POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Software Release Document

Dispatchment of air quality data through interactive maps and time series

Authors
Milad Ramezani Ziarani
10930504

Deliverable:	SRD
Title:	Software Release Document
Authors:	Milad Ramezani Ziarani
Version:	1.0
Date:	18th of June 2023
Copyright:	Copyright © 2023, Milad Ramezani Ziarani – All rights reserved

Revision history

Version	Date	Change
version 1.0	18 th of June, 2023	Second submitted version

Table of Contents

1 Introduction

1.1 Introduction

We are excited to announce the launch of our interactive Jupyter dashboard application! In this release, we will provide you with a comprehensive overview of the features and functionality available. Our application is designed to offer convenient access to air quality data collected from ground sensor stations across European countries. With interactive visualizations at its core, our dashboard empowers both public authorities and citizens to gain valuable insights into air pollution trends and detailed information.

1.2 Short Overview

In today's world, addressing the impact of air pollution on the environment and human health is a significant challenge. To contribute to the collective efforts, we are proud to introduce our interactive Jupyter dashboard application. This application serves as a valuable tool for accessing and analyzing air quality data in European countries.

In this initial release, we offer users a stable and reliable version of the application, ensuring a seamless experience. With our dashboard, users can explore air quality measurements collected online for the previous 48 hours. The data includes information on pollutants such as Benzene, nitrogen dioxide, sulfur dioxide, carbon monoxide, ozone, and nitrogen oxides.

Our dashboard provides intuitive visualizations and interactive features, empowering users to identify pollution patterns and gain a deeper understanding of air quality trends. By making this information accessible and easy to analyze, we aim to contribute to the efforts of individuals, organizations, and authorities in combating air pollution and promoting a healthier environment.

