

## An interactive application for air quality monitoring

Public authorities collect air quality and weather observations in near-real time from ground sensor stations and store them in digital archives. Generally, ground sensors data are composed of long time series of observations and sensors metadata (including coordinates, type of measured variable, etc.). Often, observations from different sensors and/or providers require different patterns for data accessing, harmonization, and processing. To this end, interactive applications and dashboards, capable of facilitating such tasks, are key for supporting both public authorities as well as ordinary citizens in data processing and visualization. The purpose of this project is to design and develop an interactive client-server application to perform the above tasks.

### Project goals

You should develop an application to support users in accessing, querying, and visualizing air quality and weather sensor data retrieved from existing public digital archives. The system is composed of three main components:

- A **database** where data are ingested beforehand.
- A **web server** (backend) that exposes a REST API used for querying the database and retrieve data.
- A **dashboard** that exploits the web server to provide means for requesting, processing (e.g., space-time aggregations), and visualizing data (e.g., maps, dynamic graphs, etc.).

### Your tasks

- A. Identify and reuse public digital archives of air quality and weather sensor data. The selected dataset must be upload in the application database.
  - Strategies for the continuous data integration between the selected data archive and the database can be implemented and considered a plus but they are not mandatory.
- B. Develop a client-server application that offers the following features:
  - Design and develop a proper database schema to store the selected data
  - Design and develop a REST API to query and retrieve data form the database. Note that the web server may need to perform data cleaning and preprocessing before returning the results to users. Data are returned to users using the JSON format.
  - Develop an interactive dashboard (built using Jupyter Notebooks) that allows for processing and visualizing the data retrieved from the web server using some original manipulation strategy. This may include leveraging both geographic content (map-based views) as well as attributes (interactive graphs). The dashboard should also let users generate custom views of the data. Additional base maps and layer can be added to the map-based views and considered a plus, but they are not mandatory.

## Deliverables and deadlines

As part of the project development, you are required to produce some deliverables:

- **Requirement Analysis and Specification Document (RASD)**  
*Deadline 20th of April 2023*
- **Design Document (DD)**  
*Deadline 25th of May 2023*
- Release your app together with an accompanying **Software Release Document (SRD)** that describes the procedure for installing and running your software as well as any known limitation your software may have.  
*Deadline 18th of June 2023 at 23:59:59CET*

*The deliverables and the source code of the project must be uploaded on the group's GitHub repository.*

## Suggested software tools (example will be presented during the lectures):

- **Database:** [PostgreSQL](#) and [PostGIS](#)
- **Web server:** [Flask](#)
- **Data processing and utilities:** [Pandas](#), [GeoPandas](#), [Dask](#), [Xarray](#), [Rasterio](#)
- **Dashboard:** [Jupyter Notebook](#), [Jupyter Widgets](#), [ITables](#), [Mercury](#), [Matplotlib](#), [Plotly](#), [Bokeh](#), [IpyLeaflet](#), [Folium](#), [geemap](#)

## Suggested datasets

Air quality and weather sensor data

- Lombardy region: <https://tinyurl.com/yutn8rcp>
- Europe: <https://discomap.eea.europa.eu/map/fme/AirQualityUTDExport.htm>
- Global: <https://aqicn.org> (check air quality stations in the different countries to find the local data provider)

Example of background geodata that you can include in your application:

Lombardy region: <https://www.geoportale.regione.lombardia.it/download-ricerca>

Global coverage: <https://github.com/samapriya/awesome-gee-community-datasets>

## Groups

The project should be developed in groups of around 4 persons. It is important to check whether the schedule of all team members is aligned. Please allow for at least two or three hours of joint work per week.

As a first step, register on GitHub (all members of the group) and create a repository (only one member of the group). Invite in the repository (under Settings / Collaborators) all the members of the group and user *gioenn*.

If you have already created a group and the GitHub repository, please fill this form:

<https://forms.gle/3LQH7uTrJEzxQ7k6>.

If you are looking for partners, please write a post on the WeBeep forum.