

Jean Souza – jeancfsouza@gmail.com

1 – INTRODUÇÃO

1.1 – Motivação

Com o grande número de indústrias, veículos automotores e outras diversas fontes de gases poluentes, é necessário que exista um controle da qualidade do ar, e isso é regulamentado pela resolução CONAMA nº005, de 15 de junho de 1989, que institui o Programa Nacional de Controle de Qualidade do Ar – PRONAR, tanto para atender a padrões estabelecidos quanto para melhorar a qualidade do ar visando uma melhoria na qualidade de vida da população (CONAMA, 2013a).

O CONAMA teve suas competências atribuídas pela lei nº 6.938 / 1981 e ele estabelece desde então normas para controle de emissão de poluentes do ar por fontes fixas e móveis, e uma dessas competências é estabelecer padrões nacionais para controle de poluição de veículos automotores (PEREIRA, 2013).

Além do que já foi citado, existem diversas leis que regulamentam e controlam o que diz respeito à qualidade do ar, leis que estabelecem limites de poluentes e definem ainda sanções penais e administrativas para crimes de poluição (PEREIRA, 2013).

A má qualidade do ar, além de prejudicar a qualidade de vida, influencia na ocorrência de doenças respiratórias como gripe e pneumonia, principalmente em idosos, e esse é um fato muito preocupante, tendo em vista a quantidade de gases poluentes presentes em cidades como São Paulo, por isso a medição da qualidade do ar é muito importante (MARTINS, 2002). Os fatores que causam o aumento nos problemas respiratórios, problemas como os que foram citados, não são precisamente definidos, porém estudos realizados indicam que a poluição está associada com esses problemas, sendo que o grande uso de derivados de petróleo pela indústria e por veículos automotores é um fator significativo no aumento da poluição (CROCE, 1998).

Pode ser citado como exemplo o SO₂ (dióxido de enxofre), poluente que afeta principalmente pessoas que já possuem doenças respiratórias, que mesmo não ultrapassando a quantidade máxima estipulada para a efetiva qualidade do ar (CONAMA, 2013a), está associado à gripe e pneumonia em idosos (MARTINS, 2002).

Até mesmo mortes de idosos por problemas respiratórios podem estar associados com poluentes, poluentes como o PM10 (material particulado) e o CO (monóxido de carbono) (FREITAS, 2004).

Já foi dito nesse trabalho que grande parte dos poluentes presentes no ar são originados por veículos automotores e indústrias e por conta disso existem órgãos regulamentadores como o CONAMA, e com o passar do tempo vem surgindo cada vez mais tecnologias para o controle da emissão de poluentes (RANGEL, 2003), justamente por conta desses problemas, mas para que se possa saber se essas ações estão tendo sucesso, é necessário que haja a medição da qualidade do ar.

A já mencionada resolução Nº 5 do CONAMA ressalta a importância da medição da qualidade do ar, inclusive tratando da importância da criação de uma rede de medição para o devido acompanhamento e comparação com os níveis mínimos de qualidade do ar estabelecidos pelas resoluções e leis (CONAMA, 2013a).

A CETESB (Companhia Ambiental do Estado de São Paulo) possui uma rede de medição de qualidade do ar que cobre diversas cidades do estado, porém existem regiões que ainda não tem grande cobertura dessa rede. No vale do Paraíba não são todas as cidades que possuem o monitoramento da CETESB (CETESB, 2011), sendo que as cidades do Vale do Paraíba podem ser afetadas com a poluição da cidade de São Paulo, por conta do alcance que a poluição de São Paulo pode atingir e da direção dos ventos, além de que São José dos Campos possui um grande parque industrial (NASCIMENTO, 2006), e por conta disso uma medição da qualidade do ar mais abrangente é muito importante.

1.2 - Objetivos

Nas subseções a seguir serão apresentados os objetivos deste trabalho.

1.2.1 - Objetivo Geral

O objetivo deste trabalho é utilizar a Internet das Coisas para o desenvolvimento de um sistema de medição da qualidade do ar, enviar os resultados das medições para o *web service* Xively e para um *web service* RESTful a ser criado, e desenvolver uma aplicação *web* cliente para exibir os dados enviados para esse *web service*.

1.2.2 - Objetivos Específicos

Os objetivos específicos são:

- Desenvolver um dispositivo de medição da qualidade do ar que tenha acesso a uma rede cabeada para transmitir os dados coletados;
- Alocar o dispositivo em um local externo para a captação de dados para os testes e para registro dos dados da qualidade do ar;
- Enviar os dados para o *web service* Xively para que os usuários desse serviço possam utiliza-los;
- Desenvolver um *web service* RESTful para o envio dos dados, que ficarão salvos em um banco de dados NoSQL orientado a documentos chamado MongoDB;
- Desenvolver uma aplicação cliente *web* para exibição dos dados que serão obtidos do banco de dados MongoDB.

1.3 - Metodologia

O dispositivo de medição será composto por uma placa Arduino Ethernet e um sensor de qualidade do ar. O Arduino Ethernet possui um microcontrolador que será responsável pelo controle do processo de medição da qualidade do ar e já é preparado para comunicação com redes *ethernet* (ARDUINO, 2013a). O sensor de qualidade do ar é um dispositivo que será controlado pelo Arduino e coletará os dados, dados que serão úteis para a geração das informações a respeito da qualidade do ar.

Os dados serão coletados de uma área externa pelo dispositivo, que irá enviar os dados por meio de um cabo *ethernet* com troca de mensagens do protocolo HTTP para o *web service* Xively (XIVELY, 2013). O Xively é um *web service* de uso gratuito que utiliza a arquitetura RESTful (FIELDING, 2000), os usuários desse *web service* o alimentam com diversos dados provenientes de dispositivos como o que será construído nesse trabalho e esses dados são disponibilizados pelo próprio Xively para os usuários que desejarem consumi-los (XIVELY, 2013) (SOUZA, 2012).

Além de serem enviados para o Xively, os dados serão enviados para um *web service* RESTful a ser desenvolvido, sendo que esse *web service* será desenvolvido na linguagem Java utilizando a API Jersey, e os dados serão salvos em um banco de dados NoSQL orientado a documentos chamado MongoDB.

As medições serão realizadas durante XX dias, e os dados obtidos nessas medições serão enviados para o *web service* Xively, que já foi apresentado. Também será desenvolvida uma aplicação Java web para exibir os resultados.

1.4 - Organização do Trabalho

Este trabalho está dividido em cinco capítulos, que estão descritos abaixo:

- Capítulo 2 - Revisão bibliográfica: responsável por apresentar as tecnologias utilizadas para o desenvolvimento do dispositivo e apresentar alguns projetos já existentes na área.
- Capítulo 3 - Desenvolvimento: descreve o processo de desenvolvimento do dispositivo.
- Capítulo 4 – Resultados: responsável por apresentar os resultados obtidos com os testes.
- Capítulo 5 - Considerações finais: responsável por apresentar as considerações finais do trabalho desenvolvido.

2 – REVISÃO BIBLIOGRÁFICA

Esse capítulo vai tratar a respeito de assuntos relacionados ao dispositivo, apresentando as tecnologias utilizadas e alguns projetos já presentes nessa área.

Esse capítulo está organizado como segue: a seção 2.1 trata a respeito da poluição do ar e suas consequências, a 2.2 descreve o *hardware* que será utilizado para a coleta dos dados, a 2.3 fala sobre a linguagem de programação utilizada no Arduino, a 2.4 apresenta a linguagem Java, que será utilizada para a criação de uma aplicação *web* e de um *web service* RESTful, a 2.5 descreve o banco de dados MongoDB, banco de dados que é NoSQL e orientado a documentos, sendo que os dados enviados para o *web service* RESTful serão salvos no MongoDB, a 2.6 trata a respeito da linguagem de marcação HTML5, da linguagem de leiaute CSS3 e da linguagem de programação JavaScript, a 2.7 fala sobre o OpenShift, plataforma PaaS (*Platform as a Service*) da Red Hat, a 2.8 apresenta a internet das coisas e a 2.9 mostra alguns projetos na área da internet das coisas.

