Smart Air Pollution Monitoring System

The Air Pollution Monitoring System is a Raspberry Pi-based project that monitors air quality and environmental factors such as gas levels, temperature, and humidity. It provides real-time data on a web dashboard and sends alerts through SMS for critical conditions. It also integrates with ThingSpeak for data logging and uses a buzzer for physical alerts.

Required Libraries

1. Backend Libraries

- **Flask:** Web framework to host the backend API.
    1. **Install:** pip install flask
- **Adafruit_DHT:** Interface with the DHT11 temperature and humidity sensor.
    1. **Install:** pip install Adafruit_DHT
- **Adafruit_GPIO and Adafruit_MCP3008:** Manage GPIO pins and MCP3008 ADC.
    1. **Install:** pip install Adafruit-GPIO Adafruit-MCP3008
- **RPi.GPIO:** GPIO library for Raspberry Pi
    1. **Install:** pip install RPi.GPIO
- **Twilio:** Send SMS alerts.
    1. **Install:** pip install twilio
- **Requests:** Send HTTP requests to ThingSpeak.
    1. **Install:** pip install requests

2. Frontend Libraries

- **Bootstrap:** CSS for a responsive, styled UI.
- **Font Awesome:** Icons for visual enhancements.
- **Chart.js:** Render dynamic charts.
    1. All are included via CDNs in the HTML files.

Project Setup

1. Hardware Setup:

- Connect the DHT11 sensor, MCP3008 ADC, and buzzer to Raspberry Pi GPIO pins as specified in app.py.

2. Software Setup:

- Clone or place project files.
- **app.py**: Backend server code.
- **login.html:** Simple login page for access control.
- **dashboard.html**: Displays sensor data and charts.

3. Configuration:

- Update app.py with:
    1. **Twilio Credentials:** Replace placeholders for TWILIO_ACCOUNT_SID, TWILIO_AUTH_TOKEN, and TWILIO_PHONE_NUMBER.
    2. **ThingSpeak API Key:** Replace THINGSPEAK_WRITE_API_KEY.
    3. **Phone Number:** Replace ALERT_PHONE_NUMBER for SMS alerts.

4. Execution:

- **Start the Flask server:** python app.py.
- Access http://:5000 in a web browser.
- Open the dashboard with Login credentials userid and password.

5. Testing:

- Simulate high AQI values to test SMS alerts and buzzer functionality.
- Check ThingSpeak for data logs.

