PROJECT

**AIR QUALITY FORECAST**

A webpage that allows users to search for a city and get its latest Air Quality Index (AQI), as well as concentration of six (6) pollutants: Carbon Monoxide, Nitrogen Dioxide, Ozone, Sulphur Dioxide, Fine Particles Matter, and Course Particulate Matter. This uses three (3) APIs: **IP Geolocation** of IP-API, **Air Pollution API** of OpenWeatherMap, and **Geocoding API** of OpenWeatherMap. It runs on **localhost**.

**Project Authors**

Moreno, Kim Brian

Marjalino, Charles

Oliquino, Mathew

Deladia, John Andre

Simbahon, Meleth

**Prerequisites**

* Web Browser (Google Chrome, Microsoft Edge, Brave Browser, or any browser)
* Internet Connection to allow API calls

**Getting Started**

Simply locate the index.html file inside the src folder of project directory. Open it with any of your computer browser and it is ready to go.

On page load, there will be two cards. The first (or upper) card contains the title of the project and a search field where you can search a city. The second (or lower) card contains the **Air Quality Index** **(AQI)** and **Pollutants Concentration**. The webpage will gather your location (latitude and longitude) using your IP Address and use this information to find the AQI and pollutants concentration.

Need to know the air quality of other cities? Use the dedicated search bar! Just search the name of the city you want to look and wait to the second (or lower) card to update its information.

**Website Structure**

[Main Directory]

|-**src/**

| └-index.html

| └-style.css

| └-script.js

|

└-**res/**

└-.jpg files

src folder contains essential code files.

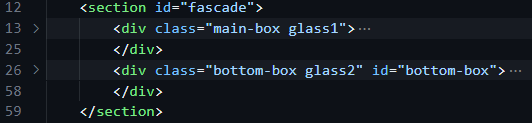
index.html - is the only html file of the website, which contains all the content.

style.css - contains all the CSS styles.

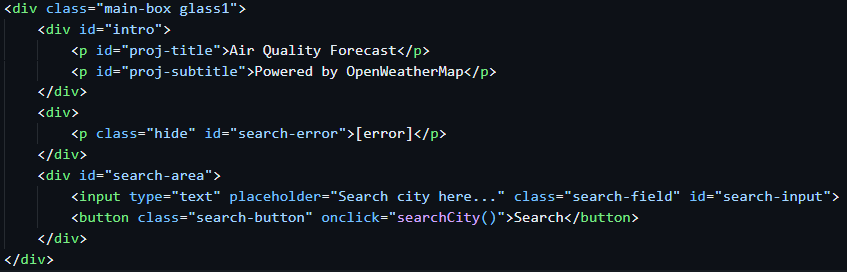
script.js – for handling API calls and other background functions.

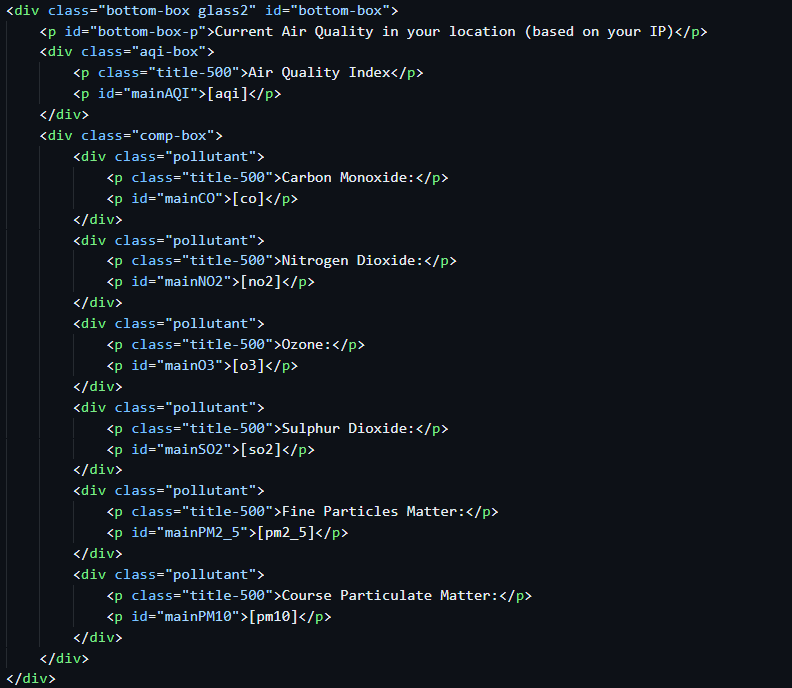
res folder holds the images for the website.

