Interface gráfica do usuário, Texto

Descrição gerada automaticamente

**Documentação de um**

**Produto de Software**

**Projeto SID**

**Nome dos Alunos:**

|  |  |
| --- | --- |
| **RA** | **Nome** |
| 24.01402-8 | Alvaro Nogueria Junqueira Souza |
| 24.00682-3 | Eike Marchiori Ulinski |
| 24.00918-0 | Gabriel Cezar Ferrassini |
| 24.00020-5 | Lucas de Mattia Peres |
| 24.00947-4 | Victor Hugo Gaetano Oliveira Tavares de Pinho |

[**1** **Descrição do Projeto** 4](#_Toc167474012)

[**2** **Levantamento de Requisitos do Sistema de Software** 4](#_Toc167474013)

[**2.1** **Extração de Requisitos** 4](#_Toc167474014)

[**2.1.1** **Análise da Coleta de Requisitos** 5](#_Toc167474015)

[**3** **Especificação dos Requisitos do Sistema de Software** 5](#_Toc167474016)

[**3.1** **Requisitos Funcionais** 5](#_Toc167474017)

[**3.2** **Requisitos Não-Funcionais** 6](#_Toc167474018)

[**4** **Projeto** 7](#_Toc167474019)

[**4.1** **Diagrama de Casos de Uso** 7](#_Toc167474020)

[**4.1.1** **Especificação de Casos de Uso** 8](#_Toc167474021)

[**4.2** **Diagrama de Classe** 16](#_Toc167474022)

[**4.3** **Diagrama de Sequência** 16](#_Toc167474023)

[**4.4** **Modelo de Banco de Dados** 17](#_Toc167474024)

[**5** **Implementação** 17](#_Toc167474025)

[**6** **Testes** 17](#_Toc167474026)

[**7** **Resultados e Considerações** 18](#_Toc167474027)

# **Descrição do Projeto**

Faremos um jogo no estilo desafio quiz, onde o jogador tem como objetivo responder o máximo de perguntas certas.

* Um quiz onde terão perguntas geradas aleatoriamente, e o usuário tem como objetivo responder o máximo de perguntas corretas possíveis, visando bater o próprio recorde e dos demais jogadores.

As perguntas serão sobre o sistema digestório humano. Usaremos a linguagem Java com bibliotecas de interface gráfica (javax.Swing.\*; java.awt.\*) e outras para diversas funções (java.util\*). Integraremos a linguagem SQL para fazer o uso de banco de dados, com uma tabela para armazenar as perguntas a serem feitas durante o jogo e outra para armazenar os dados do ranking dos melhores jogadores.

# **Levantamento de Requisitos do Sistema de Software**

## **Extração de Requisitos**

**Ferramenta utilizadas**

* **Entrevista**

Fizemos uma reunião com o professor Barbassa, do colégio Piaget com intuito de levantarmos os requisitos do software. Perguntamos sobre os interesses dele, como ele gostaria que fosse o sistema.

**Pergunta: Qual a faixa etária dos usuários do software?**

Resposta: Alunos do fundamental II e ensino médio.

**Pergunta: O cliente tem um código para ser utilizado como código de administrador?**

Resposta: -Não há um código de administrador para ser utilizado.

**Pergunta: O local onde o software será utilizado tem condições para ter um celular?**

Resposta: Sim.

**Pergunta: O protótipo do software é aceitável?**

           Resposta: -Sim

* **Brainstorming**

           Nosso grupo se reuniu, e estipulamos dez minutos para um brainstorming, nessa etapa tivemos muitas ideias que definiram muitos processos do nosso sistema.

### **Análise da Coleta de Requisitos**

Analisando os requisitos do cliente em relação ao software, foi identificado os seguintes requisitos.

* O software deve ser um game educativo com o tema: *Sistema digestório humano*.
* O cliente gostaria de um sistema de ranking para avaliar os alunos;
* O cliente não tem interesse em se cadastrar no sistema;
* O sistema deve ter perguntas retiradas de vestibular.