2.1- Poluição do ar

O aumento da poluição do ar ao redor do mundo tem afetado todos os seres vivos ao longo dos anos (CASTRO, 2003), trazendo para o organismo desses seres vivos materiais como óxidos de enxofre, óxidos de nitrogênio, monóxido de carbono e até chumbo particulado (VANZ, 2003) (CASTRO, 2003), materiais que causam problemas para a saúde desses seres vivos (PEITER, 1998).

A qualidade do ar no Brasil é regulamentada pelo Conselho Nacional do Meio Ambiente - CONAMA (PEREIRA, 2013) (CONAMA, 2013b), e essa regulamentação é muito importante, pois a poluição proveniente de queimadas, indústrias, veículos automotores, entre outros, pode afetar bens materiais, qualidade do solo, qualidade das águas, prejudicar a visibilidade, e o que já foi citado e que é o mais grave, prejudicar a saúde dos seres vivos (PORTAL BRASIL, 2013). A figura 2.1 (PORTAL BRASIL, 2013) mostra uma queimada, uma das principais causas da poluição do ar.

Figura 2.1 – Queimadas: uma das principais fontes de poluição do ar



2.2- Hardware para coleta de dados

Para o desenvolvimento do sistema de medição de qualidade do ar tratado nesse trabalho serão utilizados alguns dispositivos de *hardware*, e esses dispositivos serão descritos nessa seção.

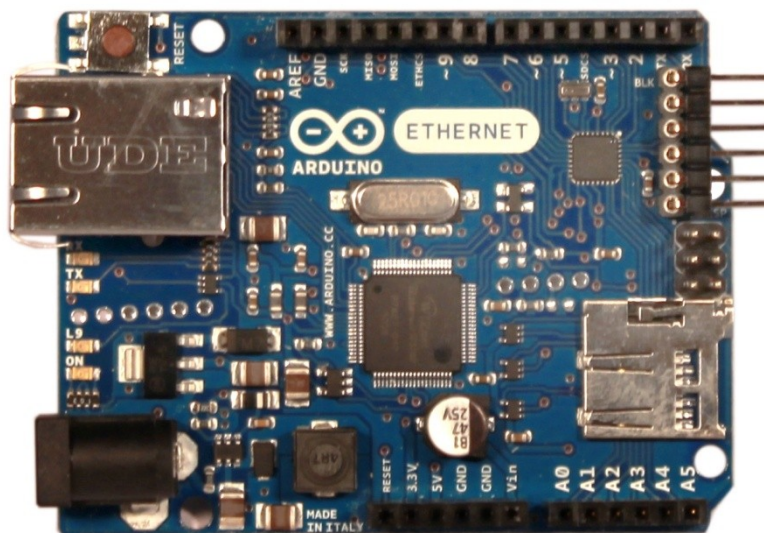
2.2.1- Arduino

Para a medição da qualidade do ar será necessária a utilização de um dispositivo para o controle dessa medição, e esse dispositivo é o Arduino.

O Arduino é uma placa microcontroladora, de *hardware* e *software open source* (BANZI, 2009) que pode ser utilizada para o desenvolvimento de diversas ferramentas, ferramentas essas que podem ser utilizadas para a interação entre objetos.

O Arduino pode controlar diversos sensores e dispositivos, sendo que muitas vezes o Arduino é uma solução barata quando comparado a outros dispositivos (ARDUINO, 2013b). A figura 2.2 (ARDUINO, 2013a) mostra o Arduino Ethernet, modelo utilizado no dispositivo de medição de qualidade do ar tratado no presente trabalho.

Figura 2.2 – Arduino ETHERNET



O Arduino é conectado ao computador via cabo USB para a sua programação, e é compatível com Windows, Linux e Macintosh, fato que associado a facilidade da utilização do Arduino, o torna uma ótima ferramenta (BANZI, 2009).

Além do já foi dito, vale ressaltar que o ambiente de desenvolvimento do Arduino é simples e prático, tornando a codificação fácil de aprender para os iniciantes ao mesmo tempo que o ambiente é prático e útil para os desenvolvedores experientes.

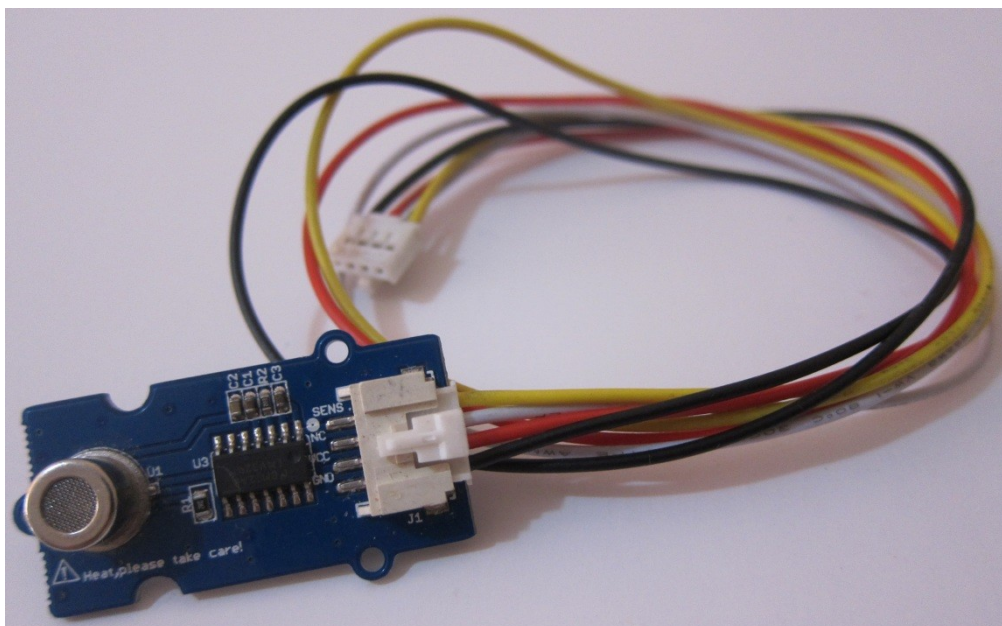
2.2.2- Sensores

Sensores são dispositivos que detectam algum acontecimento físico em específico e convertem esse acontecimento em sinais elétricos que serão interpretados e utilizados para gerar dados (SOUZA, 2012).

Juntamente com o Arduino podem ser utilizados diversos sensores, entre eles estão sensores de proximidade (FREITAS, 2011), sensores de umidade e temperatura do ar (INÁCIO, 2012), entre outros.

No sistema de medição de qualidade do ar que será desenvolvido nesse trabalho será utilizado um sensor de qualidade do ar controlado pelo Arduino. A figura 2.3 (ARDUINO, 2013d) mostra o sensor de qualidade do ar que será utilizado para o dispositivo tratado nesse trabalho.

Figura 2.3 – Sensor de Qualidade do Ar que será utilizado no sistema



2.3- Processing

O dispositivo Arduino (ARDUINO, 2013b) deve ser programado para o seu devido funcionamento, e para essa programação é utilizada uma linguagem chamada Processing (PROCESSING, 2013).

O Processing é uma linguagem de programação simples que quando foi desenvolvida tinha em foco os desenvolvedores que desejavam fazer um código simples antes de desenvolver a versão final (PROCESSING, 2013).

O Processing começou visando a simplicidade, porém com o tempo a linguagem ganhou mais importância sendo utilizado também para trabalhos mais avançados (PROCESSING, 2013), e como já mencionado, é utilizado para a programação do Arduino (ARDUINO, 2013c).

2.4- Java

Java é uma linguagem de programação criada pela Sun Microsystems, sendo que o seu desenvolvimento começou em 1991 (ORACLE, 2013a). Atualmente o Java pertence a Oracle, pois em 2009 a Oracle comprou a Sun Microsystems (ORACLE, 2013b).

A linguagem de programação Java é multiplataformas, pois aplicativos desenvolvidos em Java são executados em uma máquina virtual (Java Virtual Machine - JVM), possibilitando que esses aplicativos possam ser executados em qualquer sistema que tenha suporte a JVM.

O Java é uma linguagem de fácil aprendizado, e por conta disso é a escolha de muitas instituições de ensino ao redor do mundo (ORACLE, 2013c). Esta é uma linguagem de programação orientada a objetos, fortemente tipada e de alto nível, que visa ser uma linguagem de produção, e não de pesquisa (ORACLE, 2013d).