2 Features

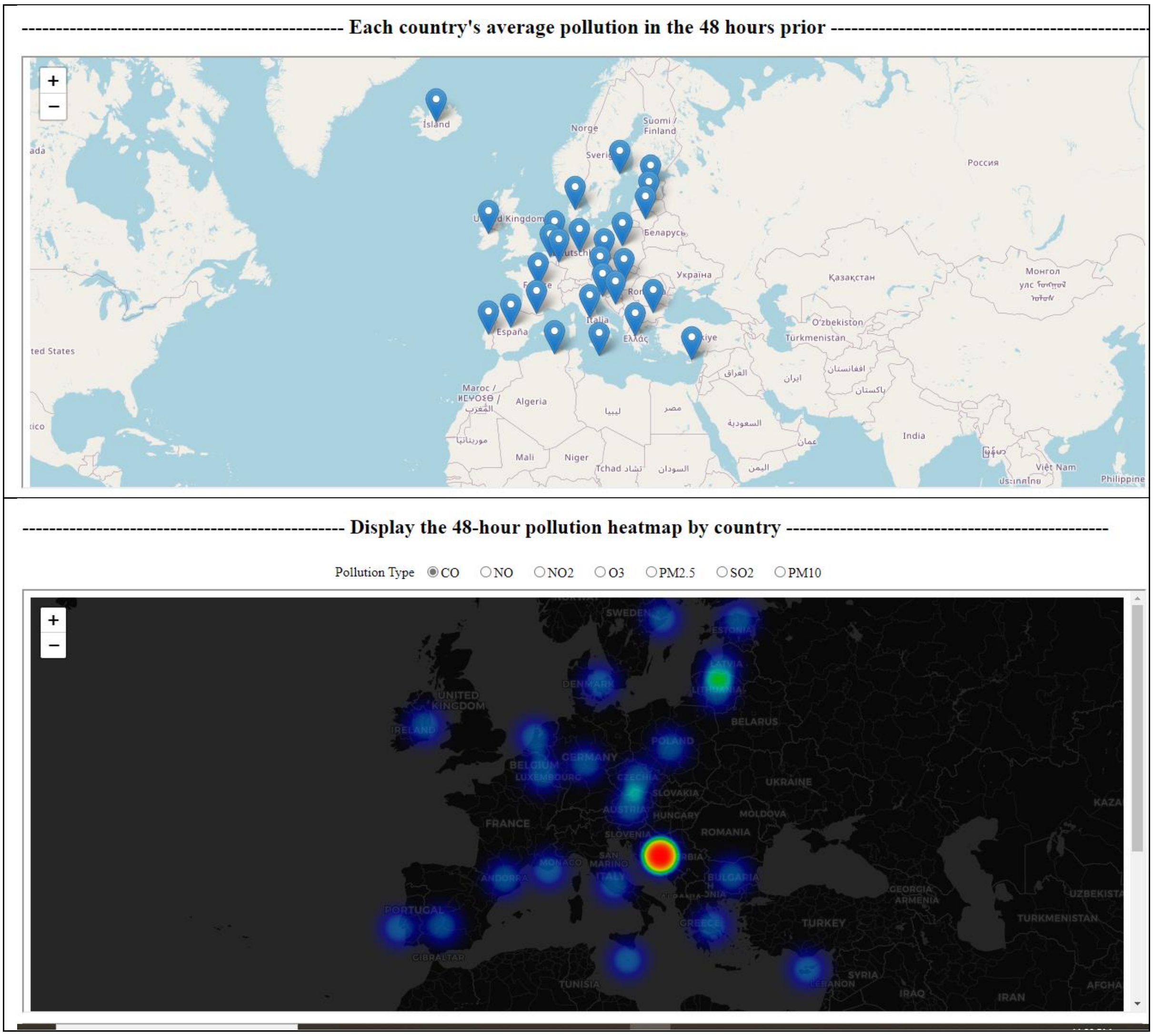
2.1 Overview of Stations

The initial section of our interactive Jupyter dashboard provides a comprehensive overview of the distribution of stations across European countries. This section includes two distinct maps that offer valuable insights to users.

The first map, titled "Stations and Sensor Types," presents a visual representation of the spatial distribution of stations throughout the European countries. By interacting with this map, users can click on individual stations to retrieve specific information, including the station name and the corresponding sensor type. Each sensor type is responsible for monitoring a specific pollutant in the region, enabling users to understand the monitoring capabilities of each station.

In the second map, labeled "Station Density," a heat map is displayed to visualize the density of stations across the European countries. This heat map provides a clear indication of the concentration and clustering of stations in different areas, allowing users to identify regions with a higher density of monitoring stations. This information is crucial for assessing the coverage and accessibility of air quality monitoring in various parts of the European countries.

Both maps serve as informative tools, empowering users to explore the station network and gain a deeper understanding of the distribution of monitoring stations and their associated sensor types across the European countries. This knowledge facilitates better decision-making and enhances the ability to analyze air quality data effectively within the Lombardi region.



