

Sistema de aplicação de provas

Documentação

Versão 1.0

Carlos Henrique

Allan Tavares

Gabriel Henrique

Histórico de Revisões

| Data | Autor | Versão | Modificações Efetuadas |
| --- | --- | --- | --- |
|  |  | 1.0.0 | Versão Inicial |
| 21/09/2016 | Allan, Carlos e Gabriel | 1.1.0 | Revisão dos casos de usos |
| 05/10/2016 | Allan e Carlos | 1.2.0 | Revisão e ajustes gerais |

SUMÁRIO

[1. INTRODUÇÃO AO DOCUMENTO 1](#_Toc463556369)

[1.1. Propósito do Documento 1](#_Toc463556370)

[1.2. Tema 1](#_Toc463556371)

[1.3. Escopo do Produto 1](#_Toc463556372)

[1.4. Convenções, Termos e Abreviações 2](#_Toc463556373)

[1.4.1. Identificação dos Requisitos 2](#_Toc463556374)

[1.4.2. Prioridade dos Requisitos 2](#_Toc463556375)

[2. VISÃO GERAL DO PRODUTO 3](#_Toc463556376)

[2.1. Perspectiva do Produto 3](#_Toc463556377)

[2.2. Principais envolvidos no sistema e suas características 3](#_Toc463556378)

[2.3. Regras de Negócio 4](#_Toc463556379)

[3. REQUISITOS DO SISTEMA 4](#_Toc463556380)

[3.1. Requisitos Funcionais 4](#_Toc463556381)

[3.1.1. Desktop 4](#_Toc463556382)

[3.1.2. Web 5](#_Toc463556383)

[3.1.3. Mobile 6](#_Toc463556384)

[3.2. Requisitos Não Funcionais 7](#_Toc463556385)

[4. ANÁLISE E DESIGN 8](#_Toc463556386)

[4.1. Diagramas de Casos de Uso 8](#_Toc463556387)

[4.2. Narrativas de Casos de Uso 9](#_Toc463556388)

[4.2.1. Desktop 9](#_Toc463556389)

[4.2.2. Web 15](#_Toc463556390)

[4.2.3. Mobile 51](#_Toc463556391)

[4.3. Diagrama de Classes 55](#_Toc463556392)

[4.4. Modelo de Dados 56](#_Toc463556393)

[4.4.1. Modelo Lógico da base de dados 56](#_Toc463556394)

[4.4.2. Modelo Conceitual da base de dados 57](#_Toc463556395)

[4.4.3. Modelo Físico da base de dados 58](#_Toc463556396)

[4.5. Ambiente de desenvolvimento 63](#_Toc463556397)

[4.5.1. Desktop 63](#_Toc463556398)

[4.5.2. Web 63](#_Toc463556399)

[4.5.3. Mobile 63](#_Toc463556400)

[4.5.4. Armazenamento de Dados 63](#_Toc463556401)

[5. TESTES 63](#_Toc463556402)

[5.1. Caso de Teste 63](#_Toc463556403)

[5.2. Execução do Plano de Testes 64](#_Toc463556404)

[5.3. Relatório de incidentes de testes 64](#_Toc463556405)

[6. PROTÓTIPOS 64](#_Toc463556406)

[6.1. Desktop 64](#_Toc463556407)

[6.2. Web 64](#_Toc463556408)

[6.3. Mobile 64](#_Toc463556409)

[7. MANUAL DO USUÁRIO 64](#_Toc463556410)

[REFERÊNCIAS BIBLIOGRÁFICAS 64](#_Toc463556411)

[GLOSSÁRIO 64](#_Toc463556412)

# INTRODUÇÃO AO DOCUMENTO

## Propósito do Documento

Este documento tem como objetivo apresentar o sistema de aplicação e gerenciamento de provas e simulados, o Monkeys Tests.

## Tema

Sistema para aplicação e gerenciamento de provas e simulados em instituições de ensino.

## Escopo do Produto

O sistema é voltado para aplicação e gerenciamento de provas e simulados digitais para instituições de ensino.

O sistema auxilia os professores na hora da criação e facilita a aplicação de provas fornecendo diversas opções de personalização, tais como: tema, nível de dificuldade das questões, questões objetivas e dissertativas para melhor atender às metodologias específicas de cada professor e/ou instituição de ensino. É possível também a criação de simulados para aprimorar e preparar os alunos para futuras provas.

Tendo como base um banco de questões e respostas fornecidas pelos próprios professores, o sistema conta com métodos de alta facilidade para a correção de gabaritos e oferece métodos de segurança eficazes para evitar fraudes e aumentar a confiabilidade dos resultados obtidos nas provas aplicadas.

## Convenções, Termos e Abreviações

## Identificação dos Requisitos

• RNXX – Regra de Negócio, XX numeração do documento.

• RFXX – Requisito Funcional, XX numeração do documento.

• RNFXX – Requisito Não Funcional, XX numeração do documento.

• RFDXX – Requisito Funcional Desktop, XX numeração do documento.

• RFWXX – Requisito Funcional Web, XX numeração do documento.

• RFMXX – Requisito Funcional Mobile, XX numeração do documento.

• CSUDXX – Caso de Uso Desktop, XX numeração do documento.

• CSUWXX – Caso de Uso Web, XX numeração do documento.

• CSUMXX – Caso de Uso Mobile, XX numeração do documento.

## Prioridade dos Requisitos

• Essencial – O sistema não funcionará corretamente sem este requisito.

• Importante – O sistema terá recursos limitados sem este requisito.

• Desejável – Informação não crucial ao sistema, porém desejável.

# VISÃO GERAL DO PRODUTO

## Perspectiva do Produto

O sistema é direcionado para auxiliar na aplicação de provas, testes, simulados e atividades em instituições de ensino. Provas são uma tarefa em comum em todas as instituições de ensino, porém sua criação, aplicação e correção (na maioria dos casos) são feitas de forma manual, exigindo grande esforço do professor, além da falta de organização gerada pela grande quantidade de provas, que na maioria das vezes são impressas. Isso também pode causar perda de documentos e a falta de fiscalização, gerando grandes transtornos e altas taxas de fraudes.

O sistema foi desenvolvido para resolver estes problemas, fornecendo opções de personalização de provas, para se adaptar com as metodologias específicas de cada professor e/ou instituição de ensino, autenticação confiável por meio de login, senha, biometria e token, fácil acesso a consulta de resultados das provas e notas, entre diversas outras funcionalidades.

Conclui-se então que o sistema auxiliará em todas as etapas da prova. Antes (geração da prova), durante (aplicação da prova) e após (correção de gabaritos e consulta de notas), simplificando todo o processo.

## Principais envolvidos no sistema e suas características

O sistema foi desenvolvido para implementação em instituições de ensino, onde se encontram basicamente três tipos de usuário: os professores, os alunos e os administradores. Os professores serão responsáveis por aplicar, gerar e corrigir provas e simulados. Os alunos são os usuários mais comuns a utilizar o sistema, eles podem fazer consultas e realizam provas aplicadas pelos professores. Já os administradores são responsáveis por cadastrar funcionários, alunos, turmas e disciplinas no sistema, além de fazer o gerenciamento das questões aceitando ou negando questões cadastradas pelos professores.

## Regras de Negócio

• **RN01** – Os professores são responsáveis pela aplicação, criação e correção das provas e questões.

• **RN02** – Os diretores e coordenadores são responsáveis pela administração dos registros dos alunos, funcionários e ementas da instituição.

• **RN03** – Todas as informações das provas devem ser arquivadas, tais como: professor responsável, data de aplicação, turma em que foi aplicada e notas obtidas.

• **RN04** – Turmas são um conjunto de alunos que seguem o mesmo curso, frequentando a mesma sala.

• **RN05** – Grupos são um conjunto de alunos que seguem ou não o mesmo curso, podendo conter alunos de diversas turmas.

• **RN06** – Simulados são compostos apenas por questões objetivas.

• **RN07** – Os professores não podem considerar simulados como valor somativo.

• **RN08** – Os registros de alunos e funcionários devem ser mantidos mesmo não pertencendo mais à instituição.

