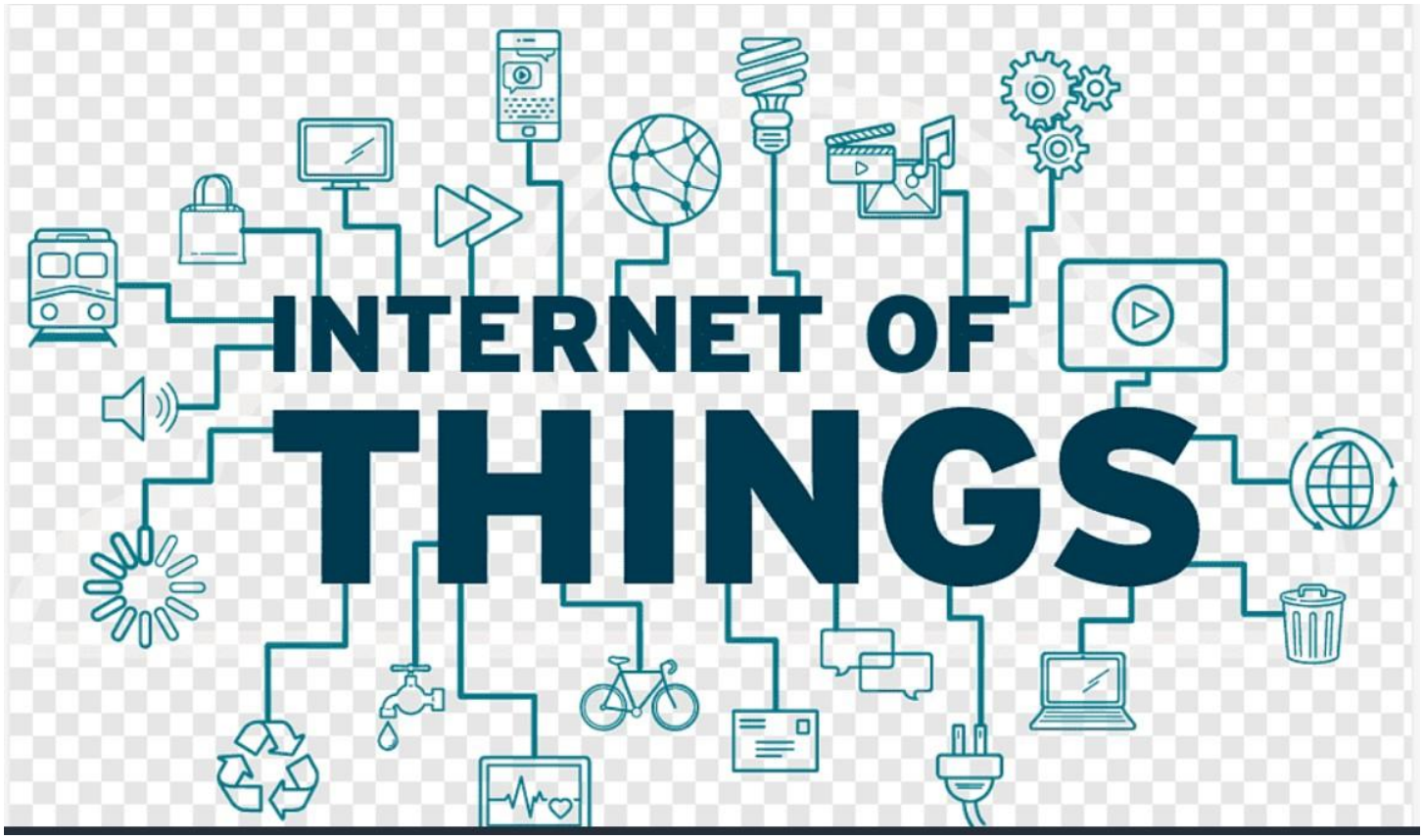


EKREM ALTUNHAN

20200808079

Smart Weather and Home Cover Control System

1.Student Information
Full Name: Ekrem ALTUNHAN
Student ID: 20200808079

2.Project Title:

Smart Weather and Home Cover Control System

3. Introduction

In today's world, IoT systems are widely used in home automation, agriculture, and environmental monitoring. This project was developed to demonstrate how ESP32 can be used to build an efficient and smart environmental monitoring and actuator control system integrated with a cloud platform.

