



INSTITUTO FEDERAL

Farroupilha

Campus Uruguaiana

CURSO TÉCNICO EM INFORMÁTICA INTEGRADO AO ENSINO MÉDIO

GABRIELA DE OLIVEIRA TAVARES

FERRAMENTA PARA AUXILIAR NO ENSINO DE PROGRAMAÇÃO

URUGUAIANA

2025

GABRIELA DE OLIVEIRA TAVARES

FERRAMENTA PARA AUXILIAR NO ENSINO DE PROGRAMAÇÃO

Trabalho de Conclusão de Curso apresentado ao Curso Técnico em Informática Integrado ao Ensino Médio do *Campus* Uruguaiana do Instituto Federal de Educação, Ciência e Tecnologia Farroupilha como requisito parcial para a obtenção do título de Técnico em Informática.

Orientadores:

Me. Leandro Martins Dallanora

Me. Kauana Rodrigues Amaral

URUGUAIANA

2025

Tavares, Gabriela.

Título : Ferramenta para auxiliar no ensino de programação /
Gabriela de Oliveira Tavares — 2025.

[qtd. de folhas] f.

Trabalho de Conclusão de Curso Técnico – Instituto
Federal de Educação, Ciência e Tecnologia Farroupilha,
Uruguaiana, 2023.

1. Ensino de programação. 2. []. 3. []. I. Ferramenta para
auxiliar no ensino de programação.

CDD [número da CDD].

GABRIELA DE OLIVEIRA TAVARES

FERRAMENTA PARA AUXILIAR NO ENSINO DE PROGRAMAÇÃO

Trabalho de Conclusão de Curso apresentado ao Curso Técnico em Informática Integrado ao Ensino Médio do *Campus* Uruguaiana do Instituto Federal de Educação, Ciência e Tecnologia Farroupilha como requisito parcial para a obtenção do título de Técnico em Informática.

Este trabalho foi defendido e aprovado pela banca em 09/01/2026.

BANCA EXAMINADORA

Prof.Me. Leandro Martins Dallanora
Orientador

Me. Kauana Rodrigues Amaral
Co-orientadora

Prof.Me. Thiago Cassio Krug
Avaliador

Crislaine Espíndola
Avaliadora

AGRADECIMENTOS

É com muita gratidão que reconheço e agradeço a todos que, de alguma forma, contribuíram para a realização deste Trabalho de Conclusão de Curso. A Deus, agradeço por ter sido minha fonte de força e persistência em cada etapa. Aos meus orientadores, Leandro Dallanora e Kauana Amaral, agradeço pela paciência, dedicação, sabedoria e orientação durante todo o processo, cujas contribuições foram essenciais para o desenvolvimento deste trabalho. À banca examinadora, agradeço pela compreensão e pelas valiosas contribuições, sugestões e avaliações realizadas, as quais contribuíram significativamente para o aprimoramento e a qualidade deste trabalho. À minha família, especialmente aos meus pais, agradeço por todo o amor, apoio e encorajamento incondicional, sendo minha base e motivação nos momentos de incerteza e dificuldade. Às minhas amigas, agradeço pela motivação, troca de ideias, pelo apoio e por tornarem meus dias mais leves, com boas risadas e memórias que sempre levarei comigo. Ao meu namorado Dimitri Fernandes, agradeço pela compreensão, paciência e incentivo constantes ao longo dessa trajetória. Sua presença, carinho e palavras de encorajamento foram fundamentais para que eu seguisse firme até a conclusão deste trabalho.

“O impossível é só questão de opinião.
CHARLIE BROWN JR.

RESUMO

Este trabalho apresenta o desenvolvimento de uma ferramenta educacional interativa baseada em elementos de gamificação voltada ao ensino de programação. A proposta tem como foco principal criar uma experiência de aprendizado mais leve, interessante e eficaz, capaz de estimular a motivação dos alunos e fortalecer a assimilação dos conteúdos técnicos. Este trabalho tem como objetivos desenvolver uma ferramenta educacional interativa, fundamentada em elementos de gamificação, com o intuito de auxiliar o processo de ensino e aprendizagem na área de programação; promover o engajamento dos alunos por meio de uma abordagem mais dinâmica e divertida no processo de aprendizagem; desenvolver jogos digitais que auxiliem na fixação dos conteúdos relacionados à programação, utilizando práticas interativas e implementar a solução utilizando as tecnologias PHP, HTML, CSS e MySQL, de modo a garantir funcionalidade e compatibilidade com a proposta educacional. O projeto resultará na criação do jogo *CodGame*. O público alvo é o aluno do primeiro ano do curso técnico em Informática que geralmente não possui contato prévio com linguagens de programação.

Palavras-chave: gamificação; ensino de Programação.

LISTA DE ILUSTRAÇÕES

Figura 1 – Diagrama de Casos de Uso	22
Figura 2 – Modelo Banco de Dados	29
Figura 3 – Tela Inicial	30
Figura 4 – Cadastro dos Usuários	31
Figura 5 – Escolha dos Quizzes	31
Figura 6 – Escolha da Linguagem	32
Figura 7 – Quizzes	32
Figura 8 – Resultado	33
Figura 9 – Quizzes do Docente	33
Figura 10 – Perfil do Docente	34
Figura 11 – Criar Quizzes	34
Figura 12 – Gerenciar Quizzes	35

LISTA DE QUADROS

Quadro 1 - Pontos positivos e negativos dos sistemas semelhantes	18
Quadro 2 - [UC 001]	22
Quadro 3 - [UC 002]	23
Quadro 4 - [UC 003]	23
Quadro 5 - [UC 004]	23
Quadro 6 - [UC 005]	24
Quadro 7 - [UC 006]	24
Quadro 8 - [UC 007]	25
Quadro 9 - [UC 008]	25
Quadro 10 - [UC 009]	26
Quadro 11 - [UC 010]	26
Quadro 12 - [UC 011]	27
Quadro 13 - [UC 012]	27
Quadro 14 - [UC 012]	28

LISTA DE ABREVIATURAS E SIGLAS

TCC Trabalho de Conclusão de Curso

CSS Folhas de Estilo em Cascata

HTML Linguagem de Marcação de Hipertexto

PHP Pré-Processador de Hipertexto

MySQL Sistema de Gerenciamento de Banco de Dados

SUMÁRIO

1 INTRODUÇÃO.....	12
1.1 OBJETIVOS.....	12
1.1.1 Objetivo Geral.....	13
1.1.2 Objetivos Específicos.....	13
2 METODOLOGIA.....	13
3 REVISÃO BIBLIOGRÁFICA.....	14
3.1 Estudos Relacionados.....	14
3.2 Gamificação e Ensino de programação.....	15
3.3 Sistemas Semelhantes.....	17
3.3.1 CodinGame.....	18
3.3.2 CodeCombat.....	18
3.3.1 CodeMonkey.....	18
4 DESENVOLVIMENTO DO SISTEMA.....	19
4.1 Propriedades e Requisitos.....	19
4.1.1 Atores do Sistema.....	19
4.1.2 Requisitos Funcionais.....	20
4.2 Casos de Uso.....	21
4.2.1 Documentação dos Casos de Uso.....	22
4.3 Banco de Dados.....	28
4.4 Interfaces do Sistema.....	30
5 CONSIDERAÇÕES FINAIS.....	35
REFERÊNCIAS.....	37

1 INTRODUÇÃO

Muitos estudantes dos cursos técnicos enfrentam dificuldades para organizar seus estudos, acompanhar os conteúdos das aulas e se preparar adequadamente para as avaliações. Métodos tradicionais de ensino, frequentemente utilizados, acabam se tornando cansativos e pouco atrativos, o que compromete o engajamento dos discentes e a eficiência do processo de aprendizagem.

