



**UAST**  
Unidade Acadêmica  
de Serra Talhada - PE  
*Desde 2006*



**PROJETO DE DISCIPLINA:  
PROJETO DE BANCO DE DADOS**

**Vinicius Alexandre Feitoza**

**SERRA TALHADA, PE  
[OUTUBRO/2019]**

**Vinicius Alexandre Feitoza**

Projeto desenvolvido para efetivação da [1ª ou 2ª ou 3ª] etapa apresentado para avaliação na Disciplina Projeto de Banco de Dados Relacionais ministrada pelo Prof. Hidelberg Oliveira, período letivo 2019.2.

## PROJETO DE BANCO DE DADOS

### 01. DEFINIÇÃO DA PROBLEMÁTICA (DOMÍNIO DA APLICAÇÃO)

O Sistema acadêmico ARGUS é voltado para escolas do Ensino Fundamental e Médio. O objetivo do sistema é de prover o acompanhamento do aluno em todas as fases de seu crescimento estudantil, desde o acompanhamento de notas, pagamentos e pedagógico.

### 02. DEFINIÇÃO DE REQUISITOS

#### 2.1. Requisitos Funcionais:

<b>Identificação:</b>	[RF01] Manter Pessoa
<b>Descrição:</b>	<i>Manter inclui o cadastro, a remoção, a atualização e a busca do Pessoas no banco de dados.</i>
<b>Prioridade:</b>	(X) Essencial   ( ) Importante   ( ) Desejável

<b>Identificação:</b>	[RF02] Manter Matricula
<b>Descrição:</b>	<i>Manter inclui o cadastro, a remoção, a atualização e a busca da Matricula do aluno no banco de dados.</i>
<b>Prioridade:</b>	( x ) Essencial   ( ) Importante   ( ) Desejável

<b>Identificação:</b>	[RF03] Manter Disciplina
<b>Descrição:</b>	<i>Manter inclui o cadastro, a remoção, a atualização e a busca das Disciplinas no banco de dados.</i>
<b>Prioridade:</b>	( x ) Essencial   ( ) Importante   ( ) Desejável

<b>Identificação:</b>	[RF04] Manter Acompanhamento Pedagógico
<b>Descrição:</b>	<i>Manter inclui o cadastro, a remoção, a atualização e a busca do Acompanhamento Pedagógico, no banco de dados.</i>
<b>Prioridade:</b>	( x ) Essencial   ( ) Importante   ( ) Desejável

<b>Identificação:</b>	[RF05] Manter Notas
<b>Descrição:</b>	<i>Manter inclui o cadastro, a remoção, a atualização e a busca das Notas do aluno no banco de dados.</i>
<b>Prioridade:</b>	( x ) Essencial   ( ) Importante   ( ) Desejável

<b>Identificação:</b>	[RF06] Cadastrar Administração
<b>Descrição:</b>	<i>Cadastrar perfis de usuários como Administradores do Sistema.</i>
<b>Prioridade:</b>	( ) Essencial   ( x ) Importante   ( ) Desejável

<b>Identificação:</b>	[RF07] Realizar Login
<b>Descrição:</b>	<i>O Sistema solicita o Login e a Senha ao Usuário para a validação de acesso.</i>
<b>Prioridade:</b>	( x ) Essencial    ( ) Importante    ( ) Desejável

<b>Identificação:</b>	[RF08] Resetar Senha
<b>Descrição:</b>	<i>O Usuário pode solicitar ao administrador resetar sua senha, após isso o usuário deverá criar um nova senha.</i>
<b>Prioridade:</b>	( ) Essencial    ( x ) Importante    ( ) Desejável

<b>Identificação:</b>	[RF09] Quadro de Solicitações
<b>Descrição:</b>	<i>O quadro de solicitações apresenta as solicitações dos demais usuários ao administrador principal do sistema.</i>
<b>Prioridade:</b>	( ) Essencial    ( ) Importante    (x) Desejável

<b>Identificação:</b>	[RF10] Gerar Log
<b>Descrição:</b>	O sistema deverá para cada ação de inserção, atualização e remoção feita no sistema gerar um log de alteração.
<b>Prioridade:</b>	( ) Essencial    ( ) Importante    (x) Desejável

<b>Identificação:</b>	[RF11] Gerar Log
<b>Descrição:</b>	O sistema deverá para cada ação de inserção, atualização e remoção feita no sistema gerar um log de alteração.
<b>Prioridade:</b>	( ) Essencial    ( ) Importante    (x) Desejável

