



Sistema de Gerenciamento Acadêmico da UniverSem

Informatização e Otimização de Processos

Apresentado por: Vinícius Tomaz Bonilha e Guilherme Santos Ismerio

Setembro de 2025

Visão Geral do Sistema

A **Universidade UniverSem** busca informatizar seu sistema acadêmico para otimizar as rotinas diárias e eliminar a demora nos processos de emissão de documentos, como certificados e declarações, que atualmente são feitos de forma física.

Objetivo principal: Criar um banco de dados relacional robusto e eficiente para gerenciar informações de alunos, cursos, professores, disciplinas, turmas, aulas e pagamentos de forma centralizada.

Componentes Principais:



Gestão de Alunos e Matrículas



Gestão de Cursos e Disciplinas



Gestão de Professores e Aulas



Gestão Financeira (Pagamentos e Notas Fiscais)

Justificativa: Soluções para Desafios Universitários

Universidades enfrentam desafios diários na gestão acadêmica e administrativa, resultando em ineficiências e impacto na experiência de alunos e professores. O sistema da **UniverSem** oferece soluções eficazes para esses problemas.

⚠ Problemas Comuns em Universidades

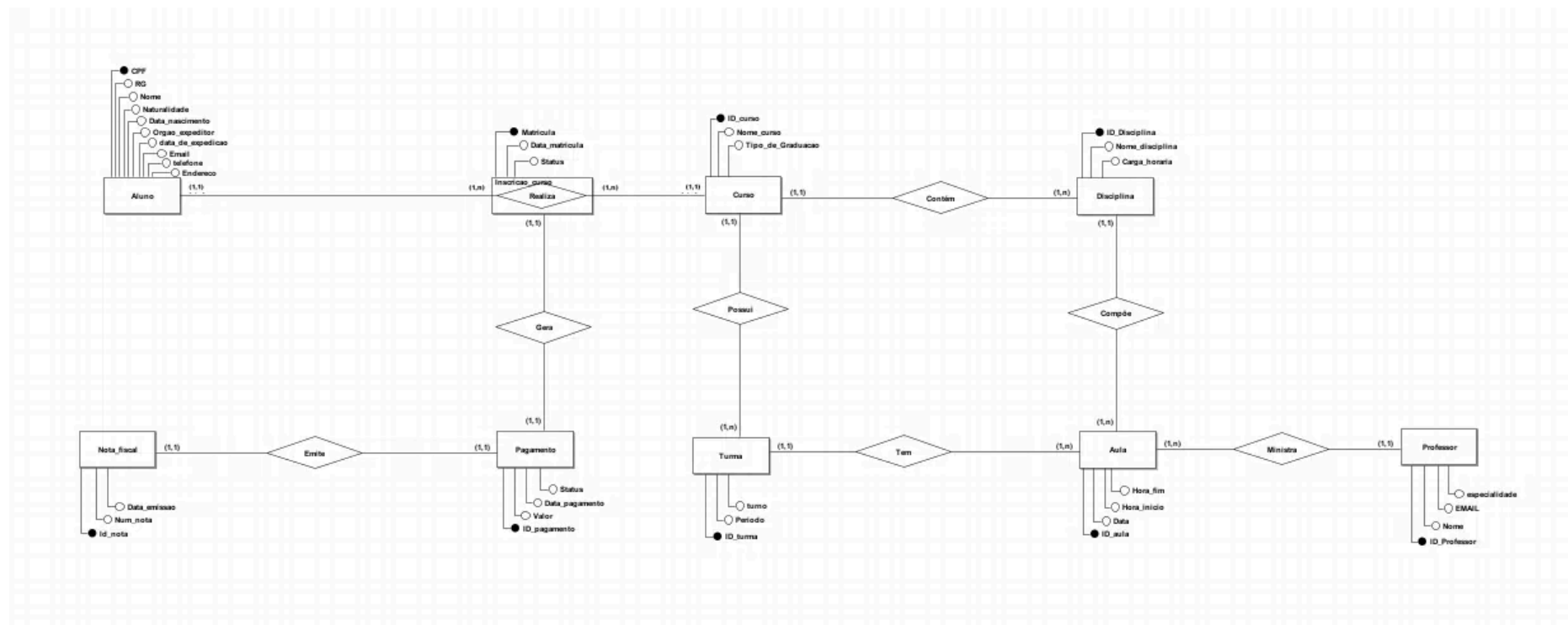
- ✗ Demora na emissão de documentos e certificados
- ✗ Dificuldade na gestão de matrículas e turmas
- ✗ Processos manuais propensos a erros e inconsistências
- ✗ Falta de integração entre departamentos e dados dispersos

