Documento de Banco de Dados

e-Certificate

<Versão 1.0>



**Histórico de Alterações**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| 10/11/2015 | 1.0 | Criação da versão inicial do documento | Amanda Souza, Edvaldo Azevedo |

**Tabela de Siglas**

|  |  |
| --- | --- |
| ***Ator*** | ***Sistema*** |
| *e-Certificate* | *Simulador de Certificações* |
| *TI* | *Tecnologia da Informação* |
| *PK* | *Primary Key* |
| *FK* | *Foreign Key* |
| *Postgres* | *Sistema gerenciador de banco de dados* |

Sumário

[Sumário 4](#_Toc435373829)

[1. Introdução 6](#_Toc435373830)

[1.1 Visão geral deste documento 6](#_Toc435373831)

[1.2 Convenções, termos e abreviações. 6](#_Toc435373832)

[1.2.1 Identificação dos requisitos 6](#_Toc435373833)

[1.2.2 Prioridades dos requisitos. 6](#_Toc435373834)

[1.2.3 Outros termos importantes. 6](#_Toc435373835)

[2. Descrição geral do sistema 7](#_Toc435373836)

[2.1 Especificação do Produto 7](#_Toc435373837)

[2.2 Descrição dos Usuários 8](#_Toc435373838)

[Usuários do Sistema e-Certificate 8](#_Toc435373839)

[*3.* *Módulo Modelos* 9](#_Toc435373841)

[3.1 Modelo Conceitual 9](#_Toc435373842)

[3.2 Modelo Lógico 9](#_Toc435373843)

[4. Dicionário de Dados 10](#_Toc435373844)

[[TB 001 – Aluno] 10](#_Toc435373845)

[[TB 002 – Area] 10](#_Toc435373846)

[[TB 003 – Professor] 11](#_Toc435373847)

[[TB 004 – Usuario] 11](#_Toc435373848)

[[TB 005– Questão] 12](#_Toc435373849)

[[TB 006 – Prova] 12](#_Toc435373850)

[[TB 007 – Questao\_Prova] 13](#_Toc435373851)

[[TB 008 – Historico] 13](#_Toc435373852)

[5. Códigos 14](#_Toc435373853)

[5.1 DDL 14](#_Toc435373854)

[5.2 DML 16](#_Toc435373855)

# Introdução

Este documento é destinado ao cliente e aos integrantes da equipe de desenvolvimento do Sistema *e-Certificate*, servindo como instrumento de comunicação importante, tanto entre os analistas e o cliente, quanto entre os desenvolvedores do sistema durante todo o seu ciclo de vida. O seu objetivo é apresentar os requisitos para o desenvolvimento da base de dados do sistema, e simular o preenchimento dos dados do software.

## Visão geral deste documento

Esta introdução fornece as informações necessárias para fazer um bom uso deste documento, explicitando seus objetivos e as convenções que foram adotadas no texto, além de conter uma lista de referências para outros documentos relacionados. As demais seções apresentam a especificação do Sistema *e-Certificate* e estão organizadas como descrito abaixo.

**Seção 2 – Descrição geral do sistema**: apresenta uma visão geral do sistema, caracterizando qual é o seu escopo e descrevendo suas qualificações.

**Seção 3 – Modelos Lógico e Conseitual**: apresenta os modelos lógico e conceitual do banco de dados.

**Seção 4 – Dicionário de Dados**: especifica todas as tabelas e os seus respectivos atributos para o sistema.

## Convenções, termos e abreviações.

A correta interpretação deste documento exige o conhecimento de algumas convenções e termos específicos, que são descritos a seguir.

### Identificação dos requisitos

Por convenção, a referência das tabelas são feitas através do identificador da tabela, seguido do nome da tabela onde elas estão inseridas, de acordo com o esquema abaixo:

[*Identificador da tabela (TB 001)- Nome da tabela*]

### Prioridades dos requisitos.

Para estabelecer a prioridade das tabelas, foram adotadas as denominações “essencial”, “importante” e “desejável”.

**Essencial** é a tabela sem o qual o sistema não entra em funcionamento. Tabelas essenciais são imprescindíveis ao sistema.

**Importante** é a tabela sem o qual o sistema entra em funcionamento, mas de forma não satisfatória. Tabelas importantes devem ser implementados, mas, se não forem, o sistema poderá ser implantado e usado mesmo assim.

