

Universidade de São Paulo  
Instituto de Matemática e Estatística  
Bacharelado em Ciência da Computação

Diego Martos Buoro

## **QASP - Estimativa da qualidade do ar na cidade de São Paulo**

São Paulo  
Novembro de 2016

# **QASP - Estimativa da qualidade do ar na cidade de São Paulo**

Monografia final da disciplina  
MAC0499 – Trabalho de Formatura Supervisionado.

Supervisor: Prof. Dr. Alfredo Goldman vel Lejbman  
Cosupervisor: Prof. Dr. Rômulo Cássio de Moraes Bertuzzi

São Paulo  
Novembro de 2016

# Resumo

A poluição do ambiente externo é um problema presente e invisível ao dia a dia das pessoas, mas poucas sabem que está associado à mortalidade. Estima-se que em torno de 68 mil pessoas foram internadas em todo o estado e que anualmente, 4,6 mil pessoas são vítimas fatais em decorrência do ar poluído apenas na cidade de São Paulo [EXA \[2014\]](#). Apesar de informações fornecidas por organizações de saúde ou agências de controle ambiental para a preservação de saúde em razão da poluição, o acesso a elas ainda é um grande obstáculo para o cidadão. Além disso, as estações de monitoramento na cidade mostram a qualidade do ar em uma área limitada, e na maioria dos casos distante do local do indivíduo, e, portanto há dúvidas no valor a ser utilizado com segurança. Diante das dificuldades expostas, este trabalho tem como objetivo apresentar na forma de um aplicativo, informações relacionadas às estações de monitoramento de maneira clara e obter a qualidade do ar no local do usuário, informando eventuais sugestões para a preservação de sua saúde. O aplicativo, nomeado de QASP (Qualidade do Ar de São Paulo), foi criado para a arquitetura Android, e suporta versões do sistema de codinome *JellyBean* e mais recentes (4.1 ou mais). A localização do usuário é obtida através do GPS. Utilizando o método do Inverso das Potências, é possível estimar a qualidade do ar no local do usuário com os dados das estações. O arquivo que contém os dados é constantemente atualizado em um servidor remoto por um programa que contém duas rotinas escritas em Python. A primeira rotina consiste em fazer uma navegação automatizada para as páginas das estações, extraindo os dados através de um *parser*. Em seguida, a segunda tarefa realiza a transferência do arquivo atualizado para o servidor através de um protocolo de transferência de dados. Finalmente, o aplicativo requisita o arquivo de dados existente no servidor.

**Palavras-chave:** Aplicativo, Android, Qualidade do Ar, Cidade de São Paulo, GPS,



# Abstract

External air pollution is a present and invisible problem to the daily lives of people, but few know that is associated with mortality. It is estimated that around 68k people were hospitalized throughout the state and that each year, 4,6k people are fatal victims of the polluted air only in the city of São Paulo EXA [2014]. Although there is information for health preservation of air pollution from health organizations or environmental control agencies, their access is still a major obstacle to the citizen. Moreover, the monitoring stations of the city show air quality in a limited area and in most cases, there is a certain distance from the location of the individual, and hence there is doubt in the value to be used. Given the difficulties presented, this work aims at presenting in the form of an application, informations related to monitoring stations clearly and get the air quality at the user site, informing any measures to preserve their health. The application, named QASP (Qualidade de Ar de São Paulo), was created for the Android architecture, and supports versions of codename system *JellyBean* or later (4.1 or more). The user's location is obtained by GPS, and using the method Inverse Distance Weighting, you can get the air quality at the user site with data from working stations. The file containing the data is constantly updated on a remote server from a program that contains two routines written in Python. The first routine is to make an automated navigation to pages of the air monitoring stations, extracting the data through a parser. Then, the second task performs the transfer of the updated file to the server via a data transfer protocol.

**Keywords:** App, Android, Air Quality, City of São Paulo, GPS.



# Sumário

<b>Lista de Abreviaturas</b>	<b>vii</b>
<b>Lista de Figuras</b>	<b>ix</b>
<b>Lista de Tabelas</b>	<b>xi</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Motivação . . . . .	1
1.2 Objetivos . . . . .	2
1.3 Estrutura dos Capítulos . . . . .	2
<b>2 Conceitos Teóricos</b>	<b>5</b>
2.1 Poluição Atmosférica . . . . .	5
2.1.1 História e evolução . . . . .	5
2.1.2 Poluentes . . . . .	6
2.1.3 Índice de qualidade do ar . . . . .	7
2.1.4 A preocupação com o $PM$ e $O_3$ . . . . .	8
2.1.5 Obtenção de dados . . . . .	8
2.2 Geodésia . . . . .	11
2.2.1 Latitude e Longitude . . . . .	11
2.2.2 Sistema de Posicionamento Global . . . . .	11
2.2.3 Circulo Máximo . . . . .	12
2.2.4 Fórmula de Haversine . . . . .	13
2.3 Análise Numérica . . . . .	15
2.3.1 Inverso da Potência das Distâncias . . . . .	15
2.4 Protocolo de Transferência de Arquivos . . . . .	17
<b>3 Desenvolvimento do aplicativo</b>	<b>19</b>
3.1 Android . . . . .	19
3.1.1 Interface de Programação de Aplicações . . . . .	19
3.1.2 Java . . . . .	20
3.1.3 Android Studio . . . . .	20
3.2 <i>Script</i> gerador . . . . .	21

3.2.1	Selenium Web Driver . . . . .	21
3.2.2	BeautifulSoup . . . . .	22
3.2.3	Pyhton . . . . .	22
3.2.4	X Virtual Framebuffer . . . . .	23
3.2.5	Cron . . . . .	23
<b>4</b>	<b>Aplicativo QASP</b>	<b>25</b>
4.1	Navegação Principal . . . . .	25
4.2	Preferências . . . . .	29
<b>5</b>	<b>Conclusões</b>	<b>33</b>
5.1	Trabalhos Futuros . . . . .	33
	<b>Referências Bibliográficas</b>	<b>35</b>



# Lista de Abreviaturas

OMS	Organização Mundial da Saúde( <i>World Health Organization</i> )
IDW	Método do Inverso das Potências das Distâncias( <i>Inverse Distance Weighting</i> )
FTP	Protocolo de Transferência de Arquivos( <i>File Transfer Protocol</i> )
HTML	Linguagem de Marcação de Hipertexto( <i>HyperText Markup Language</i> )
GPS	Sistema de Posicionamento Global( <i>Global Positioning System</i> )
JDK	Kit de Desenvolvimento Java( <i>Java Development Kit</i> )
IDE	Ambiente de Desenvolvimento Integrado( <i>Integrated Development Environment</i> )
SO	Sistema Operacional( <i>Operational System</i> )
URL	Localizador Uniforme de Recursos( <i>Uniform Resource Locator</i> )
PM	Material Particulado( <i>Particulate Matter</i> )
ODM	OpenDustMap
CETESB	Companhia de Tecnologia e Saneamento Ambiental
RMSP	Região Metropolitana de São Paulo
QASP	Qualidade do Ar de São Paulo



# Lista de Figuras

2.1	Poluição do ar na cidade de São Paulo. Fonte: <a href="#">freeimages</a> . . . . .	5
2.2	Sensor utilizado pelo grupo CódigoUrbano. Autor: Ana Elisa Santana, Fonte: <a href="#">CódigoUrbano</a> . . . . .	9
2.3	Funcionamento do GPS. Os satélites usados para a determinação da localização são os pontos vermelhos, e as informações deles são usadas para determinar o resultado final. . . . .	12
2.4	O menor arco (em vermelho), obtido ao determinar o círculo máximo entre P e Q, representa a menor distância entre esses pontos na esfera. . . . .	13
2.5	Ilustração da Lei dos haversines na esfera unitária. . . . .	14
2.6	Ilustração do IDW no plano. Pontos A e B tem maior influência no resultado de $u(x)$ do ponto vermelho do que os pontos E e D. . . . .	17
4.1	Tela inicial ao abrir o QASP. . . . .	26
4.2	Exemplo do resultado ao pressionar o botão de seta para baixo, com as estações devidamente atualizadas. . . . .	27
4.3	Tela com a o resultado do índice da qualidade do ar no local do usuário. . . . .	28
4.4	Tela para o envio de <i>feedbacks</i> . . . . .	29
4.5	Tela que mostra as opções existentes em “Configurações” . . . . .	30
4.6	Seleção do grupo (perfil de saúde), note que é possível apenas uma escolha das três. . . . .	31



# Lista de Tabelas

2.1	Índices e cor definidos para cada qualidade do ar. . . . .	8
2.2	Poluentes medidos por estação . . . . .	10
3.1	Dados da distribuição dos dispositivos por codinome. . . . .	20



# Capítulo 1

## Introdução

Desde o surgimento da Revolução Industrial, a poluição atmosférica é uma preocupação crescente devido à atividade industrial e quantidade de veículos automotores. Apesar dos esforços dos governos por um controle quantitativo mais rigoroso nas emissões dos poluentes, a população em grandes cidades e aglomerações urbanas ainda sofre com períodos de alta concentração. Desse modo, para que seja possível a implementação de políticas e medidas eficientes com o objetivo de minimizar os transtornos ocasionados por esses poluentes, órgãos e agências governamentais, através de estações de medição espalhadas pela cidade, monitoram em tempo real a concentração dos poluentes. Assim, os dados obtidos são inseridos em modelos matemáticos que fazem o cálculo do índice da qualidade do ar para aquela estação, classificando a qualidade do ar em relação aos efeitos na saúde do ser humano.

Definir a qualidade do ar é fundamental para esclarecer qual vai ser o impacto na saúde de um indivíduo. Uma exposição contínua e prolongada aos poluentes atmosféricos pode trazer problemas agudos (curto prazo) e crônicos (longo prazo) ao sistema respiratório de um indivíduo. De todos os poluentes medidos, o Ozônio e principalmente o Material Particulado ou Particulado, recebem o maior destaque por conta de seus malefícios e presença na Troposfera [EEA \[2011\]](#), especialmente na época do inverno onde a falta de chuvas e circulação de ventos impede a dispersão de ambos. De exposição imediata, definida como a constante inalação dos contaminantes por um longo período de tempo, são comuns sintomas de problemas no sistema circulatório ou respiratório. Por outro lado, através de uma exposição diária constante vincula-se a poluição do ar com a mortalidade e morbidade. A Organização Mundial da Saúde (OMS) em uma publicação de 2012 estimou que em torno de 3,7 milhões de pessoas morreram prematuramente devido à poluição do ar do ambiente externo [OMS \[2014\]](#).