## 2.2. Requisitos Não-Funcionais:

<b>Identificação:</b>	[RNF01] Disponibilizar Dados
<b>Tipo:</b>	Disponibilidade
<b>RF Relacionado:</b>	Nenhum
<b>Descrição:</b>	O sistema funcionará 24 horas por dia.
<b>Prioridade:</b>	(x) Essencial    ( ) Importante    ( x ) Desejável

<b>Identificação:</b>	[RNF02] Utilização da Linguagem Java
<b>Tipo:</b>	Padrão
<b>RF Relacionado:</b>	Nenhum
<b>Descrição:</b>	O projeto foi desenvolvido com a linguagem Java.
<b>Prioridade:</b>	( ) Essencial    ( ) Importante    ( x ) Desejável

<b>Identificação:</b>	[RNF03] Acessar Dados
<b>Tipo:</b>	Segurança

<b>RF Relacionado:</b>	Todos
<b>Descrição:</b>	O sistema é controlado por login e senha, permitindo definir diferentes níveis de acesso ao sistema através de permissões de acordo com o perfil do usuário, logo os dados estão protegidos contra modificações e exclusões não autorizadas.
<b>Prioridade:</b>	( x ) Essencial   ( ) Importante   ( ) Desejável

<b>Identificação:</b>	[RNF04] Armazenamento de Dados
<b>Tipo:</b>	Integridade
<b>RF Relacionado:</b>	RF01, RF02, RF03, RF04, RF05, RF06
<b>Descrição:</b>	Os dados inseridos, alterados ou removidos permanecem na situação desejada pelos operadores.
<b>Prioridade:</b>	( ) Essencial   ( x ) Importante   ( ) Desejável

<b>Identificação:</b>	[RNF05] Backup
<b>Tipo:</b>	Segurança
<b>RF Relacionado:</b>	Nenhum
<b>Descrição:</b>	O sistema deve ser configurado para fazer o backup do banco de dados uma vez por dia, forçando o usuário a fazer o backup que sempre que alcançar uma determinada hora do dia.
<b>Prioridade:</b>	( ) Essencial   ( x ) Importante   ( x ) Desejável

<b>Identificação:</b>	[RNF06] Padrão de Projeto
<b>Tipo:</b>	Implementação
<b>RF Relacionado:</b>	Nenhum
<b>Descrição:</b>	O sistema deve usar os padrões de projeto MVC e DAO.
<b>Prioridade:</b>	( ) Essencial   ( x ) Importante   ( x ) Desejável

### 2.3. Requisitos de Domínio (se existir):

<b>Identificação:</b>	[RNF01] Matrícula
<b>Descrição:</b>	<ul style="list-style-type: none"> <li>• Um aluno não pode pular de anos. Uma vez matriculado, deve seguir o fluxo.</li> <li>• Considere que não há transferidos. O sistema não pode permitir matrícula em anos anteriores não cursados ou reprovados.</li> <li>• Um aluno pode se matricular em um currículo anual progressivo, composto por: <ul style="list-style-type: none"> <li>• EF: Anos Iniciais (anos): 1º ao 5º</li> <li>• EF: Anos Finais (anos): 6º ao 9º</li> <li>• EM: (anos): 1º ao 3º.</li> </ul> </li> <li>• Um responsável financeiro deve obrigatoriamente possuir CPF.</li> <li>• Se o aluno for maior de idade, ele pode ser responsável financeiro por si próprio, mas o inverso deve ser proibido.</li> <li>• No momento da matrícula, o sistema deve gerar as 12 parcelas referentes aos 12 meses do aluno, conforme tabela a seguir: <ul style="list-style-type: none"> <li>• EF-Anos Iniciais: 400,00</li> </ul> </li> </ul>

	<ul style="list-style-type: none"> <li>● EF-Anos Finais: 600,00</li> <li>● EM: 800,00</li> </ul>
<b>Prioridade:</b>	( ) Essencial    (x) Importante    ( ) Desejável

<b>Identificação:</b>	[RD02] Senhas
<b>Descrição:</b>	<ul style="list-style-type: none"> <li>• O campo de senha dos usuários deve aceitar caracteres alfanuméricos e numéricos, com tamanho entre 6 a 20, com distinção de caracteres maiúsculos de minúsculos. A senha deverá ser criptografada no banco de dados.</li> <li>• Os usuários são cadastrados por grupos, dentro do sistema, definidos pelo administrador.</li> <li>• Cada usuário do sistema tem uma senha de acesso, com permissão de alterá-la.</li> <li>• No momento do cadastramento do usuário, deve ser gerada uma senha padrão.</li> <li>• Caso o usuário perca a senha, deverá solicitar pelo sistema ao administrador, que deverá receber um aviso no sistema da solicitação.</li> </ul>
<b>Prioridade:</b>	( ) Essencial    (x) Importante    ( ) Desejável

