

BANCO DE DADOS

Trabalho – Relatório

Curso:	CTS ANÁLISE E DESENVOLVIMENTO DE SISTEMAS – DISTÂNCIA
Aluno(a):	Marcus Vinícius da Silva Debon
RU:	4293438

1. 1ª Etapa – Modelagem

Pontuação: 25 pontos.

Dado o estudo de caso abaixo, elabore o Modelo Entidade-Relacionamento (MER), isto é, o modelo conceitual.

O Modelo Entidade-Relacionamento (MER) deve contemplar os seguintes itens:

- Entidades;
- Atributos;
- Relacionamentos;
- Cardinalidades.

Uma companhia aérea necessita controlar os dados de seus voos. Para isso, contratou um profissional de Banco de Dados, a fim de modelar o Banco de Dados que armazenará os dados dos voos.

As regras de negócio são:

- Voo – Deverão ser armazenados os seguintes dados: identificação do voo, número do avião, cidade de origem, cidade destino, data do voo e hora do voo;
- Assentos – Deverão ser armazenados os seguintes dados: identificação do assento e quantidade;
- Passageiro – Deverão ser armazenados os seguintes dados: CPF, nome, telefone, e-mail e endereço (rua, número, complemento, bairro, CEP, cidade e estado);

- Dependentes – Deverão ser armazenados os seguintes dados: nome e data de nascimento;
- Um voo pode ter zero ou vários assentos, assim como zero ou vários assentos pertencem a um voo;
- Um passageiro pode ter zero ou várias reservas de assentos, assim como zero ou várias reservas de assentos pertencem a um passageiro;
- Um passageiro pode ter zero ou vários dependentes, assim como zero ou vários dependentes são de um passageiro;
- Da reserva, deverão ser armazenados os seguintes dados: data da reserva e hora da reserva.

