



DESENVOLVIMENTO DE UM APLICATIVO EDUCACIONAL COM GAMIFICAÇÃO PARA O APOIO AO APRENDIZADO DE PROGRAMAÇÃO

Por

VICTOR LOPES DOS SANTOS

Trabalho de Conclusão de Curso



Instituto Federal de Educação, Ciência e Tecnologia da Bahia
ads@ifba.edu.br
<portal.ifba.edu.br/salvador>

SALVADOR/2025

Victor Lopes Dos Santos

**DESENVOLVIMENTO DE UM APLICATIVO EDUCACIONAL COM
GAMIFICAÇÃO PARA O APOIO AO APRENDIZADO DE
PROGRAMAÇÃO**

*Trabalho apresentado a Coordenação da Tecnologia em
Análise e Desenvolvimento de Sistemas do Instituto Federal
de Educação, Ciência e Tecnologia da Bahia como requisito
parcial para obtenção do grau de Tecnologia em Análise e
Desenvolvimento de Sistemas.*

Orientador: Jose Dihego Da Silva Oliveira

SALVADOR
2025

Resumo

As dificuldades enfrentadas pelos estudantes que ingressam nos cursos da área de tecnologia podem ser consequências de diversos fatores, entre eles a falta de engajamento e metodologias adequadas de ensino. A alta evasão nos cursos de tecnologia tem sido um dos principais desafios enfrentados pelas instituições de ensino superior, muitas vezes relacionada à dificuldade inicial de adaptação e à complexidade das disciplinas introdutórias de programação. Uma estratégia promissora para tornar esse processo mais atrativo é o uso da gamificação no ensino, que aplica elementos de jogos, para aumentar o engajamento e a motivação dos estudantes. Por meio de trilhas de aprendizado, o estudante pode avançar por etapas estruturadas que refletem seu progresso e o ajudam a visualizar sua evolução ao longo do curso. O uso de um sistema de pontos e conquistas reforça o engajamento, recompensando o esforço e estimulando a superação de desafios cada vez mais complexos. Além disso, a inclusão de exercícios práticos e feedback imediato permite que o estudante aprenda de forma ativa, corrigindo erros rapidamente e consolidando o conhecimento adquirido de maneira dinâmica. Essa abordagem busca transformar o aprendizado em uma experiência mais dinâmica e satisfatória, contribuindo para reduzir a evasão e melhorar o desempenho acadêmico.

Dessa forma, este trabalho tem como objetivo desenvolver um aplicativo de aprendizagem multi-plataforma centralizado na ideia de gamificação para auxiliar os estudantes no aprendizado dos fundamentos de programação, de modo mais interativo e envolvente. O aplicativo, desenvolvido para as plataformas Android e iOS utilizando o framework Flutter, abrangerá desde conceitos básicos até tópicos intermediários da programação. A gamificação será utilizada para promover a motivação e a prática contínua, sem substituir a educação formal, mas atuando como uma ferramenta complementar. Acredita-se que a aplicação contribua para a redução das taxas de reprovação em disciplinas de programação e para a melhor preparação dos estudantes frente às demandas do mercado de tecnologia. Ressalta-se, entretanto, que a mensuração desses resultados será objeto de estudos e validações em trabalhos futuros.

Palavras-chave: Ensino de programação. Gamificação. Aplicativos educacionais. Aplicativo mobile.

Abstract

The difficulties faced by students entering technology courses can stem from various factors, including a lack of engagement and inadequate teaching methodologies. High dropout rates in technology courses have been a major challenge for higher education institutions, often related to initial adaptation difficulties and the complexity of introductory programming subjects. A promising strategy to make this process more attractive is the use of gamification in education, which applies game elements to increase student engagement and motivation. Through learning paths, students can advance through structured stages that reflect their progress and help them visualize their evolution throughout the course. The use of a points and achievements system reinforces engagement, rewarding effort and encouraging the overcoming of increasingly complex challenges. Furthermore, the inclusion of practical exercises and immediate feedback allows students to learn actively, correcting errors quickly and consolidating acquired knowledge dynamically. This approach seeks to transform learning into a more dynamic and satisfying experience, contributing to reducing dropout rates and improving academic performance.

Therefore, this work aims to develop a cross-platform learning application centered on the idea of gamification to assist students in learning the fundamentals of programming in a more interactive and engaging way. The application, developed for Android and iOS platforms using the Flutter framework, will cover basic concepts to intermediate programming topics. Gamification will be used to promote motivation and continuous practice, without replacing formal education, but acting as a complementary tool. It is believed that the application will contribute to reducing failure rates in programming courses and to better preparing students for the demands of the technology market. It should be noted, however, that the measurement of these results will be the subject of studies and validations in future work.

Keywords: Teaching programming. Gamification. Educational apps. Mobile app.

Sumário

1	INTRODUÇÃO	6
1.1	JUSTIFICATIVA	8
1.2	OBJETIVOS	9
1.2.1	Objetivo Geral	9
1.2.2	Objetivos Específicos	9
2	FUNDAMENTAÇÃO TEÓRICA	10
2.1	Evasão nos Cursos de Computação	10
2.2	Ambientes Online de Aprendizado de Programação	11
2.3	Gamificação no Ensino	12
2.3.1	Gamificação Aplicada à Educação	12
2.3.2	Benefícios e Desafios	13
3	TRABALHOS RELACIONADOS	16
4	METODOLOGIA	17
5	DESENVOLVIMENTO	19
5.1	Requisitos Funcionais	19
5.2	Requisitos Não Funcionais	20
5.3	Modelagem do Sistema	21
5.3.1	Casos de Uso	22
5.3.2	Diagrama de Classes	23
5.4	Arquitetura do Sistema	24
5.5	Tecnologias Utilizadas	24
5.5.1	Flutter	24
5.5.2	Dart	25
5.5.3	Vaden	25
5.5.4	Supabase	25
5.5.5	Figma	25
5.5.6	Android Studio	25
5.5.7	Git	26
5.5.8	GitHub	26
5.6	Protótipo de Interfaces	26
5.7	Funcionalidades	29
5.7.1	Trilhas de Aprendizado	29
5.7.2	Flashcards	30

5.7.3	Jogos Propostos	30
5.7.4	Sistema de Pontos e Conquistas	31
6	CONSIDERAÇÕES FINAIS	33
	Referências	35

1

INTRODUÇÃO

Os cursos de graduação na área de tecnologia enfrentam há anos um desafio recorrente: as elevadas taxas de evasão e reprovação nas disciplinas introdutórias de programação. Esse fenômeno é amplamente discutido por MORAES; COSTA; SCHOLZ (2022), que destacam que, segundo dados do Censo do Ensino Superior de 2019, essas disciplinas apresentam índices médios de reprovação e evasão próximos de 33%, podendo ultrapassar 50% em alguns casos. As causas desse cenário incluem dificuldades de adaptação ao ambiente universitário, condições socioeconômicas, e principalmente a falta de familiaridade dos estudantes com metodologias voltadas à lógica e ao raciocínio computacional. Para muitos ingressantes, o primeiro contato com a lógica de programação representa uma barreira que pode gerar frustração e desmotivação, refletindo-se diretamente no desempenho acadêmico e na permanência no curso.

Nesse contexto, novas abordagens pedagógicas têm sido exploradas com o objetivo de tornar o processo de ensino-aprendizagem mais atrativo, como metodologias ativas que incluem a aprendizagem baseada em problemas e projetos, a instrução pelos pares, a sala de aula invertida, a aprendizagem baseada em equipes e o uso de estudos de caso, COSTA; AZEVEDO (2019). Além dessas abordagens, destaca-se a gamificação, uma estratégia que consiste na aplicação de elementos de jogos — como pontos, níveis, conquistas e recompensas — em ambientes não lúdicos, visando aumentar o envolvimento e a motivação dos participantes. Segundo ELSHIEKH; BUTGERIT (2017), a gamificação tem se mostrado uma ferramenta eficaz para estimular o interesse e a prática entre estudantes de programação, atuando como um meio de reforçar a motivação e promover o aprendizado ativo.

Sistemas baseados em desafios, missões e progressão por níveis estimulam a prática contínua e tornam o aprendizado mais dinâmico, criando uma experiência imersiva e significativa para o aluno, LI et al. (2013). De forma semelhante, pesquisas recentes destacam que a integração de elementos lúdicos, como recompensas e conquistas, pode contribuir para o fortalecimento da motivação e para o desenvolvimento de habilidades cognitivas importantes, como o raciocínio lógico e a resolução de problemas, LIMA et al. (2024). Além disso, a incorporação de elementos gamificados, através de recompensas, conquistas e tabela de classificação, cria um ambiente interativo que sustenta a motivação MOHTAR et al. (2023).

Este trabalho propõe o desenvolvimento de um aplicativo de aprendizagem multiplata-

forma baseado em gamificação, voltado ao ensino dos fundamentos de programação. O aplicativo, desenvolvido em Flutter, vai permitir uma experiência interativa que combina conceitos teóricos, exercícios práticos e elementos de jogos. A solução busca estimular o aprendizado autônomo, aumentar a motivação dos estudantes e apoiar a educação formal por meio de uma ferramenta complementar. Acredita-se que a aplicação possa favorecer o engajamento e a integração dos estudantes, o que, de forma indireta, pode contribuir para uma melhor adaptação ao curso.

Ressalta-se que o escopo deste trabalho está concentrado na análise do problema e no projeto e desenvolvimento de um protótipo funcional do aplicativo, não contemplando, neste momento, a implementação em ambiente real de sala de aula e a coleta sistemática de feedback de usuários. O professor orientador atuou como stakeholder da solução, exercendo o papel de validador e fornecendo sugestões.