<b>Identificação:</b>	[RD03] Controle dos dados por usuário
<b>Descrição:</b>	<ul style="list-style-type: none"> <li>• Os funcionários da secretaria deveram cadastrar as médias dos alunos e as disciplinas.</li> <li>• A Direção não pode fazer alterações, somente visualização.</li> <li>• A Coordenação Pedagógica somente deve cadastrar os dados de acompanhamento.</li> <li>• Somente o administrador cadastra os funcionários (em todos os níveis).</li> </ul>
<b>Prioridade:</b>	( ) Essencial    (x) Importante    ( ) Desejável

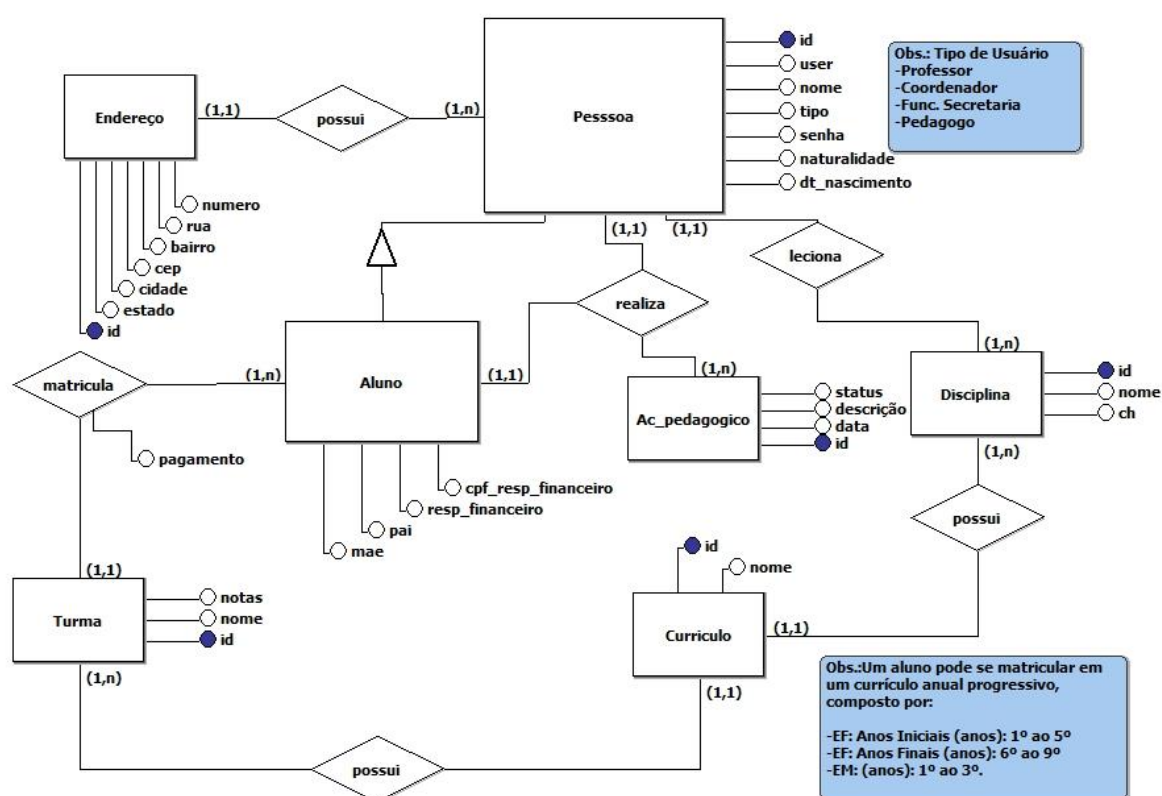
<b>Identificação:</b>	[RD04] Disciplinas
<b>Descrição:</b>	<ul style="list-style-type: none"> <li>• As notas das disciplinas devem ser cadastradas dentro do intervalo de 0.0 à 10.0, com até um algarismo de ponto flutuante.</li> <li>• Para ser aprovado:  Por Média: precisa obter média parcial <math>\geq 7.0</math>  Na Final: precisa obter média geral <math>\geq 5.0</math> </li> </ul>
<b>Prioridade:</b>	( ) Essencial    (x) Importante    ( ) Desejável