Considerando esse cenário, este trabalho apresenta o desenvolvimento de um quiz educacional gamificado voltado ao ensino de programação para alunos iniciantes. A ferramenta foi elaborada com o objetivo de tornar o aprendizado mais dinâmico, atrativo e interativo, estimulando o raciocínio lógico por meio de desafios progressivos e *feedback* imediato. A proposta tem como foco principal criar uma experiência de aprendizado mais leve, interessante e eficaz, capaz de estimular a motivação dos alunos e fortalecer a assimilação dos conteúdos técnicos.

O projeto resultará em um ambiente denominado CodGame, no qual o aluno poderá responder questões relacionadas à lógica de programação, acumular pontuação, visualizar seu desempenho e comparar resultados por meio de ranking. O público-alvo principal são estudantes do primeiro ano do Curso Técnico em Informática, que geralmente têm seu primeiro contato com conceitos lógicos de programação nesse período.

Este trabalho justifica-se pela necessidade de atender às demandas contemporâneas do ensino técnico, sobretudo diante do desinteresse causado pelos métodos convencionais. Ao incorporar elementos de gamificação, o quiz busca promover motivação, engajamento e desenvolvimento cognitivo, possibilitando que o aluno aprenda por meio do desafio e da experimentação. Dessa forma, o CodGame surge como uma alternativa prática e atrativa para apoiar o ensino de programação no contexto da educação técnica.

1 OBJETIVOS

1.1.1 Objetivo Geral

Desenvolver uma ferramenta educacional interativa, fundamentada em elementos de gamificação, com o intuito de auxiliar o processo de ensino e aprendizagem na área de programação.

1.1.2 Objetivos Específicos

1. Promover o engajamento dos alunos por meio de uma abordagem mais dinâmica e divertida no processo de aprendizagem.
2. Desenvolver jogos digitais que auxiliem na fixação dos conteúdos relacionados à programação, utilizando práticas interativas.
3. Desenvolver e implementar a solução proposta com base em princípios de design educacional e de gamificação, garantindo sua funcionalidade e aplicabilidade no contexto do ensino técnico.

2 METODOLOGIA

Este capítulo descreve os procedimentos adotados para o desenvolvimento deste trabalho, onde serão descritas todas as etapas que foram necessárias, e seus respectivos objetivos.

Revisão bibliográfica: Processo de busca, seleção e análise de publicações acadêmicas relevantes sobre o tema. Essa revisão teve como objetivo compreender os conceitos relacionados à tecnologia educacional, gamificação e ensino técnico, além de identificar abordagens já existentes que pudessem fundamentar teoricamente o projeto.

Análise de sistemas semelhantes: Pesquisa sobre softwares já existentes na área, com o intuito de identificar boas práticas, funcionalidades relevantes e possíveis falhas. Essa análise contribuiu diretamente para o desenvolvimento de uma solução mais eficaz, baseada em referências reais e voltada às necessidades do público-alvo.

Definição dos requisitos funcionais: A definição das necessidades funcionais e estruturais da plataforma foi realizada por meio de leituras, conversas com o orientador e a análise dos principais desafios enfrentados pelos alunos. Esse processo possibilitou a identificação e organização das funcionalidades fundamentais da ferramenta.

Desenvolvimento do sistema: O sistema foi desenvolvido utilizando as linguagens de programação PHP, HTML, CSS e JavaScript, com o apoio de um banco de dados MySQL para o armazenamento e gerenciamento das informações.

Testes do sistema: Após a conclusão do desenvolvimento, foram realizados testes no sistema com o objetivo de verificar seu correto funcionamento e garantir que todas as funcionalidades propostas fossem atendidas de forma eficiente.

3 REVISÃO BIBLIOGRÁFICA

Para a construção deste trabalho foi realizada uma breve pesquisa nos sites de tecnologia e os jogos Codingame, Combate, Monkey, serviram como exemplo. Além do Scratch que é uma linguagem de programação visual projetada para ensinar programação de forma acessível e divertida, especialmente para crianças e novatos.

Este trabalho se diferencia porque é voltado para alunos do primeiro ano do ensino técnico na área de informática que nunca tiveram contato com linguagens de programação.

3.1 ESTUDOS RELACIONADOS

Frosi e Jaques, em 2020 publicaram uma revisão sistemática sobre o uso dos jogos para o ensino de programação no período de 2015 a 2019. Eles concluíram que “existem diversos trabalhos com enfoque no ensino fundamental, médio e técnico, tanto no contexto público quanto privado. Esse aspecto é apontado como positivo, dado que o desenvolvimento do pensamento computacional pode ser considerado uma construção que trará resultados em longo prazo, se tiver como enfoque o ensino fundamental”. (FROSI; JAQUES, 2020, p. 661)

Um outro estudo realizado em 2021, por Souza, Falcão e Mello concluíram que fica evidente as diversas contribuições positivas que a aprendizagem de

programação pode proporcionar, inclusive tornando os alunos mais aptos a resolverem problemas de outras disciplinas, através de conceitos computacionais. E a maioria dos trabalhos selecionados de acordo com os critérios estabelecidos teve o Scratch como ferramenta escolhida para abordar os conteúdos de lógica de programação.

O trabalho dos autores Sousa e Leite publicado em 2020 investiga o ensino de programação por meio de juízes online. Como resultado, identificou-se que os estudantes que utilizaram a gamificação, geralmente, tiveram melhores desempenhos e que esta abordagem ajudou a minimizar problemas como o entendimento da lógica de programação.

O artigo publicado também em 2020 por Stephan, Oliveira e Renhe, apresenta o GameProg, uma abordagem para apoiar o ensino e aprendizagem de programação de maneira lúdica e interativa usando como recurso o desenvolvimento de jogos. A sua implementação foi pautada no uso da linguagem C, a partir do CodeBlocks e apoiado pela biblioteca SFML. O objetivo do GameProg é adotar estratégias inovadoras para aumentar o interesse e a motivação do aluno, bem como reduzir a evasão dos cursos de Ciência da Computação e áreas afins. Foi apresentada aos alunos uma primeira versão do GameProg e a avaliação desta versão foi feita através de um questionário. A análise das respostas indicou que a abordagem motivou os alunos com relação ao estudo de programação.

Por fim, o estudo realizado por Passos et al. (2020), compreendeu a aplicação do jogo digital Code Combat, em sala de aula, com o intuito de verificar como os alunos interagiam com esse jogo visando o aprendizado de programação. A programação é muito importante para o tempo atual e seu ensino e aprendizagem é fundamental nos cursos voltados à Tecnologia da Informação (TI). No entanto, no processo de aprendizagem, muitos estudantes se deparam com dificuldades em raciocinar e abstrair respostas para a solução de problemas encontrados. Assim, esse levantamento serviu como embasamento teórico para este trabalho, conforme o capítulo a seguir.

3.2 Gamificação e o ensino de programação

De acordo com Ferraz Junior; Homem; Santos (2017, p. 212) “Os jogos eletrônicos, na visão das crianças e adolescentes, são um meio muito divertido de

aprender. Além disso, propiciam habilidade cognitiva, pois exercem a função de exercício mental, aumentando as conexões neurais e alterando a circulação sanguínea no cérebro quando o jogador entra em estado de imersão e interação.” Nesse sentido, entende-se que os jogos contribuem para aprendizagem, pois os sujeitos aprendem sozinhos se desafiando e cumprindo tarefas através da interação com outros jogadores ou até mesmo com o software.

Ainda, os referidos autores acima apresentam uma abordagem chamada *serious game* quando se trata de jogos educacionais que tem por objetivo ensinar algo ao jogador. “O termo serious refere-se a produtos e situações que áreas como defesa, educação, exploração científica, saúde, gestão de emergência, planejamento urbano, engenharia, religião e política”. Ferraz Junior; Homem; Santos (2017, p. 216) A intenção é combinar ensino com aspectos lúdicos através do game. Na década de 1980 surge o primeiro *game* com características de aprendizagem, o Army Battlezone, desenvolvido para treinar militares em situação de batalha.