O capítulo 2 apresenta a justificativa e relevância do estudo. O capítulo 3 descreve os objetivos geral, enquanto o capítulo 4 reúne a fundamentação teórica, abordando evasão em cursos de computação, ambientes online de aprendizado e gamificação na educação. O capítulo 5 detalha o desenvolvimento do sistema, incluindo requisitos, modelagem, tecnologias e principais funcionalidades. Por fim, o capítulo 6 traz as conclusões.

1.1 JUSTIFICATIVA

Estudos de revisão recentes indicam que as dificuldades mais recorrentes no aprendizado de programação introdutória concentram-se na compreensão da sintaxe e da semântica das linguagens, nos conceitos fundamentais de programação, na capacidade de resolver problemas e nos processos cognitivos associados, bem como na identificação e correção de tentativas incorretas. Esses achados têm sido consistentemente observados na literatura, reforçando a natureza multifatorial das dificuldades enfrentadas por estudantes iniciantes, HENKLAIN; CARVALHO; FEITOSA (2025).

Nesse cenário, diferentes estratégias de ensino-aprendizagem vêm sendo investigadas com o objetivo de minimizar essas dificuldades. A revisão sistemática realizada por HOLANDA; PAIVA FREIRE; SILVA COUTINHO (2019) evidencia que abordagens tradicionais, centradas apenas na exposição teórica, têm se mostrado insuficientes para promover uma aprendizagem efetiva em programação introdutória. Nesse sentido, estudos apontam que o uso de jogos educacionais pode contribuir significativamente para o aumento da motivação, do engajamento e da retenção do conteúdo, uma vez que combinam interatividade, *feedback* imediato, desafios e objetivos claros, estimulando processos cognitivos semelhantes aos exigidos na resolução de problemas e no desenvolvimento de algoritmos, RAPKIEWICZ et al. (2007).

O mapeamento sistemático conduzido por MORAES; COSTA; SCHOLZ (2022) reforça essa constatação ao identificar uma predominância de propostas voltadas ao aumento da motivação e do engajamento discente, incluindo abordagens gamificadas e ambientes digitais de aprendizagem. Apesar disso, o estudo aponta que muitas dessas soluções ainda enfrentam limitações quanto à escalabilidade, à personalização do ensino e à adoção em larga escala, indicando a necessidade de novas propostas que integrem aspectos pedagógicos e tecnológicos de forma mais consistente.

Diante desse cenário, a utilização de um ambiente interativo e gamificado busca contribuir para a redução das dificuldades conceituais e comportamentais identificadas na literatura, oferecendo uma alternativa complementar ao ensino tradicional. Assim, espera-se que a solução proposta possa favorecer o engajamento contínuo dos estudantes, estimular a prática ativa e auxiliar no processo de ensino-aprendizagem, alinhando-se às lacunas e recomendações evidenciadas nos estudos analisados.

Esta pesquisa justifica-se, portanto, pela necessidade de preencher tais lacunas, propondo uma solução que integre, de forma sistemática e teoricamente fundamentada, elementos de gamificação a princípios pedagógicos baseados em evidências. O aplicativo desenvolvido organiza o conteúdo em trilhas progressivas, oferece diferentes formatos de atividades interativas, alinhados aos tipos de dificuldade reportados por estudantes iniciantes, e incorpora um sistema de flashcards que estimula a recuperação ativa e a consolidação do conhecimento.

1.2 OBJETIVOS

O presente trabalho tem como propósito central investigar os fatores que contribuem para as dificuldades enfrentadas por estudantes no aprendizado dos fundamentos de programação, bem como desenvolver uma solução tecnológica que auxilie na melhoria desse processo de ensino, buscando aumentar o engajamento dos estudantes e reduzir as dificuldades iniciais que frequentemente resultam em reprovação ou evasão. Pretende-se, por meio da aplicação de mecânicas de gamificação em um aplicativo móvel, criar um ambiente de aprendizagem mais motivador, dinâmico e eficaz, capaz de proporcionar prática contínua, feedback imediato e estímulo à autonomia do estudante.

1.2.1 Objetivo Geral

Investigar as dificuldades enfrentadas por estudantes no processo de aprendizagem dos fundamentos de programação e projetar, implementar e avaliar um aplicativo móvel gamificado como ferramenta de apoio educacional, design de interação e gamificação, com o objetivo de melhorar o engajamento, a retenção de conteúdo e o desempenho dos estudantes, contribuindo para a redução dos índices de reprovação e evasão.

1.2.2 Objetivos Específicos

- Identificar dificuldades iniciais comuns no aprendizado de programação.
- Estruturar o conteúdo de programação em módulos e níveis progressivos, permitindo aos estudantes aprender conceitos de forma incremental.
- Desenvolver um protótipo funcional de aplicativo móvel gamificado, aplicando práticas de design de interface.

2

FUNDAMENTAÇÃO TEÓRICA

Guiado por uma revisão da literatura, a fundamentação teórica deste trabalho tem como objetivo investigar o ensino de programação, as taxas de evasão em ciência da computação e o papel das tecnologias online e baseadas em gamificação. Seu objetivo é compreender os determinantes do engajamento, da motivação e do desempenho dos estudantes e, conseqüentemente, identificar estratégias pedagógicas que possam superar obstáculos recorrentes na aprendizagem. Nesse contexto, são abordados estudos nacionais e internacionais que sustentam a relevância da adoção de abordagens inovadoras no ensino de programação, especialmente em disciplinas introdutórias, que exercem papel determinante na permanência e no sucesso acadêmico dos estudantes.

2.1 Evasão nos Cursos de Computação

Diversos estudos apontam que os índices de evasão e reprovação permanecem elevados ao longo dos anos, especialmente nas disciplinas introdutórias de programação, que concentram dificuldades significativas para estudantes ingressantes MORAES; COSTA; SCHOLZ (2022). Investigações internacionais indicam que, embora haja variações entre instituições e contextos, disciplinas iniciais de programação apresentam taxas de reprovação relevantes, reforçando a percepção de que o ensino introdutório de programação constitui um obstáculo recorrente na formação em computação BENNEDSEN; CASPERSEN (2019).

Entre os fatores associados à evasão, a literatura destaca dificuldades relacionadas à compreensão da lógica de programação, ao nível de abstração exigido para a resolução de problemas e ao domínio da sintaxe das linguagens utilizadas. O mapeamento sistemático realizado por MORAES; COSTA; SCHOLZ (2022) evidencia que essas disciplinas constam entre aquelas com maiores índices de reprovação, contribuindo diretamente para a desmotivação e o abandono do curso. Estudos como o de BENNEDSEN; CASPERSEN (2019) reforçam que, mesmo após mais de uma década de pesquisas e intervenções pedagógicas, o ensino de programação introdutória continua sendo percebido como desafiador, mantendo taxas de insucesso relativamente estáveis ao longo do tempo.

A revisão sistemática conduzida por HOLANDA; PAIVA FREIRE; SILVA COUTINHO

(2019) aponta que abordagens tradicionais de ensino, baseadas predominantemente em aulas expositivas e exercícios pouco contextualizados, tendem a dificultar o engajamento dos estudantes iniciantes. A ausência de estratégias que promovam participação ativa, feedback contínuo e acompanhamento do progresso pode intensificar sentimentos de frustração e baixa autoconfiança, elementos frequentemente associados ao abandono dos cursos. Diante desse cenário, a literatura ressalta a necessidade de estratégias de ensino-aprendizagem que integrem aspectos cognitivos, pedagógicos e motivacionais, especialmente nas disciplinas iniciais, que desempenham papel decisivo na permanência e no sucesso acadêmico nos cursos de computação.

2.2 Ambientes Online de Aprendizado de Programação

Os ambientes online de aprendizado de programação, como *freeCodeCamp*, *Udemy* e *Coursera*, têm se consolidado como uma das principais estratégias para democratizar o acesso ao ensino de computação, especialmente diante da expansão da internet e da popularização dos dispositivos móveis. Esses ambientes combinam recursos multimídia, exercícios práticos, sistemas de avaliação automática e interação social, permitindo que o estudante desenvolva competências técnicas de forma gradual, interativa e autônoma.

A evolução desses ambientes está diretamente relacionada ao avanço dos dispositivos móveis e à mudança nos hábitos de consumo de tecnologia. MAHMOUD; DYER (2008) já destacavam, em meados dos anos 2000, o crescimento do uso de dispositivos móveis sem fio no contexto educacional. Posteriormente, MAHMOUD; POPOWICZ (2010) ampliaram essa perspectiva ao propor o uso de aplicações móveis como artefato pedagógica no ensino introdutório de programação. Os autores argumentam que, embora o ambiente móvel apresente restrições técnicas, como menor capacidade de processamento e limitações de interface, ele oferece vantagens pedagógicas relevantes, pois os conceitos de lógica, estruturas de controle e organização de código são integralmente transferíveis para outras plataformas. Além disso, a familiaridade dos estudantes com smartphones contribui para o aumento do engajamento e da motivação. Na prática, os autores sugerem atividades como o desenvolvimento de *quizzes*, calculadoras e editores de texto simples, incluindo elementos visuais como barras de progresso, proporcionando *feedback* imediato ao aluno, MAHMOUD; POPOWICZ (2010).

Outro aspecto essencial dos ambientes online modernos é a acessibilidade. PAMBUDI; SUBAGYO (2018) destacam que aplicações móveis possuem a vantagem de não serem limitadas por tempo ou localização, podendo ser acessadas a qualquer momento e em qualquer lugar, característica que amplia significativamente as oportunidades de aprendizagem. Esse tipo de plataforma favorece o aprendizado contínuo, fragmentado em pequenas sessões, mais compatíveis com a rotina dos estudantes contemporâneos.

Além da dimensão tecnológica, os ambientes online de aprendizagem de programação têm incorporado progressivamente elementos de interação social. LI et al. (2013) descrevem um ambiente de aprendizagem colaborativa baseado em rede social, no qual os estudantes

podem personalizar seus perfis, compartilhar interesses e interagir academicamente. A pesquisa destaca que a combinação de aprendizagem online com mecanismos inspirados em redes sociais potencializa o senso de pertencimento e aumenta o envolvimento dos estudantes nas atividades educacionais.