<b>Identificação:</b>	[RD05] Currículo
<b>Descrição:</b>	<ul style="list-style-type: none"> <li>• O currículo pode ser baseado em bimestres ou trimestres, configurável pelo sistema.</li> <li>• Se bimestres: Cada aluno em cada disciplina deve obter: <ul style="list-style-type: none"> <li>• Nota de 4 bimestres.</li> <li>• Média Parcial.</li> <li>• Cálculo: média aritmética dos 4 bimestres</li> <li>• Média para aprovação: 7.0</li> <li>• Prova final</li> <li>• Média Geral.</li> </ul> </li> <li>• Se o aluno for aprovado por média, esta média é igual à parcial.</li> <li>• Se foi para final: <ul style="list-style-type: none"> <li>• <math>(\text{Média Parcial} + \text{Prova Final})/2</math></li> </ul> </li> <li>• Se trimestres: <ul style="list-style-type: none"> <li>• Cada aluno em cada disciplina deve obter: <ul style="list-style-type: none"> <li>• Nota de 3 trimestres.</li> <li>• Média Parcial.</li> <li>• Cálculo: média aritmética dos 3 bimestres</li> <li>• Média para aprovação: 7.0</li> <li>• Prova final</li> <li>• Média Geral.</li> </ul> </li> <li>• Se o aluno for aprovado por média, esta média é igual à parcial.</li> <li>• Se foi para final: <ul style="list-style-type: none"> <li>• <math>(\text{Média Parcial} + \text{Prova Final})/2</math></li> </ul> </li> </ul> </li> <li>• Para ambos (bi ou trimestres): Status dos alunos: <ul style="list-style-type: none"> <li>• Por disciplina: <ul style="list-style-type: none"> <li>• AM = aprovado por média</li> <li>• AP = aprovado</li> <li>• RP = reprovado.</li> </ul> </li> <li>• Por currículo: <ul style="list-style-type: none"> <li>• APROVADO</li> <li>• REPROVADO</li> </ul> </li> </ul> </li> </ul>
<b>Prioridade:</b>	( ) Essencial    (x) Importante    ( ) Desejável

<b>Identificação:</b>	[RD06] Acompanhamento pedagógico
<b>Descrição:</b>	<ul style="list-style-type: none"> <li>• O sistema também deve fornecer o acompanhamento</li> </ul>

	<p>pedagógico dos alunos.</p> <ul style="list-style-type: none"> <li>O sistema deve emitir o: <ul style="list-style-type: none"> <li>Relatório do acompanhamento por aluno.</li> <li>Relatório de acompanhamento por profissional (relação de nome de alunos e datas de atendimento).</li> </ul> </li> </ul>
<b>Prioridade:</b>	( ) Essencial    (x) Importante    ( ) Desejável

### 03. DIAGRAMA ENTIDADE-RELACIONAMENTO



**Figura 1 - Diagrama Entidade Relacionamento**



#### 04. MAPEAMENTO DO ESQUEMA CONCEITUAL PARA O ESQUEMA LÓGICO-RELACIONAL

##### Regra 1: Mapeamentos das entidades fortes

Endereço (**id**, estado, cidade, bairro, cep, rua)

Pessoa (**id**, nome, naturalidade, dta\_nasc, tipo, user, senha)

Disciplina (**id**, nome, ch)

Turma (**id**, nome, notas)

Ac\_pedagogico (**id**, data, descrição, status)

Aluno (**id**, pai, mae, resp\_financ, cpf\_resp\_finance, *id\_pessoa*)

*id\_endereco* referência entidade Endereço.

Curriculo (**id**, nome)

Matricula (**id**, pagamento)

##### Regra 2: Mapeamentos das entidades fracas

Neste projeto não serão considerados entidades fracas.

##### Regra 3: Mapeamento de Relacionamentos Binários Um para Um (1:1)

Neste projeto não houve nenhum relacionamento binário (1:1) entre as entidades existentes no DER

##### Regra 4: Mapeamento de Relacionamentos Binários Um para Muitos (1:N)

Pessoa (**id**, nome, naturalidade, dta\_nasc,, tipo, user, senha, *id\_endereco*)

*id\_endereco* referência entidade Endereço.