-- **Each country's average pollution in the 48 hours prior** --

-- **Display the 48-hour pollution heatmap by country** --

Pollution Type ● CO ○ NO ○ NO2 ○ O3 ○ PM2.5 ○ SO2 ○ PM10

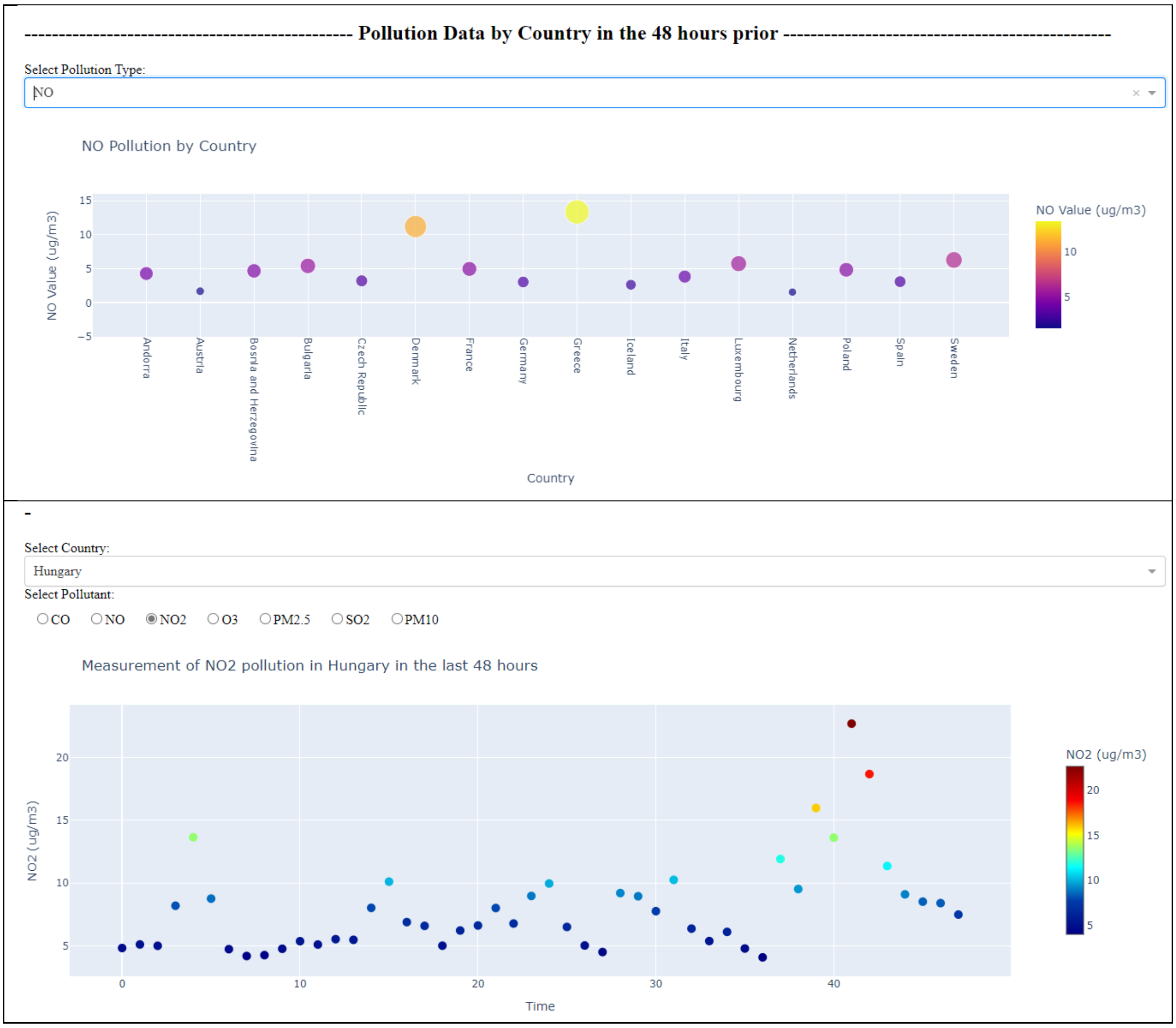
2.2 Statistics

The "Statistics" section of our interactive Jupyter dashboard offers users two informative graphs to facilitate the analysis of sensor observations.

The first graph presents the lowest recorded values for each sensor over the specified period. By examining this graph, users can gain insights into the minimum values observed by each sensor, providing a reference for the lowest levels recorded during the designated timeframe.

The second graph showcases the values observed in each country for each sensor. This graph allows users to analyze and compare the recorded values across different countries, providing an overview of the sensor readings during the specified timeframe. By studying this graph, users can identify any variations or patterns in the sensor observations among the countries.

These graphs serve as valuable tools for users to analyze and interpret the statistical characteristics of sensor observations. By examining the minimum, maximum, and mean values displayed in the graphs, users can develop a better understanding of the typical observation ranges and tendencies for each sensor over the designated time period. This information enables users to make informed decisions and draw meaningful conclusions based on the statistical analysis of sensor data.