Nessa perspectiva, este trabalho tem por objetivo desenvolver um jogo que facilite o aprendizado de programação, pois “jogos que possuem objetivos de aprendizado, geralmente são criados para equilibrar o assunto com a jogabilidade e a habilidade do jogador de reter e aplicar parte do assunto no mundo real” Ferraz Junior; Homem; Santos (2017, p. 217).

A gamificação possibilita o engajamento do estudante, é uma forma de motivá-los a aprender programação de forma mais prazerosa, pois essas plataformas de jogos geralmente tem recompensas. Além de que os jogos possibilitam e estimulam o raciocínio lógico de forma lúdica e dinâmica, aumentando o interesse por parte do aluno pela disciplina de programação.

É importante ressaltar que os jogos são ferramentas eficazes para ensinar programação, pois tornam o processo de aprendizagem mais divertido e envolvente. Por meio de desafios e interações atrativas, os jogos ajudam os alunos a compreenderem os conceitos básicos da programação e a desenvolverem habilidades práticas. Além disso, jogos educacionais específicos para programação fornecem um ambiente seguro e experimental para os alunos praticarem e aplicarem seus conhecimentos (SILVA et al., 2021 apud GENESIO et al., 2023 p.3)

Os jogos têm se mostrado uma maneira eficiente de ensinar programação, pois deixam o aprendizado mais interessante e envolvente. Ao propor desafios e

oferecer interações dinâmicas, eles facilitam a compreensão dos conceitos básicos da programação e incentivam a prática. (GENESIO et. al. 2023)

A disciplina de Programação exige conhecimentos prévios de raciocínio lógico e como o aluno chega no primeiro ano sem essa base, acaba não compreendendo o conteúdo. De maneira geral, (SOUSA; LEITE, 2020, p. 33340) afirmam que “o primeiro contato dos estudantes com programação é a disciplina “Algoritmos”, a qual apresenta os elementos básicos da linguagem, tais como os fundamentos de programação utilizada, estruturas condicionais e de repetição, além de conceitos mais avançados, tais como funções, vetores, matrizes” e considera-se um agravante os estudantes não terem um contato com linguagens de programação durante o ensino fundamental e também podem apresentar dificuldade com um idioma estrangeiro.

Diante do exposto, reforça-se a necessidade de estratégias pedagógicas que tornem o aprendizado mais acessível e motivador. A falta de engajamento pode prejudicar ainda mais o desempenho dos alunos, tornando essencial a adoção de práticas pedagógicas que estimulem o interesse e a participação ativa durante as aulas. Portanto, os conceitos de Programação e a quantidade de informações que os alunos precisam absorver não são simples nem fáceis de compreender, especialmente para iniciantes, diante disso, esses fatores reforçam a importância dos jogos como ferramenta no ensino de programação.

Portanto, a educação digital e inclusão digital devem caminhar juntas, pois alguns alunos chegam no ensino médio sem contato com o computador, mesmo vivendo em uma sociedade informatizada e assim os games precisam ser inseridos na comunidade escolar. E através da programação mudar a forma de entender o cotidiano e que através dos jogos o estudante seja independente na sua aprendizagem e utilize as ferramentas tecnológicas a seu favor.

3.3 Sistemas Semelhantes

Para realização deste trabalho foi necessário a consulta de outros sistemas já existentes .

3.3.1 CodinGame

O CodinGame é uma plataforma online voltada para o ensino e prática de programação por meio de desafios e minigames interativos. Os usuários resolvem problemas utilizando linguagens reais como Python, Java, C++ e muitas outras. Os desafios possuem níveis de dificuldade variados, e a progressão ocorre à medida que o aluno supera obstáculos programando em tempo real. A ferramenta é amplamente usada por programadores iniciantes e experientes.

3.3.2 CodeCombat

O CodeCombat é uma plataforma gamificada que ensina programação por meio de um ambiente de RPG. O aluno controla um personagem que progride pelo jogo escrevendo códigos para executar ações (andar, atacar, defender, etc.). É intuitivo e usa Python e JavaScript como principais linguagens. Além disso, possui recursos visuais atrativos e uma narrativa envolvente.

3.3.3 CodeMonkey

CodeMonkey é uma plataforma educacional voltada para o ensino de programação para crianças e adolescentes. Utiliza jogos baseados em missões, com linguagens simples, onde os alunos resolvem desafios programando. Seu visual é amigável e voltado para a educação básica e ensino fundamental.

Quadro 1 – Pontos positivos e negativos dos sistemas semelhantes

Sistema	Pontos Positivos	Pontos Negativos
CodinGame	Desafios com linguagens reais, progressão por níveis, ranking global.	Pouco indicado para iniciantes absolutos; sem foco pedagógico estruturado.
CodeCombat	Ambiente RPG visual, ideal para iniciantes, ensina Python/JavaScript.	Versão gratuita limitada; depende de internet.

CodeMonkey	Voltado para iniciantes jovens, visual amigável, bom para introdução básica.	Linguagem inicial pouco usada (CoffeeScript); menos indicado para ensino técnico.
------------	--	---

Fonte: Autoria própria.

4 DESENVOLVIMENTO DO SISTEMA

Nesta seção, será apresentado as etapas do desenvolvimento do sistema proposto: requisitos funcionais, casos de uso, documentação de casos de uso, base de dados e interfaces.

4.1 Propriedades e Requisitos

Para definir a prioridade dos requisitos do sistema, foram utilizadas três classificações: **essencial**, **importante** e **desejável**. Cada uma dessas categorias representa um nível de necessidade para o funcionamento adequado da aplicação, conforme descrito a seguir:

- **Essencial:** representa o requisito fundamental para o funcionamento do sistema. Sem ele, a aplicação não opera. São funcionalidades obrigatórias, cuja implementação é indispensável.
- **Importante:** refere-se a funcionalidades que, embora não impeçam o sistema de funcionar, comprometem sua eficiência ou qualidade. São desejadas para o bom uso do sistema, mas sua ausência não inviabiliza o lançamento inicial.
- **Desejável:** são funcionalidades complementares que não afetam diretamente as funções principais do sistema. Podem ser incorporadas em futuras atualizações, caso o tempo de desenvolvimento não permita sua inclusão na primeira versão.

4.1.1 Atores do Sistema

O sistema apresenta dois atores que são diferentes no que diz respeito a suas “vantagens” no contexto do sistema.

Docente: tem permissão para se auto cadastrar, acessar(login), excluir, recuperar e alterar sua conta, além de poder criar quizzes, excluir e editar eles.

Discente: tem permissão para se auto cadastrar, acessar(login), excluir, recuperar e alterar sua conta, além de poder visualizar os tipos de quizzes, ranking, pontuação acumulada e jogar os quizzes.

4.1.2 Requisitos Funcionais

Os requisitos funcionais definem as funcionalidades que o sistema deve realizar para atender aos objetivos propostos. Com base no contexto do sistema, foram identificados os seguintes requisitos funcionais:

[RF01] – Cadastrar-se

Descrição: O sistema deve permitir que o usuário crie uma conta.

Prioridade: Essencial.

[RF02] – Logar Conta

Descrição: O sistema deve permitir que o usuário realize login em sua conta.

Prioridade: Essencial.

[RF03] – Criar quiz

Descrição: O sistema deve permitir que o docente faça quizzes, informando título, linguagem e status (ativo ou inativo).

Prioridade: Importante.

[RF04] – Editar Conta

Descrição: O sistema deve permitir que o usuário altere os dados pessoais da conta.

Prioridade: Importante.

[RF05] – Listar Quizzes do Docente

Descrição: O sistema deve permitir que o docente visualize uma lista de todos os quizzes que cadastrou.