### 1.1 Motivação

Devido às consequências na saúde, criaram-se orientações a fim de prevenir os efeitos dos poluentes no ambiente externo. Essas indicações detalhadas não só atendem a população em geral, mas também grupos com características específicas, como pessoas sensíveis e atletas. Entretanto, apesar da existência de dados por parte das estações e das formas de prevenção pelas organizações de saúde, a comunicação dessas informações de maneira fácil e prática para o usuário ainda é um desafio. Normalmente em páginas da web, apenas a qualidade do ar é exibida sem quaisquer outras informações que beneficiariam o usuário, como medidas de prevenção à exposição a esses poluentes. Ademais, as medidas de prevenção devem levar em consideração não só apenas a qualidade do ar, mas também o contaminante responsável.

Além disso, o fato dos dados obtidos se restringirem ao entorno das estações, somado

à ausência de monitoramento em certos locais da cidade levam à falta ou inconsistência de informação, como acontece em São Paulo. Em outras palavras, a falta de conhecimento da qualidade do ar no local do usuário, por conta dos empecilhos descritos anteriormente impede o utilizador de tomar medidas que preservariam sua saúde.

## 1.2 Objetivos

Em razão dos problemas acima, propõe-se neste trabalho o desenvolvimento de um aplicativo sob a licença Pública Geral GNU GPL [2002] de forma a exibir medidas preventivas, adequadas tanto para a população como para grupos específicos, a serem tomadas a partir da qualidade de ar exibida, dada a localização do usuário em São Paulo. Espera-se:

- Obtenção de dados das estações de monitoramento na zona Alto Tietê, exibindo em uma lista das informações pertinentes ao usuário comum: nome e endereço da estação, qualidade do ar e o maior valor do índice de qualidade (o pior caso), e o poluente responsável pela qualidade do ar.
- Cálculo da qualidade do ar no local do usuário, exibindo informações relevantes como o local do usuário, a qualidade do ar e o maior valor do índice de qualidade e eventual recomendações para a preservação de saúde.
- Envio de *feedback* para atualização e correções do aplicativo.
- Acesso simples, rápido e fácil entre os três itens citados acima no aplicativo.
- Seleção de opções pelo usuário para melhor exibição das estações e recomendações de saúde mais precisas a depender do perfil de saúde do usuário.

## 1.3 Estrutura dos Capítulos

Sendo assim, uma série de capítulos será apresentada a seguir para fundamentar a base teórica e os métodos que serão utilizados:

- No Capítulo 2, será feita a introdução de conceitos e definições da literatura científica usadas pelo aplicativo, começando com o estudo sobre a poluição externa na seção 2.1, fornecendo informações detalhadas sobre os poluentes e a coleta de seus dados na Região Metropolitana de São Paulo (RMSP). Em seguida na seção 2.2, exibir-se-ão conceitos e formulações das ciências geodésicas que são utilizados na representação geográfica do planeta. Finalmente, examinar-se-á o método de interpolação que é responsável pela determinação da qualidade do ar no local do usuário na seção 2.3, encerrando-se o capítulo com o protocolo utilizado para a transferência de informações na seção 2.4.
- No Capítulo 3, o sistema operacional Android será abordado na seção 3.1, dando mais ênfase na discussão da versão mínima necessária para o aplicativo, e posteriormente, a descrição do *script* externo utilizado, em conjunto com ferramentas externas, para a obtenção e tratamento dos dados na seção 3.2.
- No Capítulo 4, serão exibidos as funcionalidades do aplicativo com imagens de exemplo, explicando alguns detalhes de otimização na realização de ações.



- Finalmente, no Capítulo 5, concluir-se-á o trabalho fazendo um balanço entre dificuldades e acertos, explicitando ideias com potencial aplicabilidade visando a melhoria do aplicativo.



# Capítulo 2

## Conceitos Teóricos

### 2.1 Poluição Atmosférica

#### 2.1.1 História e evolução

Antes da Revolução Industrial, a poluição atmosférica era limitada a queima de madeira e a pequena atividade de artesanato e, portanto nunca houve um estudo acerca de seus impactos devido a baixa e localizada quantidade emitida. Entretanto, o advento da Revolução Industrial trouxe além da introdução de máquinas e novos processos na manufatura, a popularização do uso de carvão como combustível. Esse uso comum e em larga escala pelas indústrias deu origem aos primeiros problemas de poluição no ambiente externo, com o surgimento de nevoeiros nas cidades altamente industrializadas. Atualmente, com a formação de grandes centros urbanos e o número cada vez maior de veículos automotores nessas regiões, a atividade industrial deixou de ser o principal fator de impacto na qualidade do ar, como ilustra a figura 2.1. Tendo em vista que o estudo e monitoramento dos poluentes atmosféricos tornaram-se fundamentais para o controle das emissões dos contaminantes, descreve-se abaixo os poluentes que normalmente são medidos pelas estações de monitoramento do mundo e da cidade de São Paulo.



**Figura 2.1:** *Poluição do ar na cidade de São Paulo. Fonte: [freeimages](#)*

### 2.1.2 Poluentes

Os poluentes são classificados de acordo com a sua origem. Aqueles que são gerados diretamente da fonte, como acontece na combustão incompleta de combustíveis fósseis em veículos automotores, são chamados de poluentes primários. Por outro lado, se a produção é realizada através de uma série de reações químicas na atmosfera, então classifica-se como poluente secundário. É possível que o mesmo poluente receba ambas classificações.

#### Monóxido de Carbono

O monóxido de carbono ( $CO$ ) é um poluente primário formado a partir da queima incompleta de combustíveis a base de carbono liberados em queimadas feitas pelo homem ou por motores a combustão. A redução de oxigenação nos tecidos é um efeito bem conhecido e notório do  $CO$ , e alguns de seus sintomas incluem dor de cabeça, náuseas, tontura, febre, vômitos e sensação de fraqueza. Apesar de ser considerado letal em ambientes fechados com alta concentração do gás, no ambiente externo é um poluente temporário e possui mais relevância como um dos reagentes para a formação de Ozônio.

#### Dióxido de Enxofre

As principais origens do dióxido de enxofre ( $SO_2$ ) são a queima de combustíveis que contém sulfeto ou como subproduto da atividade industrial e, portanto classificando-se como um poluente primário. O combustível fóssil, principalmente o de carvão ou óleo, são os que possuem maior concentração de sulfeto. Esse composto químico, quando presente no combustível, irá liberar também  $SO_2$  para atmosfera. O uso de carvão para aquecedores domésticos, geração de energia estacionária e veículos automotores são as fontes dominantes de emissão. A exposição a esse poluidor se dá pela inalação, e como o  $SO_2$  é solúvel em meio aquoso, a absorção acontece nas mucosas do nariz e do trato respiratório superior. O exercício físico reduz essa absorção, porém permite a passagem do poluente para as pequenas vias aéreas. Uma exposição curta (de até no máximo 1 hora sem interrupção) implica em uma redução de índices de capacidade ventilatória dos pulmões e sintomas como chiado e falta de ar. Esses sintomas são amplificados em situação de realização de exercícios físicos ou com grupos sensíveis que possuem uma menor resistência.

#### Dióxido de Nitrogênio

O dióxido de nitrogênio ( $NO_2$ ) tem origem principalmente na queima de combustíveis fósseis na combustão interna dos motores, liberando então o poluente primário pelo escape do veículo quando o mesmo não possui um funcionamento adequado do catalisador. Como no ambiente se apresenta na forma de gás, a exposição humana ao poluente também se dá pela inalação. O contaminante pode ser absorvido pelo trato respiratório, e a maior exposição se dá na região de troca de gases nos pulmões. Sabe-se, em experiências com outros animais, que há efeitos no metabolismo, estrutura, função e inflamação dos pulmões. Entretanto, devido à falta de informação sobre o tipo de exposição ou dosagem inalada que poderiam causar tais mudanças em humanos, é difícil deduzir se é tóxico ou não quando o gás se encontra disperso no ambiente. O  $NO_2$  é outro poluidor temporário e também o mais importante para a formação do Ozônio, por ter um tempo de vida menor e ser produzido em maior quantidade que o  $CO$ .

## Ozônio

O Ozônio( $O_3$ ) é formado através de uma série de reações químicas complexas quando uma mistura de Oxigênio( $O_2$ ) e  $NO_2$  ou  $CO$  é exposta a luz solar. Diferente dos poluidores descritos anteriormente, onde a origem se dá na fonte (poluente primário), a formação do  $O_3$  se dá na atmosfera, recebe a classificação de poluente secundário. A concentração do gás sofre uma variação espacial considerável, já que pode ser transportado para outras áreas com a presença de ventos. Desse modo, pode existir uma concentração maior nos arredores da cidade, em subúrbios, ou até mesmo em áreas mais arborizadas e que não possuem uma grande concentração de veículos. Essa locomoção do gás é a explicação mais plausível para que o Campus da Universidade de São Paulo tenha sido considerado a região mais poluída pelo segundo ano consecutivo FOL [2016]. A combinação de megacidades com emissões significativas de precursores do Ozônio, somado ao clima que favorece as reações fotoquímicas potencializam ainda mais a formação do gás. Os efeitos nos seres humanos, quando em alta concentração, têm relevância em respostas agudas, como a inflamação e redução da defesa pulmonar, a morbidez e a mortalidade principalmente para idosos.