# **Especificação dos Requisitos do Sistema de Software**

## **Requisitos Funcionais**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificador | Descrição | Prioridade | Requisitos Relacionados |
| RF1 | O game deverá permitir que o usuário administrador crie um cadastro para os alunos e professores, a partir do e-mail e senha. | Alta | RNF1 |
| RF3 | O game deverá conter um sistema de regulamento de áudio. | Média | RNF2, RNF3 |
| RF4 | O game deverá ter um sistema de Ranqueamento. | Alta | RNF1 |
| RF5 | O game deverá ter um sistema de criação de perguntas. | Alta | RNF1 |
| RF6 | O game deverá ter um sistema de Perguntas e respostas no modelo Quiz. | Média | RNF1 |
| RF10 | O game deverá ter um sistema de recuperação de senha através do e-mail do usuário. | Média | RNF1 |

## **Requisitos Não-Funcionais**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificador | Descrição | Prioridade | Requisitos Relacionais |
| RNF1 | O game deverá conter um sistema de Banco de dados | Alta | RF1, RF2, RF4, RF5, RF6, RF7, RF8, RF9, RF10, RF11 |
| RNF2 | O game deverá conter uma trilha sonora | Media | RF3 |
| RNF4 | O game deverá seguir o sistema de design da paleta de cores RGB | Alta |  |

# **Projeto**

## **Diagrama de Casos de Uso**

O diagrama de caso de uso é uma representação visual que descreve as interações entre atores externos e um sistema em questão. Ele é uma parte fundamental da modelagem de requisitos em engenharia de software e é frequentemente usado durante a fase de análise de sistemas. No caso de uso temos os atores representados por bonecos, e os casos de uso representados por elipses.

Diagrama

Descrição gerada automaticamente

### **Especificação de Casos de Uso**

**Especificação de Atores**

|  |  |
| --- | --- |
| **Nome** | **Descrição** |
| Aluno | Usuário do sistema responsável por logar, jogar, e consultar ranking. |
| Professor | Usuário do sistema responsável por logar, criar e excluir perguntas no sistema, e visualizar ranking dos jogadores. |
| Administrador | Usuário do sistema responsável por logar, cadastrar professores e alunos e controlar o acesso dos professores. |
| SysAdmin | Desenvolvedor do sistema, será responsável por criar seu próprio cadastro e do ator administrador, direto no banco de dados |

**Especificação de Casos de Uso – Cadastrar ADM no banco de dados**

**Breve Descrição:** 

Este caso de uso tem como objetivo permitir que o SysAdmin insira um novo usuário diretamente no banco de dados.

**Fluxo Básico**

O SysAdmin interage diretamente com o banco de dados para inserir um novo registro de usuário, fornecendo as informações necessárias (como e-mail, senha etc.)

**Fluxos Alternativo**

Não se aplica.

**Pré-condições**

O SysAdmin possuir acesso ao sistema de banco de dados.

**Pós-condições**

O cadastro do administrador ser registrado no banco de dados

.

**Regras de Negócio**

O sistema necessita de um responsável para controlar o acesso dos usúarios.

**Especificação de Casos de Uso – Cadastro**

**Breve Descrição:** 

Este caso de uso tem como objetivo o cadastro dos atores aluno e professor.

**Fluxo Básico**

Este caso de uso se inicia quando o ator administrador cria um cadastro tanto para professor como para aluno.

1. O ator administrador cadastra o ator aluno ou professor com os dados nome, e-mail e senha.
2. O sistema valida os dados e apresenta tela de confirmação de cadastro.
3. O sistema apresenta a tela de login.

**Fluxos Alternativos**

**Fluxo Alternativo 1: Ator já cadastrado**

O sistema exibe uma mensagem que o ator já está cadastrado e apresenta a tela inicial.

**Pré-condições**

O usuário para qual está sendo criado o e-mail, não pode possuir um cadastro já realizado.

**Pós-condições**

O usuário recebe uma notificação ou confirmação de que o cadastro foi concluído com sucesso.