SNAPSHOTS

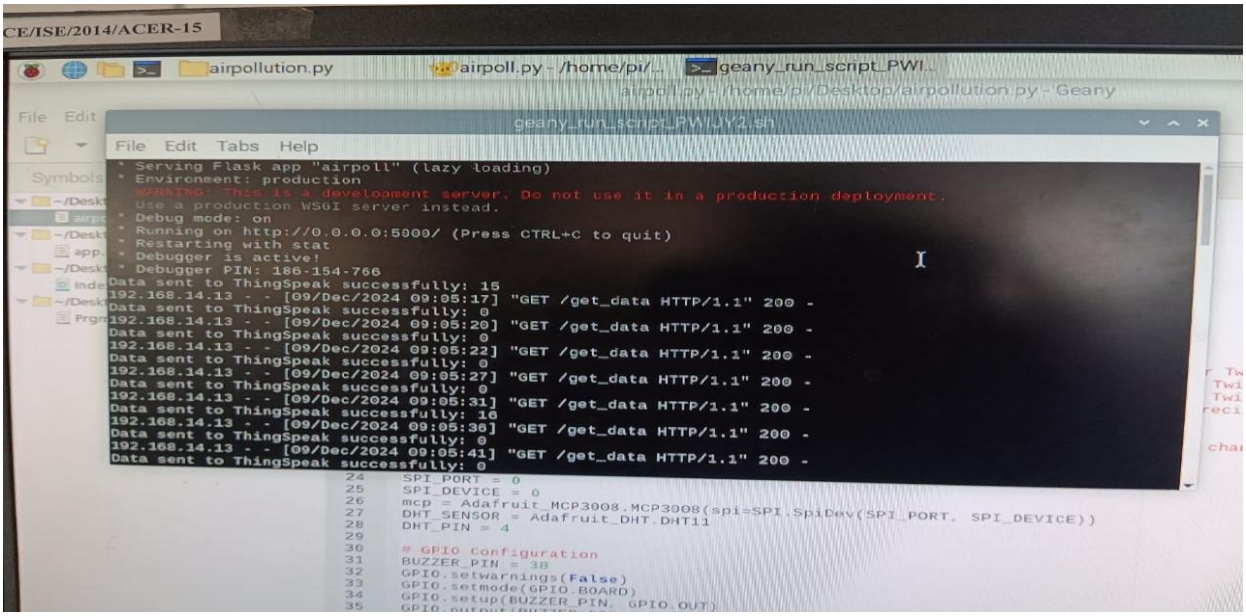

Figure1: Flask app running

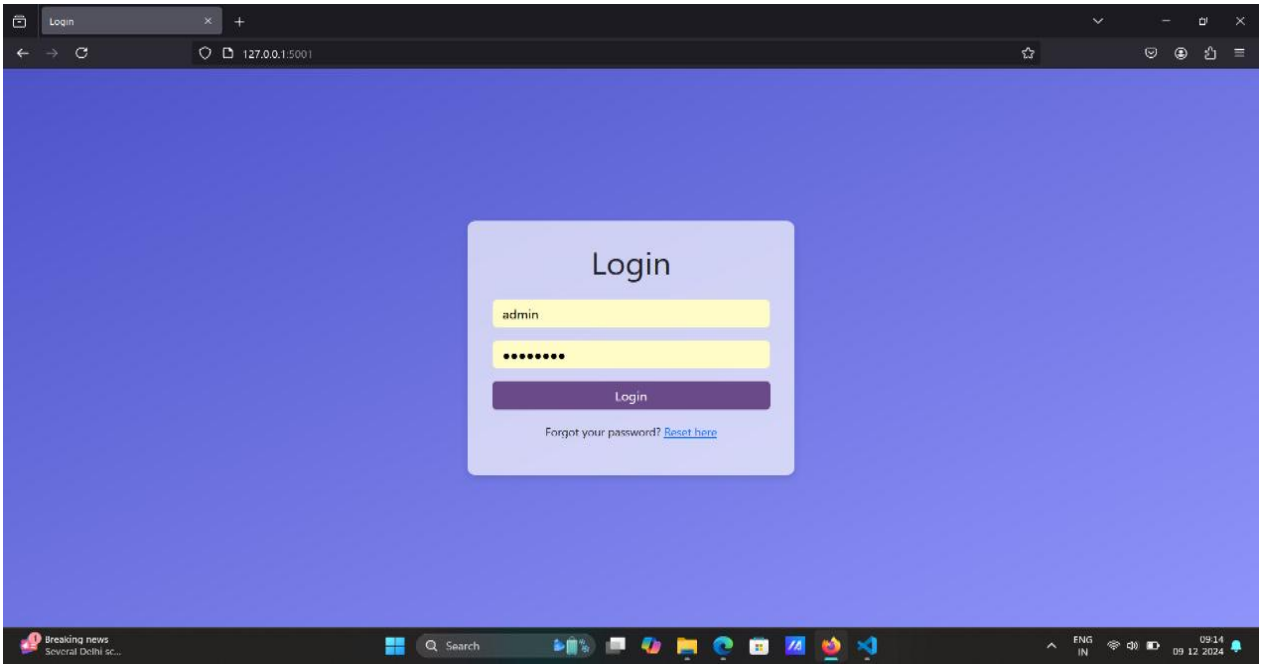

Figure2: Login Page

