**ENVIRONMENTAL MONITORING USING IOT**

**PHASE 5**

**AIM**

To create a platform that displays real-time environmental data using web develop technologies.

**Project Objectives:**

Many regions around the world are facing increasing environmental challenges, such as air pollution, water contamination, deforestation, and climate change. To effectively address these issues and mitigate their impacts, there is a need for real-time, comprehensive environmental monitoring systems. Traditional methods of data collection are often manual, sporadic, and costly, making it difficult to obtain timely and accurate environmental data. Therefore, the problem statement is:

"How can we develop an efficient and cost-effective environmental monitoring system using IoT (Internet of Things) technology to continuously collect, analyse, and report environmental data for better decision-making and sustainable environmental management?"

**Solution Design:**

**1. Sensor Deployment:**

* Select and deploy a variety of sensors (e.g., temperature, humidity ) in strategic locations across the target area.
* Ensure sensors are rugged, reliable, and capable of continuous data collection.
* Sensors should be connected to IoT devices with wireless communication capabilities (e.g., Wi-Fi or cellular networks).

**2. Data Collection and Transmission:**

* Establish a robust data collection infrastructure to gather data from sensors in real-time.
* Use IoT gateways or edge computing devices to preprocess and transmit data securely to a central server or cloud platform.

**3. User Interface and Visualization**:

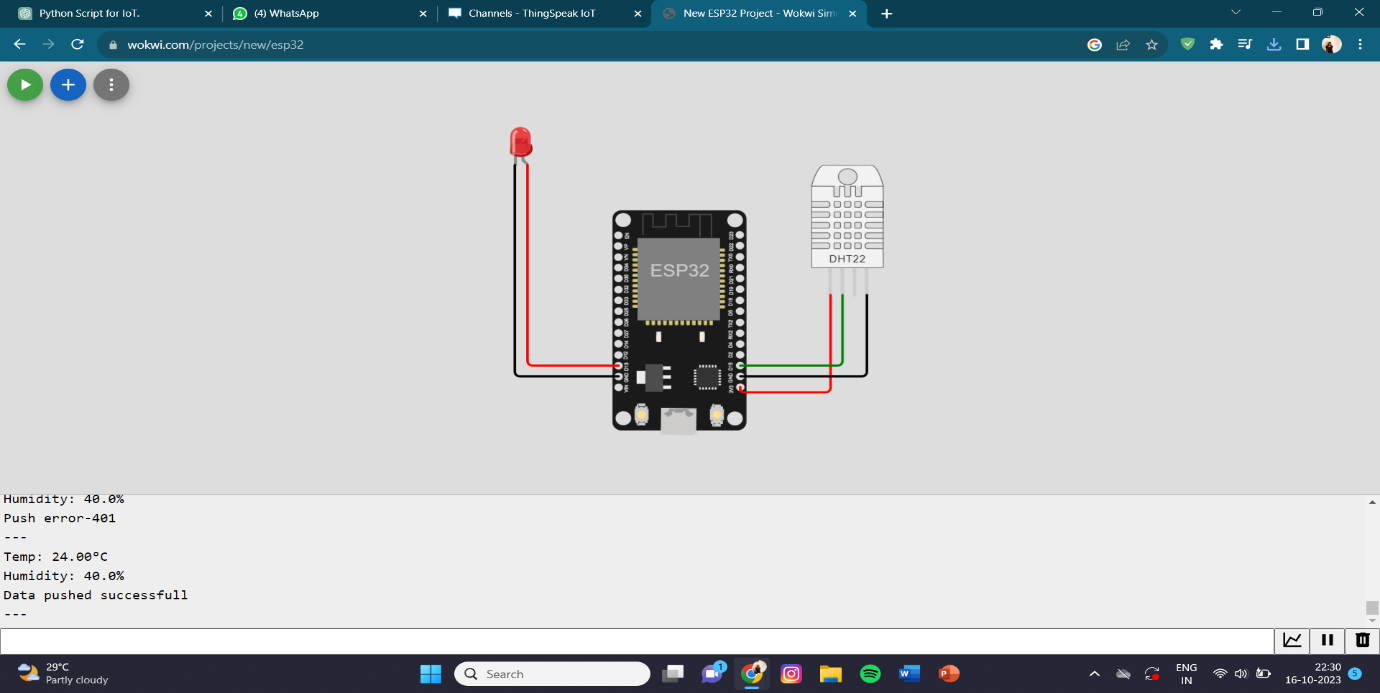
* + Develop a user-friendly web or mobile application for stakeholders, including environmental agencies, researchers, and the public.
  + Provide real-time dashboards and visualizations that display environmental data in an easily understandable format.

