



UNIVERSITÀ DI PISA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

CORSO DI LAUREA TRIENNALE IN INGEGNERIA INFORMATICA

*Integrazione del servizio OpenStreetMap all'interno del portale  
MonIQA*

I RELATORI

- I. Prof. Giuseppe Anastasi*
- II. Ing. Carlo Vallati*
- III. Ing. Francesca Righetti*

IL CANDIDATO

*Giacomo Pellicci*

Anno Accademico 2017-2018

## Indice

1 Introduzione.....	3
1.1 Principali agenti inquinanti.....	3
1.2 Il progetto.....	5
2 Sezione Mappa.....	6
2.1 Introduzione a Leaflet .....	6
2.2 Il server e le strutture dati.....	7
2.3 Composizione della mappa .....	7
3 Sezione Copertura IQA.....	12
3.1 Che cosa è una heatmap .....	12
3.2 Implementazione.....	12
4 Sezione Regioni .....	14
4.1 Come si suddivide la mappa in regioni.....	14
5 Sezione Ricerca.....	16
5.1 Contenuto della sezione.....	16
5.2 Risultati della ricerca .....	17
5.3 Implementazione della funzione di ricerca .....	20
Conclusioni .....	22
Bibliografia .....	23

## 1 Introduzione

L'inquinamento dell'aria, nonostante i molteplici passi in avanti fatti per la riduzione delle emissioni di sostanze inquinanti nel corso degli anni, continua ad avere un impatto significativo sulla vita delle persone, in particolare per quanto concerne le aree urbane. Influisce inoltre in maniera considerevole sull'economia, sugli indici demografici di mortalità e sull'incremento dei costi sanitari. Può danneggiare la fauna e la flora, caratteristiche dei diversi ecosistemi, e apportare serie modificazioni alla qualità dell'acqua e del terreno.

### 1.1 Principali agenti inquinanti

Le principali forme di polveri sottili sono PM<sub>10</sub> e PM<sub>2.5</sub>, dove il numero posto dopo la sigla PM (Particulate Matter) indica il diametro massimo in  $\mu\text{m}$  che la particella deve avere per rientrare in tale categoria. Questo tipo di inquinante si genera sia da fonti naturali che antropogeniche. Le prime includono il sale marino, la polvere naturalmente sospesa nell'aria, il polline e le ceneri vulcaniche, mentre le altre sono composte dal combustibile per la produzione di energia, il riscaldamento domestico, i trasporti, l'industria, incenerimento dei rifiuti e agricoltura, oltre al consumo di sistemi molto diffusi come freni e pneumatici dei veicoli. Queste polveri, in particolare le PM<sub>2.5</sub>, a causa della loro dimensione ridotta, vengono filtrate con difficoltà dal naso e riescono quindi ad annidarsi negli alveoli polmonari causando diverse patologie tra cui l'asma. Stime dell'effetto sulla salute, attribuibili all'inquinamento indicano che le concentrazioni di PM<sub>2.5</sub> nel 2014[1] sono responsabili di circa 428 000 morti premature, dovute alla lunga esposizione a questo inquinante in Europa.

L'inquinante più pericoloso per il rischio degli ecosistemi è il cosiddetto ozono a bassa quota ( $\text{O}_3$ ). Esso non è direttamente immesso nell'atmosfera; si forma invece attraverso la reazione dell'ossigeno ( $\text{O}_2$ ) e il biossido di azoto ( $\text{NO}_2$ ), in presenza di forte irraggiamento solare ed elevate temperature. Ne segue che le concentrazioni di ozono sono nettamente più elevate nelle ore pomeridiane dei mesi estivi, anche se variano sensibilmente in funzione delle condizioni meteorologiche. Alti livelli di  $\text{O}_3$  danneggiano i raccolti agricoli, le foreste e le piante, diminuendone il tasso di crescita. L'obiettivo imposto dall'Unione Europea (EU) per la protezione della vegetazione da  $\text{O}_3$  a bassa quota è stato superato circa nel 18% delle aree agricole dei 28 paesi appartenenti a EU-28 nel 2014 [1]. L'obiettivo a lungo termine per la salvaguardia della vegetazione da  $\text{O}_3$  è stato superato nell'86% delle aree agricole di questi Paesi dell'Unione Europea.

Pur essendo presenti in atmosfera diverse specie di ossidi di azoto, per quanto riguarda l'inquinamento dell'aria si fa quasi esclusivamente riferimento al termine  $\text{NO}_x$  che sta ad indicare la somma pesata del monossido di azoto (NO) e del biossido di azoto ( $\text{NO}_2$ ).

L'ossido di azoto (NO) è un gas incolore, insapore ed inodore ed è prodotto soprattutto nel corso dei processi di combustione ad alta temperatura, assieme al biossido di azoto (NO<sub>2</sub>). Viene poi ossidato in atmosfera dall'ossigeno, e più rapidamente dall'ozono, producendo biossido di azoto. La tossicità del monossido di azoto è limitata, al contrario di quella del biossido di azoto che risulta invece notevole. Il biossido di azoto (NO<sub>2</sub>) è un gas tossico con grande potere irritante, inoltre è un energico ossidante, molto reattivo e quindi altamente corrosivo. Il ben noto colore giallognolo delle foschie che ricoprono le città ad elevato traffico è dovuto, per l'appunto, al biossido di azoto. Quest'ultimo svolge un ruolo fondamentale nella formazione dello smog fotochimico, in quanto costituisce l'intermedio di base per la produzione di tutta una serie di inquinanti secondari molto pericolosi come l'ozono.

Si stima che gli ossidi di azoto contribuiscano per il 30% alla formazione delle piogge acide [1], mentre il restante è imputabile al biossido di zolfo (SO<sub>2</sub>) e ad altri inquinanti.

L'acidificazione del suolo, dei laghi e dei fiumi è da attribuirsi al biossido di zolfo (SO<sub>2</sub>), capace di ridurre la biodiversità e provocare la morte di piante e animali. La principale fonte di inquinamento è costituita dalla combustione di combustibili fossili, quali carbone e derivati del petrolio, in cui lo zolfo è presente come impurità. Vengono inoltre emesse nell'atmosfera durante le eruzioni vulcaniche. Dopo decenni di diminuzioni delle emissioni di questo inquinante in Europa, il processo di acidificazione sta rallentando e alcune foreste e laghi stanno mostrando segni di recupero. Tuttavia, si stima che il 7% dell'ecosistema europeo fosse a rischio acidificazione nel 2014 [1].

Il monossido di carbonio (CO) proviene in gran parte dai gas di scarico dei veicoli, soprattutto funzionanti a bassi regimi, come nelle situazioni di traffico intenso e rallentato. Per questo motivo è molto presente nelle aree urbane. Altre sorgenti sono gli impianti di riscaldamento e alcune tipologie di industrie, come le raffinerie e gli altiforni. Le conseguenze sull'ambiente sono da considerarsi trascurabili, mentre sull'uomo ha effetti particolarmente pericolosi, in quanto il monossido di carbonio forma un composto fisiologicamente inattivo con l'emoglobina del sangue, che impedisce l'ossigenazione dei tessuti.