No presente trabalho a linguagem Java será utilizada para o desenvolvimento de uma aplicação *web* cliente para exibir os dados enviados ao *web service* Xively pelo dispositivo de medição de qualidade do ar, e também para o desenvolvimento de um *web service* RESTful que também receberá os dados a respeito da qualidade do ar.

2.5- MongoDB

O MongoDB é um banco de dados NoSQL orientado a documentos,

2.6- HTML5, CSS3 e JavaScript

Para o desenvolvimento da aplicação web que exibirá as informações obtidas a partir do dispositivo de medição da qualidade do ar serão utilizadas a tecnologia HTML5 e CSS3 para a parte de visualização da aplicação, e essas tecnologias serão apresentadas nesta seção.

2.6.1- HTML5

O HTML é uma linguagem de marcação utilizada para a criação de páginas web criada em 1990 por Tim Berners-Lee no Conseil Européen pour la Recherche Nucléaire (ou European Council for Nuclear Research – CERN), localizado na Suíça (W3C, 2013b) (CERN, 2013).

O HTML possibilita que o desenvolvedor publique em uma página web textos, tabelas, imagens, listas, entre outros, e também da a possibilidade de redirecionamento para outras páginas por meio de links. Para a utilização do HTML são utilizadas *tags* de marcação, *tags* que o navegador interpreta para poder mostrar o resultado desejado pelo desenvolvedor (W3C, 2013a).

Para desenvolver sites com mais recursos e com melhor aparência diversas tecnologias foram tentadas, porém com o HTML5 e o CSS3 sites com diversos recursos e efeitos podem ser criados, não sendo necessária a utilização de nenhuma outra tecnologia (HOGAM, 2011).

2.6.2- CSS3

O CSS (Cascading Style Sheets) é uma linguagem que tem a finalidade de descrever como páginas da web devem ser apresentadas, ou seja, o CSS basicamente define qual será a aparência de um site (W3C, 2013a). Essa linguagem foi lançada por Håkon Wium Lie e Bert Bos juntamente ao World Wide Web Consortium (W3C) em 1996 (W3C, 2013c) (W3C, 2013d).

Juntamente com o HTML5, o CSS3 é um dos principais responsáveis pela construção da web moderna, pois os dois trazem muitas vantagens, como compatibilidade com diversos browsers, facilidade para criar bons efeitos visuais, entre outras diversas vantagens (HOGAN, 2011), porém o CSS pode ser utilizado em qualquer linguagem de marcação baseada em XML, e não só no HTML (W3C, 2013a).

O CSS ganhou seu espaço ao longo do tempo, acompanhando o surgimento e evolução de novos *browsers*, que cada vez mais se utilizaram do CSS, pois assim não foi necessário criarem uma própria linguagem, e com isso as páginas da web podem ser executadas em diversos *browsers* (GASSTON, 2011).

2.6.3- JavaScript

JavaScript é uma linguagem de programação de *script* criada em maio de 1995 por Brendan Eich, que na época trabalhava para a Netscape, sendo que a linguagem tinha o nome Mocha no momento de sua criação, sendo que no mesmo ano o nome foi alterado para JavaScript, com a devida autorização da Sun Microsystems, sendo que isso foi como uma estratégia de *marketing* por conta da popularidade do Java.

2.7- OpenShift

O OpenShift é uma plataforma PaaS (Platform as a Service) da Red Hat, plataforma essa que permite aos desenvolvedores criarem aplicações e rapidamente hospedarem essas aplicações em um ambiente em nuvem, não tendo que se preocupar com variáveis de ambiente.

O OpenShift disponibiliza diversas ferramentas e linguagens, sendo que essas são apresentadas para o desenvolvedor no momento em que ele cria uma aplicação, e o ambiente é configurado automaticamente quando a aplicação é criada, restando para o desenvolvedor apenas se preocupar com o código de sua aplicação.

2.8- Internet das Coisas

A internet nos dias de hoje está muito presente no dia-dia de muitas pessoas ao redor do mundo, e cada vez mais dispositivos estão conectados à internet, facilitando o cotidiano do ser humano e, além disso, proporcionando mais segurança (UCKELMANN, 2011).

A cada dia que se passa mais pessoas tem dispositivos conectados à internet, mais dispositivos passam a conectar a internet e esses dispositivos começam a se comunicar entre si e com outros dispositivos a sua volta (RAUNIO, 2010).

A internet das coisas é basicamente a junção de diversos componentes (componentes como os já mencionados) que compõem uma rede onde diversos objetos se comunicam para gerar dados de onde serão obtidas informações que contribuirão para a sociedade, sendo que esse recurso pode ser utilizado por diversas áreas como saúde, segurança, meio ambiente, entre outras (COUNCIL, 2013).

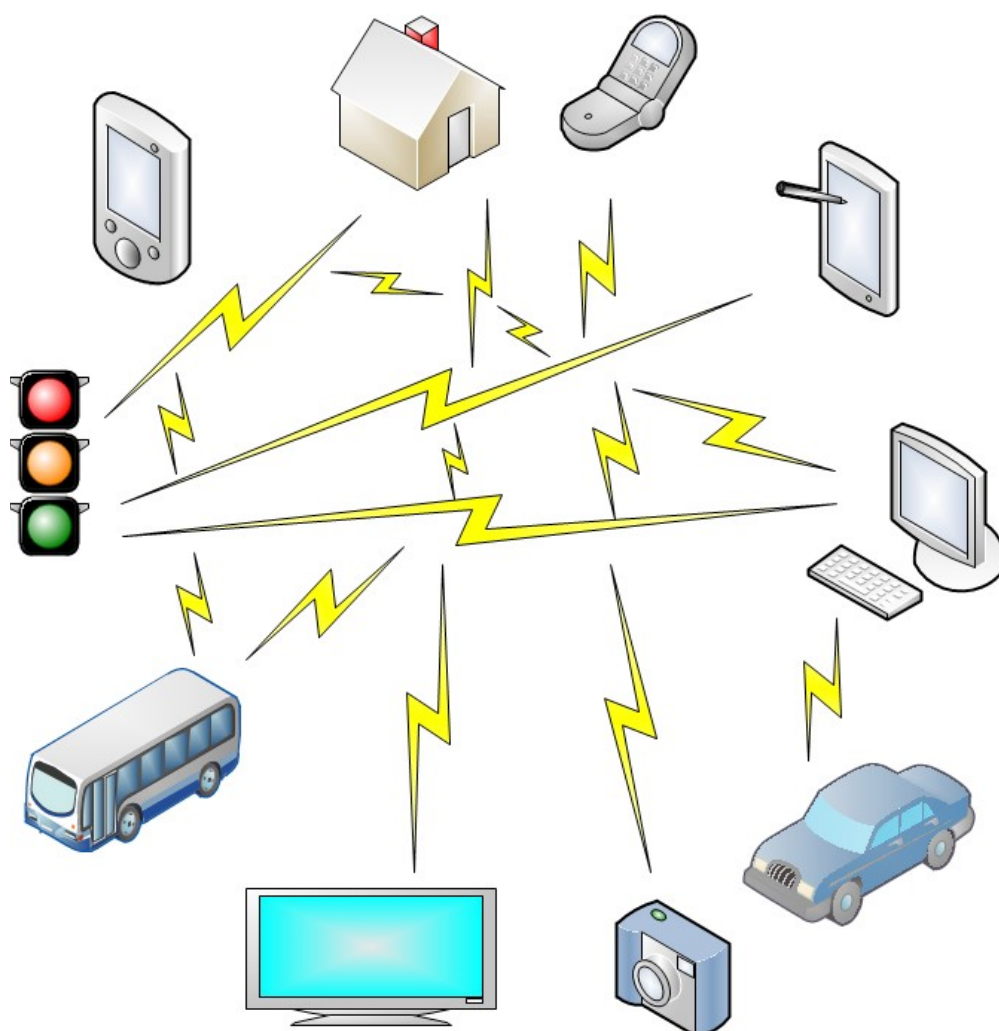
Tanto no meio empresarial como no doméstico e ambiental a internet é muito útil, porém em alguns desses meios ela ainda pode ser mais explorada, e facilitar os

processos, e a internet das coisas proporciona essa evolução dos processos com todas as suas possibilidades (ATZORI, 2010).