Prioridade: Essencial.

[RF06] – Editar Quiz do Docente

Descrição: O sistema deve permitir que o docente edite quizzes que cadastrou, alterando informações como título, linguagem e status (ativo ou inativo).

Prioridade: Importante.

[RF07] – Excluir Quiz do Docente

Descrição: O sistema deve permitir que o docente exclua ou desative quizzes que cadastrou, de forma que deixem de ficar disponíveis para os discentes jogarem.

Prioridade: Importante.

[RF08] – Recuperar Senha via Email

Descrição: O sistema deve permitir que o usuário recupere a senha por meio de envio de e-mail.

Prioridade: Essencial.

[RF09] – Visualizar tipos de questionários

Descrição: O sistema deve permitir que o discente visualize os tipos de quizzes disponíveis por conteúdo.

Prioridade: Essencial.

[RF10] – Jogar Quizzes Disponíveis

Descrição: O sistema deve permitir que o discente jogue quizzes cadastrados pelo sistema ou pelos docentes.

Prioridade: Essencial.

[RF11] – Visualizar Ranking de Pontuação

Descrição: O sistema deve permitir que o discente visualize o ranking geral de resultados.

Prioridade: Importante.

[RF12] – Visualizar Pontuação Acumulada

Descrição: O sistema deve permitir que o discente visualize sua pontuação total acumulada.

Prioridade: Importante.

[RF13] – Excluir Conta

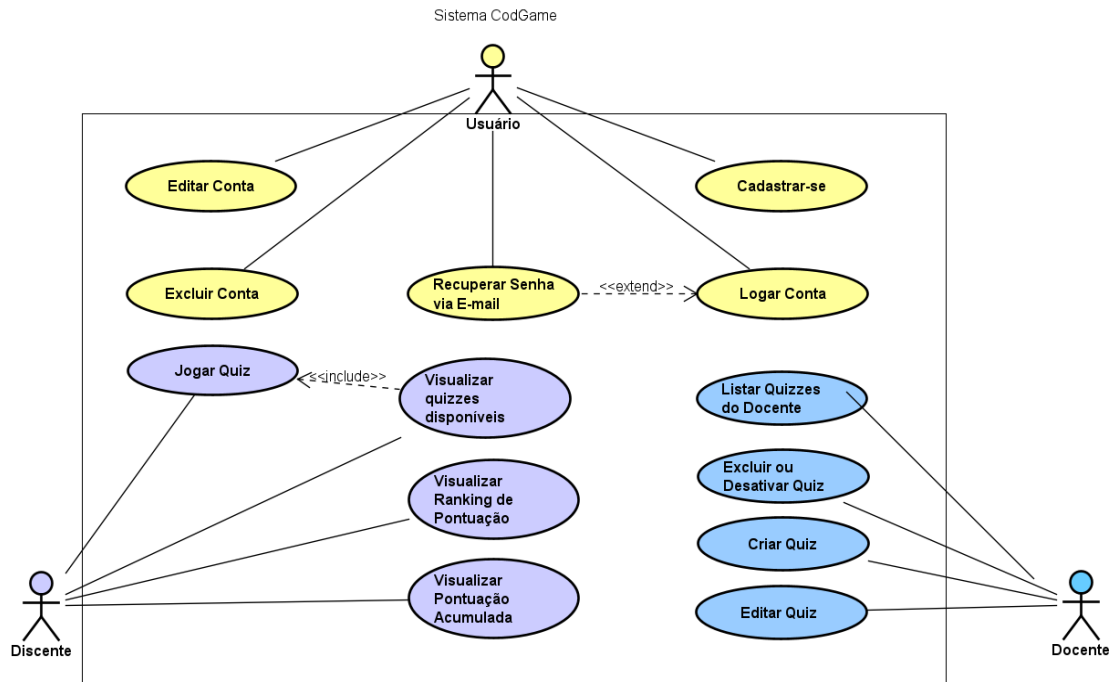
Descrição: O sistema deve permitir que o usuário exclua sua conta.

Prioridade: Desejável.

4.2 CASOS DE USO

O diagrama de casos de uso do sistema CodGame representa graficamente as interações entre os usuários e as principais funcionalidades da aplicação. Nele, são identificados dois atores: Discente e Docente, cada um com permissões específicas. O Discente pode realizar login, visualizar disciplinas, participar de desafios, acompanhar seu progresso e editar o perfil, enquanto o Docente pode se cadastrar, criar desafios e acompanhar o desempenho dos alunos. Assim, o diagrama destaca as principais interações entre os atores e o sistema, conforme os requisitos funcionais do CodGame.

Figura 1 – Diagrama de Casos de Uso



Fonte: Autoria própria.

4.2.1 Documentação dos Casos de Uso

A seguir segue a documentação de Casos de Uso:

Quadro 2 – [UC001]

CASO DE USO	[UC 001] Cadastrar-se
Atores	Docente, Discente
Pré-Condições	Estar na página de cadastro.
Pós-Condições	Acesso ao sistema.
FLUXO PRINCIPAL	
1. O sistema exibe o formulário de cadastro. 2. O usuário preenche os campos e solicita o registro. 3. O sistema registra as informações, exibe o formulário de login e apresenta uma mensagem informando que o cadastro foi realizado com sucesso.	
FLUXO ALTERNATIVO	
Não tem.	
FLUXO EXCEÇÃO	
3.a Erro ao registrar os dados. 3.a.1 O sistema encontra uma falha ao salvar os dados. 3.a.2 O sistema exibe uma mensagem de erro e solicita nova tentativa.	

Fonte: Autoria própria.

Quadro 3 – [UC 002]

CASO DE USO	[UC 002] Logar Conta
Atores	Docente, Discente
Pré-Condições	Possuir conta cadastrada.
Pós-Condições	Usuário autenticado no sistema.
FLUXO PRINCIPAL	
1. O usuário acessa a tela de login. 2. Informa e-mail e senha. 3. O sistema valida as credenciais. 4. O sistema permite o acesso à área principal de cada usuário.	
FLUXO ALTERNATIVO	
Não tem.	
FLUXO EXCEÇÃO	
3.a.Credenciais inválidas. 3.a.1 O sistema exibe mensagem de erro.	

Fonte: Autoria própria.

Quadro 4 – [UC 003]

CASO DE USO	[UC 003] Editar Conta
Atores	Docente, Discente
Pré-Condições	Estar logado e acessar o perfil.
Pós-Condições	Dados atualizados.
FLUXO PRINCIPAL	
1. O usuário acessa o perfil. 2. O usuário modifica suas informações. 3. O sistema salva as alterações.	
FLUXO ALTERNATIVO	
Não tem.	
FLUXO EXCEÇÃO	
3.a Erro ao salvar alterações 3.a.1 O sistema exibe uma mensagem e não atualiza os dados.	

Fonte: Autoria própria.

Quadro 5 – [UC 004]

CASO DE USO	[UC 004] Excluir Conta
Atores	Docente, Discente
Pré-Condições	Estar logado.

Pós-Condições	Conta removida do sistema.
FLUXO PRINCIPAL	
1. O usuário acessa o perfil e seleciona a opção “Excluir Conta”. 2. O sistema solicita confirmação. 3. Após confirmação, a conta é removida.	
FLUXO ALTERNATIVO	
2.a O usuário inicia o processo, mas desiste antes de confirmar. 2.a.1 A exclusão é cancelada.	
FLUXO EXCEÇÃO	
3.a Erro ao excluir conta 3.a.1 O sistema informa falha e orienta tentar mais tarde.	

Fonte: Autoria própria.