Il benzene (C<sub>6</sub>H<sub>6</sub>) è un inquinante facilmente assorbito per inalazione e dannoso per la salute umana. La sorgente più rilevante è rappresentata dal traffico veicolare, principalmente dai gas di scarico dei veicoli alimentati a benzina, nei quali viene aggiunto al carburante come antidetonante. Il benzene è stato classificato dallo IARC (Agenzia Internazionale per la Ricerca sul Cancro) in classe 1 come cancerogeno certo per l'uomo. È infatti accertata la capacità di causare leucemie acute e croniche, con rischio proporzionale alla dose cumulativa.

## 1.2 Il progetto

Gli inquinanti precedentemente citati sono tra i più pericolosi per la salute umana e il benessere degli ecosistemi, motivo per il quale ho ritenuto opportuno accogliere la proposta di un progetto volto ad ampliare il servizio offerto dal portale MonIQA<sup>1</sup>, offerto dal Dipartimento di Ingegneria dell'Informazione dell'Università di Pisa. Questo servizio si occupa di raccogliere i dati offerti dalle ARPA<sup>2</sup> delle varie regioni italiane e di rappresentarli su una mappa geografica, agevolando così l'interpretazione dei dati, anche grazie ad una scala di colori, facendo comunque in modo di non far mancare informazioni chiave ai fruitori più interessati. Noto è che la maggior parte degli utenti non è esperta per quanto concerne l'inquinamento dell'aria, motivo per cui una semplice rappresentazione dei dati in forma tabellare, abbinata a dei grafici con funzione di descrivere l'andamento giornaliero dell'inquinamento, risulta di immediata comprensione per qualunque target di utenza.

Il portale si propone di illustrare al pubblico che cosa sia un Indice di qualità dell'aria e il metodo con il quale si valuta la bontà di quest'ultimo. All'interno del sito è possibile consultare il D. Lgs. 155/2010 che tratta appunto il tema della qualità dell'aria in Italia, ed impone dei limiti ai valori degli inquinanti rilevati. Inoltre, si può trovare una spiegazione sintetica dell'origine e delle cause che hanno i vari inquinanti analizzati all'interno del portale, che corrispondo a quelli precedentemente elencati. Sono presenti ulteriori sezioni, in cui è possibile visualizzare un giudizio espresso sulla qualità dell'aria a livello regionale, verificare la copertura delle *stazioni di rilevamento* (che chiameremo semplicemente stazioni) sul territorio nazionale ed infine una funzione di ricerca, capace di mostrare le misurazioni pregresse di particolari stazioni.

---

<sup>1</sup> MonIQA è l'acronimo di Monitoraggio dell'Indice di Qualità dell'Aria.

<sup>2</sup> ARPA, anche nota come Agenzia Regionale per la Protezione Ambientale

## 2 Sezione Mappa

La precedente versione del portale MonIQA si basava sul servizio di cartografia offerto da Google Maps ma, a fronte della necessità di aggiungere ulteriori funzioni ed una maggiore personalizzazione, così come la comunicazione da parte di Google dell'entrata in vigore di un nuovo piano tariffario per i servizi di Google Maps che rendono a pagamento l'utilizzo delle API oltre un certo numero di chiamate mensili, si è ritenuto opportuno convertire il portale al servizio OpenStreetMap, che come suggerisce il nome è di natura open, cioè gratuito. OpenStreetMap tramite MapBox permette la possibilità di scegliere tra diverse tipologie di mappa, tra cui satellitare e stradale. La libreria javascript scelta per implementare nel portale il servizio OpenStreetMap è Leaflet. Anche quest'ultima è completamente open source e vanta un'ottima documentazione delle API [2], oltre che ad una community molto attiva per quanto riguarda il rilascio di plugin.

### 2.1 Introduzione a Leaflet

Leaflet<sup>3</sup> è una libreria javascript *open source* che permette lo sviluppo di mappe geografiche interattive (WebGIS<sup>4</sup>), le caratteristiche più importanti sono la possibilità di mostrare punti di interesse con dei marker, linee o figure geometriche e strutture dati come file GeoJSON. Leaflet può vantare un peso ridotto ed una facilità di uso che si adatta perfettamente alle necessità di questo compito. Per questo motivo l'ho preferita alla libreria OpenLayers che, pur offrendo alcune funzionalità extra tuttavia non necessarie, è più pesante. Leaflet viene fornito sotto forma di due file, la libreria javascript *leaflet.js* contenente tutti i metodi messi a disposizione del programmatore ed il file CSS che *leaflet.css* in cui è possibile trovare ed eventualmente personalizzare lo stile di tutti gli elementi che compongono la mappa. La creazione di una mappa è molto semplice: si tratta di creare una mappa, impostandone la centratura e il livello di zoom, assegnandola ad un oggetto del DOM, quale un DIV. Successivamente è necessario creare un layer con l'effettiva tipologia di carta da visualizzare, nell'esempio una mappa stradale, ottenuta tramite MapBox. Possiamo quindi aggiungere un marker con il relativo popup in una determinata posizione per poi aprirlo.

```
var myMap = L.map('map-div').setView([43.721, 10.389], 17);

L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?access_token={myToken}', {
  id: 'mapbox.streets',
  accessToken: 'myToken'
}).addTo(myMap);

L.marker([43.721, 10.389], 17).addTo(myMap).bindPopup('I am a popup')
.openPopup();
```

<sup>3</sup> È possibile reperire la libreria assieme ai suoi plugin presso <https://leafletjs.com>

<sup>4</sup> Sono detti WebGIS i Sistemi Informativi Geografici (GIS) pubblicati sul web.

L'argomento della funzione `bindPopup()` può essere una qualsiasi stringa, in particolare quelle contenenti tag HTML verranno interpretate dal browser come tali, permettendo quindi la creazione di popup di qualunque tipo. Questo sarà molto conveniente per creare dei popup ricchi di informazioni raccolte dinamicamente tra quelle disponibile nel server, ma allo stesso tempo con uno stile semplice ed immediato per l'utente e la possibilità di personalizzarne l'aspetto tramite CSS.

## 2.2 Il server e le strutture dati

Il server su cui è attivo il portale ospita anche un database relazionale MySQL, nel quale sono conservate informazioni rilevanti per l'applicazione. Queste comprendono lo storico delle misurazioni effettuate, stazioni di rilevamento, tipologia di inquinanti e limiti di riferimento, regioni e città italiane. La parte più sostanziosa dei record del database è composta da tutte le misurazioni effettuate dalle centrali da quando il servizio è attivo, cioè dal Dicembre 2014. Questo è il motivo per il quale, anche in ottica futura, pensando all'aumento del numero di record, l'applicazione non richiede mai di eseguire una query per ottenere le misurazioni più recenti, al fine di illustrarle all'utente, invece per più volte nell'arco di una giornata si occupa di eseguire la stessa query e di memorizzare i risultati in un file di cache. Quest'ultimo non conterrà sicuramente informazioni aggiornate al minuto, ma contribuirà notevolmente a ridurre il carico del server che, al momento della scrittura di questo testo, contiene già diversi milioni di record solamente nella tabella delle misurazioni. La query in questione è la seguente:

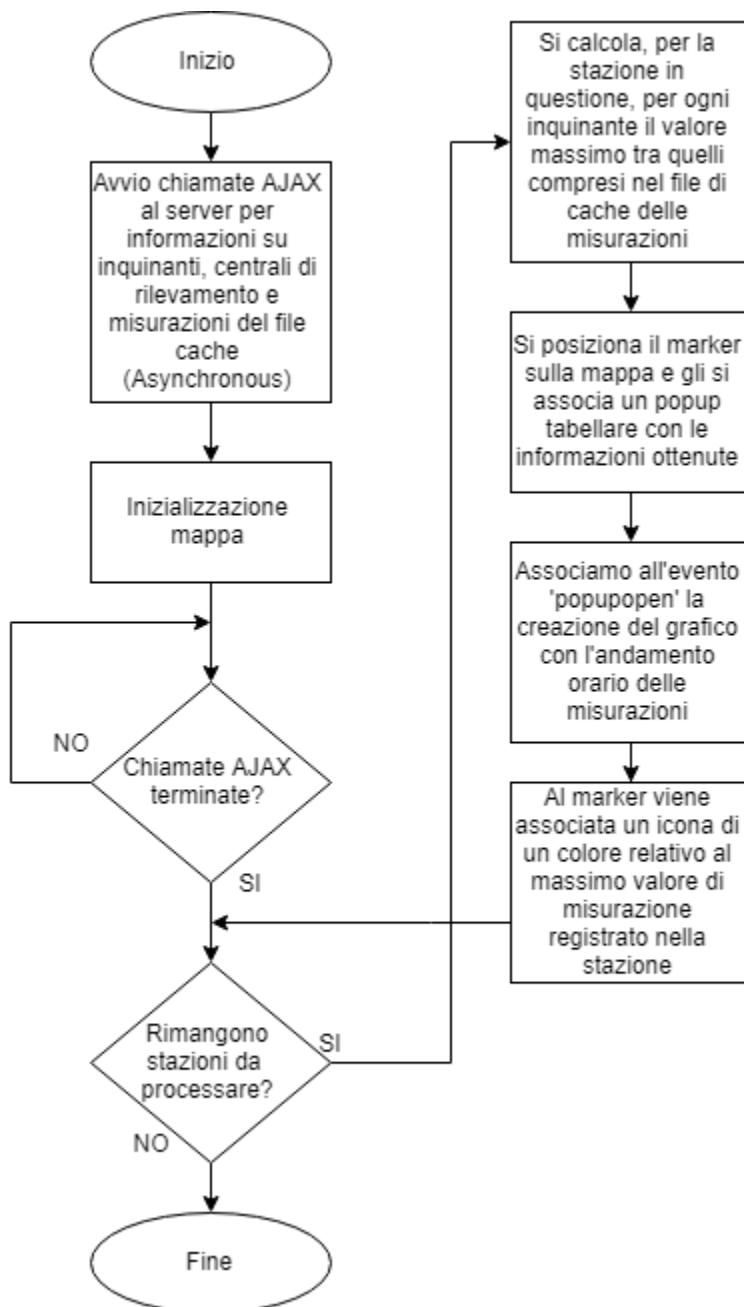
```
SELECT devices.name, T.fk_device, measure, fk_sensortype, maxdate AS date
FROM measureview_minimal JOIN (SELECT fk_device, MAX( DATE ) AS maxdate
                                FROM measureview_minimal GROUP BY fk_device ) T
ON
                                T.fk_device = measureview_minimal.fk_device
JOIN devices ON
                                T.fk_device = devices.pk_device
WHERE date > maxdate - 86400000
GROUP BY fk_device, fk_sensortype, measure
ORDER BY T.fk_device ASC, fk_sensortype ASC, date DESC;
```

Tutte le altre informazioni quali coordinate delle stazioni, città e regioni di appartenenza, tipologia di sensori e tipi di inquinanti con i relativi limiti di legge vengono ottenute con una normalissima query SQL eseguita all'occorrenza, considerando comunque la modesta dimensione di queste strutture dati ed il basso impatto di esse sulle prestazioni del sistema.

## 2.3 Composizione della mappa

Il server raccoglie periodicamente i valori delle misurazioni delle varie stazioni sul territorio nazionale tramite uno script, conservandoli poi all'interno di un database MySQL.

Considerando che il processo di prelievo dati dal server, composto dall'invio della query e ricezione del dataset risultante ha una durata relativamente lunga, tutte queste operazioni sono di tipo asincrono, cioè all'avvio ritornano il controllo del programma e al loro completamento generano un event da sfruttare per proseguire con le operazioni. Questo tempo di attesa può comunque essere utilizzato per svolgere elaborazioni lato client, motivo per cui è la prima operazione ad essere svolta all'avvio della procedura. Di seguito è riportato il diagramma di flusso che descrive sinteticamente l'intero processo, evidenziandone i passaggi chiave. Questa procedura all'interno del codice sorgente è svolta dal metodo `dataProcess()`:





Si avvia l'operazione di recupero dati dal server, in particolare si vuole ottenere la lista degli inquinanti ed i relativi limiti di riferimento, nomi e coordinate delle stazioni di rilevamento ed infine i dati sulle misurazioni più recenti che sono contenuti nel file di cache precedentemente presentato.

Subito dopo l'avvio di questa procedura, che è di tipo asincrono, si inizializza la mappa, in modo simile a quanto mostrato nell'introduzione. Al termine del trasferimento dei dati dal server al client, si inizia il processo di elaborazione dati vero e proprio.

Questo è il punto più critico dell'intero algoritmo. È quindi necessario strutturare nel modo giusto il ciclo affinché sia efficiente. Visto che i dataset restituiti dal server sono ordinati, per data o per nome, è possibile rendere questo processo più efficiente, annidando i cicli come mostrato:

```
for(var i=0; i<array_Stazioni.length; i++){
  var iteration = 0;
  for(var j=iteration; j<array_Misurazioni.length; j++){
    if(array_Misurazioni[j].id_stazione == array_Stazioni[i].id_stazione){
      //operazioni sui dati
      iteration ++;
    }
    else{
      if(iteration!=0)
        break;
    }
  }
}
```

Le stazioni sono ordinate per id\_stazione così come le misurazioni. Si procede quindi analizzando prima una stazione e le sue varie misurazioni ed appena le misurazioni non sono più relative a quella stazione si esce dal ciclo con `break;`, rientrando poi al ciclo successivo con un indice, nel codice chiamato `iteration`, che sicuramente mostrerà la prima misurazione appartenente alla prossima stazione.

Per ogni stazione e per ciascun inquinante rilevato da essa, sarà necessario calcolare il valore massimo fra le varie misurazioni di una stessa sostanza. Questo valore sarà quello mostrato all'interno del popup di fianco all'inquinante, mentre il suo rapporto percentuale rispetto al limite di riferimento imposto dalla legge verrà detto indice. Inoltre, il massimo fra gli indici di una stessa stazione denoterà il colore del popup, dando una prima impressione all'utente riguardo alla qualità dell'aria nei dintorni della stazione. Si posiziona quindi il marker sulla mappa, in base alle coordinate della stazione, quindi si costruisce il contenuto del popup grazie alle informazioni ottenute.