Outra grande vantagem da internet das coisas é o fato de que todos podem contribuir (UCKELMANN, 2011), e com ferramentas como o Xively (XIVELY, 2013) disponíveis gratuitamente, dispositivos de baixo custo como o Arduino (ARDUINO, 2013b) e todo o conhecimento disponível na web qualquer pessoa pode contribuir com a internet das coisas (UCKELMANN, 2011).

A figura 2.4 (IBM, 2013) representa a internet das coisas, que interconecta diferentes objetos formando uma rede de informações.

Figura 2.4 – Representação da Internet das Coisas



2.9- Projetos relacionados a Internet das Coisas

Nesta seção serão apresentados alguns projetos na área da Internet das Coisas para que o leitor possa ter um exemplo prático do dispositivo que será desenvolvido nesse trabalho.

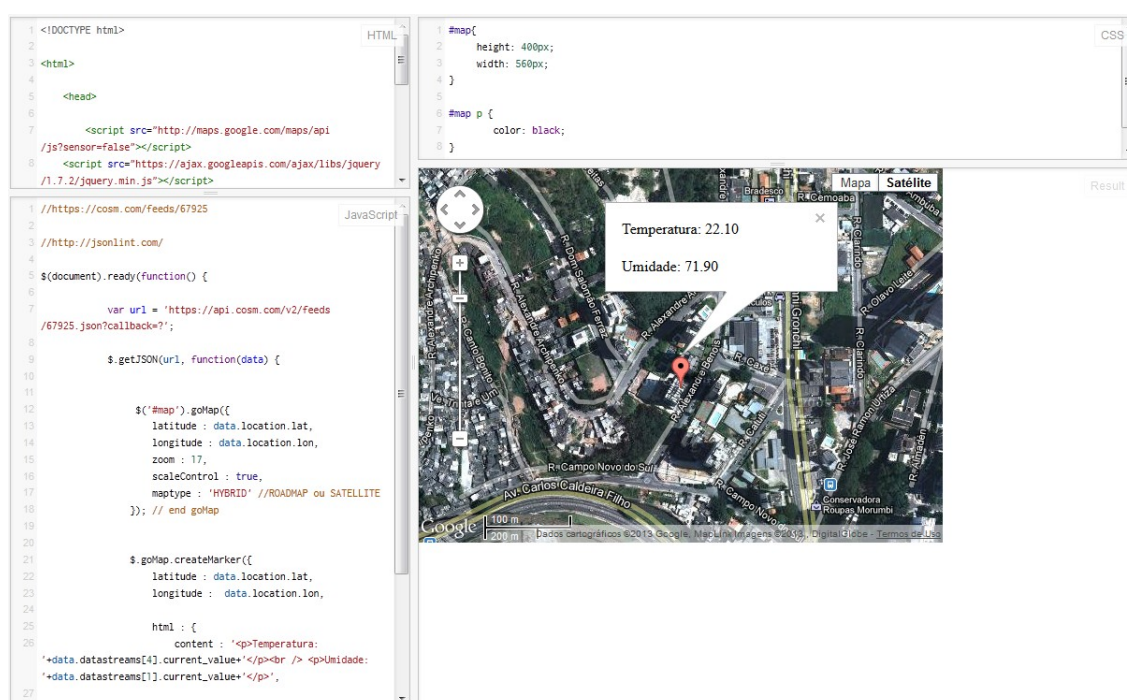
2.9.1- Aplicação para exibição de umidade do ar e temperatura

A internet das coisas permite o desenvolvimento de diversos dispositivos, em diversas áreas, e como primeiro exemplo será apresentada uma aplicação presente na plataforma jsFiddle (JSFIDDLE, 2013) (BERTOTI, 2013).

A aplicação consiste em mapa georreferenciado (GOOGLE MAPS, 2013), mapa esse que tem um ponto plotado referente ao local onde se encontra um dispositivo desenvolvido com uma placa Arduino (ARDUINO, 2013b), sendo que esse dispositivo conta com um sensor de temperatura e outro de umidade do ar (BERTOTI, 2013).

Os dados colhidos pelos sensores são enviados para o *web service* Xively (XIVELY, 2013) de onde os dados são obtidos pela aplicação por meio de JavaScript e JSON, e então são plotados no mapa. A figura 2.5 (BERTOTI, 2013) mostra a tela da aplicação sendo executada no jsFiddle.

Figura 2.5 – Aplicação de exibição de umidade do ar e temperatura



2.9.2- XIVELY

O Xively é um *web service* de utilização gratuita que possibilita aos seus usuários o envio dos dados coletados por dispositivos semelhantes ao que será desenvolvido nesse trabalho, entre outros, e esses dados ficam salvos pelo Xively. O Xively foi fundado em 2008 com o nome de Pachube (XIVELY, 2013a) (XIVELY, 2013b).

Os dados enviados pelos usuários podem também ser utilizados por outras aplicações que obtém esses dados do *web service* utilizando o padrão RESTful com comunicação JSON (XIVELY, 2013a).

O Xively é uma importante ferramenta relacionada a internet das coisas, pois possibilita que desenvolvedores que possuem um dispositivo que fornece informações relevantes disponibilizem essas informações de maneira prática (XIVELY, 2013b).

3– DESENVOLVIMENTO

Esse capítulo é responsável por apresentar ao leitor o processo de desenvolvimento do dispositivo, mostrando toda a arquitetura do sistema. As seções do capítulo são divididas conforme os módulos do sistema, que também são descritos neste capítulo.

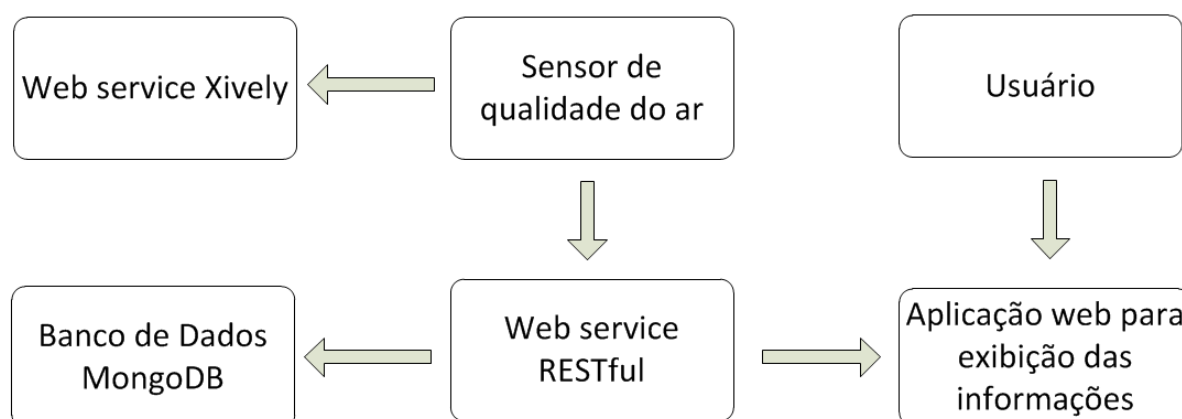
Esse capítulo está organizado como segue: a seção 3.1 apresenta a arquitetura do sistema, a seção 3.2 descreve o dispositivo de medição da qualidade do ar, a seção 3.3 apresenta o sistema principal, a seção 3.4 mostra o software de controle do Arduino e a seção 3.5 descreve o *web service* Xively.

3.1 – Arquitetura geral

O sistema desenvolvido possui uma arquitetura que envolve todas as aplicações e recursos apresentadas no capítulo 2. Nessa seção será apresentada a arquitetura geral do sistema e a divisão entre módulos, mostrando onde e como as aplicações e dispositivos são utilizados.

A figura 3.1 apresenta a arquitetura geral do sistema, representando os componentes envolvidos e a relação entre esses componentes.