**Code Explanation**

**** **HTML**

The whole content of this project is contained in <section> tag to allow an easy management of elements.

**** Inside the <section> tag is two <div> tags that have two different purpose.

**** The main-box container contains paragraph tags for displaying the title of the project, input tag for accepting user input, and button tag for calling necessary JavaScript functions.

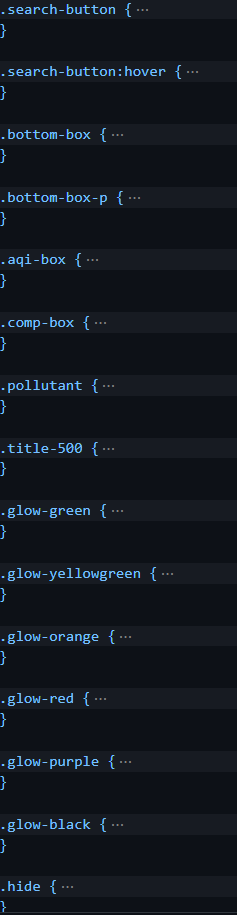
Finally, the bottom-box container contains multiple <div> tags. Each <div> tag contains element used to display their own assigned information.

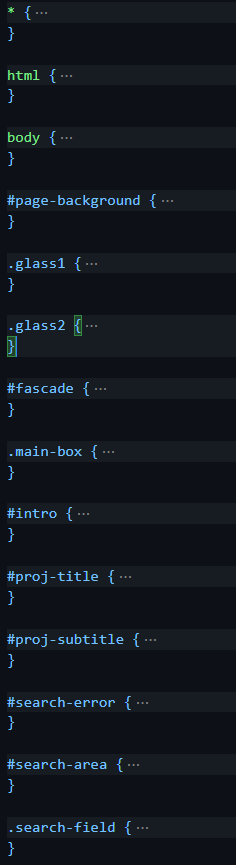
For example, the <div class = “aqi-box”> is used to display the AQI. In the other hand, <div class = “comp-box”> is used to contain inner <div> tags for each pollutant (Carbon Monoxide, Nitrogen Dioxide, Ozone, Sulphur Dioxide, Fine Particles Matter, and Course Particulate Matter.)

**CSS**

@import url('https://fonts.googleapis.com/css2?family=Nunito:ital,wght@0,200..1000;1,200..1000&family=Open+Sans:ital,wght@0,300..800;1,300..800&display=swap');

We imported fonts from **Google Font API** to enhance the visual of the project.





Elements are assigned with its own class and unique ids to control how it is display on the screen.

For example:

.glass {

background-color: rgba(200, 255, 255, 0.2);

box-shadow: 8px 12px 18px rgba(0, 0, 60, 0.4);

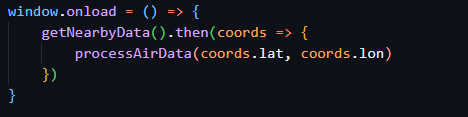
backdrop-filter: blur(4px);

}

In this code snippet, the element with a class name ‘.glass’ have its background-color, box-shadow, and backdrop-filter property customized. The goal of this customization is to achieve a glass-like design which is used on the first card.

**JAVASCRIPT**

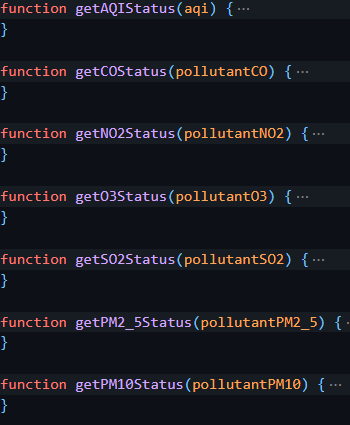
The script does all the API callings that are essential for this project.



In this code snippet, when the page is fully loaded, a function is called to get the latitude and longitude of the current location of the user, based on their IP address. Once gathered, another function is called to handle the Air Pollution API call.

 The function processAirData(lat, lon) calls the Air Pollution API and …

… assigned every data from the Json object to its designated element.

These functions are containing logics that checks the micrograms per cubic meter (μg/m3) for each pollutant or the index of an AQI. It returns a string value that indicates if the pollutant or the AQI is in one of the following states:

Good

Fair

Moderate

Poor

Very Poor

(The range of values that are used for the logic came from the OpenWeather scale for Air Quality Index levels [1])

**KNOWN ISSUES**

Aside from the internet connection dependency of this webpage, there is no known issues.

**REFERENCES**

**[1]** <https://openweathermap.org/air-pollution-index-levels>