# REQUISITOS DO SISTEMA

## Requisitos Funcionais

## Desktop

|  |  |  |  |
| --- | --- | --- | --- |
| Código | Descrição | Prioridade | Narrativa |
| RFD01 | O sistema deve permitir o login do aluno mediante ID de usuário e senha. | Essencial | CSUD01 |
| RFD02 | O sistema deve apresentar uma lista de provas relacionadas ao aluno logado | Essencial | CSUD02 |
| RFD03 | O sistema deve dar acesso ao aluno, por meio de login, biometria ou token, para que ele realize sua prova. | Essencial | CSUD03 |

## Web

|  |  |  |  |
| --- | --- | --- | --- |
| Código | Descrição | Prioridade | Narrativa |
| RFW01 | O sistema deve permitir o login de alunos e funcionários mediante ID de usuário e senha. | Essencial | CSUW01 |
| RFW02 | O sistema deve permitir somente ao administrador manter disciplinas, alunos, funcionários e turmas. | Essencial | CSUW02, CSUW03, CSUW04, CSUW05 |
| RFW03 | O sistema deve permitir ao professor a criação de questões objetivas e dissertativas. | Essencial | CSUW07, CSUW08 |
| RFW04 | O sistema deve permitir ao professor a criação de simulados. | Desejável | CSUW29 |
| RFW05 | O sistema deve dar ao aluno a opção de realizar simulados. | Desejável | CSUW31 |
| RFW06 | O sistema deve permitir somente ao administrador checar, aprovar ou reprovar as questões cadastradas pelos professores. | Importante | CSUW14 |
| RFW07 | O sistema deve permitir ao professor a criação de provas personalizadas ou randômicas. | Essencial | CSUW15 |
| RFW08 | O sistema deve permitir ao professor o controle de prova. | Importante | CSUW17 |
| RFW09 | O sistema deve permitir a geração de relatórios sobre o desempenho de provas especificas. | Desejável | CSUW24 |
| RFW10 | O sistema deve permitir ao professor a correção das provas realizadas. | Essencial | CSUW23 |
| RFW11 | O sistema deve permitir ao professor a criação e gerenciamento de grupos. | Desejável | CSUW06, CSUW25 |
| RFW12 | O sistema deve permitir ao administrador o gerenciamento de turmas. | Desejável | CSUW26 |

## 3.1.3. Mobile

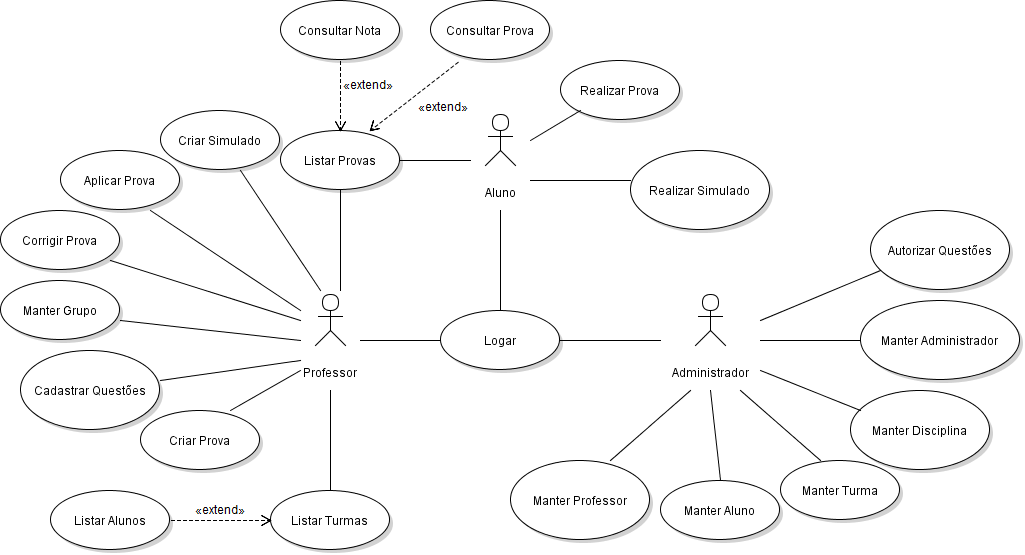
|  |  |  |  |
| --- | --- | --- | --- |
| Código | Descrição | Prioridade | Narrativa |
| RFM01 | O sistema deve permitir o login do aluno mediante ID de usuário e senha. | Essencial | CSUM01 |
| RFM02 | O sistema deve permitir ao aluno a consulta de informações e resultados obtidos em suas provas. | Desejável | CSUM02 |
| RFM03 | O sistema deve permitir a criação de eventos na agenda para avisar o aluno sobre provas pendentes. | Desejável | CSUM03 |
| RFM04 | O sistema deve permitir ao aluno a consulta de token para sua autenticação | Importante | CSUM04 |

## 3.2. Requisitos Não Funcionais

|  |  |  |
| --- | --- | --- |
|  | Código | Descrição |
| **U S A B I L I D A D E** | RNF01 | O sistema deve ter uma interface simples, clara e objetiva com navegação intuitiva. |
| RNF02 | O sistema deve notificar o aluno a respeito de uma prova relacionada a ele por e-mail ou notificação em dispositivo mobile. |
| RNF03 | O sistema deve corrigir automaticamente as questões objetivas das provas realizadas. |
| RNF04 | O sistema deve gerar relatórios sobre as provas realizadas. |
| RNF05 | O sistema deve “congelar” as questões após serem utilizadas quatro vezes. |
| RNF06 | O sistema deve disponibilizar apenas questões “congeladas” para a criação de simulados. |
| **S E G U R A N Ç A & I N T E G R I D A D E** | RNF07 | O sistema deve armazenar o resultado obtido pelos alunos após o termino das provas. |
| RNF08 | O sistema deve bloquear o acesso a outras janelas durante a aplicação de provas sem consulta. |
| RNF09 | O sistema deve criptografar as notas e senhas dos alunos. |
| RNF10 | O sistema deve garantir as permissões de acesso de acordo com o tipo de usuário. |
| RNF11 | O sistema deve gerar token para provas. |
| RNF12 | O sistema não deve permitir a exclusão total de nenhum registro. Os dados julgados desnecessários serão desativados, mas mantidos no banco. |

# ANÁLISE E DESIGN

## Diagramas de Casos de Uso

****

## Narrativas de Casos de Uso

## 4.2.1. Desktop

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUD01 / Login  Objetivo: Logar no sistema.  Ator Principal: Aluno.  Entradas: ID de usuário e senha. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator acessa o sistema.  2. O sistema mostra os campos de login e senha.  3. O ator digita seu ID e senha nos campos.  4. O ator clica em logar.  5. O sistema valida os campos preenchidos. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| 5.1. Caso o ID ou senha sejam inválidos o sistema deve avisar o autor e retornar ao passo 2. |
| PRÉ-CONDIÇÕES |
| 1. O ator deve estar cadastrado no sistema. |
| PÓS-CONDIÇÕES |
| 1. O ator é logado no sistema e é levado ao CSUD02 (Selecionar prova). |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUD02 / Selecionar prova  Objetivo: Selecionar uma prova para ser realizada.  Ator Principal: Aluno.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia após o CSUD01 (Login).  2. O sistema dispõe uma lista de provas agendadas e relacionadas ao ator.  3. O ator seleciona uma prova e seleciona a opção “Realizar”. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| 3.1. Caso a prova não esteja liberada, o sistema avisa o ator e o impede de prosseguir.  3.2. Caso a prova seja anulada, o sistema avisa o ator e o impede de prosseguir. |
| PRÉ-CONDIÇÕES |
| 1. O ator deve ter ao menos uma prova agendada. |
| PÓS-CONDIÇÕES |
| 1. O ator é encaminhado ao CSUD03 (Liberar prova). |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUD03 / Autenticar aluno  Objetivo: Autenticar aluno para realização.  Ator Principal: Aluno e Professor.  Entradas: Token. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia após o CSUD02 (Selecionar prova).  2. O Aluno é levado para uma tela de espera onde deve aguardar a liberação da prova pelo Professor. |
| CENÁRIO ALTERNATIVO |
| 2.1. Token  2.1.1. Caso a prova necessite de autenticação por token o Aluno é direcionado para uma tela de autenticação.  2.1.2. O sistema apresenta ao aluno um campo para a digitação do token.  2.1.3. O Professor deve liberar o token via aplicação web.  2.1.4. O Aluno pode acessar o token via aplicação ou dispositivo mobile.  2.1.5. O Aluno digita o token no campo e seleciona a opção “Autenticar”.  2.2. Biometria  2.2.1. Caso a prova necessite de autenticação biométrica o Aluno é direcionado para uma tela de espera.  2.2.2. O aluno deve se dirigir ao professor para a autenticação. |
| CENÁRIO DE EXCEÇÃO |
| 2.1. Em caso de erro de digitação ou caso o sistema não consiga validar os dados, é apresentado ao Aluno uma mensagem para tentar novamente. |

