

UNINORTE – Centro Universitário do Norte

Curso de Engenharia da Computação

CRUD Planejamento

Professor: Erico Borgonove Cruz

Manaus - AM

2025

Elionay Ramalho - 03345778

Vitor Henrique Duarte Santiago - 03346242

Eduarda Maquiné dos Santos - 03348708

CRUD Planejamento

Trabalho requisitado pelo prof. Erico Borgonove Cruz para Curso de
Engenharia da Computação na disciplina Back-End Frameworks

Manaus - AM

2025

1. Introdução e Justificativa

1.1 Contexto do Problema

A TechManage Solutions enfrenta desafios significativos no controle de informações internas devido ao processo manual atual de registro de dados. Esta abordagem tradicional resulta em:

Inconsistências nos dados: Registros duplicados e informações desatualizadas

Perda de informações: Dados críticos de projetos e tarefas não são adequadamente armazenados

Baixa produtividade: Tempo excessivo gasto em processos manuais de registro e busca de informações

Falta de padronização: Diferentes formatos e estruturas de registro entre equipes

1.2 Justificativa do Estudo de Caso

O sistema proposto visa automatizar o gerenciamento de projetos, usuários e tarefas, criando um ambiente centralizado e padronizado para o controle de informações corporativas. A escolha deste modelo baseia-se na necessidade urgente de

Digitalizar processos

Melhorar a rastreabilidade de projetos

Otimizar a alocação de recursos

Facilitar o acompanhamento do progresso das atividades

2. Objetivos do Sistema

2.1 Objetivo Geral

Desenvolver um sistema web CRUD completo para automatizar o controle de informações de projetos, usuários e tarefas na TechManage Solutions.

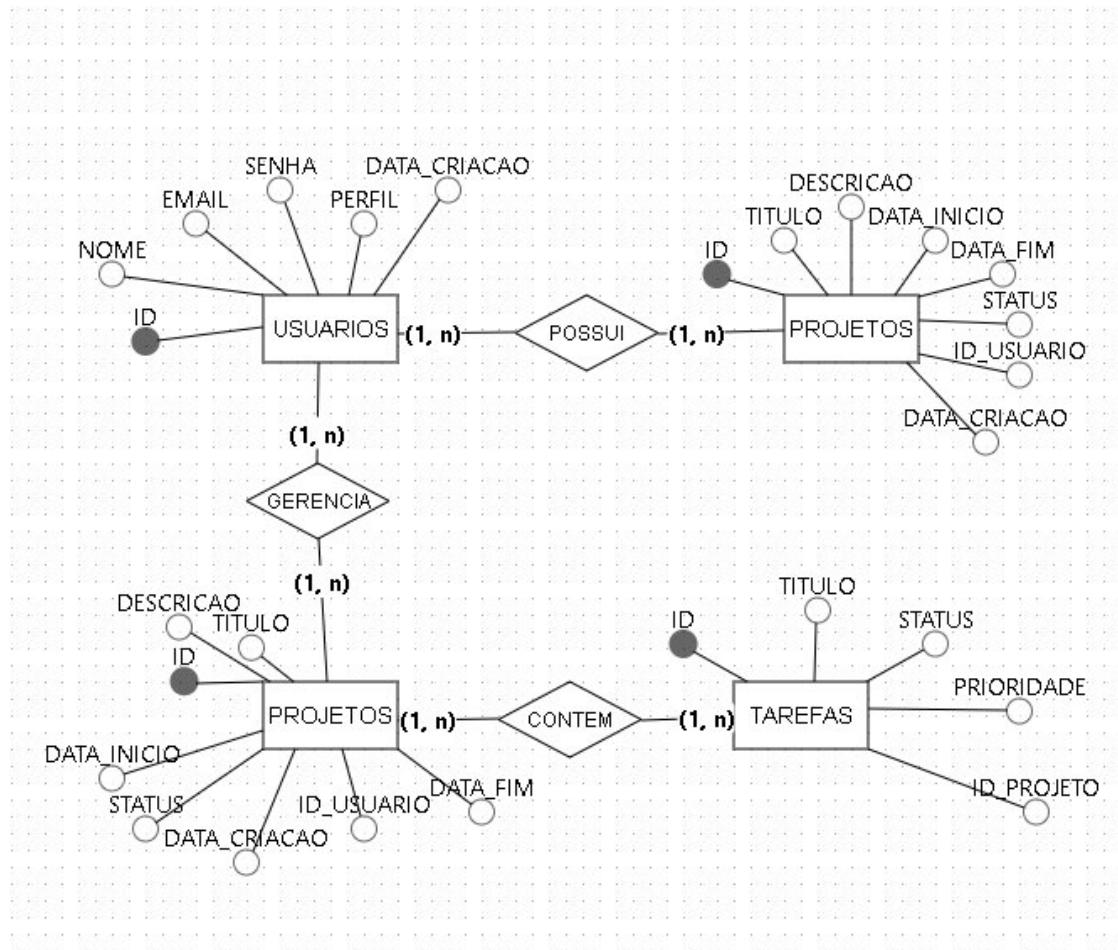
2.2 Objetivos Específicos

Implementar cadastro e gerenciamento de usuários com diferentes perfis

Criar sistema de controle de projetos com datas e responsáveis
 Desenvolver módulo de gestão de tarefas com status e prioridades
 Estabelecer relacionamentos adequados entre as entidades
 Garantir consistência e integridade dos dados
 Fornecer interface web intuitiva para operações CRU

