**Título do Projeto:**

**Criando uma Base de Dados para que Atenda um Sistema de uma Faculdade**

**Projeto:**

Os proprietários de uma faculdade precisam de um sistema que viabilize o armazenamento de informações sobre seus **alunos, cursos, matérias e professores** para que seja possível realizar controles básicos como **montar turmas e armazenar notas** dos alunos.

Com base no que foi apresentado, o objetivo é criar um banco de dados que ofereça suporte para que o sistema possa atender às necessidades do cliente. Abaixo estão as respostas detalhadas para cada questão proposta, além da modelagem sugerida.

**3. Objetivos**

**3.1 Objetivos de Aprendizagem (Plano de Aprendizagem):**

* **Análise de Requisitos**: Identificar as necessidades do cliente e definir quais dados serão necessários.
* **Modelagem Conceitual**: Estruturar as entidades e relacionamentos de forma clara, compreendendo como as informações se interconectam.
* **Modelagem Lógica**: Definir as tabelas, atributos e relacionamentos que irão compor o banco de dados no sistema relacional.

**1. Necessidades dos Clientes**

**a. Quais informações precisam ser armazenadas?**

* Dados dos alunos (nome, CPF, curso, contatos).
* Informações dos professores (nome, formação, contatos).
* Detalhes sobre cursos oferecidos e suas disciplinas.
* Cadastro das turmas e seus respectivos professores e disciplinas.
* Matrículas de alunos em turmas específicas.
* Notas obtidas pelos alunos ao longo das disciplinas.

**b. Quais os dados precisam ser guardados?**

* Alunos: Nome, data de nascimento, CPF, e-mail, telefone e curso vinculado.
* Professores: Nome, formação, e-mail e telefone.
* Cursos: Nome e duração.
* Disciplinas: Nome, carga horária e curso ao qual pertence.
* Turmas: Disciplina, professor, semestre/ano.
* Matrículas: Aluno, turma e data de matrícula.
* Notas: Avaliações e notas obtidas pelos alunos.

**c. O que será feito com os dados posteriormente?**

* **Controle acadêmico**: Armazenar notas e aprovações.
* **Gestão de professores**: Acompanhar a alocação de professores em turmas.
* **Relatórios**: Gerar históricos escolares e calcular médias.
* **Planejamento**: Auxiliar na alocação de alunos e professores nas turmas futuras.

### Análise de Requisitos

#### 1. **Gerenciamento de Alunos**

* **Necessidades do Cliente**:
  + Registrar informações pessoais e acadêmicas dos alunos.
  + Associar cada aluno a um curso específico.
* **Dados Necessários**:
  + Identificador único do aluno (id\_aluno).
  + Nome completo (nome).
  + Data de nascimento (data\_nascimento).
  + CPF (cpf) - Deve ser único no sistema.
  + E-mail e telefone para contato.
  + Identificador do curso ao qual o aluno pertence (id\_curso), que é uma chave estrangeira para a tabela de Cursos.

#### 2. **Gerenciamento de Professores**

* **Necessidades do Cliente**:
  + Registrar informações sobre os professores que ministram as disciplinas.
  + Armazenar detalhes da formação acadêmica para verificação e requisitos de qualificação.
* **Dados Necessários**:
  + Identificador único do professor (id\_professor).
  + Nome, e-mail e telefone de contato.
  + Formação acadêmica do professor (formacao\_academica).

#### 3. **Gerenciamento de Cursos**

* **Necessidades do Cliente**:
  + Manter um registro dos cursos oferecidos, com informações como nome e duração.
  + Associar cada curso com suas disciplinas específicas.
* **Dados Necessários**:
  + Identificador único do curso (id\_curso).
  + Nome do curso (nome\_curso).
  + Duração do curso em semestres (duracao).