|  |
| --- |
| PRÉ-CONDIÇÕES |
| 1. Token.  1.1. O Aluno deve ter acesso ao dispositivo mobile ou a aplicação web.  2. Biometria.  2.1. O Aluno deve ter sido previamente cadastrado no sistema biométrico. |
| PÓS-CONDIÇÕES |
| Não há |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUD04 / Realização da Prova  Objetivo: Realizar a prova.  Ator Principal: Aluno.  Entradas: Token. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia após o CSUD03 (Autenticar aluno).  2. Após o CSUW20 (Liberar prova) ser executada pelo professor, o ator seleciona a opção “Começar Prova”.  3. O sistema inicia a prova.  4. Execute CSUD05 (Procedimentos da prova).  5. Ao finalizar a prova, o sistema retorna ao passo CSUD02 (Selecionar prova). |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| 1. A prova deve estar liberada pelo professor. |
| PÓS-CONDIÇÕES |
| Não há. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUD05 / Procedimentos da prova  Objetivo: Realizar a prova.  Ator Principal: Aluno.  Entradas: N/A |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia junto à prova.  2. Respondendo questões.  2.1. Questões objetivas.  2.1.1. O sistema mostra uma questão e suas alternativas.  2.1.2. O ator seleciona uma das alternativas e clica em próximo ou seleciona outra questão.  2.1.3. O sistema retorna ao passo 2.1.1.  2.2. Questões dissertativas.  2.2.1. O sistema mostra uma questão e um campo para a digitação da resposta.  2.2.2. O aluno digita sua resposta e clica em próximo ou seleciona outra questão.  2.2.3. O sistema retorna ao passo 2.2.1.  3. Ao termino da prova o sistema mostra a opção de revisar a prova ou finaliza-la. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| 1.1. Caso o tempo da prova acabe durante sua realização o sistema deve finalizar a prova. |
| PRÉ-CONDIÇÕES |
| 1. A prova deve ser liberada pelo professor. |
| PÓS-CONDIÇÕES |
| Não há. |