**IOT DESIGN:**

**COMPONENTS USED:**

* SOFTWARE USED-WOKWI
* IoT DEVICE -ESP32
* SENSOR USED -DHT22

**CIRCUIT DESIGN:**

****

**PYTHON CODE:**

import machine

importdht

import network

importurequests

import time

DHT\_PIN = 15

LED\_PIN = 2  # Assuming you are using NodeMCU or similar with built-in LED

WIFI\_NAME = "Wokwi-GUEST"

WIFI\_PASSWORD = ""

myChannelNumber = 2306875

myApiKey = "LGD2VGLYYVTP3YV9"

server = "api.thingspeak.com"

dhtSensor = dht.DHT22(machine.Pin(DHT\_PIN))

led = machine.Pin(LED\_PIN, machine.Pin.OUT)

defconnect\_wifi():

    wlan = network.WLAN(network.STA\_IF)

    ifnotwlan.isconnected():

        print("Connecting to WiFi...")

        wlan.active(True)

        wlan.connect(WIFI\_NAME, WIFI\_PASSWORD)

        whilenotwlan.isconnected():

            pass

    print("WiFi connected!")

    print("Local IP:", wlan.ifconfig()[0])

defpush\_to\_thingspeak(data):

    url = "https://api.thingspeak.com/update?api\_key={0}&field1={1}&field2={2}".format(myApiKey, data["temperature"], data["humidity"])

    response = urequests.get(url)

    returnresponse.status\_code

defread\_dht\_sensor():

    dhtSensor.measure()

    return {

        "temperature": dhtSensor.temperature(),

        "humidity": dhtSensor.humidity()

    }

connect\_wifi()

whileTrue:

    sensor\_data = read\_dht\_sensor()

    led.value(1ifsensor\_data["temperature"] >35orsensor\_data["temperature"] <12orsensor\_data["humidity"] >70orsensor\_data["humidity"] <40else0)

    response\_code = push\_to\_thingspeak(sensor\_data)

    print("Temp: {:.2f}°C".format(sensor\_data["temperature"]))

    print("Humidity: {:.1f}%".format(sensor\_data["humidity"]))

    ifresponse\_code == 200:

        print("Data pushed successfully")

