IOT – INTERNET OF THINGS

PHASE THREE PROJECT

Start building the IoT-enabled Environmental Monitoring in Parks system.

- ➢ Creating an IoT-enabled Environmental Monitoring in Parks system involves several steps, including deploying IoT devices with sensors and developing a Python script to send real-time environmental data to a monitoring platform.

- ➢ Here's a high-level overview of how you can get started

HARDWARE SETUP

- ➢ **Choose Sensors**: Select the appropriate sensors for measuring environmental conditions in the parks. In this case, temperature and humidity sensors would be essential.

- ➢ Depending on your requirements, you may also consider sensors for other parameters like air quality, light intensity, soil moisture, etc.

- ➢ **IoT Hardware:** Choose an IoT hardware platform to connect and gather data from these sensors.

- ➢ Popular choices include Raspberry Pi, Arduino, or specialized IoT development boards

- ➢ **Connect Sensors:** Connect the selected sensors to your IoT hardware. Depending on the sensor and hardware, this might involve wiring and setting up libraries to read data from the sensors.

TO Develop a Python script on the IoT devices to send real-time environmental data to the monitoring platform

Python Script for IoT Devices:

> You'll need to develop a Python script to read data from the sensors and send it to a monitoring platform. Here's a basic example using a Raspberry Pi and DHT22 sensor for temperature and humidity monitoring:

CODE

```python
import Adafruit_DHT
import paho.mqtt.client as mqtt
import time

# MQTT parameters
mqtt_broker_address = "your_broker_address"
mqtt_topic = "environmental_data"

# Sensor setup
sensor = Adafruit_DHT.DHT22
sensor_pin = 4  # GPIO pin where the sensor is connected

# Create an MQTT client
client = mqtt.Client("Park_Environmental_Monitor")

# Connect to the MQTT broker
client.connect(mqtt_broker_address, 1883)

try:
    while True:
        # Read data from the sensor
        humidity, temperature = Adafruit_DHT.read_retry(sensor,
sensor_pin)

        if humidity is not None and temperature is not None:
            data = {
                "temperature": temperature,
                "humidity": humidity,
            }

            # Publish data to the MQTT topic
            client.publish(mqtt_topic, str(data))
```

```
                print(f"Published: {data}")
            else:
                print("Failed to retrieve data from the sensor.")

            # Send data at regular intervals
            time.sleep(300)  # Adjust the interval as needed

    except KeyboardInterrupt:
        print("Script terminated by the user.")

    # Disconnect from the MQTT broker
    client.disconnect()
    }
```

Monitoring Platform:

> **Set up an MQTT Broker**: You need an MQTT broker (e.g., Mosquitto) to handle the incoming data from the IoT devices.

> **Data Processing**: Develop a component or application to subscribe to the MQTT topic, retrieve data, and process it.

> You can store data in a database, perform real-time analysis, and create visualizations.

> **User Interface**: Create a web-based or mobile app for users to access real-time and historical environmental data.

> **Alerts and Notifications**: Implement alerting mechanisms for abnormal environmental conditions.

> This is a basic starting point for your project.

➢ Depending on your specific requirements, you may need to add more sensors, data validation, security measures, and advanced features.

➢ Additionally, consider the power source and data communication options suitable for remote park locations.