<p>Disciplina (<b>id</b>, nome, ch, <i>id_pessoa</i>)</p> <p><i>id_pessoa</i> faz referência a entidade Pessoa do tipo Professor</p>
<p>Turma (<b>id</b>, nome, notas, <i>id_curriculo</i>)</p> <p><i>id_curriculo</i> faz referência a entidade Curriculo</p>
<p>Ac_pedagogico (<b>id</b>, data, desc, status, <i>id_pessoa</i>, <i>id_aluno</i>)</p> <p><i>id_pessoa</i> faz referência a entidade Aluno</p> <p><i>id_pessoa</i> faz referência a entidade Pessoa do tipo Pedagogo</p>
<p>Matricula (<b>id</b>, <i>id_aluno</i>, <i>id_turma</i>, pagamento)</p> <p><i>id_aluno</i> faz referência a entidade Aluno</p> <p><i>id_turma</i> faz referência a entidade Turma</p>

#### Regra 5: Mapeamento de Relacionamentos Binários Muitos para Muitos (N:M)

Obs: não houve nenhum relacionamento binário entre as entidades existentes no DER

#### Regra 6: Mapeamento de atributos multivalorados

Obs: não houve nenhum atributo multivalorado entre as entidades existentes no DER

#### Regra 7: Mapeamento de Relacionamentos N-ários

Obs: não houve nenhum relacionamento N-ário entre as entidades existentes no DER

## 05. DICIONÁRIO DE DADOS

Pessoa			
Atributo	Descrição	Tipo de Dado	Restrições
Id	Atributo que representa o código de identificação da Pessoa	Long	<ul style="list-style-type: none"><li>• Primary Key</li><li>• Auto Increment</li><li>• Not Null</li></ul>
Nome	Atributo que representa o nome da Pessoa.	String	<ul style="list-style-type: none"><li>• Not Null</li></ul>
User	Atributo que representa o login da Pessoa.	String	<ul style="list-style-type: none"><li>• Not Null</li><li>• Unique</li></ul>
Senha	Atributo que representa a senha da Pessoa.	String	<ul style="list-style-type: none"><li>• Not Null</li></ul>
Naturalidade	Atributo que representa naturalidade da Pessoa.	String	
Dta_nasc	Atributo que representa a data de nascimento da Pessoa.	Date	
Tipo	Atributo que representar o tipo de Pessoa	String	<ul style="list-style-type: none"><li>• Not Null</li><li>• Unique</li></ul>
Id_endereco	Atributo que referencia o código de identificação do Endereço da Pessoa.	Long	<ul style="list-style-type: none"><li>• Foreign Key (Tabela Endereço)</li></ul>

Endereço			
Atributo	Descrição	Tipo de Dado	Restrições
Id	Atributo que representa o código de identificação do Endereço.	Long	<ul style="list-style-type: none"><li>• Primary Key</li><li>• Auto Increment</li><li>• Not Null</li></ul>
Estado	Atributo que representa o nome do estado do Endereço.	String	<ul style="list-style-type: none"><li>• Not Null</li></ul>
Cidade	Atributo que representa o nome da cidade do Endereço.	String	<ul style="list-style-type: none"><li>• Not Null</li></ul>
Bairro	Atributo que representa o nome do bairro do Endereço	String	
Cep	Atributo que representa o CEP do Endereço.	String	
Rua	Atributo que representa o nome da rua do Endereço	String	

Aluno			
Atributo	Descrição	Tipo de Dado	Restrições
Id	Atributo que representa o código de identificação do Aluno.	Long	<ul style="list-style-type: none"> <li>Primary Key</li> <li>Auto Increment</li> <li>Not Null</li> </ul>
Mae	Atributo que representa o nome da mãe do Aluno.	String	<ul style="list-style-type: none"> <li>Not Null</li> </ul>
Pai	Atributo que representa o nome do pai do Aluno.	String	<ul style="list-style-type: none"> <li>Not Null</li> </ul>
resp_financ	Atributo que representa o responsável financeiro do Aluno.	String	<ul style="list-style-type: none"> <li>Not Null</li> </ul>
cpf_resp_financ e	Atributo que representa o CPF do responsável financeiro do Aluno.	String	<ul style="list-style-type: none"> <li>Not Null</li> </ul>
Id_pessoa	Atributo que referência código de identificação da Pessoa do Aluno.	Long	<ul style="list-style-type: none"> <li>Foreign Key (Tabela Pessoa)</li> </ul>

Matricula			
Atributo	Descrição	Tipo de Dado	Restrições
Id	Atributo que representa o código de identificação da Matricula.	Long	<ul style="list-style-type: none"> <li>Primary Key</li> <li>Auto Increment</li> <li>Not Null</li> </ul>
Pagamento	Atributo que representa o valor pago da Matricula.	Double	<ul style="list-style-type: none"> <li>Not Null</li> </ul>
Id_aluno	Atributo que referência código de identificação do Aluno na Matricula.	Long	<ul style="list-style-type: none"> <li>Foreign Key (Tabela Aluno)</li> </ul>
Id_turma	Atributo que referência código de identificação da Turma do Aluno na Matricula.	Long	<ul style="list-style-type: none"> <li>Foreign Key (Tabela Turma)</li> </ul>