## Material Particulado

O Material Particulado ( $PM$ ) pode ser encontrado na forma de partículas sólidas, ou suspensas em gotículas no ar ou uma mistura dos dois, e é considerado tanto um poluente primário como secundário. Sua geração natural provém da atividade de vulcões, tempestades de areia e queimada. Já a queima de combustíveis fósseis, principalmente no uso do Diesel, e a atividade industrial são as principais fontes pela atividade humana. Além disso, o poluente recebe uma outra subclassificação dependendo do tamanho do seu diâmetro aerodinâmico em micrômetros ( $\mu m$ ). São classificadas como Material Particulado Grosso ( $PM_{10}$ ) as partículas que possuem um diâmetro menor ou igual que  $10\mu m$  e maior ou igual a  $2,5\mu m$ . Se esse diâmetro for menor que  $2,5\mu m$ , então passa para a classe de Material Particulado Fino( $PM_{2.5}$ ). Como regra, quanto menor o tamanho do diâmetro para uma partícula, maior são os malefícios que essa pode causar devido a exposição. Dessa forma, as estações mantêm medições separadas tanto para  $PM_{2.5}$  como para  $PM_{10}$ . Seja de qual subclasse, os sintomas causados são os mais variados: induzir inflamação pulmonar, redução da função pulmonar, disfunção do sistema vascular e pulmonar, aumento do risco de câncer e artrite reumatóide, aumento da pressão arterial em descanso, redução da variabilidade da frequência cardíaca além de contribuir com o aumento da mortalidade e morbidade.

### 2.1.3 Índice de qualidade do ar

Diante dos vários poluentes e seus diferentes níveis de concentrações que o tornam tóxico, criou-se a necessidade de descobrir uma maneira simples de classificar a qualidade do ar. Para isso, foi criado o índice de qualidade do ar, que é um valor inteiro adimensional obtido ao mapear a medida de concentração do poluente. Uma vez calculado o índice, é possível avaliar a qualidade do ar e atribuir uma cor.

Apesar do uso comum de técnicas de interpolação para o cálculo do índice de qualidade do ar, os três diferentes modelos diferem na escala para o mapeamento desse índice. O primeiro, baseia-se no sistema criado pelos Estados Unidos, onde as concentrações de cada poluente são transformados em uma escala numérica normalizada de 0 a 500. Em alguns países, como na Austrália, o segundo modelo troca a escala normalizada utilizada pelo sistema americano por uma escala linear ou proporcional, a fim de levar em conta a variação de concentração. Por último, o modelo europeu faz uso de uma escala de 1 a 10, onde cada número representa

uma faixa de cor com a sua a qualidade do ar. Este modelo é utilizado pelo Reino Unido, por exemplo. Usada como referência no aplicativo, a tabela 2.1 mostra para cada qualidade do ar, os valores de índice possíveis na forma de um intervalo, e a cor associada.

**Tabela 2.1:** Índices e cor definidos para cada qualidade do ar.

Qualidade	Índice	Cor
Boa	[0,40]	Verde
Mediana	[41,80]	Amarelo
Ruim	[81,120]	Laranja
Muito Ruim	[121,200]	Vermelho
Péssima	[200, infinito]	Roxo

Como os índices são extraídos diretamente (mais detalhes na seção 3.2) e a classificação da qualidade do ar segue os mesmos critérios da Companhia de Tecnologia e Saneamento Ambiental (CETESB)<sup>1</sup>, que baseia-se no modelo americano, não há a necessidade de fazer ou modificar esse mapeamento dentro do aplicativo e, portanto, otimizando o processamento das informações.

#### 2.1.4 A preocupação com o $PM$ e $O_3$

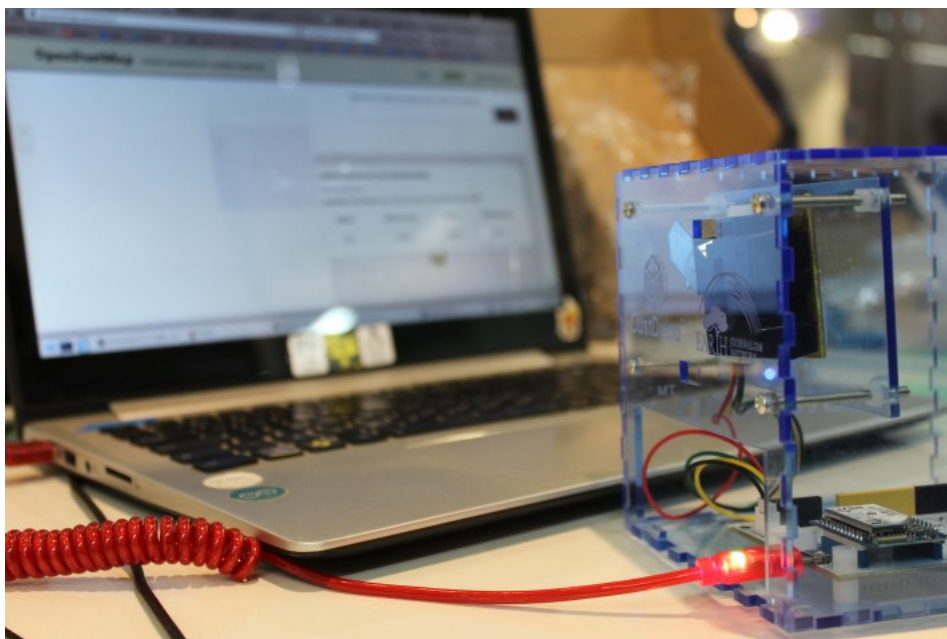
De todos os poluentes, o Particulado e o  $O_3$  são os que recebem mais atenção dentro das organizações de saúde. Apesar do ozônio não apresentar tantos malefícios, esse poluente merece destaque por sua alta produção na atmosfera, principalmente no verão. Por outro lado, o Particulado, especialmente o  $PM_{2.5}$ , merece atenção devido tanto aos efeitos agudos (imediatos) e crônicos (dosagens ao longo do tempo). Essa preocupação com os dois poluentes se justifica ainda mais em grupos sensíveis da população que são definidos pelo estado de saúde ou por características específicas, formado por crianças em fase de desenvolvimento, mulheres no período gestativo, doenças pré-existentes em indivíduos e adultos com idade avançada. Assim, os indivíduos que se encontram nesses grupos são mais propensos a sentir efeitos danosos à saúde diante da exposição aos poluentes. Os atletas representam outro grupo em que a exposição é maior durante o exercício físico, em virtude da alta ventilação do organismo. Entre os dois poluentes, o Particulado representa o maior risco para os atletas, já que não é incomum a prática de exercícios próxima a vias com alto tráfego. Desse modo, a prática de exercícios em ambientes poluídos deve ser evitada para preservar o desempenho e a saúde do atleta.

#### 2.1.5 Obtenção de dados

Mesmo com o estudo científico dos malefícios que esses contaminantes relacionado à poluição atmosférica possam causar, um outro desafio é a obtenção das concentrações dos poluentes. Necessita-se de muitas estações bem distribuídas dentro da região desejada para se obter uma boa cobertura de monitoração. No Brasil, entretanto, muitas regiões do país ainda carecem de sistemas para tal monitoramento. Segundo o relatório de 2014 do Instituto de Saúde e Sustentabilidade sobre o monitoramento da qualidade do ar, apenas as regiões mais populosas no centro-sul do país possuem um sistema de medição por parte do órgão ambiental do estado ISS [2014].

<sup>1</sup>Detalhes no site <http://ar.cetesb.sp.gov.br/padroes-de-qualidade-do-ar/>

Na cidade de São Paulo, a CETESB desenvolveu o sistema QUALAR [QUA \[2009\]](#), e instalou estações tanto na região metropolitana como em outras cidades do estado. Apesar de ser comum ter esse tipo de monitoramento por parte de um departamento do estado, é possível que uma iniciativa livre ou empresa venha fazer suas próprias medições, como é o caso do grupo Código Urbano. Essa frente, utilizando um sensor mais barato, como mostra a figura 2.2, mede a qualidade do ar com melhor precisão em tempo real e manda as informações para a internet, que podem ser acessadas no site OpenDustMap ([ODM](#)). Uma grande vantagem é que a instalação desses sensores pode ser feita por pessoas comuns que estejam interessadas em medir a qualidade do ar em seus bairros.



**Figura 2.2:** Sensor utilizado pelo grupo CódigoUrbano. Autor: Ana Elisa Santana, Fonte: *CódigoUrbano*

O número de estudos e pesquisas a respeito do impacto ou concentração da poluição do ar de São Paulo são vastos, mas apenas o QUALAR e o ODM dispõem dos recursos necessários para fazerem as medições de hora em hora sobre a qualidade do ar. Como não existe nenhum interesse por parte do governo para o uso de sistemas mais sofisticados feitos pelas grandes empresas de tecnologia, o QUALAR e o ODM são os únicos sistemas capazes de obter informações dos poluentes do ambiente em São Paulo. Devido à quantidade maior de estações de monitoramento e por estarem mais bem distribuídas na cidade de São Paulo, e mesmo com algumas áreas carentes de acompanhamento usa-se o sistema QUALAR para a obtenção de dados nesse trabalho. A tabela 2.2 mostra para cada estação da região do Alto Tietê, os poluentes medidos.

**Tabela 2.2:** *Poluentes medidos por estação*

Estação	PM2,5	PM10	O3	SO2	NO2	CO
Carapicuíba		X	X		X	X
Diadema		X	X			
Guarulhos - Paço Municipal		X	X		X	
Guarulhos - Pimenta	X	X	X	X	X	X
Mauá*						
Osasco		X		X	X	X
Santo André - Capuava		X	X	X		
Santo André - Centro		X				X
São Bernardo - Pauliceia		X				
São Bernardo - Centro	X		X		X	X
Capão Redondo		X	X		X	
São Caetano do Sul	X	X	X		X	X
Congonhas	X	X				
Ibirapuera	X		X		X	X
USP - Ipen	X		X		X	
Interlagos		X	X	X	X	
Itaim Paulista		X	X			
Itaquera			X			
Marigonal Tietê	X	X		X	X	X
Mooca		X	X			X
Parelheiros	X	X	X		X	X
Parque Dom Pedro II		X	X	X	X	X
Pinheiros			X		X	X
Santana		X	X			
Santo Amaro		X	X			X



## 2.2 Geodésia

Naturalmente, dado o objetivo do aplicativo, a obtenção da localização do usuário e os cálculos das distâncias entre ele e as estações na superfície terrestre, apresentam-se como um problema. A solução foi usar conceitos básicos e fundamentais das ciências geodésicas, pois é possível representá-los na implementação de funcionalidades no aplicativo. A Geodésia, uma ramificação da matemática aplicada e geociência, é a ciência dedicada a medição e representação de planetas, incluindo a Terra.