Figure3: Dashboard

Figure4: Chart & Recommendations

Figure5: SMS Alert

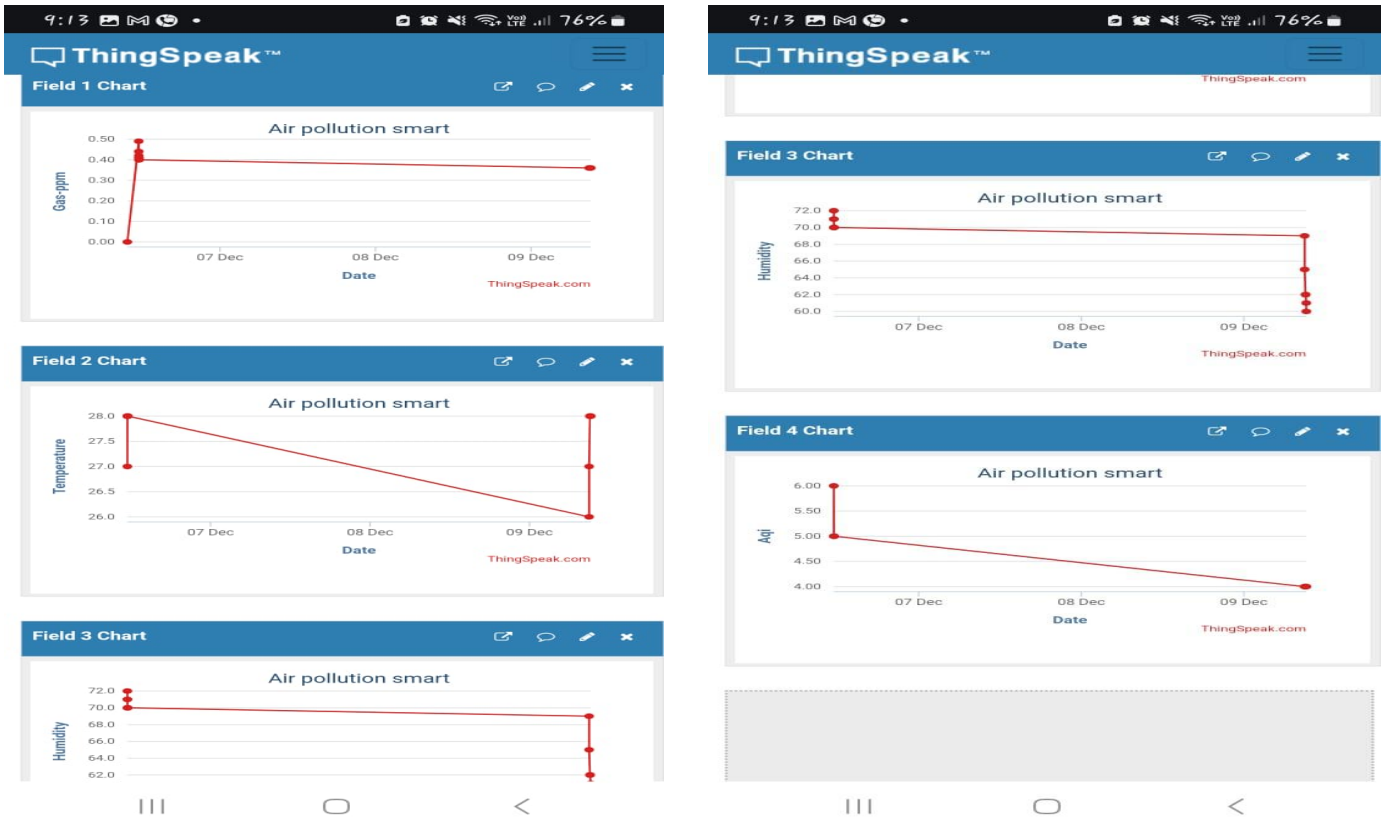

Figure6&7: ThingSpeak Data