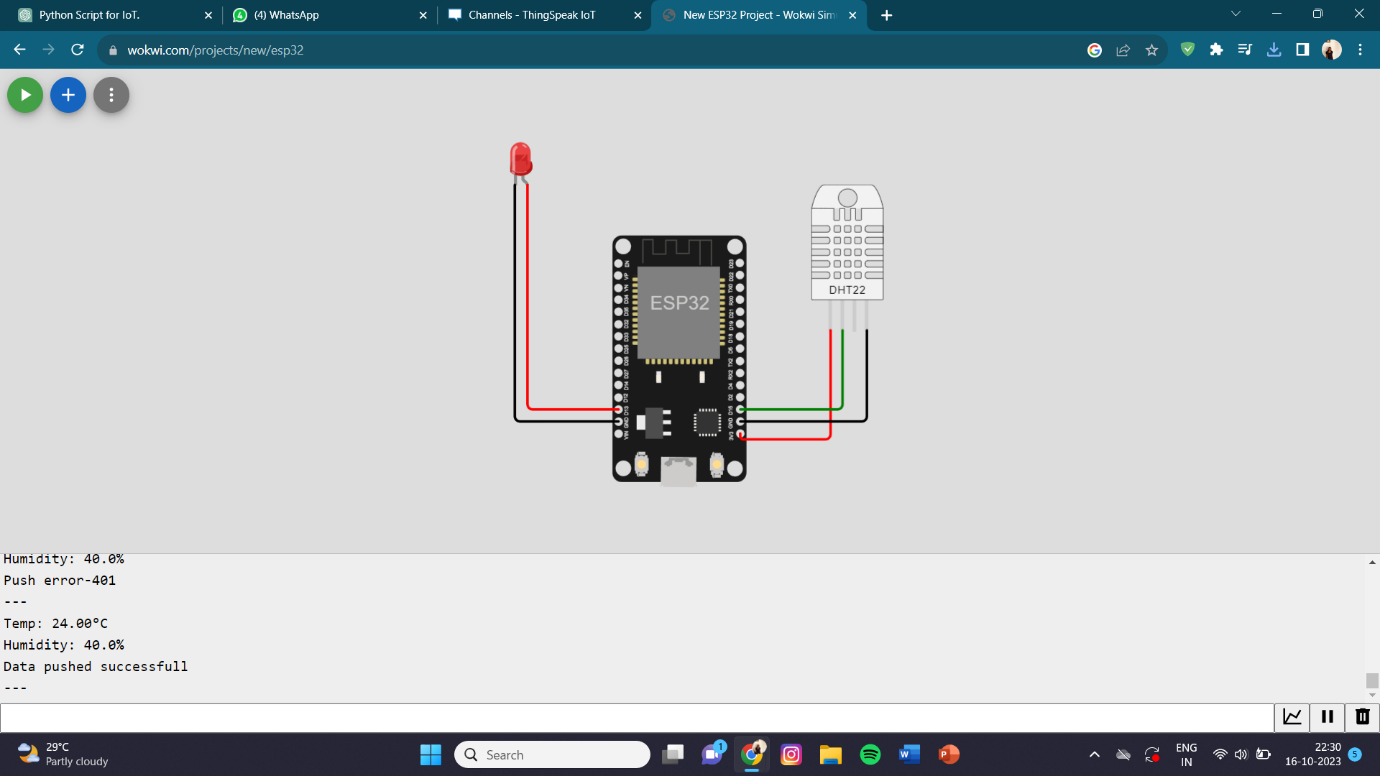
    else:

        print("Push error", response\_code)

    print("---")

    time.sleep(10)  # Sleep for 10 seconds

**OUTPUT WITH CIRCUIT :**



**DISPLAYING REAL TIME DATA USING WEB DEVELOPMENT**

**PLATFORM USED:**

* Notepad++

**LANGUAGES USED**

* HTML
* CSS
* JavaScript
* Python

**BROWSER FOR DISPLAY**

* Microsoft Edge

**CODES FOR DISPLAYING REAL TIME DATA USING WEB DEVELOPMENT**

**HTML CODE**

<!DOCTYPE html>

<html>

<head>

<title>ENVIRONMENTAL MONITORING USING IOT</title>

<link rel="stylesheet" type="text/css" href="styles.css">

</head>

<body>

<h1>REAL TIME ENVIRONMENTAL DATA DISPLAY</h1>

<div id="data-display">

<p>Loading IoT data...</p>

</div>

<script src="java.js"></script>

</body>

</html>

**CSS CODE**

body {

font-family: Arial, sans-serif;

text-align: center;

margin: 20px;

}

h1 {

font-size: 24px;

margin-bottom: 20px;

}

.data-container {

display: inline-block;

margin: 10px;

}

.data-label {

font-weight: bold;

margin-bottom: 5px;

}

.data-value {

font-size: 18px;