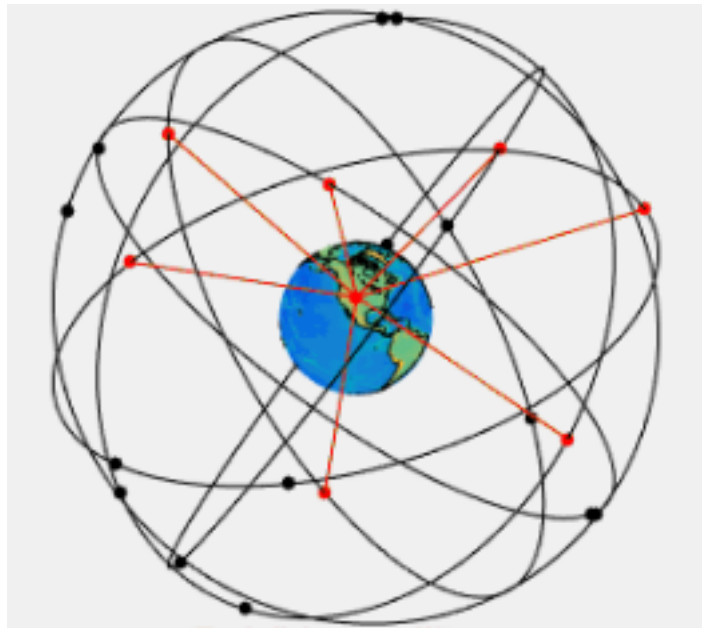
Por conta das características geográficas do planeta Terra, necessita-se da utilização de outros sistemas de coordenadas para a distância e a localização de pontos. Em relação a geometria geodésica, existem dois tipos de problemas dentro dessa área científica: o problema direto e indireto. No primeiro, dado um ponto com suas coordenadas, a direção do Azimute (ângulo medido no plano horizontal entre o meridiano do lugar do observador e o plano vertical que contém o ponto observado) do primeiro, e a distância até o segundo ponto, determinam-se as coordenadas do segundo ponto. Já no caso indireto, dado as coordenadas dos dois pontos, normalmente calcula-se o Azimute dos dois pontos e a distância em linha reta entre eles. Como o interesse está na obtenção do valor da distância, a apresentação e explicação dos conceitos abaixo é direcionado ao problema indireto. A solução desse problema, isto é, encontrar o valor da distância entre os dois pontos, será usado para o método particular no próximo capítulo.

### 2.2.1 Latitude e Longitude

A partir do uso das duas coordenadas Latitude e Longitude, é possível descrever qualquer posição na superfície do planeta. A Latitude, representada pela letra grega  $\phi$ , é a distância, em graus, do ponto desejado em relação à Linha do Equador. Por definição, a linha se encontra no  $0^\circ$ , pois separa os hemisférios norte e sul. O valor da distância acima da linha é positivo e vai até  $90^\circ$  e abaixo da linha é negativo e vai até  $-90^\circ$ . Por outro lado, a Longitude, representada pela letra grega  $\gamma$ , é a distância, em graus, do ponto desejado em relação ao Meridiano de Greenwich. Essa linha vertical imaginária é a separação entre os hemisférios ocidental e oriental, e também se encontra  $0^\circ$ . Assim, a leste do meridiano se encontram os valores positivos, chegando ao máximo a  $180^\circ$ , e a oeste se encontram as distâncias negativas chegando a no mínimo  $-180^\circ$ . Observa-se que é utilizado para as coordenadas geográficas o sistema de notação de graus, minutos e segundos ao invés do ângulo decimal por convenção.

### 2.2.2 Sistema de Posicionamento Global

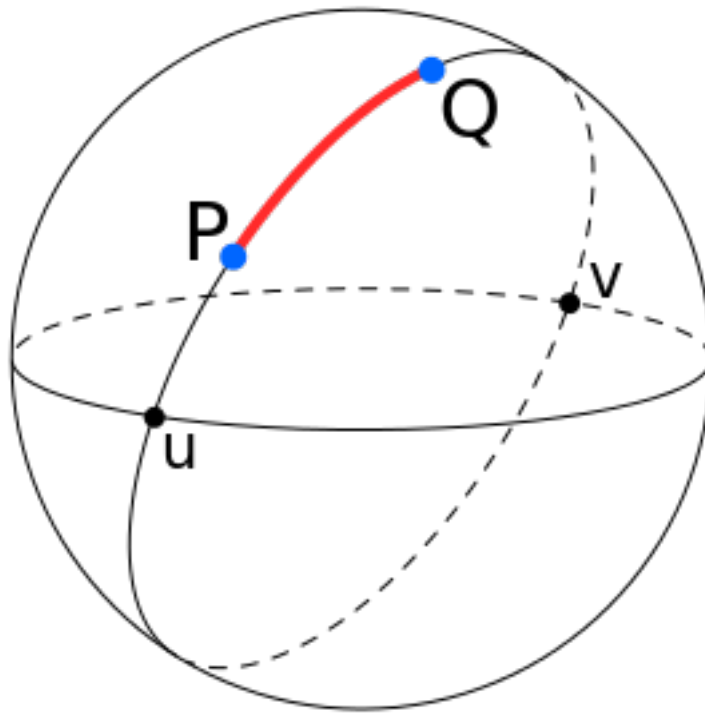
O Sistema de Posicionamento Global, mais conhecido como GPS (do inglês *Global Positioning System*), foi criado em 1995 pelo Departamento de Defesa Americano, originalmente para fins militares. Ativo desde então, o sistema é o mais popular para encontrar a posição de um indivíduo no planeta, fazendo uso de informações coletadas por satélites de comunicação que orbitam a Terra. Essas informações são transmitidas por sinal de rádio para um receptor no planeta, e então a partir do cruzamento das localizações obtidas de vários satélites, é possível determinar a posição de uma pessoa no planeta com altíssima precisão, como ilustra a figura 2.3. Portanto, serviços que fazem uso do GPS possuem dados como as coordenadas geográficas, descritas anteriormente na seção 2.2.1, e o endereço do local do usuário. Estimam-se que para fazer a cobertura do planeta, existem aproximadamente 2 mil satélites em operação na órbita da Terra [Labrador \[2015\]](#).



**Figura 2.3:** *Funcionamento do GPS. Os satélites usados para a determinação da localização são os pontos vermelhos, e as informações deles são usadas para determinar o resultado final.*

### 2.2.3 Circulo Máximo

Antes de entrar em detalhes nos métodos abaixo, necessita-se descrever alguns conceitos que são utilizados para o cálculo da distância. O mais importante, que recebe o nome de Circulo Máximo ou Grande Círculo, é a intersecção entre uma esfera e de um plano passando no centro da mesma. Como o diâmetro do círculo máximo coincide com o diâmetro da esfera, todos os círculos máximos possuem a mesma circunferência e o mesmo centro da esfera. Dessa forma, dois pontos dentro do Circulo Máximo dividem o comprimento de sua circunferência em dois arcos, como mostra a figura 2.4. Assim, o menor arco é chamado de ortodromia ou distância do círculo máximo e representa a menor distância entre dois pontos em linha reta na superfície de uma esfera. Existe em geral apenas um único Circulo Máximo para cada 2 pontos, com exceção no caso em que os pontos são chamados de antipodais, que são diametralmente opostos, e, portanto o comprimento dos arcos são iguais. Nessa situação, encontram-se infinitos grandes círculos, onde um exemplo seria a distância entre polo norte e sul da Terra.

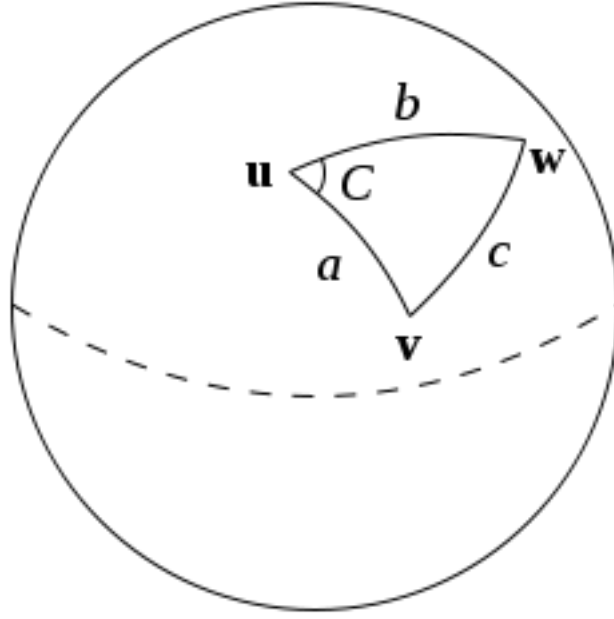


**Figura 2.4:** O menor arco (em vermelho), obtido ao determinar o círculo máximo entre P e Q, representa a menor distância entre esses pontos na esfera.

## 2.2.4 Fórmula de Haversine

### Lei de Haversines

Foi descoberto em 1805 uma das fórmulas mais importantes dentro da navegação, que fornecia o valor das ortodromias baseado na Latitude e Longitude dos pontos fornecidos, chamada de Fórmula de Haversine. Considerando que a Terra é uma esfera, a fórmula é obtida através de um caso especial, descrito em mais detalhes abaixo, da lei dos haversines, que relaciona lados e ângulos em triângulos esféricos como mostra a figura 2.5.