2.3 Gamificação no Ensino

A gamificação tem sido definida como um processo de aprimoramento de serviços por meio da incorporação de elementos motivacionais, com o objetivo de provocar experiências semelhantes às de jogos e influenciar comportamentos desejados. Essa concepção é apresentada por HAMARI; KOIVISTO; SARSA (2014), a partir da definição proposta originalmente por HUOTARI; HAMARI (2012), destaca que a gamificação vai além do uso superficial de elementos lúdicos, estando relacionada à criação de experiências significativas que estimulem o engajamento e a participação ativa. No contexto educacional, essa abordagem tem sido amplamente investigada devido ao seu potencial de tornar o processo de aprendizagem mais envolvente e motivador.

De acordo com a revisão de literatura realizada por CAPONETTO et al. (2014), a gamificação passou a ser explorada de forma mais consistente na educação a partir de 2011, consolidando-se como uma estratégia distinta de abordagens como o *game-based learning*, ao não exigir o desenvolvimento de jogos completos. A literatura aponta que a principal motivação para sua adoção no ensino está associada ao estímulo ao engajamento discente, sendo frequentemente empregados elementos como sistemas de pontuação, níveis, recompensas e rankings para favorecer comportamentos como persistência, colaboração e autonomia. Nessa perspectiva, a gamificação é compreendida como uma estratégia de design educacional voltada à motivação e ao envolvimento contínuo dos estudantes LI et al. (2013).

2.3.1 Gamificação Aplicada à Educação

A aplicação da gamificação na educação tem se mostrado particularmente eficaz no ensino de programação e áreas correlatas. HUITEMA; WONG (2025) apresentam um estudo de caso sobre o ensino de criptografia em programas de cibersegurança, demonstrando como a gamificação pode ser utilizada para tornar conteúdos altamente abstratos mais acessíveis. Os autores destacam que o avanço de ferramentas de inteligência artificial e a crescente complexidade do cenário digital aumentaram a demanda por educação em segurança da informação. Nesse âmbito, os autores exploram o uso de jogos educacionais inspirados em práticas como *Capture the Flag* (CTF), cifras clássicas como a cifra de César e ambientes interativos como o *Codex*. Os resultados indicam que a utilização de níveis, recompensas e feedback imediato contribui significativamente para o aumento do engajamento e para a consolidação do aprendizado, especialmente por meio da visualização do progresso do estudante ao longo das atividades.

LI et al. (2013) também demonstram como a gamificação pode ser integrada a ambientes educacionais online por meio de redes sociais educacionais. Em sua proposta, os estudantes

acumulam pontos a partir de suas interações e atividades, sendo a pontuação calculada com base em uma curva sigmoide transformada, o que permite uma distribuição mais equilibrada dos resultados. Além disso, o uso de rankings públicos (leaderboards) incentiva a competição saudável e fortalece o engajamento coletivo.

Integração de elementos como pontos, distintivos (*badges*) e rankings (*leaderboards*) permite que os estudantes acompanhem seu progresso ao longo do processo de aprendizagem, promovendo *feedback* constante e a percepção de avanço, ZAKIAH; DERIS; HIDAYAH (2023). Dessa forma, a gamificação no ensino não implica transformar o conteúdo educacional em um jogo completo, mas sim incorporar, de maneira estratégica, características dos jogos para tornar o processo de aprendizagem mais dinâmico, participativo e menos passivo.

No ensino mediado por aplicativos educacionais, o Método Duolingo, descrito por PAMBUDI; SUBAGYO (2018), é fundamentado em evidências científicas da aprendizagem e organizado em cinco pilares: aprendizagem ativa (*learn by doing*), personalização do processo de ensino conforme o desempenho do aluno, foco em conteúdos essenciais alinhados a padrões educacionais, uso de elementos de gamificação para manutenção da motivação e a promoção de uma experiência de aprendizagem emocionalmente positiva. Segundo os autores, a combinação entre engajamento, desafio adequado e feedback contínuo contribui para a eficácia do processo de aprendizagem em ambientes digitais.

A gamificação tem sido utilizada como estratégia para tornar ambientes educacionais mais interativos, por meio da incorporação de elementos como recompensas, conquistas, *rankings* e narrativas, os quais contribuem para estimular o engajamento e sustentar a motivação dos estudantes. SMIRANI; YAMANI (2024) conduzem uma análise abrangente sobre o impacto das técnicas de gamificação no engajamento, na motivação e na retenção do conhecimento, buscando preencher lacunas existentes na literatura. Para isso, os autores adotam métricas quantitativas, como o tempo de permanência na plataforma, as taxas de conclusão dos cursos e os resultados obtidos pelos estudantes em avaliações, a fim de mensurar os efeitos da gamificação em ambientes de aprendizagem digitais.

Essas evidências reforçam que a gamificação aplicada à educação não se limita a um recurso estético, mas constitui uma estratégia pedagógica estruturada, capaz de transformar a relação do estudante com o conteúdo, especialmente em áreas tradicionalmente consideradas complexas, como a programação.

2.3.2 Benefícios e Desafios

Os benefícios da gamificação no ensino de programação são amplamente documentados na literatura. ZAKIAH; DERIS; HIDAYAH (2023), apontam que a utilização de mecânicas como pontos, níveis, distintivos e *rankings* promove o aumento da motivação, do engajamento e da retenção dos estudantes. Além disso, os autores destacam que a gamificação contribui para a melhoria dos resultados de aprendizagem, uma vez que estimula a prática contínua e a superação

de desafios progressivos.

OMUNA; GOULART (2023), reforçam que, embora uma grande variedade de softwares educacionais disponíveis, a gamificação se destaca pela sua capacidade de promover engajamento efetivo, uma vez que transforma o estudante em agente ativo do próprio aprendizado. Essa característica é especialmente relevante no ensino de programação, onde a prática constante é fundamental para o desenvolvimento do raciocínio lógico e da autonomia na resolução de problemas.

Os resultados do estudo conduzido por SMIRANI; YAMANI (2024) indicam que a utilização de elementos de gamificação, como sistemas de pontos, *badges*, *rankings* e desafios, pode estimular os estudantes a se envolverem de forma mais ativa com os conteúdos educacionais e a permanecerem por mais tempo em plataformas de ensino a distância. Os autores também destacam que a inclusão de elementos lúdicos, como desafios e interações envolventes, contribui para tornar os módulos de aprendizagem mais agradáveis e interessantes, sendo o fator prazer um componente relevante para o fortalecimento do engajamento, da motivação e da percepção de eficácia do aprendizado. De modo geral, os achados apontam que a gamificação exerce impacto positivo sobre a retenção do conhecimento, a motivação e o aproveitamento das experiências educacionais em ambientes digitais.

Entretanto, a literatura também evidencia desafios relevantes associados à gamificação, especialmente quando seus elementos são implementados de forma inadequada. LI et al. (2013) alertam que a competição excessiva pode afetar negativamente estudantes com menor desempenho, além de existir o risco de que o foco exagerado em recompensas externas comprometa a motivação intrínseca ao longo do tempo. Resultados semelhantes são observados no estudo de SMIRANI; YAMANI (2024), que identificam que ambientes altamente competitivos tendem a reduzir o engajamento dos aprendizes, possivelmente em função do aumento da pressão e do estresse, desviando a atenção do conteúdo educacional. Embora a competitividade apresente efeitos positivos sobre a motivação, o prazer e a percepção de eficácia da aprendizagem, os autores apontam uma relação negativa com o engajamento, bem como uma leve percepção de superficialidade ou distração associada à gamificação. Esses achados reforçam a necessidade de um equilíbrio cuidadoso no design pedagógico de ambientes gamificados, de modo a potencializar seus benefícios sem comprometer a qualidade do processo de aprendizagem.

HUITEMA; WONG (2025), ressaltam que, especialmente em áreas sensíveis como a cibersegurança, a gamificação deve ser cuidadosamente planejada para não simplificar excessivamente conteúdos críticos. O equilíbrio entre ludicidade e rigor conceitual é essencial para garantir que o aprendizado seja não apenas envolvente, mas também tecnicamente consistente.

Por fim, estudos como os de MAHMOUD; POPOWICZ (2010), evidenciam que o uso de plataformas móveis, embora ampliem o acesso e o engajamento, impõem limitações técnicas que devem ser consideradas no desenho das atividades, como restrições de interface e processamento. Assim, a gamificação no ensino se apresenta como uma abordagem com alto potencial, mas que exige planejamento pedagógico cuidadoso, alinhamento com objetivos educacionais e constante

avaliação de seus impactos.

3

TRABALHOS RELACIONADOS

O uso de aplicativos móveis e técnicas de gamificação no ensino de programação tem sido explorado como alternativa para aumentar o engajamento e reduzir as dificuldades enfrentadas por estudantes iniciantes. Entre os trabalhos relacionados, destacam-se iniciativas acadêmicas e plataformas comerciais, como LIMA et al. (2024), SANTOS et al. (2023) e *Coddy*¹. Eles utilizam abordagens interativas para apoiar o processo de ensino-aprendizagem.

- ***GamifyCode***: por meio de um aplicativo móvel e aplicação Web, a proposta do utilizar a gamificação no ensino de programação Web, atendendo estudantes com diferentes níveis de conhecimento.
- ***ProgLing***: propõe um aplicativo educacional focado no ensino de linguagens de programação, com ênfase na organização de conteúdos e exercícios práticos, apresentando menor exploração de mecânicas gamificadas.
- ***Coddy***: oferece uma solução comercial com desafios progressivos e feedback imediato, porém sem foco específico em currículos formais ou personalização para contextos educacionais.

Em comparação, o presente trabalho propõe o desenvolvimento de um protótipo gamificado voltado aos fundamentos de programação, com foco na estrutura pedagógica, progressão por níveis e apoio ao ensino formal, diferenciando-se pela abordagem modular e pela integração sistemática de elementos de gamificação.