💡 Benefícios do Sistema UniverSem

- ✔ **Agilidade** na emissão de documentos, com acesso instantâneo a informações
- ✔ **Otimização** da gestão de matrículas, cursos e professores
- ✔ **Redução de erros** e aumento da precisão dos dados com automação
- ✔ **Integração total** entre todos os setores acadêmicos e financeiros
- ✔ **Melhora na experiência** de alunos e funcionários com processos simplificados
- ✔ **Tomada de decisões** estratégicas baseada em dados precisos e atualizados

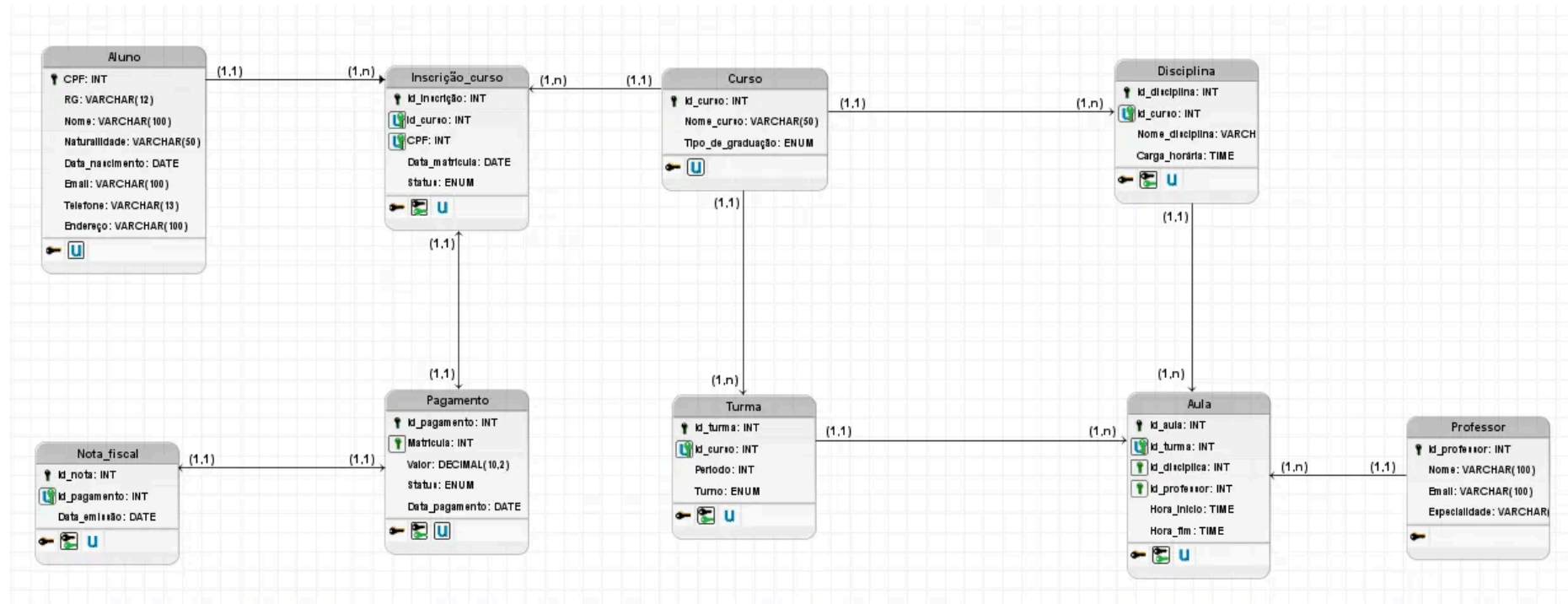
Modelo Conceitual (DER)

O **Diagrama de Entidade-Relacionamento (DER)** representa a estrutura conceitual do banco de dados, mostrando as entidades, seus atributos e os relacionamentos entre elas.