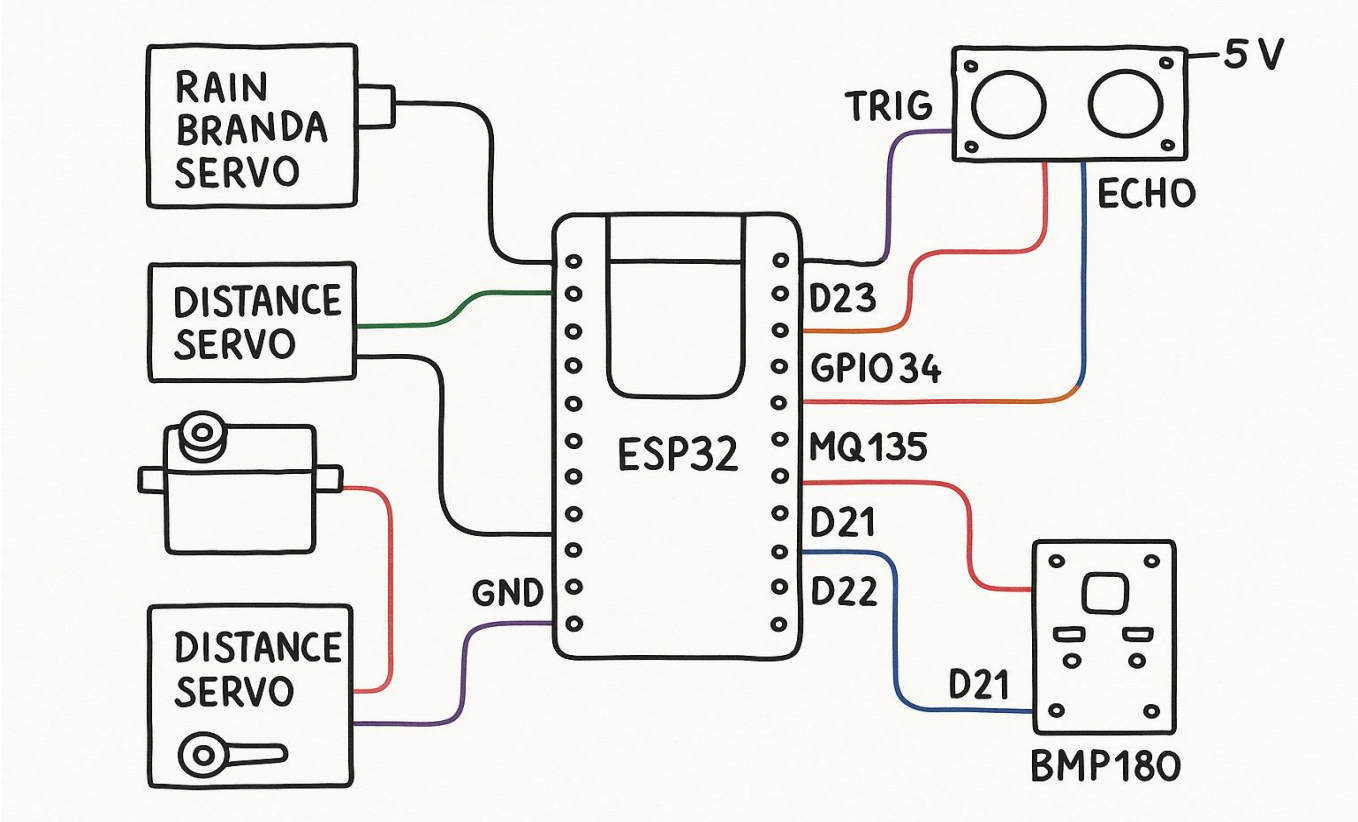
The project combines environmental sensing and servo control with Arduino IoT Cloud integration, allowing both autonomous and manual operation of actuators based on real-time sensor data.

Problem Definition: To develop a system that can autonomously detect environmental conditions (e.g., rain, proximity, gas levels, temperature, pressure) and take necessary action while allowing cloud-based monitoring and control.

---

4. System Architecture

4.1 System Diagram



4.2 Components Used

- **ESP32-WROOM-32**: Main controller with built-in WiFi
- **Rain Sensor (MH-RD)**: Detects precipitation (digital output)
- **Ultrasonic Sensor (HC-SR04)**: Measures proximity
- **Gas Sensor (MQ-135)**: Monitors air quality (analog output)
- **BMP180**: Measures ambient temperature and pressure via I2C
- **2x SG90 Servo Motors**: Performs mechanical movement (covers)
- **5V External Power Supply**: Provides stable current to servos