Tabela 3.1: Comparação entre trabalhos relacionados

Recursos	Null Academy	Coddy	GamifyCode	ProgLing
Foco em fundamentos de programação	Sim	Parcial	Parcial	Parcial
Uso sistemático de gamificação	Sim	Sim	Sim	Parcial
Progressão estruturada por níveis	Sim	Sim	Sim	Não
Integração com ensino formal	Sim	Não	Sim	Parcial
Uso de flashcards para revisão	Sim	Não	Não	Não

¹<https://coddy.tech/>

4

METODOLOGIA

A metodologia deste trabalho tem como objetivo descrever de forma detalhada os procedimentos adotados para o desenvolvimento do aplicativo educacional gamificado proposto, possibilitando a compreensão do percurso metodológico seguido e garantindo a reprodutibilidade.

Quanto à finalidade, esta pesquisa caracteriza-se como uma pesquisa aplicada, uma vez que tem como propósito o desenvolvimento de uma solução tecnológica concreta, voltada ao apoio ao ensino dos fundamentos de programação, com potencial de aplicação prática no contexto educacional.

Em relação ao método científico, o trabalho fundamenta-se no método dedutivo, partindo de conceitos gerais apresentados na literatura, para a definição de uma solução específica, materializada no desenvolvimento de um protótipo de aplicativo educacional.

O desenvolvimento da pesquisa foi estruturado em quatro etapas principais:

- **Revisão bibliográfica:** Inicialmente, foi realizada uma revisão da literatura com o objetivo de identificar as principais dificuldades enfrentadas por estudantes iniciantes no aprendizado de programação, bem como analisar abordagens pedagógicas e estratégias de gamificação aplicadas ao contexto educacional. Essa etapa serviu como base conceitual para a definição do escopo, dos conteúdos e das mecânicas de gamificação incorporadas à solução proposta.
- **Prototipagem da interface:** A partir do embasamento teórico, foi conduzida a prototipagem da interface do aplicativo. Foram definidos os fluxos de navegação, a organização dos módulos de aprendizado e a disposição dos elementos visuais, buscando uma experiência intuitiva para o usuário.
- **Desenvolvimento da aplicação frontend:** Com base nos protótipos definidos, realizou-se o desenvolvimento da interface do aplicativo utilizando o framework Flutter, permitindo a criação de uma aplicação multiplataforma.
- **Implementação do backend e integração dos serviços:** Por fim, foi realizada a implementação do backend da aplicação, responsável pelo gerenciamento de dados, autenticação de usuários e persistência das informações.

Ao final dessas etapas, obteve-se um protótipo funcional de aplicativo educacional gamificado. Ressalta-se que não foram realizadas avaliações empíricas com estudantes nem coleta sistemática de feedback, limitando o escopo do trabalho ao projeto, prototipagem e desenvolvimento da solução.

5

DESENVOLVIMENTO

Este capítulo descreve o desenvolvimento do aplicativo educacional gamificado proposto neste trabalho. Inicialmente, são apresentados os requisitos funcionais e não funcionais do sistema, que definem, respectivamente, as funcionalidades disponibilizadas aos estudantes e as restrições técnicas e de qualidade que orientaram sua implementação. Em seguida, é abordada a modelagem do sistema, responsável por representar de forma abstrata a estrutura e o comportamento da aplicação.

O objetivo deste capítulo é detalhar as principais decisões técnicas adotadas durante o desenvolvimento, evidenciando como a solução proposta atende aos objetivos pedagógicos definidos, bem como aos requisitos de usabilidade, desempenho e escalabilidade esperados de um aplicativo móvel voltado ao apoio ao aprendizado de programação.

O processo de desenvolvimento do sistema adotou o modelo de **desenvolvimento iterativo**, no qual o aplicativo foi construído e evoluído por meio de incrementos sucessivos de funcionalidades. Cada incremento contemplou as etapas de levantamento de requisitos, modelagem, implementação e testes manuais, abrangendo testes *ad hoc*, permitindo a entrega progressiva de partes funcionais do sistema.

5.1 Requisitos Funcionais

Os requisitos funcionais descrevem as funcionalidades que o sistema deve oferecer aos estudantes, especificando os serviços e comportamentos esperados da aplicação. Esses requisitos foram definidos com base nos objetivos educacionais do projeto e nas características de sistemas de aprendizagem gamificados.

A Tabela 5.1 apresenta os requisitos funcionais identificados para o sistema.

Tabela 5.1: Requisitos Funcionais do Sistema

ID	Descrição
RF01	Permitir o cadastro e autenticação de usuários por meio de e-mail e senha.
RF02	Permitir a visualização e edição do perfil do usuário, incluindo informações básicas e estatísticas de progresso.
RF03	Disponibilizar trilhas de aprendizado organizadas de forma progressiva.
RF04	Disponibilizar jogos educativos vinculados às lições, com diferentes formatos de atividades interativas.
RF05	Registrar o desempenho do usuário em cada sessão de atividades, contabilizando tentativas e acertos.
RF06	Calcular a pontuação, nota conceitual e experiência obtida ao final de cada sessão.
RF07	Atualizar automaticamente o nível do usuário conforme o acúmulo de XP (<i>Experience Points</i>).
RF08	Gerenciar o sistema de conquistas, desbloqueando recompensas conforme critérios definidos.
RF09	Disponibilizar <i>flashcards</i> para revisão de conteúdos, incluindo decks padrão e personalizados.
RF10	Permite que o usuário visualize um <i>ranking</i> (<i>leaderboard</i>) baseado no seu desempenho.
RF11	Permitir o acompanhamento do progresso do usuário ao longo das trilhas de aprendizado.

5.2 Requisitos Não Funcionais

Os requisitos não funcionais estabelecem critérios relacionados à qualidade, desempenho, segurança e usabilidade do sistema. Eles são fundamentais para garantir que a aplicação opere de forma eficiente, confiável e adequada às expectativas dos usuários.

A Tabela 5.2 apresenta os principais requisitos não funcionais definidos para o sistema.

Tabela 5.2: Requisitos Não Funcionais do Sistema

ID	Descrição
RNF01	O sistema deve ser desenvolvido como um aplicativo móvel multi-plataforma, compatível com Android e iOS.
RNF02	A interface do usuário deve ser intuitiva, abstrata, subjetiva, responsiva e de fácil navegação.
RNF03	O sistema deve apresentar tempos de resposta adequados, garantindo fluidez na execução das atividades.
RNF04	Os dados dos usuários devem ser armazenados de forma segura, assegurando confidencialidade e integridade.
RNF05	O sistema deve ser escalável, permitindo a inclusão de novas trilhas e funcionalidades.
RNF06	O código-fonte deve seguir boas práticas de organização, facilitando manutenção e evolução do sistema.

5.3 Modelagem do Sistema

A modelagem do sistema tem como objetivo representar, de forma abstrata e estruturada, a organização e o funcionamento do aplicativo educacional desenvolvido. Essa etapa possibilita a compreensão das interações entre usuários e sistema, das funcionalidades oferecidas e da relação entre os componentes internos, garantindo o atendimento aos requisitos funcionais e não funcionais estabelecidos, SOMMERVILLE (2011).

Neste trabalho, a modelagem foi realizada por meio de diagramas de casos de uso, diagrama de classes e do Modelo C4, BROWN (2023). Os casos de uso descrevem as interações entre o usuário e o sistema sob a perspectiva funcional, o diagrama de classes apresenta a estrutura estática da aplicação, evidenciando classes, atributos e relacionamentos, e o Modelo C4 permite visualizar a arquitetura do sistema em um nível mais alto de abstração, destacando seus principais componentes e formas de comunicação.

5.3.1 Casos de Uso

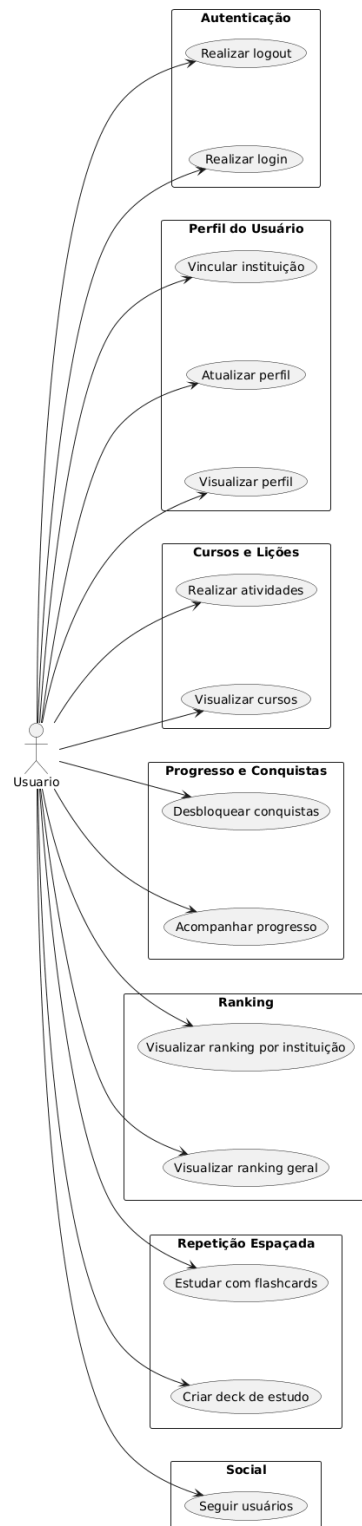


Figura 5.1: Diagrama de casos de uso do sistema, elaborado com PlantUML pelo autor (2025)

5.3.2 Diagrama de Classes

O diagrama de classes apresenta a estrutura do sistema, evidenciando as principais classes e seus atributos, bem como os relacionamentos existentes entre elas.