Figura 3.1 – Arquitetura geral do sistema



O funcionamento do sistema começa a partir do dispositivo de medição da qualidade do ar (ARDUINO, 2013a) (ARDUINO, 2013d), que coletará os dados e os enviará para o web service Xively (XIVELY, 2013a) e para o *web service* RESTful (FIELDING, 2000). Os dados recebidos pelo *web service* RESTful são salvos em um banco de dados MongoDB (MONGODB, 2013) e disponibilizados em uma aplicação web para acesso livre de qualquer usuário.

Além de ficarem disponíveis para o usuário por meio da aplicação web, os dados coletados ficam disponíveis para acesso também no Xively, e podem ser obtidos do *web service* RESTful para uso de aplicações por meio do formato JSON (JSON, 2013).

Para uma melhor apresentação do sistema, ele foi dividido em módulos, sendo que cada módulo abrange os componentes de uma parte do sistema.

A tabela 3.1 apresenta os módulos do sistema, e os componentes que cada módulo possui.

Tabela 3.1 - Módulos do sistema

	MÓDULO	COMPONENTES
HARDWARE	Dispositivo de medição de qualidade do ar	Arduino
		Sensor de qualidade do ar
		<i>Web service</i> RESTful
SOFTWARE	Sistema principal	Aplicação Web
		Banco de dados MongoDB
	Software de controle do Arduino	Software de controle do Arduino
	Web service Xively	Web service Xively

3.2 – Dispositivo de medição de qualidade do ar

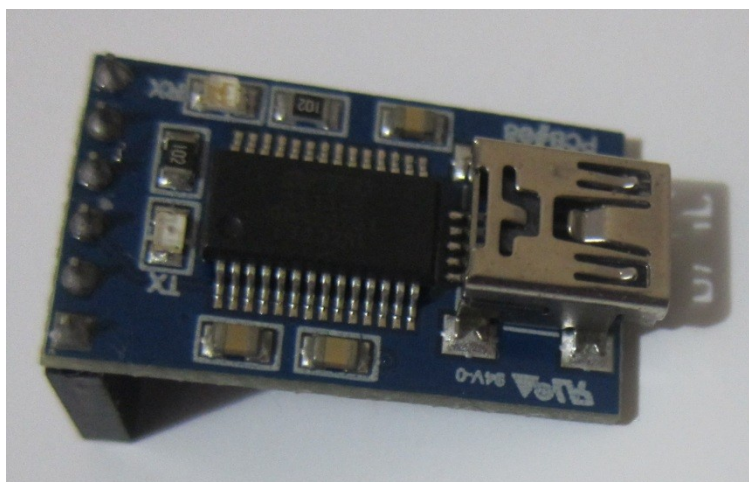
A realização das medições da qualidade do ar é realizada por um dispositivo formado por um microcontrolador Arduino (ARDUINO, 2013a) e por um sensor de qualidade do ar (SEEED STUDIO, 2013a). Ambos os dispositivos foram adquiridos prontos para uso, bastando apenas realizar a programação do Arduino.

O Arduino utilizado é o modelo Ethernet, com esse modelo não é necessário o uso de um módulo ethernet, pois ele já possui uma porta para a comunicação com a internet.

O Arduino Ethernet tem as seguintes especificações: Microcontrolador ATmega328, voltagem de operação de 5V, voltagem de alimentação de 7 a 12V, 14 pinos digitais, 6 pinos analógicos, memória flash de 32 KB (onde 0.5 KB é utilizado para *boot*), velocidade de *clock* de 16 MHz, entrada para cartão Micro SD e controlador ethernet W5100 TCP/IP integrado. Para a sua programação o Arduino Ethernet é conectado ao computador por meio de sua porta serial de seis pinos, sendo que para essa conexão é utilizado um conversor de serial para USB (ARDUINO, 2013d). O Arduino Ethernet pode ser alimentado pela conexão USB, por uma bateria de 9V ou por uma fonte ligada a uma tomada.

A figura 3.2 (ARDUINO, 2013d) mostra o conversor serial / USB utilizado para a conexão do Arduino com o computador para sua programação.

Figura 3.2 – Conversor Serial/USB



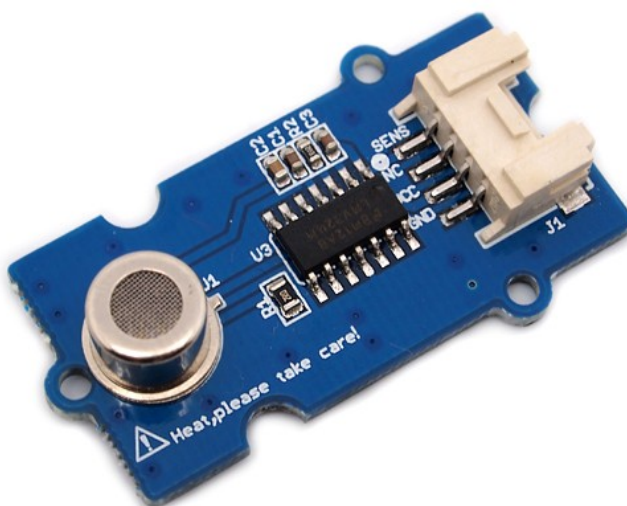
O Arduino (ARDUINO, 2013a) controla o dispositivo e para a medição da qualidade do ar é utilizado um sensor de qualidade do ar (SEEED STUDIO, 2013a). O sensor utilizado é sensível a um amplo número de gases, gases como monóxido de carbono, álcool, acetona entre outros (SEEED STUDIO, 2013b).

O sensor não apresenta dados quantitativamente (ele não é capaz, por exemplo, de dizer a concentração de um gás no ar), ele apresenta dados qualitativos, como seu próprio nome já diz, ele apresenta ao usuário qual é a qualidade atual do ar.

O sensor de qualidade do ar utilizado é desenvolvido pela Seeed Studio, empresa situada na China que produz diversos componentes de hardware *open source* (SEEED STUDIO, 2013c).

Na figura 3.3 (SEEED STUDIO, 2013b) pode ser visto o sensor de qualidade do ar utilizado no sistema.