Questo è un classico esempio di popup, dove è possibile ottenere le principali informazioni riguardo alla qualità dell'aria nei pressi di un'area di interesse. Si visualizzano il nome della stazione, l'elenco delle sostanze misurate in essa, il limite di riferimento imposto dalla legge e l'indice. Si ha inoltre un pallino colorato a fianco di ogni sostanza per valutarla singolarmente, basata sulla stessa scala di colori utilizzata per i popup ed una barretta colorata, dello stesso colore del marker, per dare una valutazione generale basata sul peggior indice, cioè quello più alto. Il popup è strutturato in due tab, tutte le stazioni mostrate hanno il tab Indici, mentre solo alcune hanno il tab Andamento Giornaliero, che viene mostrato solamente per le stazioni che offrono più di una misurazione di almeno una sostanza nell'intera giornata.



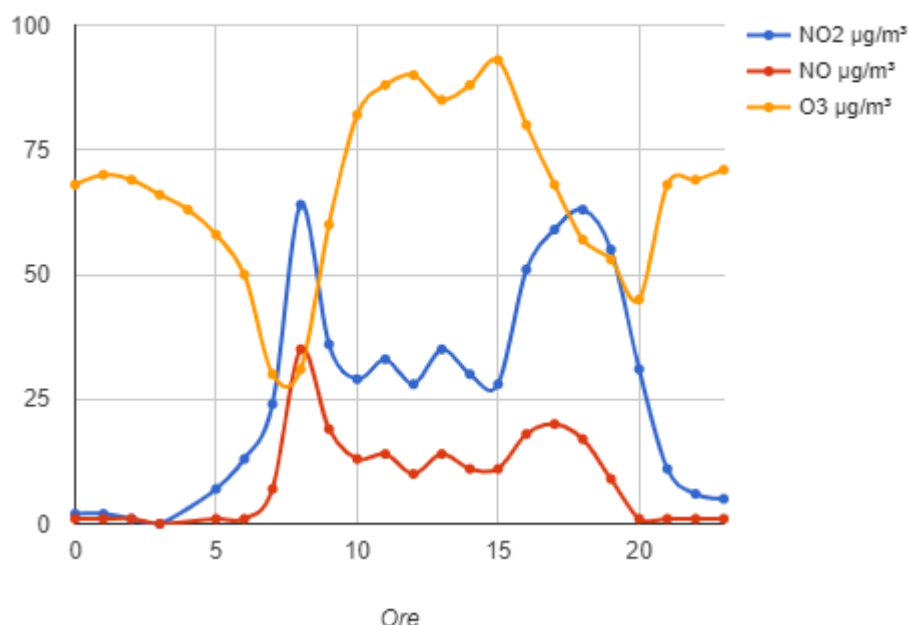
Questo tab, all'apertura del popup è assente in ogni caso. È premura della funzione `drawPopupGraph()` disegnare il grafico se si hanno sufficienti misurazioni e mostrare quindi, tramite un'animazione implementata in jQuery, il tab Andamento Giornaliero. Al contrario, in caso di assenza di misurazioni multiple il tab rimarrà nascosto, e nessun contenuto verrà caricato. L'utente potrebbe essere intenzionato a visualizzare solamente alcune sostanze: ecco perché è disponibile un filtro, selezionabile direttamente dalla mappa, che permette di rimuovere determinate sostanze dai risultati, sia tabellari che sotto forma di grafici. Tra le varie operazioni effettuate all'apertura di un singolo popup è presente anche la verifica dello stato attuale del filtro, effettuata come segue:

```
function get_filters(id) {
    if($(document).find("input[name='sostanza-" + id + "']").is(':checked'))
        return true;
    else
        return false;
}

marker.on('popupopen', function(e) {
    for(var i=0; i<array_Inquinanti.length; i++){
        if(get_filters(i)){
            $('.sostanza-' + i).show(); //selezionato
            //mostra
        }
        else{
            $('.sostanza-' + i).hide(); //nascondi
        }
    }
});
```

Per ognuno degli inquinanti si controlla lo stato del form che costituisce il filtro e in caso di selezione si mostra il risultato, altrimenti si nasconde. Similmente, avviene per la creazione del grafico: si nascondono i dati relativi alle sostanze filtrate.

I grafici permettono di avere una esposizione immediata delle misurazioni giornaliere, agevolando l'utente nell'interpretarle senza difficoltà. Di seguito un esempio di grafico:



Un semplice grafico cartesiano realizzato tramite Google Chart, in cui possiamo osservare i valori delle misurazioni effettuate ora per ora nell'arco della giornata dalla stazione. Si fornisce una legenda che permette di riconoscere grazie ai colori l'andamento di un inquinante e la sua unità di misura.

Passando il mouse sopra ai vari punti, o con un tap da smartphone/tablet, è possibile visualizzare un popup che mostra ora e valore della misurazione.

## 3 Sezione Copertura IQA

Il portale MonIQA è stato arricchito con questa sezione, attraverso la quale l'utente può farsi un'idea della copertura sul territorio nazionale delle stazioni di rilevamento. Per far questo è stata realizzata una heatmap<sup>5</sup>, sfruttando un plugin di leaflet sviluppato dalla community. In particolare, il plugin scelto è stato Leaflet.heat<sup>6</sup>, fornito di una breve ma esaustiva documentazione [3].

### 3.1 Che cosa è una heatmap

Una heatmap è una rappresentazione bidimensionale dei dati, nella quale i valori sono rappresentati dai colori. Una semplice heatmap fornisce una sintesi immediata delle informazioni, mentre le più elaborate, in particolare quelle dotate di una funzione di visualizzazione della densità, permettono alla heatmap di rappresentare la concentrazione di punti all'interno una mappa. Pensato nell'ottica del portale, si ha l'opportunità di rappresentare la densità di stazioni di rilevamento nel territorio nazionale tramite i colori, riferiti ad una opportuna scala a disposizione dell'utente. Il punto di forza di questo metodo di visualizzazione dei dati è la possibilità di comunicare quasi istantaneamente una vasta quantità di informazioni, che in forma tabellare sarebbero difficili da comprendere.

### 3.2 Implementazione

L'implementazione della heatmap sfrutta, come già detto, la libreria Leaflet.heat. Tuttavia, gran parte del processo viene svolto utilizzando le funzioni standard di leaflet.

Come per la sezione mappa, iniziamo recuperando dal server tutte le informazioni necessarie. Subito dopo viene inizializzata la mappa, che in questo momento non ha alcuna differenza da quella della sezione mappa. Una volta ottenuti i dati dal server, si procede alla creazione della heatmap. Per prima cosa vengono processati i dati, cioè si stabilisce, senza analizzarne il valore, quali stazioni abbiano misurazioni in data odierna. Queste stazioni costituiranno i vari punti presenti sulla mappa. La libreria offre molteplici possibilità di personalizzazione per i punti, dal gradiente di colore al livello di sfumatura dei punti. Di seguito, è illustrato come vengono creati e memorizzati i singoli punti, composti da latitudine, longitudine ed intensità, che per questa applicazione è uguale per tutti i punti:

```
for(i ...){
  ...
  if(MisuraOggiPresente == true)
    array_Heat.push([array_Stazioni[i].lat, array_Stazioni[i].lon, 1]);
  ...
}
```

