



# Deasfio Hackathon IFNMG

Cédric Lamalle

Version 1.0.0, 02/11/2020

# Apresentação do desafio

O objetivo do documento é apresentar o desafio repassado aos participantes do Hackathon que acontecerá nos dias 15, 16 e 17 de janeiro de 2021.

## Poluição sonora

A poluição sonora é um problema de saúde pública persistente para quem reside em ambiente urbano. Segundo alguns estudos <sup>[1]</sup> de 70% a 90% das pessoas que vivem nas maiores cidades são expostos a barulhos ultrapassando as recomendações da Organização Mundial da Saúde (OMS) <sup>[2]</sup>.

A exposição a sons fortes provoca danos a saúde com a destruição de células sensoriais do ouvido que pode provocar uma perda auditiva temporária ou permanente. Além desses problemas a poluição sonora pode provocar:

- Distúrbio de sono.
- Estresse e ansiedade.
- Dores de cabeça.
- Perda de concentração e consequentemente diminuição da produtividade, que seja no trabalho ou nos estudos.
- Alteração da frequência cardíaca e respiratória.
- Aumento da pressão arterial.

Além dos humanos, os animais são prejudicados da mesma forma. Estudos <sup>[3]</sup> demonstram que várias espécies sofrem da poluição sonora, por exemplo morcegos que dependem de sinais acústicos para encontrar presas. Anfíbios, pássaros, insetos e mamíferos dependem do som para transmitir informações essenciais, como acasalamento e sinais de alerta. Muitas espécies estão fortemente ameaçadas pela poluição sonora.

## Smart Cities e IoT

Existem leis sobre o ruído que visam reduzir a poluição sonora, mas as avaliações e monitoramento desses sons são realizadas com pouca frequência e são principalmente baseados em reclamações feitas pelos moradores.

As cidades inteligentes, as chamadas *Smart Cities*, são as que usam tipos diferentes de sensores eletrônicos para coletar dados e usá-los para gerenciar recursos e ativos eficientemente <sup>[4]</sup>. Dentro dos itens que podem ser monitorados em cidades inteligentes temos:

- Iluminação pública.
- Poluição do ar.
- Transporte.
- Distribuição de energia.
- Monitoramento dos sons da cidade.

Os sensores conectados utilizados para efetuar as medições fazem parte da Internet das Coisas (IoT - *Internet of Things* em inglês). Todos eles estão interconectados e enviam as medidas para serviços encarregados do processamento.

[1] Ver por exemplo <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4918656/>

[2] Ver documentação no site da OMS: <https://www.who.int/docstore/peh/noise/Comnoise-1.pdf> e <https://www.who.int/docstore/peh/noise/Comnoise-4.pdf>.

[3] Exemplo de estudo sobre o impacto dos sons produzidos pelos humanos nos animais: <https://royalsocietypublishing.org/doi/full/10.1098/rsbl.2019.0649>

[4] Wikipedia - [https://pt.wikipedia.org/wiki/Cidade\\_inteligente](https://pt.wikipedia.org/wiki/Cidade_inteligente)

# Desafio

O objeto do desafio é construir um sistema de monitoração e alerta de poluição sonora. Esse sistema será composto de vários módulos com objetivos diferentes. Descreveremos em seguida esses módulos e algumas sugestões de tecnologias para implementação da solução.

Embora falamos sobre *Smart Cities*, o projeto pode se aplicar em um ambiente de tamanho menor, por exemplo, uma fábrica onde os incômodos provocados por altos níveis de barulho podem ser prejudiciais.

## Módulo 1 - Sensores

Nesse módulo deverão ser desenvolvidos os aparelhos que enviarão os dados para os serviços. As informações deverão ser enviadas de tempo em tempo e conterão: \* Data/Hora da aferição. \* Localização. \* Medição em decibéis.

A localização poderá um ponto definido pela latitude e longitude, neste caso pode ser incluído um módulo tipo GPS <sup>[5]</sup> ou a posição pode ser configurada na instalação. No caso de monitoramento em um ambiente fechado, essa posição poderá ser uma *String*, Refeitório canto noroeste.