#### 4. **Gerenciamento de Disciplinas**

* **Necessidades do Cliente**:
  + Listar as disciplinas oferecidas em cada curso.
  + Registrar a carga horária de cada disciplina.
* **Dados Necessários**:
  + Identificador único da disciplina (id\_disciplina).
  + Nome da disciplina (nome\_disciplina).
  + Identificador do curso ao qual a disciplina pertence (id\_curso), criando uma relação com a tabela de Cursos.
  + Carga horária da disciplina (carga\_horaria).

#### 5. **Gerenciamento de Turmas**

* **Necessidades do Cliente**:
  + Criar turmas para cada disciplina, associadas a um professor específico e a um período (semestre/ano).
  + Facilitar a vinculação de alunos a essas turmas através das matrículas.
* **Dados Necessários**:
  + Identificador único da turma (id\_turma).
  + Identificador da disciplina (id\_disciplina) e do professor (id\_professor) associados à turma.
  + Ano e semestre da turma (ano\_semestre), no formato "YYYY/S".

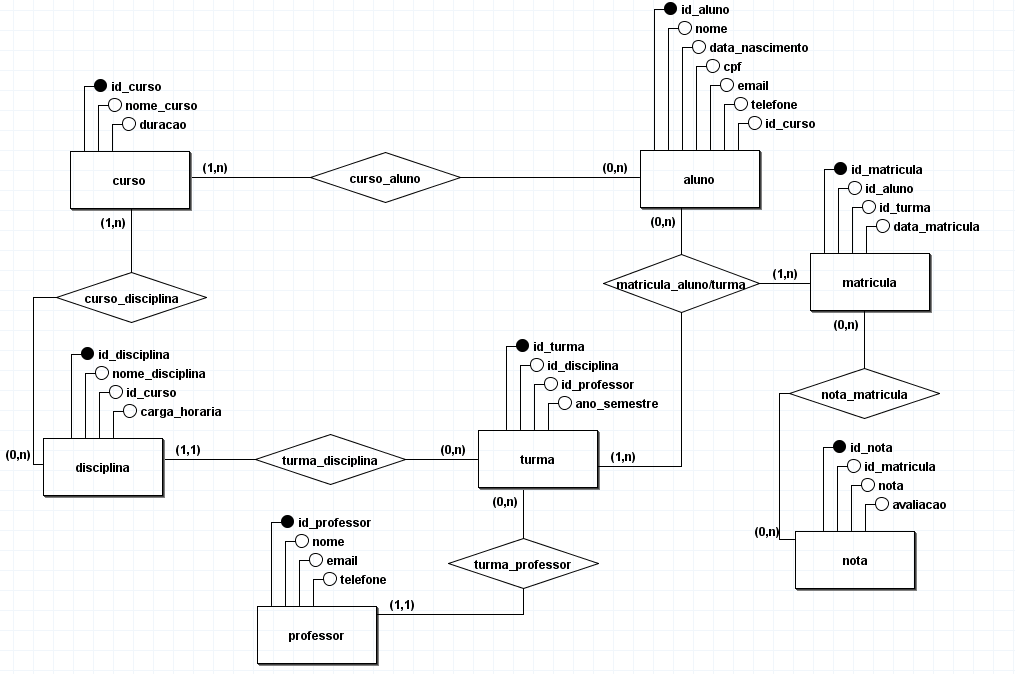
#### 6. **Gerenciamento de Matrículas**

* **Necessidades do Cliente**:
  + Registrar a inscrição dos alunos em turmas específicas.
  + Armazenar a data de matrícula para histórico e verificação de tempo de curso.
* **Dados Necessários**:
  + Identificador único da matrícula (id\_matricula).
  + Identificador do aluno (id\_aluno) e da turma (id\_turma), associando o aluno à turma específica.
  + Data da matrícula (data\_matricula).

