



SISTEMA PARA APRENDIZADO DE LÓGICA DE PROGRAMAÇÃO

José A. M. Fiel¹; Paulo C. dos SANTOS²

RESUMO

Em síntese, o projeto propõe um sistema web que permite o desenvolvimento e aprendizado do pensamento computacional de forma intuitiva e eficiente em vista da dificuldade encontrada no processo inicial de lógica. Foram analisados softwares com o intuito semelhantes como a linguagem de programação visual “Scratch” e literaturas científicas. Dentre os recursos desta aplicação estão: Aprendizagem do conteúdo básico programacional, resolução de questões, interação entre estudantes e fontes complementares de estudo. O objetivo é abordar uma aplicação que permita que o usuário consiga abstrair o conteúdo inicial da área de programação com leituras complementares e seção de comunidade. O desenvolvimento foi iniciado com o levantamento, análise e documentação de requisitos, além da elaboração de diagrama de caso de uso. A codificação envolveu o uso de linguagens como HTML, CSS e JavaScript, além do framework Django, baseado em Python, no backend. O armazenamento de dados foi realizado com o PostgreSQL. É importante ressaltar que este projeto ainda está em fase de protótipo e, portanto, ainda não foi validado com usuários reais.

Palavras-chave: Pensamento computacional; Ferramenta Educacional; Ensino Colaborativo; Plataformas educacionais; Ensino de programação; Django; Bootstrap.

1. INTRODUÇÃO

O pensamento computacional e o desenvolvimento de software são pilares fundamentais na formação dos estudantes da área de informática. Martins, Marin e Alves (2024) afirmam que a aprendizagem de programação, tanto em cursos técnicos quanto na graduação, tem se revelado uma tarefa desafiadora, gerando dificuldades significativas na assimilação dos conceitos de lógica de programação, o que contribui para altos índices de reprovação e evasão nas disciplinas introdutórias.

A habilidade de programar é considerada como complexa por diversos autores, tanto brasileiros como estrangeiros, ou seja, tal problema é visto no mundo todo, sendo um desafio global desenvolver tal pensamento lógico. Portanto, para encontrar bons resultados requer esforços, paciência e adoção de métodos. (GOMES, et al., 2008).

Nesse sentido, destaca-se a dificuldade de encontrar ferramentas para a aprendizagem da lógica de programação. Ao analisar as soluções existentes há o Scratch, projeto do MIT amplamente utilizado no meio acadêmico, oferece um ambiente gamificado que estimula o pensamento computacional, embora não ensine explicitamente lógica de programação, mas sim a possibilidade de construção de algoritmos baseados em representações visuais. Com isso, ele permite que os usuários desenvolvam esses conceitos de forma intuitiva, por meio da prática com blocos visuais

¹Discente do Técnico em Informática Integrado, IFSULDEMINAS – Campus Muzambinho. E-mail: : jose.mendes@alunos.ifsuldeminas.edu.br

²Orientador, IFSULDEMINAS - Campus Muzambinho. E-mail: paulo.santos@muz.ifsuldeminas.edu.br

(DIEMER; SCHAEFFER; BERCHT, 2021). Além disso, há também pseudo-códigos e fluxogramas que possibilitam uma interação com comunicação mais simples para os usuários iniciantes.

No entanto, muitos estudantes iniciantes ainda têm dificuldades e não sabem onde procurar ou a quem pedir ajuda. Com isso, tem-se o objetivo geral desse projeto que é proporcionar um software intuitivo que possibilita conteúdos iniciais de programação, junto a leituras complementares mais aprofundadas, e uma “comunidade ” que possibilite a discussão entre os estudantes para se ajudarem.

2. MATERIAL E MÉTODOS

O desenvolvimento do projeto teve início com o levantamento, análise e documentação dos requisitos de software. Em seguida, foi utilizada a linguagem de modelagem UML para a elaboração dos diagramas de casos de uso, classes e atividades, com o apoio da ferramenta Astah Professional.