Quadro 6 – [UC 005]

CASO DE USO	[UC 005] Criar Quiz
Atores	Docente.
Pré-Condições	Estar logado como docente.
Pós-Condições	Quiz criado.
FLUXO PRINCIPAL	
1. O docente acessa a área de criação de quizzes. 2. Informa título, linguagem, status, enunciado, alternativas e alternativa correta. 3. O sistema salva o quiz.	
FLUXO ALTERNATIVO	
Não tem.	
FLUXO EXCEÇÃO	
3.a Erro ao salvar quiz.	

Fonte: Autoria própria.

Quadro 7 – [UC 006]

CASO DE USO	[UC 006] Listar Quizzes do Docente
Atores	Docente.
Pré-Condições	Estar logado como docente.
Pós-Condições	Lista de quizzes exibida.
FLUXO PRINCIPAL	
1. O docente acessa seus quizzes. 2. O sistema exibe a lista.	
FLUXO ALTERNATIVO	
2.a Não existem quizzes cadastrados.	
FLUXO EXCEÇÃO	

2.b Erro ao carregar lista.

Fonte: Autoria própria.

Quadro 8 – [UC 007]

CASO DE USO	[UC 007] Editar Quiz
Atores	Docente
Pré-Condições	Estar logado e possuir quiz cadastrado.
Pós-Condições	Quiz atualizado.
FLUXO PRINCIPAL	
1. O docente seleciona um quiz. 2. Altera suas informações. 3. O sistema salva as alterações.	
FLUXO ALTERNATIVO	
2.a O docente desiste da edição 2.a.1 O docente acessa o formulário de edição do quiz, mas não confirma as alterações. 2.a.2 O docente seleciona a opção “Cancelar”. 2.a.3 O sistema descarta as alterações e mantém os dados originais do quiz.	
FLUXO EXCEÇÃO	
3.a Erro ao carregar as alterações.	

Fonte: Autoria própria.

Quadro 9 – [UC 008]

CASO DE USO	[UC 008] Excluir ou Desativar Quiz
Atores	Docente
Pré-Condições	Estar logado.
Pós-Condições	Quiz indisponível para discentes.
FLUXO PRINCIPAL	
1. O docente seleciona excluir ou desativar quiz. 2. O sistema confirma a ação. 3. O quiz é removido ou desativado.	
FLUXO ALTERNATIVO	
2.a O docente desiste da exclusão/desativação 2.a.1 O docente seleciona a opção de excluir ou desativar o quiz. 2.a.2 O sistema solicita confirmação da ação. 2.a.3 O docente cancela a operação. 2.a.4 O sistema mantém o quiz ativo e não realiza nenhuma alteração.	
FLUXO EXCEÇÃO	
3.a Erro ao executar ação.	

Fonte: Autoria própria.

Quadro 10 – [UC 009]

CASO DE USO	[UC 009] Visualizar Quizzes Disponíveis
Atores	Discente
Pré-Condições	Estar logado.
Pós-Condições	Quizzes exibido.
FLUXO PRINCIPAL	
1.O discente acessa a área de quizzes. 2.O sistema lista os quizzes ativos.	
FLUXO ALTERNATIVO	
2.a Não há quizzes disponíveis 2.a.1 O sistema verifica que não existem quizzes ativos cadastrados. 2.a.2 O sistema <i>exibe a mensagem: “Não há quizzes disponíveis no momento.”</i>	
FLUXO EXCEÇÃO	
3.a Erro ao carregar os quizzes 3.a.1 O sistema identifica falha na busca dos quizzes. 3.a.2 O sistema <i>exibe uma mensagem informando que não foi possível carregar os quizzes.</i>	

Fonte: Autoria própria.

Quadro 11 – [UC 010]

CASO DE USO	[UC 010] Jogar Quiz
Atores	Discente
Pré-Condições	Estar logado e selecionar um quiz.
Pós-Condições	Pontuação registrada.
FLUXO PRINCIPAL	
1. O discente inicia o quiz. 2. Responde às questões. 3. O sistema calcula e salva a pontuação. 4. O sistema registra a pontuação do discente.	
FLUXO ALTERNATIVO	
2.a O discente abandona o quiz antes de finalizar 2.a.1 O discente sai do quiz sem concluir todas as questões. 2.a.2 O sistema encerra o quiz. 2.a.3 O sistema não registra pontuação.	
FLUXO EXCEÇÃO	
5.a Erro ao registrar a pontuação 5.a.1 O sistema identifica falha no salvamento da pontuação. 5.a.2 O sistema <i>exibe uma mensagem de erro informando que a pontuação não pôde ser registrada.</i>	

Fonte: Autoria própria.

Quadro 12 – [UC 011]

CASO DE USO	[UC 011] Visualizar Ranking de Pontuação
Atores	Discente
Pré-Condições	Ter realizado quizzes.
Pós-Condições	Ranking exibido.
FLUXO PRINCIPAL	
1. O discente acessa a área de ranking. 2. O sistema busca os dados de pontuação dos discentes. 3. O sistema exibe o ranking geral de pontuação.	
FLUXO ALTERNATIVO	
2.a O discente ainda não possui pontuação registrada 2.a.1 O sistema verifica que o discente não possui pontuação. 2.a.2 O sistema exibe a mensagem: <i>“Você ainda não possui pontuação para exibir no ranking.”</i>	
FLUXO EXCEÇÃO	
3.a Erro ao carregar o ranking 3.a.1 O sistema identifica falha ao exibir o ranking. 3.a.2 O sistema exibe uma mensagem informando que não foi possível carregar o ranking.	

Fonte: Autoria própria.

Quadro 13 – [UC 012]

CASO DE USO	[UC 012] Visualizar Pontuação Acumulada
Atores	Discente
Pré-Condições	Ter realizado quizzes.
Pós-Condições	Pontuação total exibida.
FLUXO PRINCIPAL	
1. O discente acessa o perfil. 2. O sistema exibe a pontuação acumulada.	
FLUXO ALTERNATIVO	
2.a O discente ainda não possui pontuação acumulada 2.a.1 O sistema verifica que o discente não possui pontuação registrada. 2.a.2 O sistema exibe a mensagem: <i>“Você ainda não possui pontuação acumulada.”</i>	
FLUXO EXCEÇÃO	
3.a Erro ao exibir a pontuação acumulada 3.a.1 O sistema identifica falha ao exibir os dados. 3.a.2 O sistema exibe uma mensagem informando que não foi possível carregar a pontuação.	

Fonte: Autoria própria.

Quadro 14 – [UC 013]