#### 7. **Gerenciamento de Notas**

* **Necessidades do Cliente**:
  + Registrar e armazenar as notas obtidas pelos alunos em avaliações específicas.
  + Associar cada nota a uma matrícula para vinculação entre aluno e turma.
* **Dados Necessários**:
  + Identificador único da nota (id\_nota).
  + Identificador da matrícula (id\_matricula) para associar a nota a um aluno e turma específicos.
  + Nota obtida (nota), com duas casas decimais para precisão.
  + Tipo de avaliação (avaliacao), como "Prova 1" ou "Trabalho Final".

**2.Modelo Conceitual**

****

**3. Tabelas e Atributos**

**Tabela: Alunos**

* **id\_aluno** (INT) - Chave Primária
* **nome** (VARCHAR(100))
* **data\_nascimento** (DATE)
* **cpf** (VARCHAR(11)) - Único
* **email** (VARCHAR(100))
* **telefone** (VARCHAR(15))
* **id\_curso** (INT) - Chave Estrangeira (Cursos)

**Tabela: Professores**

* **id\_professor** (INT) - Chave Primária
* **nome** (VARCHAR(100))
* **email** (VARCHAR(100))
* **telefone** (VARCHAR(15))

**Tabela: Cursos**

* **id\_curso** (INT) - Chave Primária
* **nome\_curso** (VARCHAR(100))
* **duracao** (INT) - Duração em semestres

**Tabela: Disciplinas**

* **id\_disciplina** (INT) - Chave Primária
* **nome\_disciplina** (VARCHAR(100))
* **id\_curso** (INT) - Chave Estrangeira
* **carga\_horaria** (INT)

**Tabela: Turmas**

* **id\_turma** (INT) - Chave Primária
* **id\_disciplina** (INT) - Chave Estrangeira
* **id\_professor** (INT) - Chave Estrangeira
* **ano\_semestre** (VARCHAR(10)) - Ex: "2024/1"

**Tabela: Matrículas**

* **id\_matricula** (INT) - Chave Primária
* **id\_aluno** (INT) - Chave Estrangeira
* **id\_turma** (INT) - Chave Estrangeira
* **data\_matricula** (DATE)