**Figura 2.5:** Ilustração da Lei dos haversines na esfera unitária.

Seja uma esfera unitária, isto é, o conjunto de pontos  $(x_1, x_2, x_3)$  que satisfaz  $x_1^2 + x_2^2 + x_3^2 = 1$ , e um triângulo esférico cujos vértices são  $u, v$  e  $w$ . Se  $a$  é a distância de  $u$  a  $v$ ,  $b$  de  $u$  a  $w$  e  $c$  de  $v$  a  $w$ , e o ângulo oposto a  $c$ ,  $C$ , então a lei dos haversines diz que:

$$\text{hav}(c) = \text{hav}(a - b) + \sin(a) \sin(b) \text{hav}(C) \quad (2.1)$$

onde  $\text{hav}$  representa a função Haversine, dada por:  $\text{hav}(\theta) = \sin^2(\frac{\theta}{2})$

### Fórmula

A partir da enunciação da lei de haversines, agora é possível obter a Fórmula de Haversines. Considere o caso particular em que  $u$  está localizado no polo norte. Seja  $d$  a distância a ser obtida, que separa  $v$  e  $w$ . Nesse caso,  $a = \frac{\pi}{2} - \phi_1$  e  $b = \frac{\pi}{2} - \phi_2$ ,  $C$  é separação de longitude  $\gamma_2 - \gamma_1$  e  $c = \frac{d}{R}$ , onde  $R$  é o raio da esfera. Então, substituindo na eq(2.1):

$$\text{hav}\left(\frac{d}{R}\right) = \text{hav}(\phi_2 - \phi_1) + \sin\left(\frac{\pi}{2} - \phi_1\right) \sin\left(\frac{\pi}{2} - \phi_2\right) \text{hav}(\gamma_2 - \gamma_1) \quad (2.2)$$

Fazendo uso das relações trigonométricas ( $\sin(a - b) = \sin(a) \cos(b) - \sin(b) \cos(a)$ ):

$$\sin\left(\frac{\pi}{2} - \phi_t\right) = \sin\left(\frac{\pi}{2}\right) \cos(\phi_t) - \sin(\phi_t) \cos\left(\frac{\pi}{2}\right) = \cos(\phi_t) \quad (2.3)$$

Portanto:

$$\text{hav}\left(\frac{d}{R}\right) = \text{hav}(\phi_2 - \phi_1) + \cos(\phi_1) \cos(\phi_2) \text{hav}(\gamma_2 - \gamma_1) \quad (2.4)$$

A eq(2.4) obtida é a famosa Fórmula de Haversine. Entretanto, a fórmula do modo que está não fornece diretamente a distância  $d$ , e, portanto, apresenta-se apenas como uma relação de igualdade. Para poder obter o valor de  $d$  é necessário que se utilize a função Haversine na equação eq(2.4):

$$\sin^2\left(\frac{d}{2R}\right) = \sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + \cos(\phi_1) \cos(\phi_2) \sin^2\left(\frac{\gamma_2 - \gamma_1}{2}\right) \quad (2.5)$$

$$\sin\left(\frac{d}{2R}\right) = \sqrt{\sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + \cos(\phi_1) \cos(\phi_2) \sin^2\left(\frac{\gamma_2 - \gamma_1}{2}\right)} \quad (2.6)$$

$$\frac{d}{2R} = \arcsin\left(\sqrt{\sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + \cos(\phi_1) \cos(\phi_2) \sin^2\left(\frac{\gamma_2 - \gamma_1}{2}\right)}\right) \quad (2.7)$$

Finalmente:

$$d = 2R \arcsin\left(\sqrt{\sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + \cos(\phi_1) \cos(\phi_2) \sin^2\left(\frac{\gamma_2 - \gamma_1}{2}\right)}\right) \quad (2.8)$$

A equação (2.8) obtida permite o cálculo de  $d$  diretamente, ou seja, a distância entre  $v$  e  $w$ . Observa-se que os valores de  $\gamma_1$ ,  $\gamma_2$ ,  $\phi_1$  e  $\phi_2$  se apresentam em graus decimais, e devem ser convertidos preferencialmente para radianos. Além disso, deve-se escolher cuidadosamente o tamanho de  $R$ , pois como foi descrito acima, o raio da Terra pode sofrer uma variação de aproximadamente 22km, ou seja, entre 6356.752 km nos polos e 6378.137 na linha do Equador. O valor mais comum aceito e usado é  $R = 6371km$ .

Na situação em que os dois pontos são próximos, a fórmula de Haversine não possui problemas de arredondamento devido ao uso dos senos para o cálculo. Ademais, a equação é bem condicionada, isto é, uma pequena perturbação nos valores de Latitude e Longitude implica em um pequeno erro proporcional no valor final da distância. Outro benefício da fórmula é o balanço entre precisão e velocidade para o processamento, ao contrário de outras alternativas que priorizam precisão mas apresentam velocidade menor porque calculam o resultado através de métodos iterativos e não diretamente. Entretanto, na situação em que os pontos são antipodais (polo norte e polo sul, por exemplo), erros numéricos relativos são grandes. Isso acontece porque o termo dentro da raiz da equação (2.8) se aproxima do valor 1, e, portanto não é aconselhado usar a fórmula nessa situação. Além disso, pelo fato de supor que a Terra é uma esfera, somado as diferenças no seu raio de curvatura, a fórmula de Haversine pode acumular um erro de até 0.5% de sua medida.

## 2.3 Análise Numérica

### 2.3.1 Inverso da Potência das Distâncias

O método do Inverso da Potência das Distâncias (IDW, abreviado do inglês *Inverse Distance Weighting*) calcula o valor de certa propriedade (ou fenômeno) de um novo local

desejado. Esse procedimento de interpolação determinístico, a partir de um conjunto de dados onde cada dado possui o valor de certa propriedade e o valor da distância em relação ao lugar desejado, calcula-se através de uma média ponderada o valor do fenômeno nesse novo local. Uma definição formal desse fenômeno é dada por:

$$u(x) : x \rightarrow \mathbb{R}, x \in \mathbb{R}^n \subset D$$

onde:  $D$  é a região de estudo. (No caso de  $n = 2$ , um plano, por exemplo)

Observe que para geometria esférica, mencionada na seção 2.2.1,  $x \in \mathbb{R}^2$ , pois  $x = (\phi, \lambda)$ . Seja um conjunto finito de pontos conhecidos no espaço,  $[(x_1, u(x_1)), (x_2, u(x_2)), \dots, (x_n, u(x_n))]$  em  $D$ . Obtêm-se o valor do fenômeno  $u(x_{n+1})$  desconhecido de um ponto desejado  $x_{n+1}$  com a utilização de uma média ponderada ( $w_i(x)$ ) dos fenômenos  $u(x_i)$  dados pelo inverso das suas respectivas distâncias ao quadrado. Sua fórmula é dada por:

$$u(x) = \frac{\sum_{i=1}^n w_i(x) u_i}{\sum_{i=1}^n w_i(x)}$$

onde:

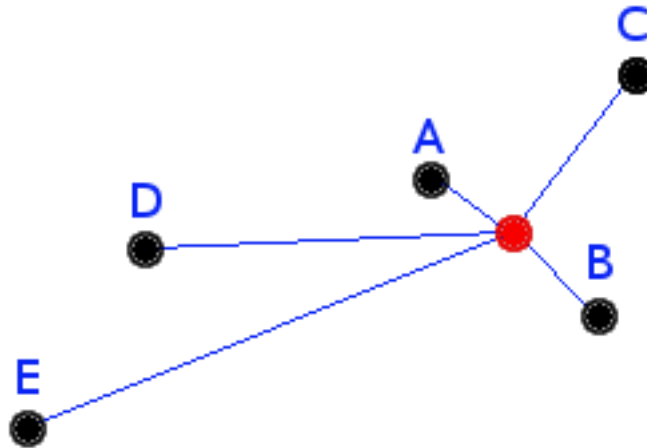
$$w_i(x) = \frac{1}{d(x, x_i)^p}$$

$d(x, x_i)$ : Distância entre  $x$  e  $x_i$ .

$p$ : Parâmetro da potência.

O método acima faz a suposição de que pontos que estão mais próximos da localização a ser calculada contribui mais para o valor final obtido, como ilustra a figura 2.6, porém não se leva em consideração a estrutura espacial. Conforme a distância vai aumentando, a relevância das medidas dos pontos mais afastados para o valor final diminuem. Desse modo, o parâmetro real  $p$  simboliza a que taxa ocorre essa diminuição, servindo também como controle do tamanho da vizinhança. Portanto, o crescimento de  $p$  significa a exclusão ou menor relevância de pontos mais afastados, enquanto que se utilizarmos o menor valor, que é  $p = 1$ , o peso é apenas o inverso da distância medida, permitindo assim a inclusão de pontos mais afastados no cálculo do valor desconhecido. Tipicamente, usa-se  $p = 2$ , e nessa forma, o modelo se chama inverso do quadrado das distâncias.

Apesar do erro cometido pelo modelo geral estar diretamente relacionado com o valor  $p$  e ao conjunto de dados, o inverso do quadrado das distâncias é uma boa solução quando é desejado que as medidas mais próximas do ponto a ser obtido tenham um peso maior no resultado final da medida calculada, perdendo pouca precisão se comparado a outros métodos Jin Li [2008].



**Figura 2.6:** Ilustração do IDW no plano. Pontos A e B tem maior influência no resultado de  $u(x)$  do ponto vermelho do que os pontos E e D.

Finalmente, o cálculo da distância  $d(x, x_i)$  depende do contexto da região de estudo e do valor da dimensão espacial. No caso das coordenadas geográficas, que são utilizadas nesse trabalho, o cálculo dessa distância deve ser feito de acordo com a técnica apresentada na seção 2.2.4.

## 2.4 Protocolo de Transferência de Arquivos

O Protocolo de Transferência de Arquivos (FTP, do inglês *File Transfer Protocol*) é uma forma rápida de transferência de arquivos na Internet, permitindo receber ou enviar arquivos dado um endereço no navegador ou através de um software. O local no qual os arquivos ficam hospedados é chamado de servidor, enquanto as operações de transferência são realizadas por um computador cliente. O utilizador faz uma solicitação, fornecendo o nome ou endereço IP, assim como também o seu usuário e senha. Se todos os dados estiverem corretos, a conexão entre o cliente e servidor é criada, permitindo o acesso e transferência de arquivos. Tanto o aplicativo como o *script* gerador de dados, que serão apresentados mais adiante, utilizam o FTP, mas com bibliotecas diferentes. O *script* faz uso da implementação da biblioteca interna e nativa *ftplib* para a realização da conexão, mas o aplicativo faz uso de bibliotecas externas. Outra diferença está no uso da biblioteca nas suas funções: o *script* envia os dados ao servidor, enquanto o aplicativo faz *download* dos mesmos.

Como descrito no capítulo introdutório, um dos vários objetivos era o cálculo do índice no local do usuário, exibindo a qualidade de ar, o poluente responsável pela qualidade do ar (em figura), o local do usuário, e eventuais sugestões de prevenção. Para finalizar o capítulo, apresenta-se como os conceitos apresentados anteriormente estão relacionados com o aplicativo e esse objetivo em particular. O método IDW apresentado na seção 2.3.1 é utilizado para o cálculo do índice de qualidade, e consequentemente, a qualidade do ar e o seu poluente responsável, fazendo uso de definições básicas e fórmulas das ciências geodésicas na seção 2.2 para a representação de locais e na obtenção das distâncias na superfície da Terra. Além disso, o GPS, que também faz parte da Geodésia, é responsável por fornecer a localização do usuário. Finalmente, a explicação da origem e dos sintomas dos poluentes na seção 2.1.2 são usados para mostrar as possíveis recomendações de prevenção.