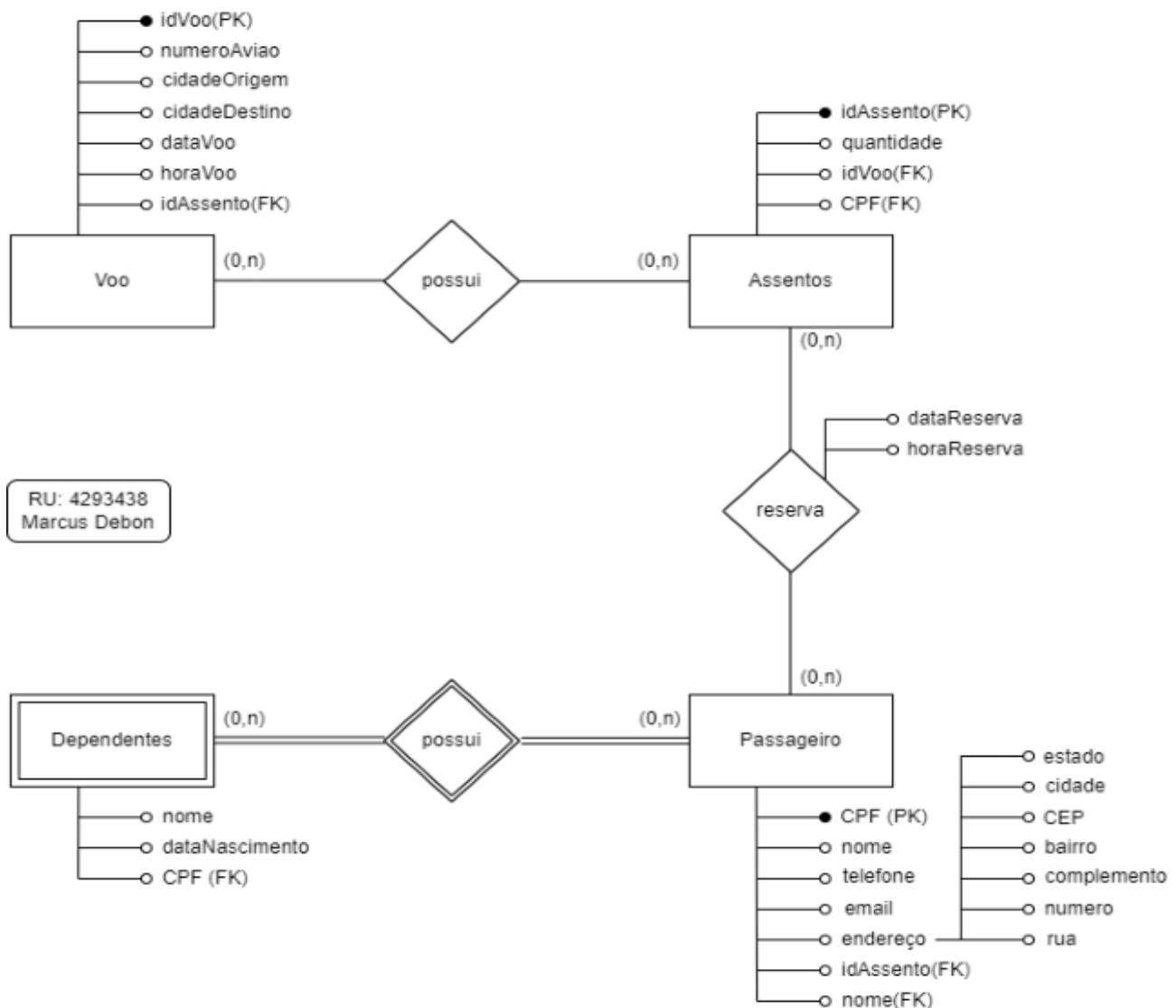
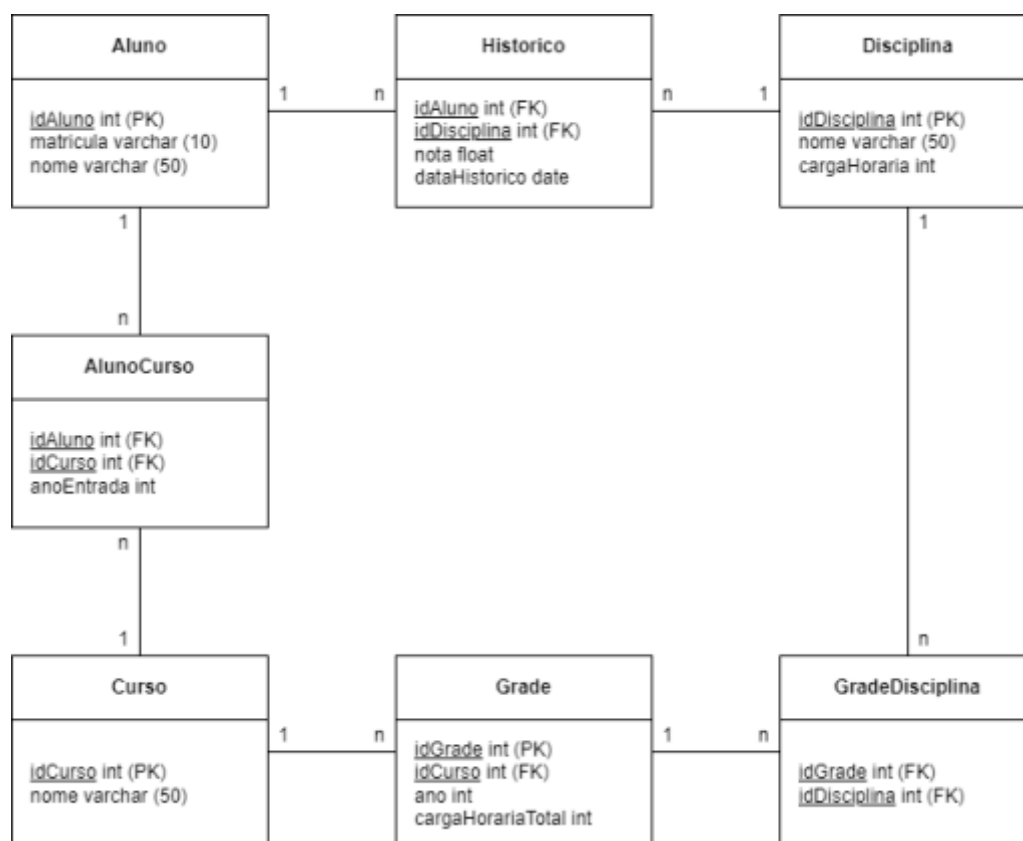


Figura 1: Modelo conceitual

2. 2ª Etapa – Implementação

Considere o seguinte Modelo Relacional (lógico):



Com base no Modelo Relacional dado e utilizando a *Structured Query Language* (SQL), no MySQL Workbench, implemente o que se pede.

