Test Strategy Document

Airscape

Prepared By : Thanuga Soysa

Date : 2025 / 04 / 21

Project Name : Airscape Realtime AQI Monitoring for Colombo

Version : 1.0

1. Objective

This test strategy outlines the structured approach to verify the quality, stability, and correctness of the Airscape Real-Time Air Quality Monitoring System. The goal is to ensure that:

- All core functionalities are implemented as per the project requirements
- Sensor data simulation and alert rules behave correctly
- The system is secure, user-friendly, and stable across real-world scenarios
- The system is ready for real-time public and administrator usage

This document acts as a blueprint for both manual and automation testing phases.

2. Scope of Testing

In-Scope Features:

- Public User Dashboard (Map, AQI Popup, about, contact)
- Admin Panel Modules:
- Dashboard Overview
- Sensor Management (CRUD, status)
- Data Management (filter, summary)
- Alert Configuration (thresholds, system/email alerts)
- Admin User Management (add/edit/delete Admins)
- Role-Based Authentication (Web Master vs Admin access)
- Simulated AQI Data Logic
- Real-time & historical data display
- User interaction flows across UI

Out of Scope:

- Load/Stress Testing (handled later in automation)
- Mobile responsiveness (unless specified)

3. Types of Testing

- Sanity Testing - Verify critical functions work before starting full test cycles
- Functional Testing - Check if all features meet functional requirements
- UI/UX Testing - Validate design consistency, usability, responsiveness
- Data Testing - Validate AQI simulation logic & database consistency
- Exploratory Testing - Conduct unscripted sessions to uncover hidden bugs
- Regression Testing - Re-test modules after changes or bug fixes

4. Test Levels

- Sanity Testing - Run after deployment or new builds.
- System Testing - End-to-end verification of all modules
- Integration Testing - Verify communication between modules (e.g., sensors & alerts)
- Regression Testing - Re-check old features after changes
- Exploratory Testing - Based on user intuition and creative usage patterns

5. Tools & Frameworks

- Test Case Management - `.md` files
- API Testing - Postman
- Browser Testing - Chrome DevTools, Firefox
- Manual Execution - VS Code for test case writing, browser for testing
- Automation (Phase 2) - Playwright
- Reporting - Notion, Screenshots, GitHub Issues (optional)

6. Deliverables

- test-strategy.md
- test-plan.md (with schedule, timeline, features)
- Test cases by module (Admin Dashboard, Sensor, Data, AdminUser, Alert, User)
- Bug report file with evidence
- Automation test scripts (Playwright phase)
- Final QA report with summary & status

7. Entry & Exit Criteria

Entry Criteria:

- Development completed for Airscape main modules (Admin Dashboard, Sensor, Data, AdminUser, Alert, User)
- Simulated data generation is functional
- Sensor data saved and displayed on the dashboard

Exit Criteria:

- All critical and high test cases are passed
- No unresolved critical bugs
- All regression cycles completed and verified
- QA documentation and test evidence submitted

8. Risk Management

- Simulated data delays - Use manual AQI input for edge case tests
- Role mismatch - Test role-based login on all pages
- Alert not triggering - Simulate boundary values to test rules

9. QA Author

Thanuga Soysa

Manual Testing – Airscape QA Project