-- Pollution Data by Country in the 48 hours prior --

Select Pollution Type:

NO

Select Country:

Hungary

Select Pollutant:

○CO ○NO ●NO2 ○O3 ○PM2.5 ○SO2 ○PM10

2.3 Sensors Map

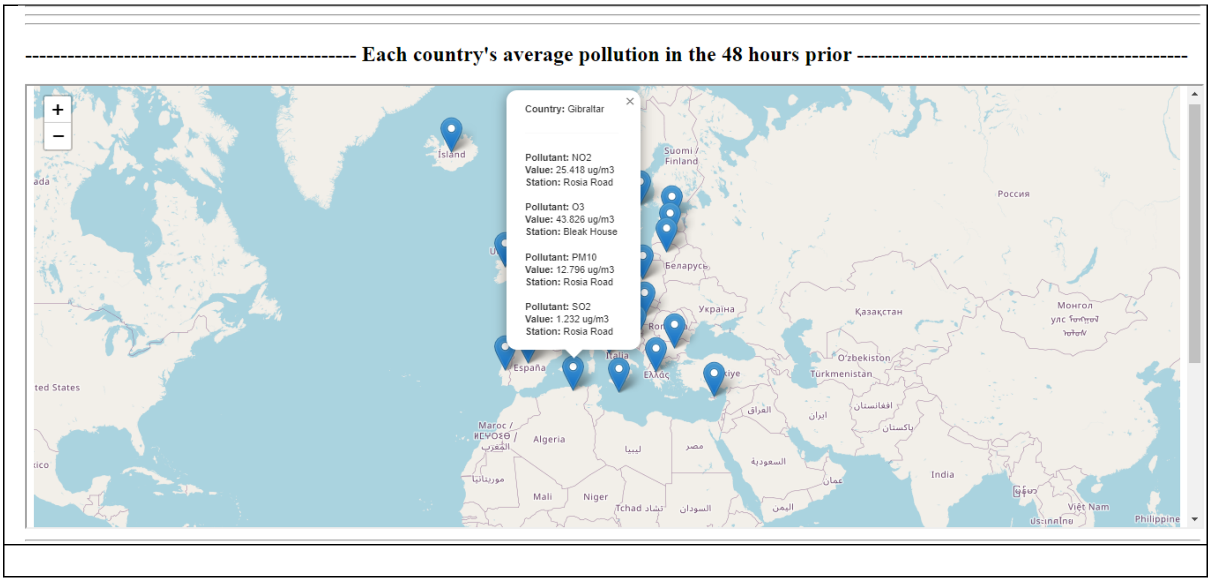
The purpose of this section in our interactive Jupyter dashboard is to provide users with comprehensive information about the sensors. It offers various features designed to enhance the user experience and facilitate the exploration of sensor data.

To enhance usability, users can utilize a dropdown menu to sort sensors based on the countries they are located in or the type of pollution they measure. This sorting functionality enables users to quickly identify and focus on specific sensors of interest. Additionally, there is an option labeled "All sensors" that allows users to view all sensors on the map simultaneously, providing a comprehensive overview.

By clicking on individual sensors, users can access a popup window that displays detailed information about each sensor, including the sensor ID, sensor type, and the country where it is located. This feature provides users with valuable insights into the specific attributes and locations of the sensors, allowing for a deeper understanding of the data collected.

Once users have selected their desired sensor, they can utilize the subsequent feature to plot the data associated with that sensor. This integration of data plotting and mapping allows users to visualize and analyze the data collected by their chosen sensor. The mapping component of this feature utilizes Folium, a powerful library that provides additional functionalities such as zoom control, full-screen mode, and a scale bar, enhancing the overall user experience.

In summary, this section offers a user-friendly interface that facilitates the exploration and understanding of sensor details. The combination of sorting options, detailed sensor information, and data plotting capabilities empowers users to effectively analyze the sensor data within the countries.