**Regras de Negócio:**

O ator precisa de um e-mail institucional.

**Especificação de Casos de Uso - Login**

**Breve Descrição:** 

Este caso de uso tem como objetivo que o usuário faça o login na sua conta.

**Fluxo Básico**

1. O sistema apresenta as opções ao usuário: preenchimento de e-mail e senha.
2. usuário preenche os dados: e-mail e senha.
3. usuário confirma na opção [entrar].
4. sistema valida os dados e apresenta tela de menu.

**Fluxos Alternativos**

**Fluxo Alternativo 1: Usuário não cadastrado**

1. O sistema apresenta as opções ao usuário: preenchimento de e-mail e senha.
2. O ator aluno preenche os dados: e-mail e senha.
3. O usuário confirma na opção [entrar].
4. O sistema apresenta uma mensagem “Usuário não possui login cadastrado”.

**Pré-condições**

Possuir um login no sistema.

**Pós-condições**

Ter acesso ao menu do jogo.

**Regras de Negócio:**

O sistema devera validar e diferenciar o acesso de administrador, aluno e professor.

**Especificação de Casos de Uso - Jogar**

**Breve Descrição:** 

Este caso de uso tem como objetivo que o ator aluno jogue o game.

**Fluxo Básico**

1. O sistema apresenta uma questão gerada aleatoriamente em formato de texto, e quatro alternativas de resposta.
2. O usuário confirma uma da alternativa que acredita ser a correta.
3. O sistema valida se o usuário acertou ou não a resposta.
4. No caso de acerto, é apresentado uma caixa informativa com os seguintes conteúdos: um texto “Parabéns, você acertou”, um botão [Próximo] e outro botão [Sair].
5. Quando o usuário clicar em [Próximo], o sistema apresenta a próxima pergunta
6. Quando o usuário clicar em [Sair], é apresentado uma tela informando quantas perguntas foram respondidas, quantos acertos e erros, e porcentagem de acertos e porcentagem de erros, e um botão [Voltar menu].
7. Usuário confirma [Voltar menu], o sistema apresenta tela de menu aluno.

**Fluxos Alternativos**

**Fluxo Alternativo 1: Usuário erra reposta**

1. O sistema apresenta uma questão gerada aleatoriamente em formato de texto, e quatro alternativas de resposta.
2. O usuário confirma uma da alternativa que acredita ser a correta.
3. O sistema valida se o usuário acertou ou não a resposta.
4. No caso de erro, é apresentado uma caixa informativa com os seguintes conteúdos: um texto “Que pena você errou”, texto “Resposta certa:”, um botão [Próximo] e outro botão [Sair].
5. e informa a justificativa da resposta certa.
6. Quando o usuário clicar em [Próximo], o sistema apresenta a próxima pergunta
7. Quando o usuário clicar em [Sair], é apresentado uma tela informando quantas perguntas foram respondidas, quantos acertos e erros, e porcentagem de acertos e porcentagem de erros, e um botão [Voltar menu].
8. Usuário confirma [Voltar menu], o sistema apresenta tela de menu.

**Pré-condições**

O sistema deve gerar perguntas aleatórias.

**Pós-condições**

Salvar a melhor jogada do usuário no ranking.

**Regras de Negócio:**

o usuário deverá responder perguntas sobre o tema sistema digestivo.

**Especificação de Casos de Uso – Administrar perguntas**

**Breve Descrição:** 

Este caso de uso tem como objetivo que o ator professor visualize, crie e delete perguntas já criadas.

**Fluxo Básico**

O sistema apresenta as perguntas criadas, e opções de criar e excluir as perguntas.

1. O sistema apresenta as perguntas criadas, e opção de criar perguntas.
2. O usuário decide criar ou excluir alguma pergunta, clicando em [Criar] para criar uma pergunta, e clicando em [Excluir] para excluir a pergunta.
3. O sistema interage com o botão executando sua função.