Os aparelhos IoT são geralmente conectados a um *Gateway* local que efetua o envio dos dados para os serviços de processamento. Para esse desafio a transmissão pode ser feita diretamente pelo sensor para simplificar, o objetivo sendo mostrar a viabilidade de construção de um aparelho desse tipo por um custo baixo.

A geração dos dados dos sensores pode ser simulada, como por exemplo com o uso do [Kafka Connect Datagen](#). Ele pode ser usado como método para simular os dados coletados dos sensores e enviá-los para um ponto único.

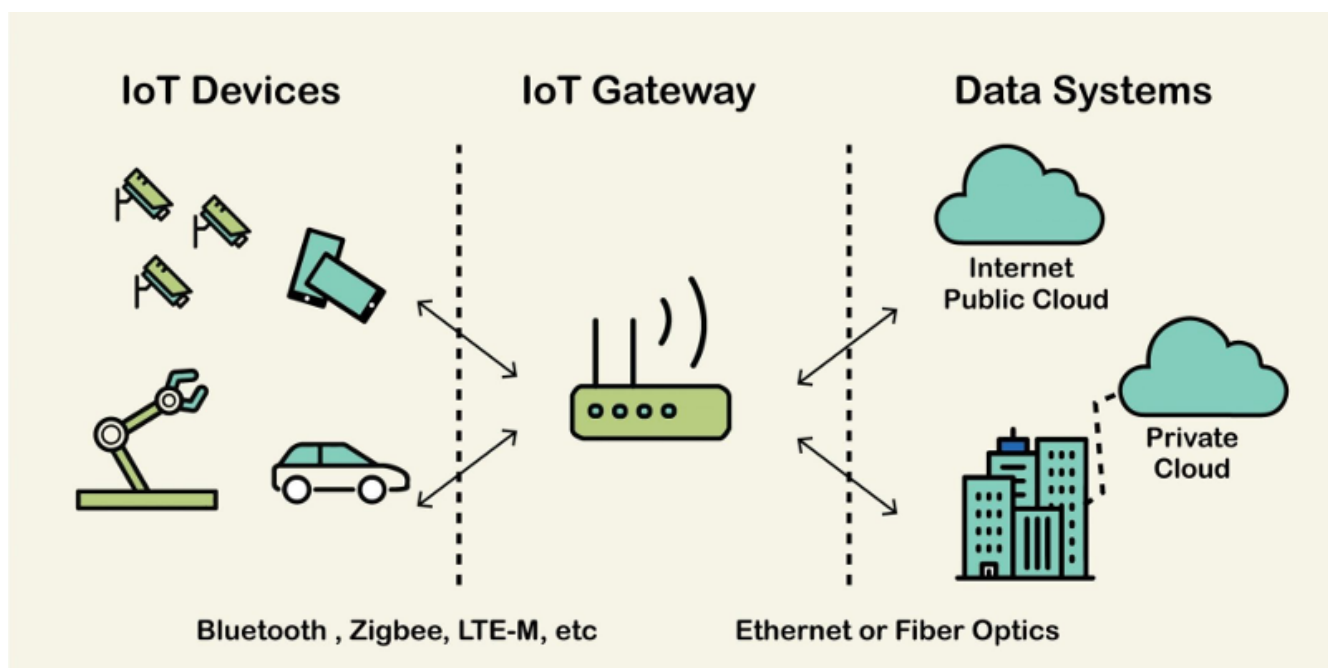


Figura 1. Exemplo de aparelhos IoT se comunicando através de um Gateway

Algumas **sugestões** de tecnologia para o desenvolvimento são:

- Plataforma: Arduino, Raspberry PI, M5 Stack
- Sensor de som tipo FC04
- ESP8266/NodeMCU para comunicações



Figura 2. Sensor de som FC04

## Módulo 2 - Serviços de recebimento e agregação dos dados

Nesse módulo serão desenvolvidas as funcionalidades de captura e processamento das medições. Deve ser previsto o processamento de milhares de dados oriundos dos sensores espalhados na cidade (ou em vários locais, fábricas).