A obtenção e tratamento de dados, e a interação do usuário no aplicativo de forma fácil, que foram outros elementos citados nos objetivo, serão abordados nos próximos capítulos.



# Capítulo 3

## Desenvolvimento do aplicativo

### 3.1 Android

Android é um sistema operacional (SO) desenvolvido pela empresa de tecnologia Google e projetado para dispositivos móveis (*smartphones* e *tablets*). Foi lançado em 28 de Setembro de 2008, e o sistema operacional é baseado no núcleo (*kernel*, em inglês) do Linux e seu código fonte é lançado pela Google sob licença de código aberto. O seu baixo custo e a alta capacidade de adaptação foram as principais razões para que as empresas fabricantes de dispositivos móveis adotassem o SO. Esse fato contribuiu tanto para a popularização do SO, que em 2015, a *Nielsen Company* estimou que mais da metade dos americanos tivessem em seu *smarthphone* o sistema Android Nie [2015]. Apesar de o núcleo ser escrito na linguagem de programação C, os aplicativos são escritos em Java, usando o kit de desenvolvimento do Android (SDK). A instalação de programas é feita através de um arquivo no formato *APK*, ou através de um programa de loja de aplicativos virtual, como a *Play Store*, que é a loja primária e pertencente a Google.

#### 3.1.1 Interface de Programação de Aplicações

Naturalmente, desde o lançamento do SO pelo Google, o Android passou por diversas atualizações. Essas atualizações refletem diretamente no valor inteiro e seu respectivo codinome, marcado como o *level* da Interface de Programação de Aplicações, ou simplesmente API (abreviado do inglês *Application Programming Interface*). A API é um importante *framework* que fornece a comunicação com funcionalidades já construídas dentro do sistema ao programador. Além disso, o sistema foi projetado para que mesmo quando alguns recursos são trocados por funções novas, a API mantenha antigas implementações para não prejudicar os programas mais antigos. A tabela abaixo mostra a porcentagem dos dispositivos em relação aos codinomes da APIs, ordenados pela versão do sistema<sup>1</sup>:

---

<sup>1</sup>Informações retirada do site <https://developer.android.com/>

**Tabela 3.1:** *Dados da distribuição dos dispositivos por codinome.*

Versão	Api(s)	Codinome	Dispositivos (%)
2.2	8	Froyo	0,1
2.3.3 a 2.3.7	10	Gingerbread	1,5
4.0.3 a 4.04	15	Ice Cream Sandwich	1,4
4.1 a 4.3	16,17,18	Jelly Bean	15,6
4.4	19	Kit Kat	27,7
5.0 a 5.1	21,22	Lollipop	35
6.0	23	Marshmallow	18,7

Obviamente, a tabela 3.1 mostra que menos de 20% dos dispositivos utilizam versões do sistema mais atual, que são as APIs de codinome *Marshmallow*. Desse modo, é evidente que antes de construir o aplicativo em si, é necessário estabelecer quais dispositivos ou a necessidade de uso de ferramentas mais complexas. Claramente, revela-se que o uso de recursos mais avançados significa um alcance menor no número de dispositivos que podem rodar um aplicativo. Assim, tendo em vista essas informações, o aplicativo construído nesse trabalho fará uso da API 16, pois se deseja que o programa possua uma alta cobertura de dispositivos, já que como foi explicado anteriormente, sistemas que possuem um valor maior do que 16 serão compatíveis com o programa. Portanto, significa que o programa executará nos sistemas androids com os codinomes: *Jelly Bean*, *Kit Kat*, *Lollipop*, *Marshmallow*, ou seja, poderá ser instalado em ao menos 96% dos dispositivos com sistema Android. O motivo técnico de escolher a API 16 é por conta de uma maior facilidade e simplicidade para exibir as informações relevantes para interface do usuário, e por conta desse fato, o aplicativo certamente não será compatível com versões de codinome *Gingerbread* ou anteriores.

### 3.1.2 Java

Java é uma linguagem orientada a objetos, e como dito anteriormente, usada para a escrita de aplicativos para o sistema Android. Dessa maneira, não basta especificar apenas a SDK utilizada, mas também o kit de desenvolvimento Java (JDK, abreviado do inglês *Java Development Kit*). O aplicativo faz uso de uma biblioteca externa, chamada de *Apache Commons Net 3.5*. Funcional apenas para as versões 1.6 ou mais do Java, essa biblioteca é chave para o download de informações atualizadas, por fazer uso do protocolo de transferências de arquivos (explicado em mais detalhes nas seções abaixo). A versão utilizada é a 1.8, mais especificamente, como 1.8.0\_101

### 3.1.3 Android Studio

Lançado em Dezembro de 2014 pelo Google, o Android Studio é o ambiente de desenvolvimento integrado (IDE) de multi plataforma oficial para o desenvolvimento de aplicativos e também utilizado para o desenvolvimento do aplicativo citado neste trabalho. O ambiente conta tanto com ferramentas avançadas de auto-completamento como de refatoração para a escrita do código em Java. A criação do *layout* poderia ser trabalhoso e complicado, por serem gerados arquivos na forma de *eXtensible Markup Language* (XML), mas a presença do editor gráfico do *layout* torna a adição de elementos dinâmica e fácil. Além disso, diferente do plugin *Android Development Tools* (ADT) feito para a IDE *Eclipse* que usam o sistema de automação de compilação *Apache Maven*, as dependências são carregadas com a utilização de uma ferramenta semelhante chamada *Gradle*. Nesse sistema de automação

de compilação, integrado ao Android Studio, apesar de usar os mesmos conceitos *Apache Maven*, o arquivo de dependências não é um XML, mas sim em uma linguagem baseada em *Apache Groovy*. A maior simplicidade na instalação e configuração, mais o fato do plugin ADT ter sido descontinuado para o uso no Eclipse desde Junho de 2015 ADT [2015], estão entre os principais motivos para o uso do Android Studio.

## 3.2 *Script* gerador

Antes de ilustrar brevemente o funcionamento e interação do aplicativo, é importante descrever quais foram os mecanismos utilizados para geração do arquivo de dados. Apesar de não estar incluído no programa final em si, o *script* faz a geração e formatação de dados, finalizando ao salvar o arquivo final em um servidor web. Dessa maneira, o aplicativo apenas importa direto do servidor um único arquivo que contém todos os dados. Essas informações se apresentam em várias páginas da web, ou seja, em um arquivo HTML por estação de monitoramento. Porém, para fazer o acesso a cada página, necessita-se não só preencher um formulário de login para ter acesso ao QUALAR, como também o preenchimento de informações para visualização por estação. Assim, a dificuldade resume-se em realizar de forma automatizada, navegar nas diferentes páginas, extraíndo os índices dos respectivos poluentes e a marcação de dia e hora para cada estação. Em uma primeira tentativa, optou-se por implementar essa tarefa dentro do próprio aplicativo, porém a quantidade de bibliotecas externas, somado ao fato do tempo longo necessário para concluir as operações (em torno de 5 minutos) como ao razoável uso dos recursos do sistema, tornou a confecção praticamente inviável. Então, como descrito antes, a solução foi implementar um *script shell*, que executa dois módulos em Python chamados de *fetchData* e *ftp*, em uma máquina Linux devidamente configurada para a realização dessa tarefa. Primeiramente, abordam-se as ferramentas ou bibliotecas externas e o seu papel, para depois descrever essas duas rotinas de forma geral. O *script* gerador pode ser baixado do link: <https://github.com/dxhunter/monQASP2016/blob/master/executaveis/scriptgerador/scriptgerador.zip>.

### 3.2.1 Selenium Web Driver

O Selenium Web Driver (SEL) é uma ferramenta originalmente desenvolvida para o auxílio de testes automatizados, mas que ficou conhecida também por ser uma ferramenta para a escrita de programas que realizam navegação automatizada. Suas bibliotecas permitem simular um navegador web de forma mais simples e eficiente, com suporte até mesmo de páginas que executam rotinas em javascript. Algumas de suas ações incluem: abrir uma janela, detectar e apertar botões, detectar e preencher formulários, selecionar valores de um *dropbox*, trocar de janela, detectar alertas e fechar o navegador e suas janelas restantes. Essa biblioteca, em um primeiro momento, auxilia o módulo *fetchData* a acessar o site e realizar o login no sistema, acessando a *Uniform Resource Locator* (URL) representado por “Dados horários por estação” em “Consulta” uma vez dentro do sistema. Depois, para cada estação, um *pop-up* é gerado ao ter o botão “Consultar” clicado, onde a URL dessa janela é criada e contém as informações pertinentes. Obviamente, uma vez salva as informações do HTML da URL obtida, esse *pop-up* é fechado, para repetir o processo e dar lugar a URL da próxima estação. Naturalmente, falhas podem ocorrer, e a detecção dos alertas é indispensável no tratamento desses erros: seja pular para próxima estação, ou tentar novamente do começo.

### 3.2.2 BeautifulSoup

Em princípio, extrair os dados de um documento HTML pode parecer simples por se tratar de uma linguagem de marcação. Entretanto, o tamanho do documento, somado a quantidade de *tags* de marcação dificulta a obtenção de dados que estão entre essas. Assim, uma primeira tentativa de extrair as informações, é a partir do uso de expressões regulares (abreviado em RegEx, no inglês), identificando os padrões de caracteres que contém a informação. Porém, uma expressão regular construída para marcações HTML se torna longa e complicada demais rapidamente, tornando essa implementação complexa. A outra e utilizada alternativa, é o *BeautifulSoup* (BEA), um *parser* de HTML que permite de forma simples e flexível a obtenção do conteúdo dentro das *tags* de marcação. No *script*, a biblioteca é utilizada no arquivo HTML de cada estação para o fácil filtro da tabela de dados, extraindo o valor do índice de cada poluente. Não só o código é mais legível, como também é menor se comparado ao uso de expressões regulares.