---

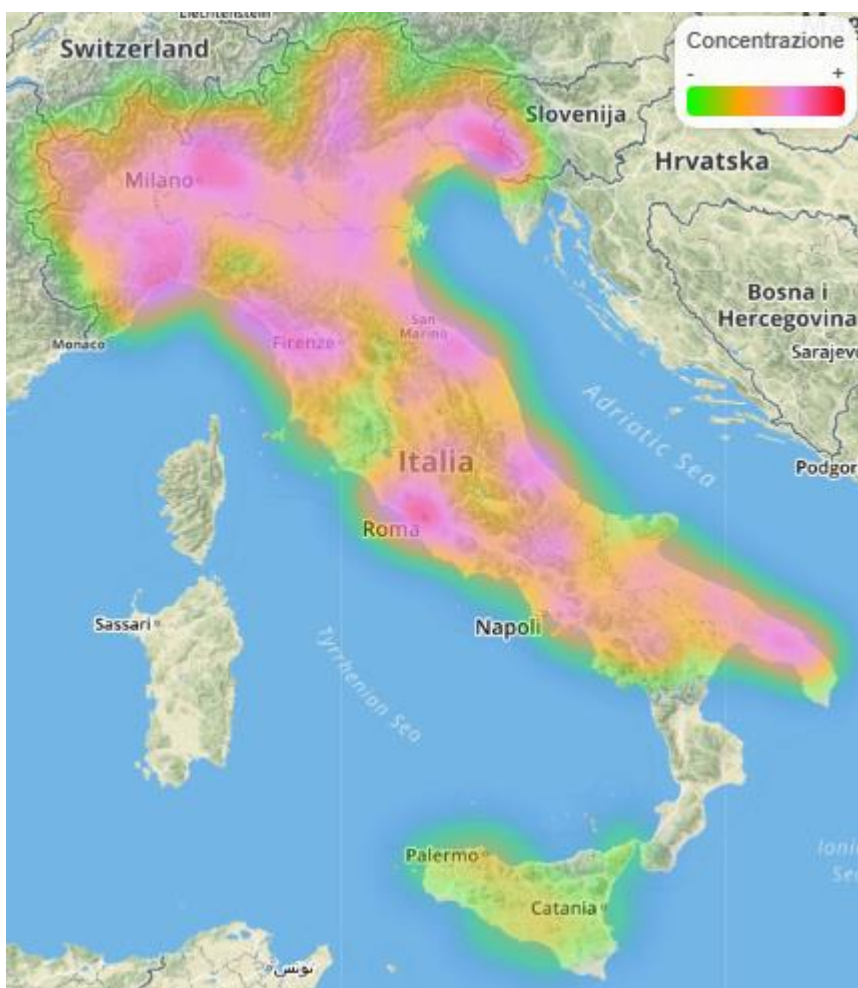
<sup>5</sup> Heatmap: mappa termica

<sup>6</sup> Ottenibile presso <https://github.com/Leaflet/Leaflet.heat>

Una volta raccolte le informazioni riguardo a tutti i punti che si vogliono rappresentare sulla heatmap, si può procedere alla sua creazione, fornendo i punti e scegliendo le varie opzioni come segue:

```
var heatmap = L.heatLayer(array_Heat, {  
  gradient: {  
    0.0: 'blue',  
    0.25: 'lime',  
    0.5: 'orange',  
    0.75: 'violet',  
    1: 'red'  
  }  
}).addTo(myHeatmap);
```

Così facendo è possibile ottenere una heatmap che mostra la copertura sul territorio nazionale delle stazioni di rilevamento. Il risultato finale è di questo tipo:



## 4 Sezione Regioni

La sezione regioni ha il compito di fornire dati sulla qualità dell'aria su base regionale. Questo è possibile grazie alla conoscenza della regione di appartenenza di ogni singola stazione. Il procedimento è molto simile all'elaborazione dei dati vista nella sezione mappa, con la sola differenza che in questo caso si calcola per ogni regione la media degli indici delle varie stazioni. Una informazione immediata sulla qualità dell'aria a livello regionale è data dalla colorazione delle varie regioni, secondo una scala di colori disponibile all'utente. È anche possibile, selezionando col mouse la regione, o con il tap in caso di smartphone/tablet, visualizzare il valore esatto dell'indice percentuale medio della regione in questione.

### 4.1 Come si suddivide la mappa in regioni

La possibilità di suddividere la mappa in aree geografiche è data dal metodo `geoJSON()` offerto dalla libreria leaflet che permette di sfruttare file GeoJSON<sup>7</sup>. Questa necessita appunto del file e di alcune opzioni, tra cui lo stile del bordo delle aree. È possibile applicare ad una normalissima mappa realizzata con leaflet delle aree geografiche visibilmente distinguibili nel seguente modo:

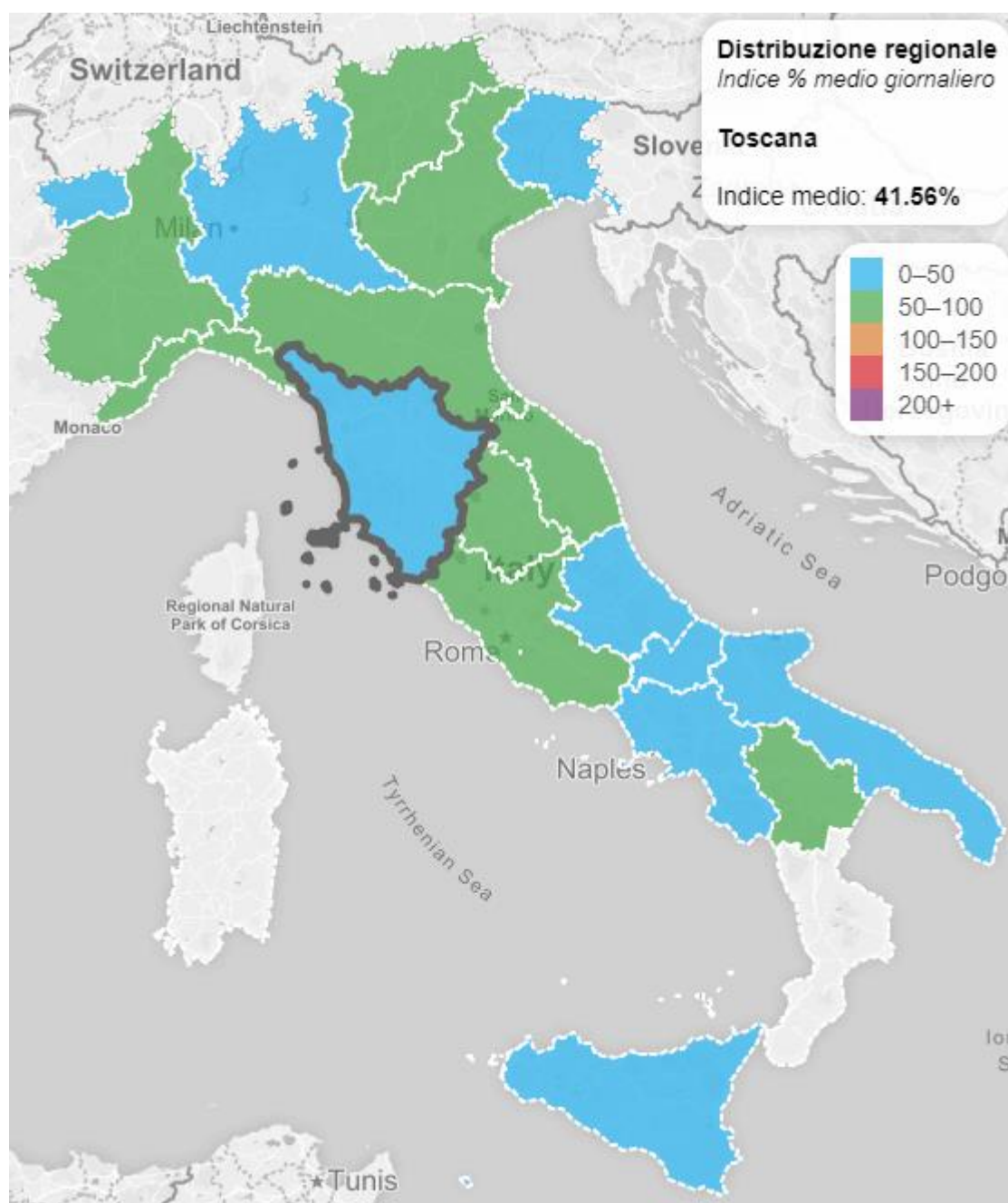
```
function getColor(d) {
  return num 200 ? '#792978' :
    num 150 ? '#d51e29' :
    num 100 ? '#d87c2e' :
    num 50 ? '#46a64a' :
    '#1dadea';
}

function style(){
  return {
    weight: 2,
    opacity: 1,
    color: 'white',
    dashArray: '3',
    fillOpacity: (indicePresente) ? 0 : 0.7,
    fillColor: getColor(indiceMedio)
  };
}

for(var i=0; i<regionData.length; i++)
  geojson = L.geoJson(regionData[i], {
    style: style
  }).addTo(myMap);
}
```