Esse processamento deverá permitir trabalhar com séries temporais de dados e geolocalização. O resultado desse módulo como cálculos, limpeza, etc. será utilizado como insumo para a criação dos dashboards, relatórios. Entre nesse módulo a geração de alertas quando os valores medidos passam de um determinado valor (configurável) durante um período (ajustável também). No escopo dos serviços entra o armazenamento dos dados para execução de queries e relatórios do módulo 3.

Outro ponto a considerar é o monitoramento dos sensores, por exemplo, verificar quais não mandam dados desde um certo intervalo de tempo, ou envia medidas com uma diferença muito grande comparando com aparelhos próximos.

Para o desenvolvimento, seguem algumas **sugestões** de tecnologia:

- Stack baseada em Kafka
  - Entrada dos dados com [MQTT](#) ou [Kafka Rest Proxy](#) enviados para um cluster [Kafka](#).
  - Processamento dos dados com [Kafka Streams](#) ou [Ksql](#)
  - Envio dos dados para uma base que trabalha com séries temporais como [Elasticsearch](#) ou [InfluxDb](#).

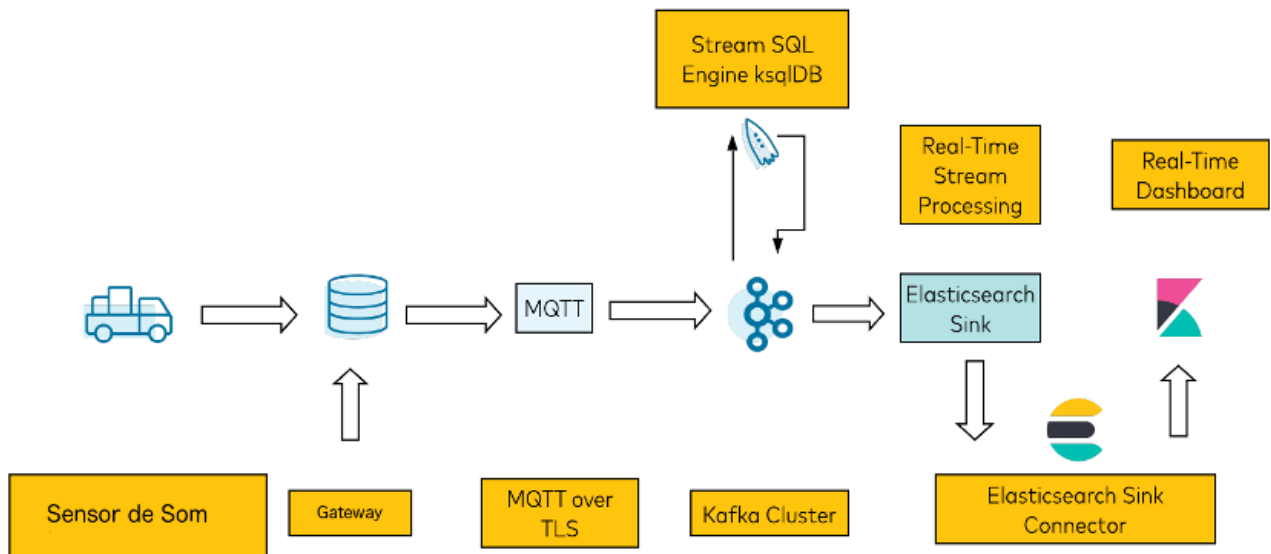


Figura 3. Exemplo de cadeia de tratamento com Kafka

- Stack Elastic
  - Ingestão de dados enviados em filas MQTT ou serviços REST com [Filebeat](#).
  - Processamento com filtros [Logstash](#) e armazenamento no Elasticsearch.

