Project Title: Autonomous Indoor Air Quality Monitoring and Control System

Objective:

Design and develop a smart system that monitors indoor air quality and automatically regulates it to ensure optimal conditions using sensors, actuators, and a microcontroller-based platform.

Key Features:

- Air Quality Monitoring: Measure particulate matter (PM2.5/PM10), temperature, and humidity using sensors.

- Data Display: Display real-time sensor data on an LCD, OLED, or TFT display.

- Data Logging: Store data on an SD card or upload it to the cloud for long-term tracking using Wi-Fi.

- Control Mechanism: Actuate devices like fans, air purifiers, or humidifiers based on predefined thresholds.

- User Interaction: Include buttons or a touchscreen interface for user inputs and advanced control.

- Communication (optional): Send notifications to a smartphone or computer via Bluetooth or Wi-Fi.

- Autonomous Operation: Automatically adjust actuators based on sensor readings.

- Enclosure and Aesthetic: Design a compact, aesthetically pleasing enclosure to house the system.

Suggested Hardware:

1. Microcontroller Platform: Arduino, Raspberry Pi, or STM32, TI TivaWare Launchpad

2. Sensors: PM2.5 sensor, temperature and humidity sensor, VOC sensor.

3. Actuators: DC/AC fan, relay modules.

4. Display: 16x2 LCD, OLED, or TFT display.

5. Power Supply: Ensure compatibility with system components.

6. Optional Modules: Wi-Fi module, Bluetooth module.

Suggested Software Features:

- Microcontroller Programming: Arduino IDE, Python, or Bare-metal C/C++.

- Threshold-Based Automation: Control actuators when sensor readings exceed thresholds.

- Data Visualization: Use libraries like Adafruit_GFX for microcontrollers or Matplotlib for Raspberry Pi, or other GUI library

- Data Logging and Communication: Use SD libraries or MQTT protocols for logging and communication.

Suggested Timeline:

Semester 1:
1. Research and Component Selection
2. Hardware Prototyping, and Unit Testing
3. Basic Data Acquisition and Display
4. Control Algorithm Design

Semester 2:
5. Implement Automation and Actuator Control
6. Add Communication Features
7. Enclosure Design and Final Assembly
8. Testing, Debugging, and Optimization

Learning Outcomes:

- Design and integration of sensor-actuator systems.

- Experience with embedded programming and microcontroller interfacing.

- Data visualization and user interface development.

- Project management and iterative design methodology.

Budget: $500