**Desejável** é a tabela que não compromete as funcionalidades básicas do sistema, isto é, o sistema pode funcionar de forma satisfatória sem ela. Tabelas desejáveis podem ser deixadas para versões posteriores do sistema, caso não haja tempo hábil para implementá-las na versão que está sendo especificada.

### Outros termos importantes.

**Pré-condições** é o termo que representa o estado em que a tabela deve estar para realizar a sua execução.

**Ator** é o termo que representa tudo que troca informações com o sistema, incluindo os usuários. São identificados por um nome**.**

# Descrição geral do sistema

O e*-Certificate* é um sistema para certificação de candidatos utilizando banco de questões próprio, permitindo que as provas sejam aplicadas remotamente (através da internet) ou em redes locais. O acesso ao sistema é concedido apenas aos usuários cadastrados e mediante senha criptografada. Sua arquitetura permite ajustes de acordo com a necessidade do cliente.

Através de uma plataforma completa para elaborar, publicar e aplicar provas de certificações, o *e-Certificate* atende aos rigorosos processos de certificação em diversas áreas do conhecimento.

O produto disponibiliza, ainda, a geração de ranking do aluno de forma instantânea, além da disponibilização de históricos do simulado.

## Especificação do Produto

O *e-Certificate* automatiza o grande volume de informação para a elaboração de uma prova, de maneira rápida e intuitiva. De acordo com as especificações para elaboração automática de provas, o *e-Certificate* dispõe:

* **Questões** - As questões podem ser dos tipos V ou F ou Múltipla Escolha com quantidade

ilimitada de alternativas. Podem ser classificadas por área.

* **Área -** Podem ser cadastradas obedecendo a uma hierarquia de áreas.
* **Aprovação -** O aluno é considerado aprovado caso obtenha um número maior de acertos ao de erros.

## Descrição dos Usuários

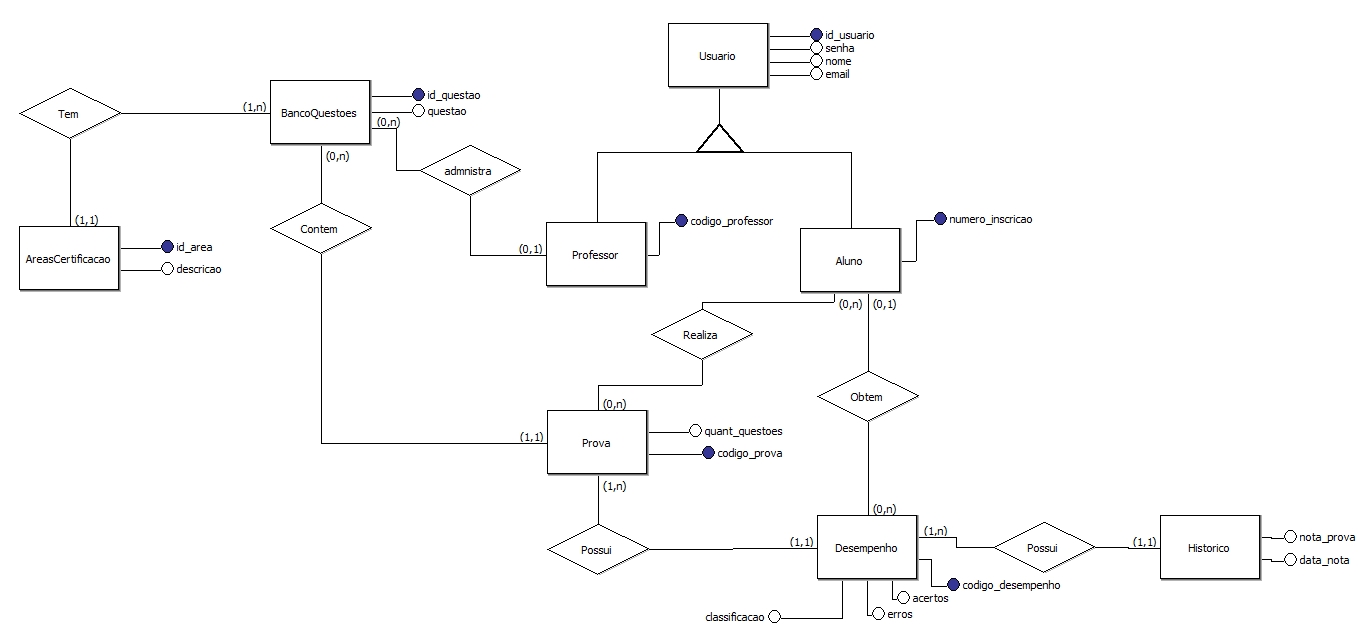
|  |
| --- |
| Usuários do Sistema e-Certificate |
|  |