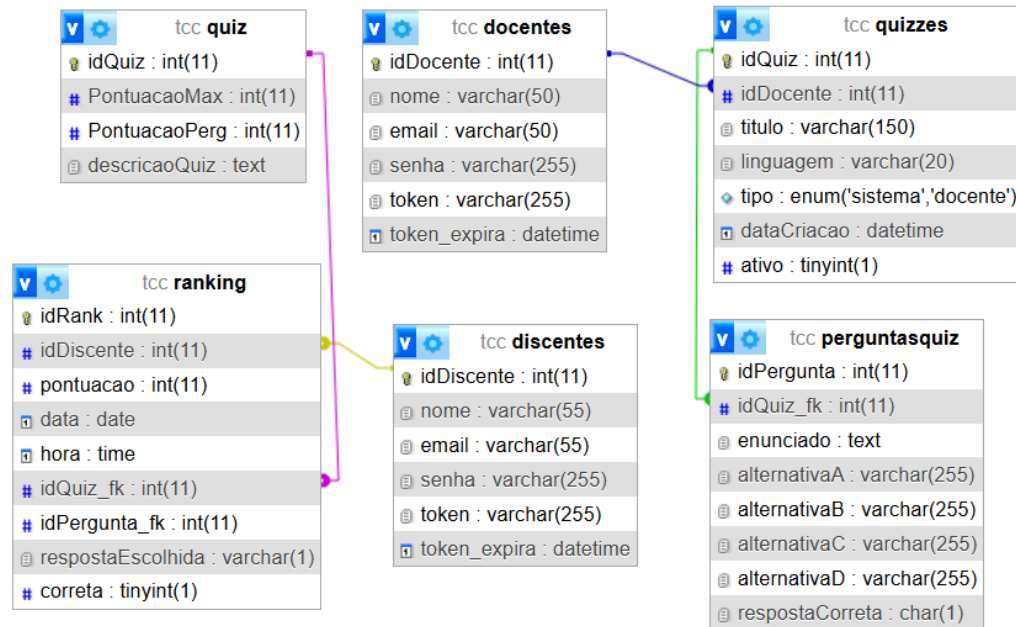
CASO DE USO	[UC 0013] Recuperar Senha via E-mail
Atores	Discente.
Pré-Condições	Ter cadastro com e-mail válido e estar na tela de login.
Pós-Condições	Link de redefinição enviado por e-mail e senha redefinida com sucesso.
FLUXO PRINCIPAL	
1.O discente clica em “Esqueci minha senha” na tela de login. 2.O sistema solicita o e-mail cadastrado. 3.O discente informa seu e-mail. 4.O sistema verifica se o e-mail está cadastrado. 5.O sistema envia um link de redefinição para o e-mail informado. 6.O discente acessa o e-mail e clica no link. 7.O sistema redireciona para a tela de nova senha. 8.O discente insere e confirma a nova senha. 9.O sistema atualiza a senha e exibe mensagem de sucesso.	
FLUXO ALTERNATIVO	
Não tem.	
FLUXO EXCEÇÃO	
2.a E-mail não cadastrado 2.a.1 O sistema exibe a mensagem: “E-mail não encontrado.”	

Fonte: Autoria própria.

4.3 BANCO DE DADOS

A base de dados é fundamental para o funcionamento do sistema, pois permite armazenar as informações de forma organizada e segura. Nela são guardados dados como notícias, times, classificações, competições, jogos, administradores, usuários e outras informações importantes do sistema. A figura apresenta o modelo de banco de dados, mostrando as tabelas utilizadas, os dados armazenados em cada uma e como elas se relacionam entre si.

Figura 2 – Modelo de Banco de Dados

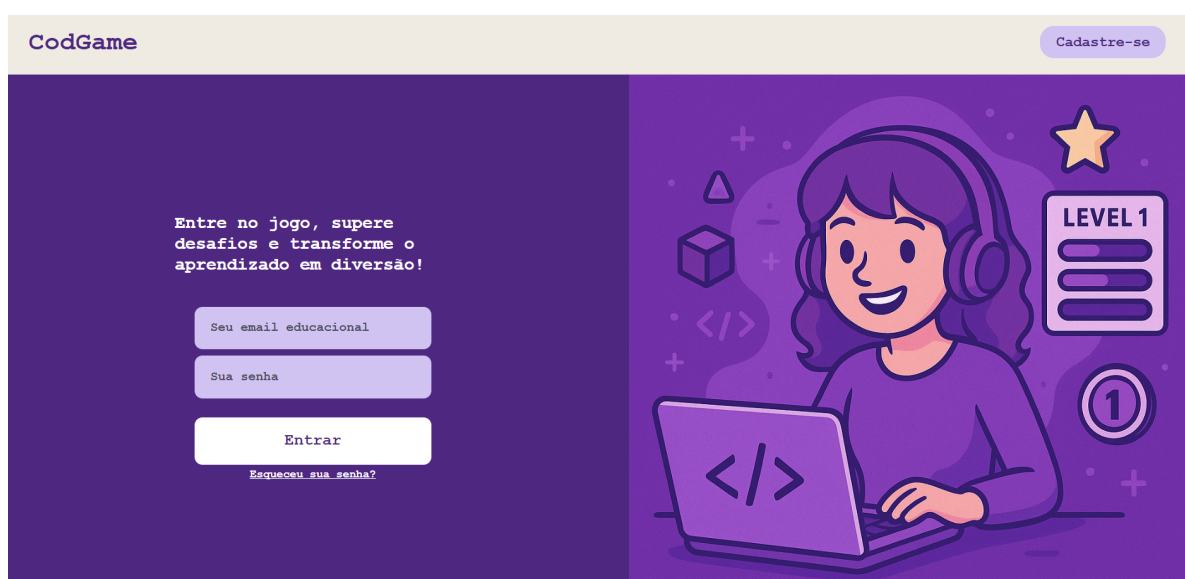


Fonte: Autoria própria

4.4 INTERFACES DO SISTEMA

As interfaces são as telas do sistema responsáveis pela interação dos usuários com a aplicação. Elas apresentam as informações e permitem a execução das funcionalidades disponíveis, sendo desenvolvidas de forma simples e intuitiva para garantir uma navegação fácil e eficiente.

Figura 3 – Tela Inicial



Fonte: Autoria própria

A Figura 3 apresenta a tela inicial do sistema CodGame, onde o usuário pode realizar o login ou optar pelo cadastro, além de visualizar elementos que representam a proposta educacional e gamificada da plataforma.

Figura 4 – Cadastro dos Usuários

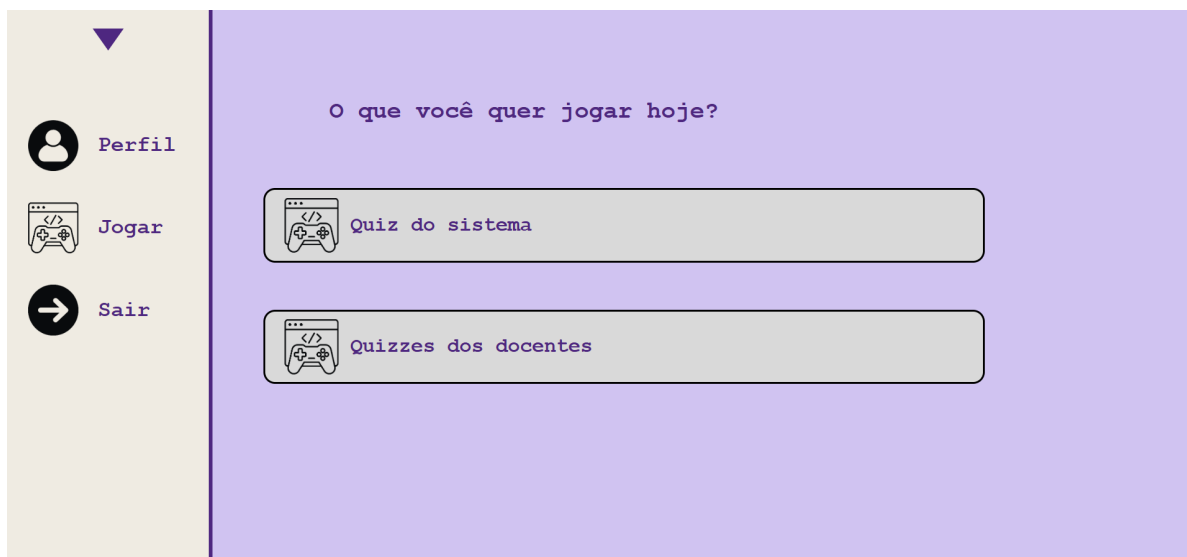


A tela de cadastro do sistema CodGame, com fundo roxo escuro. No centro, há um formulário branco com o título 'CodGame' e o subtítulo 'Cadastre-se para acessar o site'. O formulário contém os seguintes campos: 'Nome completo', 'Email institucional' (com exemplo 'aluno123@escola.com ou professor@escola'), 'Você é:' (com opção 'Selecione seu perfil'), 'Senha' e 'Confirmar senha' (ambas com ícones de olho para alternar visibilidade). Abaixo dos campos está um botão 'Continuar' e um link 'Já possui conta? Voltar para a tela de Login'.