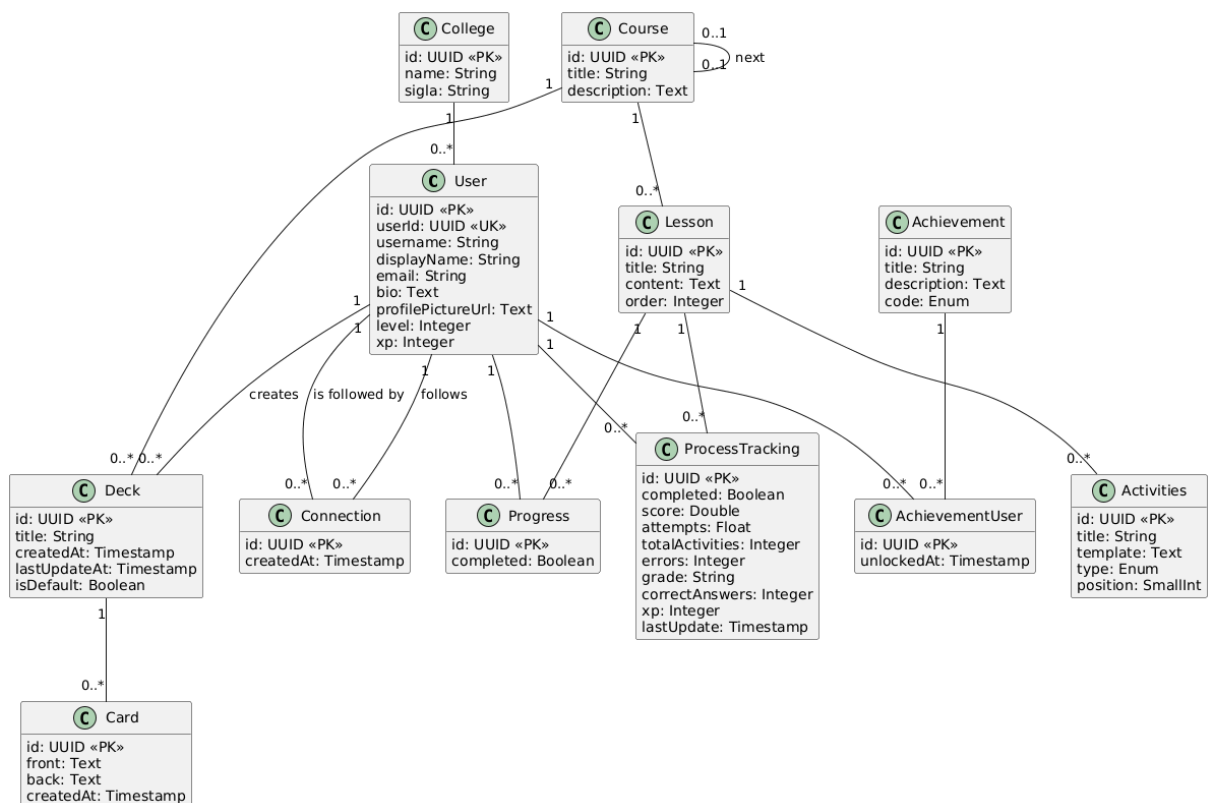


Figura 5.2: Diagrama de classes do sistema, elaborado com PlantUML pelo autor (2025)

5.4 Arquitetura do Sistema

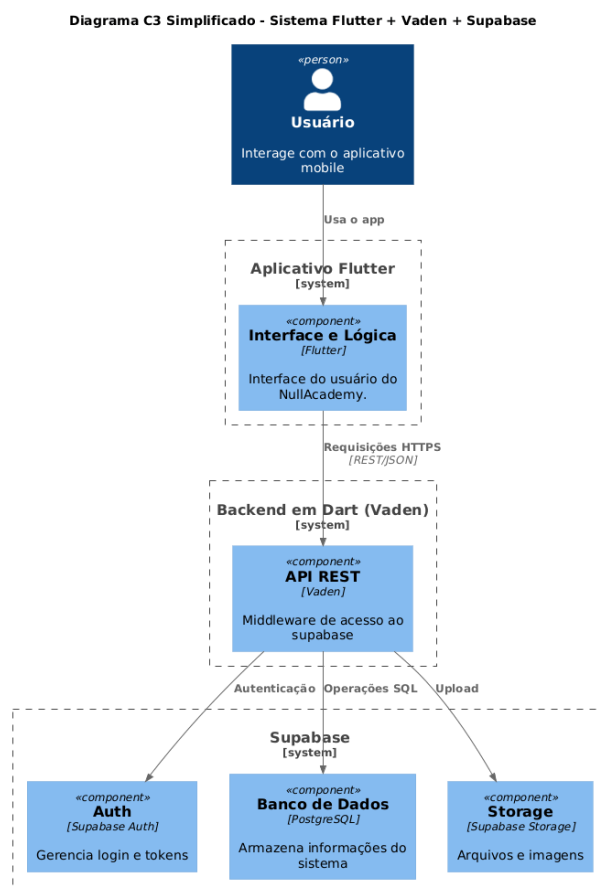


Figura 5.3: Diagrama de Container — Modelo C4.

5.5 Tecnologias Utilizadas

Esta seção apresenta as principais tecnologias empregadas no desenvolvimento da solução proposta, descrevendo suas características e finalidades. A escolha das tecnologias utilizadas foi justificada pelas características que favorecem o desenvolvimento. A seleção das tecnologias considerou aspectos como desempenho, escalabilidade, facilidade de manutenção, compatibilidade com dispositivos móveis e aderência aos objetivos educacionais do projeto. Optou-se pela adoção de tecnologias recentes e modernas, selecionadas em função de seu potencial técnico.

5.5.1 Flutter

O *Flutter* é um *framework* de código aberto desenvolvido pela Google para a criação de aplicações multiplataforma utilizando uma única base de código. Ele permite o desenvolvimento para Android, iOS, Web e Desktop Flutter Team (2025). A escolha do Flutter deve-se à velocidade de desenvolvimento proporcionada pelo recurso de *hot reload* e à vasta coleção de componentes visuais personalizáveis (*widgets*), que facilitam a construção de interfaces modernas e responsivas.

5.5.2 Dart

A linguagem *Dart* é a base do ecossistema Flutter e foi utilizada tanto no desenvolvimento da interface quanto na lógica de negócio da aplicação. Por ser uma linguagem fortemente tipada, orientada a objetos e com suporte a programação assíncrona, o Dart permite um código mais seguro e performático Dart Team (2025).

5.5.3 Vaden

O Vaden é um *framework* de *backend* desenvolvido em Dart e criado pela comunidade brasileira Flutterando. Inspirado em soluções consolidadas como o *Spring Boot* (Java) e o *NestJS* (TypeScript), o Vaden oferece uma estrutura moderna e completa para construção de APIs e serviços escaláveis. Por permitir o uso do mesmo ecossistema de desenvolvimento do Flutter, o Vaden possibilita uma abordagem *full-stack* em Dart, reduzindo a complexidade de integração entre o *frontend* e o *backend* Vaden Framework (2025).

5.5.4 Supabase

O *Supabase* foi utilizado como plataforma de banco de dados e autenticação. Baseado em tecnologias de código aberto, como o *PostgreSQL*, o Supabase oferece uma infraestrutura escalável e segura, além de fornecer funcionalidades como autenticação, armazenamento de arquivos e funções *serverless*. Sua adoção simplificou o gerenciamento do banco de dados e permitiu maior agilidade no desenvolvimento, eliminando a necessidade de configurações complexas de servidores, Supabase (2025).

5.5.5 Figma

O *Figma* foi a ferramenta escolhida para o design e prototipagem das interfaces do usuário. Trata-se de uma plataforma colaborativa baseada em nuvem, que permite a criação de protótipos interativos e facilita a comunicação entre desenvolvedores e designers. O uso do Figma possibilitou a definição prévia de fluxos de navegação e componentes visuais, garantindo maior consistência e coerência estética ao longo do projeto, Figma Inc. (2025).

5.5.6 Android Studio

Desenvolvido pela Google, o Android Studio fornece um ambiente de execução estável e compatível com o *Flutter SDK*, possibilitando a verificação de desempenho, layout e responsividade durante o processo de desenvolvimento. O *Android Studio* foi utilizado principalmente pelo seu emulador integrado, que permitiu testar e validar o funcionamento da aplicação em diferentes dispositivos e versões do sistema operacional Android Google (2025).

5.5.7 Git

O *Git* foi a ferramenta de controle de versão adotada no projeto, permitindo o acompanhamento de alterações no código-fonte de forma segura e colaborativa. Por meio do *Git*, foi possível gerenciar diferentes versões do sistema e garantir a rastreabilidade das modificações realizadas durante o desenvolvimento. Essa prática contribuiu para a organização, manutenção e confiabilidade do código, Git Project (2025).

5.5.8 GitHub

O *GitHub* foi a plataforma escolhida para o armazenamento remoto do repositório e controle de versão do projeto. Ela oferece recursos de versionamento integrados ao *Git*. GitHub (2025).

5.6 Protótipo de Interfaces

Segundo SOMMERVILLE (2011), um protótipo consiste em uma versão inicial de um sistema de software, utilizada para demonstrar conceitos, experimentar alternativas de design e compreender melhor o problema e suas possíveis soluções.

No desenvolvimento do aplicativo educacional com gamificação para o aprendizado de programação, foram criados protótipos das telas utilizando o *Figma*, com foco em organizar o conteúdo didático de forma clara e engajante. As telas simulam os principais recursos do aplicativo, incluindo desafios de programação, níveis de progresso, pontuação e recompensas, proporcionando uma visão prévia da experiência do usuário.

Durante o processo de prototipagem, as telas passaram por diversas revisões, garantindo que o fluxo de navegação fosse intuitivo e que os elementos de gamificação estivessem distribuídos de forma coerente, refletindo a dinâmica do aprendizado proposta pelo aplicativo.

