Thomas Dias Costa

202400383

**Dashboard Final Project**

### **Introduction :**

My project consists of the creation of an interactive and dynamic dashboard in order to be able to find out about the weather and air quality conditions of a given city. This site aims to provide users with real-time updated information on temperature, humidity, air quality, and other environmental parameters. For this, another objective is to offer different users a pleasant, intuitive and ergonomic graphical interface. This dashboard is also designed to display graphs that evolve over time in order to better understand the data and visualize it better. It aims to be a convenient tool for users interested in local weather conditions, while providing a pleasant and smooth user experience.

### **Tools Used :**

To develop this meteorological dashboard, several technologies were used to ensure performance, interactivity and aesthetics of the site. HTML and CSS (with the use of the Bootstrap Framework) made it possible to structure and style the interface in a responsive manner, thus guaranteeing an optimal user experience on different devices and screen sizes. These two tools really represent the skeleton of the project with on one side HTML allowing all the structure and primary visual aspect of the site and the other CSS with the help of Bootstrap acting on the entire design part allowing the final visual result.

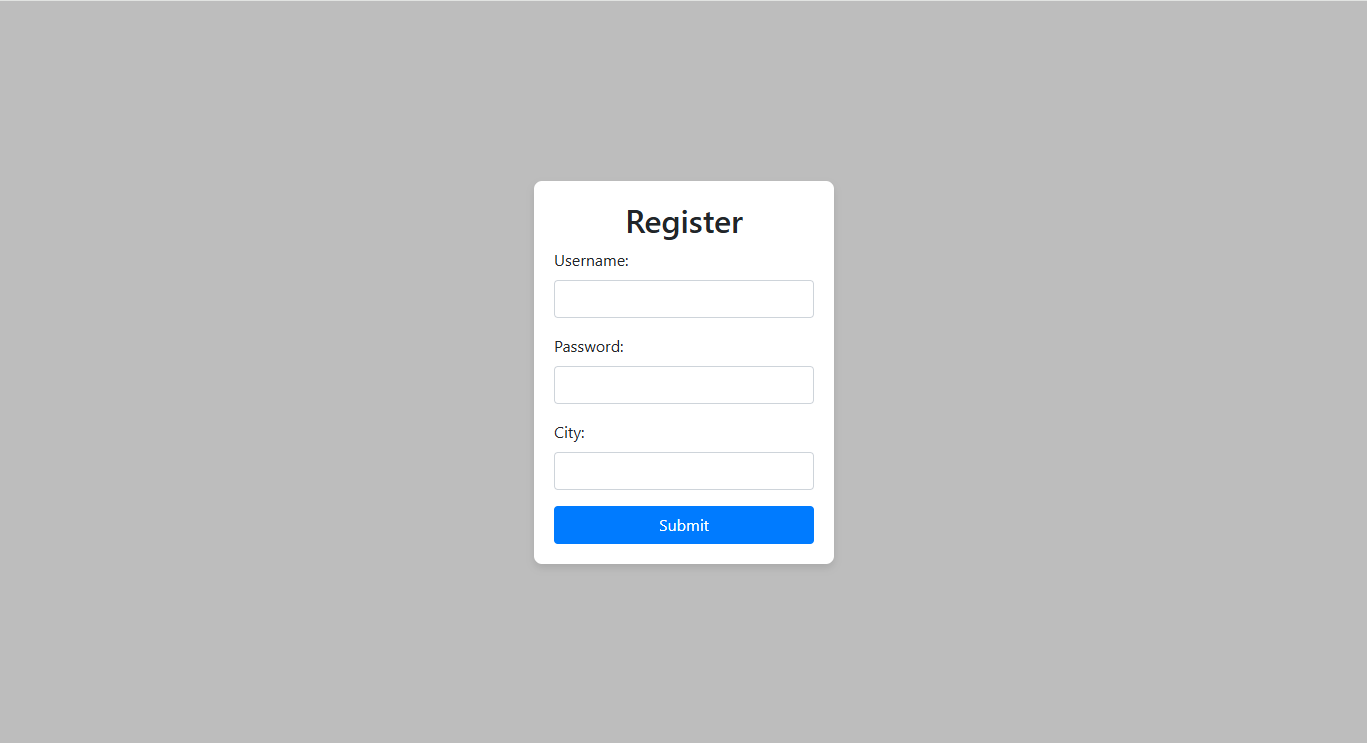
With the aim of making a dynamic visual and real-time display of the different meteorological data retrieved from the APIs, I use the JavaScript language which will allow a real link between my HTML skeleton and OpenWeatherMap. As said previously, I use OpenWeatherMap as a meteorological API, which will allow me to retrieve different data from several cities around the world and be able to use them.

