**FACULDADES INTEGRADAS DE ARARAQUARA**

**BACHARELADO EM SISTEMAS DE INFORMAÇÃO**

**LUIS FILIPE CAMBIAGHE GUEDES**

**PROJETO DE BANCO DE DADOS**

**TRABALHO 4**

**Prof. Dr. Fábio Papini Fornazari**

**ARARAQUARA – SP.**

**2018**

Sumário

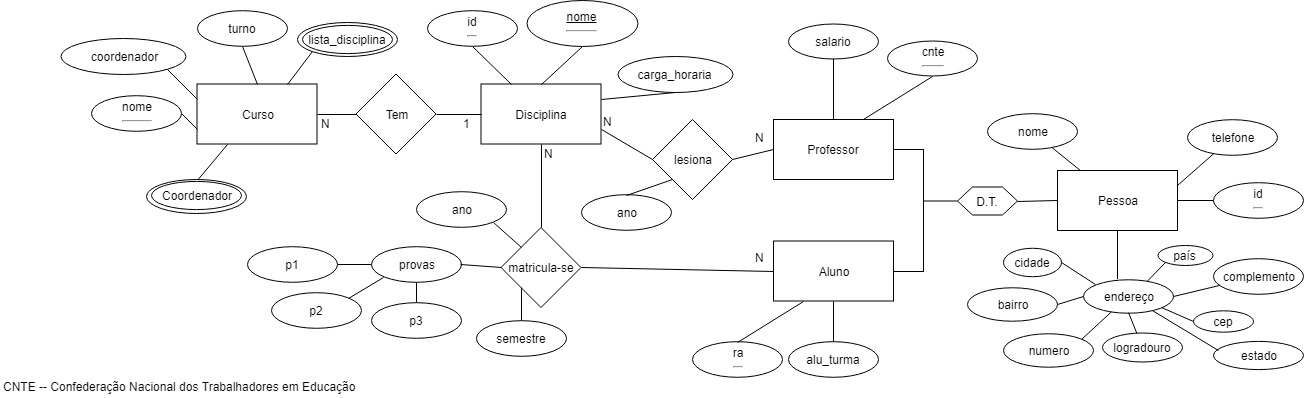
[1. Modelo Entidade Relacionamento (MER): 3](#_Toc517376471)

[2. Modelo relacional: 3](#_Toc517376472)

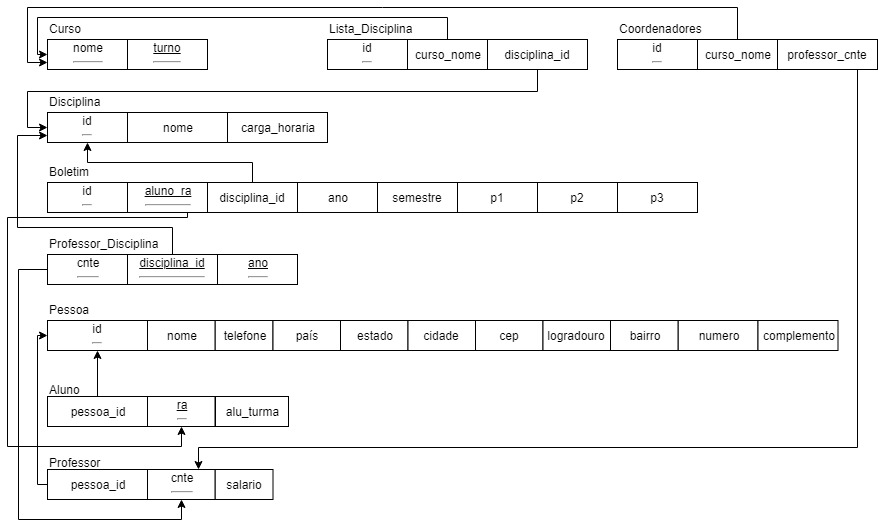
[3. Criação das tabelas ou “create table”: 4](#_Toc517376473)

[4. Consulta: 6](#_Toc517376474)

# Modelo Entidade Relacionamento (MER):



# Modelo relacional:



# Criação das tabelas ou “create table”:

CREATE TABLE Pessoa (

id int identity not null,

nome varchar (100) not null,

    telefone varchar (20) not null,

    pais varchar (100) not null,

estado varchar (100) not null,

cidade varchar (100) not null,

cep varchar (100) not null,

    logradouro varchar (100) not null,

bairro varchar (100) not null,

    numero int not null,

complemento varchar (100),

constraint PK\_Pessoa\_Id primary key (id),

);

CREATE TABLE Aluno (

pessoa\_id int not null,

ra int not null,

alu\_turma varchar (100),

constraint PK\_Aluno\_Id primary key (ra),

constraint REF\_Aluno\_Pessoa\_Id foreign key (pessoa\_id) references Pessoa (id),

constraint UNQ\_Aluno\_Pessoa\_Id unique (pessoa\_id),

);

CREATE TABLE Professor (

pessoa\_id int not null,

cnte int not null,

salario float not null,

constraint PK\_Professor\_Id primary key (cnte),

constraint REF\_Professor\_Pessoa\_Id foreign key (pessoa\_id) references Pessoa (id),

constraint UNQ\_Professor\_Pessoa\_Id unique (pessoa\_id),

);

