**Banco de Dados**

**Para que serve:**

O banco de dados tem como principal função armazenar, organizar e gerenciar as informações utilizadas pelo sistema de forma estruturada e segura. Ele permite que os dados sejam acessados, inseridos, atualizados e removidos de maneira eficiente, garantindo integridade e consistência nas operações realizadas. No contexto do projeto, o banco de dados serve como a base central onde todas as informações essenciais como registros de usuários, transações, ou qualquer outro dado relevante são mantidas, possibilitando a comunicação entre as diferentes partes do sistema e assegurando que os dados estejam sempre disponíveis quando necessários.

Segundo Heuser (2009) afirma que “um banco de dados é uma coleção de dados inter-relacionados, organizada de forma a permitir fácil acesso, gerenciamento e atualização”. Essa definição destaca o quão crucial é ter uma estrutura bem organizada no banco de dados para assegurar a eficiência e a confiabilidade do sistema como um todo.

**Como foi utilizado:**

No desenvolvimento do projeto, utilizou-se o brModelo para modelar o banco de dados relacional. Através dessa ferramenta, foi possível criar o modelo entidade-relacionamento (MER), definindo as principais entidades do sistema, seus atributos e os relacionamentos que existem entre esses elementos. Isso possibilitava uma melhor compreensão do esquema do banco e que o modelo estivesse dentro das regras de negócio do projeto.

Depois da modelagem conceitual e lógica, um script SQL foi gerado com os comandos de CREATE TABLE, que posteriormente foi modificado e executado em um SGBD. Utilizaram SQL tanto para a estruturação dos bancos quanto para a manipulação através de comandos DML, como INSERT, SELECT, UPDATE e DELETE.

Dessa maneira, foi possível obter uma base de dados que estivesse com a estrutura, integridade e consistência necessárias para a utilização em conjunto com os outros sistemas.

O funcionário será cadastrado com algumas informações, nome, email, senha, setor e RE, como no é mostrado abaixo

Texto preto sobre fundo branco

O conteúdo gerado por IA pode estar incorreto.

Diagrama

O conteúdo gerado por IA pode estar incorreto.E os chamados serão armazenados e registrados de acordo com o seu titulo, a sua descrição, status, nível de atendimento e sua data de abertura e fechamento

Diagrama

O conteúdo gerado por IA pode estar incorreto.Os chamados serão abertos pelos funcionários, e resolvidos pelos técnicos ou pela AI, de acordo com o nível de atendimento e descrição

A categoria irá separar nossos chamados para termos um controle melhor sobre, se são sobre harware, o software, bugs etc..

Diagrama

O conteúdo gerado por IA pode estar incorreto.**Representação do Sistema em Modelo Conceitual**

Diagrama, Esquemático

O conteúdo gerado por IA pode estar incorreto.**Representação do Sistema em Modelo Lógico**

**Código em SQL**

CREATE TABLE Funcionarios (

id\_funcionario INTEGER PRIMARY KEY,

nome\_funcionario VARCHAR,

email VARCHAR,

senha VARCHAR,

setor VARCHAR,

RE NUMERIC

);

CREATE TABLE Chamados (

Id\_chamado INTEGER PRIMARY KEY,

fk\_funcionario\_idfuncionario INTEGER,

fk\_categoria\_idcategoria INTEGER,

titulo VARCHAR,

descricao VARCHAR,

data\_abertura DATE,

nivel\_atendimento VARCHAR,

data\_encerramento DATE,

fk\_AI\_Respostas\_Id\_resposta INTEGER,

fk\_Atendimentos\_tecnicos\_Id\_atendimento INTEGER

);

CREATE TABLE AI\_Respostas (

Id\_resposta INTEGER PRIMARY KEY,

fk\_chamado\_idchamado INTEGER,

resposta\_ai VARCHAR,

data\_resposta DATE,

foi\_resolvido BOOLEAN

);