Finally, the weather data is exchanged in JSON format, making it easier to integrate and manipulate in the dashboard.

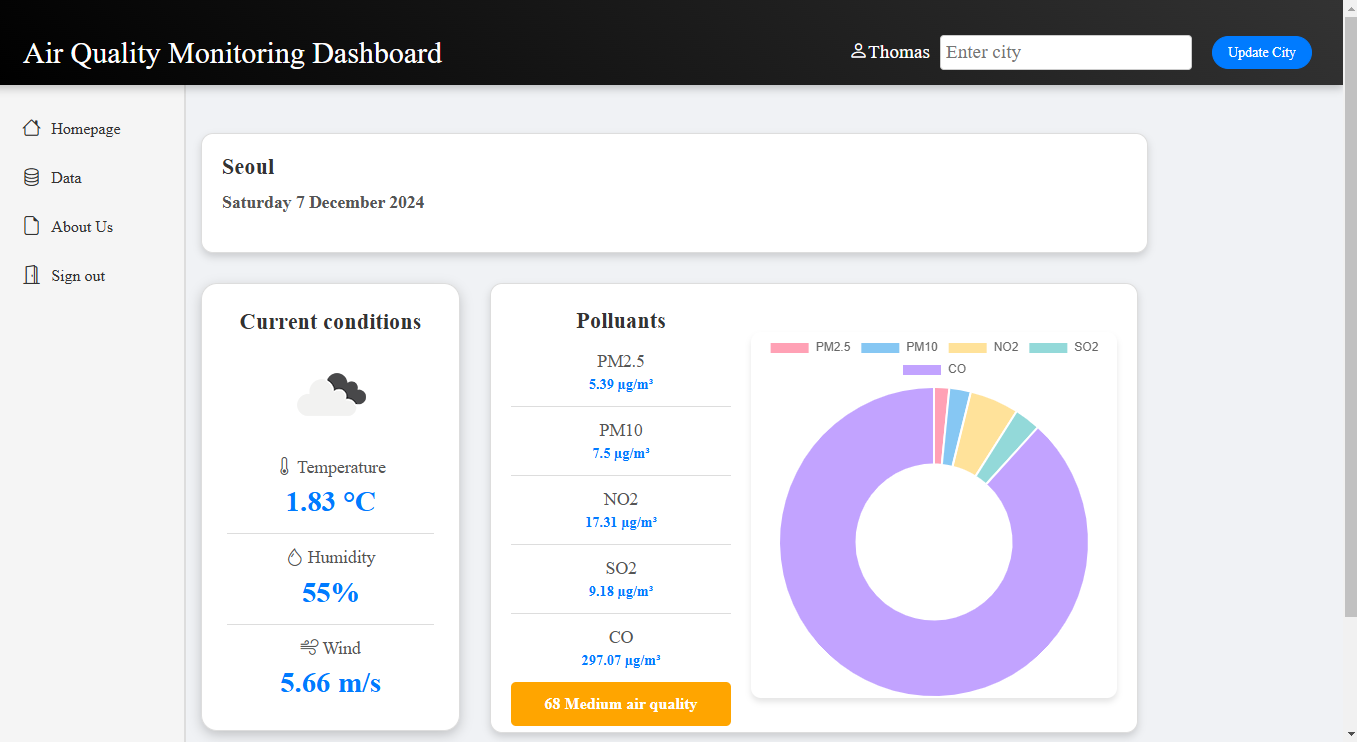
### **General presentation:**

The first page of my weather dashboard is dedicated to user registration. It allows each user to create their personal profile by giving a username, password and city. This information is then used to personalize the user experience, including displaying weather data for the returned city during check-in. The objective is to make the site more interactive and specific to each person to best meet user needs.