Fonte: Autoria própria

A Figura 4 demonstra a tela de cadastro do sistema CodGame, onde o usuário informa seus dados pessoais, seleciona o perfil e define uma senha para criar sua conta na plataforma.

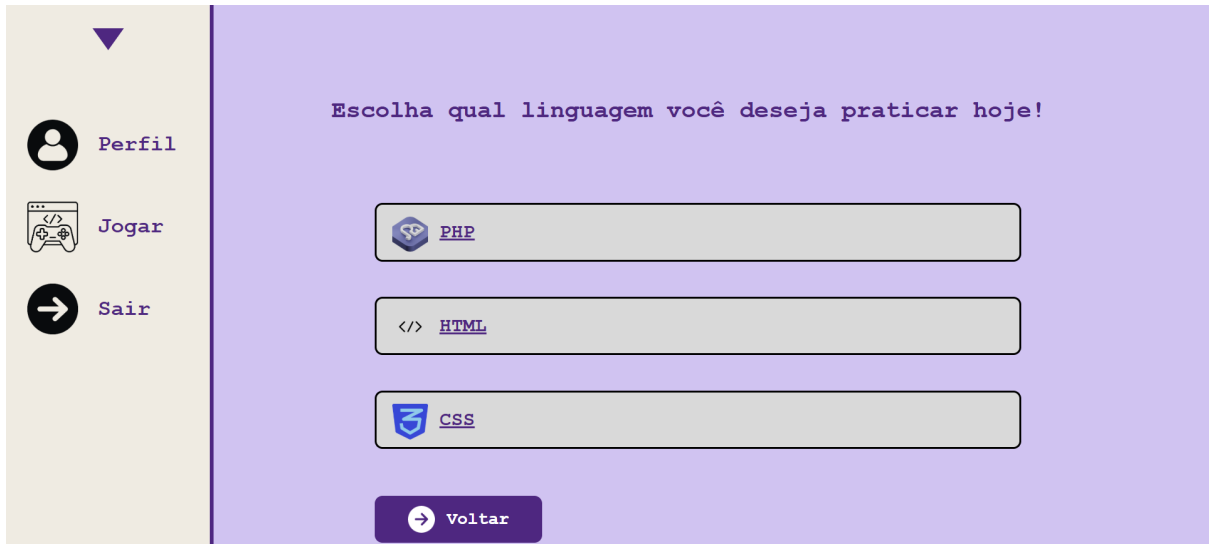
Figura 5 – Escolha dos Quizzes



Fonte: Autoria própria

A Figura 5 ilustra a tela de escolha dos quizzes, na qual o discente pode selecionar entre quizzes do sistema ou quizzes desenvolvidos pelos docentes.

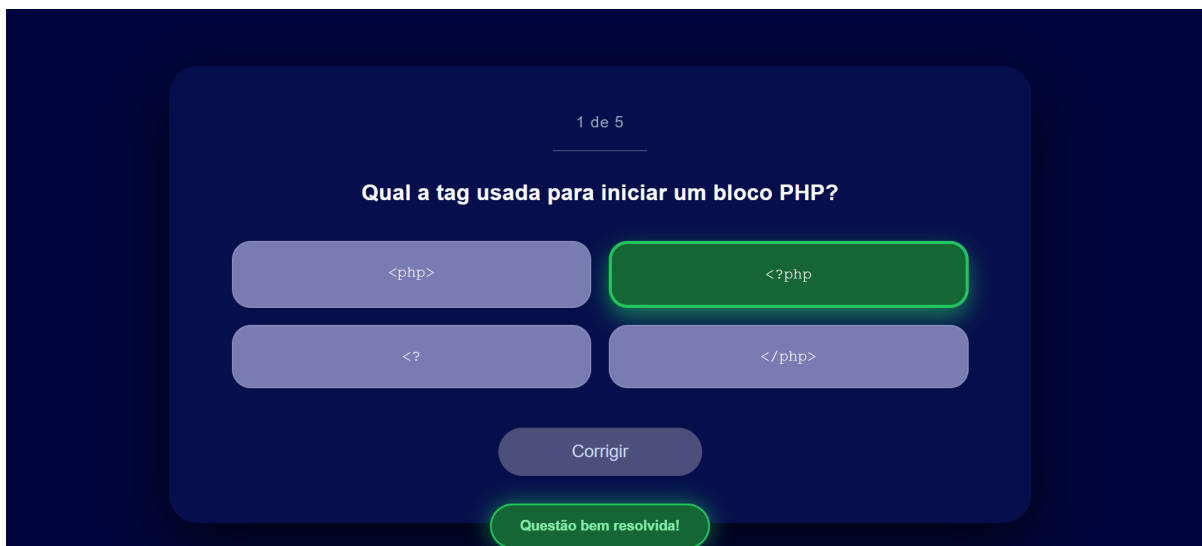
Figura 6 – Escolha da Linguagem



Fonte: Autoria própria

A Figura 6 apresenta a tela de escolha da linguagem de programação a ser utilizada nos quizzes do sistema.

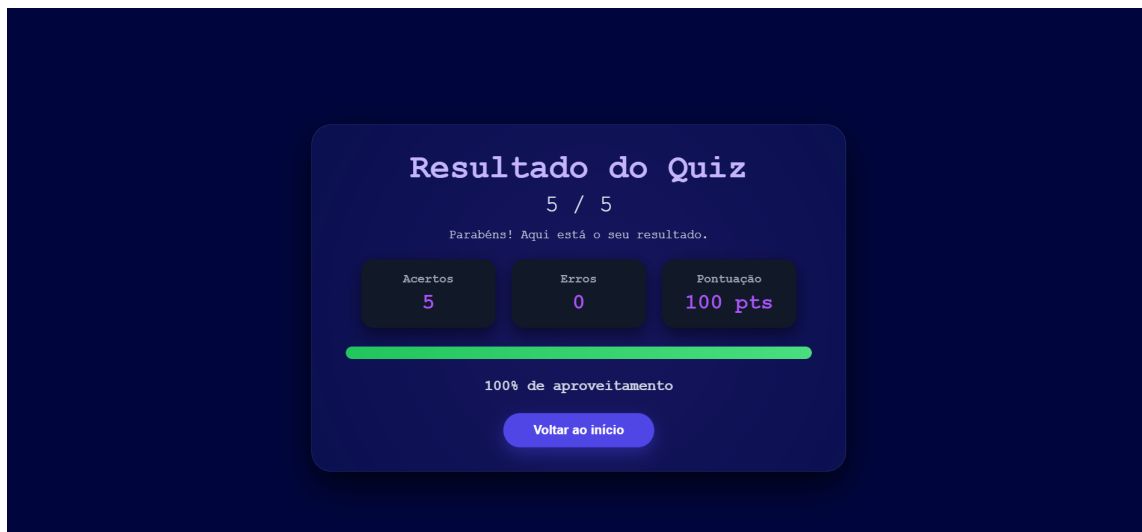
Figura 7 – Quizzes



Fonte: Autoria própria

A Figura 7 exibe a tela de realização dos quizzes, na qual o aluno responde às questões de múltipla escolha e recebe feedback sobre suas respostas.

Figura 8 – Resultado



Fonte: Autoria própria

A Figura 8 ilustra a tela de resultado do quiz, mostrando o desempenho do aluno.

Figura 9 – Quizzes do Docente



Fonte: Autoria própria

A Figura 9 mostra a tela exibida quando o discente opta por jogar os quizzes elaborados pelo docente.

Figura 10 – Perfil do Docente



Fonte: Autoria própria

A Figura 10 ilustra a tela de perfil do docente, na qual ele tem acesso às funcionalidades de criação de quizzes.