Turma			
Atributo	Descrição	Tipo de Dado	Restrições
Id	Atributo que representa o código de identificação da Turma	Long	<ul style="list-style-type: none"> <li>Primary Key</li> <li>Auto Increment</li> <li>Not Null</li> </ul>
Nome	Atributo que representa o nome da Turma.	String	<ul style="list-style-type: none"> <li>Not Null</li> </ul>
Notas	Atributo que representa as notas da Turma.	String	<ul style="list-style-type: none"> <li>Not Null</li> </ul>
Id_curriculo	Atributo que referência código de identificação do Curriculo da Turma.	Long	<ul style="list-style-type: none"> <li>Foreign Key (Tabela Curriculo)</li> </ul>

Curriculo			
Atributo	Descrição	Tipo de Dado	Restrições
Id	Atributo que representa o código de identificação do Curriculo.	Long	<ul style="list-style-type: none"> <li>Primary Key</li> <li>Auto Increment</li> <li>Not Null</li> </ul>
Nome	Atributo que representa o nome do Curriculo.	String	<ul style="list-style-type: none"> <li>Not Null</li> </ul>

Disciplina			
Atributo	Descrição	Tipo de Dado	Restrições
Id	Atributo que representa o código de identificação do Disciplina.	Long	<ul style="list-style-type: none"> <li>Primary Key</li> <li>Auto Increment</li> <li>Not Null</li> </ul>
Nome	Atributo que representa o nome do Disciplina.	String	<ul style="list-style-type: none"> <li>Not Null</li> </ul>
Ch	Atributo que representa a carga-horária da Disciplina.	Int	<ul style="list-style-type: none"> <li>Not Null</li> </ul>
Id_curriculo	Atributo que referencia código de identificação do currículo da Disciplina.	Long	<ul style="list-style-type: none"> <li>Foreign Key (Tabela Curriculo)</li> </ul>
Id_pessoa	Atributo que referencia código de identificação da Pessoa da Disciplina.	Long	<ul style="list-style-type: none"> <li>Foreign Key (Tabela Pessoa)</li> </ul>

Ac_pedagogico			
Atributo	Descrição	Tipo de Dado	Restrições
Id	Atributo que representa o código de identificação do Ac_pedagogico.	Long	<ul style="list-style-type: none"> <li>Primary Key</li> <li>Auto Increment</li> <li>Not Null</li> </ul>
Data	Atributo que representa a data do Ac_pedagogico.	Date	<ul style="list-style-type: none"> <li>Not Null</li> </ul>
Status	Atributo que representa o status do Ac_pedagogico.	String	<ul style="list-style-type: none"> <li>Not Null</li> </ul>
Desc	Atributo que representa a descrição do Ac_pedagogico.	String	<ul style="list-style-type: none"> <li>Not Null</li> </ul>
Id_aluno	Atributo que referencia código de identificação do Aluno do Ac_pedagogico.	Long	<ul style="list-style-type: none"> <li>Foreign Key (Tabela Aluno)</li> </ul>
Id_pessoa	Atributo que referencia código de identificação da Pessoa do Ac_pedagogico.	Long	<ul style="list-style-type: none"> <li>Foreign Key (Tabela Pessoa)</li> </ul>

## 05. SCRIPTS SQL

### 5.1 DDL

#### Endereco

```
CREATE TABLE public.endereco
(
  id integer NOT NULL,
  bairro character varying(50) NOT NULL,
  cep character varying(15) NOT NULL,
  cidade character varying(80) NOT NULL,
  estado character varying(3) NOT NULL,
  numero character varying(30) NOT NULL,
  rua character varying(80) NOT NULL,
  CONSTRAINT endereco_pkey PRIMARY KEY (id)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE public.endereco
  OWNER TO postgres;
```

#### Pessoa

```
CREATE TABLE public.pessoa
(
  id integer NOT NULL,
  dtanascimento timestamp without time zone,
  login character varying(20),
  naturalidade character varying(20),
  nome character varying(80) NOT NULL,
  senha character varying(20),
  tipo character varying(20),
  endereco_id integer NOT NULL,
  CONSTRAINT pessoa_pkey PRIMARY KEY (id),
  CONSTRAINT fkl50oo5tcc0u0n3i2k10pnfh8 FOREIGN KEY (endereco_id)
    REFERENCES public.endereco (id) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT uk_fw0821si73kteng9pk5n6wwk3 UNIQUE (endereco_id)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE public.pessoa
  OWNER TO postgres;
```