-- **Each country's average pollution in the 48 hours prior** --

Country: Gibraltar

Pollutant: NO2
Value: 25.418 ug/m3
Station: Rosia Road

Pollutant: O3
Value: 43.826 ug/m3
Station: Bleak House

Pollutant: PM10
Value: 12.796 ug/m3
Station: Rosia Road

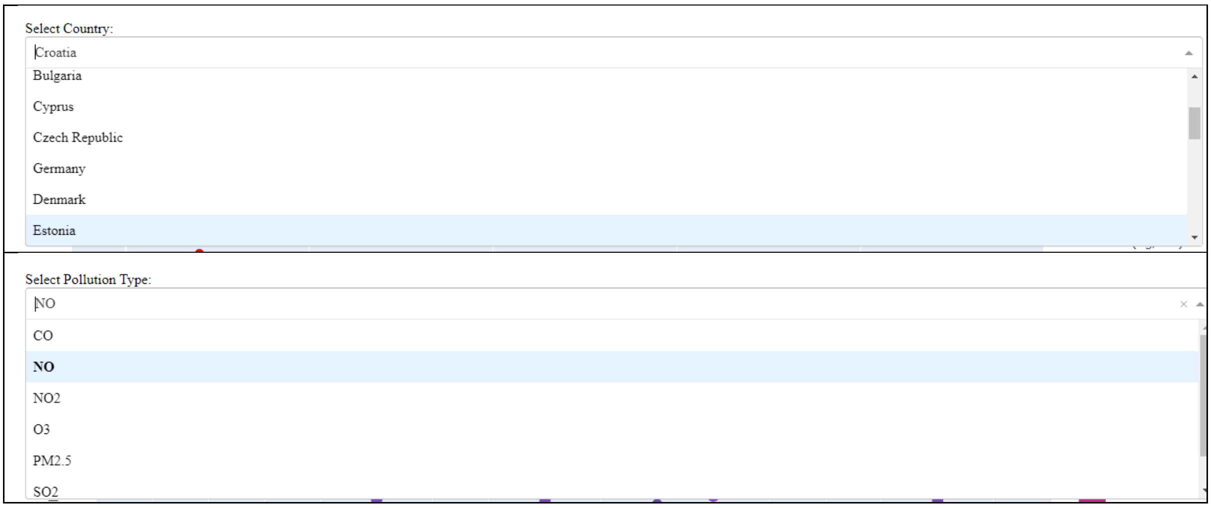
Pollutant: SO2
Value: 1.232 ug/m3
Station: Rosia Road

2.4 Pollution Value Plot

In this section of our interactive Jupyter dashboard, users can refer to the information about their desired sensors obtained from the previous section. They can select a specific sensor ID from a drop-down menu provided.

Once a sensor ID is selected, a plot will be generated to visually represent the observations recorded by the chosen sensor. The plot will display the observations as heatmap dots on the graph, accurately reflecting the data collected during the specified time period.

This feature empowers users to delve deeper into the sensor data by focusing on a specific sensor and examining its observations within a defined timeframe. The interactive nature of the dashboard enhances the user experience, allowing users to explore and analyze the sensor's measurements over time. By visually representing the data on a graph, users can identify patterns, trends, and anomalies, gaining valuable insights into the behavior and performance of the chosen sensor.



Select Country:
Croatia
Bulgaria
Cyprus
Czech Republic
Germany
Denmark
Estonia

Select Pollution Type:
NO
CO
NO
NO2
O3
PM2.5
SO2

3 Limitations

Our dashboard has a primary limitation related to the availability of data. At present, it only utilizes data from a 2-day period.

While the dashboard offers various features and functionalities, such as sensor sorting, detailed sensor information, and data plotting, these capabilities are limited to the available data from the 2-day timeframe. Our development team has ensured that all features are fully functional based on the data within this specific period.

We acknowledge the importance of expanding the dataset in future iterations or updates of the dashboard. By incorporating additional data from different years, users would have the opportunity to select dates beyond the constraints of the 2-day period. This expansion would provide them with a broader range of temporal analysis and insights.

