

# **Documento de Diagrama de Classes**

**Autor:**

**Luís Campos**

**Março, 2023**

# Índice

<b>1. Introdução.....</b>	<b>4</b>
<b>1.1. Descrição do Sistema.....</b>	<b>4</b>
<b>1.2. Objetivo do Diagrama de Classes.....</b>	<b>4</b>
<b>2. Diagrama de Classes .....</b>	<b>5</b>
<b>3. Descrição das Classes .....</b>	<b>6</b>
<b>3.1. Classe Organizador .....</b>	<b>6</b>
<b>3.2. Classe Alerta .....</b>	<b>6</b>
<b>3.3. Classe Sensor .....</b>	<b>7</b>
<b>3.4. Classe Espaco .....</b>	<b>7</b>
<b>4. Relações .....</b>	<b>8</b>

## Índice de Figuras

Figura 1 - Diagrama de Classes.....	5
-------------------------------------	---

# 1. Introdução

## 1.1. Descrição do Sistema

O Sistema de Monitoramento de Eventos é um sistema web que permite monitorar eventos em tempo real. O sistema coleta dados de sensores de temperatura, humidade, qualidade do ar e ruído, e os apresenta em um dashboard para os organizadores do evento. Os organizadores também podem configurar alertas para serem notificados quando os valores dos sensores ultrapassam limites predefinidos.

## 1.2. Objetivo do Diagrama de Classes

O diagrama de classes documenta as classes principais do sistema e seus relacionamentos. O objetivo do diagrama é descrever a estrutura do sistema, facilitar a comunicação entre os membros da equipe de desenvolvimento e documentar as regras de negócio do sistema.