**Fluxos Alternativos**

**Fluxo Alternativo 1: Criar perguntas**

1. O sistema apresenta as perguntas criadas, e opção de criar perguntas.
2. O usuário decide criar pergunta clicando em [Criar].
3. O sistema apresenta um campo para criar o enunciado, três campos para criar alternativas erradas, e um para criar a alternativa correta.
4. O usuário preenche os campos como desejar e clica em [Salvar].
5. O Sistema salva a pergunta no banco de dados, e apresenta a tela anterior.

**Fluxo Alternativo 2: Excluir perguntas**

1. O sistema apresenta as perguntas criadas, e opção de excluir as perguntas.
2. O usuário seleciona a pergunta que quer excluir.
3. O usuário clica em [Excluir].
4. O sistema exclui a pergunta do banco dados.

**Pré-condições**

Ter acesso a conta de professor.

**Pós-condições**

Sistema salvar edição no banco de dados.

**Regras de Negócio:**

O sistema deve dar a opção de administrar perguntas.

**Especificação de Casos de Uso - Deletar acesso**

**Breve Descrição:** 

Este caso de uso tem como objetivo que o ator administrador controle o acesso dos professores.

**Fluxo Básico**

1. O sistema apresenta uma tela com opções de acessar cadastro de professor, e de aluno.
2. O usuário acessa uma das opções.
3. O sistema apresenta uma tela com todos os cadastros salvos no sistema.
4. O usuário clica no cadastro.
5. O sistema apresenta a opção de excluir.
6. O usuário confirma em [Excluir].
7. O sistema deleta o cadastro do banco de dados.

**Fluxos Alternativos**

**Fluxo Alternativo 1: Não existir cadastro**

1. O sistema apresenta uma tela com as opções de acessar cadastro de professor, e de aluno.
2. O usuário acessa uma das opções.
3. O usuário apresenta uma tela sem cadastros e um botão de voltar
4. O usuário confirma em [Voltar].
5. O sistema apresenta a tela de “Menu administrador"

**Pré-condições**

Ter logado com conta de administrador.

**Pós-condições**

O sistema deve apagar o acesso deletado.

**Regras de Negócio:**

O administrador tem que ser o responsável por criar e excluir acesso usuários.

**Especificação de Casos de Uso – Recuperar senha**

**Breve Descrição:** 

Este caso de uso tem como objetivo o usuário recuperar sua senha.

**Fluxo Básico**

1. O sistema apresenta campos “Login” e “Senha” e um botão [Recuperar senha].
2. O usuário clica em [Recuperar senha].
3. O sistema apresenta uma janela para inserir o e-mail, e um botão [Gerar token].
4. O usuário insere seu e-mail cadastrado no sistema e confirma no botão [Gerar o token].
5. O sistema gera um token, e envia no e-mail do usuário, e apresenta um capo para inserir token.
6. O usuário visualiza o token no e-mail, e insere no sistema.
7. O sistema valida o token.
8. O sistema apresenta um campo “Inserir senha”.
9. O usuário redefine a senha.
10. O sistema apresenta a tela de login.

**Fluxos Alternativos**

**Fluxo Alternativo 1: Inserir token errado**

1. O sistema apresenta campos “Login” e “Senha” e um botão [Recuperar senha].
2. O usuário clica em [Recuperar senha].
3. O sistema apresenta uma janela para inserir o e-mail, e um botão [Gerar token].
4. O usuário insere seu e-mail cadastrado no sistema e confirma no botão [Gerar o token].
5. O sistema gera um token, e envia no e-mail do usuário, e apresenta um capo para inserir token.
6. O usuário visualiza o token no e-mail, e insere no sistema.
7. O sistema informa que o token está errado, e da a opção de inserir o token novamente.

**Pré-condições**

O usuário possuir um cadastro no sistema e decidir mudar senha.

**Pós-condições**

O sistema alterar no banco de dados a senha do usuário.

**Regras de Negócio:**

Possibilidade de recuperar senha.

**Especificação de Casos de Uso -  Consultar ranking**

**Breve Descrição:** 

Este caso de uso tem como objetivo que o ator professor e aluno consulte o ranking dos alunos.

