
Software Engineering
Software Requirements Specification
(SRS) Document

Air Quality Management System

08-12-2021
Version: 1.0

Group: Marpions
SUEZ Matteo, ANDRIANO Giancarlo

Table of Contents

1. Introduction.....	2
1.1 Purpose.....	2
1.2 Intended Audience.....	2
1.3 References.....	2
2. General Description.....	2
2.1 Product Features.....	2
2.3 User Class and Characteristics.....	2
2.4 Operating Environment.....	2
2.5 Assumptions and Dependencies.....	2
3. System Requirements.....	3
3.1 Functional Requirements.....	3
4. Interface Requirements.....	3
4.1 User Interfaces.....	3
4.2 Software Interfaces.....	4

1. Introduction

1.1 Purpose:

The goal of the software is to provide functionalities for AQI index evaluation and comparison between locations, basing on the provided dataset of registered data. The program works using the concentration of pollutants associated to bad air quality: SO₂, O₃, PM₁₀ and NO₂ and compares them with the European Environment Agency standards.

1.2 Intended audience:

INSA Lyon IST-SOE professor and class

1.3 References:

<https://www.eea.europa.eu/themes/air/air-quality-index>

https://en.wikipedia.org/wiki/Air_quality_index

2. General Description

2.1 Product features:

- calculate pollutants specifics basing on location and date
- find different location similarities studying their AQI or pollutants concentrations
- compare pollutants concentrations from two different locations
- compute AQI of a given location, and provide official european recommendations

2.3 User class and characteristics:

An user-friendly GUI provides easy and intuitive accessibility to all functionalities. There is no specific end user.

2.4 Operating environment:

Tested and working on Windows 10 64-bit

2.5 Assumptions and dependencies:

The software needs only files in the Installation Folder. This is the folder content:

- Folders: app , assets , data, __pycache__, .vscode , tests
- Files: aggregator.py, evaluator.py , gui.py , main.py , system.py , utilities.py

3. System Requirements

3.1 Functional requirements:

The software can run on any 64-bit laptop/Desktop with Windows 10 installation.

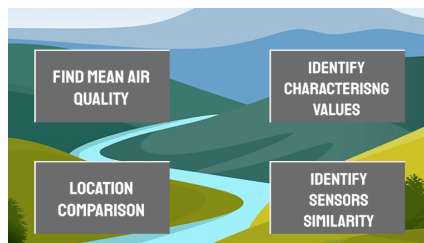
4. Interfaces Requirements

4.1 User Interfaces

Physical interactions are required. Here the general flowchart:

USER → GUI → APPLICATION

In the GUI, the user interacts with those two windows:

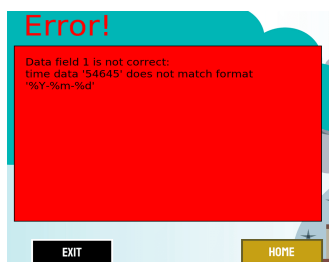


select action

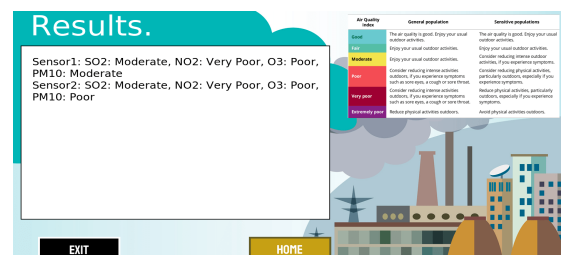
The "insert data" window has a green forest background. It features two dropdown menus for "Location" and "Location #2 (for comparison)". Each dropdown menu lists three sensor locations with their coordinates: Sensor0: 8.16° W, 34.77° S; Sensor1: 30.06° W, 76.34° S; and Sensor2: 38.92° E, 89.24° S. Below the dropdowns are two text input fields for "Date [yyyy-mm-dd]" and "Date #2 [yyyy-mm-dd] (for times)". At the bottom, there are three buttons: "HOME" (orange), "DELETE" (red), and "SUBMIT" (green).

insert data

In the Insert Data window, the user can either submit the data and get the results, or reset all the fields. Then, after submission, the output can be: an error signaling window, in case of wrong/not correct inputs or a window containing the results.



bad inputs



results + EU suggestion table

4.2 Software Libraries

The software already contains all necessary libraries, thus only files in the Installation Folder are required. For completeness, here there's a list of all Python3 libraries that have been used in the software development:

- Pandas
- datetime
- Tkinter
- PIL
- pathlib