<sup>7</sup> GeoJSON è un formato utilizzato per memorizzare oggetti geometrici come poligoni per definire aree geografiche. Viene utilizzato il JavaScript Object Notation.

I dati riguardano i valori odierni, motivo per il quale, in caso una regione non avesse a disposizione alcun valore, verrà rappresentata senza colore, cioè trasparente. Interagendo con una regione sulla mappa è possibile ottenere delle informazioni come il suo nome ed il valore dell'indice percentuale medio calcolato. Successivamente viene riportato un esempio in cui possiamo anche trovare due regioni prive di misurazioni.





## 5 Sezione Ricerca

La novità più importante per il portale MonIQA è costituita dalla funzione di ricerca. Tramite questa sezione l'utente può visualizzare lo storico delle misurazioni effettuate dalle varie stazioni di rilevamento. Si vuole rendere possibile per l'utente la visualizzazione di risultati d'interesse, rendendoli facilmente accessibili nel tempo e nello spazio.

### 5.1 Contenuto della sezione

La sezione si presenta come un normalissimo form ma senza tasto di invio:

Il form è semplice: permette all'utente di selezionare una città, mostrando successivamente le stazioni appartenenti alla città scelta. Una volta selezionata la stazione, è possibile scegliere la data grazie ad un selettore con calendario.

Come evidente, non è presente un pulsante per il submit. Questo perché il processo è stato automatizzato, in modo tale che alla modifica della stazione o della data la ricerca si avvii in automatico. Solitamente, un utente vuole ottenere informazioni riguardo ad una particolare stazione in una determinata data per poi interessarsi alle date vicine, oppure controllare le misurazioni di stazioni di una stessa città nella stessa data.

Grazie alla presenza di placeholder nei campi del form, la comprensione del funzionamento di questa sezione è immediata per gli utenti.

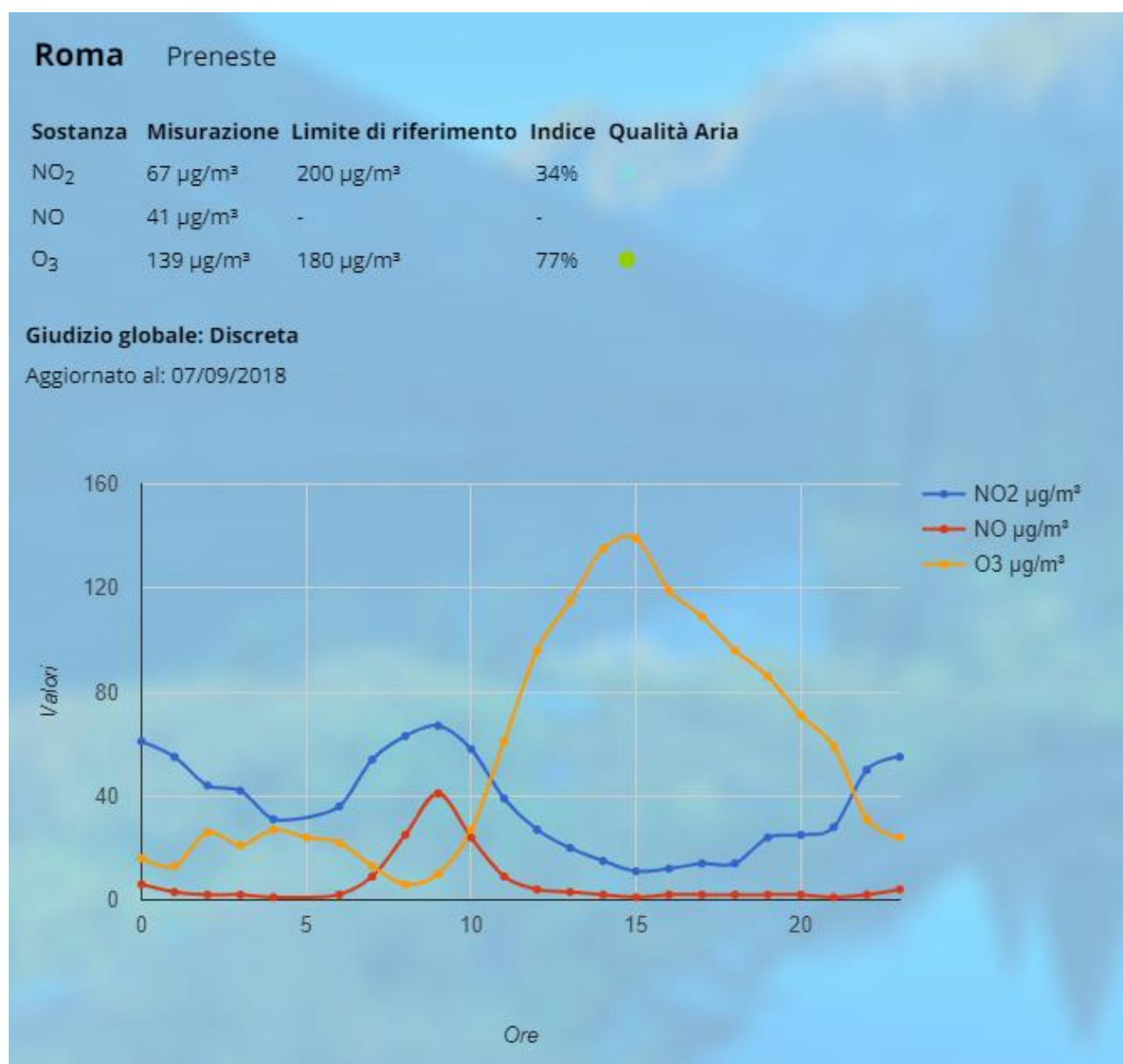
Per mantenere lo stile di questa sezione in linea con il resto del portale, si è deciso di mostrare una tabella identica a quella contenuta nei popup della sezione mappa, così come un grafico che mostri all'utente l'andamento orario degli inquinanti per la data scelta.