### 3.2.3 Python

Uma vez descrita as ferramentas utilizadas, finalmente é possível descrever o funcionamento em alto nível dos dois módulos.

#### fetchData

Destaca-se nesse módulo a forma de armazenamento das estações válidas que são as estações apresentadas na tabela 2.2 na seção 2.1. Utiliza-se um dicionário ordenado, onde a chave representa a palavra utilizada para o acesso de cada estação no aplicativo, e o elemento se trata do valor inteiro utilizado na *checkbox* da estação no HTML de consulta. Em outras palavras, define-se para cada estação da RMSP, seu “nome chave” e respectivo valor de consulta, onde esse último é utilizado para acessar página da estação com os dados. Mais importante ainda, é manter a ordem na inserção a medida que cada par chave-valor é inserido, pois otimiza a performance do aplicativo quando está criando ou manipulando as estações a partir do arquivo final gerado por esse módulo.

Uma vez definido esse conjunto chave-valor, o *script* prossegue para a fase de navegação automática, utilizando a ferramenta *Selenium Web Driver*, descrita na seção 3.2.1, com o objetivo de acessar a página com as medidas. Dentro da página, armazena-se o HTML na memória, para que o mesmo seja analisado pelo *Beautiful Soup*, descrito na seção 3.2.2. Completa a análise, como os índices da qualidade do ar se encontram em *tags* do tipo `<th>...</th>`, seleciona-se todas as *tags* com esse nome, e filtrando através da posição dos dados, a fim de obter as informações pertinentes da estação: os poluentes medidos e seus respectivos índices. Esse processo, começando desde a navegação automática, até a filtragem de informações, é realizado para cada estação, ou seja, para cada chave do conjunto chave-valor definido. Finalmente, observa-se que os dados são escritos em um arquivo temporário e depois são copiados para um arquivo definitivo porque o funcionamento do sistema pode ser interrompido para a atualização das páginas com as novas informações e, portanto, o arquivo de dados ficará incompleto e possivelmente fora da formatação esperada. Nesse caso, o aplicativo irá exibir informações inconsistentes, e por isso, é preferível não sobrescrever os dados da última atualização caso a leitura mais nova de dados falhe.

## ftp

Desde que exista conexão com a internet, o segundo módulo chamado de ftp transfere o arquivo gerado pelo primeiro módulo para um servidor, pelo protocolo que foi descrito conceitualmente na seção 2.4.

### 3.2.4 X Virtual Framebuffer

Apenas com as ferramentas apresentadas em 3.2.1 e 3.2.2, já é possível executar manualmente o *script* gerador, descrito na seção 3.2, que gera o arquivo de dados. Todavia, para tornar a execução do *script* de forma automática e periódica, é necessário ainda fazer uso do chamado *Xvfb* ou *X Virtual Framebuffer*. Como o próprio nome em inglês sugere, o *Xvfb* é um display virtual que segue os protocolos X11, que faz as operações gráficas na memória em uma janela comum criada, mas sem mostrar a tela diretamente para o usuário. O motivo pelo qual é necessário o *Xvfb*, é por conta do agendador de tarefas *Cron* (mais detalhes abaixo), que necessita de um display ao executar o *script*. Uma vez fornecido o display virtual e a tarefa iniciada, todas as ações relacionadas ao navegador na subseção do Selenium Web Driver são realizadas, mas sem mostrar o navegador ao usuário.

### 3.2.5 Cron

O *Cron* é um programa que executa tarefas agendadas, e está presente nos sistemas *Unix*. O agendamento de tarefas é feito por um programa, chamado de *crontab*, editando o arquivo onde estão especificados os comandos a serem executados. Mostra-se abaixo a sintaxe que deve ser seguida para a execução de um comando:

```
1 mm hh dd MM ss meuscript
```

mm = minuto (de 0 até 59)

hh = hora (de 0 até 23)

dd = dia (de 1 até 31)

MM = mes (de 1 até 12)

ss = dia da semana (de 0 até 6)

Assim, especificando alguns dos valores acima e o *script*, a execução acontece em um tempo periódico determinado no arquivo.



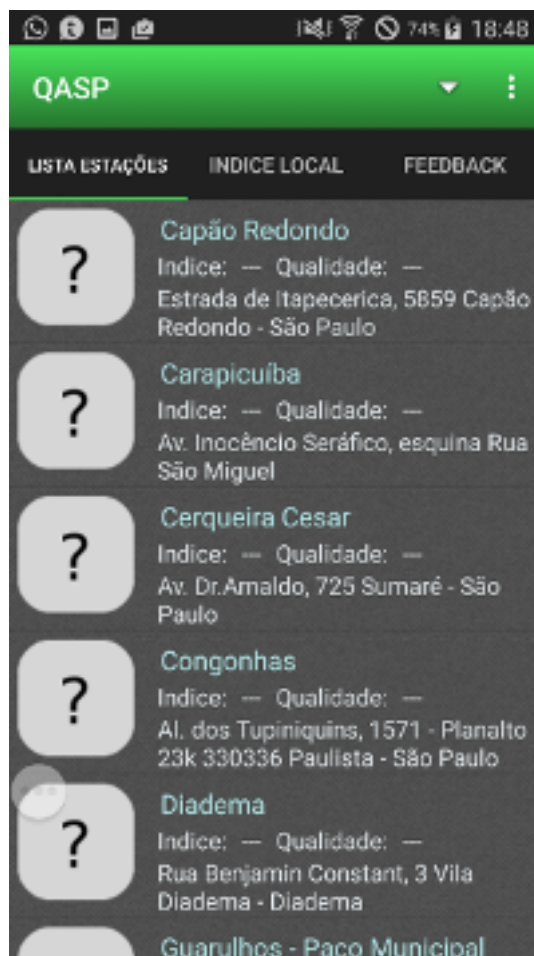
## Capítulo 4

# Aplicativo QASP

Concluído a descrição de como e quais ferramentas foram essenciais para o funcionamento, e também da explicação do módulo, agora é possível demonstrar a navegação e algumas configurações importantes do programa. O aplicativo, batizado de Qualidade de Ar de São Paulo, ou simplesmente abreviado para QASP, é o resultado final gerado por esse trabalho. O aplicativo pode ser baixado do link: <https://www.linux.ime.usp.br/~dmbuoro/mac0499/files/QASP.apk>.

### 4.1 Navegação Principal

A tela inicial, como mostra a figura 4.1, é composta pela famosa *Action bar*, que possui a cor de fundo verde, *tabs* com diferentes funções (Lista Estações, Índice Local, Feedback), e por fim o conteúdo a ser exibido. A *tab* sublinhada “Lista Estações” é iniciada por padrão, exibindo para cada estação as informações, que ainda necessitam serem definidas. A barra do *menu*, aonde ficam informações como a porcentagem de carga da bateria e a hora atual, também é exibida com outra cor de fundo se a versão do sistema é pelo menos a *Lollipop*.



**Figura 4.1:** Tela inicial ao abrir o QASP.

A *Action bar* do aplicativo possui dois ícones: uma seta para baixo e três pontos verticais, que representam a atualização de dados e um menu de preferências respectivamente (descrição mais abaixo). Ao apertar o ícone de seta para baixo, o QASP faz download dos dados mais atuais, fazendo uso do protocolo apresentado na seção 2.4, e atualizando as informações das estações. A figura 4.2 mostra um exemplo do resultado dessa ação.

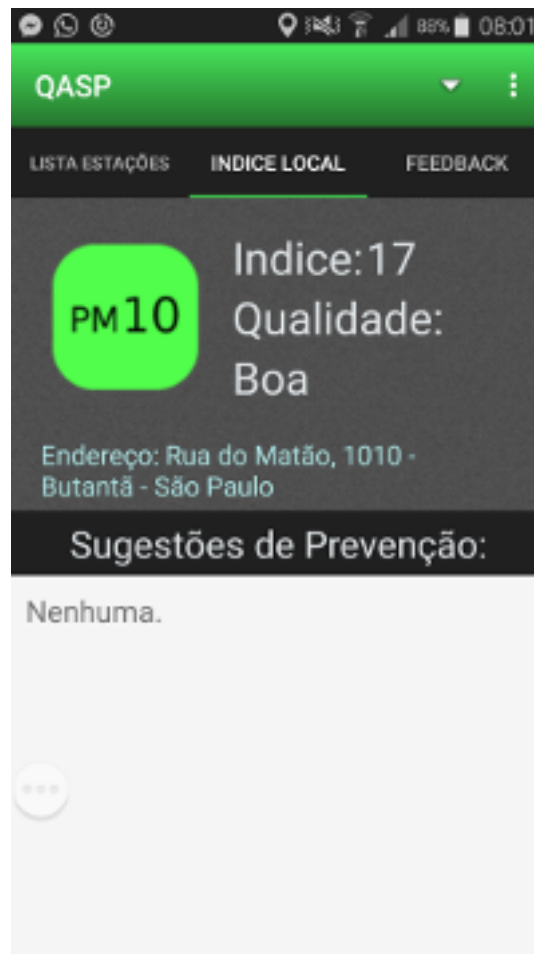
Para cada estação é mostrado seu nome, endereço, índice de qualidade do ar, e um quadrado com bordas arredondadas que representa o poluente responsável pelo índice, com a cor de fundo em sintonia com a qualidade do ar. Como para cada poluente há a possibilidade de cinco tipos de qualidade de ar, isso totaliza 30 figuras para atender cada caso esperado. Entretanto, a figura 4.2 mostra também dois casos a mais: o “?” e “OFF”. A imagem de “?”, como escrito anteriormente, é mostrada por padrão quando ainda não houve a atualização de dados, e também caso a estação, mesmo em funcionamento, não possua dados de poluente algum em sua página após a sincronia das informações. Por outro lado, o “OFF” se refere ao estado de funcionamento da página de exibição, que não está operando. Isso não implica que a estação em si não esteja operando, mas que a página usada para consulta no sistema QUALAR não está disponível.