O fluxo de criação do usuário representa a etapa inicial de interação do estudante com o aplicativo. Esse processo foi projetado com o objetivo de promover uma entrada simples, rápida e acessível, reduzindo barreiras de uso e incentivando a continuidade no ambiente digital.

Ao iniciar o aplicativo pela primeira vez, o usuário é direcionado para a tela de autenticação, onde fornece informações básicas, como nome, e-mail e senha. Após a conclusão da autenticação, o usuário é direcionado para a tela de cadastro, onde ele completa as suas informações, como *username*, biografia e uma instituição de ensino, caso ele seja vinculado a uma, como mostra na figura 5.4.

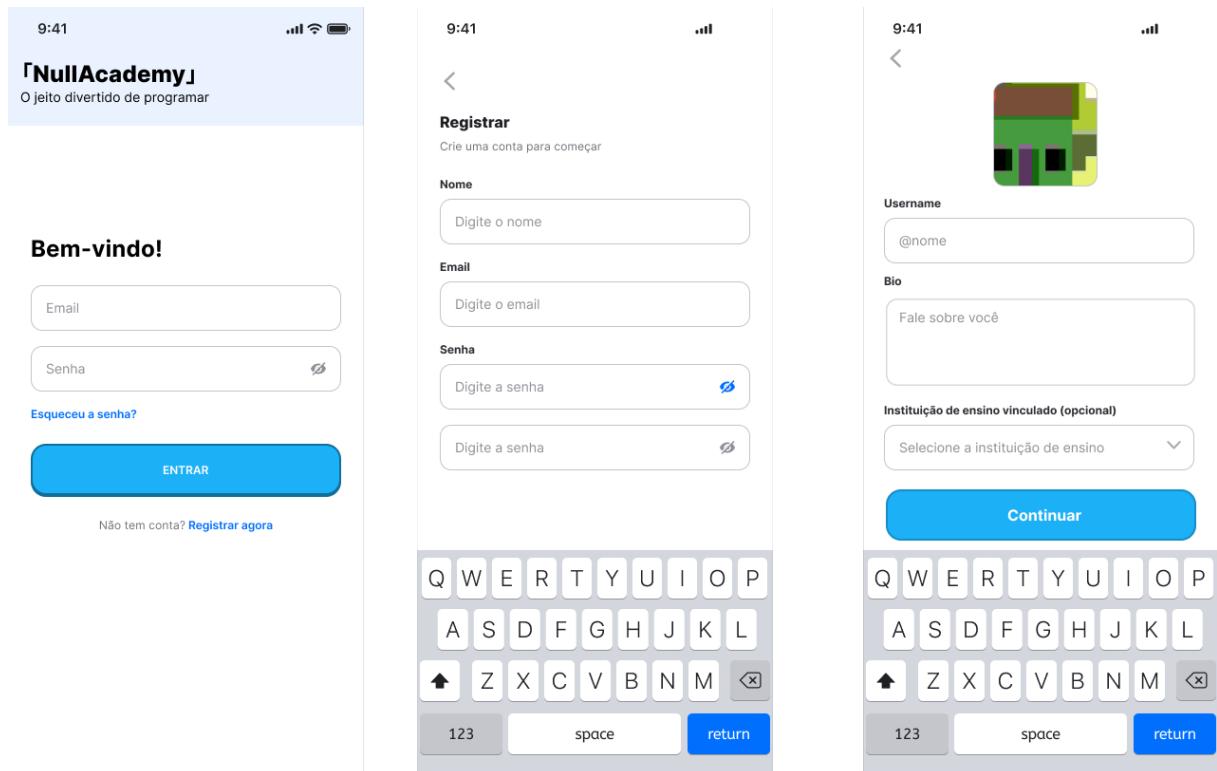


Figura 5.4: Protótipo das Telas (Fluxo de Autenticação).

Depois disso, o estudante é conduzido a uma tela introdutória que apresenta as orientações iniciais, preparando-o para utilizar os recursos gamificados e acessar os conteúdos de aprendizagem. Após a introdução, o usuário é direcionado para a tela inicial do aplicativo, onde pode visualizar suas conquistas em andamento e acompanhar a trilha de aprendizado atual. A partir da tela inicial, o usuário acessa as trilhas de ensino, que estão organizadas de maneira progressiva para facilitar o aprendizado gradual, as estatísticas e *ranking*, e por fim, seu perfil com suas informações, figura 5.5.

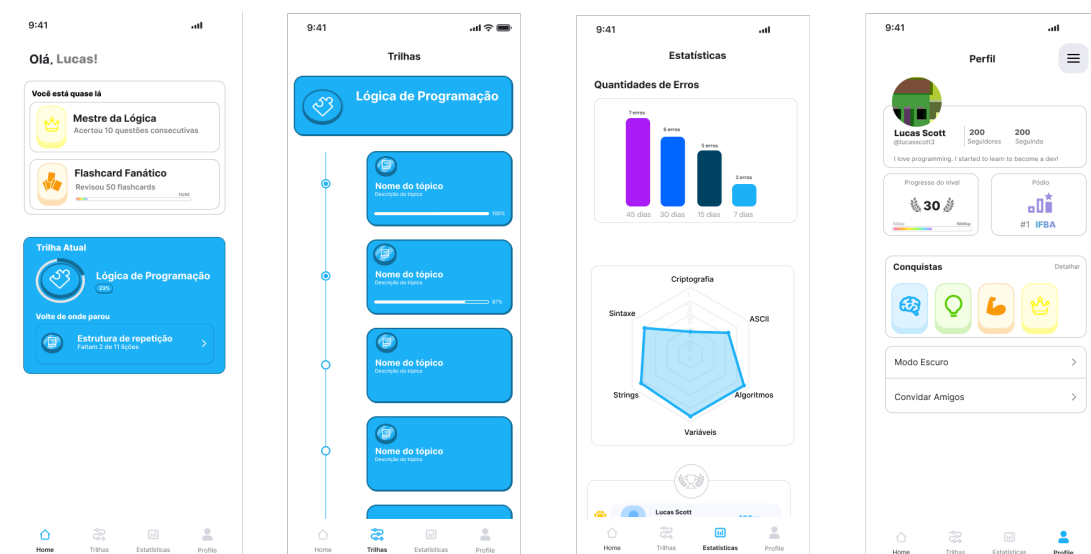


Figura 5.5: Protótipo das Telas (Fluxo das páginas iniciais).

Dentro de cada trilha, o usuário encontra uma variedade de jogos educativos, como é mostrado na figura 5.6. As atividades incluem questões de múltipla escolha ou verdadeiro e falso, respostas curtas que estimulam a lembrança ativa, ordenação por arrastar elementos, preenchimento de lacunas e desafios de prever resultados. Cada formato reforça os conteúdos estudados de maneiras diferentes, ajudando a fixar conceitos, exercitar lógica e tornar a prática mais intuitiva e motivadora.

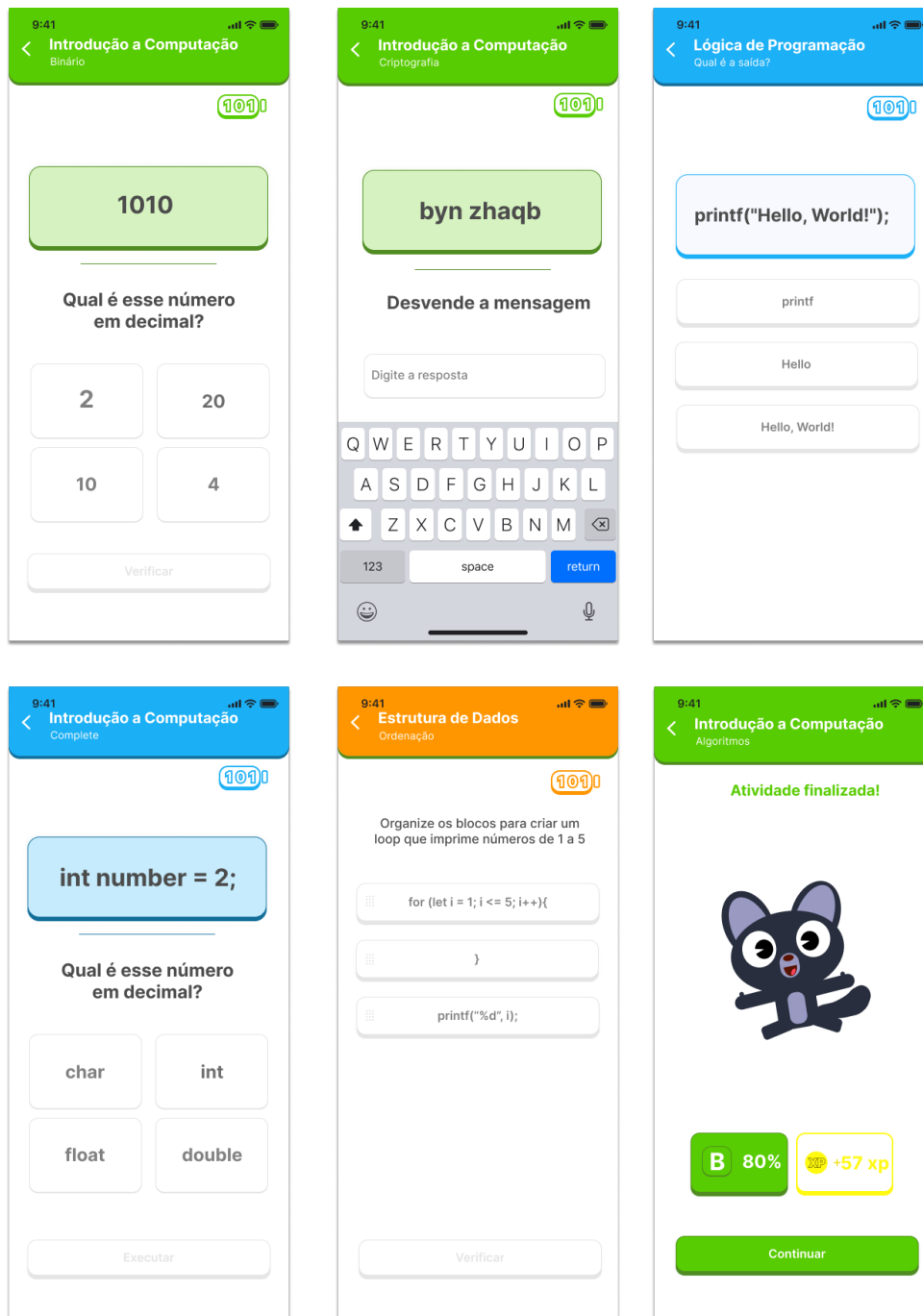


Figura 5.6: Protótipo das Telas (Fluxo de Jogos).