### Aluno

```
CREATE TABLE public.aluno
(
  id integer NOT NULL,
  dtanascimento timestamp without time zone,
  login character varying(20),
  naturalidade character varying(20),
  nome character varying(80) NOT NULL,
  senha character varying(20),
  tipo character varying(20),
  endereco_id integer NOT NULL,
  cpfrespfinan character varying(20),
  mae character varying(100),
  pai character varying(100),
  respfinan character varying(100),
  turma_id integer,
  CONSTRAINT aluno_pkey PRIMARY KEY (id),
  CONSTRAINT fk_1lyv1j9qqbmi98wjiamep6c2w FOREIGN KEY (endereco_id)
    REFERENCES public.endereco (id) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT fkc2jbyom2ws00xwire0hms5bny FOREIGN KEY (turma_id)
    REFERENCES public.turma (id) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT uk_1lyv1j9qqbmi98wjiamep6c2w UNIQUE (endereco_id)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE public.aluno
  OWNER TO postgres;
```

### Matricula

```
CREATE TABLE public.matricula
(
  id integer NOT NULL,
  valormensalidade double precision,
  valortotal double precision,
  aluno_id integer NOT NULL,
  turma_id integer NOT NULL,
  CONSTRAINT matricula_pkey PRIMARY KEY (id),
  CONSTRAINT fk3rpufqk0vd9j50nvu24h3vg1u FOREIGN KEY (aluno_id)
    REFERENCES public.aluno (id) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT fk5cbvj745fx2o25diw6b4ccr11 FOREIGN KEY (turma_id)
    REFERENCES public.turma (id) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT uk_2qws1v68dbl6ptgei17t48biq UNIQUE (turma_id),
  CONSTRAINT uk_pytlp6gf4mutb4wmofntslkmk UNIQUE (aluno_id)
)
```

```
WITH (  
  OIDS=FALSE  
);  
ALTER TABLE public.matricula  
  OWNER TO postgres;
```

### **Turma**

```
CREATE TABLE public.turma  
(  
  id integer NOT NULL,  
  nome character varying(20),  
  notas double precision,  
  curriculo_id bigint NOT NULL,  
  CONSTRAINT turma_pkey PRIMARY KEY (id),  
  CONSTRAINT fk34xmlat8wcrbyqt5q1ol68gtk FOREIGN KEY (curriculo_id)  
    REFERENCES public.curriculo (id) MATCH SIMPLE  
    ON UPDATE NO ACTION ON DELETE NO ACTION,  
  CONSTRAINT uk_jyduy8lkxaboql9jxgywr5gkk UNIQUE (curriculo_id)  
)  
WITH (  
  OIDS=FALSE  
);  
ALTER TABLE public.turma  
  OWNER TO postgres;
```

### **Curriculo**

```
CREATE TABLE public.curriculo  
(  
  id bigint NOT NULL,  
  nome character varying(20),  
  CONSTRAINT curriculo_pkey PRIMARY KEY (id)  
)  
WITH (  
  OIDS=FALSE  
);  
ALTER TABLE public.curriculo  
  OWNER TO postgres;
```

### **Disciplina**

```
CREATE TABLE public.disciplina  
(  
  id integer NOT NULL,  
  ch integer,  
  nome character varying(100),  
  curriculo_id bigint,  
  pessoa_id integer,  
  CONSTRAINT disciplina_pkey PRIMARY KEY (id),  
  CONSTRAINT fkqu0gqrt0g1acye4rt8k6bi8yo FOREIGN KEY (curriculo_id)  
    REFERENCES public.curriculo (id) MATCH SIMPLE
```



```

    ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
    OIDS=FALSE
);
ALTER TABLE public.disciplina
    OWNER TO postgres;

```

### Ac\_pedagogico

```

CREATE TABLE public.acpedagogico
(
    id integer NOT NULL,
    data timestamp without time zone,
    descricao character varying(1000),
    status character varying(100),
    aluno_id integer NOT NULL,
    pessoa_id integer NOT NULL,
    CONSTRAINT acpedagogico_pkey PRIMARY KEY (id),
    CONSTRAINT fkq8m4mw8gssjtakakqg7h4c39x FOREIGN KEY (aluno_id)
        REFERENCES public.aluno (id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT uk_ab2nb4dxdj648eu42v8lvibw0 UNIQUE (pessoa_id),
    CONSTRAINT uk_fg8vkstlihk8l3foj5t845rew UNIQUE (aluno_id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE public.acpedagogico
    OWNER TO postgres;