**Figura 4.2:** Exemplo do resultado ao pressionar o botão de seta para baixo, com as estações devidamente atualizadas.

A atualização das estações mais a localização obtida pelo GPS permite o cálculo do índice local, que pode ser feito ao mover-se para a aba “Índice Local”. Entretanto, dependendo do local, a obtenção das coordenadas pode levar algum tempo, e uma janela de erro é exibida caso essas ainda não tenham sido atualizadas. Caso contrário, calcula-se o índice de qualidade do ar no local do usuário, para cada poluente, pelo método apresentado na seção 2.3.1, utilizando  $p = 2$ , e mostrando o pior entre os 6 valores. O resultado é construído rapidamente, isso porque é usada uma matriz onde as linhas são as estações e as colunas os poluentes, e como consequência o acesso a essas informações são feitas de forma veloz, apesar do uso maior da memória do sistema. A figura 4.3, mostra a exibição do resultado de forma parecida com a listagem de estações, mas incluindo o endereço do local dado pelo GPS, e possíveis recomendações para preservar a saúde do usuário. As sugestões de prevenção, baseadas nos sintomas dos poluentes abordados na seção 2.1.2, são influenciadas por três parâmetros: a qualidade do ar, o poluente responsável e o perfil de usuário selecionado (explicado mais adiante na seção 4.2). Obviamente, recomendações mais drásticas são exibidas conforme a qualidade do ar diminui. Em casos onde há apenas um pequeno desequilíbrio, onde a qualidade do ar é registrada como “mediana”, evitar a prática de exercícios intensos por um longo período de tempo ou a realização dos exercícios em ambientes fechados são algumas das possíveis recomendações. Em contrapartida, evitar qualquer tipo de exercício físico é a recomendação mais comum quando a qualidade do ar está “péssima”.



**Figura 4.3:** Tela com o resultado do índice da qualidade do ar no local do usuário.

Finalmente, a última aba, nomeada de “Feedback”, permite ao usuário avaliar as funcionalidades do aplicativo. Exige-se uma classificação (Ruim, OK, Bom) e um texto opcional para escrever os motivos, como mostra a figura 4.4. Como o aplicativo necessita de constante atualização pelo usuário, é fundamental esse tipo de resposta na eventualidade de uma modificação no sistema de origem dos dados. O envio é feito através do cliente de e-mail de preferência do usuário.



**Figura 4.4:** Tela para o envio de feedbacks

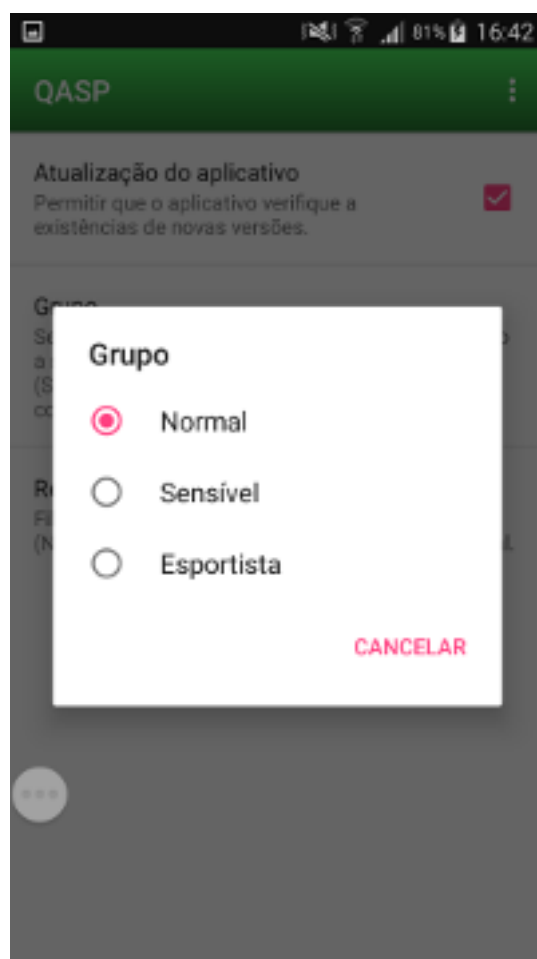
## 4.2 Preferências

Acessada através da figura dos três pontos verticais na *Action Bar*, o usuário que entrar em “Configurações” poderá escolher opções que se encaixem melhor a sua experiência, como mostra a figura 4.5. A mais importante, “Grupo”, modifica as recomendações de preservação de saúde se baseando no perfil de suas atividades ou no quadro de saúde do Usuário. Essa opção de personalização, como descrito na seção 2.1.4, permite atender melhor inclusive usuários com perfis diferentes, e possui as opções: “Normal”, “Sensível” e “Esportista”, como mostra a figura 4.6. Um usuário que tenha selecionado “Sensível”, por exemplo, poderá ver sugestões de preservação da saúde quando, dependendo do poluente, a qualidade do ar estiver “Mediana”, o que não ocorreria para alguém que use o perfil “Normal”.

Outra opção, a “Região de Exibição”, possibilita apenas filtrar estações que são mostradas na aba “Lista Estações”. Essa opção é puramente estética, não interferindo no cálculo do índice local, isto é, as estações que não aparecerem na lista ainda são usadas para o resultado final. Assim, para não ficar tão grande a lista de exibição, o usuário pode remover estações fora da RMSP, ou ainda exibir somente as estações da cidade de São Paulo.



**Figura 4.5:** Tela que mostra as opções existentes em “Configurações”



**Figura 4.6:** *Seleção do grupo (perfil de saúde), note que é possível apenas uma escolha das três.*



# Capítulo 5

## Conclusões

A proposta deste trabalho foi executada com sucesso, apesar de empecilhos não esperados. O mais pertinente, que quase impediu a conclusão desejada, foi a dificuldade de obtenção de dados para o funcionamento do aplicativo. Tais informações, como são de interesse público, deveriam ser disponibilizadas de forma mais fácil e intuitiva, para eliminar a necessidade de abertura de dezenas de páginas. Dessa forma, a atualização poderia ser realizada dentro do próprio aplicativo, eliminando tanto a utilização do servidor remoto como o *script* gerador separado. Felizmente, dada a existência da ferramenta de navegação automatizada e sua ótima documentação, foi possível contornar esse problema.

A literatura científica sobre as consequências na saúde devido à poluição teve seus aspectos positivos e negativos no desenvolvimento do trabalho. A quantidade de relatórios, notícias e até mesmo artigos contribuíram bastante para a riqueza de exposição dos dados e conceitos, porém a literatura é insuficiente quando se trata de informações mais específicas a um determinado perfil de pessoas.

Finalmente, vale ressaltar que a utilização dos métodos apresentados relacionado ao uso das distâncias não são absolutos. Foram utilizados dentro de um contexto, isto é, visando desempenho para o aplicativo. Técnicas que são mais precisas para o cálculo da distância ou levam em consideração outras propriedades (a altitude, por exemplo) para a obtenção da qualidade local, ainda podem ser uma alternativa dependendo da aplicação.

### 5.1 Trabalhos Futuros

Há espaço para melhorar ou incluir novas funcionalidades para a interface do usuário ou nos métodos utilizados para o cálculo da qualidade do ar localmente. Algumas ideias incluem:

- Levar em consideração a diferença de altitude entre o usuário e as estações.
- Implementar a funcionalidade de incluir estações para o cálculo do índice local até certo raio.
- Adicionar outras opções de filtro ao listar as estações.
- Mostrar sugestões de prevenção de outros poluentes a depender de seu índice, e não apenas o pior caso.





# Referências Bibliográficas

- ADT(2015)** An update on eclipse android developer tools, jun 2015. URL <http://android-developers.blogspot.com.br/2015/06/an-update-on-eclipse-android-developer.html>. Citado na pág. 21
- BEA()** URL <https://www.crummy.com/software/BeautifulSoup/>. Citado na pág. 22
- EEA(2011)** Ozone and particulates most serious air quality problems in europe, Nov 2011. URL <http://www.eea.europa.eu/highlights/ozone-and-particulates-most-serious>. Citado na pág. 1
- EXA(2014)** Poluição do ar é mais fatal que acidente de trânsito em sp, Apr 2014. URL <http://exame.abril.com.br/mundo/poluicao-do-ar-mata-mais-que-acidente-de-transito-em-sp/>. Citado na pág. i, iii
- FOL(2016)** Campus da usp é área mais poluída de sp pelo segundo ano consecutivo, May 2016. URL <http://www1.folha.uol.com.br/cotidiano/2016/05/1773232-campus-da-usp-e-area-mais-poluida-de-sp-pelo-segundo-ano-consecutivo.shtml>. Citado na pág. 7
- GPL(2002)** 2002. URL <https://www.gnu.org/licenses/gpl-howto.pt-br.html>. Citado na pág. 2
- ISS(2014)** Monitoramento da qualidade do ar no brasil, 2014. URL <http://www.saudeesustentabilidade.org.br/site/wp-content/uploads/2014/07/Monitoramento-da-Qualidade-do-Ar-no-Brasil-2014.pdf>. Citado na pág. 8
- Nie(2015)** Smartphone owners are as diverse as their devices, May 2015. URL <http://www.nielsen.com/us/en/insights/news/2015/smartphone-owners-are-as-diverse-as-their-devices.html>. Citado na pág. 19
- ODM()** URL <http://opendustmap.com>. Citado na pág. 9
- OMS(2014)** Air pollution estimates, Mar 2014. URL [http://www.who.int/phe/health\\_topics/outdoorair/databases/FINAL\\_HAP\\_AAP\\_BoD\\_24March2014.pdf](http://www.who.int/phe/health_topics/outdoorair/databases/FINAL_HAP_AAP_BoD_24March2014.pdf). Citado na pág. 1
- QAS()** URL <https://www.linux.ime.usp.br/~dmbuoro/mac0499/files/QASP.apk>. Citado na pág.
- QUA(2009)** Set 2009. URL <http://ar.cetesb.sp.gov.br/qualar/>. Citado na pág. 9
- SEL()** URL <http://www.seleniumhq.org/projects/webdriver/>. Citado na pág. 21
- SGE()** URL <https://www.linux.ime.usp.br/~dmbuoro/mac0499/files/scriptgerador.zip>. Citado na pág.

**Jin Li(2008)** Andrew D. Heap Jin Li. A review of spatial interpolation methods for environmental scientists, 2008. URL [http://www.ga.gov.au/webtemp/image\\_cache/GA12526.pdf](http://www.ga.gov.au/webtemp/image_cache/GA12526.pdf). Citado na pág. 16

**Labrador(2015)** Virgil Labrador. Satellite communication, Feb 2015. URL <https://global.britannica.com/technology/satellite-communication>. Citado na pág. 11