Modelo Lógico

O **Modelo Lógico** representa a estrutura do banco de dados mais próxima da implementação, detalhando tabelas, campos, tipos de dados, chaves primárias e estrangeiras.



Entidades do Sistema Universitário

Entidades Principais



Aluno

- 🔑 CPF (PK)
- 📄 RG, Nome
- 📍 Naturalidade
- 📅 Data_nascimento
- ✉ Email, Telefone



Curso

- 🔑 Id_curso (PK)
- 📄 Nome_curso
- ⚙ Tipo_graduacao



Professor

- 🔑 Id_professor (PK)
- 👤 Nome, Email
- 🧠 Especialidade

Entidades Acadêmicas



Disciplina

- 🔑 Id_disciplina (PK)
- 🔗 Id_curso (FK)
- 📄 Nome_disciplina
- 🕒 Carga_horaria



Turma

- 🔑 Id_turma (PK)
- 🔗 Id_curso (FK)
- ⬆️ Período
- ⚙ Turno



Aula

- 🔑 Id_aula (PK)
- 🔗 Id_turma (FK)
- 🔗 Id_disciplina (FK)
- 🔗 Id_professor (FK)
- 🕒 Data_Hora_inicio/fim

Entidades Financeiras



Inscricao_curso

- 🔑 Id_inscricao (PK)
- 🔗 Id_curso (FK)
- 🔗 CPF (FK)
- 📅 Data_matricula
- 📍 Status



Pagamento

- 🔑 Id_pagamento (PK)
- 🔗 Id_inscricao (FK)
- 💵 Valor
- 📅 Data_pagamento
- 📍 Status



Nota_fiscal

- 🔑 Id_nota (PK)
- 🔗 Id_pagamento (FK)
- 📅 Data_emissao

Relacionamentos e Cardinalidades

Os principais relacionamentos entre as entidades do sistema universitário:

Aluno - Inscricao_curso - Curso

Aluno (1,1) --- (1,N) Inscricao_curso --- (1,1) Curso

Inscricao_curso - Pagamento - Nota_fiscal

Inscricao_curso (1,1) <---> Pagamento <---> (1,1) Nota_fiscal

Curso - Disciplina

Curso (1,1) ---> (1,N) Disciplina

Curso - Turma - Aula

Curso (1,1) ---> (1,N) Turma (1,1) ---> (1,N) Aula

Disciplina - Aula - Professor

Disciplina (1,1) ---> (1,N) Aula <--- (1,N) Professor (1,1)



Implementação da Criação das Tabelas

Criação do Banco e Entidades Principais

```
CREATE DATABASE IF NOT EXISTS Sistema_universidade;  
USE Sistema_universidade;  
  
CREATE TABLE Aluno (  
    CPF VARCHAR(11) PRIMARY KEY,  
    Nome VARCHAR(100) NOT NULL,  
    Data_nascimento DATE,  
    Email VARCHAR(100)  
);
```

Entidades Acadêmicas

```
CREATE TABLE Disciplina (  
    Id_disciplina INT PRIMARY KEY AUTO_INCREMENT,  
    Id_curso INT,  
    Nome_disciplina VARCHAR(50) NOT NULL,  
    Carga_horaria INT,  
    FOREIGN KEY (Id_curso) REFERENCES Curso(Id_curso)  
);
```

Entidades Financeiras

```
CREATE TABLE Inscricao_curso (  
    Id_inscricao INT PRIMARY KEY AUTO_INCREMENT,  
    Id_curso INT,  
    CPF VARCHAR(11),  
    FOREIGN KEY (Id_curso) REFERENCES Curso(Id_curso),  
    FOREIGN KEY (CPF) REFERENCES Aluno(CPF)  
);
```



O código SQL acima implementa as tabelas do sistema universitário conforme o modelo lógico. Cada tabela inclui suas chaves primárias e estrangeiras para garantir a integridade referencial do banco de dados.