|  |  |
| --- | --- |
| ***Ator*** | ***Sistema*** |
| *Administrador* | *Usuário que terá acesso a todas as funções do sistema, incluindo algumas funções adicionais para manutenção do software. Este papel será incorporado por um colaborador gerencial e equipes de TI da empresa ou do site.* |
| *Professor* | *Usuário responsável pelas principais funcionalidades do sistema, como preenchimento das questões de prova. Papel desenvolvido por um professor ou coordenador responsável pelas avaliações das provas.* |
| *Aluno* | *Usuário responsável por atuar sobre a realização das provas no sistema. Papel incorporado por alunos devidamente matriculados na área para o perfil requerido.* |

# *Módulo Modelos*

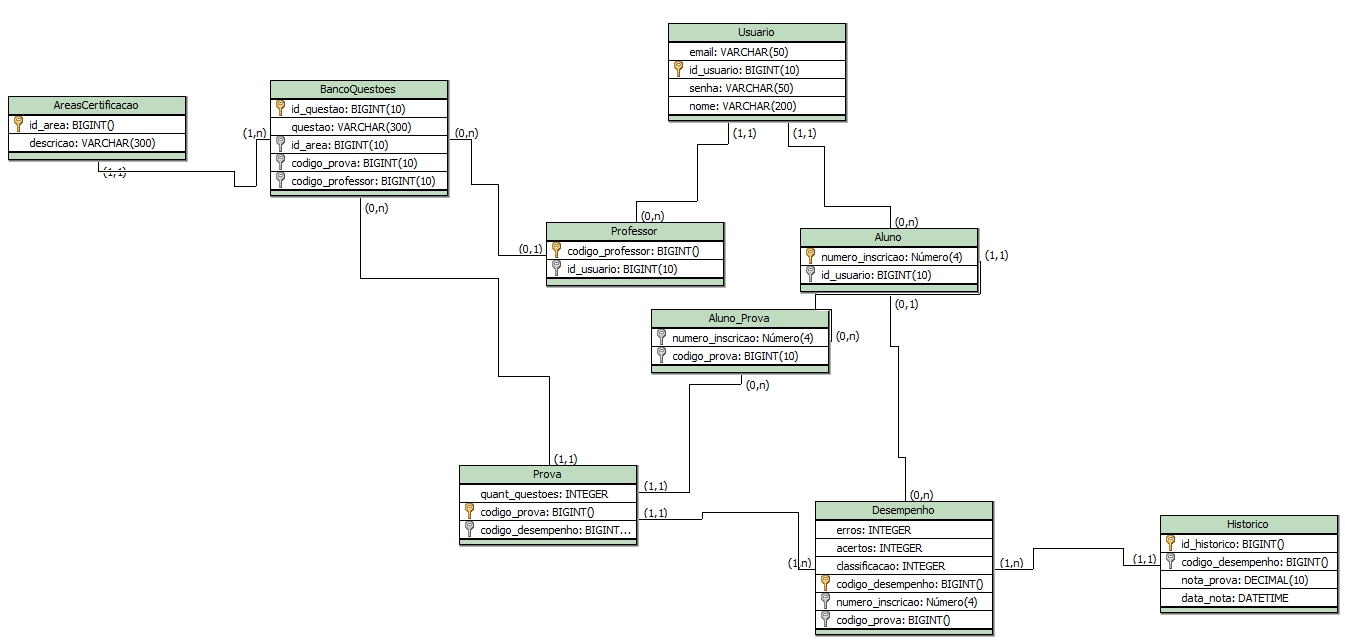
Nesta seção serão detalhados os modelos lógico e relacional do banco de dados. Além dos atributos das tabelas com seus respectivos dicionário de dados.

