Name : P.Arivumani
Nm ID : au621421106004

ENVIRONMENTAL MONITORING IN PARKS

Environmental monitoring :

First, let's talk about an environmental monitoring system definition and how IoT supports these processes. IoT-based environmental monitoring is the consistent collection of measurements and data from our physical environment, using sensors and connected devices. Sensors embedded in irrigation systems, pipelines, tanks, weather stations, oceanic applications, and industrial equipment – anywhere on the planet – can detect temperature, moisture, water levels, leaks, and other physical properties.

These monitoring systems can be programmed to detect abnormalities or specific conditions, then trigger alerts via email or text, as well as automated processes. These can include anything from launching service tickets to shutting systems down to thwart a disaster.

Components :

There are four essential components for IoT-based environmental monitoring to support critical insights and decision making:

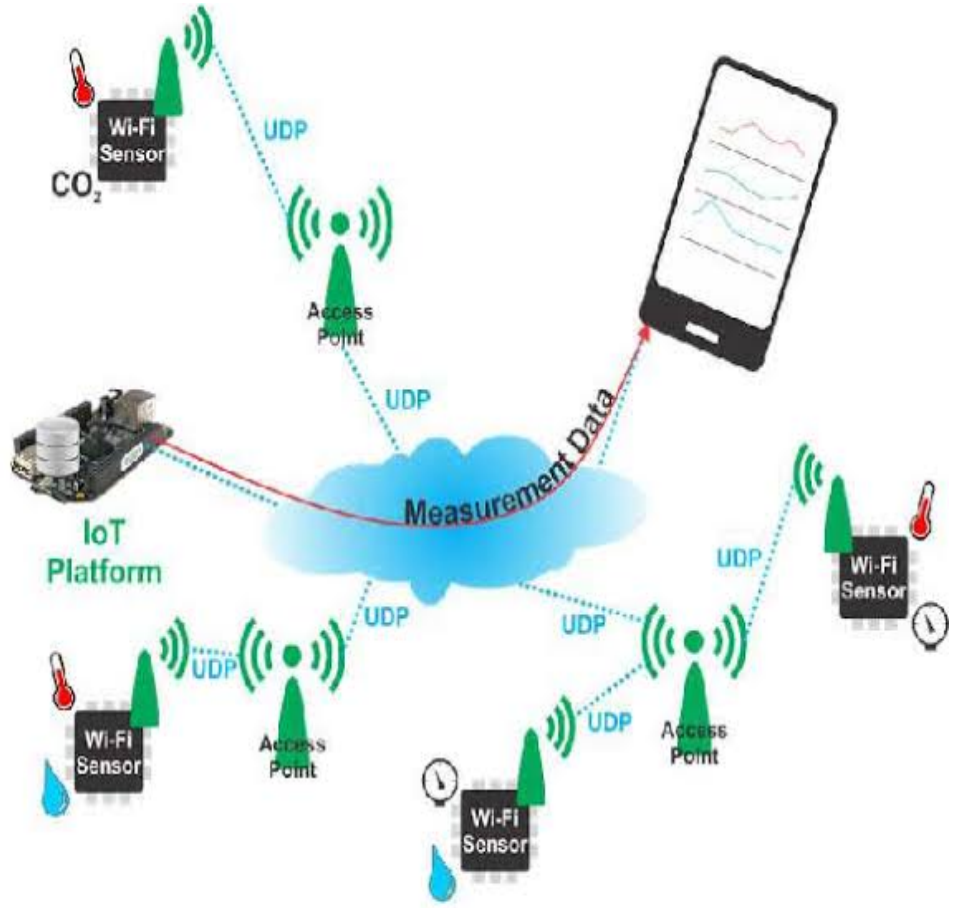
1. Monitor the Environment

2. Measure Data

3. Catalog Data

4. Provide Actionable Insights from the Data and Analysis

Diagram :



g. 1. CPS for environmental monitoring.

Python script :

Creating an environmental monitoring Python script for parks can be a complex task, but I can provide you with a basic outline to get you started. This script will focus on monitoring temperature and humidity, but you can expand it to include other environmental parameters as needed.

```python
import random
import time

# Simulate environmental data (replace with actual sensors)
def get_environment_data():
    temperature = random.uniform(10, 30)
    humidity = random.uniform(30, 70)
    return temperature, humidity

# Logging function
def log_data(temperature, humidity):
    with open("environmental_data.csv", "a") as file:
        file.write(f"{time.time()}, {temperature}, {humidity}\n")

# Main loop for continuous monitoring
while True:
    temperature, humidity = get_environment_data()
    log_data(temperature, humidity)
    print(f"Temperature: {temperature}°C, Humidity: {humidity}%")
    time.sleep(60)  # Log data every minute (adjust as needed)
```

This script generates random temperature and humidity values (you should replace this with actual sensor readings) and logs them to a CSV file for further analysis. Make sure to adjust the code according to the specific sensors and data storage requirements of your park's environmental monitoring system.

Additionally, you can extend this script to include more sensors, set up alerts, or visualize the data. Libraries like matplotlib and pandas can be helpful for data visualization and analysis, while libraries like NumPy can assist with data manipulation.