Nel paragrafo seguente verranno illustrate le varie possibilità che permettono all'utente di avere un'esperienza piacevole anziché un semplice messaggio di errore in caso di ricerca priva risultati.

## 5.2 Risultati della ricerca

In caso di esito positivo della ricerca, si mostreranno appunto i risultati ottenuti. Con particolare attenzione ad una presentazione semplice dei dati, l'aspetto sarà di questo tipo:



Si mostrano per prima cosa la città, in grassetto, ed il nome della stazione. Subito dopo riportiamo la tabella dei valori, con i relativi limiti di riferimento imposti dalla legge. Infine, abbiamo il grafico

orario delle misurazioni della giornata in questione. L'utente può ottenere i valori precisi delle misurazioni interagendo con i punti del grafico. Come introdotto precedentemente, al fine di non rallentare le operazioni di ricerca, modificare la stazione o la data avvia automaticamente una nuova ricerca.

Tuttavia, non necessariamente si ottengono dei risultati per la ricerca effettuata. Se per la data selezionata non si hanno delle misurazioni da mostrare, sono offerte due possibilità. Vengono automaticamente fornite le due date più vicine, precedente e successiva, se esistenti con misurazioni disponibili. L'utente può quindi, con un semplice click ottenere i risultati per la stessa stazione in una delle date consigliate. Altrimenti l'utente può richiedere la ricerca di stazioni di rilevamento in un raggio di 30km dalla stazione selezionata, che hanno misurazioni disponibili per la data in esame.

Il calcolo della distanza tra due punti geografici, identificati da latitudine e longitudine, deve essere eseguito tenendo conto della sfericità della superficie terrestre. Per questo è stata utilizzata la *'Haversine formula'* che, seppur ignorando la non perfetta sfericità della Terra, non introduce errori significativi, essendo il territorio nazionale di estensione ridotta. Questa formula permette di calcolare l'angolo al centro della Terra  $\theta$  presente tra due punti geografici, utilizzando in seguito la semplice relazione che lega angolo e raggio con la corda che congiunge i due punti  $d = \theta r$  per ottenere l'effettiva distanza in metri.

*Presi due punti qualsiasi, con latitudine  $\varphi$  e longitudine  $\lambda$ , noto che la formula haversine per un qualsiasi angolo  $\vartheta$  è:*

$$hav(\psi) = \sin^2\left(\frac{\psi}{2}\right) = \frac{1 - \cos(\psi)}{2}$$

*Si ha che la distanza  $d$  è pari a:*

$$d = r \, hav^{-1}(c) = 2r \, \arcsin(\sqrt{c}) \quad \text{In cui } c = hav(\theta) \text{ con } \theta \text{ angolo al centro, per } r = 6371 \text{ km (raggio terrestre)}$$

*Noto che:*

$$hav(\theta) = hav(\varphi_2 - \varphi_1) + \cos(\varphi_1) \cos(\varphi_2) hav(\lambda_2 - \lambda_1)$$

*In definitiva si ha che la distanza in metri tra due punti è pari a:*

$$d = r \, hav^{-1}(hav(\varphi_2 - \varphi_1) + \cos(\varphi_1) \cos(\varphi_2) hav(\lambda_2 - \lambda_1))$$

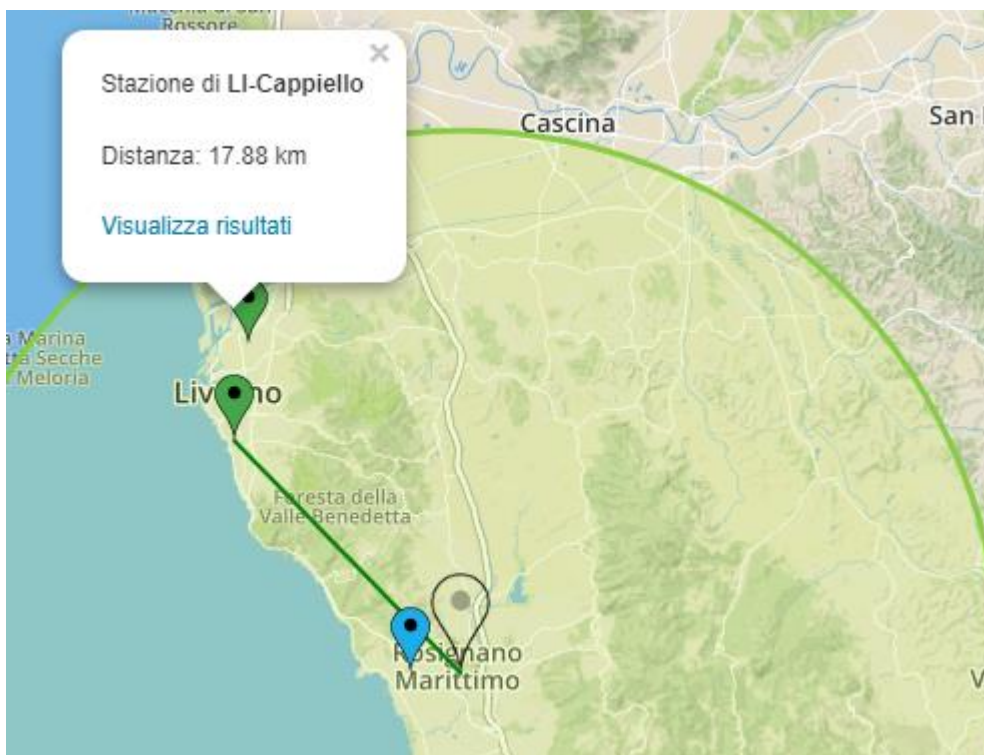
*Che semplificata in termini di funzioni trigonometriche elementari è:*

$$d = 2r \, \arcsin\left(\sqrt{\sin^2\left(\frac{\varphi_2 - \varphi_1}{2}\right) + \cos(\varphi_1) \cos(\varphi_2) \sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)}\right)$$

Si riporta in seguito, secondo quanto detto, l'implementazione in javascript:

```
function haversine(coord1, coord2){
  var R = 6371e3; // metri
  var phi1 = toRadians(coord1.lat);
  var phi2 = toRadians(coord2.lat);
  var deltaphi = toRadians(coord2.lat-coord1.lat);
  var deltalambda = toRadians(coord2.lng-coord1.lng);
  var a = Math.sin(deltaphi/2) * Math.sin(deltaphi/2) +
    Math.cos(phi1) * Math.cos(phi2) *
    Math.sin(deltalambda/2) * Math.sin(deltalambda/2);
  var c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1-a));
  var d = R * c ;
  return d;          //distanza in metri
}
```

Il portale MonIQA non si limita a fornire all'utente una lista di stazioni con le relative distanze, eventualmente in ordine crescente. Si permette all'utente di visualizzare tutte quante le stazioni sotto forma di una mappa, fornita di popup contenenti informazioni come il nome, la distanza ed un link che permette di ottenere i risultati associati immediatamente.



Ancora una volta si cerca di esporre in modo chiaro i risultati all'utente, mostrando un cerchio di raggio 30km ed una linea dello stesso colore del marker selezionato, che collega la stazione del popup a quella in analisi. Inoltre, la stazione più vicina è evidenziata da un colore differente per essere immediatamente riconoscibile.