}

**JAVASCRIPT CODE**

constchannelID = '2306875';

constapiUrl = `https://api.thingspeak.com/channels/${channelID}/feeds.json?results=1`;

fetch(apiUrl)

.then((response) =>response.json())

.then((data) => {

if (data.feeds.length> 0) {

constlastEntry = data.feeds[0];

const field1Data = lastEntry.field1;

const field2Data = lastEntry.field2;

dataDisplay.innerHTML = `

<p>Temperature: ${field1Data}'C</p>

<p>Humidity: ${field2Data}g/m^3</p>

`;

} else {

dataDisplay.innerHTML = '<p>No data available.</p>';

}

})

.catch((error) => {

console.error('Error fetching data from ThingSpeak:', error);

dataDisplay.innerHTML = '<p>Error fetching ThingSpeakdata.</p>';

});

}

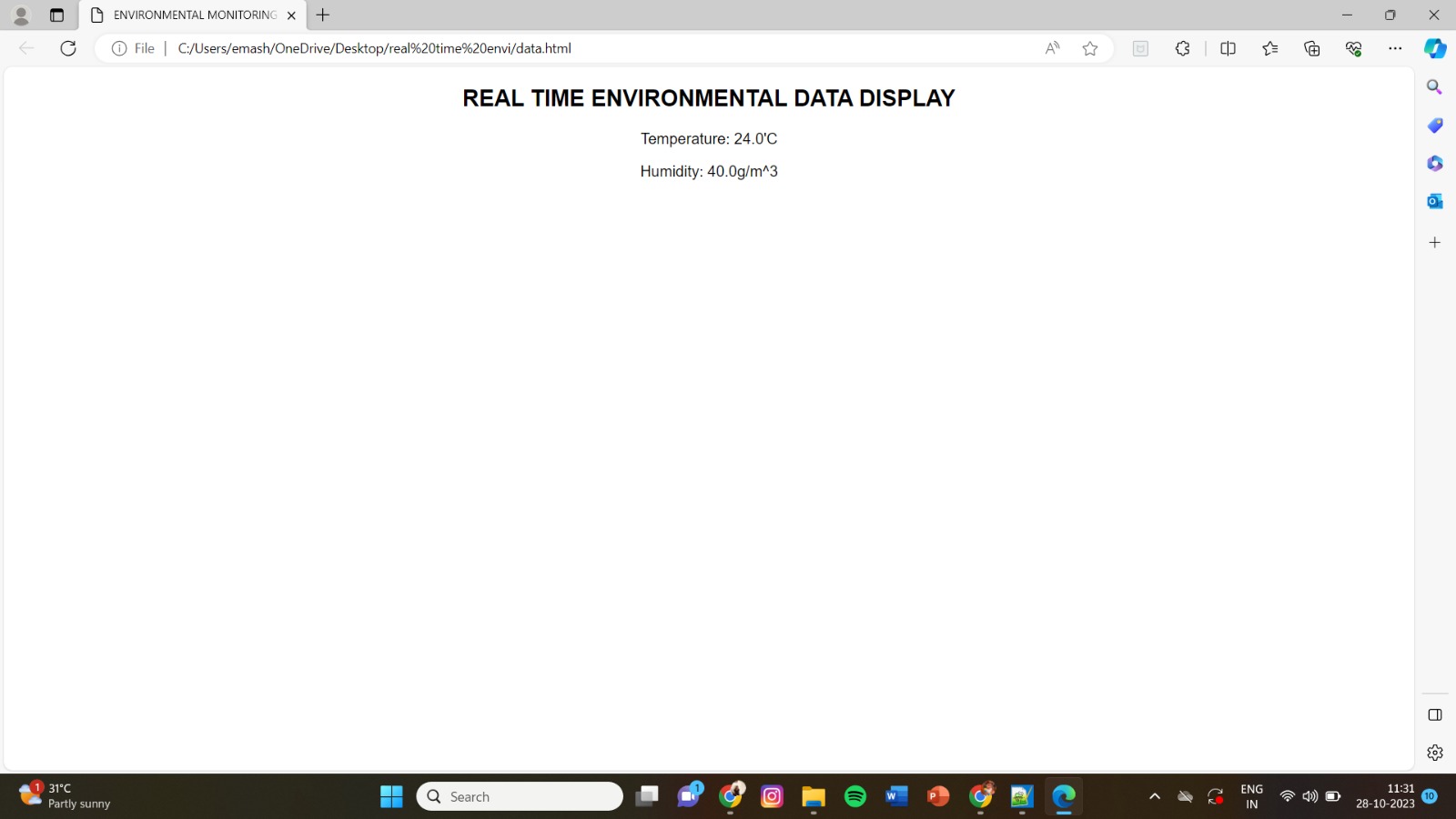
// Fetch ThingSpeak data initially

fetchThingSpeakData();

// Fetch ThingSpeak data periodically (e.g., every 30 seconds)

setInterval(fetchThingSpeakData, 30000); // 30,000 milliseconds = 30 seconds

**WEB DISPLAY**



**Real-time environmental monitoring system benefits:**

A real-time environmental monitoring system provides numerous benefits to park visitors and promotes outdoor activities in several ways. This technology leverages various sensors and data collection methods to continuously track and report environmental conditions in and around a park. Here are some of the key advantages of such a system:

**1. Safety Assurance:**

* + Early Warning: Real-time monitoring can detect and alert visitors to environmental hazards, such as severe weather, flash floods, or wildfires, helping them make informed decisions and stay safe.
  + Air Quality: Monitoring air quality can warn visitors about poor conditions due to pollution or wildfires, ensuring they avoid exposure to harmful substances.

**2. Improved Planning:**

* + Weather Forecasts: Visitors can access real-time weather data to plan their outdoor activities effectively, including hiking, picnics, or camping.
  + Crowd Control: Data on park occupancy and popular areas can help visitors plan their visits during less crowded times, improving their overall experience.

**3. Enhanced Experience:**

* + Wildlife Observation: Real-time monitoring can provide data on wildlife sightings and behavior, increasing the chances of visitors spotting animals in their natural habitat.
