ENVIRONMENTAL MONITORING IN PARK

Creating an environmental monitoring program for a park using IoT (Internet of Things) involves several components, including sensors, a microcontroller, and cloud connectivity. Below is a simplified example using an Arduino board and the Adafruit IO platform for cloud connectivity. This code monitors temperature and humidity and sends the data to the cloud for visualization.

cpp

#include <Wire.h>

#include <Adafruit\_Sensor.h>

#include <Adafruit\_BME280.h>

#include <WiFiNINA.h>

#include <AdafruitIO\_WiFi.h>

#define WIFI\_SSID "your\_wifi\_ssid"

#define WIFI\_PASS "your\_wifi\_password"

#define IO\_USERNAME "your\_io\_username"

#define IO\_KEY "your\_io\_key"

#define BME\_SDA 2

#define BME\_SCL 3

AdafruitIO\_WiFi io(IO\_USERNAME, IO\_KEY, WIFI\_SSID, WIFI\_PASS);

Adafruit\_BME280 bme;

void setup() {

Serial.begin(115200);

// Initialize the BME280 sensor

if (!bme.begin(0x76)) {

Serial.println("Could not find a valid BME280 sensor, check wiring!");

while (1);

}

// Connect to Wi-Fi

connectToWiFi();

// Connect to Adafruit IO

io.connect();

// Set up a feed for temperature and humidity

AdafruitIO\_Feed \*temperature = io.feed("temperature");

AdafruitIO\_Feed \*humidity = io.feed("humidity");

// Wait for a connection

while (io.status() < AIO\_CONNECTED) {

delay(500);

}

// Print connection status

Serial.println("Connected to Adafruit IO!");

}

void loop() {

// Read temperature and humidity from BME280 sensor

float temperatureValue = bme.readTemperature();

float humidityValue = bme.readHumidity();

// Send data to Adafruit IO

io.feed("temperature")->save(temperatureValue);

io.feed("humidity")->save(humidityValue);

// Print data to serial monitor

Serial.print("Temperature: ");

Serial.println(temperatureValue);

Serial.print("Humidity: ");

Serial.println(humidityValue);

delay(60000); // Delay for 1 minute before sending data again

}

void connectToWiFi() {

int status = WL\_IDLE\_STATUS;

// Attempt to connect to Wi-Fi

while (status != WL\_CONNECTED) {

Serial.print("Attempting to connect to SSID: ");

Serial.println(WIFI\_SSID);

status = WiFi.begin(WIFI\_SSID, WIFI\_PASS);

delay(10000); // Delay for 10 seconds before retrying

}

Serial.println("Connected to Wi-Fi");

}

Make sure to replace `"your\_wifi\_ssid"`, `"your\_wifi\_password"`, `"your\_io\_username"`, and `"your\_io\_key"` with your actual Wi-Fi credentials and Adafruit IO credentials.

This code uses the Adafruit BME280 sensor to measure temperature and humidity and sends the data to Adafruit IO for monitoring. You can create a dashboard on Adafruit IO to visualize and analyze the environmental data collected from the park. Additionally, you can expand the code to include more sensors and data points as needed.