As interfaces da aplicação foram prototipadas no Figma e implementadas com HTML, CSS e JavaScript, utilizando o framework Bootstrap para estilização e responsividade. A lógica do backend foi desenvolvida com o framework Django (Python), e o banco de dados utilizado foi o PostgreSQL.

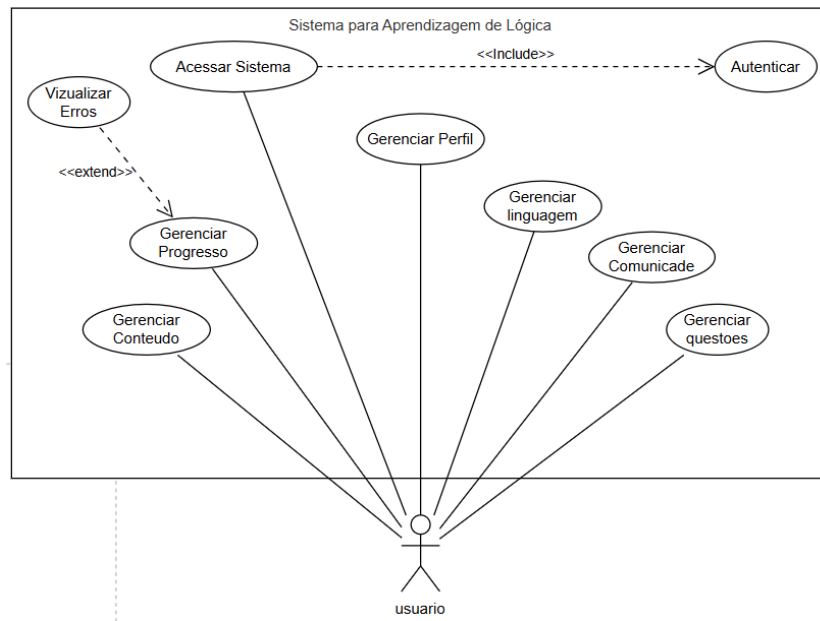
A implementação foi realizada em dois dispositivos: um desktop institucional (Intel Core i3-9100U, 8 GB RAM, Windows 11, 64 bits) e um notebook pessoal (Intel Core i7, 8 GB RAM, Windows 11, 64 bits). Para organização e armazenamento da documentação, utilizou-se o Google Drive, enquanto o código-fonte foi desenvolvido no Visual Studio Code e versionado via GitHub.

3. RESULTADOS E DISCUSSÃO

Por fim, a elaboração do sistema possibilitou compreender, de forma prática, a relevância do processo de prototipação no desenvolvimento de software. Essa etapa abrange desde o levantamento e a análise de requisitos até a produção de artefatos como diagramas de banco de dados, casos de uso e diagramas de classes. A realização dessas atividades foi essencial para garantir que a aplicação desenvolvida estivesse alinhada tanto com os objetivos do projeto quanto com as demandas específicas dos usuários.

O diagrama de casos de uso apresentado na Figura 1 descreve de forma detalhada a especificação dos requisitos do software, contemplando todas as funcionalidades previstas, assim como os atores responsáveis por sua interação. Além disso, esse artefato contribui para a validação dos recursos do sistema e a análise de sua aplicabilidade.