## 2. Diagrama de Classes

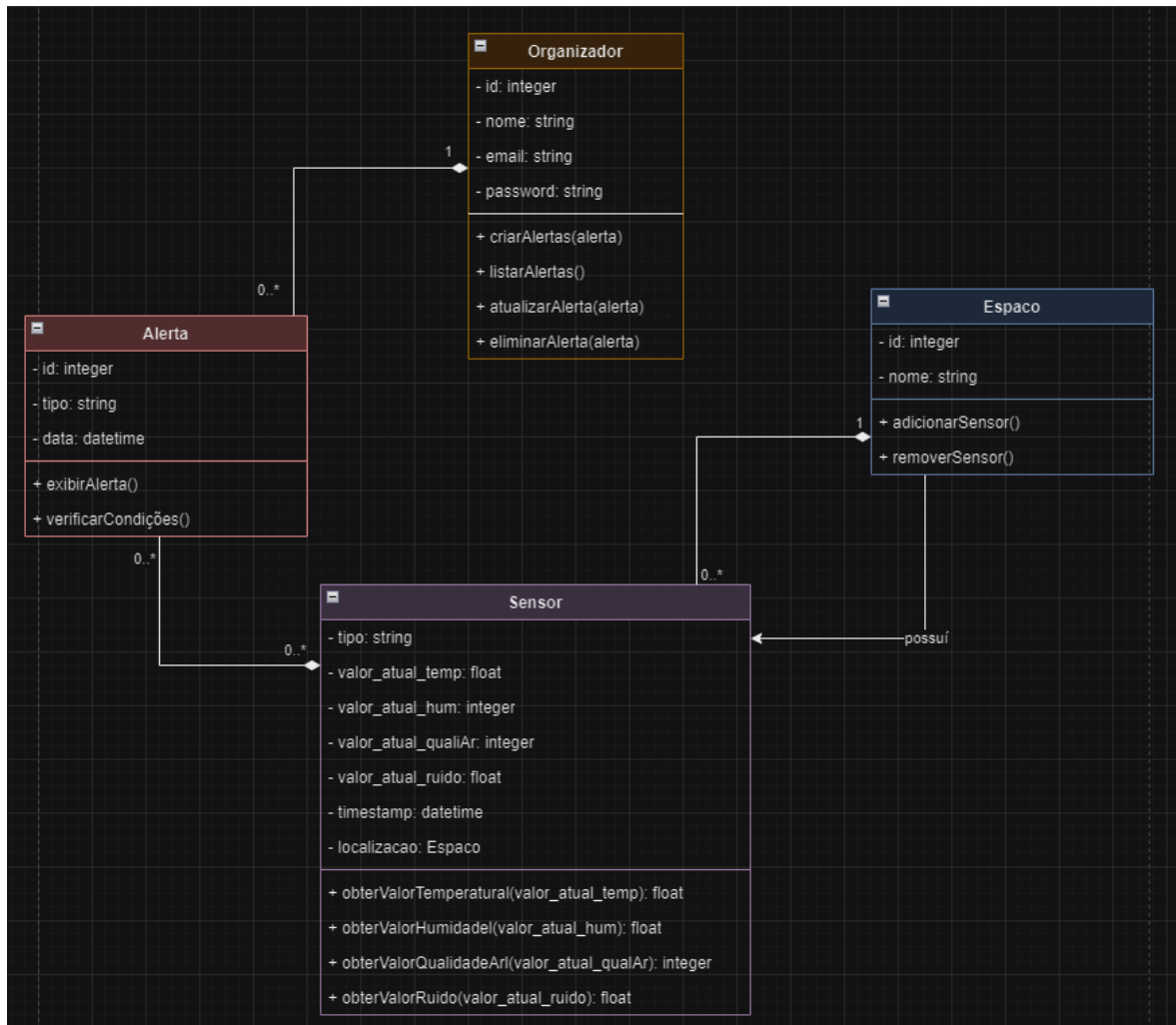


Figura 1 - Diagrama de Classes

## 3. Descrição das Classes

### 3.1. Classe Organizador

- **Descrição:**

Representa a entidade que vai gerir os alertas e dados para assim agir conforme o necessário para manter as condições de conforto.

- **Atributos**

- id: integer : Identificador único do organizador.
- nome: string : Nome do organizador.
- email: string : E-mail do organizador.
- password: string : Password do organizador

- **Métodos**

- criarAlertas(alerta) : Alerta:
- listarAlertas() :
- atualizarAlerta(alerta) : Alerta :
- eliminarAlerta(alerta) : Alerta :

### 3.2. Classe Alerta

- **Descrição:**

Representa um alerta e o seu tipo que é exibido ocorra alguma variação nas condições de conforto e também fornece a data e hora do ocorrido.

- **Atributos:**

- id: int : Identificador único do Alerta.
- tipo: string : Define o tipo de alerta.
- data\_hora: datetime : data e a hora que ocorreu o alerta

- **Métodos:**

- **exibirAlerta()** : Exibe um alerta caso algum dado dos sensores esteja fora do normal.
- **verificarCondições()** : verifica se os dados estão fora do normal recebidos dos sensores estão fora do normal.

### 3.3. Classe Sensor

- **Descrição:**

Representa um dispositivo que coleta dados do ambiente.

- **Atributos:**

- **tipo: string** : Tipo do sensor (ex: temperatura, ruído, etc..).
- **valor\_atual\_temp: float** : valor medido pelo sensor exclusivamente de temperatura.
- **valor\_atual\_hum: integer** : valor medido exclusivamente pelo sensor de humidade.
- **valor\_atual\_qualAr: integer** : valor medido exclusivamente pelo sensor de qualidade do ar.
- **valor\_atual\_ruído: float** : valor medido exclusivamente pelo sensor de ruído.
- **timestamp: datetime** : o timestamp das medições feitas pelo sensor.
- **localizacao: Espaco** : localização do sensor em um espaco especifico do evento.

- **Métodos:**

- **obterValorTemperatural(valor\_atual\_temp): float** :
- **obterValorHumidadel(valor\_atual\_hum): float** :
- **obterValorQualidadeArl(valor\_atual\_qualAr): integer** :
- **obterValorRuído(valor\_atual\_ruído): float** :

### 3.4. Classe Espaco

- **Descrição:**

Representa um espaço físico dentro do evento (ex: sala, corredor) onde estam instalados os sensores.

- **Atributos:**

- **id: integer** : Identificador único do evento.
- **nome: string** : Nome do evento.

- **Métodos:**

- **adicionarSensor(sensor): Sensor** : Adiciona um sensor ao espaço.
- **removerSensor(sensor): Sensor** : Remove ao espaço.

## 4. Relações

### Organizador <-> Alerta

- Relação: 1 -> 0..\*
- Descrição: Um organizador pode criar e gerir vários alertas. Esta relação é representada pela multiplicidade 1 para "Organizador" e 0..\* para "Alerta", indicando que um organizador deve criar pelo menos um alerta, mas pode criar vários, enquanto um alerta pode estar associado a um ou mais organizadores.

### Espaco <-> Sensor

- Relação: 1 -> 0..\*
- Descrição: Um espaço pode ter um ou mais sensores. A multiplicidade 1 para "Espaco" e 0..\* para "Sensor" sugere que cada espaço possui vários sensores, mas cada sensor está associado a um único espaço.

### Sensor <-> Alerta

- Relação: 1..\* -> 0..\*
- Descrição: Um sensor pode gerar vários alertas e cada alerta pode estar associado a um ou mais sensores. A multiplicidade 1..\* para "Sensor" e 0..\* para "Alerta" indica que cada sensor pode gerar um ou mais alertas, enquanto um alerta pode ser gerado por um ou mais sensores.