**Fluxo Básico**

1. O sistema apresenta uma interface com uma lista de nome, pontuação e posição dos jogadores, da sua melhor jogada. E um botão de saída.
2. O usuário pode visualizar a lista rolando o scroll do mouse para cima e para baixo.
3. O usuário confirma no botão de saída.
4. O sistema apresenta a tela de menu.

**Fluxos Alternativos**

**Fluxo Alternativo 1: Não ter cadastro de jogadas**

1. O sistema apresenta um campo em branco sem informações, e um botão de saída.
2. O usuário confirma no botão saída.
3. O sistema apresenta a tela de menu.

**Pré-condições**

Ter cesso de aluno e professor.

**Pós-condições**

Não se aplica

**Regras de Negócio:**

Existir um ranking.

## **Diagrama de Classe**

O diagrama de classe abaixo está representando a estrutura do nosso sistema, mostrando classes, seus atributos, métodos e as relações entre as classes.

Diagrama

Descrição gerada automaticamente

## **Diagrama de Sequência**

Nosso diagrama de sequência modela e ilustra a experiencia do ator aluno, simulando a sequência de interação do usuário com os objetos encontrados no nosso sistema. No caso de aluno, desde o login até jogar e consultar ranking.

*Não fizemos o diagrama para os casos de crud.*

Diagrama

Descrição gerada automaticamente

## **Modelo de Banco de Dados**

A modelagem de banco de dados é um processo que envolve a criação de representações abstratas e organizadas dos dados que serão armazenados em um sistema de banco de dados.

 Na modelagem de banco de dados decidimos criar tabelas para ranking, professor, perguntas, administrador e aluno.

Diagrama

Descrição gerada automaticamente

# **Implementação**

Repositório no GitHub dos códigos usados no software.

[GitHub](https://github.com/alvarojnq1/projeto-integrador-2024)

# **Testes**

Link do software sendo testado em vídeo.

[Teste do sistema](https://www.youtube.com/watch?v=vz6kYrM1F_k)

# **Resultados e Considerações**

Prints das telas do sistema de software desenvolvido, e considerações do gerais do projeto, sob o ponto de vista dos requisitos.

**Telas de Login e Redefinição de Senha**

* Na tela de login o usuário vai colocar seu e-mail e senha
* O sistema vai identificar se ele é um aluno, professor ou administrador
* Com essa verificação do sistema, cada usuário vai ter um menu diferente com suas funcionalidades
* O usuário vai ter a opção de clicar em esqueceu a senha para recuperar sua senha
* Ao clicar em recuperar senha ele será direcionado para tela de redefinição de senha onde vai colocar seu e-mail e vai gerar um token.
* Ao gerar um token o usuário vai ser direcionado para outra tela de redefinição de senha
* Nessa tela ele vai colocar o token que recebeu no e-mail, sua nova senha e a confirmação da nova senha
* Com isso ao clicar em redefinir senha, ele altera a senha para qual você escolheu

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

**Telas do Aluno**

O Menu do Aluno tem as funcionalidades de jogar, consultar ranking e acessar configurações.

- **Jogar:**

* + Na tela jogar o aluno vai responder as questões.
  + São 4 questões sendo elas, 3 erradas e 1 certa.
  + Ao acertar a questão vai aparecer uma tela dizendo que o aluno acertou a questão e sua pontuação.
  + Ao errar a questão vai aparecer uma tela dizendo que ele errou a questão e a justificativa dela.
  + Conforme o Aluno vai respondendo as questões, ele vai somando pontos e ao final de responder todas as questões ele vai poder consultar no ranking sua pontuação.

- **Ranking:**

* + Na tela ranking o aluno visualiza uma lista de jogadores, com nomes, pontuação e posição.

- **Configurações**:

Na tela configurações o aluno vai poder aumentar e diminuir a música do jogo.