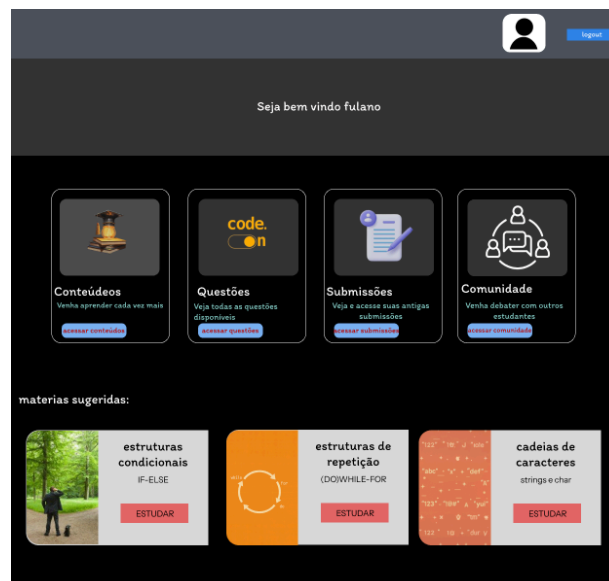
Figura 1: Diagrama de Caso de Uso



Fonte: elaborado pelos autores (2025)

A página inicial do site, representada pela Figura 2, inclui tópicos sugeridos para estudo aos usuários, além de uma barra de navegação que contempla os links para as páginas contendo as funcionalidades do sistema. A página de conteúdo disponibiliza temas para estudo, redirecionando o usuário a materiais elaborados pelo projeto, bem como URLs para fontes complementares. Adicionalmente, há uma aba destinada a questões, que oferece exercícios para serem resolvidos, acesso a questões previamente solucionadas e também há uma página de comunidade, na qual os usuários podem interagir, expressar suas dúvidas e discutir.

Figura 2: Página inicial do sistema



Fonte: elaborado pelos autores (2025)

5. CONCLUSÃO

Conclui-se que a aprendizagem de lógica de programação é vital para que alguém interessado na área de desenvolvimento de software possa aprimorar o raciocínio fundamental na área. Afinal, ela é o alicerce para todo o resto a ser aprendido. Dentre os principais pontos que devem ser levados em consideração no pensamento computacional é o viés teórico, que é essencial, mas também, resolver problemas para entender situações que se pode usar tal assunto. Somado a isso, um lugar em que se possa conversar com outros indivíduos, muitas vezes mais experientes e compreende como é o início, contribui para tornar mais fácil o desenvolvimento da lógica computacional. Ao contrário de softwares como o scratch o projeto visa expor tanto a parte teórica do aprendizado quanto a parte prática com resoluções comentadas das resoluções, ampliando assim o entendimento

Diante de todos esses requisitos, o software em desenvolvimento neste estudo oferece soluções intuitivas e eficientes por meio de uma interface clara, apresentação didática dos conteúdos e resolução de exercícios, além de suporte colaborativo por meio da interação com terceiros, contribuindo assim para um aprendizado mais eficiente do usuário.

Ressalta-se, ainda, que esta aplicação web encontra-se em estágio de prototipação e, portanto, ainda não foi submetida à validação com usuários reais. Futuramente, após a realização de testes de validação e usabilidade, serão efetuados os ajustes necessários para adequá-la às demandas e expectativas dos usuários.

REFERÊNCIAS

DIEMER, Mouriac Halen; SCHAEFFER, Tainá; BERCHT, Magda. Collaby: Um ambiente colaborativo para ensino de Lógica de Programação. Revista Destaques Acadêmicos, v. 13, n. 4, 2021. Disponível em: <https://doi.org/10.22410/issn.2176-3070.v13i4a2021.3061>. Acesso em 23 jun. 2025.

GOMES, Anabela et al. Aprendizagem de programação de computadores: dificuldades e ferramentas de suporte. Revista portuguesa de pedagogia, p. 161-179, 2008. Disponível em: https://doi.org/10.14195/1647-8614_42-2_9. Acesso em 23 jun. 2025

MARTINS, Marcos Winícios Pereira; MARIN, Regina Paiva Melo; ALVES, Laura Beatriz. Desafios na aprendizagem de lógica de programação entre estudantes de tecnologia: uma revisão sistemática da literatura. Revista EDAPECI, v. 24, n. 3, p. 12-24, 2024. Disponível em: <https://doi.org/10.29276/redapeci.2024.24.321348.12-24>. Acesso em 23 jun. 2025.