Implementação dos Valores nas Entidades



Inserção em Entidades Principais

```
-- Inserção de dados na tabela Aluno
INSERT INTO Aluno (CPF, RG, Nome, Naturalidade, Data_nascimento)
VALUES
('11122233344', '112223334', 'Ana Carvalho', 'Belo Horizonte', '1998-03-12'),
('22233344455', '223334445', 'Bruno Oliveira', 'Porto Alegre', '2000-07-25'),
('33344455566', '334445556', 'Carla Mendes', 'Salvador', '1999-11-05');

-- Inserção de dados na tabela Curso
INSERT INTO Curso (Nome_curso, Tipo_graduacao) VALUES
('Direito', 'Bacharelado'),
('Pedagogia', 'Licenciatura'),
('Ciência da Computação', 'Bacharelado');
```



Inserção em Entidades Acadêmicas

```
-- Inserção de dados na tabela Disciplina
INSERT INTO Disciplina (Id_curso, Nome_disciplina, Carga_horaria)
VALUES
(1, 'Direito Civil', 60),
(2, 'Didática', 80),
(3, 'Infraestrutura de Redes', 100);

-- Inserção de dados na tabela Turma
INSERT INTO Turma (Id_curso, Periodo, Turno) VALUES
(1, 1, 'Manha'),
(2, 1, 'Tarde'),
(3, 1, 'Noite');
```



Inserção em Entidades Financeiras

```
-- Inserção de dados na tabela Inscricao_curso
INSERT INTO Inscricao_curso (Id_curso, CPF, Data_matricula, Status)
VALUES
(1, '11122233344', '2023-01-15', 'Ativo'),
(2, '22233344455', '2023-02-20', 'Ativo'),
(3, '33344455566', '2023-03-10', 'Ativo');

-- Inserção de dados na tabela Pagamento
INSERT INTO Pagamento (Id_inscricao, Valor, Data_pagamento, Status)
VALUES
(1, 1500.00, '2023-01-20', 'Pago'),
(2, 1200.00, '2023-02-25', 'Pago'),
(3, 900.00, '2023-03-15', 'Pendente');
```



Técnicas Avançadas de Inserção

```
-- Inserção com SELECT
INSERT INTO Nota_fiscal (Id_pagamento, Data_emissao)
SELECT
    Id_pagamento,
    DATE_ADD(Data_pagamento, INTERVAL 1 DAY) as Data_emissao
FROM Pagamento
WHERE Status = 'Pago'
AND Id_pagamento NOT IN (SELECT Id_pagamento FROM Nota_fiscal);

-- Inserção com dados aleatórios
INSERT INTO Inscricao_curso (Id_curso, CPF, Data_matricula, Status)
SELECT
    FLOOR(1 + RAND() * 3) as Id curso,
```

Conclusão

O **Sistema de Gerenciamento Universitário** foi projetado para atender às necessidades de instituições de ensino superior, integrando processos acadêmicos e administrativos.

Principais Benefícios

- ✓ Gestão centralizada de alunos, cursos, professores e disciplinas
- ✓ Controle eficiente de matrículas e pagamentos
- ✓ Organização otimizada de turmas e aulas
- ✓ Estrutura de banco de dados robusta e escalável

*A modelagem adequada do banco de dados é **fundamental** para garantir a integridade, consistência e eficiência do sistema, permitindo o crescimento da instituição e a adaptação a novos requisitos.*



Créditos



Vinícius Tomaz Bonilha



Guilherme Santos Ismerio

Obrigado pela atenção!