Observação: Para testar o Banco de Dados após a criação, utilize os comandos contidos no arquivo “Trabalho – Populando o Banco de Dados”, o qual contém todos os comandos de inserção de dados (fictícios) necessários para a realização dos testes.

Pontuação: 25 pontos.

1. Implemente um Banco de Dados chamado “Faculdade”. Após, crie as tabelas, conforme o Modelo Relacional dado, observando as chaves primárias e as chaves estrangeiras. Todos os campos, de todas as tabelas, não podem ser nulos.

Código desenvolvido:

```
create database Faculdade;  
use Faculdade;
```

```
create table Aluno(  
    idAluno int not null,  
    matricula varchar(10) not null,  
    nome varchar(50) not null,  
    primary key(idAluno)  
);
```

```
create table Disciplina(  
    idDisciplina int not null,  
    nome varchar(50) not null,  
    cargaHoraria int not null,  
    primary key(idDisciplina)  
);
```

```
create table Curso(  
    idCurso int not null,  
    nome varchar(50) not null,  
    primary key(idCurso)  
);
```

```
create table Historico(  
    idAluno int not null,  
    idDisciplina int not null,  
    foreign key(idAluno) references Aluno(idAluno),  
    foreign key(idDisciplina) references Disciplina(idDisciplina),  
    nota float not null,  
    dataHistorico date not null  
);
```

```
create table AlunoCurso(  
    idAluno int not null,  
    idCurso int not null,  
    foreign key(idAluno) references Aluno(idAluno),  
    foreign key(idCurso) references Curso(idCurso),  
    anoEntrada int not null  
);
```

```
create table Grade(  
    idGrade int not null,  
    idCurso int not null,  
    foreign key(idCurso) references Curso(idCurso),  
    ano int not null,  
    cargaHorariaTotal int not null,  
    primary key(idGrade)  
);
```

```
create table GradeDisciplina(  
    idGrade int not null,  
    idDisciplina int not null,  
    foreign key(idGrade) references Grade(idGrade),  
    foreign key(idDisciplina) references Disciplina(idDisciplina)  
);
```

Pontuação: 10 pontos.

2. Implemente uma consulta para listar o quantitativo de cursos existentes.

```
select count(idCurso) as 'Total de cursos' from Curso;
```

Pontuação: 10 pontos.

3. Implemente uma consulta para listar o nome das disciplinas existentes.

```
select nome as 'Disciplinas existentes' from Disciplina;
```

Pontuação: 10 pontos.

4. Implemente uma consulta para listar o nome de todos os cursos e seus respectivos alunos. A listagem deve ser mostrada em ordem decrescente pelo nome dos cursos.

```
select c.nome as 'Nome do curso',a.nome as 'Nome do aluno' from curso c  
inner join alunocurso ac  
on c.idcurso = ac.idcurso  
inner join aluno a  
on a.idaluno = ac.idaluno  
order by c.nome desc, a.nome asc;
```

Pontuação: 10 pontos.

5. Implemente uma consulta para listar a média das notas das disciplinas de todos os cursos. Para isso, utilize o comando *group by*.

```
select avg(nota) as 'Média das notas', nome 'Disciplina' from historico h  
inner join disciplina d  
on h.iddisciplina = d.iddisciplina  
inner join alunocurso ac  
group by disciplina;
```

Pontuação: 10 pontos.

6. Implemente uma consulta para listar o nome de todos os cursos e a quantidade de alunos em cada curso. Para isso, utilize os comandos *join* e *group by*.

```
select nome 'Nome do curso', count(idaluno) 'Nº de Alunos' from curso c  
inner join alunocurso ac  
on c.idcurso = ac.idcurso  
group by nome;
```