Figura 3.3 – Sensor da qualidade do ar

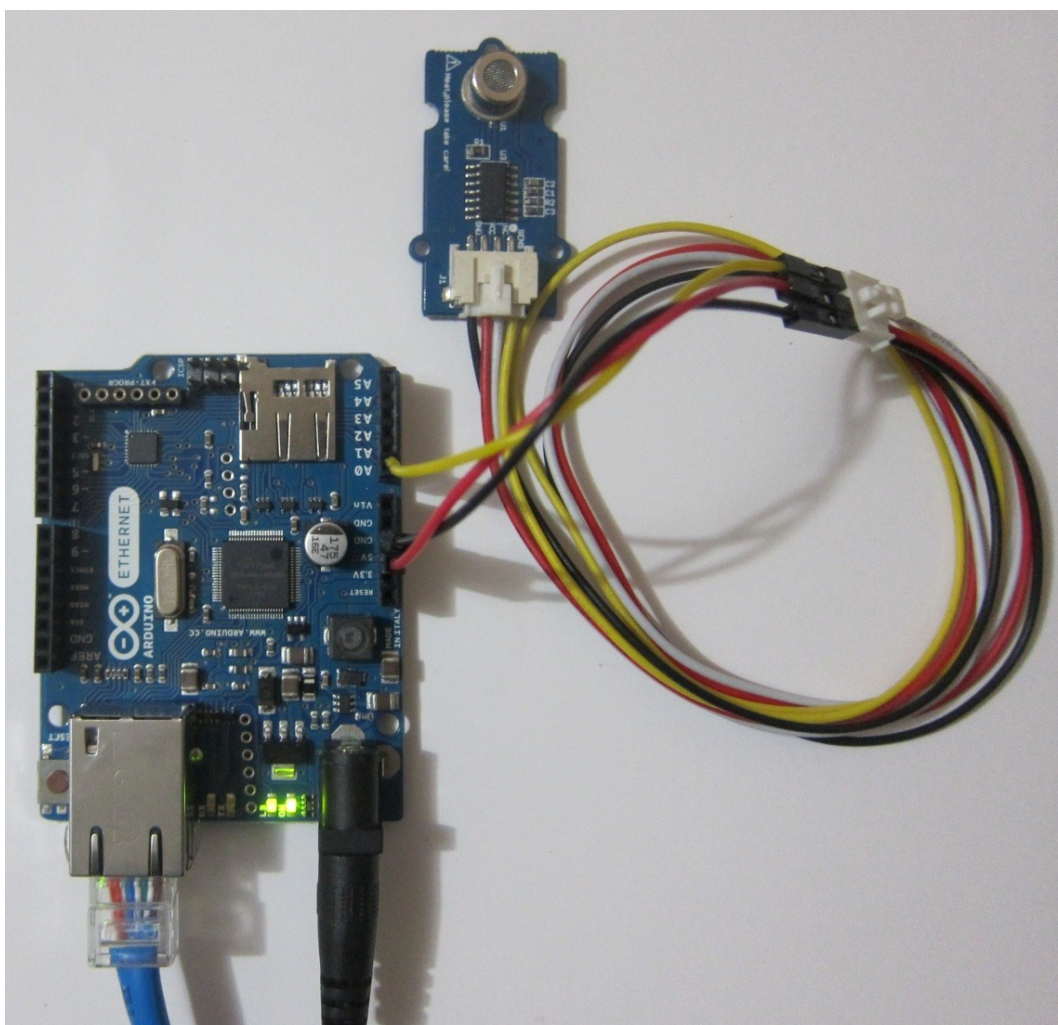


Para o funcionamento do dispositivo o sensor de qualidade do ar é conectado a uma porta analógica, a uma porta de alimentação de 5V e a uma porta de conexão terra (GND) do Arduino.

Para a alimentação do Arduino é utilizada uma fonte de 9V com positivo interno e para a comunicação com a internet é conectado ao Arduino um cabo TP CAT 5e.

Na figura 3.4 (ARDUINO, 2013d) pode ser visto o dispositivo de medição da qualidade do ar conectado e em funcionamento.

Figura 3.4 – Dispositivo de medição da qualidade do ar conectado e em funcionamento



3.3 – Sistema principal

Os dados obtidos pelo dispositivo de medição da qualidade do ar são enviados ao *web service* Xively (XIVELY, 2013a) e são também enviados a um *web service* RESTful (FIELDING, 2000). Esse *web service* possui também uma interface que pode

ser acessada via navegador, e essa é aplicação web responsável por apresentar aos usuários os dados obtidos pelo dispositivo de medição.

Além de serem disponibilizados na aplicação web os dados ficam disponíveis no *web service* RESTful, dados que quando acessados por algum usuário são retornados no formato JSON. Esse recurso tem por objetivo atender a desenvolvedores que desejam utilizar os dados em outras aplicações, e assim podem obtê-los diretamente pelo seu código fonte.

Os dados são enviados para o *web service* RESTful pelo próprio Arduino por meio de uma requisição GET. Inicialmente seria utilizado o método POST, porém foram encontradas dificuldades para realizar a comunicação entre o Arduino e o *web service* via POST. Como via GET qualquer usuário pode fazer uma requisição por meio de um navegador, juntamente com os dados da qualidade do ar, é enviado para o *web service* uma chave especial para que ele possa reconhecer que é uma requisição do Arduino, e assim apenas o Arduino consegue enviar os dados para o *web service*.

O *web service* RESTful é desenvolvido na linguagem Java, utiliza a API Jersey e é hospedado no OpenShift (OPENSIFT, 2013a), que é uma Plataforma como um Serviço (PaaS) (OPENSIFT, 2013b). O OpenShift é a PaaS da Red Hat, e tem suporte a diversas tecnologias e ferramentas. Com o OpenShift os desenvolvedores não se preocupam com o ambiente, e podem ficar focados apenas no desenvolvimento.

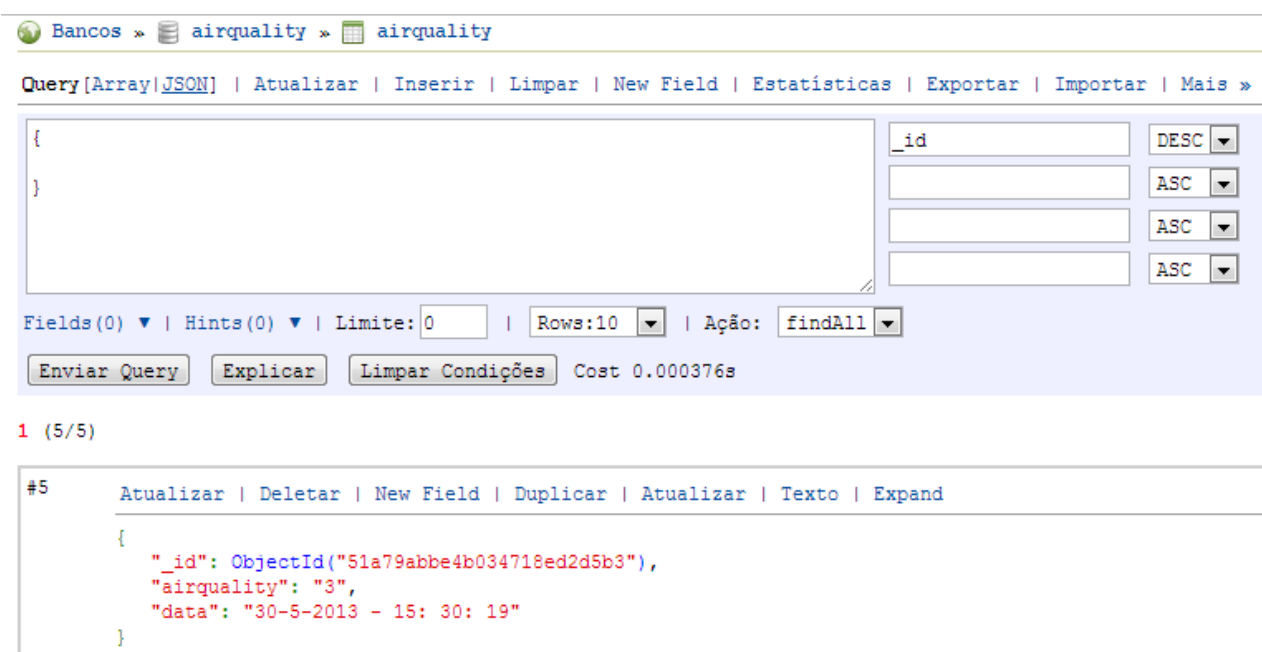
A partir do *web service* RESTful, os usuários podem obter não só a qualidade atual do ar, mas também uma média da qualidade do ar de um certo período de tempo.

Os dados enviados pelo Arduino para o *web service* RESTful são salvos em um banco de dados MongoDB (MONGODB, 2013a).

O MongoDB é um banco de dados NoSQL orientado a documentos e como ele utiliza o formato BSON (*binary* JSON) ele é mais prático de ser utilizado pelo *web service* do que seria um banco de dados relacional, isso porquê o formato BSON é uma variação do JSON, e o *web service* utiliza a comunicação via JSON (MONGODB, 2013b).

Na figura 3.4 (MONGODB, 2013a) pode ser visto um registro salvo no banco de dados, registro esse que tem a informação a respeito da qualidade do ar e a data e hora que essa informação foi medida pelo dispositivo.

Figura 3.4 – Registro salvo no MongoDB



O banco de dados criado salva as informações das medições e os horários destas medições. A partir dos dados salvos nesse banco, são geradas diversas informações que ficam disponíveis tanto no *web service* RESTful (FIELDING, 2000) como na aplicação web.

A aplicação web exibe os dados aos usuários por meio de gráficos, passando informações de diversos períodos, sendo que esses gráficos são atualizados a cada vez que mais dados são enviados ao *web service* RESTful e são gravados no banco de dados.

A aplicação web é desenvolvida nas linguagens Java, HTML5, CSS3 e JavaScript, e nela também é utilizada a API Chart.js (CHART.JS, 2013a) para a exibição das informações por meio de gráficos.

A aplicação consulta os dados a respeito da qualidade do ar no banco de dados MongoDB, formata esses dados e exibe as informações nos gráficos, sendo que existem três gráficos, que exibem a média da qualidade do ar dos dias do período de sete dias, de trinta dias e de seis meses. Além dos gráficos também é exibido ao usuário a qualidade atual do ar e a localização do dispositivo.