It's essential to understand that the current version of the dashboard serves as a foundation, demonstrating its potential and the value it can offer to users. However, it is crucial to consider the limitations regarding data availability when utilizing and interpreting the features and functionalities provided.

4 Installation Guide

To access and utilize the offline dashboard application on your local host, please follow these step-by-step instructions:

1. Ensure that you have the dataset accessible through the provided link. This dataset contains the necessary data for the proper functioning of the dashboard.

2. Set up a PostgreSQL environment on your PC. Modify the "Data Base.py" code by updating the data directory path to point to the location where the "merged_data.csv" and "mean_data.csv" files are stored. Additionally, set the appropriate password for your PostgreSQL environment. Running this modified code will create the required tables using the data from the Excel files provided.

3. Make sure you have the required libraries installed, including pandas, Flask, request (from Flask), create_engine, and text (from SQLAlchemy).

4. Execute the backend code, and upon successful execution, you should see the following message displayed in the console: "Running on http://127.0.0.1:9012/"

5. Install Anaconda on your system and ensure it is properly set up. Then, run the "Dash.ipynb" file. This will generate a link as the output.

6. Click on the link provided in the output. It will redirect you to a new tab, which serves as the dashboard interface. From there, you can explore and utilize all the features and functionalities of the dashboard.

Make sure that the link is accessible and correctly located in your system to ensure a smooth experience when accessing the dashboard.

5 Test Execution and Reporting

Test Case 1:
- Date of Test Execution: 18.06.23
- Inputs: merged_data.csv
- Expected Output: Users send a request to get the data about the sensors, and the endpoint returns the table of sensor data.
- Actual Output: Users send a request to get the data about the sensors, and the endpoint returns the table of sensor data.
- Test Result: Successful

Test Case 2:
- Date of Test Execution: 18.06.23
- Inputs: Data.csv
- Expected Output: Users request to get the detailed observation data that was observed by the sensors, and the endpoint returns that data.
- Actual Output: Users request to get the detailed observation data that was observed by the sensors, and the endpoint returns that data.
- Test Result: Successful

Test Case 3:
- Date of Test Execution: 18.06.23
- Inputs: data table
- Expected Output: Users send a request to merge the sensors and data tables, and the endpoint returns the merged AllData table.
- Actual Output: Users send a request to merge the sensors and data tables, and the endpoint returns the merged AllData table.
- Test Result: Successful

Test Case 4:
- Date of Test Execution: 18.06.23
- Inputs: "station_name": "Milano v.Senato"
- Expected Output: The endpoint should return the station with the provided name in JSON format, and it should not be empty.
- Actual Output: The endpoint should return the station with the provided name in JSON format, and it should not be empty.
- Test Result: Successful

Test Case 5:
- Date of Test Execution: 18.06.23
- Inputs: "sensor_type": "Benzene"
- Expected Output: The endpoint should return the sensors with the provided type in JSON format, and it should not be empty.
- Actual Output: The endpoint should return the sensors with the provided type in JSON format, and it should not be empty.
- Test Result: Successful

Test Case 6:
- Date of Test Execution: 18.06.23
- Inputs: "country": "Italy"
- Expected Output: The endpoint should return the sensors from the provided comune in JSON format, and it should not be empty.
- Actual Output: The endpoint should return the sensors from the provided comune in JSON format, and it should not be empty.
- Test Result: Successful

6 Bibliography

- Flask Documentation. (2022). Flask: A Microframework for Python. Retrieved from https://flask.palletsprojects.com/
- https://cordis.europa.eu/docs/projects/cnect/3/609023/080/deliverables/001-MYWAYD21RequirementSpecificationAndAnalysiscnectddg1h520142537023.pdf
- Robert L. Lux and C. Arden. Pope. Air quality index. U.S. Environmental Protection Agency, 2009.
- Atlassian's "Release Notes Guide"
- PostgreSQL Documentation. (2022). PostgreSQL: The World's Most Advanced Open-Source Relational Database. Retrieved from https://www.postgresql.org/docs/