3. Modelagem de Dados

3.1 Modelo Entidade-Relacionamento (MER)



3.4 Relacionamentos

3.4.1 Usuários → Projetos (1:N)

Um usuário pode ser responsável por múltiplos projetos

Cada projeto possui exatamente um usuário responsável

Cardinalidade: 1 para N

3.4.2 Projetos → Tarefas (1:N)

Um projeto pode conter múltiplas tarefas

Cada tarefa pertence a exatamente um projeto

Cardinalidade: 1 para N

4. Descrição Técnica das Tecnologias Utilizadas

4.1 Backend

4.1.1 Node.js

Versão: LTS 18.x ou superior

Justificativa: Ambiente de execução JavaScript assíncrono e orientado a eventos

Vantagens: Alta performance, grande ecossistema de pacotes, suporte a ES6+

4.1.2 Express.js

Versão: 4.18.x

Justificativa: Framework web minimalista para Node.js

Funcionalidades: Roteamento, middlewares, tratamento de requisições HTTP

4.1.3 Sequelize ORM

Versão: 6.x

Justificativa: ORM (Object-Relational Mapping) para Node.js

Vantagens:

Supporte a múltiplos bancos de dados

Migrations e seeders

Validações de dados

Associações entre modelos

4.2 Banco de Dados

4.2.1 MySQL

Versão: 8.0 ou superior

Justificativa: SGBD relacional robusto e amplamente utilizado

Características: Transações ACID, integridade referencial, performance

4.3 Configuração e Ambiente

4.3.1 Variáveis de Ambiente (.env)

```
PORT = 3000

DB_HOST=localhost
DB_PORT=3306
DB_DATABASE=mydatabase
DB_USER=root
DB_PASS=Senha123@
```

4.3.2 Scripts do Package.json

```
{
  "name": "crud_tech",
  "version": "1.0.0",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "dev": " nodemon src/server.js"
  },
  "author": "",
  "license": "ISC",
  "keywords": [],
  "description": "",
  "dependencies": {
    "bcrypt": "^6.0.0",
    "cors": "^2.8.5",
    "dotenv": "^17.2.3",
    "express": "^5.1.0",
    "express-validator": "^7.3.0",
    "jsonwebtoken": "^9.0.2",
    "mysql2": "^3.15.3",
    "path": "^0.12.7",
    "sequelize": "^6.37.7"
  },
  "devDependencies": {
    "nodemon": "^3.1.10"
  }
}
```

5. Funcionalidades Implementadas

5.1 Operações CRUD por Entidade

Usuários

POST /users - Criar novo usuário

GET /users - Listar todos os usuários

GET /users/:id - Buscar usuário por ID

PUT /users/:id - Atualizar usuário

DELETE /users/:id - Excluir usuário (físico)

Projetos

POST /projects - Criar novo projeto

GET /projects - Listar todos os projetos

GET /projects/:id - Buscar projeto por ID

PUT /projects/:id - Atualizar projeto

DELETE /projects/:id - Excluir projeto (lógico)

Tarefas

POST /tasks - Criar nova tarefa

GET /tasks - Listar todas as tarefas

GET /tasks/:id - Buscar tarefa por ID

PUT /tasks/:id - Atualizar tarefa

DELETE /tasks/:id - Excluir tarefa (lógico)

5.2 Validações e Regras de Negócio

Validação de e-mail único para usuários

Verificação de datas consistentes em projetos

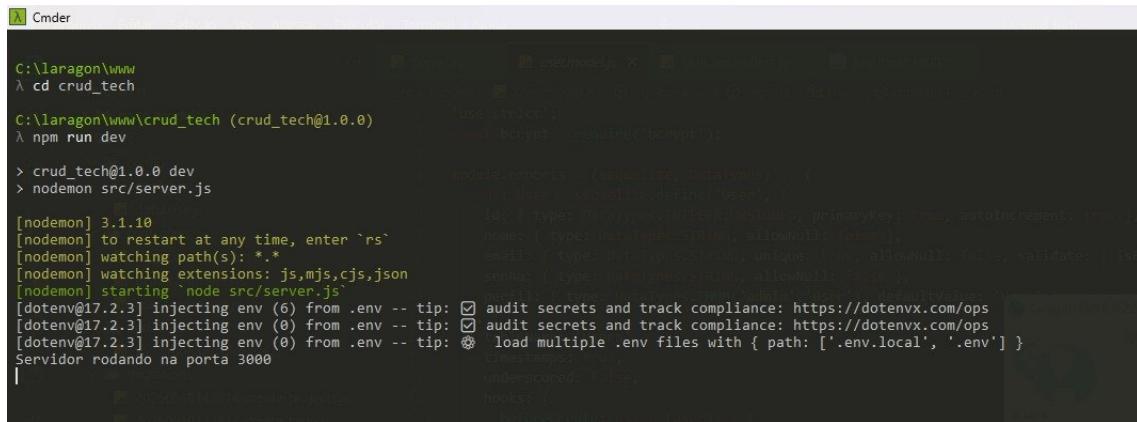
Controle de integridade referencial nos relacionamentos