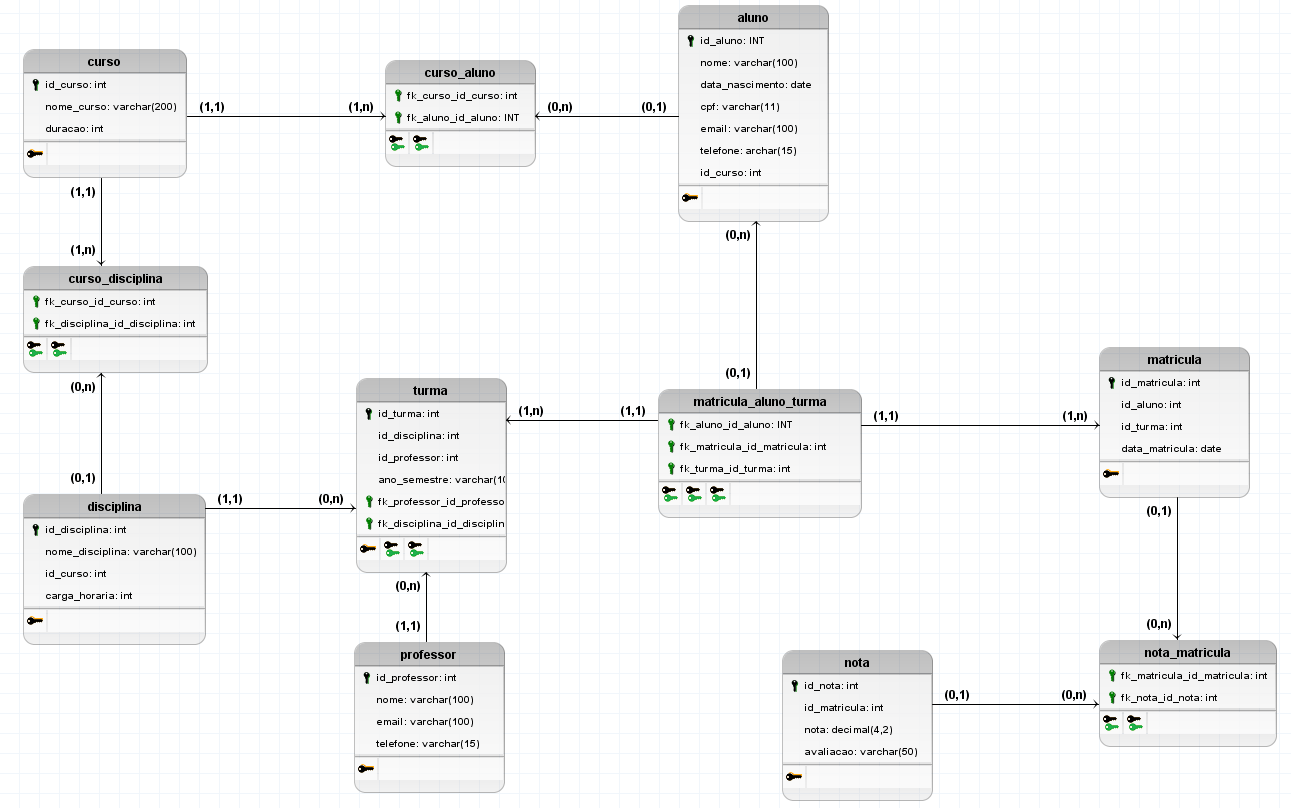
**Tabela: Notas**

* **id\_nota** (INT) - Chave Primária
* **id\_matricula** (INT) - Chave Estrangeira
* **nota** (DECIMAL(4,2))
* **avaliacao** (VARCHAR(50)) - Ex: "Prova 1", "Trabalho Final"

**4. Tipos de Dados Utilizados**

* **INT**: Para identificadores e duração.
* **VARCHAR**: Para nomes, e-mails, CPF e telefones.
* **DATE**: Para datas como nascimento e matrícula.
* **DECIMAL(4,2)**: Para notas, com duas casas decimais de precisão.

**5.Modelo Lógico**

****

**6. Relacionamentos Entre as Tabelas**

1. **Cursos e Alunos**

**Relacionamento**: Um curso pode ter muitos alunos, e cada aluno pode estar matriculado em mais de um curso.

**Cardinalidade**:

Curso para Aluno: **1**(um curso pode ter de zero até muitos alunos).

Mínima: 0 (um curso pode não ter nenhum aluno inscrito).

Máxima: N (um curso pode ter muitos alunos).

Aluno para Curso: **N**(um aluno pode estar matriculado em vários cursos).

Mínima: 0 (um aluno pode não estar matriculado em nenhum curso).

Máxima: N (um aluno pode estar matriculado em vários cursos).

1. **Cursos e Disciplinas**

**Relacionamento**: Um curso pode ter várias disciplinas, e cada disciplina pode estar associada a mais de um curso.

**Cardinalidade**:

Curso para Disciplina: **1**(um curso pode ter várias disciplinas).

Mínima: 1 (um curso precisa ter ao menos uma disciplina).

Máxima: N (um curso pode ter várias disciplinas).

Disciplina para Curso: **N**(uma disciplina pode estar em mais de um curso).

Mínima: 0 (uma disciplina pode não estar associada a nenhum curso).

Máxima: N (uma disciplina pode estar associada a vários cursos).

1. **Turmas, Professores e Disciplinas**

**Relacionamento**:

Uma turma está vinculada a um professor e uma disciplina (1:1).

Professores podem ministrar várias turmas.

**Cardinalidade**:

Turma para Professor e Disciplina: **1:1** (cada turma tem um professor e uma disciplina específicos).

Mínima e máxima: 1 (cada turma precisa de um professor e uma disciplina).

Professor para Turma: **1**(um professor pode lecionar em várias turmas).

Mínima: 0 (um professor pode não estar ministrando nenhuma turma no momento).

Máxima: N (um professor pode ministrar várias turmas).