3.4 – Software de controle do Arduino

Para o funcionamento do dispositivo de medição de qualidade do ar o Arduino deve ser programado. O software utilizado para o controle do Arduino é compilado e salvo na memória da placa, e a partir do momento que a placa é alimentada o software é executado (ARDUINO, 2013d).

O Arduino é programado na linguagem Processing (PROCESSING, 2013), como já foi citado, e utiliza as seguintes bibliotecas:

- SPI.h: possibilita a comunicação do Arduino com periféricos como cartões MicroSD e dispositivos conectados na porta serial.
- Xively.h: utilizada para a comunicação entre o Arduino e o *web service* Xively;
- AirQuality.h: facilita a comunicação entre o Arduino e o sensor de qualidade do ar, pois interpreta o valor apurado pelo sensor e retorna para o Arduino de maneira simples qual é a atual qualidade do ar.
- Ethernet.h: possibilita que o Arduino se comunique com a internet por meio da porta Ethernet presente no dispositivo.
- HttpClient.h: por meio dessa biblioteca é possível utilizar os métodos HTTP na placa Arduino.

3.5 – Web service Xively

Para que fiquem disponíveis para outros usuários, os dados obtidos a respeito da qualidade do ar são enviados para o *web service* Xively, e todos esses dados ficam disponíveis no próprio Xively dispostos em gráficos (XIVELY, 2013a), e também podem ser obtidas por outras aplicações por meio de JSON (JSON, 2013).

Além da linguagem Java, o Xively pode ser utilizado juntamente com outras linguagens como Objective C, C, JavaScript e Ruby, e além de JSON podem ser utilizados XML e formatos de dados CSV (XIVELY, 2013a).

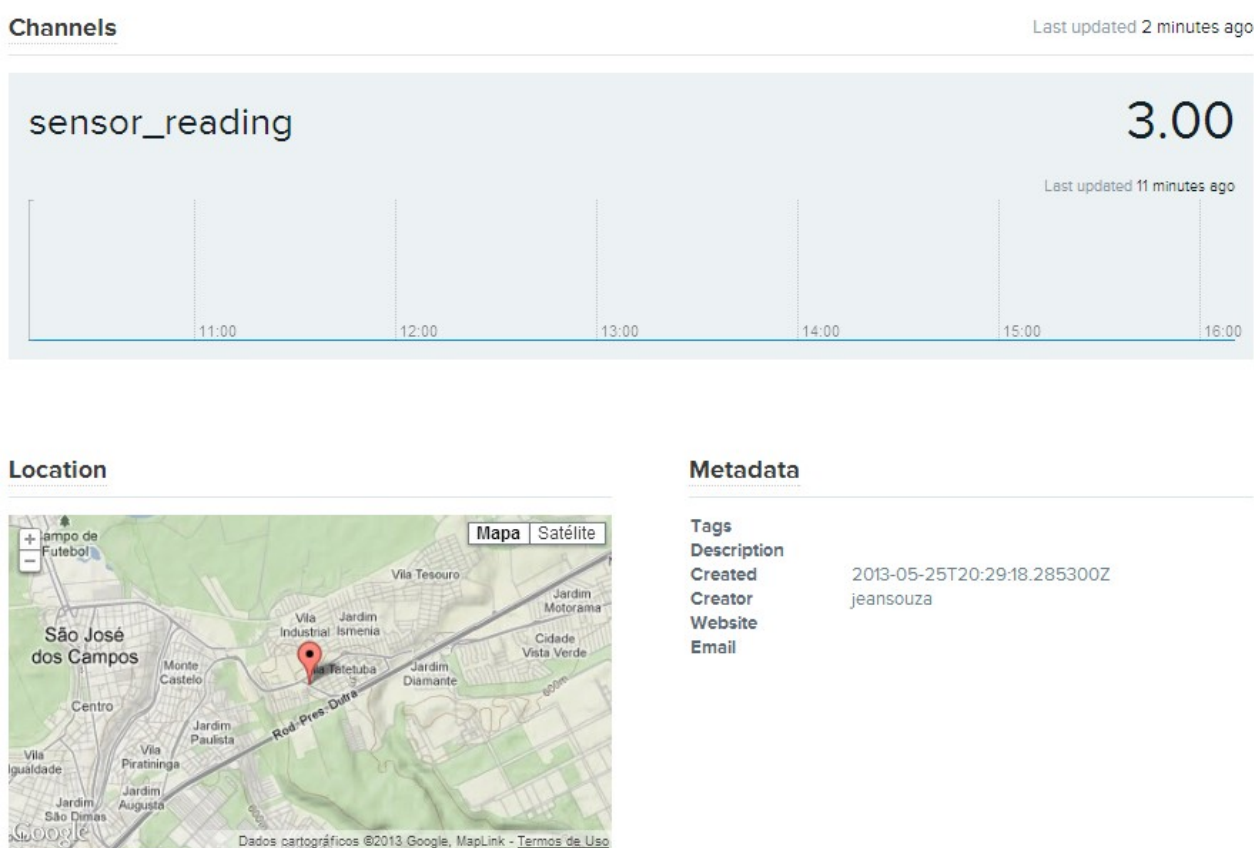
O Xively disponibiliza a biblioteca para o envio dos dados e toda a documentação para o desenvolvimento do código que envia os dados para o *web service*. Para realizar o acesso aos dados enviados para o *web service* basta que o usuário tenha a identificação do *feed* criado no Xively.

Para realizar o acesso a um *feed* do Xively basta adicionar a identificação do *feed* ao final da URL <https://xively.com/feeds/>. O *feed* criado para o envio dos dados do sistema pode ser acessado através da URL <https://xively.com/feeds/140271570>, onde o número 140271570 é a identificação do *feed*.

Acessando um *feed* é possível visualizar o valor atual do dispositivo, um gráfico com as medições das últimas horas e também a localização do dispositivo.

A figura 3.5 (XIVELY, 2013) mostra o *feed* criado para o sistema já apresentando valores obtidos pelo dispositivo de medição de qualidade do ar e a localização do dispositivo.

Figura 3.5 – Feed criado para a aplicação no Xively



REFERENCIAS BIBLIOGRÁFICAS

ARDUINO. **Arduino Ethernet.** Disponível em <http://arduino.cc/en/Main/ArduinoBoardEthernet>, acessado em: 06/03/2013 (a).

ARDUINO. **Introduction.** Disponível em <http://arduino.cc/en/Guide/Introduction>, acessado em: 12/03/2013 (b).

ARDUINO. **Examples.** Disponível em <http://arduino.cc/en/Tutorial/HomePage?from=Main.LearnArduino>, acessado em: 22/03/2013 (c).

ARDUINO. **Arduino.** Disponível em <http://www.arduino.cc/>, acessado em: 24/03/2013 (d).

ATZORI, L.; IERA, A. e MORABITO, G.: *The Internet of Things: A survey.* **Elsevier.** Vol. 54, n. 15, p. 2787-2805, 2010.

BANZI, M. **Getting Started With Arduino.** Sebastopol: O'Reilly Media, Inc., 2009.

BERTOTI. **Notas da aula de Interação Humano Computador.** Disponível em <http://fatecsjc.edu.br/ead/>, acessado em: 26/03/2013.

CASTRO, H. A.; GOUVEIA, N. e CEJUDO, J. A. E.: Questões metodológicas para a investigação dos efeitos da poluição do ar na saúde. **Revista Brasil Epidemiologia.** Vol. 6, Nº 2, p. 135 - 149, 2003.

CERN. **About CERN.** Disponível em <http://home.web.cern.ch/about>, acessado em: 23/03/13.

CETESB - Companhia Ambiental do Estado de São Paulo. **Relatório de qualidade do ar no Estado de São Paulo de 2011.** Disponível em <http://www.cetesb.sp.gov.br/ar/qualidade-do-ar/31-publicacoes-e-relatorios>, acessado em: 06/03/2013.

CONAMA. **Resolução Conama Nº 005, de 15 de junho de 1989.** Disponível em http://portal.saude.gov.br/portal/arquivos/pdf/resolucao_conama_n_005_pronar.pdf, acessado em: 27/02/2013 (a).