## 4.2.2. Web

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW01 / Login  Objetivo: Logar no sistema.  Ator Principal: Administrador, Professor e Aluno.  Entradas: ID de usuário e senha. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o autor acessa o sistema. 2. O sistema mostra os campos de login e senha. 3. O ator digita seu ID e senha nos campos.   4. O ator seleciona a opção logar.  5. O sistema valida os campos preenchidos. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| 5.1. Caso o ID ou senha sejam inválidos o sistema deve avisar o autor e retornar ao passo 2. |
| PRÉ-CONDIÇÕES |
| 1. O ator deve estar cadastrado no sistema. |
| PÓS-CONDIÇÕES |
| 1. O ator é logado no sistema e ganha acesso a outras opções. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW02 / Cadastro de alunos  Objetivo: Cadastrar um aluno no sistema.  Ator Principal: Administrador.  Entradas: Dados do aluno. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu de Cadastros e seleciona a opção “Cadastrar Aluno”.  3. O sistema dispõe os campos para o preenchimento dos dados do aluno que será cadastrado. Tais como: Nome completo, sexo, CPF, data de nascimento, e-mail, telefone, celular, login, senha e turma.  4. O sistema valida os campos e o aluno é cadastrado. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| 4.1. Caso o CPF, e-mail e/ou login já estejam cadastrados, o sistema avisa o ator e o impede de seguir. |
| PRÉ-CONDIÇÕES |
| 1. O usuário logado precisa ser um Administrador. |
| PÓS-CONDIÇÕES |
| 1. Aluno cadastrado no sistema. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW03 / Cadastro de funcionários  Objetivo: Cadastrar um funcionário no sistema.  Ator Principal: Administrador.  Entradas: Dados do funcionário. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu de Cadastros e seleciona a opção “Cadastrar Funcionário”.  3. O sistema dispõe os campos para o preenchimento dos dados do funcionário que será cadastrado. Tais como: Nome completo, sexo, CPF, data de nascimento, e-mail, telefone, celular, login e senha.  4. O ator tem a opção de escolher o Tipo do Funcionário: Professor ou Administrador.  5. O sistema valida os campos e o funcionário é cadastrado. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| 3.1. Caso o CPF, e-mail e/ou login já estejam cadastrados, o sistema avisa o ator e o impede de seguir. |
| PRÉ-CONDIÇÕES |
| 1. O usuário logado precisa ser um Administrador. |
| PÓS-CONDIÇÕES |
| 1. Funcionário cadastrado no sistema. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW04 / Cadastro de disciplinas  Objetivo: Cadastrar uma disciplina no sistema.  Ator Principal: Administrador.  Entradas: Descrição da disciplina. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu de Cadastros e seleciona a opção “Cadastrar Disciplina”.  3. O sistema dispõe o campo “Descrição” para preenchimento.  4. O sistema valida o campo e a disciplina é cadastrada. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| 4.1. Caso a descrição da disciplina já esteja cadastrado, o sistema avisa o ator e o impede de seguir. |
| PRÉ-CONDIÇÕES |
| 1. O usuário logado precisa ser um Administrador. |
| PÓS-CONDIÇÕES |
| 1. Disciplina cadastrada no sistema. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW05 / Cadastro de turmas  Objetivo: Cadastrar uma turma no sistema.  Ator Principal: Administrador.  Entradas: Descrição da turma e período. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu de Cadastros e seleciona a opção “Cadastrar Turma”.  3. O sistema dispõe o campo “Descrição” para preenchimento e a opção de período da turma: Manhã, Tarde e Noite.  4. O sistema valida os campos e a turma é cadastrada. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| 1.1. Caso a descrição da turma já esteja cadastrado, o sistema avisa o ator e o impede de seguir. |
| PRÉ-CONDIÇÕES |
| 1. O usuário logado precisa ser um Administrador. |
| PÓS-CONDIÇÕES |
| 1. Turma cadastrada no sistema. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW06 / Cadastro de grupos  Objetivo: Cadastrar um grupo no sistema.  Ator Principal: Professor.  Entradas: Descrição do grupo. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu de Cadastros e seleciona a opção “Cadastrar Grupo”.  3. O sistema dispõe o campo “Descrição” para preenchimento.  4. O sistema valida os campos e o grupo é cadastrado. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| 2.1. Caso a descrição do grupo já esteja cadastrada, o sistema avisa o ator e o impede de seguir. |
| PRÉ-CONDIÇÕES |
| 1. O usuário logado precisa ser um Professor. |
| PÓS-CONDIÇÕES |
| 1. Grupo cadastrado no sistema. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW07 / Cadastro de questões objetivas  Objetivo: Cadastrar uma questão objetiva no sistema.  Ator Principal: Professor.  Entradas: Dados da questão. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu de Cadastros e seleciona a opção “Cadastrar Questão Objetiva”.  3. O sistema dispõe os campos para preenchimento dos dados da questão objetiva como: Enunciado, palavra-chave, dificuldade, disciplina e alternativas.  4. O sistema valida os campos e a questão é direcionada para o Administrador. |
| CENÁRIO ALTERNATIVO |
| 3.1. O ator tem a opção de associar uma mídia (Foto e texto) para a questão. |
| CENÁRIO DE EXCEÇÃO |
| 3.1. É necessário informar no mínimo três alternativas. |
| PRÉ-CONDIÇÕES |
| 1. Ter no mínimo uma disciplina cadastrada no sistema. |
| PÓS-CONDIÇÕES |
| 1. Requisição enviada para o Administrador para aprovação ou reprovação da questão. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW08 / Cadastro de questões dissertativas  Objetivo: Cadastrar uma questão dissertativa no sistema.  Ator Principal: Professor.  Entradas: Dados da questão. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu de Cadastros e seleciona a opção “Cadastrar Questão Dissertativa”.  3. O sistema dispõe os campos para preenchimento dos dados da questão dissertativa como: Enunciado, palavra-chave, dificuldade, disciplina e resposta base.  4. O sistema valida os campos e a questão é direcionada para o Administrador. |
| CENÁRIO ALTERNATIVO |
| 3.1. O ator tem a opção de associar uma mídia (Foto e texto) para a questão. |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| 1. Ter no mínimo uma disciplina cadastrada no sistema. |
| PÓS-CONDIÇÕES |
| 1. Requisição enviada para o Administrador para aprovação ou reprovação da questão. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW09 / Listagem de alunos  Objetivo: Listar os alunos cadastrados no sistema.  Ator Principal: Professor e Administrador.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu de Listas e seleciona a opção “Listar Alunos”.  3. O sistema dispõe uma lista com todos os alunos cadastrados no sistema.  4. O sistema dispõe meios de filtragem para melhor orientação do ator, como: filtro de turma, períodos ou ordem alfabética. |
| CENÁRIO ALTERNATIVO |
| 1.1. Caso o ator seja um Administrador, a lista terá a opção de edição dos registros. |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| Não há. |
| PÓS-CONDIÇÕES |
| Não há. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW010 / Listagem de funcionários  Objetivo: Listar os funcionários cadastrados no sistema.  Ator Principal: Administrador.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu de Listas e seleciona a opção “Listar Funcionários”.  3. O sistema dispõe uma lista com todos os funcionários cadastrados no sistema.  4. O sistema dispõe meios de filtragem para melhor orientação do ator, como: filtro de cargo e ordem alfabética. |
| CENÁRIO ALTERNATIVO |
| 4.1. O ator tem a opção de edição dos registros. |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| Não há. |
| PÓS-CONDIÇÕES |
| Não há. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW11 / Listagem de disciplinas  Objetivo: Listar as disciplinas cadastradas no sistema.  Ator Principal: Professor e Administrador.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu de Listas e seleciona a opção “Listar Disciplinas”.  3. O sistema dispõe uma lista com todos as disciplinas cadastradas no sistema. |
| CENÁRIO ALTERNATIVO |
| 3.1. Caso o ator seja um Administrador, a lista terá a opção de edição dos registros. |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| Não há. |
| PÓS-CONDIÇÕES |
| Não há. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW12 / Listagem de turmas  Objetivo: Listar as turmas cadastradas no sistema.  Ator Principal: Professor e Administrador.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu de Listas e seleciona a opção “Listar Turmas”.  3. O sistema dispõe uma lista com todos as turmas cadastradas no sistema. |
| CENÁRIO ALTERNATIVO |
| 3.1. Caso o ator seja um Administrador, a lista terá a opção de edição dos registros. |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| Não há. |
| PÓS-CONDIÇÕES |
| Não há. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW13 / Listagem de grupos  Objetivo: Listar os grupos cadastrados no sistema.  Ator Principal: Professor e Administrador.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu de Listas e seleciona a opção “Listar Grupos”.  3. O sistema dispõe uma lista com todos os grupos cadastrados no sistema. |
| CENÁRIO ALTERNATIVO |
| 1. A lista terá a opção de edição dos registros.  2. O sistema disponibiliza uma opção para gerenciamento de grupos. Execute CSUW25 (Gerenciar grupos). |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| Não há. |
| PÓS-CONDIÇÕES |
| Não há. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW14 / Listagem de questões  Objetivo: Listar as questões cadastradas no sistema.  Ator Principal: Professor e Administrador.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu de Listas e seleciona a opção “Listar Questões”.  3. O sistema dispõe uma lista com todos as questões cadastradas no sistema. |
| CENÁRIO ALTERNATIVO |
| 3.1. Os atores têm a opção de visualizar as mídias associadas as questões.  3.2. Administrador  3.2.1. O administrador terá a opção de aprovar ou recusar questões enviadas pelos Professores.  3.3. Professor  3.3.1. O professor terá a opção de alteração e reenvio das questões. |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| 1. Ao menos uma questão precisa estar cadastrada. |
| PÓS-CONDIÇÕES |
| 3.2.1.1. Caso a questão seja aprovada a mesma é inclusa no sistema.  3.2.1.2. Caso a questão seja reprovada o status é alterado para “Recusada” e o professor responsável terá a opção de edição e reenvio da mesma. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW15 / Criação de provas  Objetivo: Criar provas.  Ator Principal: Professor.  Entradas: Dados da prova. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu e seleciona a opção “Gerar Prova”.  3. O sistema dispõe uma tela com campos para entrada de dados iniciais da prova. Tais como:  - Título;  - Data da realização;  - Tempo total da prova (em minutos);  - Valor total da prova (de 5 a 100);  - Opção de ativar autenticação ou não;  - Opção de consulta ou não;  - Tipo de criação da prova (manual ou randômica);  - Turma ou grupo em que será aplicada a prova;  - Tipo de realização (digital ou impressa).  4. O sistema valida os campos;  5. O sistema direciona o ator para o próximo passo de acordo com o Tipo de criação de prova (manual ou randômica);  6. O sistema dispõe uma lista com todas as questões selecionadas.  7. O ator tem a opção de visualizar detalhes das questões e remoção das mesmas.  8. Ao término do processo, o ator tem a opção de finalizar a criação da prova ou retroceder ao passo anterior. |
| CENÁRIO ALTERNATIVO |
| 5.1. Prova randômica.  5.1.1. O sistema dispõe uma tela com opções para a criação da prova randomicamente.  5.1.2. O ator seleciona uma Disciplina pré cadastrada.  5.1.3. O ator seleciona as quantidades de questões objetivas e/ou dissertativas de acordo com o seu nível de dificuldade.  5.1.4. O ator pressiona a opção “Adicionar Questões”.  5.1.5. O sistema valida as informações.  5.1.6. O ator retorna ao passo 5.1.1.  5.1.7. Ao término do processo, o ator tem a opção de avançar para o próximo passo ou retroceder ao passo anterior.  5.2. Prova manual.  5.2.1. O sistema dispõe uma tela com opções para a criação da prova manualmente.  5.2.2. O sistema dispõe opções de filtragem de questões. Tais como:  - Disciplina;  - Dificuldade;  - Professor;  - Tipo de questão (objetivas ou dissertativas);  - Enunciado;  - Palavra-chave;  5.2.3. O ator seleciona uma questão e pressiona a opção “Adicionar” para inclui-la na lista de questões selecionadas.  5.2.4. Ao término do processo, o ator tem a opção de avançar para o próximo passo ou retroceder ao passo anterior. |
| CENÁRIO DE EXCEÇÃO |
| 1. O sistema valida a quantidade de questões selecionadas e caso a quantidade de questões seja inferior a cinco ou superior a duzentas, o sistema avisa o ator e o impede de seguir com o processo. |
| PRÉ-CONDIÇÕES |
| 1. As questões selecionadas devem ter a quantidade de no mínimo cinco e no máximo duzentas. |
| PÓS-CONDIÇÕES |
| 1. Os alunos são notificados sobre a data de realização da prova. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW16 / Listagem de provas  Objetivo: Listar as provas cadastradas no sistema.  Ator Principal: Professor.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu de Listas e seleciona a opção “Listar Provas”.  3. O sistema dispõe uma lista com todas as provas relacionadas com o ator. |
| CENÁRIO ALTERNATIVO |
| 3.1. O ator tem a opção de acessar o Painel de Controle de cada prova. Execute CSUW17 (Painel de controle da prova). |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| 1. Ao menos uma prova relacionada ao ator precisa estar cadastrada. |
| PÓS-CONDIÇÕES |
| Não há. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW17 / Painel de controle da prova  Objetivo: Autenticar, liberar, corrigir e anular provas.  Ator Principal: Professor.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW16 (Listagem de provas).  2. O ator seleciona uma prova e seleciona a opção “Painel de Controle da prova”  3. O sistema dispõe informações da prova e uma lista de todos os alunos que realizarão ou realizaram a mesma. |
| CENÁRIO ALTERNATIVO |
| 3.1. Prova agendada.  3.1.1. O sistema dispõe opções para o controle da prova antes da realização. Tais como:  - Reagendar prova; (CSUW18);  - Autenticar alunos (CSUW19);  - Liberar provas (CSUW20);  - Cancelar prova (CSUW21).  3.2. Prova realizada.  3.2.1. O sistema dispõe opções para a finalização da prova. Tais como:  - Corrigir prova (CSUW22);  - Anular prova (CSUW23);  - Gerar relatório (CSUW24). |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| 1. Ao menos uma prova relacionada ao ator precisa estar cadastrada. |