Disciplina para Turma: **1**(uma disciplina pode ser lecionada em várias turmas).

Mínima: 0 (uma disciplina pode não estar sendo lecionada em nenhuma turma).

Máxima: N (uma disciplina pode ser lecionada em várias turmas).

1. **Matrículas e Alunos/Turmas**

**Relacionamento**:

Alunos podem se matricular em várias turmas, e cada turma pode ter vários alunos.

**Cardinalidade**:

Aluno para Turma: **N**(um aluno pode estar matriculado em várias turmas e uma turma pode ter vários alunos).

Mínima: 0 (um aluno pode não estar matriculado em nenhuma turma).

Máxima: N (um aluno pode estar matriculado em várias turmas).

Turma para Aluno: **M**(uma turma pode ter vários alunos matriculados).

Mínima: 0 (uma turma pode não ter alunos matriculados).

Máxima: N (uma turma pode ter vários alunos matriculados).

1. **Notas e Matrículas**

**Relacionamento**:

Cada nota está vinculada a uma matrícula específica.

**Cardinalidade**:

Nota para Matrícula: **N:1** (uma nota corresponde a uma matrícula).

Mínima: 1 (cada nota precisa estar associada a uma matrícula).

Máxima: 1 (cada nota está associada a uma única matrícula).

Matrícula para Nota: **1**(uma matrícula pode ter várias notas associadas).

Mínima: 0 (uma matrícula pode não ter nenhuma nota associada).

Máxima: N (uma matrícula pode ter várias notas associadas).

**7. Tabelas e Atributos**

**Tabela: Alunos**

* **id\_aluno** (INT) - Chave Primária
* **nome** (VARCHAR(100))
* **data\_nascimento** (DATE)
* **cpf** (VARCHAR(11)) - Único
* **email** (VARCHAR(100))
* **telefone** (VARCHAR(15))
* **id\_curso** (INT) - Chave Estrangeira (Cursos)

**Tabela: Professores**

* **id\_professor** (INT) - Chave Primária
* **nome** (VARCHAR(100))
* **email** (VARCHAR(100))
* **telefone** (VARCHAR(15))

**Tabela: Cursos**

* **id\_curso** (INT) - Chave Primária
* **nome\_curso** (VARCHAR(100))
* **duracao** (INT) - Duração em semestres

**Tabela: Disciplinas**

* **id\_disciplina** (INT) - Chave Primária
* **nome\_disciplina** (VARCHAR(100))
* **id\_curso** (INT) - Chave Estrangeira
* **carga\_horaria** (INT)

**Tabela: Turmas**

* **id\_turma** (INT) - Chave Primária
* **id\_disciplina** (INT) - Chave Estrangeira
* **id\_professor** (INT) - Chave Estrangeira
* **ano\_semestre** (VARCHAR(10)) - Ex: "2024/1"

**Tabela: Matrículas**

* **id\_matricula** (INT) - Chave Primária
* **id\_aluno** (INT) - Chave Estrangeira
* **id\_turma** (INT) - Chave Estrangeira
* **data\_matricula** (DATE)

**Tabela: Notas**

* **id\_nota** (INT) - Chave Primária
* **id\_matricula** (INT) - Chave Estrangeira
* **nota** (DECIMAL(4,2))
* **avaliacao** (VARCHAR(50)) - Ex: "Prova 1", "Trabalho Final"

**8. Tipos de Dados Utilizados**

* **INT**: Para identificadores e duração.
* **VARCHAR**: Para nomes, e-mails, CPF e telefones.
* **DATE**: Para datas como nascimento e matrícula.
* **DECIMAL(4,2)**: Para notas, com duas casas decimais de precisão.