CREATE TABLE Tecnicos (

Id\_tecnico INTEGER PRIMARY KEY,

nome\_tecnico VARCHAR,

email VARCHAR,

especialidade VARCHAR

);

CREATE TABLE Atendimentos\_tecnicos (

Id\_atendimento INTEGER PRIMARY KEY,

fk\_chamado\_idchamado INTEGER,

fk\_tecnico\_idtecnico INTEGER,

descricao\_tecnico VARCHAR,

data\_inicio DATE,

data\_fim DATE

);

CREATE TABLE Categoria (

Id\_categoria INTEGER PRIMARY KEY,

nome\_categoria VARCHAR

);

CREATE TABLE Abre (

fk\_Chamados\_Id\_chamado INTEGER,

fk\_Funcionarios\_id\_funcionario INTEGER

);

CREATE TABLE encaminha (

fk\_Atendimentos\_tecnicos\_Id\_atendimento INTEGER,

fk\_Tecnicos\_Id\_tecnico INTEGER

);

CREATE TABLE Define (

fk\_Categoria\_Id\_categoria INTEGER,

fk\_Chamados\_Id\_chamado INTEGER

);

ALTER TABLE Chamados ADD CONSTRAINT FK\_Chamados\_2

FOREIGN KEY (fk\_AI\_Respostas\_Id\_resposta)

REFERENCES AI\_Respostas (Id\_resposta)

ON DELETE RESTRICT;

ALTER TABLE Chamados ADD CONSTRAINT FK\_Chamados\_3

FOREIGN KEY (fk\_Atendimentos\_tecnicos\_Id\_atendimento)

REFERENCES Atendimentos\_tecnicos (Id\_atendimento)

ON DELETE RESTRICT;

ALTER TABLE Abre ADD CONSTRAINT FK\_Abre\_1

FOREIGN KEY (fk\_Chamados\_Id\_chamado)

REFERENCES Chamados (Id\_chamado)

ON DELETE RESTRICT;

ALTER TABLE Abre ADD CONSTRAINT FK\_Abre\_2

FOREIGN KEY (fk\_Funcionarios\_id\_funcionario)

REFERENCES Funcionarios (id\_funcionario)

ON DELETE RESTRICT;

ALTER TABLE encaminha ADD CONSTRAINT FK\_encaminha\_1

FOREIGN KEY (fk\_Atendimentos\_tecnicos\_Id\_atendimento)

REFERENCES Atendimentos\_tecnicos (Id\_atendimento)

ON DELETE RESTRICT;

ALTER TABLE encaminha ADD CONSTRAINT FK\_encaminha\_2

FOREIGN KEY (fk\_Tecnicos\_Id\_tecnico)

REFERENCES Tecnicos (Id\_tecnico)

ON DELETE RESTRICT;

ALTER TABLE Define ADD CONSTRAINT FK\_Define\_1

FOREIGN KEY (fk\_Categoria\_Id\_categoria)

REFERENCES Categoria (Id\_categoria)

ON DELETE RESTRICT;

ALTER TABLE Define ADD CONSTRAINT FK\_Define\_2

FOREIGN KEY (fk\_Chamados\_Id\_chamado)

REFERENCES Chamados (Id\_chamado)

ON DELETE RESTRICT;

**Referencias bibliográficas**

* Elisângela Rocha da Costa BANCOS DE DADOS RELACIONAIS 2011 https://www.fatecsp.br/dti/tcc/tcc0025.pdf?utm\_source=chatgpt.com Acesso em: 26 abr. 2025
* HEUSER, Carlos Alberto. *Projeto de Banco de Dados*. 6. ed. Porto Alegre: Bookman, 2009. Acesso em: 19 abr. 2025
* Prof. Flávio de Oliveira Silva Projeto Banco de Dados <https://www.facom.ufu.br/~flavio/sbd/files/02-sbd-projeto-banco-dados.pdf> Acesso em: 28 abr. 2025