Validação de enums para status e prioridades

Mensagens de erro padronizadas em JSON

6. Imagens de funcionamento

6.1 Servidor rodando

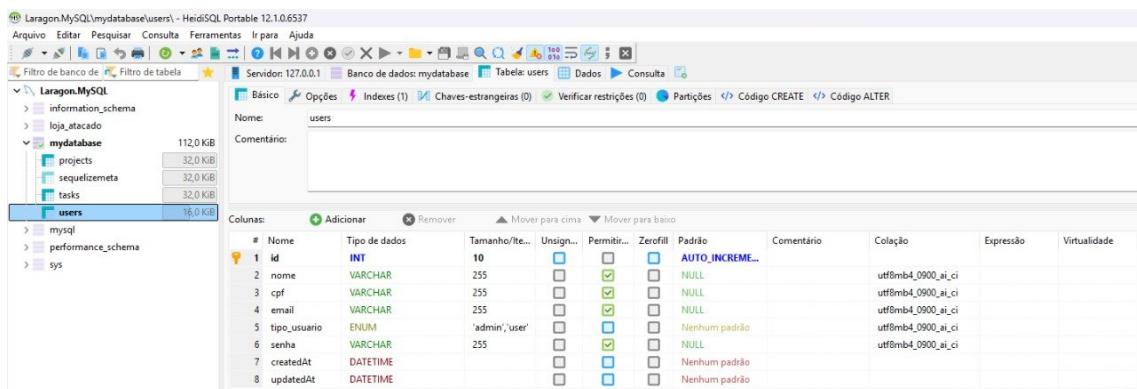


```
C:\laragon\www
λ cd crud_tech
C:\laragon\www\crud_tech (crud_tech@1.0.0)
λ npm run dev

> crud_tech@1.0.0 dev
> nodemon src/server.js

[nodemon] 3.1.10
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): ***!
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node src/server.js`
[dotenv@17.2.3] injecting env (6) from .env -- tip: ⚡ audit secrets and track compliance: https://dotenvx.com/ops
[dotenv@17.2.3] injecting env (8) from .env -- tip: ⚡ audit secrets and track compliance: https://dotenvx.com/ops
[dotenv@17.2.3] injecting env (0) from .env -- tip: ⚡ load multiple .env files with { path: ['.env.local', '.env'] }
Servidor rodando na porta 3000
|
```

6.2 Banco de dados funcionando



Laragon.MySQL\mydatabase\users - HeidiSQL Portable 12.1.0.6537

Arquivo Editar Pesquisar Consulta Ferramentas Ir para Ajuda

Filtro de banco de Filtro de tabela Servidor: 127.0.0.1 Banco de dados: mydatabase Tabela: users Dados Consulta

Básico Opcões Índices () Chaves-estrangeiras () Verificar restrições () Partições () Código CREATE () Código ALTER ()

Nome: users

Comentário:

Colunas: Adicionar Remover Mover para cima Mover para baixo

#	Nome	Tipo de dados	Tamanho/Itens	Unsigned	Permitir...	Zerofill	Padrão	Comentário	Colação	Expressão	Virtualidade
1	id	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT	utf8mb4_0900_ai_ci		
2	nome	VARCHAR	255	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL	utf8mb4_0900_ai_ci		
3	cpf	VARCHAR	255	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL	utf8mb4_0900_ai_ci		
4	email	VARCHAR	255	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL	utf8mb4_0900_ai_ci		
5	tipo_usuario	ENUM	'admin','user'	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Nenhum padrão	utf8mb4_0900_ai_ci		
6	senha	VARCHAR	255	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL	utf8mb4_0900_ai_ci		
7	createdAt	DATETIME		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Nenhum padrão			
8	updatedAt	DATETIME		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Nenhum padrão			

7. Considerações Finais

O sistema proposto atende integralmente aos requisitos da TechManage Solutions, proporcionando uma solução robusta e escalável para o gerenciamento de informações corporativas. A arquitetura definida permite futuras expansões e a tecnologia selecionada garante performance e confiabilidade.

A implementação seguirá as melhores práticas de desenvolvimento web moderno, com ênfase na qualidade do código, segurança dos dados e experiência do usuário final.