



Universidade de Brasília

INSTITUTO DE EXATAS - IE

DEPARTAMENTO DE CIÊNCIAS DA COMPUTAÇÃO - CIC

BANCO DE DADOS 1/2025

Projeto do Banco de Dados

Discentes: Bruno Eduardo dos Santos - 211066249

Matheus de Melo Fellet - 222015201

Gustavo Alencar Valadares - 232002548

Introdução

Este projeto apresenta o Sistema de Controle de Qualidade de Patrimônio (SCQP), desenvolvido com o objetivo de otimizar a gestão dos bens patrimoniais da organização. O SCQP integra informações detalhadas sobre espaços físicos (salas e edifícios) e seus respectivos equipamentos, proporcionando uma visão abrangente e atualizada do patrimônio.

Adicionalmente, o sistema monitora o status operacional desses ativos, identificando equipamentos em funcionamento ou com avarias, o que possibilita a implementação de manutenção preventiva e a redução de interrupções operacionais. Uma funcionalidade essencial do SCQP é o registro estruturado de ocorrências, permitindo que usuários reportem problemas, os quais são detalhadamente documentados para otimizar o fluxo de trabalho das equipes de manutenção.

Descrição

O sistema será composto por 10 entidades interconectadas, cuidadosamente selecionadas para o escopo do projeto. As entidades e suas breves descrições seguem abaixo.

Tabela de entidades

Entidade	Descrição
Sala	Local físico onde os equipamentos estão instalados.
Equipamento	Patrimônio alocado em uma sala (cadeiras, ventiladores, projetores etc.).
TipoEquipamento	Classificação do equipamento (ex.: Cadeira, Projetor, Tomada).
StatusEquipamento	Estado atual do equipamento (Funcionando, Quebrado, Em manutenção).
Manutencao	Registro das intervenções realizadas nos equipamentos.
Funcionario	Pessoa responsável pela operação e manutenção dos equipamentos.
Departamento	Unidade organizacional responsável pela sala ou pelos equipamentos (ex.: TI, Engenharia).
Predio	Edifício que agrupa várias salas.
Usuario	Usuário do sistema, que pode reportar problemas ou solicitar serviços.
Ocorrencia	Registro de defeito ou problema aberto para um equipamento.



Dando seguimento, os relacionamentos foram detalhados em um nível mais baixo. Isso explicita as referências dentro do sistema.

Relacionamentos

Predio(id, nome, local)

Sala(codigo, nome, foto_binaria, id_predio)

id_predio relaciona Predio.id

TipoEquipamento(id, descricao)

StatusEquipamento(id, descricao)

Equipamento(id, data_aquisicao, fabricante, descricao, id_tipo_equipamento, id_status_equipamento, id_sala)

id_tipo_equipamento relaciona TipoEquipamento.id

id_status_equipamento relaciona StatusEquipamento.id

id_sala relaciona Sala.codigo

Usuario(cpf, nome, email, senha)

Funcionario(cpf_usuario, cargo)

cpf_usuario relaciona Usuario.cpf

Docentes(cpf_usuario, area)

cpf_usuario relaciona Usuario.cpf

Ocorrencia(codigo, problema, data_abertura, data_fechamento, cpf_usuario)

cpf_usuario relaciona Usuario.cpf

Manutencao(id, servico, data, id_equipamento, cpf_funcionario, id_ocorrencia)