|  |
| --- |
| PÓS-CONDIÇÕES |
| 1. Status da prova alterado. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW18 / Reagendar prova  Objetivo: Alterar data de realização das provas no sistema.  Ator Principal: Professor.  Entradas: Nova data de realização da prova. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW17 (Painel de controle da prova).  2. O ator seleciona a opção “Reagendar Prova”.  3. O sistema dispõe um campo para a nova data.  4. O ator insere a nova data e confirma a operação. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| 1. O sistema não aceita datas anteriores a data atual. |
| PRÉ-CONDIÇÕES |
| 1. A prova precisar estar com status de agendada. |
| PÓS-CONDIÇÕES |
| 1. Data de realização modificada e notifica os alunos relacionados à prova. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW19 / Autenticar aluno  Objetivo: Confirmar identidade do aluno.  Ator Principal: Professor e Aluno.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW17 (Painel de controle da prova).  2. O Professor seleciona a opção “Autenticar” encontrada na lista de alunos relacionados à aquela prova.  3. O sistema apresenta duas opções de autenticação:  3.1. Token.  3.1.1. O sistema gera um token de seis dígitos que é enviado ao aluno.  3.1.2. O Aluno tem acesso a este token via aplicativo web ou dispositivo mobile.  3.1.3. Ao digitar o token na aplicação desktop a prova é liberada ao aluno.  3.2. Biometria.  3.2.1. O sistema apresenta ao Professor uma tela para identificação das digitais do aluno.  3.2.2. O sistema valida as informações via dispositivo biométrico e libera ou não a prova ao aluno; |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| 3.2.1. O aluno deve ter sua biometria previamente cadastrado. |
| PÓS-CONDIÇÕES |
| 1. Aluno autenticado para a prova. |
| Narrativa de caso de uso |
| Sigla/nome: CSUW20 / Liberar prova  Objetivo: Libera prova para ser realizada.  Ator Principal: Professor.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW17 (Painel de controle da prova).  2. O ator seleciona a opção “Liberar Prova”.  3. O sistema pede a confirmação do ator e libera a prova. |
| CENÁRIO ALTERNATIVO |
| 1. Caso a prova possua autenticação, execute CSUW19 (Autenticar aluno). |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| 1. Caso a prova possua autenticação, ao menos um aluno precisa estar autenticado. |
| PÓS-CONDIÇÕES |
| 1. Prova liberada ao aluno para realização. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW21 / Cancelar prova  Objetivo: Cancelar prova.  Ator Principal: Professor.  Entradas: Motivo de cancelamento. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW17 (Painel de controle da prova).  2. O ator seleciona a opção “Cancelar Prova”.  3. O sistema dispõe um campo para o motivo do cancelamento.  4. O ator insere os dados requeridos ou não e seleciona a opção de prosseguir.  5. O sistema pede a confirmação do ator e cancela a prova. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| 1. A prova precisar estar com status de agendada. |
| PÓS-CONDIÇÕES |
| 1. Prova cancelada e o sistema notifica os alunos relacionados à prova. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW22 / Anular prova  Objetivo: Anular prova de um aluno especifico.  Ator Principal: Professor.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW17 (Painel de controle da prova).  2. O ator seleciona a opção “Anular Prova” encontrada na lista de alunos relacionados à aquela prova.  3. O sistema dispõe um campo para o motivo da anulação.  4. O ator insere os dados requeridos ou não e seleciona a opção de prosseguir.  5. O sistema pede a confirmação do ator e anula a prova. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| Não há. |
| PÓS-CONDIÇÕES |
| 1. Prova do aluno é anulada. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW23 / Corrigir provas  Objetivo: Corrigir as provas cadastradas no sistema.  Ator Principal: Professor.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW17 (Painel de controle da prova).  2. O ator seleciona a opção “Corrigir Provas”.  3. O sistema dispõe ao ator as questões dissertativas dos alunos uma a uma para serem corrigidas.  4. O ator preenche o campo de consideração relacionada a resposta do aluno e seleciona entra três opções: Certo, meio-certo, incorreto.  5. O ator pressiona a opção de prosseguir e retorna ao passo 3.  6. Ao término do processo o sistema avisa o ator que não há mais provas a serem corrigidas e direciona o ator para o painel de controle de prova. |
| CENÁRIO ALTERNATIVO |
| 1.1. O ator seleciona a opção “Corrigir prova” de determinado aluno localizado ao lado do registro de alunos que realizaram a prova em questão.  1.1.1. O sistema dispõe as questões dissertativas daquele aluno para serem corrigidas.  5.1. O ator tem a opção de suspender a correção das provas e dar continuidade em outra ocasião. |
| CENÁRIO DE EXCEÇÃO |
| 1. Para provas que não possuem questões dissertativas a serem corrigidas, o sistema avisa o ator. |
| PRÉ-CONDIÇÕES |
| 1. Ao menos uma prova relacionada ao ator precisa ter sido realizada. |

|  |
| --- |
| PÓS-CONDIÇÕES |
| 1. Status da prova alterado e notas disponibilizadas aos alunos. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW24 / Gerar relatório  Objetivo: Gerar relatório de provas realizadas.  Ator Principal: Professor.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW17 (Painel de controle da prova).  2. O ator seleciona a opção “Gerar Relatório”.  3. O sistema apresenta as informações da prova e estatística das questões da mesma. |
| CENÁRIO ALTERNATIVO |
| 3.1. O ator tem a opção de exportar ou imprimir o relatório. |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| 1. A prova precisa estar finalizada. |
| PÓS-CONDIÇÕES |
| Não há. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW25 / Gerenciar grupos  Objetivo: Adicionar ou remover membros dos grupos no sistema.  Ator Principal: Professor.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu de Gerenciamento e seleciona a opção “Gerenciar Grupo”.  3. O sistema dispõe duas listas. Uma contendo os alunos e outro o grupo para qual o aluno selecionado será incluído.  4. O sistema dispõe filtro por turma para melhor localização dos alunos.  5. O sistema valida as opções. |
| CENÁRIO ALTERNATIVO |
| 3.1. O ator tem a opção de remover um ou mais membros do grupo selecionado. |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| 1. O usuário logado precisa ser um Professor.  2. Ao menos um grupo deve estar cadastrado no sistema. |
| PÓS-CONDIÇÕES |
| 1. Alteração de membros dos grupos no sistema. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW26 / Gerenciar turmas  Objetivo: Realocação de alunos nas turmas.  Ator Principal: Administrador.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu de Gerenciamento e seleciona a opção “Gerenciar Turmas”.  3. O sistema dispõe duas listas. Uma contendo os alunos e outro a turma para qual o aluno selecionado será transferido.  4. Após a seleção dos alunos, o ator seleciona a opção “Transferir” para transferir os alunos para a turma alvo. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| 1. O usuário logado precisa ser um Administrador. |
| PÓS-CONDIÇÕES |
| 1. Aluno transferido para a turma alvo. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW27 / Consultar provas  Objetivo: Consultar informações da prova.  Ator Principal: Aluno.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator seleciona a opção “Minhas Provas”.  3. O sistema apresenta uma lista de provas relacionadas ao ator com informações básicas sobre a mesma. |
| CENÁRIO ALTERNATIVO |
| 3.1. O ator tem a opção de acessar mais informações da prova.  3.2. O ator tem a opção de visualizar a correção de sua prova. |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| 1. O ator precisar ter ao menos uma prova relacionada a ele. |
| PÓS-CONDIÇÕES |
| Não há. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW28 / Consultar token  Objetivo: Acesso ao token.  Ator Principal: Aluno.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. Execute CSUW27 (Consultar provas).  2. O ator seleciona a opção “Mais informações”.  3. O ator é encaminhado a tela “Informações da Prova”.  4. O token é exibido em um campo especifico. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| 4.1. O token não será exibido caso o professor não o tenha gerado. |
| PRÉ-CONDIÇÕES |
| 1. O ator precisa ter ao menos uma prova agendada. |
| PÓS-CONDIÇÕES |
| Não há. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW29 / Criação de simulados  Objetivo: Criar simulados.  Ator Principal: Professor.  Entradas: Dados do simulado. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu e seleciona a opção “Gerar Simulado”.  3. O sistema dispõe uma tela com campos para entrada de dados iniciais do simulado. Tais como:  - Título;  - Tempo total do simulado (em minutos);  - Tipo de criação do simulado (manual ou randômica);  4. O sistema valida os campos;  5. O sistema direciona o ator para o próximo passo de acordo com o Tipo de criação de simulado (manual ou randômica);  6. O sistema dispõe uma lista com todas as questões selecionadas.  7. O ator tem a opção de visualizar detalhes das questões e remoção das mesmas.  8. Ao término do processo, o ator tem a opção de finalizar a criação do simulado ou retroceder ao passo anterior. |
| CENÁRIO ALTERNATIVO |
| 5.1. Simulado randômico.  5.1.1. O sistema dispõe uma tela com opções para a criação do simulado randomicamente.  5.1.2. O ator seleciona uma Disciplina pré cadastrada.  5.1.3. O ator seleciona as quantidades de questões objetivas e/ou dissertativas de acordo com o seu nível de dificuldade.  5.1.4. O ator pressiona a opção “Adicionar Questões”.  5.1.5. O sistema valida as informações.  5.1.6. O ator retorna ao passo 5.1.1.  5.1.7. Ao término do processo, o ator tem a opção de avançar para o próximo passo ou retroceder ao passo anterior.  5.2. Prova manual.  5.2.1. O sistema dispõe uma tela com opções para a criação da prova manualmente.  5.2.2. O sistema dispõe opções de filtragem de questões. Tais como:  - Disciplina;  - Dificuldade;  - Professor;  - Tipo de questão (objetivas ou dissertativas);  - Enunciado;  - Palavra-chave;  5.2.3. O ator seleciona uma questão e pressiona a opção “Adicionar” para inclui-la na lista de questões selecionadas.  5.2.4. Ao término do processo, o ator tem a opção de avançar para o próximo passo ou retroceder ao passo anterior. |
| CENÁRIO DE EXCEÇÃO |
| 1. O sistema valida a quantidade de questões selecionadas e caso a quantidade de questões seja inferior a cinco ou superior a duzentas, o sistema avisa o ator e o impede de seguir com o processo. |
| PRÉ-CONDIÇÕES |
| 1. As questões selecionadas devem ter a quantidade de no mínimo cinco e no máximo duzentas.  2. Apenas questões “congeladas” podem ser utilizadas. |
| PÓS-CONDIÇÕES |
| Não há. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW30 / Listagem de simulados  Objetivo: Listar os simulados disponíveis.  Ator Principal: Professor e Aluno.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUW01 (Login).  2. O ator acessa o menu de Listas e seleciona a opção “Listar Simulados”.  3. O sistema dispõe uma lista com todos os simulados relacionados aos atores. |
| CENÁRIO ALTERNATIVO |
| 3.1. O Professor tem a opção de excluir o simulado ou acessar as questões do mesmo.  3.2. O Aluno tem a opção de realizar simulado, CSUW31 (Realizar simulado). |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| 1. Ao menos um simulado relacionada aos atores precisa estar cadastrada. |
| PÓS-CONDIÇÕES |
| Não há. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW31 / Realizar simulado  Objetivo: Realizar simulado.  Ator Principal: Aluno.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia após o CSUW30 (Listagem de simulados).  2. Na lista de simulados, o ator seleciona a opção “Realizar Simulado”.  3. O sistema inicia o simulado.  4. Execute CSUW32 (Procedimentos do simulado).  5. Ao finalizar o simulado, o sistema retorna ao passo CSUD02 (Selecionar prova). |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| 1. A prova deve estar liberada pelo professor. |
| PÓS-CONDIÇÕES |
| Não há. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUW32 / Procedimentos do simulado  Objetivo: Realizar o simulado.  Ator Principal: Aluno.  Entradas: N/A |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia junto ao simulado.  2. Respondendo questões.  2.1. Questões objetivas.  2.1.1. O sistema mostra uma questão e suas alternativas.  2.1.2. O ator seleciona uma das alternativas e clica em próximo ou seleciona outra questão.  2.1.3. O sistema retorna ao passo 2.1.1.  3. Ao termino do simulado o sistema mostra a opção de revisar ou finaliza-lo.  4. Ao término do processo, o sistema dispõe a nota obtida. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| 1. Caso o ator não responda todas as questões, o sistema o notifica e o impede de concluir o simulado |
| PRÉ-CONDIÇÕES |
| Não há. |
| PÓS-CONDIÇÕES |
| Não há. |

