

Nome: Anderson Almeida da Silveira

MONITORAMENTO DE REDE

O seguinte programa visa monitorar os equipamentos de rede de uma empresa e registrar falhas e eventos que podem ter acontecido aos mesmos. Existem dois tipos de equipamentos. Os equipamentos de infraestrutura (Infra) como modems e roteadores em que podemos registrar falhas (Falha) como “modem saiu do ar” ou “roteador sobrecarregado”. Enquanto os equipamentos para o usuário final (UsoFinal) podem ser desktops, impressoras, notebooks onde além de podermos registrar falhas como “Impressora com papel atolado”, “monitor com defeito” também registramos usuários (Usuario) que estejam cadastrados na empresa (Funcionário) e tem autorização para usar o equipamento.

Os equipamentos e funcionários são salvos no banco de dados através de funções estáticas como por exemplo

```
//Nome, mac, ip
Infra.salvarNovo("Roteador", "r5634", "12345");

//Nome, mac, ip
UsoFinal.salvarNovo("PC supimpa", "34728", "123.344");

//Cpf, nome, cargo
Funcionario.SalvarNovo("12345", "Pedrinho", "programador");
```

Os dados salvos são recuperados em grupo também através de funções estáticas como

```
Infra.getAll();
```

Usuarios podem ser adicionados ou removidos de um objeto (impressora por exemplo) tendo o cpf do mesmo e usando os comandos

```
impressora.registrarUsuario("123456");
impressora.removerUsuario("123456");
```

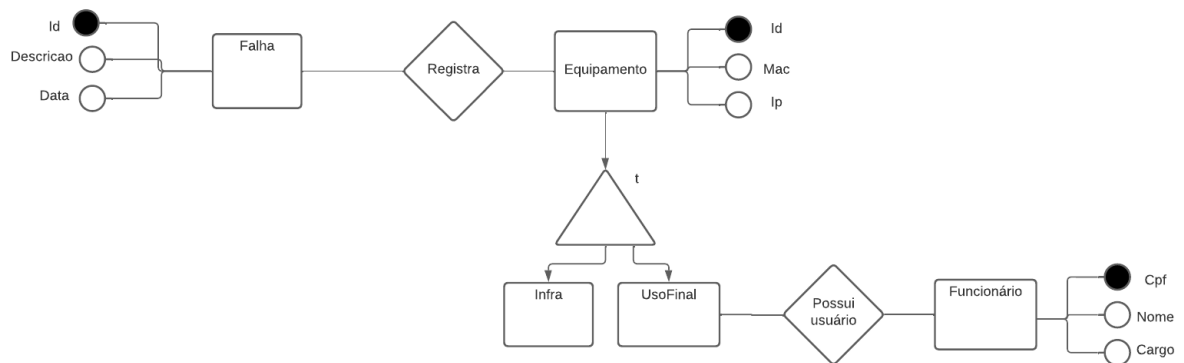
Para registrar falhas temos um comando similar

```
// Dia, mês, ano, descrição
impressora.registrarFalha(3,2,2023,"Atolou papel");
```

Todos os objetos podem ser apagados do banco de dados com o método apagar() e caso um objeto das classes Funcionario ou UsoFinal tenha uma entrada na tabela de usuários a mesma será removida.

As classes estão organizadas em uma versão simplificada do padrão MVC. A lógica de acesso ao banco de dados está dentro das classes Controller enquanto a lógica da aplicação como listar os dados de forma amigável fica nas classes principais.

Segue abaixo o diagrama de relacionamento do BD e em seguida os comandos SQLs para criá-lo no PostgreSQL.



```
CREATE TABLE IF NOT EXISTS Infra (  
id SERIAL PRIMARY KEY,  
nome character varying not NULL,  
mac character varying NOT NULL,  
ip character varying  
);
```

```
CREATE TABLE IF NOT EXISTS Usfinal (  
id SERIAL PRIMARY KEY,  
nome character varying not NULL,  
mac character varying NOT NULL,  
ip character varying
```

);

```
CREATE TABLE IF NOT EXISTS Falha (  
  id SERIAL PRIMARY KEY,  
  infra_id INT REFERENCES Infra(id),  
  usofinal_id INT REFERENCES Usofinal(id),  
  dataFalha date NOT NULL,  
  descricao character varying NOT NULL  
);
```

```
CREATE TABLE IF NOT EXISTS Funcionario(  
  cpf character varying PRIMARY KEY,  
  nome character varying NOT NULL,  
  cargo character varying NOT NULL  
);
```

```
CREATE TABLE IF NOT EXISTS Usuario(  
  cpf character varying REFERENCES Funcionario(cpf),  
  id int REFERENCES Usofinal(id)  
);
```

```
ALTER TABLE Falha ADD CONSTRAINT um_ou_outro CHECK (infra_id IS NOT NULL OR usofinal_id IS NOT NULL);
```

```
ALTER TABLE Falha ADD CONSTRAINT um_ou_outro_null CHECK (infra_id IS NULL OR usofinal_id IS NULL);
```

```
ALTER TABLE Usuario ADD CONSTRAINT pk_usuario PRIMARY KEY (cpf,id);
```