id_equipamento relaciona Equipamento.id

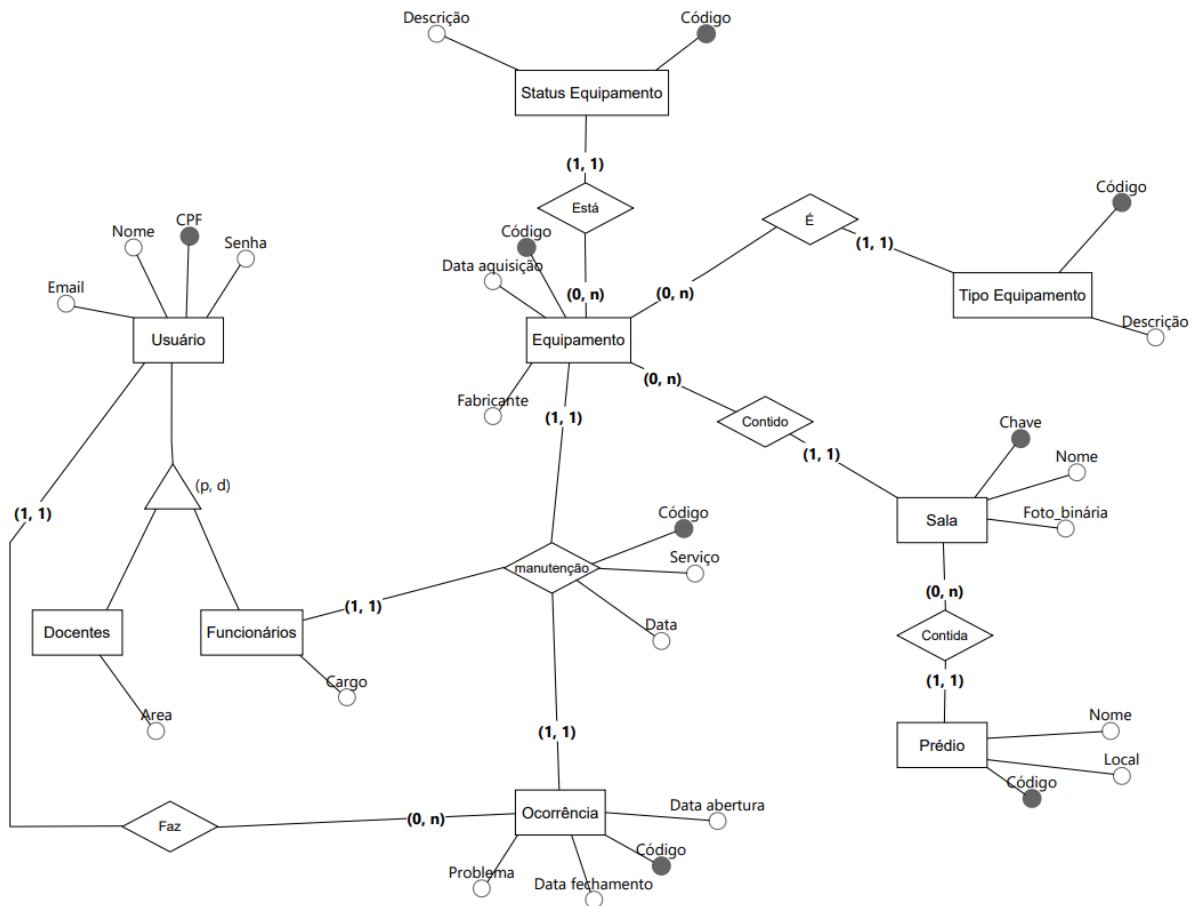
cpf_funcionario relaciona Funcionario.cpf_usuario

id_ocorrencia relaciona Ocorrencia.codigo



Modelo entidade relacionamento

O modelo entidade–relacionamento do sistema foi concebido a partir da tabela inicial. Ele passou por refinamentos e discussões entre os integrantes do grupo, garantindo sua funcionalidade e coerência. Ao final desse processo colaborativo, obteve-se o diagrama ER definitivo.