Além de cada trilha conter atividades, ela possui um fluxo de flashcards, abaixo na figura 5.7, que reforça os conteúdos apresentados por meio de revisões rápidas e interativas. Esse re-

curso contribui para fortalecer a memorização e tornar o aprendizado mais dinâmico, permitindo que o usuário avance no próprio ritmo enquanto consolida os conhecimentos adquiridos.

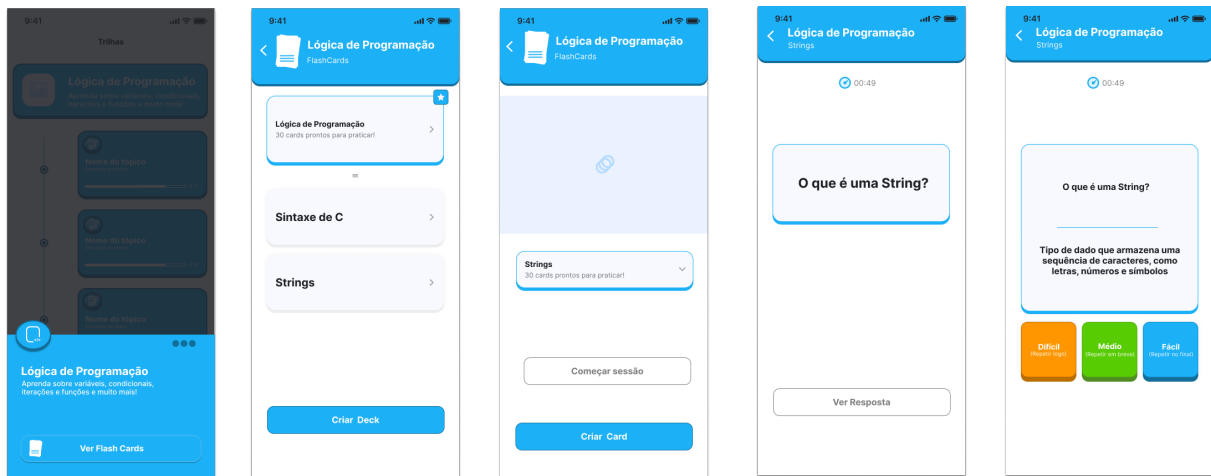


Figura 5.7: Protótipo das Telas (Fluxo de Flashcard).

5.7 Funcionalidades

Esta seção descreve as principais funcionalidades implementadas no sistema, destacando como cada recurso contribui para o processo de aprendizagem, engajamento do usuário e acompanhamento do progresso.

5.7.1 Trilhas de Aprendizado

As Trilhas de Aprendizado constituem o eixo central do sistema, sendo responsáveis por organizar e estruturar os conteúdos educacionais de forma progressiva e orientada. Essa abordagem permite que o usuário desenvolva seus conhecimentos gradualmente, respeitando uma sequência lógica de aprendizado e reduzindo lacunas conceituais. Cada trilha é composta por cursos e lições organizados em níveis, nos quais o avanço ocorre conforme a conclusão das etapas anteriores.

Atualmente, o sistema prevê duas trilhas principais de aprendizado:

- **Introdução à Computação:** Trilha voltada à apresentação dos fundamentos da computação, abordando desde o contexto histórico da área até conceitos essenciais para a formação inicial do usuário. Entre os temas contemplados estão a evolução dos computadores, componentes de hardware, princípios da lógica booleana e noções introdutórias de criptografia, proporcionando uma base sólida para a compreensão dos demais conteúdos da plataforma.
- **Lógica de Programação:** Trilha dedicada ao desenvolvimento do raciocínio lógico e do pensamento computacional, apresentando conceitos fundamentais utilizados na construção de algoritmos. Essa trilha aborda temas como variáveis, funções e

estruturas de controle, preparando o usuário para a resolução de problemas e para o aprendizado posterior de linguagens de programação.

5.7.2 Flashcards

A funcionalidade de flashcards fundamenta-se em princípios da psicologia cognitiva, especialmente na recuperação ativa da memória (*active recall*). Estudos demonstram que a prática de recuperar informações previamente aprendidas promove maior consolidação do conhecimento e melhora significativamente a retenção a longo prazo quando comparada a métodos passivos de estudo, como a releitura KARPICKE; ROEDIGER III (2008).

Os *flashcards* são estruturados no formato de perguntas e respostas, promovendo a participação ativa do usuário no processo de aprendizagem. Ao interagir com os cartões, o usuário é incentivado a refletir sobre o conteúdo antes de visualizar a resposta, o que contribui para o fortalecimento das conexões cognitivas e para a identificação de lacunas no aprendizado.

No sistema proposto, cada trilha de aprendizado possui um *deck* de flashcards padrão, disponibilizado automaticamente para os usuários como material de apoio aos conteúdos estudados. Esses *decks* padrão são elaborados de acordo com os temas abordados em cada trilha, permitindo a revisão sistemática dos conceitos essenciais. Além dos *decks* padrão, o sistema permite que o usuário crie seus próprios *decks* de *flashcards*, os quais podem ser vinculados a uma trilha específica.

5.7.3 Jogos Propostos

Os jogos propostos no sistema têm como objetivo tornar o processo de aprendizagem mais dinâmico e envolvente, permitindo que o usuário aplique, na prática, os conteúdos estudados ao longo das trilhas de aprendizado. Por meio de atividades interativas, o sistema busca estimular o raciocínio lógico, a atenção e a participação ativa do estudante, contribuindo para uma aprendizagem mais significativa.

O sistema disponibiliza diferentes tipos de jogos, cada um desenvolvido para atender a objetivos educacionais específicos, conforme descrito a seguir:

- **Múltipla escolha / Verdadeiro ou Falso (*alternatives*):** Atividades rápidas e objetivas, nas quais o estudante seleciona a alternativa correta entre opções apresentadas. Esse formato é utilizado principalmente para revisar conceitos, verificar a compreensão do conteúdo e identificar pontos que precisam de maior atenção.
- **Resposta aberta de uma palavra (*written*):** Nesse tipo de atividade, o usuário deve digitar uma palavra ou pequeno conceito relacionado ao conteúdo estudado. Essa abordagem incentiva o aluno a lembrar da informação sem o auxílio de alternativas, favorecendo a fixação dos conceitos mais importantes.

- **Ordenação (*sorting*):** Atividade em que o estudante organiza elementos na ordem correta por meio da interação de arrastar e soltar. Esse formato é especialmente útil para trabalhar sequências lógicas, etapas de processos e hierarquias, de forma visual e intuitiva.
- **Preenchimento de lacunas (*complete*):** O aluno completa partes faltantes de um texto ou trecho de código. Essa atividade auxilia na fixação de termos, sintaxe e estruturas recorrentes, servindo como apoio para o aprendizado da escrita correta do conteúdo.
- **Previsão de resultado (*output*):** Atividade em que o usuário analisa um código ou operação e indica qual será o resultado final. Esse tipo de exercício estimula o raciocínio lógico e a compreensão do funcionamento dos comandos, ajudando o estudante a entender melhor como o código é executado.

5.7.4 Sistema de Pontos e Conquistas

O Sistema de Pontos e Conquistas tem como principal objetivo incentivar a participação contínua do usuário, valorizando seu desempenho e progresso ao longo das atividades propostas no sistema. Esse mecanismo utiliza conceitos de gamificação para tornar a experiência de aprendizagem mais motivadora, ao mesmo tempo em que fornece indicadores claros sobre o desempenho do estudante. Cada atividade do sistema é composta por cinco jogos, apresentados ao usuário em forma de uma sessão de atividade. Durante essa sessão, o sistema monitora diferentes métricas relacionadas ao desempenho do usuário, como número de tentativas, acertos e tempo total de execução. O usuário inicia cada sessão com um número limitado de tentativas.

A pontuação da sessão é baseada em um sistema de experiência (XP). Cada sessão possui um valor máximo de XP, que é progressivamente reduzido conforme a ocorrência de tentativas incorretas durante a execução das atividades. O sistema também calcula uma porcentagem de aproveitamento e atribui uma nota conceitual, variando de A a F, de acordo com o número de tentativas incorretas registradas. Ao longo da sessão, o sistema registra o número total de atividades concluídas, a quantidade de respostas corretas, as tentativas incorretas cometidas e o total de tentativas realizadas. Essas informações são utilizadas para gerar um resumo final da atividade, que representa o desempenho do usuário naquela lição específica.

Além da atualização de XP e nível, o sistema também verifica se o usuário atingiu critérios específicos para o desbloqueio de conquistas. As conquistas funcionam como recompensas simbólicas por marcos importantes, como conclusão de atividades, bom desempenho ou constância de uso, reforçando positivamente o engajamento do usuário.