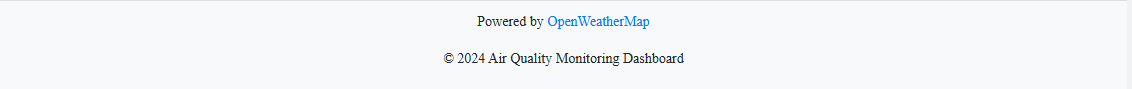
However, I couldn't complete this customization and registration page. Indeed, I have not managed to set up a database system for storing user profiles. Indeed, I was unable to functionally implement a solution to save this information and allow users to log in to their saved profiles.

When the user has finished entering his various personal information and then clicks the “submit” button, he arrives on the main page (“HomePage”) on which you can find a lot of information.

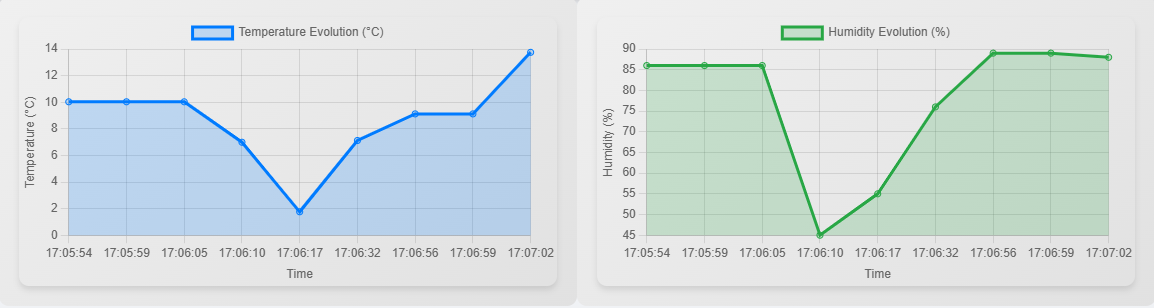


To begin, let's look at the overall structure of the page. At the top of the page is the Header where you can see the name of the dashboard (where when you click on it it sends you to the Home Page), the name of the user retrieved during registration as well as a “text area” inviting the user to search for a city and its validation button. On the left side we can see a navigation bar where we can see all the pages of the site (Home Page, Data Page, About-us Page and “Sign out” allowing the user to log out). We can now focus on the main content of the page. The latter is made up of 3 “cards”. The first simply allows you to display the city that the user will enter initially during the registration phase and will then change in correlation with the choices that the user will make when entering the name of a new city. in the text area above. The second card is the “Current condition card” displaying basic weather information. In fact, we can see an icon of the current weather, temperature, humidity and wind speed of the city in question displayed in order. All this information is directly retrieved from the weather API. The third map, “the pollution card”, displays the main air pollutants as well as a doghnut chart to better understand the distribution of these pollutants. And finally, the last element of this map is the air quality score. The latter is calculated by considering the information on the various pollutants and is divided into 3 classifications: good, average and bad. If the score is greater than 100 it will be categorized as poor and therefore the rectangle where the score is displayed will be red, if the score is between 100 and 50 then it will be considered average and orange in color and otherwise it will be classified as good and will be associated with green.