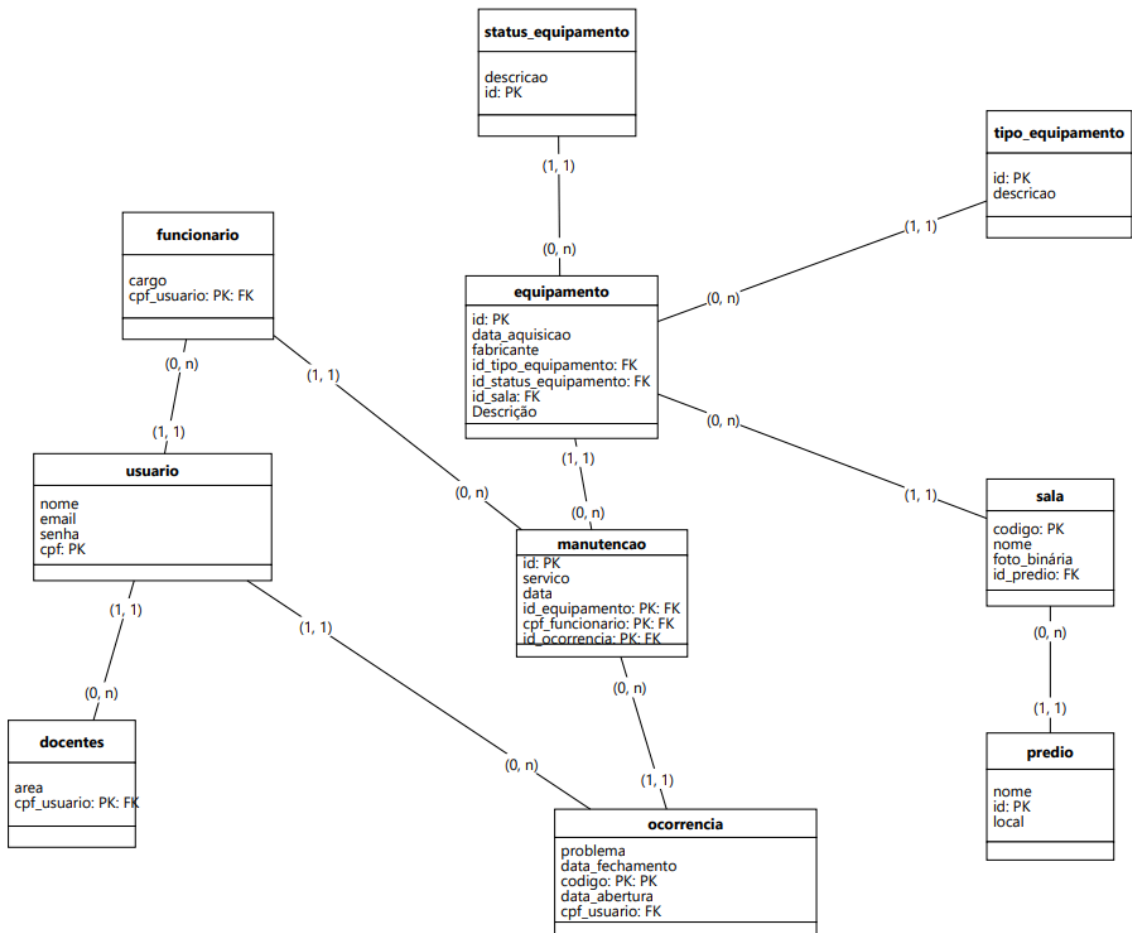




Modelo relacional (Modelo Lógico)

Em seguida, o diagrama ER foi convertido para o modelo relacional, também chamado de modelo lógico. Nesta etapa, já se trabalham com tabelas, atributos e relacionamentos em termos de banco de dados. No BRModelo, é possível especificar tipos de dados, restrições de integridade e demais regras de negócio, permitindo a geração automática do script SQL para a criação do banco de dados.

O modelo relacional foi derivado diretamente do modelo entidade relacionamento, porém ao fazer a conversão o BRModelo trouxe alguns erros. Esses erros foram corrigidos manualmente para o modelo refletir o que precisávamos.