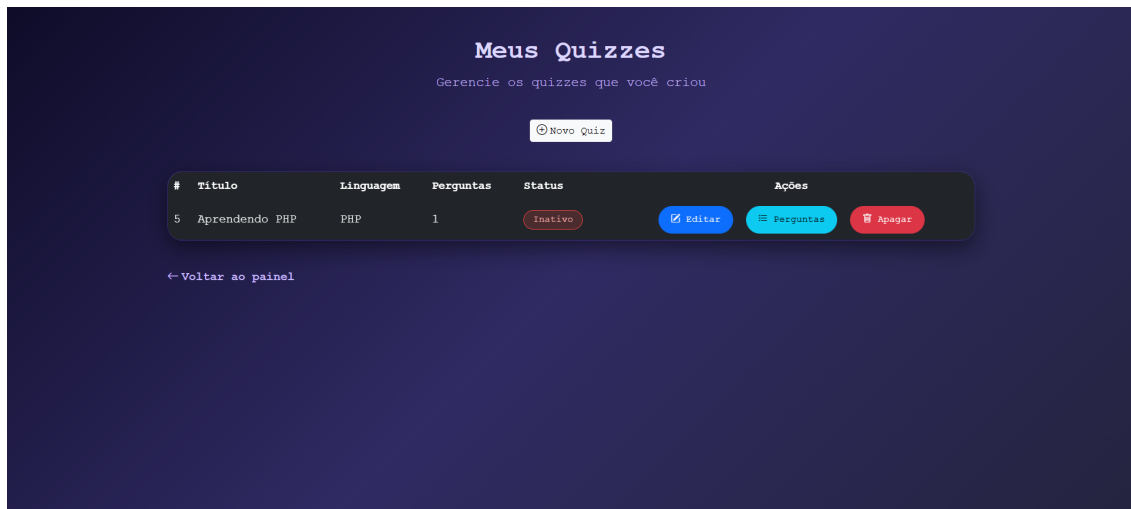
Figura 11 – Criar Quizzes



Fonte: Autoria própria

A Figura 11 exibe a tela principal utilizada pelo docente para a criação de seus quizzes.

Figura 12 – Gerenciar Quizzes



Fonte: Autoria própria

A Figura 12 demonstra como ocorre o gerenciamento dos quizzes pelo docente.

5 CONSIDERAÇÕES FINAIS

No presente trabalho, buscou-se promover o engajamento dos alunos no ensino de programação por meio do jogo **CodGame**, utilizando uma abordagem mais dinâmica e divertida no processo de aprendizagem. A proposta foi direcionada tanto aos alunos do ensino técnico em informática quanto àqueles que desejam iniciar seus estudos na área de programação, oferecendo um ambiente interativo que favorece a fixação dos conteúdos. Em relação ao primeiro objetivo específico, o sistema foi desenvolvido com a proposta de promover o engajamento dos alunos por meio da aplicação de elementos de gamificação, como desafios, pontuação e feedback imediato. No entanto, a validação desse engajamento por meio da aplicação do sistema em um grupo de alunos não foi realizada, sendo indicada como possibilidade para trabalhos futuros.

Quanto ao segundo objetivo, o desenvolvimento de jogos digitais voltados à fixação dos conteúdos de programação foi alcançado por meio da criação de quizzes organizados por linguagens de programação, permitindo ao aluno praticar conceitos de forma interativa. Essa abordagem possibilitou a aplicação prática dos conteúdos estudados, auxiliando na consolidação do aprendizado.

Por fim, o terceiro objetivo foi atendido com o desenvolvimento e implementação da solução utilizando as linguagens **PHP, HTML, Javascript e CSS**, aliadas a princípios de design educacional e gamificação. O sistema apresentou funcionalidade adequada e mostrou-se aplicável ao contexto do ensino técnico, oferecendo uma interface simples, intuitiva e alinhada à proposta educacional do projeto.

Durante o desenvolvimento do trabalho, foram enfrentados desafios relacionados à modelagem do sistema, à integração entre frontend e backend e à organização das funcionalidades. No entanto, esses desafios proporcionaram importantes aprendizados técnicos e contribuíram para o crescimento acadêmico e profissional.

Como possibilidades de trabalhos futuros, o CodGame pode ser expandido com a inclusão de novos tipos de jogos, mais linguagens de programação, níveis de

dificuldade, além de relatórios de desempenho para docentes. Dessa forma, conclui-se que o CodGame atende aos objetivos propostos, apresentando-se como uma ferramenta educacional capaz de contribuir para o ensino de programação de maneira interativa e motivadora.

REFERÊNCIAS

- FERRAZ JUNIOR, Wilton Moreira; HOMEM, Murillo Rodrigo Petruccelli; SANTOS, Carlos Henrique da Silva. O uso de games no processo ensino-aprendizagem. *In*: SANTOS, Carlos Henrique da Silva; SHIGUNOV NETO, Alexandre (Org.). **Informática aplicada à educação: algumas reflexões e práticas**. Rio de Janeiro: Ciência Moderna, 2017. 208-227.
- FROSI, Felipe Oviedo; JAQUES, Patricia A. Jogos digitais para o ensino de programação: uma revisão sistemática das pesquisas publicadas no Brasil entre 2015 e 2019. **Anais do XIX Simpósio Brasileiro de Jogos e Entretenimento Digital-SBGames**, Recife/PE, Brasil, p. 653-652, 2020.
- GENESIO, Nikolas Oliver Sales; MENEZES, Maria Clara Ribeiro de; ALMEIDA, João Victor Corrêa de; BOAVENTURA, Ana Paula Freitas Vilela; VALLE, Pedro Henrique Dias. Aprendendo Lógica de Programação Desenvolvendo Jogos Digitais: Um Relato de Experiência. *In*: WORKSHOP DE INFORMÁTICA NA ESCOLA (WIE), 29. , 2023, Passo Fundo/RS. **Anais [...]**. Porto Alegre: Sociedade Brasileira de Computação, 2023 . 12f. Disponível em: <https://sol.sbc.org.br/index.php/wie/article/view/26324> Acesso em: 10 jun. 2025.
- PASSOS, Leandro Santos; NOVAIS, Jonathan Willian Zangeski; VARELLA, Louyse Siqueira da Silva; OLIVEIRA, Allan Gonçalves de; ZANGESKI, Dahiane dos Santos Oliveira. Aplicação do Jogo Digital Code Combat no Ensino de Programação aos Alunos do Curso Médio Técnico em Informática. **Revista de Ensino, Educação e Ciências Humanas**, [S. l.], v. 21, n. 1, p. 112–116, 2020. Disponível em: <https://revistaensinoeeducacao.pgsscogna.com.br/ensino/article/view/7458>. Acesso em: 24 jul. 2025.
- SOUSA, Reudismam Rolim de; LEITE, Felipe. Torres. Usando gamificação no ensino de programação introdutória. **Brazilian Journal of Development**. [S. l.], v. 6, n. 6, p. 33338–33356, 2020. Disponível em: <https://ojs.brazilianjournals.com.br/ojs/index.php/BRJD/article/view/10995>. Acesso em: 10 jun. 2025.
- SOUZA, Franciely Alves de; FALCÃO, Taciana Pontual; MELLO, Rafael Ferreira. O Ensino de Programação na Educação Básica: Uma Revisão da Literatura. *In*: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO (SBIE), 32. , 2021, Online. **Anais [...]**. Porto Alegre: Sociedade Brasileira de Computação, 2021. p. 1265-1275. Disponível em: <https://sol.sbc.org.br/index.php/sbie/article/view/18148> Acesso em: 24 jul. 2025
- STEPHAN, João; OLIVEIRA, Alessandra; RENHE, Marcelo Caniato. O Uso de Jogos para Apoiar o Ensino e Aprendizagem de Programação. *In*: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO (SBIE), 31. , 2020, Online. **Anais [...]**. Porto Alegre: Sociedade Brasileira de Computação, 2020. Disponível em: <https://sol.sbc.org.br/index.php/sbie/article/view/12794> Acesso em: 24 jul. 2025.