4.2.3. Mobile

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUM01 / Login  Objetivo: Logar no sistema.  Ator Principal: Aluno.  Entradas: ID de usuário e senha. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator acessa o aplicativo mobile.  2. O sistema mostra os campos de login e senha.  3. O ator digita seu ID e senha nos campos.  4. O ator seleciona a opção “Logar”.  5. O sistema valida os campos preenchidos. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| 5.1. Caso o ID ou senha sejam inválidos o sistema deve avisar o autor e retornar ao passo 2. |
| PRÉ-CONDIÇÕES |
| 1. O ator deve estar cadastrado no sistema. |
| PÓS-CONDIÇÕES |
| 1. O ator é logado no sistema e ganha acesso a outras opções. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUM02 / Consultar provas  Objetivo: Consultar informações da prova.  Ator Principal: Aluno.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUM01 (Login).  2. O ator seleciona a opção “Minhas Provas”.  3. O sistema apresenta uma lista de provas relacionadas ao ator com informações básicas sobre a mesma. |
| CENÁRIO ALTERNATIVO |
| 3.1. O ator tem a opção de acessar mais informações da prova.  3.2. O ator tem a opção de visualizar a nota de uma prova especificas e uma simples estatística.  3.3. O ator tem a opção criar um evento na sua agenda pessoal. |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| 1. O ator precisar ter ao menos uma prova relacionada a ele. |
| PÓS-CONDIÇÕES |
| Não há. |

|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUM03 / Agendar prova  Objetivo: Criar um evento na agenda.  Ator Principal: Aluno.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUM02 (Consultar provas).  2. O ator seleciona uma prova apresentada na lista.  3. O sistema dispõe a opção “Marcar na agenda”.  4. O sistema cria um evento na agenda pessoal do ator. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| Não há. |
| PRÉ-CONDIÇÕES |
| 1. A prova selecionada deve estar agendada. |
| PÓS-CONDIÇÕES |
| 1. Evento criado na agenda pessoal do ator. |

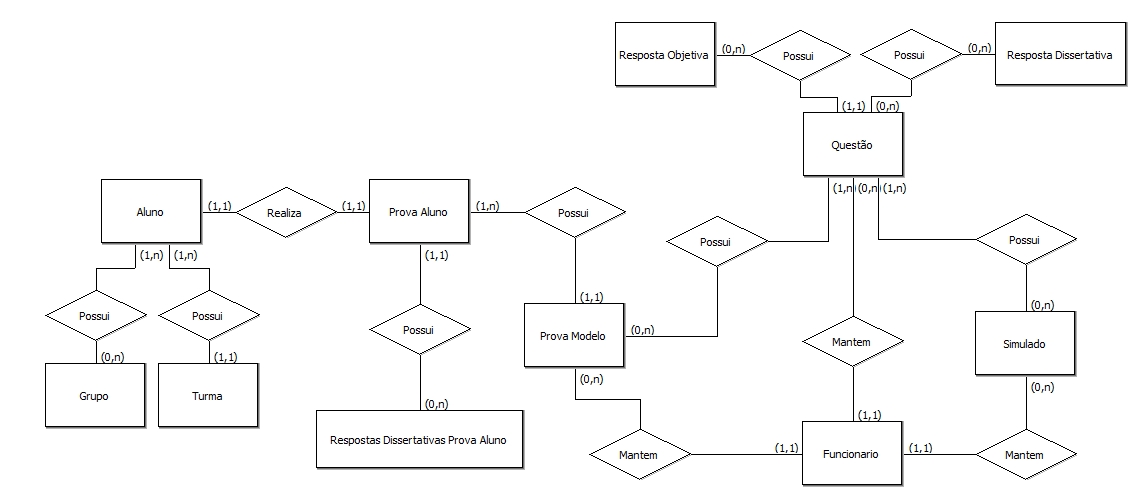
|  |
| --- |
| Narrativa de caso de uso |
| Sigla/nome: CSUM04 / Consultar token  Objetivo: Acesso ao token.  Ator Principal: Aluno.  Entradas: N/A. |
| CENÁRIO PRINCIPAL |
| 1. O caso se inicia quando o ator executa o CSUM02 (Consultar provas).  2. O ator seleciona a opção “Mais informações”.  3. O ator é encaminhado a tela “Informações da Prova”.  4. O token é exibido em um campo especifico. |
| CENÁRIO ALTERNATIVO |
| Não há. |
| CENÁRIO DE EXCEÇÃO |
| 4.1. O token não será exibido caso o professor não o tenha gerado. |
| PRÉ-CONDIÇÕES |
| 1. O ator precisa ter ao menos uma prova agendada. |
| PÓS-CONDIÇÕES |
| Não há. |