## Modelo Conceitual

**

*Figura 1- Modelo Conceitual para o Sistema e-Certificate.*

## Modelo Lógico



*Figura 2- Modelo Lógico para o Sistema e-Certificate.*

# Dicionário de Dados

*[TB 001 – Aluno]*

**Descrição:** A tabela Aluno representa o aluno e os seus dados no sistema.

**Pré-Condições:** A tabela Aluno herda os atributos da tabela Usuário.

**Ator:** Aluno

**Dicionário de Dados:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Nome da***  ***Tabela*** | ***Nome da***  ***Coluna*** | ***Tipo de Dados*** | ***Tamanho*** | ***Chave*** | ***Descrição*** |
| *Aluno* | *numero\_inscricao* | *BIGINT* | *DEFAULT* | ***PK*** | *Número da inscrição do aluno* |
| *Aluno* | *id\_usuario* | *BIGINT* | *DEFAULT* | ***FK***  ***(Usuario)*** | *Identificação do usuário* |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

*[TB 002 – Area]*

**Descrição:** A tabela Área representa uma área de certificação, como, segurança, desenvolvimento, análise.

**Pré-Condições:** None.

**Ator:** Aluno e Professor.

**Dicionário de Dados:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Nome da***  ***Tabela*** | ***Nome da***  ***Coluna*** | ***Tipo de Dados*** | ***Tamanho*** | ***Chave*** | ***Descrição*** |
| *Area* | *id\_area* | *BIGINT* | *DEFAULT* | ***PK*** | *Identificação da área* |
| *Area* | *descricao* | *VARCHAR* | *300* |  | *Nome da área* |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

*[TB 003 – Professor]*

**Descrição:** A tabela Professor representa o professor e os seus dados no sistema.

**Pré-Condições:** A tabela Professor herda os atributos da tabela Usuário.

**Ator:** Professor.

**Dicionário de Dados:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Nome da***  ***Tabela*** | ***Nome da***  ***Coluna*** | ***Tipo de Dados*** | ***Tamanho*** | ***Chave*** | ***Descrição*** |
| *Professor* | *codigo\_professor* | *BIGINT* | *DEFAULT* | ***PK*** | *Número da inscrição do professor* |
| *Professor* | *id\_usuario* | *BIGINT* | *DEFAULT* | ***FK***  ***(Usuario)*** | *Identificação do usuário* |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

*[TB 004 – Usuario]*

**Descrição:** A tabela Usuário representa os usuários e os seus dados no sistema.

**Pré-Condições:** None.

**Ator:** Administrador, Aluno e Professor.

**Dicionário de Dados:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Nome da***  ***Tabela*** | ***Nome da***  ***Coluna*** | ***Tipo de Dados*** | ***Tamanho*** | ***Chave*** | ***Descrição*** |
| *Usuario* | *id\_usuario* | *BIGINT* | *DEFAULT* | ***PK*** | *Número da inscrição do aluno* |
| *Usuario* | *senha* | *VARCHAR* | *50* |  | *Senha do usuário* |
| *Usuario* | *nome* | *VARCHAR* | *100* |  | *Nome do usuário* |
| *Usuario* | *email* | *VARCHAR* | *50* |  | *E-mail do usuário* |
| *Usuario* | *permissao* | *VARCHAR* | *20* |  | *Tipo de permissão ao sistema* |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

*[TB 005– Questão]*

**Descrição:** A tabela Questao representa as questões que os alunos e professores manipularão no sistema.

**Pré-Condições:** None.

**Ator:** Aluno e Professor.

**Dicionário de Dados:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Nome da***  ***Tabela*** | ***Nome da***  ***Coluna*** | ***Tipo de Dados*** | ***Tamanho*** | ***Chave*** | ***Descrição*** |
| *Questao* | *id\_questao* | *BIGINT* | *DEFAULT* | ***PK*** | *Identificação da Questão* |
| *Questao* | *descricao\_questao* | *VARCHAR* | *50* |  | *Descrição da Questao* |
| *Questao* | *grau\_dificuldade* | *VARCHAR* | *20* |  | *Grau de dificuldade da questão* |
| *Questao* | *alternativa\_1* | *VARCHAR* | *20* |  | *Primeira alternativa* |
| *Questao* | *alternativa\_2* | *VARCHAR* | *20* |  | *Segunda alternativa* |
| *Questao* | *alternativa\_3* | *VARCHAR* | *20* |  | *Terceira alternativa* |
| *Questao* | *alternativa\_4* | *VARCHAR* | *20* |  | *Quarta alternativa* |
| *Questao* | *alternativa\_correta* | *VARCHAR* | *10* |  | *Alternativa Correta* |
| *Questao* | *id\_area* | *BIGINT* | *DEFAULT* | ***FK***  ***(Area)*** | *Identificação da área* |
| *Questao* | *codigo\_professor* | *BIGINT* | *DEFAULT* | ***FK***  ***(Professor)*** | *Número da inscrição do professor* |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