  + Scenic Views: Information on visibility, cloud cover, and atmospheric conditions can help visitors choose the best times for enjoying scenic vistas and photography.

**4. Sustainable Practices:**

* + Resource Conservation: Monitoring can track water levels, energy consumption, and waste management, encouraging responsible use of resources and promoting a sustainable park environment.
  + Ecosystem Protection: Data on factors like soil moisture and temperature can support conservation efforts and habitat restoration projects.

**5. Educational Opportunities:**

* + Interpretive Displays: Interactive displays and smartphone apps can provide visitors with real-time information about the park's ecology, geology, and history, enhancing their understanding and appreciation of the environment.
  + Citizen Science: Real-time data can engage visitors in citizen science projects, allowing them to contribute to research and conservation efforts.

**6. Accessibility:**

Mobile Apps: Real-time data can be made accessible through park-specific apps, enabling visitors to have up-to-date information on their smartphones or other mobile devices.

**7. Emergency Response:**

Quick Assistance: In the event of an emergency, real-time data can help emergency services respond more effectively and locate visitors in need of assistance.

**8. Ecotourism and Recreation Promotion:**

* + Attracting Visitors: Real-time monitoring data can be used in marketing efforts to attract more visitors who are interested in outdoor activities and ecotourism.
  + Event Planning: Park managers can use data on weather conditions, water levels, and wildlife activity to plan and promote special events that align with natural phenomena, such as bird migrations or wildflower blooms.

In summary, a real-time environmental monitoring system benefits park visitors by enhancing their safety, planning, and overall experience while promoting outdoor activities by providing valuable data for informed decision-making, sustainable practices, and educational opportunities. This technology contributes to a more enjoyable and responsible visit to natural areas while supporting the conservation and protection of these precious environments.

**THANK YOU**