## 4.3. Diagrama de Classes

## 4.4. Modelo de Dados

## 4.4.1. Modelo Lógico da base de dados

## 4.4.2. Modelo Conceitual da base de dados



## 4.4.3. Modelo Físico da base de dados

SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;

SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;

SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='TRADITIONAL,ALLOW\_INVALID\_DATES';

**-- Criação do Banco**

CREATE SCHEMA IF NOT EXISTS `dbmonkeystests` DEFAULT CHARACTER SET utf8 ;

USE `dbmonkeystests` ;

**-- Tabela Turma**

CREATE TABLE IF NOT EXISTS `dbmonkeystests`.`turma` (

`id` BIGINT(20) NOT NULL AUTO\_INCREMENT,

`descricao` VARCHAR(255) NULL DEFAULT NULL,

`periodo` TINYINT(1) NULL DEFAULT NULL,

PRIMARY KEY (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

**-- Tabela Aluno**

CREATE TABLE IF NOT EXISTS `dbmonkeystests`.`aluno` (

`id` BIGINT(20) NOT NULL AUTO\_INCREMENT,

`biometria` TINYBLOB NULL DEFAULT NULL,

`celular` VARCHAR(255) NULL DEFAULT NULL,

`cpf` VARCHAR(255) NULL DEFAULT NULL,

`dataNasc` DATETIME NULL DEFAULT NULL,

`email` VARCHAR(255) NULL DEFAULT NULL,

`genero` TINYINT(1) NULL DEFAULT NULL,

`login` VARCHAR(255) NULL DEFAULT NULL,

`matricula` VARCHAR(255) NULL DEFAULT NULL,

`nomeCompleto` VARCHAR(255) NULL DEFAULT NULL,

`senha` VARCHAR(255) NULL DEFAULT NULL,

`status` BIT(1) NOT NULL,

`telefone` VARCHAR(255) NULL DEFAULT NULL,

`turma\_id` BIGINT(20) NULL DEFAULT NULL,

PRIMARY KEY (`id`),

UNIQUE INDEX `UK\_g6otv1ccqwf8a15re4tc1sr9q` (`cpf` ASC),

UNIQUE INDEX `UK\_o02ktdmvvw11l6g24qsw4mio4` (`email` ASC),

UNIQUE INDEX `UK\_2whmwuogxhe12curvx31c95wb` (`login` ASC),

UNIQUE INDEX `UK\_2yc6wv6b5w2aph89au30mw6gj` (`matricula` ASC),

INDEX `FKc2jbyom2ws00xwire0hms5bny` (`turma\_id` ASC),

CONSTRAINT `FKc2jbyom2ws00xwire0hms5bny`

FOREIGN KEY (`turma\_id`)

REFERENCES `dbmonkeystests`.`turma` (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

**-- Tabela Disciplina**

CREATE TABLE IF NOT EXISTS `dbmonkeystests`.`disciplina` (

`id` BIGINT(20) NOT NULL AUTO\_INCREMENT,

`descricao` VARCHAR(255) NULL DEFAULT NULL,

PRIMARY KEY (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

**-- Tabela Funcionário**

CREATE TABLE IF NOT EXISTS `dbmonkeystests`.`funcionario` (

`id` BIGINT(20) NOT NULL AUTO\_INCREMENT,

`cargo` TINYINT(1) NULL DEFAULT NULL,

`celular` VARCHAR(255) NULL DEFAULT NULL,

`cpf` VARCHAR(255) NULL DEFAULT NULL,

`dataNasc` DATETIME NULL DEFAULT NULL,

`email` VARCHAR(255) NULL DEFAULT NULL,

`genero` TINYINT(1) NULL DEFAULT NULL,

`login` VARCHAR(255) NULL DEFAULT NULL,

`nomeCpmpleto` VARCHAR(255) NULL DEFAULT NULL,

`senha` VARCHAR(255) NULL DEFAULT NULL,

`status` BIT(1) NOT NULL,

`telefone` VARCHAR(255) NULL DEFAULT NULL,

PRIMARY KEY (`id`),

UNIQUE INDEX `UK\_iq7jks9kmtnmm74tjfp3b80ha` (`cpf` ASC),

UNIQUE INDEX `UK\_hujfe9giwd0dwuktb8toq7op7` (`email` ASC),

UNIQUE INDEX `UK\_niyc18ow9xt8oaroq7hy5sjp8` (`login` ASC))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

**-- Tabela Grupos**

CREATE TABLE IF NOT EXISTS `dbmonkeystests`.`grupo` (

`id` BIGINT(20) NOT NULL AUTO\_INCREMENT,

`descricao` VARCHAR(255) NULL DEFAULT NULL,

PRIMARY KEY (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

**-- Tabela Relacional Grupo e Aluno**

CREATE TABLE IF NOT EXISTS `dbmonkeystests`.`grupo\_aluno` (

`Grupo\_id` BIGINT(20) NOT NULL,

`alunos\_id` BIGINT(20) NOT NULL,

INDEX `FK5g5d30pkolo48panqq2qxwrg4` (`alunos\_id` ASC),

INDEX `FKr3scsiuw072ba1c05gp5y089u` (`Grupo\_id` ASC),

CONSTRAINT `FK5g5d30pkolo48panqq2qxwrg4`

FOREIGN KEY (`alunos\_id`)

REFERENCES `dbmonkeystests`.`aluno` (`id`),

CONSTRAINT `FKr3scsiuw072ba1c05gp5y089u`

FOREIGN KEY (`Grupo\_id`)

REFERENCES `dbmonkeystests`.`grupo` (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

**-- Tabela Prova Modelo**

CREATE TABLE IF NOT EXISTS `dbmonkeystests`.`provamodelo` (

`id` BIGINT(20) NOT NULL AUTO\_INCREMENT,

`anotacao` VARCHAR(255) NULL DEFAULT NULL,

`consulta` BIT(1) NOT NULL,

`dataRealizacao` DATETIME NULL DEFAULT NULL,

`quantidadeQuestao` INT(11) NOT NULL,

`tempoMinTotal` INT(11) NOT NULL,

`titulo` VARCHAR(255) NULL DEFAULT NULL,

`valorProva` INT(11) NOT NULL,

`professor\_id` BIGINT(20) NULL DEFAULT NULL,

PRIMARY KEY (`id`),

INDEX `FKb8ug8w8rgm2bc1nad86jbc01x` (`professor\_id` ASC),

CONSTRAINT `FKb8ug8w8rgm2bc1nad86jbc01x`

FOREIGN KEY (`professor\_id`)

REFERENCES `dbmonkeystests`.`funcionario` (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

**-- Tabela Prova Aluno**

CREATE TABLE IF NOT EXISTS `dbmonkeystests`.`provaaluno` (

`id` BIGINT(20) NOT NULL AUTO\_INCREMENT,

`anotacao` VARCHAR(255) NULL DEFAULT NULL,

`autenticacao` BIT(1) NOT NULL,

`nota` DOUBLE NOT NULL,

`respostasObjetivas` VARCHAR(255) NULL DEFAULT NULL,

`status` TINYINT(1) NULL DEFAULT NULL,

`token` INT(11) NOT NULL,

`aluno\_id` BIGINT(20) NULL DEFAULT NULL,

`prova\_id` BIGINT(20) NULL DEFAULT NULL,

PRIMARY KEY (`id`),

INDEX `FKn57cxc86iylxe365076543d3h` (`aluno\_id` ASC),

INDEX `FKhwnr1bx0mc29l41m844upg14o` (`prova\_id` ASC),

CONSTRAINT `FKhwnr1bx0mc29l41m844upg14o`

FOREIGN KEY (`prova\_id`)

REFERENCES `dbmonkeystests`.`provamodelo` (`id`),

CONSTRAINT `FKn57cxc86iylxe365076543d3h`

FOREIGN KEY (`aluno\_id`)

REFERENCES `dbmonkeystests`.`aluno` (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