Script

Finalmente chegamos à etapa de implementação com o script SQL. Embora o BRModelo permita gerar automaticamente o código a partir do modelo relacional, enfrentamos inconsistências na conversão: diversas vezes a ferramenta alterou indevidamente os tipos de dados, resultando em definições incorretas. Para resolver o problema, revisamos manualmente o script gerado, ajustamos todos os tipos e valores que estavam equivocados e, após essas correções, chegamos ao resultado abaixo:

```
CREATE TABLE equipamento (
  id INT PRIMARY KEY AUTO_INCREMENT,
  data_aquisicao DATE NOT NULL,
  fabricante VARCHAR(100) NOT NULL, -- Ajustado (n)
  id_tipo_equipamento INT NOT NULL,
  id_status_equipamento INT NOT NULL,
  id_sala INT NOT NULL,
  Descricao VARCHAR(255) NOT NULL -- Ajustado (n)
);

CREATE TABLE tipo_equipamento (
  id INT PRIMARY KEY AUTO_INCREMENT,
  descricao VARCHAR(100) NOT NULL -- Ajustado (n)
);

CREATE TABLE status_equipamento (
  descricao VARCHAR(100) NOT NULL, -- Ajustado (n)
  id INT PRIMARY KEY AUTO_INCREMENT
);

CREATE TABLE ocorrencia (
  problema VARCHAR(500) NOT NULL, -- Ajustado (n) para permitir mais texto
  data_fechamento DATE,
  codigo INT PRIMARY KEY AUTO_INCREMENT,
  data_abertura DATE NOT NULL,
  cpf_usuario VARCHAR(14) NOT NULL -- CORRIGIDO: Para corresponder a usuario.cpf
);

CREATE TABLE sala (
  codigo INT PRIMARY KEY,
  nome VARCHAR(100), -- CORRIGIDO: Nome de sala deve ser texto
  foto_binaria BLOB, -- CORRIGIDO: Para armazenar dados binários. Se for um caminho/URL, use VARCHAR(255)
  id_predio INT
);

CREATE TABLE predio (
  nome VARCHAR(100) NOT NULL, -- Ajustado (n)
  id INT PRIMARY KEY,
  local VARCHAR(255) NOT NULL -- Ajustado (n)
);

CREATE TABLE usuario (
  nome VARCHAR(100) NOT NULL, -- Ajustado (n)
  email VARCHAR(100) NOT NULL, -- Ajustado (n)
  senha VARCHAR(255) NOT NULL, -- Ajustado (n) - Lembre-se de armazenar HASH da senha
  cpf VARCHAR(14) PRIMARY KEY, -- CORRIGIDO: CPF é melhor como VARCHAR para incluir zeros à esquerda e formatação
  UNIQUE (email)
);

CREATE TABLE manutencao (
  id INT PRIMARY KEY AUTO_INCREMENT,
  servico VARCHAR(500) NOT NULL, -- Ajustado (n) para permitir mais texto
  data DATE NOT NULL, -- OK!
  id_equipamento INT NOT NULL,
  cpf_funcionario VARCHAR(14) NOT NULL, -- CORRIGIDO: Para corresponder a funcionario.cpf_usuario -> usuario.cpf
```



Universidade de Brasília

INSTITUTO DE EXATAS - IE

DEPARTAMENTO DE CIÊNCIAS DA COMPUTAÇÃO - CIC

BANCO DE DADOS 1/2025

```
id_ocorrencia INT -- Alterado para permitir NULL, pois uma manutenção pode não estar ligada a uma ocorrência
(ex: preventiva).
);

CREATE TABLE docentes (
  area VARCHAR(100) NOT NULL, -- Ajustado (n)
  cpf_usuario VARCHAR(14) PRIMARY KEY -- CORRIGIDO: Para corresponder a usuario.cpf
);

CREATE TABLE funcionario (
  cargo VARCHAR(100) NOT NULL, -- Ajustado (n)
  cpf_usuario VARCHAR(14) PRIMARY KEY -- CORRIGIDO: Para corresponder a usuario.cpf
);

-- Equipamento FKs
ALTER TABLE equipamento ADD FOREIGN KEY(id_tipo_equipamento) REFERENCES tipo_equipamento(id);
ALTER TABLE equipamento ADD FOREIGN KEY(id_status_equipamento) REFERENCES status_equipamento(id);
ALTER TABLE equipamento ADD FOREIGN KEY(id_sala) REFERENCES sala(Codigo); -- 'Codigo' é a PK de sala

-- Ocorrencia FK
ALTER TABLE ocorrencia ADD FOREIGN KEY(cpf_usuario) REFERENCES usuario(cpf); -- 'cpf' é a PK de usuario

-- Sala FK
ALTER TABLE sala ADD FOREIGN KEY(idPrédio) REFERENCES predio(id); -- 'id' é a PK de predio

-- Manutencao FKs
ALTER TABLE manutencao ADD FOREIGN KEY(id_equipamento) REFERENCES equipamento(id); -- 'id' é a PK de equipamento
ALTER TABLE manutencao ADD FOREIGN KEY(cpf_funcionario) REFERENCES funcionario(cpf_usuario); -- 'cpf_usuario' é a
PK de funcionario
ALTER TABLE manutencao ADD FOREIGN KEY(id_ocorrencia) REFERENCES ocorrencia(codigo); -- 'codigo' é a PK de
ocorrencia

-- Docentes FK
ALTER TABLE docentes ADD FOREIGN KEY(cpf_usuario) REFERENCES usuario(cpf); -- 'cpf' é a PK de usuario

-- Funcionario FK
ALTER TABLE funcionario ADD FOREIGN KEY(cpf_usuario) REFERENCES usuario(cpf); -- 'cpf' é a PK de usuario
```