ENVIRONMENTAL MONITORING IN PARKS

Creating an IoT program for environmental monitoring in parks with Arduino involves interfacing multiple sensors, data processing, and potentially transmitting data to a server or cloud platform. Here's an example program that collects data from various sensors and sends it to a server using an ESP8266 module for Wi-Fi connectivity:

Components Required:

Arduino board (e.g., Arduino Uno)
Various sensors (e.g., DHT22 for temperature and humidity, LDR for light intensity)
ESP8266 Wi-Fi module (e.g., ESP-01)
Jumper wires
Breadboard
USB cable for Arduino

Arduino Setup:

1. Connect the sensors to the Arduino as in previous examples:
    DHT22 sensor for temperature and humidity.
    LDR sensor for light intensity.
    Other sensors for additional environmental data (e.g., air quality sensor, soil moisture sensor).
2. Connect the ESP8266 module to the Arduino. Ensure the module receives power (3.3V), connects to the Arduino's RX and TX pins, and has a common ground connection.
3. Install necessary libraries in the Arduino IDE, such as DHT, for interfacing with the sensors and ESP8266WiFi for Wi-Fi communication.

Arduino Program:

Explanation: **#include <DHT.h>**

```
#include <ESP8266WiFi.h>



#define DHTPIN 2

#define DHTTYPE DHT22



const char* ssid = "YourWiFiSSID";

const char* password = "YourWiFiPassword";

const char* serverAddress = "your-server.com";  // Replace with your server address



DHT dht(DHTPIN, DHTTYPE);



void setup() {

  Serial.begin(115200);

  dht.begin();



  // Connect to Wi-Fi

  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {

    delay(1000);
```

```
    Serial.println("Connecting to WiFi...");
  }
  Serial.println("Connected to WiFi");
}

void loop() {
  float temperature = dht.readTemperature();  // Read temperature in Celsius
  float humidity = dht.readHumidity();  // Read humidity in percentage
  int lightIntensity = analogRead(A0);  // Read light intensity from LDR

  // Send data to the server
  sendDataToServer(temperature, humidity, lightIntensity);

  delay(60000);  // Delay for a minute before taking the next reading
}

void sendDataToServer(float temp, float humid, int light) {
  WiFiClient client;
```

```
  if (client.connect(serverAddress, 80)) {

    String data = "temperature=" + String(temp) + "&humidity=" + String(humid) + "&light=" +
String(light);

    client.println("POST /your-api-endpoint HTTP/1.1");

    client.println("Host: " + String(serverAddress));

    client.println("Content-Type: application/x-www-form-urlencoded");

    client.println("Content-Length: " + String(data.length()));

    client.println();

    client.print(data);

  }


  client.stop();

}
```

In this program, the Arduino collects data from the DHT22 and LDR sensors. The ESP8266 module is used for Wi-Fi connectivity to send data to a server. The `sendDataToServer` function sends the collected data to your server. You need to replace `"your-server.com"` with the actual server address and specify the correct API endpoint.

Server-Side Setup:

On the server side, you'll need to create an API endpoint to receive data sent by the Arduino and then store or process that data as needed.

This example program provides a foundation for monitoring temperature, humidity, and light intensity in parks. You can expand it to include more sensors and adapt the server-side setup to meet your specific requirements for environmental monitoring in parks.