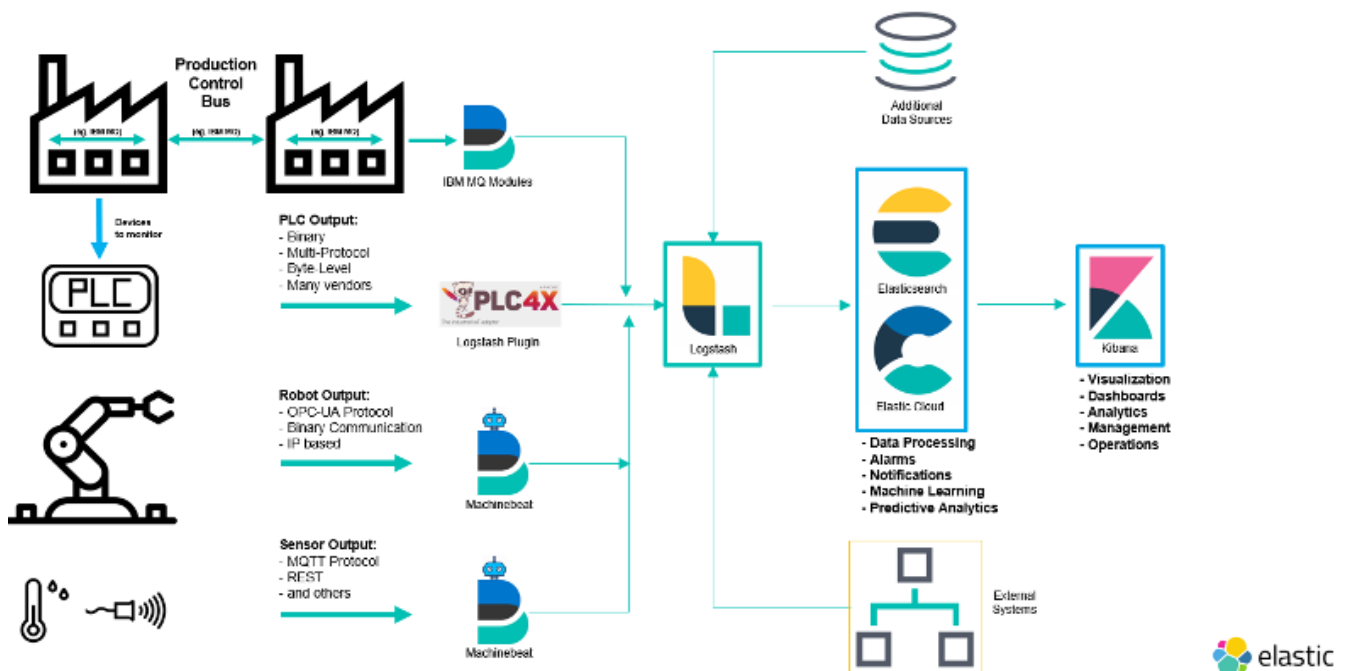


Figura 4. Exemplo de processamento usando a Stack Elastic

- KNative e Serverless
  - Uso do [Broker](#) do [Knative Eventing](#) para receber os dados dos sensores
  - Serviços rodando com [Knative](#) ativados por *Triggers* para efetuar os cálculos. Esses serviços podem ser combinados em fluxos de processamento. Os dados podem ser enviados por qualquer banco que trabalha com séries temporais.



Para um ambiente 100% Kubernetes, pode ser utilizado o [Kubeedge](#)