**9. Modelo Físico**

CREATE DATABASE **Faculdade**;

USE **Faculdade**;

CREATE TABLE **Alunos** (

id\_aluno INT PRIMARY KEY AUTO\_INCREMENT,

nome VARCHAR(100),

data\_nascimento DATE,

cpf VARCHAR(11) UNIQUE,

email VARCHAR(100),

telefone VARCHAR(15),

id\_curso INT,

FOREIGN KEY (id\_curso) REFERENCES Cursos(id\_curso)

);

-- **Inserções na tabela Alunos**

INSERT INTO Alunos VALUES (1, 'Ana Silva', '2000-01-15', '12345678901', 'ana@gmail.com', '11999999991', 1);

INSERT INTO Alunos VALUES (2, 'Bruno Santos', '1999-03-22', '12345678902', 'bruno@gmail.com', '11999999992', 1);

CREATE TABLE **Professores** (

id\_professor INT PRIMARY KEY AUTO\_INCREMENT,

nome VARCHAR(100),

email VARCHAR(100),

telefone VARCHAR(15)

);

-- **Inserções na tabela Professores**

INSERT INTO Professores VALUES (1, 'Carlos Oliveira', 'carlos@gmail.com', '11999999993', 'Mestre em Database');

CREATE TABLE **Cursos** (

id\_curso INT PRIMARY KEY AUTO\_INCREMENT,

nome\_curso VARCHAR(100),

duracao INT

);

-- **Inserções na tabela Cursos**

INSERT INTO Cursos VALUES (1, ‘Gestão da Tecnologia da Informação’, 8);

CREATE TABLE **Disciplinas** (

id\_disciplina INT PRIMARY KEY AUTO\_INCREMENT,

nome\_disciplina VARCHAR(100),

id\_curso INT,

carga\_horaria INT,

FOREIGN KEY (id\_curso) REFERENCES Cursos(id\_curso)

);

-- **Inserções na tabela Disciplinas**

INSERT INTO Disciplinas VALUES (1, ‘Database Modeling & SQL’, 1, 60);

CREATE TABLE **Turmas** (

id\_turma INT PRIMARY KEY AUTO\_INCREMENT,

id\_disciplina INT,

id\_professor INT,

ano\_semestre VARCHAR(10),

FOREIGN KEY (id\_disciplina) REFERENCES Disciplinas(id\_disciplina),

FOREIGN KEY (id\_professor) REFERENCES Professores(id\_professor));

-- **Inserções na tabela Turmas**

INSERT INTO Turmas VALUES (1, 1, 1, '2024/1');

CREATE TABLE **Matriculas** (

id\_matricula INT PRIMARY KEY AUTO\_INCREMENT,

id\_aluno INT,

id\_turma INT,

data\_matricula DATE,

FOREIGN KEY (id\_aluno) REFERENCES Alunos(id\_aluno),

FOREIGN KEY (id\_turma) REFERENCES Turmas(id\_turma)

);

-- **Inserções na tabela Matrículas**

INSERT INTO Matriculas VALUES (1, 1, 1, '2024-02-01');

CREATE TABLE **Notas** (

id\_nota INT PRIMARY KEY AUTO\_INCREMENT,

id\_matricula INT,

nota DECIMAL(4, 2),

avaliacao VARCHAR(50),

FOREIGN KEY (id\_matricula) REFERENCES Matriculas(id\_matricula)

);

-- **Inserções na tabela Notas**

INSERT INTO Notas VALUES (1, 1, 8.5, 'Portifólio');

**GitHub: https://github.com/Excelsior8Elpharah/portif-lio-Database-Modeling-SQL-UNIFECAF/tree/main**

**Conclusão**

Com essa estrutura de banco de dados, a faculdade poderá:

* **Gerenciar dados acadêmicos** de maneira eficiente.
* **Controlar o desempenho dos alunos** e alocação de professores.
* **Facilitar a criação de turmas** e matrículas.
* **Gerar relatórios e históricos** conforme necessário.

A aplicação da **análise de requisitos**, **modelagem conceitual** e **modelagem lógica** garante que o sistema atenda aos critérios solicitados e ofereça um suporte sólido para a gestão acadêmica da instituição.