*[TB 006 – Prova]*

**Descrição:** A tabela Prova representa a prova que o aluno responderá no sistema.

**Pré-Condições:** Possuir questões cadastradas no sistema.

**Ator:** Aluno.

**Dicionário de Dados:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Nome da***  ***Tabela*** | ***Nome da***  ***Coluna*** | ***Tipo de Dados*** | ***Tamanho*** | ***Chave*** | ***Descrição*** |
| *Prova* | *codigo\_prova* | *BIGINT* | *DEFAULT* | ***PK*** | *Identificação da prova* |
| *Prova* | *quantidade\_questoes* | *VARCHAR* | *10* |  | *Quantidade de Questões* |
| *Prova* | *data\_realizacao* | *TIMESTAMP* | *10* |  | *Data da Prova* |
| *Prova* | *numero\_inscricao* | *BIGINT* | *DEFAULT* | ***FK***  ***(Aluno)*** | *Número da inscrição do Aluno* |
| *Prova* | *codigo\_historico* | *BIGINT* | *DEFAULT* | ***FK***  ***(Historico)*** | *Identificação do histórico do Aluno* |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |
|  |  |  |  |  |  |  |

*[TB 007 – Questao\_Prova]*

**Descrição:** Esta tabela refere-se a uma tabela intermediária contendo apenas as chaves estrangeiras.

**Pré-Condições:** None.

**Ator:** Aluno.

**Dicionário de Dados:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Nome da***  ***Tabela*** | ***Nome da***  ***Coluna*** | ***Tipo de Dados*** | ***Tamanho*** | ***Chave*** | ***Descrição*** |
| *Questao\_Prova* | *id\_questao* | *BIGINT* | *DEFAULT* | ***FK***  ***(Questao)*** | *Identificação da Questão* |
| *Questao\_Prova* | *codigo\_prova* | *BIGINT* | *DEFAULT* | ***FK***  ***(Prova)*** | *Identificação da prova* |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

*[TB 008 – Historico]*

**Descrição:** Esta tabela refere-se ao armazenamento de todas os scores de um Aluno.

**Pré-Condições:** Ter realizado provas anteriores.

**Ator:** Aluno.

**Dicionário de Dados:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Nome da***  ***Tabela*** | ***Nome da***  ***Coluna*** | ***Tipo de Dados*** | ***Tamanho*** | ***Chave*** | ***Descrição*** |
| *Historico* | *codigo\_historico* | *BIGINT* | *DEFAULT* | ***PK*** | *Identificação da Questão* |
| *Historico* | *total\_acertos* | *VARCHAR* | *10* |  | *Score* |
| *Historico* | *desempenho* | *VARCHAR* | *10* |  | *Desempenho do Aluno* |
| *Historico* | *numero\_inscricao* | *BIGINT* | *DEFAULT* | ***FK***  ***(Aluno)*** |  |
| *Historico* | *erros* | *INTEGER* | *DEFAULT* |  | *Número de erros* |
| *Historico* | *acertos* | *INTEGER* | *DEFAULT* |  | *Número de acertos* |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

# Códigos

## DDL

-- Sequence para autoincrementar os ids de cada tabela

CREATE SEQUENCE id\_conhecimento

INCREMENT 1

MINVALUE 1

MAXVALUE 9999999999

START 1

CACHE 1;

-- Script para criação das tabelas

CREATE TABLE IF NOT EXISTS Aluno (

numero\_inscricao BIGINT DEFAULT NEXTVAL('id\_conhecimento'),

id\_usuario BIGINT,

CONSTRAINT pk\_ numero\_inscricao PRIMARY KEY (numero\_inscricao),

FOREIGN KEY(id\_usuario) REFERENCES Usuario (id\_usuario)

);

CREATE TABLE IF NOT EXISTS Area (

id\_area BIGINT DEFAULT NEXTVAL('id\_conhecimento'),

descricao VARCHAR(300)

CONSTRAINT pk\_ id\_area PRIMARY KEY (id\_area)

);