Tabela 5.3: Critérios de Pontuação, Nota e Avaliação do Desempenho

Critério	Descrição	Resultado
XP Máximo por sessão	Valor máximo de experiência possível	200 XP
Perda de XP por tentativa incorreta	Redução aplicada a cada tentativa incorreta cometida	20 XP
Tentativas iniciais	Número máximo de tentativas por sessão	5 tentativas
Porcentagem final	Cálculo baseado no XP obtido	0% a 100%
Nota A	Nenhuma tentativa incorreta	Excelente desempenho
Nota B	1 tentativa incorreta	Muito bom desempenho
Nota C	2 tentativas incorretas	Bom desempenho
Nota D	3 tentativas incorretas	Desempenho regular
Nota E	4 a 5 tentativas incorretas	Desempenho insuficiente
Nota F	6 ou mais tentativas	Atividade não satisfatória
Progressão de nível	XP acumulado atinge o limite definido	Aumento de nível
Conquistas	Verificação de critérios específicos	Desbloqueio de conquistas

6

CONSIDERAÇÕES FINAIS

Este trabalho teve como ponto de partida o problema recorrente das elevadas taxas de reprovação e evasão nas disciplinas introdutórias de programação, frequentemente associadas à falta de engajamento, à dificuldade na compreensão dos conceitos iniciais e à adoção de metodologias pouco motivadoras. Diante desse cenário, o objetivo geral consistiu em desenvolver um aplicativo educacional gamificado, multiplataforma, que atuasse como ferramenta complementar ao ensino formal, buscando tornar o processo de aprendizagem mais interativo, motivador e acessível para estudantes ingressantes na área de tecnologia.

Como principal resultado, foi possível projetar, modelar e prototipar um aplicativo educacional com foco no aprendizado dos fundamentos de programação. A solução desenvolvida contempla funcionalidades como trilhas de aprendizado progressivas, jogos educativos interativos, sistema de flashcards para revisão de conteúdos e um sistema de pontos, níveis e conquistas baseado em experiência (XP). Além disso, foram definidos requisitos funcionais e não funcionais, elaborados diagramas de modelagem do sistema e proposta uma arquitetura que integra frontend em Flutter, backend em Vaden e serviços de persistência e autenticação via Supabase. Esses elementos demonstram a viabilidade técnica e conceitual da solução proposta.

No que é tocante ao problema de pesquisa, a solução desenvolvida se propõe a focar diretamente o desengajamento dos estudantes ao incorporar elementos de gamificação alinhados a objetivos pedagógicos claros. Apesar dos avanços alcançados, este trabalho apresenta algumas limitações. Embora o aplicativo tenha sido desenvolvido e estruturado, as funcionalidades relacionadas a jogos mais interativos ainda não foram implementadas em sua totalidade. Recursos como animações avançadas, escrita livre de código, compilação e execução automática dos programas diretamente no aplicativo ainda se encontram como propostas conceituais. Além disso, não foi realizada, nesta etapa, uma validação pedagógica com estudantes reais, nem estudos experimentais que permitam mensurar de forma quantitativa o impacto da aplicação no aprendizado ou no engajamento dos estudantes.

Atualmente, o protótipo desenvolvido contempla um conjunto de 53 atividades distribuídas entre os módulos propostos, as quais ainda demandam validação pedagógica e técnica mais aprofundada. Além disso, torna-se necessária a ampliação da base de atividades, de modo a cobrir um espectro mais abrangente de conteúdos e níveis de complexidade.

Como sugestões para trabalhos futuros, recomenda-se a implementação completa dos jogos interativos, incluindo mecanismos de escrita de código, compilação e execução automática, possibilitando uma experiência mais próxima da prática real de programação. Outra perspectiva relevante consiste na integração do aplicativo com soluções baseadas em inteligência artificial, com o objetivo de fornecer feedback mais personalizado e contextualizado aos estudantes, auxiliando na identificação de dificuldades, na explicação de conceitos e na orientação do processo de aprendizagem. Ademais, proponha-se a realização de estudos de usabilidade com estudantes ingressantes, a fim de avaliar o impacto do aplicativo no desempenho acadêmico, no engajamento e na redução das taxas de reprovação e evasão.

- BENNEDSEN, J.; CASPERSEN, M. E. Failure rates in introductory programming: 12 years later. **ACM inroads**, [S.l.], v.10, n.2, p.30–36, 2019.
- BROWN, S. The C4 model for visualising software architecture. URL <https://c4model.com>, [S.l.], 2023.
- CAPONETTO, I. et al. Gamification and education: a literature review. In: EUROPEAN CONFERENCE ON GAMES BASED LEARNING. **Anais...** [S.l.: s.n.], 2014. v.1, n.1, p.50–57.
- COSTA, G. M. C.; AZEVEDO, G. X. de. Metodologias ativas: novas tendências para potencializar o processo de ensino aprendizagem. **Iniciação & Formação Docente**, [S.l.], v.6, n.2, p.285–a, 2019.
- Dart Team. **Dart Language Overview**. Acesso em: nov. 2025.
- ELSHIEKH, R.; BUTGERIT, L. Using gamification to teach students programming concepts. **Open Access Library Journal**, [S.l.], v.4, n.8, p.1–7, 2017.
- Figma Inc. **Figma Help Center**. Acesso em: nov. 2025.
- Flutter Team. **Flutter Documentation**. Acesso em: nov. 2025.
- Git Project. **Git Documentation**. Acesso em: nov. 2025.
- GitHub. **GitHub Documentation**. Acesso em: nov. 2025.
- Google. **Android Studio Documentation**. Acesso em: nov. 2025.
- HAMARI, J.; KOIVISTO, J.; SARSA, H. Does gamification work?—a literature review of empirical studies on gamification. In: HAWAII INTERNATIONAL CONFERENCE ON SYSTEM SCIENCES, 2014. **Anais...** [S.l.: s.n.], 2014. p.3025–3034.
- HENKLAIN, M. H. O.; CARVALHO, L. S. G. de; FEITOSA, E. L. Dificuldades no aprendizado de programação: um exame baseado em mapeamento sistemático da literatura e interpretação analítico-comportamental. **Revista Brasileira de Informática na Educação**, [S.l.], v.33, p.1484–1521, 2025.
- HOLANDA, W. D. de; PAIVA FREIRE, L. de; SILVA COUTINHO, J. C. da. Estratégias de ensino-aprendizagem de programação introdutória no ensino superior: uma revisão sistemática da literatura. **Revista Novas Tecnologias na Educação**, [S.l.], v.17, n.1, p.527–536, 2019.
- HUITEMA, D.; WONG, A. A Case Study in Gamification for a Cybersecurity Education Program: a game for cryptography. **arXiv preprint arXiv:2502.06706**, [S.l.], 2025.
- HUOTARI, K.; HAMARI, J. Defining gamification: a service marketing perspective. In: PROCEEDING OF THE 16TH INTERNATIONAL ACADEMIC MINDTREK CONFERENCE. **Anais...** [S.l.: s.n.], 2012. p.17–22.

- KARPICKE, J. D.; ROEDIGER III, H. L. The critical importance of retrieval for learning. **science**, [S.l.], v.319, n.5865, p.966–968, 2008.
- LI, C. et al. Engaging computer science students through gamification in an online social network based collaborative learning environment. **International Journal of Information and Education Technology**, [S.l.], v.3, n.1, p.72, 2013.
- LIMA, K. D. B. et al. GAMIFYCODE: um estudo sobre o aprendizado de programação por meio de aplicativos móveis gamificados. **RECIMA21 - Revista Científica Multidisciplinar**, [S.l.], v.5, n.3, p.e534992, 2024.
- MAHMOUD, Q. H.; DYER, A. Mobile devices in an introductory programming course. **Computer**, [S.l.], v.41, n.6, p.106–108, 2008.
- MAHMOUD, Q. H.; POPOWICZ, P. A mobile application development approach to teaching introductory programming. In: IEEE FRONTIERS IN EDUCATION CONFERENCE (FIE), 2010. **Anais...** [S.l.: s.n.], 2010. p.T4F–1.
- MOHTAR, S. et al. The usability evaluation on mobile learning apps with gamification for middle-aged women. **Education and Information Technologies**, [S.l.], v.28, n.1, p.1189–1210, 2023.
- MORAES, R. P. de; COSTA, V. F. da; SCHOLZ, R. E. P. Mapeamento sistemático do ensino introdutório de programação nos ensinos técnico e superior no Brasil. **Revista Brasileira de Informática na Educação**, [S.l.], v.30, p.628–647, 2022.
- OMUNA, A. M.; GOULART, T. C. K. A gamificação como alternativa na aprendizagem de programação. **Revista Terra & Cultura: Cadernos de Ensino e Pesquisa**, [S.l.], v.39, n.76, p.e2875, 2023.
- PAMBUDI, T.; SUBAGYO, A. Development of Gamification in Learning. In: INTERNATIONAL CONFERENCE ON EDUCATION AND E-LEARNING, 2. **Proceedings...** IEEE, 2018. p.45–52.
- RAPKIEWICZ, C. E. et al. Estratégias pedagógicas no ensino de algoritmos e programação associadas ao uso de jogos educacionais. **RENOTE: revista novas tecnologias na educação [recurso eletrônico]. Porto Alegre, RS**, [S.l.], 2007.
- SANTOS, L. T. et al. PROGLING:: aplicativo educacional para ensino de linguagens de programação. **Revista Científica UNIFAGOC-Multidisciplinar**, [S.l.], v.8, n.3, 2023.
- SMIRANI, L.; YAMANI, H. Analysing the impact of gamification techniques on enhancing learner engagement, motivation, and knowledge retention: a structural equation modelling approach. **Electronic Journal of e-Learning**, [S.l.], v.22, n.9, p.111–124, 2024.
- SOMMERVILLE, I. Software engineering 9th Edition. **ISBN-10**, [S.l.], v.137035152, p.18, 2011.
- Supabase. **Supabase Documentation**. Acesso em: nov. 2025.
- Vaden Framework. **Documentação Oficial do Vaden**. Acesso em: nov. 2025.
- ZAKIAH, N.; DERIS, K.; HIDAYAH, W. **A review of gamification for learning programming language**. 2023.