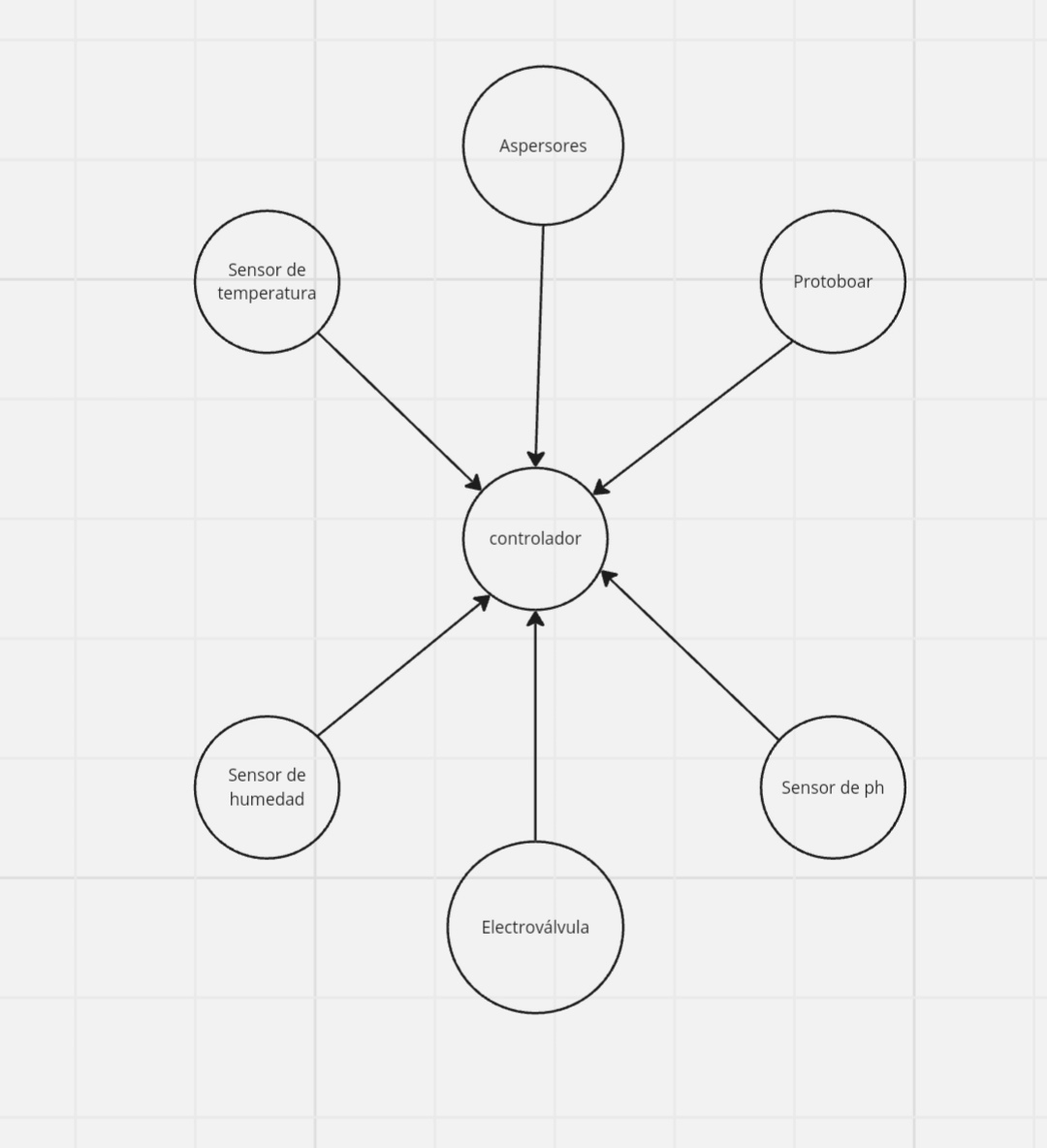
The web or mobile application queries the database and displays the sensor values to the user.

If the thresholds are crossed, the app sends a signal to the irrigation system to activate it.

**network diagram**

****

**Node diagram**

nente

**References**

Duque, F. S., & Saldarriaga, L. H. (2020). Sistema de riego automatizado para una huerta escolar, basado en una red inalámbrica de sensores. Informe de práctica social. Universidad de Antioquia.

Proyectos con Arduino. (2019). ArduRiego. Riego automático inteligente

Fecha de la conferencia: 21-25 de junio de 2004. Proceedings of the International Conference on Information Acquisition, 2004. ISBN impreso: 0-7803-8629-9.

1. [↑](#footnote-ref-1)