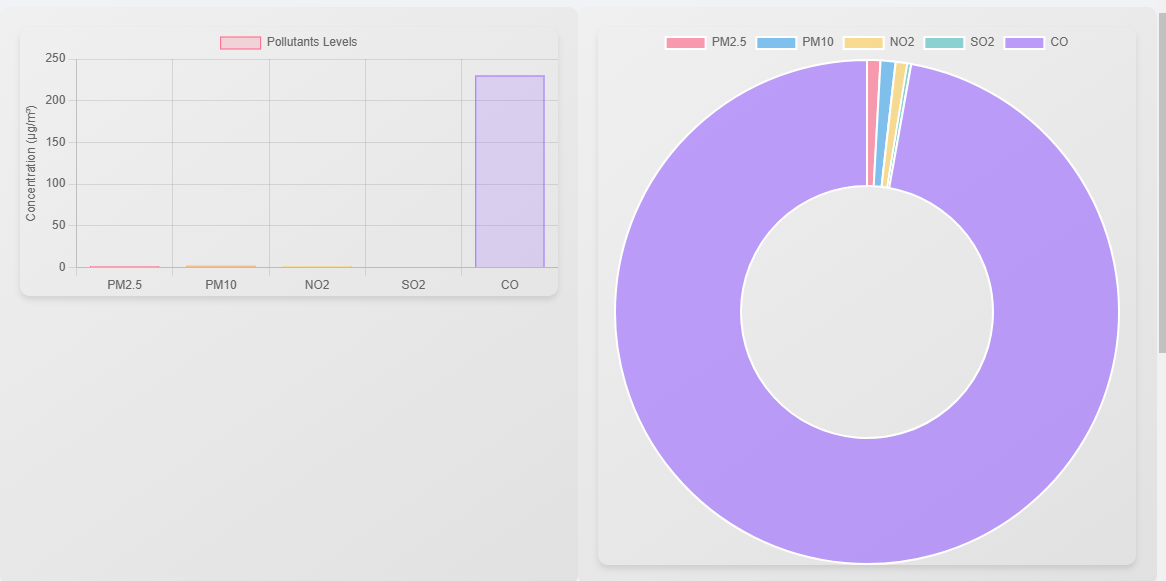
At the bottom of this page, we can find a footer containing the name of the website as well as a link to the open weather map API, a footer which will also be on all pages of the site except the account registration page.



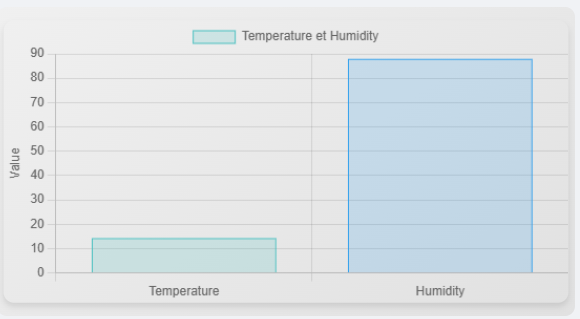
To have more precision on the data displayed, the user can go to the data page on which they can see all the graphs. The first two graphs represent real-time changes for one of humidity and the other of temperature as a function of time. Indeed, every 60s a new point will be generated displaying the current conditions. There is just one small problem with these graphics that I haven't been able to fix. When the user decides to change city the previous city data will still be displayed. In return, this allows data to be compared between different cities.



The next two graphs represent the distribution of the different pollutants with a bar chart on one side and the doghnut chart on the other that we could see on the home page.

