

# Diagrama de Caso de UML - Sistema Automatizado de Monitoramento Ambiental (SAME)

Link da UML: [Projeto](#)

Sistema de Monitoramento Ambiental (Temperatura, Umidade e Qualidade do Ar)

## 1. Visão Geral do Sistema

O sistema tem como finalidade monitorar dados ambientais (temperatura, umidade e qualidade do ar) enviados por sensores IoT. Os dados são recebidos pelo servidor Django via HTTP ou MQTT, armazenados no banco de dados e exibidos em dashboards atualizados em tempo real. Além disso, o sistema emite alertas quando valores ultrapassam limites críticos.

A fronteira do sistema é a aplicação web/Dashboard desenvolvida em Django.

## 2. Atores do Sistema

### Atores Primários (interagem diretamente com a aplicação)

- **Usuário Administrador**  
Gerencia sensores, visualiza dashboards, ajusta limites críticos e recebe alertas.
- **Usuário Comum / Técnico**  
Consulta gráficos, acompanha leituras e visualiza históricos; possui permissões reduzidas.

### Atores Secundários (interação indireta, automatizada ou externa)

- **Dispositivo IoT (Sensor)**  
Envia leituras ao sistema via HTTP/MQTT.
- **Serviço de Notificação**  
APIs externas usadas para envio de:
  - E-mail
  - Telegram
  - WhatsApp
- **Servidor MQTT (Broker)**  
Responsável por retransmitir mensagens dos sensores para o sistema.

### **3. Casos de Uso (iniciados por verbos)**

#### **Para Usuário Administrador**

1. Cadastrar sensor
2. Editar sensor
3. Remover sensor
4. Configurar limites críticos
5. Visualizar dashboard em tempo real
6. Gerar relatório de leituras
7. Gerenciar usuários
  - (opcional: pode ser considerado extensão <<extend>>)

#### **Para Usuário Comum / Técnico**

1. Visualizar dashboard em tempo real
2. Consultar histórico de leituras
3. Filtrar dados por período

#### **Para Dispositivo IoT**

1. Enviar leitura ambiental
  - <<include>> Validar dados recebidos
  - <<include>> Armazenar leitura no banco
  - <<extend>> Acionar sistema de alertas

#### **Para Sistema / Serviços Automáticos**

1. Verificar limites críticos
  - <<extend>> Emitir alerta
2. Enviar notificação externa  
(via serviços de e-mail, WhatsApp, Telegram)

### **4. Relacionamentos UML a serem aplicados no diagrama**

#### **Associação**

Conecta atores aos casos de uso que executam.

## <<include>>

Usado para processos **obrigatórios** executados sempre que um caso de uso ocorre. Exemplo:

- “Enviar leitura ambiental” **inclui** “Validar dados”.
- “Enviar leitura ambiental” **inclui** “Armazenar leitura”.

## <<extend>>

Usado para fluxos **opcionais** ou condicionais. Exemplo:

- “Accionar sistema de alertas” estende “Enviar leitura ambiental” **somente** se limites forem ultrapassados.
- “Enviar notificação externa” estende “Emitir alerta”.

# 5. Diagrama de Caso de Uso UML (Modelado em Texto – UML/Mermaid)

Se quiser, posso gerar a versão gráfica.

```
usecaseDiagram
    actor "Usuário Administrador" as Admin
    actor "Usuário Comum" as User
    actor "Dispositivo IoT" as IoT
    actor "Serviço de Notificação" as Notif
    actor "Servidor MQTT" as Broker

    rectangle "Sistema de Monitoramento Ambiental" {
        (Cadastrar sensor) as UC1
        (Editar sensor) as UC2
        (Remover sensor) as UC3
        (Configurar limites críticos) as UC4
        (Visualizar dashboard em tempo real) as UC5
        (Gerar relatório de leituras) as UC6
        (Gerenciar usuários) as UC7

        (Enviar leitura ambiental) as UC8
        (Validar dados recebidos) as UC9
        (Armazenar leitura no banco) as UC10
        (Accionar sistema de alertas) as UC11

        (Consultar histórico de leituras) as UC12
        (Filtrar dados por período) as UC13

        (Verificar limites críticos) as UC14
        (Emitir alerta) as UC15
        (Enviar notificação externa) as UC16
    }

    Admin --> UC1
    Admin --> UC2
    Admin --> UC3
    Admin --> UC4
    Admin --> UC5
    Admin --> UC6
    Admin --> UC7
```

```
User --> UC5
User --> UC12
User --> UC13

IoT --> UC8
Broker --> UC8

UC8 --> UC9 : <<include>>
UC8 --> UC10 : <<include>>
UC8 -- UC11 : <<extend>>

UC14 -- UC15 : <<extend>>
UC15 -- UC16 : <<extend>>

Admin --> UC14
```

## 6. Resumo Final

O diagrama apresenta:

- fronteira do sistema
- atores primários e secundários
- casos de uso iniciados por verbos
- relacionamentos obrigatórios (include)
- fluxos opcionais (extend)
- visão geral completa da interação com sensores IoT e serviços externos