Tela Menu aluno

Interface gráfica do usuário

Descrição gerada automaticamente

Tela das perguntas (jogo)

Diagrama

Descrição gerada automaticamente

Tela Ranking Tela Configurações

Interface gráfica do usuário

Descrição gerada automaticamente

**Telas do Professor**

O Menu do Professor tem as funcionalidades de administrar perguntas, ver ranking e ver configurações

- **Administrar perguntas:**

* + Na tela administrar perguntas o professor vai poder criar as perguntas para os alunos jogarem

**- Ranking:**

* + Na tela ranking o Professor vai poder ver a posição, nome e pontuação do Alunos

**- Configurações:**

* + Na tela configurações o Professor vai poder aumentar e diminuir a música do jogo

Tela Menu professor

Interface gráfica do usuário

Descrição gerada automaticamente

Tela administrar perguntas

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Tela criar perguntas

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

TelaRanking Tela Configurações

Interface gráfica do usuário

Descrição gerada automaticamente

**Telas do Administrador**

O Menu do Administrador tem as funcionalidades de cadastrar e administrar professores e alunos, visualizar configurações.

- **Cadastrar usuário:**

* + Na tela cadastrar usuários o administrador vai poder cadastrar alunos e professores com nome, e-mail e senha

- **Administrar usuários**

* + Na tela administrar usuários o administrador vai poder excluir alunos e professores

- **Configurações:**

* + Na tela configurações o Administrador vai poder aumentar e diminuir a música do jogo

Tela Menu administrador

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Telas cadastro usuário

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Tela administra usuários

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Tela Ranking Tela configurações

Interface gráfica do usuário

Descrição gerada automaticamente

## **Resultado**

**Requisitos alcançados**:

* + - **Jogo para alunos do fundamental II**: A lógica do desafio do nosso game é dinâmica e muito educativa, perfeita para a faixa etária do público alvo.
    - **Temática de sistema digestório:** Apesar de ser livre o tema das perguntas, para cadastrar, deixamos algumas perguntas já cadastradas no sistema, do tema pedido.
    - **Sistema de ranking:** Montamos um sistema de ranking, como foi solicitado.
    - **O jogo deve ter questões de vestibular:** Optamos por deixar o cadastro com os professores, e o próprio deve fazer as escolhas das perguntas, assim, podendo colocar perguntas de vestibular ou não, se preferir.

**Requisitos que não foram atendidos:**

* **Cliente não tem interesse em se cadastrar:** Deixamos um cadastro pronto para um usuário administrador, que cuidara do acesso dos demais usuários. Apesar de fugir do que nos foi solicitado, acreditamos que esse tenha sido o melhor método para o caso.
* **Cliente não tem interesse em cadastrar perguntas:** Criamos um sistema onde o usuário professor cria e deleta perguntas. Optamos por esse sistema para deixar o uso do software mais flexível, e seu tempo de uso ser mais aproveitado durante os anos, afinal, o professor pode excluir perguntas que já não são mais necessárias, ou criar novas.

## **Considerações**

Alcançamos grande parte dos requisitos que nos foram solicitados, e no meio do desenvolvimento do software optamos por modificar alguns detalhes, por exemplo, deixar que o professor crie as perguntas, como foi citado acima, porque acreditamos que dessa forma nosso game não ficaria obsoleto em quesito perguntas. O sistema está bem interativo e fácil de ser usado tanto para pessoas que tem facilidade com computadores, quanto para usuários que não tem muita familiaridade.

.

# **Referências**

ORACLE. Java SE Documentation. Disponível em: [https://docs.oracle.com/en/java/.](https://docs.oracle.com/en/java/.%20) Acesso em: 15 abril 2024.

ORACLE. MySQL Documentation. Disponível em: [https://dev.mysql.com/doc/.](https://dev.mysql.com/doc/.%20) Acesso em: 10 abril 2024.

GITHUB. GitHub Docs. Disponível em: <https://docs.github.com/en>. Acesso em: 24 abril 2024.

BRASIL ESCOLA. Sistema Digestivo. Disponível em: <https://brasilescola.uol.com.br/biologia/sistema-digestivo.htm>. Acesso em: 3 abril 2024.