CREATE TABLE Disciplina (

id int identity not null,

nome varchar (100) not null,

carga\_horaria int not null,

constraint PK\_Disciplina\_Id primary key (id),

);

CREATE TABLE Curso (

nome varchar (100) not null,

turno varchar (100) not null,

constraint PK\_Curso\_Id primary key (nome),

constraint UNQ\_Curso\_Turno unique (turno),

);

CREATE TABLE Lista\_Disciplina

(

id int IDENTITY(1,1) not null,

curso\_nome varchar (100) not null,

disciplina\_id int not null,

Constraint PK\_Lista\_Disciplina\_Id PRIMARY KEY(id),

constraint FK\_Lista\_Disciplina\_Curso\_nome foreign key (curso\_nome) references Curso (nome),

constraint FK\_Lista\_Disciplina\_Disciplina\_Id foreign key (disciplina\_id) references Disciplina (id),

constraint INQ\_Lista\_Disciplina unique (disciplina\_id),

);

CREATE TABLE Coordenadores -- perguntar sobre ano

(

id int IDENTITY(1,1) not null,

curso\_nome varchar (100) not null,

professor\_cnte int not null,

Constraint PK\_Coordenadores\_Id PRIMARY KEY(id),

constraint FK\_Coordenadores\_Curso\_nome foreign key (curso\_nome) references Curso (nome),

constraint FK\_Coordenadores\_Professor\_Cnte foreign key (professor\_cnte) references Professor (cnte),

constraint INQ\_Coordenadores unique (curso\_nome),

);

CREATE TABLE Boletim (

id int identity not null,

aluno\_ra int not null,

disciplina\_id int not null,

ano int not null,

semestre varchar (50) not null,

p1 int not null,

p2 int not null,

p3 int,

constraint PK\_Boletim\_Id primary key (id),

constraint REF\_Boletim\_Aluno\_Ra foreign key (aluno\_ra) references aluno (ra),

constraint REF\_Boletim\_disciplina\_id foreign key (disciplina\_id) references Disciplina (id),

constraint UNQ\_Boletim\_Aluno\_Ra unique (aluno\_ra),

);

CREATE TABLE Professor\_Disciplina (

professor\_cnte int not null,

disciplina\_id int not null,

ano int not null,

constraint PK\_Professor\_Disciplina\_Id primary key (disciplina\_id),

constraint UNQ\_Professor\_Disciplina\_Cnte unique (disciplina\_id, ano),

constraint REF\_Professor\_Disciplina\_Pessoa\_Id foreign key (professor\_cnte) references professor (cnte)

);

# Consulta:

**-- 1. Quais são os alunos de uma determinada disciplina ministrada no ano de 2015,com suas notas..**

select b.aluno\_ra as RA\_do\_Aluno,

        p.nome as Nome\_do\_Aluno,

d.nome as Nome\_da\_Disciplina,

b.ano as No\_Ano\_de,

b.semestre as Semestre,

b.p1 as Nota\_Prova\_1,

b.p2 as Nota\_Prova\_2,

b.p3 as Nota\_Prova\_3

from boletim b

INNER JOIN Aluno a on a.ra = b.aluno\_ra

INNER JOIN Disciplina d on d.id = b.disciplina\_id

        INNER JOIN Pessoa p on p.id = a.pessoa\_id

        where ano = 2015 and d.nome = 'Regar Plantação' order by p.nome

**-- 2. Quais são as notas de um aluno em todas as disciplinas por ele cursadas no ano de 2016.**

select b.aluno\_ra as RA\_do\_Aluno,

        p.nome as Nome\_do\_Aluno,

d.nome as Nome\_da\_Disciplina,

b.ano as No\_Ano\_de,

b.semestre as Semestre,

b.p1 as Nota\_Prova\_1,

b.p2 as Nota\_Prova\_2,

b.p3 as Nota\_Prova\_3

from boletim b

INNER JOIN Aluno a on a.ra = b.aluno\_ra

INNER JOIN Disciplina d on d.id = b.disciplina\_id

        INNER JOIN Pessoa p on p.id = a.pessoa\_id

        where ano = 2016 and

p.nome = 'Luis Felipe Cambiaghe Guedes'

order by d.nome

**-- 3. Quais são os alunos reprovados por nota (média inferior a cinco) no ano de 2016 e o nome das disciplinas e as médias.**

select b.aluno\_ra as RA\_do\_Aluno,

        p.nome as Nome\_do\_Aluno,

d.nome as Nome\_da\_Disciplina,

b.ano as No\_Ano\_de,

b.semestre as Semestre,

b.p1 as Nota\_Prova\_1,

b.p2 as Nota\_Prova\_2,

b.p3 as Nota\_Prova\_3

from boletim b

INNER JOIN Aluno a on a.ra = b.aluno\_ra

INNER JOIN Disciplina d on d.id = b.disciplina\_id

        INNER JOIN Pessoa p on p.id = a.pessoa\_id

WHERE (b.p1 + b.p2)/2 < 5

AND (

CASE

When b.p1 > b.p2 Then (b.p1 + b.p3)/2

Else (b.p2 + b.p3)/2

END

) < 5

AND b.ano = 2016