5. Implementation Details

Hardware Setup

- Servo motors connected to GPIO 13 and 12
- Rain Sensor DO → GPIO 23
- HC-SR04 Trig/Echo → GPIO 5/18
- MQ-135 AO → GPIO 34
- BMP180 SDA/SCL → GPIO 21/22
- All sensors and ESP32 share common GND with 5V power rail

Sensor and Actuator Integration

- Servo 1 responds to rain sensor; closes a virtual cover when rain is detected
- Servo 2 responds to proximity; opens/closes cover based on object distance

Communication Protocols Used

MQTT and **HTTPS** protocols are used via the ArduinoIoTCloud library

Devices communicate securely with the Arduino Cloud using SSL/TLS encryption

Cloud Services Used

Arduino IoT Cloud dashboard:

Real-time data display

Manual override via switch widgets

Data logging and trend analysis

Data Analytics Methods Applied

Visual dashboards plot data from sensors including gas level, temperature, pressure, and distance

Charts available in line/point format, auto-scaled based on values

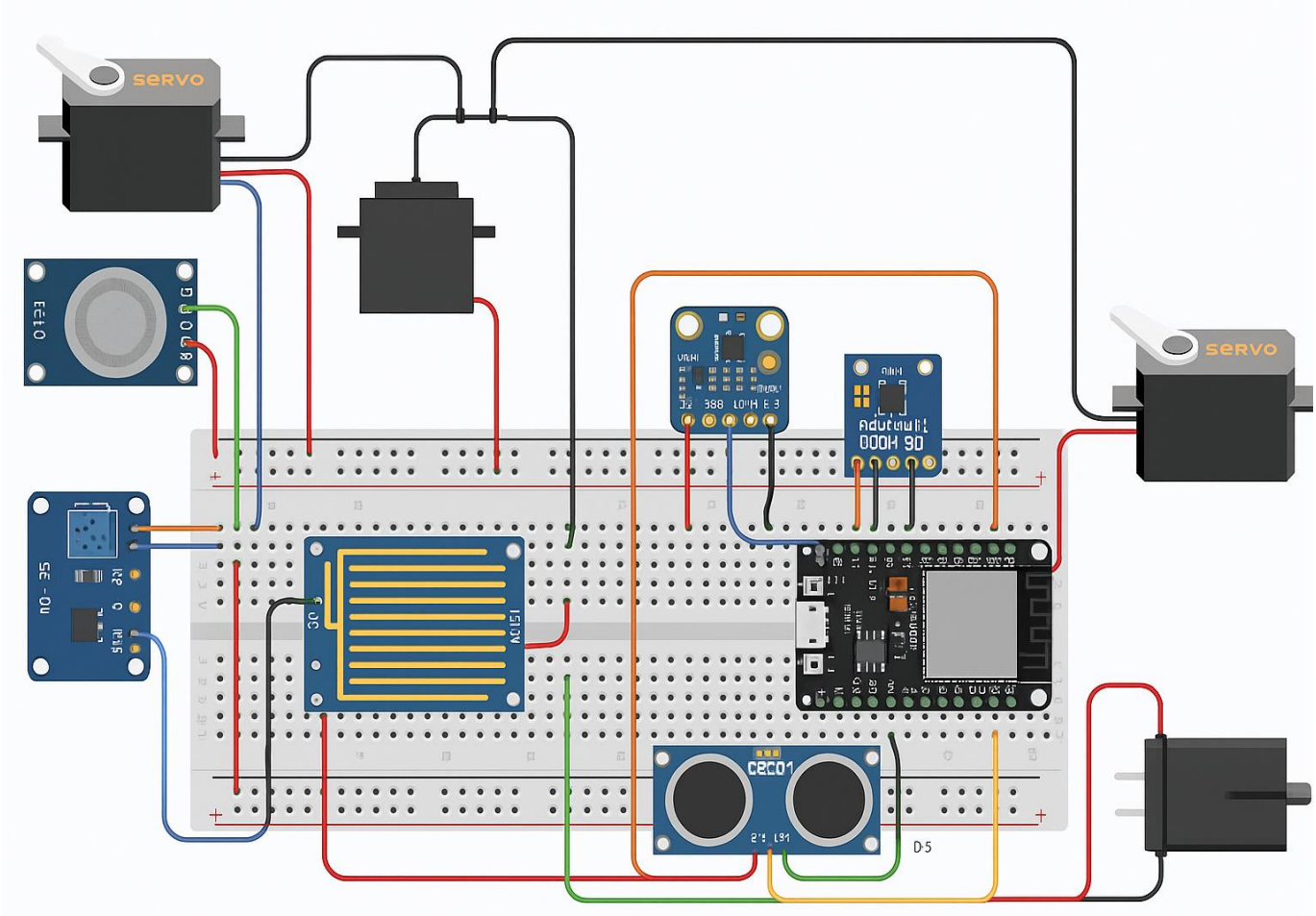
Dashboard widgets allow observation of trends over selected time ranges (1 hour, 1 day, 1 week)