```

## 5.2. DML

### Login (busca usuário por login e senha)

```
select p from Pessoa p where p.login = :login and p.senha = :senha
```

### Buscar Disciplina (busca os dados cadastrados de disciplina)

```
select * from disciplina
```

### Buscar Aluno (busca os dados cadastrados de aluno)

```
select * from aluno
```

**Buscar Acompanhamento Pedagógico**  
(busca os dados cadastrados do acompanhamento pedagógico)

select \* from acpedagogico

**Buscar Matricula**  
(busca os dados cadastrados da matricula)

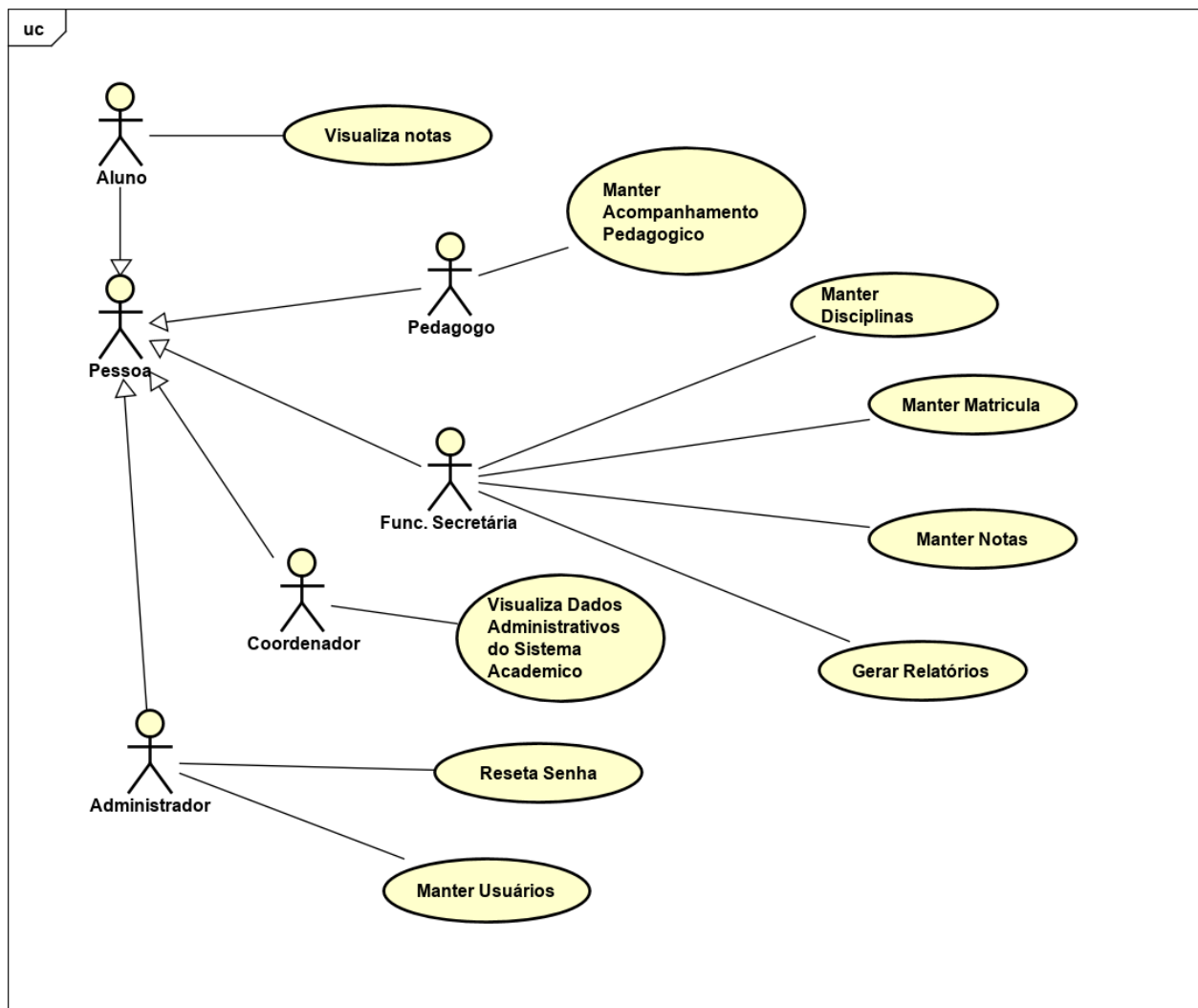
select \* from matricula

**Buscar Pessoa**  
(busca os dados cadastrados de pessoa)

select \* from pessoa

## 06. ANEXOS

### 6.1 Diagrama de Caso de Uso



*Figura 2 - Diagrama de Caso de Uso*