



UNIVERSIDADE FEDERAL DE LAVRAS

DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

Graduação em Sistemas de Informação / Ciência da Computação

Monitoramento de Ruído

Relatório de IoT

Bruno Henrique Firmino

Gabriel Furtado Teixeira

Lucas de Castro Nizio

Lavras, 03/07/2025

1. INTRODUÇÃO

Este projeto criou um sistema para monitorar o nível de ruído em duas salas diferentes ao mesmo tempo. O objetivo é saber se o ruído está normal ou muito alto em cada sala, mostrando tudo em tempo real em um site. Isso é útil para vários lugares, como escritórios ou casas, para garantir um ambiente mais tranquilo e seguro.

2. MATERIAIS UTILIZADOS

Para construir o sistema, usamos alguns equipamentos e programas:

2.1. Equipamentos (Hardware)

ESP32 (Heltec WiFi LoRa v2) - 2 unidades: Usamos duas delas, uma para cada sala, para coletar os dados de ruído.

Microfones - 2 unidades: Cada um mede o som em uma sala. Eles enviam o sinal de áudio para o ESP32, que calcula o nível de ruído.

2.2. Programas (Software)

Firestore Realtime Database: É um banco de dados na internet (nuvem) onde os dados de ruído são guardados. Ele é "em tempo real", ou seja, assim que o ESP32 envia um dado, ele já aparece no site.

Linguagens de Programação:

- **HTML**
- **CSS**
- **JavaScript:** Realiza a conexão com o Firestore, lê os dados, atualiza o que aparece na tela e faz as cores mudarem.

3. COMO FUNCIONA (METODOLOGIA)

O sistema foi montado e programado para funcionar assim:

3.1. Conectando os Microfones aos ESP32

Cada microfone foi ligado a um ESP32. O microfone capta o som e o ESP32 mede a intensidade desse som.

3.2. Enviando Dados para o Firebase

Os ESP32s foram programados para:

1. **Conectar na internet:** Cada ESP32 se conecta ao Wi-Fi.
2. **Medir o ruído:** Eles leem o som do microfone e calculam um número que representa o nível de ruído (chamado RMS).
3. **Enviar para o Firebase:** Os ESP32s mandam o valor medido e uma mensagem (ruído normal ou alto) para o Firebase. Para que as salas sejam monitoradas separadamente, uma sala envia dados para um "endereço" no Firebase (/microfone) e a outra para outro "endereço" (/microfone2).

3.3. O Site que Mostra os Dados

O site foi feito para mostrar o ruído em tempo real:

1. **Estrutura:** O site tem espaços para mostrar o ruído de cada sala, com o número do RMS, o status e mensagens de alerta.
2. **Visual:** As cores e o jeito que o site aparece mudam para mostrar o nível de ruído.
3. **Lógica:** O site se conecta ao Firebase e recebe os dados de cada sala. Quando um dado novo chega, ele atualiza a tela da sala correspondente.

3.4. As Cores do Ruído

Para facilitar a visualização, usamos um sistema de cores:

Verde: Significa que o ruído está normal (até um certo nível).

Vermelho: Indica que o ruído está alto (acima de um certo nível).

Cinza : Mostra que o sistema está esperando dados ou que há algum problema de conexão com o firebase.

3.5. Monitoramento Separado

O sistema foi feito para monitorar as duas salas de forma independente. Assim:

- Cada sala tem seu próprio ESP32 e microfone.
- Os dados de cada sala são guardados em lugares diferentes no Firebase.
- O site mostra o que acontece em cada sala sem misturar as informações. Se uma sala estiver com problema, a outra continua funcionando normalmente.

5. CONCLUSÃO

Este projeto mostrou que é possível monitorar o ruído de duas salas de forma simples, eficiente e em tempo real, usando poucos equipamentos e a internet.

Destaques do sistema:

- Cada sala é monitorada separadamente, o que aumenta a precisão.
- O uso de cores facilita a visualização do nível de ruído.
- Os dados são atualizados na hora, graças ao Firebase.
- É fácil adicionar mais salas ou sensores no futuro.

O sistema pode ser útil em diversos ambientes, como escritórios ou residências, ajudando a manter o conforto e identificar ruídos excessivos.

6. Imagens e Links

Firestore



Site



Repositório do GitHub:

<https://github.com/Bruno7k/Trabalho-IoT>

Link do site:

<https://bruno7k.github.io/Trabalho-IoT/>