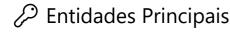
README.md 8/25/2025



# Sistema de Gestão de Ativos e Serviços

Este projeto tem como objetivo oferecer uma estrutura completa para o gerenciamento de ativos, serviços técnicos e fluxo operacional com controle de SLA.



#### **Cliente**

Representa a organização ou pessoa que possui ativos e solicita serviços.

- Atributos: id, nome, cnpj, contatos
- Relacionamentos:
  - Possui vários Ativos
  - Solicita vários Serviços

#### 

Técnicos, gestores ou solicitantes envolvidos no ciclo de atendimento.

- Atributos: id, nome, cargo, email, telefone
- Relacionamentos:
  - Pode ser **Solicitante** de serviços
  - Pode ser **Responsável** pela execução

#### P Local

Define onde os ativos estão alocados (ex: cidade, filial, prédio).

- Atributos: id, nome
- Relacionamentos:
  - Contém vários Ativos



Equipamento, sistema ou recurso monitorado.

- Atributos: id, codigo, nome, tipo, status
- Relacionamentos:
  - Pertence a um Cliente
  - o Está alocado em um Local
  - Associado a vários Serviços

### Tipo de Serviço

Classificação dos serviços com regras de SLA.

Atributos: id, nome, descricao, tempo\_medio, sla\_horas

README.md 8/25/2025

- Relacionamentos:
  - Classifica vários Serviços
- Regras
  - o Permite verificar se um serviço foi concluído dentro do SLA



Chamado, manutenção ou atendimento técnico.

- Atributos: id, titulo, descricao, status, data\_inicio, data\_fim
- Relacionamentos:
  - Associado a um Cliente
  - Vinculado a um Ativo
  - Possui um **Tipo de Serviço**
  - Possui um **Solicitante** e um **Responsável** (Usuários)

## Estrutura no Node.js

```
src/
|— app.js
 - config/
   └─ database.js
— models/
    — cliente.js
    ├─ usuario.js
    ├─ local.js
    ├─ ativo.js
     — tipoServico.js
    └─ servico.js
  - routes/
    — clientes.routes.js
    usuarios.routes.js
    ├─ locais.routes.js
    ├─ ativos.routes.js
     tiposServicos.routes.js
    └─ servicos.routes.js
 — controllers/
    clientes.controller.js
     usuarios.controller.js
    — ativos.controller.js
     servicos.controller.js
```

## Fluxo do Ciclo de Vida de um Serviço

```
Aberto → Em andamento → Concluído → Encerrado
--
```

README.md	8/25/2025