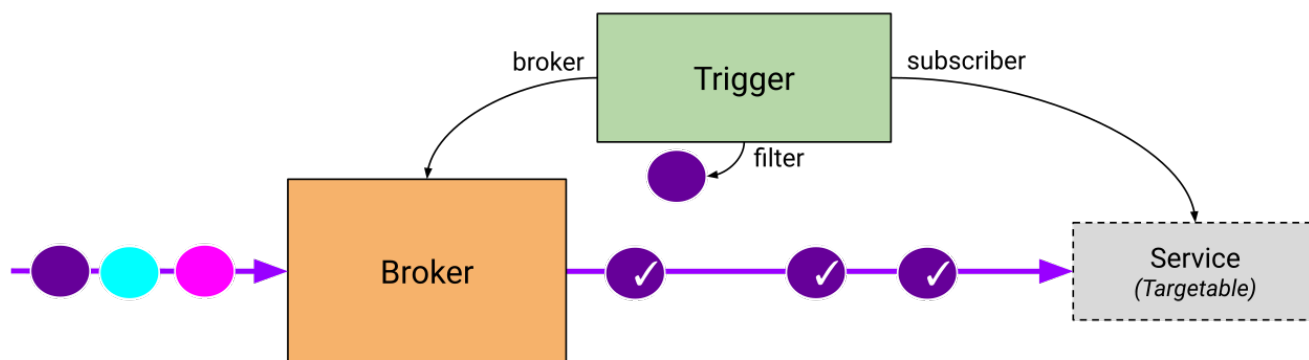


Figura 5. Exemplo de processamento com Knative Eventing

Para os serviços de Streams com Kafka, Elasticsearch ou KNative podem ser utilizados frameworks de mercado como [Spring Boot](#) (com [Spring Cloud Stream](#)) ou [Quarkus](#) (como [Quarkus Funqy](#)). O uso do Python pode ser interessante pois a linguagem possui muitas implementações de bibliotecas científicas úteis no contexto do desafio.

Os possíveis níveis de exposição do volume sonoro que são aceitos por [período de tempo](#) podem ser encontrados no site da [OSHA](#).

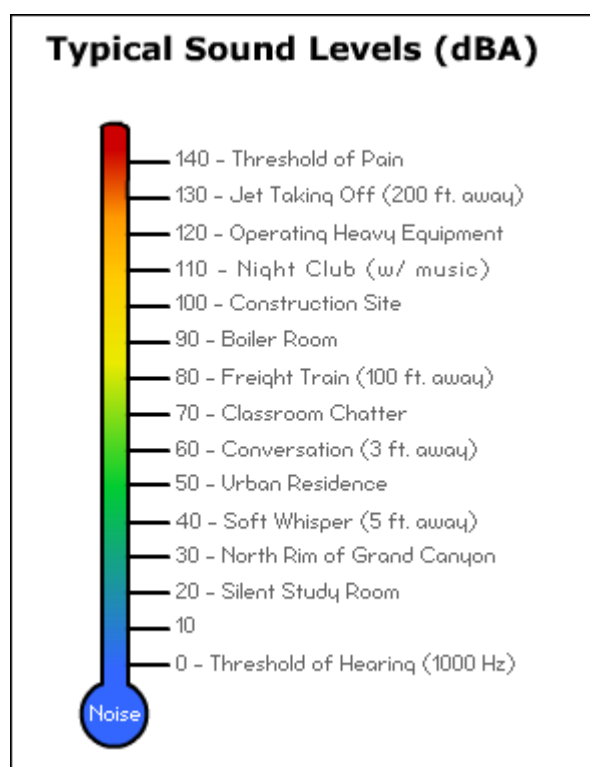


Figura 6. Quão alto é muito alto?

Table 1. Níveis de som aceitos

Duração por dia (horas)	Volume sonoro (dBA)
8	90
6	92
4	95
3	97



Duração por dia (horas)	Volume sonoro (dBA)
2	100
1 1/2	102
1	105
1/2	110
1/4 ou menor	115

## Módulo 3 - Dashboards e relatórios de monitoramento

Nesse módulo serão desenvolvidas as visualizações para acompanhar e monitorar os locais barulhentos. Deverá ser possível visualizar os dados em tempo real em mapas ou plantas. A geração de relatórios como o de locais mais barulhentos por período (por exemplo, uma rua de 10h a 12h, o refeitório de 11h30 a 12h45).