CONAMA. **Resolução Conama Nº 3, de 28 de junho de 1990.** Disponível em <http://www.mma.gov.br/port/conama/legiabre.cfm?codlegi=100>, acessado em: 27/02/2013 (b).

CHART.JS. **Chart.js.** Disponível em <http://www.chartjs.org/>, acessado em: 13/08/2013.

COUNCIL. *Internet of Things: what is it?*. Disponível em <http://www.theinternetofthings.eu/internet-of-things-what-is-it%3F>, acessado em: 18/03/2013.

CROCE, M.; VASCONCELOS, D. M.; MANSO, E. R. C. e DUARTE, A. J. S.: Poluição ambiental e alergia respiratória. **Medicina, Ribeirão Preto.** 31, p. 144-153, jan.-mar. 1998.

FIELDING, R. T.: *Architectural Styles and the Design of Network-based Software Architectures.* Dissertação (Doctor of Philosophy in Information and Computer Science) - University of California, Irvine, 2000.

FREITAS, C.; BREMNER, S. A.; GOUVEIA, N.; PEREIRA, L. A. A. e SALDIVA, P. H. N.: Interações e óbitos e sua relação com a poluição atmosférica em São Paulo, 1993 a 1997. **Rev. Saúde Pública.** 2004;38(6), p. 751-757.

FREITAS, G. S.: **Dispositivo eletrônico com sensor de distância para auxílio na apresentação de Datashow de documentos em ambiente virtual.** 79 f. Trabalho de Conclusão de Curso (Engenharia de Computação) – Centro Universitário de Brasília, Brasília, 2011.

GOOGLE MAPS. **Google Maps.** Disponível em <http://maps.google.com.br/>, acessado em: 22/03/2013.

HOGAN, B. P. *HTML5 and CSS3: Develop with Tomorrow's Standards Today.* Sebastopol: O'Reilly Media, Inc., 2011.

IBM. *What is a Smarter Planet?*. Disponível em <http://www.ibm.com/smarterplanet/us/en/overview/ideas/index.html?re=sph>, acessado em: 06/04/2013.

INÁCIO, M. J.; MAIA, R. D.; NETTO, J. C.; OLIVEIRA, H. R.; SOUZA, T. A.; CASTRO, F. T. C. e MARIA, D. G. S.: Sistema embarcado baseado em hardware livre com utilização de computação nas nuvens aplicado no monitoramento de ambientes. **Revista Pesquisa & Extensão**. ISSN 2236-7160. Vol. 2, n. 1, p. 39 – 43, 2012.

JSFIDDLE. **jsFiddle**. Disponível em <http://jsfiddle.net/>, acessado em: 22/03/2013.

JSON. **Introducing JSON**. Disponível em <http://www.json.org/>, acessado em: 04/06/2013.

MARTINS, L. C.; LATORRE, M. R. D. O.; CARDOSO, M. R. A.; GONÇALVES, F. L. T.; SALDIVA, P. H. N. e BRAGA, A. L. F.: Poluição atmosférica e atendimentos por pneumonia e gripe em São Paulo, Brasil. **Rev. Saúde Pública**. 2002;36(1), p. 88-94.

MONGODB. **MongoDB**. Disponível em <http://www.mongodb.org/>, acessado em: 30/05/2013 (a).

MONGODB. **BSON Documents**. Disponível em <http://docs.mongodb.org/manual/core/document/>, acessado em: 30/05/2013 (b).

NASCIMENTO, L. F. C.; PEREIRA, L. A. A.; BRAGA, A. L. F.; MÓDOLO, M. C. C. e CARVALHO, J. A. J.: Efeitos da poluição atmosférica na saúde infantil em São José dos Campos, SP. **Rev. Saúde Pública**. 2006;40(1), p. 77-82.

OPENSIFT. **OpenShift**. Disponível em <https://www.openshift.com/>, acessado em: 30/05/2013 (a).

OPENSIFT. **OpenShift Platform as a Service**. Disponível em <https://www.openshift.com/paas>, acessado em: 30/05/2013 (b).

ORACLE. **The History of Java Technology**. Disponível em <http://www.oracle.com/technetwork/java/javase/overview/javahistory-index-198355.html>, acessado em: 23/03/2013 (a).

ORACLE. **Oracle Buys Sun**. Disponível em <http://www.oracle.com/us/corporate/press/018363>, acessado em: 23/03/2013 (b).

ORACLE. **Java - Features & Benefits.** Disponível em <http://www.oracle.com/us/technologies/java/features/index.html>, acessado em: 23/03/2013 (c).

ORACLE. **Java - Introduction.** Disponível em <http://docs.oracle.com/javase/specs/jls/se7/html/jls-1.html>, acessado em: 23/03/2013 (d).

PEITER, P. e TOBAR, C.: Poluição do ar e condições de vida: uma análise geográfica de riscos à saúde em Volta Redonda, Rio de Janeiro, Brasil. **Cad. Saúde Pública, Rio de Janeiro.** 14(3), p. 473-485, jul.-set. 1998.

PEREIRA, J. S. J. e CONSULTORIA LEGISLATIVA DA CÂMARA DOS DEPUTADOS. **Legislação brasileira sobre poluição do ar.** Disponível em http://bd.camara.gov.br/bd/bitstream/handle/bdcamara/1542/legislacao_poluicao_ar_jos_e_pereira.pdf?sequence=1, acessado em: 27/02/2013.

PORTAL BRASIL. **Poluição.** Disponível em <http://www.brasil.gov.br/sobre/meio-ambiente/climas>, acessado em: 19/03/2013.

PROCESSING. **Processing.** Disponível em <http://www.processing.org/>, acessado em: 30/05/2013.

RANGEL, M. C. e CARVALHO, M. F. A.: Impacto dos catalisadores automotivos no controle da qualidade do ar. **Química Nova.** Vol. 26, No. 2, p. 265-277, 2003.

RAUNIO, B. **The Internet of things.** Stockholm: .SE, 2010.

SEED STUDIO. **Seed Studio.** Disponível em <http://www.seeedstudio.com>, acessado em: 30/05/2013 (a).

SEED STUDIO. **Grove - Air Quality Sensor.** Disponível em [http://www.seeedstudio.com/wiki/Grove - Air Quality Sensor](http://www.seeedstudio.com/wiki/Grove_-_Air_Quality_Sensor), acessado em: 30/05/2013 (b).

SEED STUDIO. **Company Info.** Disponível em http://www.seeedstudio.com/depot/index.php?main_page=about_us, acessado em: 30/05/2013 (c).

SOUZA, A. J.: *Interface for user-environment interaction using Open Source tools and Internet of Things*. 71 f. Trabalho de Graduação (Tecnologia em Informática com Ênfase em Banco de Dados) - Faculdade de Tecnologia de São José dos Campos, São José dos Campos, 2012.

UCKELMANN, D.; HARRISON, M. e MICHAELLES F. *Architecting the Internet of Things*. Berlin: Springer, 2011.

VANZ, A.; MIRLEAN, N. e BAISCH, P.: Avaliação de poluição do ar por chumbo particulado: uma abordagem geoquímica. **Química Nova**. Vol. 26, No. 1, p. 25-29, 2003.

W3C. **HTML & CSS**. Disponível em <http://www.w3.org/standards/webdesign/htmlcss>, acessado em: 18/03/2013 (a).

W3C. *A history of HTML*. Disponível em <http://www.w3.org/People/Raggett/book4/ch02.html>, acessado em: 23/03/2013 (b).

W3C. *The CSS saga*. Disponível em <http://www.w3.org/Style/LieBos2e/history/>, acessado em: 23/03/2013 (c).

W3C. *ABOUT W3C*. Disponível em <http://www.w3.org/Consortium/>, acessado em: 23/03/2013 (d).

W3C. *A Short History of JavaScript*. Disponível em http://www.w3.org/community/webed/wiki/A_Short_History_of_JavaScript, acessado em: 13/08/2013 (e).

XIVELY. **Xively**. Disponível em <https://Xively.com/>, acessado em: 06/03/2013 (a).

XIVELY. **About**. Disponível em https://Xively.com/about_us, acessado em: 24/03/2013 (b).

XIVELY. ***How it works***. Disponível em https://Xively.com/how_it_works, acessado em: 24/03/2013 (c).