CREATE TABLE IF NOT EXISTS Questoes (

id\_questao BIGINT DEFAULT NEXTVAL('id\_conhecimento'),

descricao\_questao VARCHAR(50),

grau\_dificuldade VARCHAR(20),

alternativa\_1 VARCHAR(20),

alternativa\_2 VARCHAR(20),

alternativa\_3 VARCHAR(20),

alternativa\_4 VARCHAR(20),

alternativa\_correta VARCHAR(10),

id\_area BIGINT,

codigo\_professor BIGINT,

CONSTRAINT pk\_ id\_questao PRIMARY KEY (id\_questao),

FOREIGN KEY(id\_area) REFERENCES Area (id\_area),

FOREIGN KEY(codigo\_professor) REFERENCES Professor (codigo\_professor)

);

CREATE TABLE IF NOT EXISTS Professor (

codigo\_professor BIGINT DEFAULT NEXTVAL('id\_conhecimento'),

id\_usuario BIGINT,

CONSTRAINT pk\_ codigo\_professor PRIMARY KEY (codigo\_professor)

);

CREATE TABLE IF NOT EXISTS Questao\_Prova (

id\_questao BIGINT,

codigo\_prova VARCHAR(10),

FOREIGN KEY(id\_questao) REFERENCES Questoes (id\_questao)

);

CREATE TABLE IF NOT EXISTS Prova (

codigo\_prova BIGINT DEFAULT NEXTVAL('id\_conhecimento'),

quantidade\_questoes VARCHAR(10),

data\_realizacao TIMESTAMP,

numero\_inscricao BIGINT,

codigo\_historico BIGINT,

CONSTRAINT pk\_ codigo\_prova PRIMARY KEY (codigo\_prova),

FOREIGN KEY(numero\_inscricao) REFERENCES Aluno (numero\_inscricao),

FOREIGN KEY(codigo\_historico) REFERENCES Historico (codigo\_historico)

);

CREATE TABLE IF NOT EXISTS Usuario (

id\_usuario BIGINT DEFAULT NEXTVAL('id\_conhecimento'),

senha VARCHAR(50),

nome VARCHAR(100),

email VARCHAR(50),

permissao VARCHAR(20),

CONSTRAINT pk\_ id\_usuario PRIMARY KEY (id\_usuario)

);

CREATE TABLE IF NOT EXISTS Historico (

codigo\_historico BIGINT DEFAULT NEXTVAL('id\_conhecimento'),

total\_acertos VARCHAR(10),

desempenho VARCHAR(10),

numero\_inscricao BIGINT,

erros INTEGER,

acertos INTEGER,

CONSTRAINT pk\_ codigo\_historico PRIMARY KEY (codigo\_historico),

FOREIGN KEY(numero\_inscricao) REFERENCES Aluno (numero\_inscricao)

);

## DML

* Funções para inserir dados

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

-- Usuario

CREATE OR REPLACE FUNCTION inserir\_usuario(VARCHAR(50), VARCHAR(100), VARCHAR(50), VARCHAR(20)) RETURNS void AS $$

BEGIN

INSERT INTO Usuario(senha, nome, email, permissao)

VALUES (md5($1), $2, &3, $4);

END;

$$ LANGUAGE plpgsql;

--Inserir usuario

SELECT inserir\_usuario(‘12345’, ’Amanda’, `amanda.minueza@gmail.com`);

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

--Aluno

CREATE OR REPLACE FUNCTION inserir\_aluno(BIGINT) RETURNS void AS $$

BEGIN

INSERT INTO Aluno(id\_usuario)

VALUES ($1);

END;

$$ LANGUAGE plpgsql;

--Inserir Aluno

SELECT inserir\_aluno(1);

--Selecionar o Aluno

SELECT \* FROM USUARIO U INNER JOIN Aluno A ON U.id\_usuario = A.id\_usuario;

## Queries requeridas

--Selecionar todos os simulados realizados por um aluno

SELECT P.codigo\_prova, U.Nome, P.data\_realizacao, H.total\_acertos, H.desempenho FROM Prova P

INNER JOIN Historico H ON H.codigo\_historico = P.codigo\_historico

INNER JOIN Aluno A ON A.numero\_inscricao = P.numero\_inscricao

INNER JOIN Usuario U ON U.id\_usuario = A.id\_usuario

WHERE U.nome = 'Amanda';

-- Selecionar todas as Notas

SELECT total\_acertos FROM Historico;

--Selecionar o desempenho histórico de um Aluno

SELECT desempenho FROM Historico H

INNER JOIN Aluno A ON A.numero\_inscricao = H.numero\_inscricao

INNER JOIN Usuario U ON U.id\_usuario = A.id\_usuario

WHERE U.nome = 'Amanda';