**PHASE -2**

**PROBLEM AND INNOVATION PROCEDURE**

|  |  |
| --- | --- |
| **Date** | 10/09/2023 |
| **Team ID** | Proj\_223986\_Team |
| **Project Name** | Environmental Monitoring in Parks |
| **Students with ID** | **Vimal K**  Nirmalkumar K  Parthiban G  Shreepal S  Sabari Nathan C |

**TITLE: Environmental Monitoring**

Certainly! Let's break down the steps to implement the IoT-based environmental monitoring system for public parks using ESP32 and other sensors, following the design thinking approach outlined in your document.

**1. EMPATHIZE:**

**Action 1:** Conduct surveys or interviews with park visitors to understand their outdoor activity planning needs.

**Action 2:** Analyse historical environmental data and trends in park areas to identify critical factors.

**Action 3:** Collaborate with park authorities and environmental organizations to gather insights from experts.

**2. DEFINE:**

**Objective 1:** Develop a reliable IoT-based environmental monitoring system with less than X% margin of error.

**Objective 2:** Create a user-friendly web application for park visitors to access environmental data.

**Action 1:** Explore various IoT devices and sensors for data collection, including selecting the ESP32 microcontroller.

**Action 2:** Investigate machine learning algorithms for data analysis and predictions.

**Action 3:** Consider user interface designs for the web platform to ensure usability.

**3. PROTOTYPE:**

**ARCHITECTURE:**

**Sensors:**

* Temperature and humidity sensors (e.g., DHT11, DHT22) placed in the public park to measure environmental conditions.

**Arduino Board:**

* Central controller responsible for reading sensor data and communicating with the ESP32.

**ESP32 Microcontroller:**

* Manages connectivity, sending data to the cloud, and handling interactions between sensors and the Arduino board.

**CLOUD PLATFORM:**

* Thing Speak, a cloud service, stores, visualizes, and analyses the sensor data.

**USER INTERFACE:**

* Web application hosted on Thing Speak provides a user-friendly interface for real-time environmental data access.

**PROTOCOLS:**

**Inter-Integrated Circuit:**

* Used for communication between Arduino and sensors.

**Serial Communication:**

* Communication between Arduino and ESP32 for data exchange.

**Wi-Fi:**

* ESP32 connects to the local Wi-Fi network for internet communication.

**HTTP (Hypertext Transfer Protocol):**

* ESP32 sends HTTP POST requests to Thing Speak for data transmission.

**Thing Speak API:**

* API is used for processing incoming data and updating specific fields in ThingSpeak.

**DESIGN:**

**Sensor Data Collection:**

* Sensors collect temperature and humidity data from the public park.

**Arduino Processing:**

* Arduino reads data from sensors using I2C and communicates with ESP32 through serial **communication.**

**ESP32 Handling Communication:**

* Manages Wi-Fi connectivity, sending HTTP requests, and interfacing with the cloud.

**Cloud Processing:**

* ThingSpeak processes incoming data using its API, updating fields for temperature and humidity.

**USER INTERFACE UPDATE:**

Web application hosted on Thing Speak provides a real-time view of environmental data.

**Action 1:** Develop a prototype system for data collection, storage, and transmission, including the integration of temperature and humidity sensors (using sensors compatible with ESP32).

**Action 2:** Design a user interface mock-up for the web platform displaying real-time temperature and humidity readings.

**Action 3:** Test the prototype with a limited dataset to ensure data accuracy, including temperature and humidity, and user interface functionality.

**FLOW DIAGRAM:**

**4. TEST:**

**Action 1:** Conduct field tests of IoT sensors, including temperature and humidity sensors, to validate data accuracy and reliability.

**Action 2:** Collect user feedback on the web platform for usability and functionality, with a focus on temperature and humidity information.

**Action 3:** Assess the prototype’s performance against predefined objectives, including temperature and humidity data quality.

**5. IMPLEMENT:**

**Action 1:** Deploy IoT devices (ESP32 with sensors) in selected public parks and establish data transmission to the cloud (using platforms like AWS IoT or Azure IoT).

**Action 2:** Develop the web platform, incorporating user feedback and improvements.

**Action 3:** Conduct comprehensive testing to ensure system robustness and user-friendliness.

**6. ITERATE:**

**Action 1:** Monitor data accuracy and refine data analysis algorithms.

**Action 2:** Address user feedback and make necessary improvements to the web interface.

**Action 3:** Stay informed about emerging technologies and environmental monitoring advancements for potential system enhancements.

**CONCLUSION:**

In conclusion, this approach involves understanding user needs, defining clear objectives, prototyping with ESP32 and environmental sensors, testing, implementing, and continuous iteration based on user feedback and technological advancements. By using ESP32 and other sensors, the project aims to create a comprehensive and reliable environmental monitoring system for public parks.