Seguem algumas ferramentas que podem ser utilizadas:

- [Pentaho](#) para criação e visualização de dashboards e relatórios criados a partir de datasources externos.
- [Kibana](#) para criação de dashboards e visualizações a partir de dados do Elasticsearch.
- [Grafana](#) para criação de dashboards

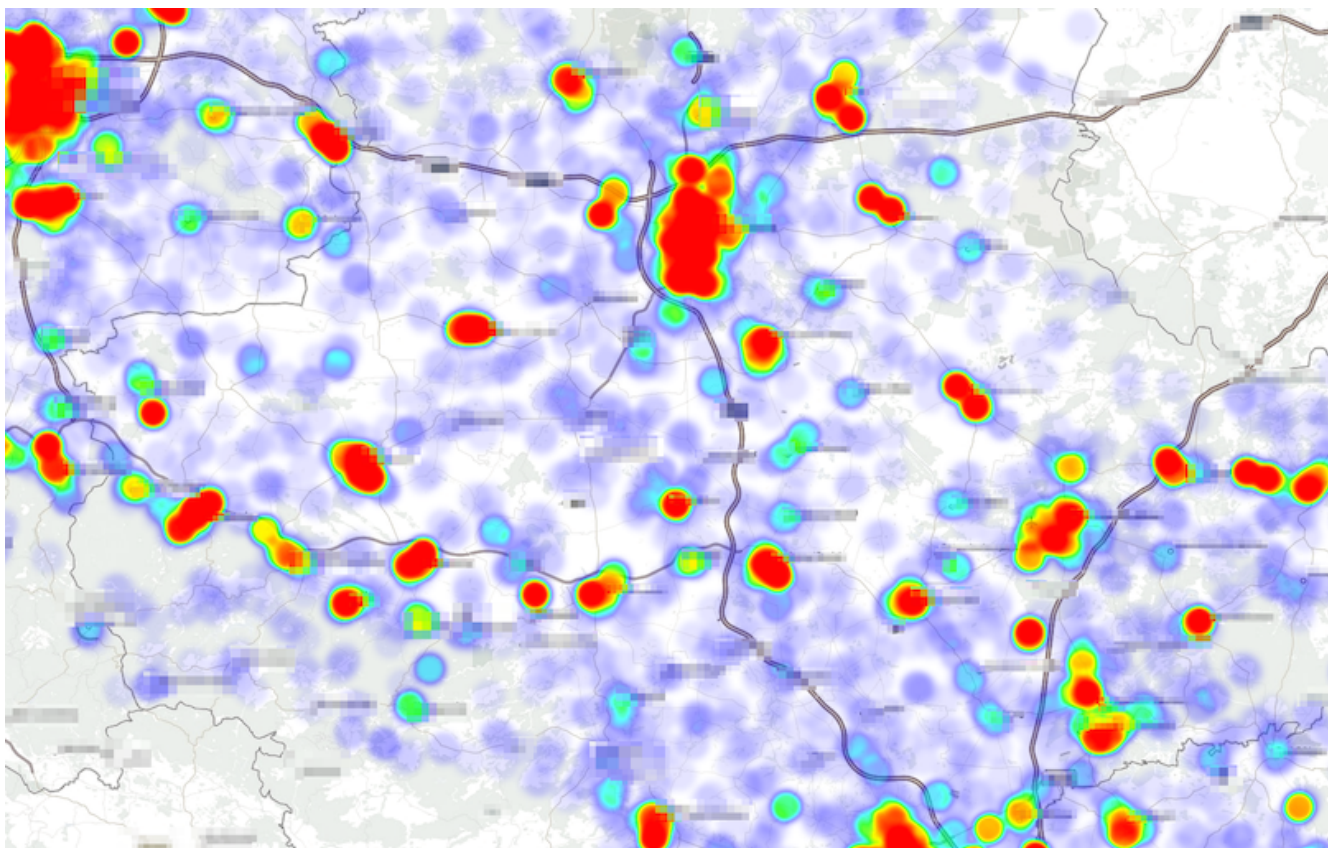


Figura 7. Exemplo de visualização de dados georreferenciados com mapa de calor no Kibana

[5] A plataforma NodeMCU pode ser utilizada para geolocalização, exemplo: <https://circuitdigest.com/microcontroller-projects/>