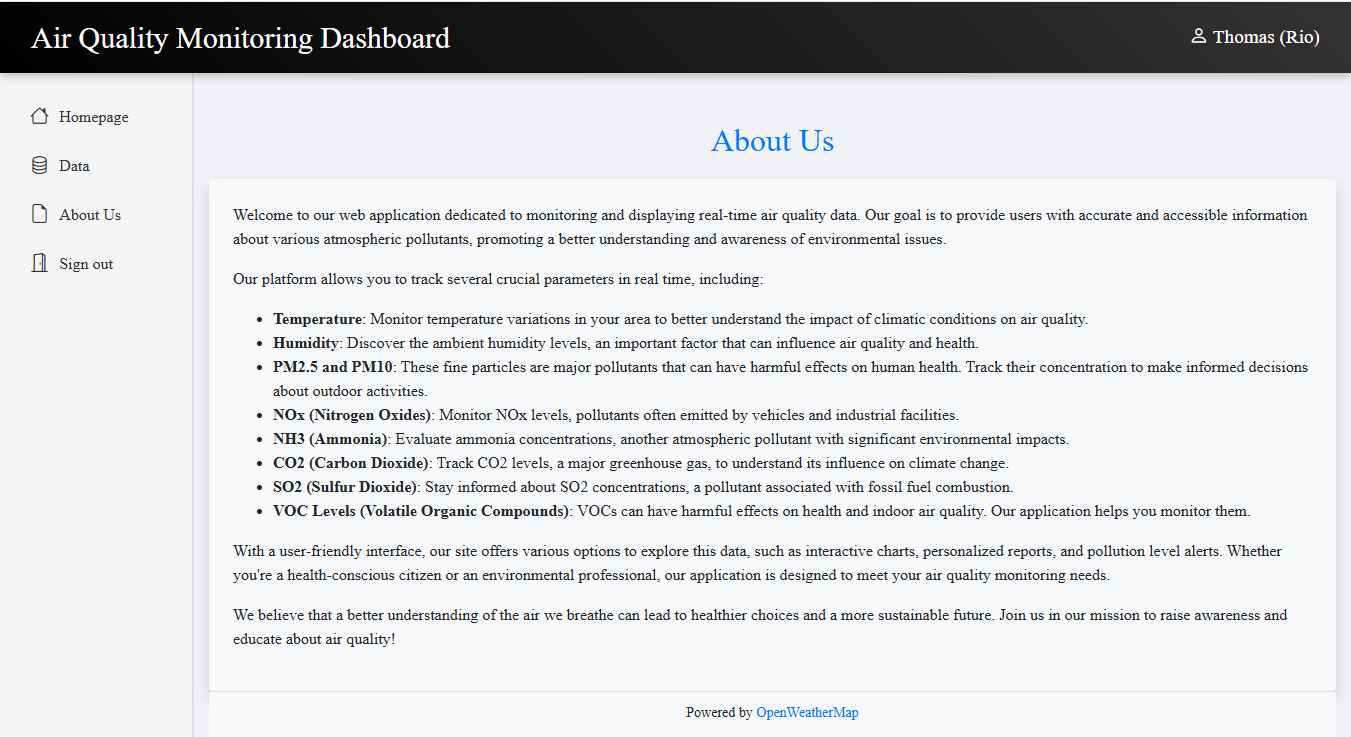


The last graph displayed is a bar chart representing the temperature and humidity levels allowing them to be compared.



The only disappointment with this page is that I was not able to display a graph showing the temperature history over the last few days because of the withdrawal conflicts with another graph. The problem came from I think the structure of how I retrieved the data from the API, and I was therefore unable to display it correctly so I decided to remove it.

The last page of this dashboard is an about us page roughly summarizing the main issues of the site as well as its objectives.



### **Conclusion**

Overall, this weather and air quality dashboard project was a rewarding learning experience, marked by successes but also some technical failures. Indeed, the dashboard's graphical interface was a success, thanks to the use of technologies such as HTML, CSS and JavaScript but also the implementation of the OpenWeatherMap API which allowed me to display real-time data. Real-time graphs, such as changes in temperature and humidity, as well as visualizations of pollutant distribution, were an important addition to the site's interactivity. However, some features that I would have liked to implement were not done as I wanted. Like the database system for storing user profiles, updating graphs in real time when the user changes city or even the history graph.

In summary, although technical challenges were encountered, the project's successes remained significant and resulted in the creation of a functional, visually appealing and informative dashboard. This project represents a solid basis for future improvements, particularly user data management and graphics optimization.

Demo Link: https://thomasdcosta.github.io/Web/