**-- Tabela Questão**

CREATE TABLE IF NOT EXISTS `dbmonkeystests`.`questao` (

`id` BIGINT(20) NOT NULL AUTO\_INCREMENT,

`dataCriacao` TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,

`dataUso` DATETIME NULL DEFAULT NULL,

`dificuldade` TINYINT(1) NULL DEFAULT NULL,

`enunciado` VARCHAR(255) NULL DEFAULT NULL,

`imagem` TINYBLOB NULL DEFAULT NULL,

`palavraChave` VARCHAR(255) NULL DEFAULT NULL,

`status` TINYINT(1) NULL DEFAULT NULL,

`texto` VARCHAR(255) NULL DEFAULT NULL,

`tipoQuestao` TINYINT(1) NULL DEFAULT NULL,

`vidaUtil` INT(11) NOT NULL,

`disciplina\_id` BIGINT(20) NULL DEFAULT NULL,

`professor\_id` BIGINT(20) NULL DEFAULT NULL,

PRIMARY KEY (`id`),

INDEX `FKhmhisi22dsgtiveh6ocem7qvi` (`disciplina\_id` ASC),

INDEX `FKm352ohj2oav2iakqmtpafkv67` (`professor\_id` ASC),

CONSTRAINT `FKhmhisi22dsgtiveh6ocem7qvi`

FOREIGN KEY (`disciplina\_id`)

REFERENCES `dbmonkeystests`.`disciplina` (`id`),

CONSTRAINT `FKm352ohj2oav2iakqmtpafkv67`

FOREIGN KEY (`professor\_id`)

REFERENCES `dbmonkeystests`.`funcionario` (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

**-- Tabela Resposta Dissertativa Prova Aluno**

CREATE TABLE IF NOT EXISTS `dbmonkeystests`.`respostadissertativaprovaaluno` (

`id` BIGINT(20) NOT NULL AUTO\_INCREMENT,

`consideracaoProf` VARCHAR(255) NULL DEFAULT NULL,

`respostaAluno` VARCHAR(255) NULL DEFAULT NULL,

`valorQuestao` DOUBLE NOT NULL,

`questao\_id` BIGINT(20) NULL DEFAULT NULL,

PRIMARY KEY (`id`),

INDEX `FK54uui937sqno7eoyts21ibg1a` (`questao\_id` ASC),

CONSTRAINT `FK54uui937sqno7eoyts21ibg1a`

FOREIGN KEY (`questao\_id`)

REFERENCES `dbmonkeystests`.`questao` (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

**-- Tabela Relacional Prova Aluno e Resposta Dissertativa Prova Aluno**

CREATE TABLE IF NOT EXISTS `dbmonkeystests`.`provaaluno\_respostadissertativaprovaaluno` (

`ProvaAluno\_id` BIGINT(20) NOT NULL,

`respostasDissertativasProvaAluno\_id` BIGINT(20) NOT NULL,

UNIQUE INDEX `UK\_remytvt6wt21w2isthf4w7jiu` (`respostasDissertativasProvaAluno\_id` ASC),

INDEX `FKsmh8cq7j3tmeq6pimcqisdwwp` (`ProvaAluno\_id` ASC),

CONSTRAINT `FK6v9tu4itya5oqk59bbsmpn4yt`

FOREIGN KEY (`respostasDissertativasProvaAluno\_id`)

REFERENCES `dbmonkeystests`.`respostadissertativaprovaaluno` (`id`),

CONSTRAINT `FKsmh8cq7j3tmeq6pimcqisdwwp`

FOREIGN KEY (`ProvaAluno\_id`)

REFERENCES `dbmonkeystests`.`provaaluno` (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

**-- Tabela Relacional Prova Modelo e Questao**

CREATE TABLE IF NOT EXISTS `dbmonkeystests`.`provamodelo\_questao` (

`ProvaModelo\_id` BIGINT(20) NOT NULL,

`questoes\_id` BIGINT(20) NOT NULL,

UNIQUE INDEX `UK\_kwa31byfa0krjtu293pl1o9xh` (`questoes\_id` ASC),

INDEX `FKe51ufm6hu7dbsqn0fe0ijpn45` (`ProvaModelo\_id` ASC),

CONSTRAINT `FKe51ufm6hu7dbsqn0fe0ijpn45`

FOREIGN KEY (`ProvaModelo\_id`)

REFERENCES `dbmonkeystests`.`provamodelo` (`id`),

CONSTRAINT `FKj2xfe6x64djgu7356b7t74rip`

FOREIGN KEY (`questoes\_id`)

REFERENCES `dbmonkeystests`.`questao` (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

**-- Tabela Resposta Dissertativa**

CREATE TABLE IF NOT EXISTS `dbmonkeystests`.`respostadissertativa` (

`id` BIGINT(20) NOT NULL AUTO\_INCREMENT,

`respostaBase` VARCHAR(255) NULL DEFAULT NULL,

`questao\_id` BIGINT(20) NULL DEFAULT NULL,

PRIMARY KEY (`id`),

INDEX `FKa2cew2aw9qgykboc0dy3sx0wm` (`questao\_id` ASC),

CONSTRAINT `FKa2cew2aw9qgykboc0dy3sx0wm`

FOREIGN KEY (`questao\_id`)

REFERENCES `dbmonkeystests`.`questao` (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

**-- Tabela Resposta Objetiva**

CREATE TABLE IF NOT EXISTS `dbmonkeystests`.`respostaobjetiva` (

`id` BIGINT(20) NOT NULL AUTO\_INCREMENT,

`reposta` VARCHAR(255) NULL DEFAULT NULL,

`taCerta` BIT(1) NOT NULL,

`questao\_id` BIGINT(20) NULL DEFAULT NULL,

PRIMARY KEY (`id`),

INDEX `FKsoqlpsnbvei45msskntq7yudl` (`questao\_id` ASC),

CONSTRAINT `FKsoqlpsnbvei45msskntq7yudl`

FOREIGN KEY (`questao\_id`)

REFERENCES `dbmonkeystests`.`questao` (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

**-- Tabela Simulado**

CREATE TABLE IF NOT EXISTS `dbmonkeystests`.`simulado` (

`id` BIGINT(20) NOT NULL AUTO\_INCREMENT,

`quantidadeQuestao` INT(11) NOT NULL,

`tempoMinTotal` INT(11) NOT NULL,

`titulo` VARCHAR(255) NULL DEFAULT NULL,

`professor\_id` BIGINT(20) NULL DEFAULT NULL,

PRIMARY KEY (`id`),

INDEX `FKr4t3mehnq5s1pb2sng56ntkxn` (`professor\_id` ASC),

CONSTRAINT `FKr4t3mehnq5s1pb2sng56ntkxn`

FOREIGN KEY (`professor\_id`)

REFERENCES `dbmonkeystests`.`funcionario` (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

**-- Tabela Relacional Simulado e Questao**

CREATE TABLE IF NOT EXISTS `dbmonkeystests`.`simulado\_questao` (

`Simulado\_id` BIGINT(20) NOT NULL,

`questoes\_id` BIGINT(20) NOT NULL,

UNIQUE INDEX `UK\_jbib4yolg5dw801v0j3vyv7wi` (`questoes\_id` ASC),

INDEX `FKra216otr7oh7ro55vb8ci2fiw` (`Simulado\_id` ASC),

CONSTRAINT `FK9ryodpxva5h6wtelr17hu3gaq`

FOREIGN KEY (`questoes\_id`)

REFERENCES `dbmonkeystests`.`questao` (`id`),

CONSTRAINT `FKra216otr7oh7ro55vb8ci2fiw`

FOREIGN KEY (`Simulado\_id`)

REFERENCES `dbmonkeystests`.`simulado` (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;

## 4.5. Ambiente de desenvolvimento

## 4.5.1. Desktop

• JavaFX;

• Scene Builder.

## 4.5.2. Web

• Spring Framework;

• Hibernate ORM;

• PrimeFaces;

• JXL (JExcelApi) - Import/Export Arquivos Excel;

• iText - Geração de PDF.

## 4.5.3. Mobile

• Android Studio;

• Genymotion.

## 4.5.4. Armazenamento de Dados

• MySQL Workbench 6.3 CE.

# 5. TESTES

## 5.1. Caso de Teste

|  |  |  |  |
| --- | --- | --- | --- |
| ID | CT001 | Tipo de teste | Funcional |
| Objetivo |  |  | |
| Requisito |  |  | |
| Pré-condição |  |  | |
| Nº do Passo | Passos | Resultados Esperados | |
|  |  |  | |
|  |  |  | |

## 5.2. Execução do Plano de Testes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N° | ID de Caso de Teste | Cenário | Resultado Esperado | Resultado Obtido |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

## 5.3. Relatório de incidentes de testes

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de Teste | Descrição | Status | Correção |
|  |  |  |  |
|  |  |  |  |

# 6. PROTÓTIPOS

## 6.1. Desktop

## 6.2. Web

## 6.3. Mobile

# 7. MANUAL DO USUÁRIO

# REFERÊNCIAS BIBLIOGRÁFICAS

# GLOSSÁRIO