Security Features Implemented

Wi-Fi credentials and device keys are stored in arduino_secrets.h (not exposed in code)

Device authentication is handled via Arduino's secure token system

Data transmission is encrypted using HTTPS/MQTT over TLS for both sending and receiving



6. Results and Analysis

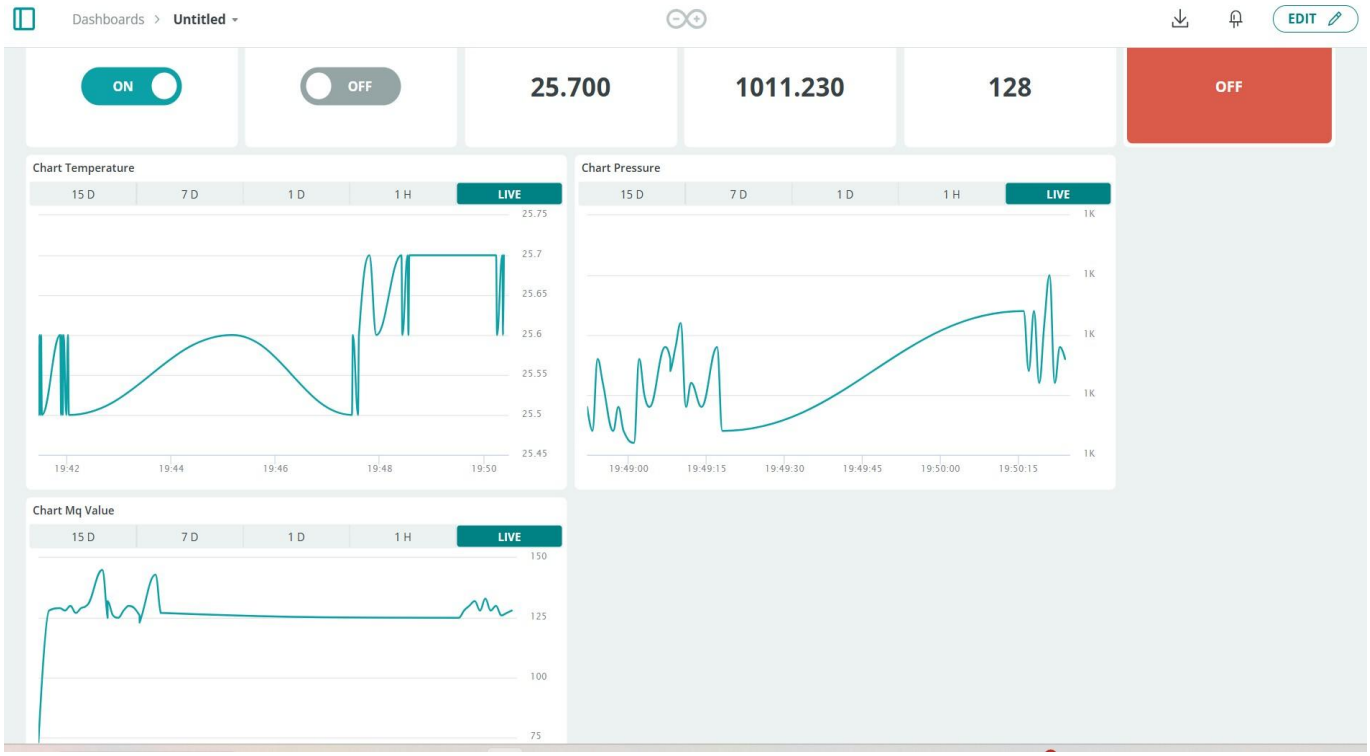
Outputs

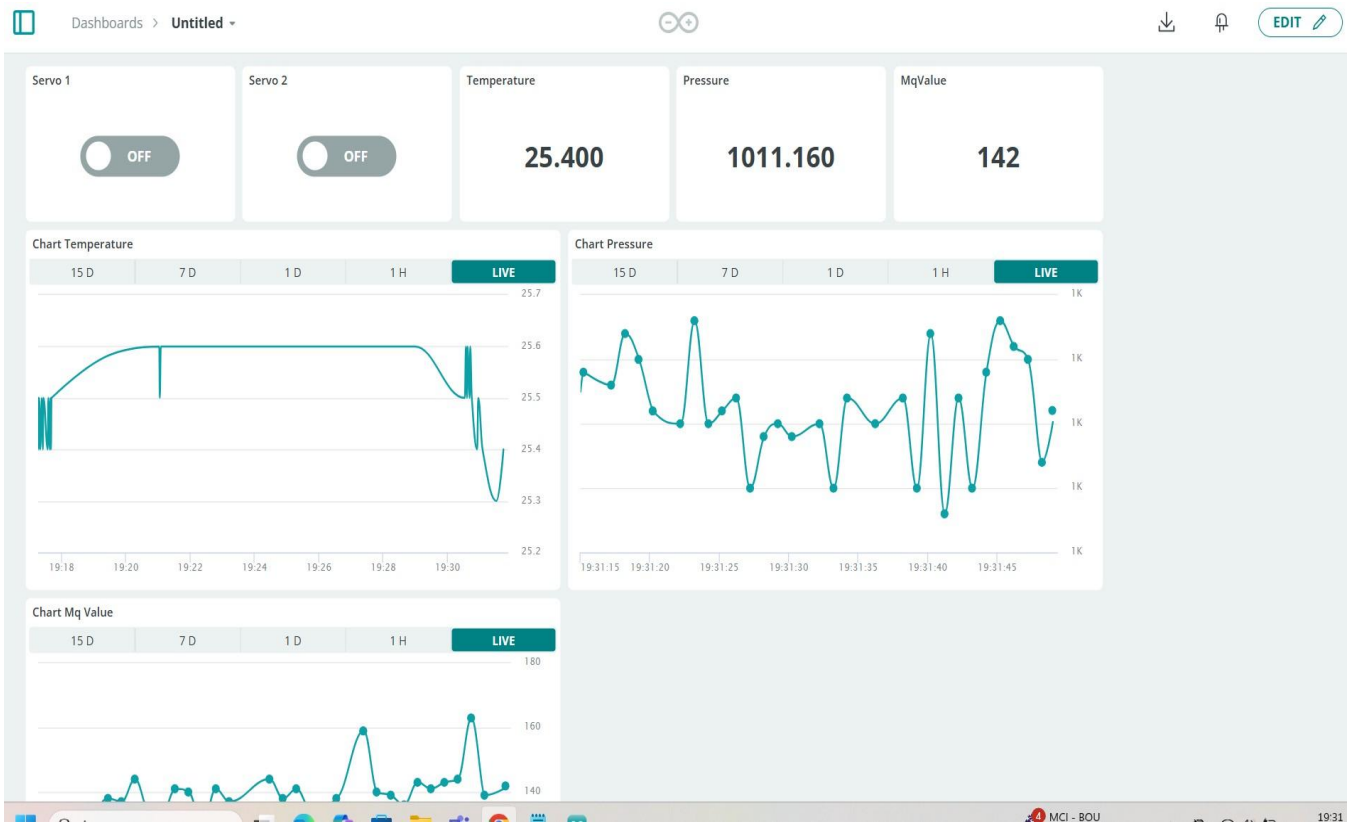
Servo 1: Automatically triggered by rain detection

Servo 2: Triggered by proximity < 10cm

Dashboard shows real-time updates for all environmental metrics

Screenshots and Plots





Observations

Servo operation was smooth and responsive

Manual override switches worked even when automatic mode was active

Dashboard accurately reflected sensor and actuator states

MQ-135 and ultrasonic sensors showed consistent readings under varying conditions

Challenges Faced

Servo jitter without separate power was fixed by using external 5V supply

BMP180 required stable I2C wiring; loose wires caused startup issues

7. Conclusion and Future Work

This project successfully implemented a smart, cloud-connected IoT system with sensor-based automation and remote manual control. It showcased integration of sensors, actuators, ESP32 programming, and cloud dashboard features.

Future Enhancements:

- Add DHT11 humidity sensor
- Add LED/buzzer for gas warnings
- Implement mobile app notifications
- Expand system to multi-zone sensing
- Use secure MQTT broker (e.g., AWS IoT Core)

8. References

- Arduino IoT Cloud: https://create.arduino.cc/iot
- Adafruit BMP180 Library
- MQ-135 Gas Sensor Datasheet
- HC-SR04 Ultrasonic Datasheet
- https://randomnerdtutorials.com/esp32-iot-cloud

9. GitHub Repository

All code and documentation:
🔗 https://github.com/EkremAltunhan/iot-20200808079.git