In questo caso, l'utente, cliccando su '*Visualizza risultati*', avvierà la ricerca per la particolare stazione di '*LI-Cappiello*' utilizzando la stessa data scelta in principio. Questa successiva ricerca ha risultato positivo garantito, in quanto le stazioni, prima di essere visualizzate sulla mappa, vengono analizzate anche sotto questo punto di vista.

### 5.3 Implementazione della funzione di ricerca

La funzione di ricerca può seguire strade diverse a seconda che siano presenti o meno dei risultati per la data o stazione selezionata. È importante che l'utente non arrivi mai ad un punto in cui è impossibile procedere. Un caso particolare potrebbe essere l'immissione di una data successiva a quella attuale o precedente all'attivazione del servizio. In quel caso si provvede a segnalare l'inesistenza di risultati e a suggerire all'utente la data più vicina contenente dati significativi.

La funzione inizia raccogliendo i dati inseriti dall'utente, in particolare servono solamente il nome della stazione e la data. Si procede procurandosi tutte le misurazioni della stazione richiesta dall'utente entro i dieci giorni<sup>8</sup> precedenti e successivi alla data selezionata. Successivamente si avvia la chiamata ad una funzione PHP durante la normale esecuzione del javascript, che a sua volta fornisce al database una query MySQL come la seguente:

```
SELECT measurements.date, measurements.measure, sensortypes.pk_sensortype,
      sensortypes.type, sensortypes.reference, sensortypes.scale
FROM devices join sensors on sensors.fk_device=devices.pk_device
      join measurements on measurements.fk_sensor=sensors.pk_sensor
      join sensortypes on sensortypes.pk_sensortype=sensors.fk_sensortype
WHERE devices.name='$name' AND
      measurements.date BETWEEN '$date'-864000000 AND '$date'+864000000
ORDER BY measurements.date ASC
```

Al completamento di questa operazione, con le misurazioni disponibili si esegue la funzione `showData()`, il cui compito è quello di determinare in quale situazione è il result set. Le varie possibilità comprendono la disponibilità di misurazioni e la loro assenza. In quest'ultimo caso si calcolano le date più vicine a quella selezionata, all'interno delle misurazioni già ottenute e si mostrano all'utente. La procedura ha essenzialmente questa struttura:

```
for(var i=0; i < tenDayMeasure.length; i++){
  tenDayMeasure[i].date = tenDayMeasure[i].date;
  var tmp = new Date(tenDayMeasure[i].date);
  if(tenDayMeasure[i].date >= actualTimestamp &&
      tmp < actualTimestamp + 864000000)
  {
    thisDate.push({ore: tmp.getHours(), type: tenDayMeasure[i].type,
      measure: tenDayMeasure[i].measure });
  }
}
```

In caso di assenza di valori per la data desiderata, si provvede a controllare l'intero storico delle misurazioni tramite la funzione `allMeasureOfStation()` fino a poter fornire all'utente

<sup>8</sup> Questo periodo di tempo è stato scelto in fase di progettazione, per avere il giusto compromesso tra numero di risultati e possibilità di controllare la presenza di risultati nelle date limitrofe.

almeno una data. Il compito di questa funzione è essenzialmente quello di far eseguire al server la seguente query MySQL:

```
SELECT measurements.date
FROM devices join sensors on sensors.fk_device=devices.pk_device
      join measurements on measurements.fk_sensor=sensors.pk_sensor
      join sensortypes on sensortypes.pk_sensortype=sensors.fk_sensortype
WHERE devices.name='$name'
ORDER BY measurements.date ASC
```

Ottenuti questi dati riusciamo poi ad estrarre la data minima e massima presenti tramite la funzione `findCloseAvailableMeasure()`, inserendole in una semplice struttura dati a due campi:

```
function findCloseAvailableMeasure() {
    var result = {previous: 0, next: 0};
    for(var i=0; i<stationAllMeasure.length; i++){
        if( result.previous < stationAllMeasure[i] && stationAllMeasure[i] <
actualTimestamp )
            result.previous = stationAllMeasure[i];
        if( stationAllMeasure[i] >= actualTimestamp+86400000) {
            result.next = stationAllMeasure[i];
            return result;
        }
    }
    return result;
}
```

Viene anche proposto un pulsante che invita l'utente a cercare tutte le stazioni in un raggio di 30km che, a differenza di quella selezionata, hanno delle misurazioni da mostrare. Considerato che questa possibilità potrebbe non essere in generale necessaria all'utente, la procedura di ricerca non viene effettuata preventivamente, ma solamente al click del pulsante. Questo processo richiede di calcolare la distanza fra tutte le stazioni, rimuovendo quelle non idonee ed escludendo successivamente tutte quelle che non hanno misurazioni per la data selezionata. Le stazioni rimanenti verranno poi mostrate in una mappa, come precedentemente visto. Di seguito viene riportato il codice semplificato della funzione:

```
function findCloseStation(stationName, wantedDate) {
    $(function() {
        $.ajax({
            global: false,
            type: "POST",
            data: {"name": stationName, "data": wantedDate},
            url: 'closeStation.php'
        }).done(function(data) {
            stationInSameDate = data;
            minDistance = 30000; //radius = 30km
            chosenStation = -1;
        });
    });
}
```

```

for(var i=0; i<stationInSameDate.length; i++){
    var tmpPoint2 = stationInSameDate[i].coord;
    var tmpDistance = haversine(actualCoord, tmpPoint2);
    if(tmpDistance < 30000){
        closeArray.push({name: stationInSameDate[i].name, latlng:
            tmpPoint2, dist: tmpDistance});
    }
    if( tmpDistance < minDistance){    //get the min distance
        minDistance = tmpDistance;
        chosenStation = i;
    }
}

if(chosenStation != -1)
    mapCloseStation(); //disegna la mappa
else(chosenStation == -1)
    el.innerHTML += '<b>Le stazioni nelle vicinanze non hanno
        misurazioni per la data selezionata.</b>';

});
});
}

```

Come risultato finale si ha che la sezione ricerca fornisce all'utente un'esperienza fluida e guidata in caso di ricerche senza risultati. Questa caratteristica è particolarmente apprezzata dagli utenti meno esperti.

## Conclusioni

L'inquinamento dell'aria è una tematica molto delicata che, come dimostrato, è in grado di arrecare danni significativi all'ambiente e alla salute dell'uomo. Solo negli ultimi anni la popolazione ha iniziato a sensibilizzarsi su questo tema, così come i vari Paesi del mondo hanno cominciato ad impegnarsi seriamente per ridurre le emissioni.

Il progetto di arricchimento del portale MonIQA è stato importante, al fine di fornire agli utenti ulteriori strumenti per il monitoraggio della qualità dell'aria. Inoltre, il passaggio verso un servizio totalmente open source, rappresentato da OpenStreetMap, permette l'accesso a queste informazioni a chiunque, in qualsiasi momento e in forma totalmente gratuita.

## Bibliografia

- [1] European Environment Agency, EEA Report No 13/2017 “Air quality in Europe – 2017”
- [2] Documentazione API Leaflet presso <https://leafletjs.com/reference-1.3.4.html>
- [3] Documentazione API Leaflet.heat presso <https://github.com/Leaflet/Leaflet.heat>