

Estado da publicação: O preprint foi submetido para publicação em um periódico

O USO DE FERRAMENTAS DE PROGRAMAÇÃO VISUAL NO ENSINO DE PROGRAMAÇÃO NO ENSINO MÉDIO

Monique Casagrande, Pâmela Félix Freitas

<https://doi.org/10.1590/SciELOPreprints.11081>

Submetido em: 2025-01-15

Postado em: 2025-01-28 (versão 1)

(AAAA-MM-DD)

ARTIGO

O USO DE FERRAMENTAS DE PROGRAMAÇÃO VISUAL NO ENSINO DE PROGRAMAÇÃO NO ENSINO MÉDIO

MONIQUE CASAGRANDA¹

ORCID: <https://orcid.org/0009-0008-9406-6380>

<monique.casagranda@ifmt.edu.br>

PÂMELA FÉLIX FREITAS²

ORCID: <https://orcid.org/0000-0003-2219-2173>

<pamela.felix@uece.br>

¹ Instituto Federal de Educação, Ciência e Tecnologias de Mato Grosso. Campo Novo do Parecis, MT, Brasil.

² Universidade Estadual do Ceará. Canindé, CE, Brasil.

RESUMO: Este estudo tem como objetivo analisar o papel das ferramentas de programação visual no ensino de programação para alunos do Ensino Médio, focando em como essas tecnologias podem melhorar a compreensão conceitual, o desempenho acadêmico e o engajamento dos estudantes. Para isso, foi realizada uma pesquisa bibliográfica nas literaturas especializadas sobre o tema englobando publicações entre 2019 e 2024, visando explorar o uso de tecnologias emergentes e plataformas interativas, como *Scratch*, *Tynker* e *MIT App Inventor*. A pesquisa aponta que essas ferramentas desempenham um papel importante ao facilitar o aprendizado de conceitos complexos de programação, promovendo o desenvolvimento do pensamento computacional e habilidades de resolução de problemas de forma prática e visual. Além disso, constatou-se que o uso de metodologias ativas, como a Aprendizagem Baseada em Projetos e a Gamificação, contribui para o engajamento dos alunos, tornando o processo de aprendizagem mais acessível, dinâmico e adaptado às necessidades individuais. No entanto, a pesquisa também identifica desafios importantes, como a falta de infraestrutura tecnológica adequada nas escolas e a necessidade de capacitação contínua dos professores para que possam integrar essas ferramentas de maneira eficaz em suas práticas pedagógicas. Conclui-se que, embora as tecnologias emergentes tenham potencial para transformar o ensino da programação, é essencial que haja um esforço conjunto entre instituições educacionais e governos para investir em infraestrutura e formação docente, garantindo que os benefícios dessas ferramentas possam ser plenamente aproveitados no ambiente escolar.

Palavras-chave: programação visual, tecnologias emergentes, metodologias ativas, gamificação.

THE USE OF VISUAL PROGRAMMING TOOLS IN TEACHING PROGRAMMING IN HIGH SCHOOL

ABSTRACT: This study aims to analyze the role of visual programming tools in teaching programming to high school students, focusing on how these technologies can improve student's conceptual understanding, academic performance, and engagement. To achieve this, a bibliographic review of specialized literature on the subject was conducted, encompassing publications from 2019 to 2024, to explore the use of emerging technologies and interactive platforms such as Scratch, Tynker, and MIT App Inventor. The research indicates that these tools play an important role in facilitating the learning of complex programming concepts, promoting the development of computational thinking and problem-solving skills in a practical and visual way. Furthermore, it was found that the use of active methodologies, such as Project-Based Learning and Gamification, contributes to student engagement, making the

learning process more accessible, dynamic, and tailored to individual needs. However, the study also identifies significant challenges, such as the lack of adequate technological infrastructure in schools and the need for ongoing teacher training to effectively integrate these tools into their pedagogical practices. The study concludes that while emerging technologies have the potential to transform programming education, a joint effort between educational institutions and governments is essential to invest in infrastructure and teacher training, ensuring that the benefits of these tools can be fully realized in the school environment.

Keywords: visual programming, emerging technologies, active methodologies, gamification.

EL USO DE HERRAMIENTAS DE PROGRAMACIÓN VISUAL EN LA ENSEÑANZA DE PROGRAMACIÓN EN LA ESCUELA SECUNDARIA

RESUMEN: Este estudio tiene como objetivo analizar el papel de las herramientas de programación visual en la enseñanza de programación para estudiantes de la escuela secundaria, centrándose en cómo estas tecnologías pueden mejorar la comprensión conceptual, el desempeño académico y el compromiso de los estudiantes. Se realizó una investigación bibliográfica en la literatura especializada sobre el tema, abarcando publicaciones entre 2019 y 2024, con el propósito de explorar el uso de tecnologías emergentes y plataformas interactivas, como *Scratch*, *Tynker* y *MIT App Inventor*. La investigación señala que estas herramientas desempeñan un papel importante al facilitar el aprendizaje de conceptos complejos de programación, promoviendo el desarrollo del pensamiento computacional y habilidades para la resolución de problemas de manera práctica y visual. Se constató que el uso de metodologías activas, como el Aprendizaje Basado en Proyectos y la Gamificación, contribuye al compromiso de los estudiantes, haciendo que el proceso de aprendizaje sea más accesible, dinámico y adaptado a las necesidades individuales. Pero, la investigación también identifica desafíos importantes, como la falta de infraestructura tecnológica adecuada en las escuelas y la necesidad de una formación continua para los profesores, que les permita integrar estas herramientas de manera eficaz en sus prácticas pedagógicas. Se concluye que, aunque las tecnologías emergentes tienen el potencial de transformar la enseñanza de la programación, es esencial que haya un esfuerzo conjunto entre las instituciones educativas y los gobiernos para invertir en infraestructura y formación docente, garantizando que los beneficios de estas herramientas puedan aprovecharse plenamente en el entorno escolar.

Palabras clave: programación visual, tecnologías emergentes, metodologías activas, gamificación.

INTRODUÇÃO

As tecnologias vêm sendo aplicadas aos processos educacionais e sua aplicação no ensino de programação tem ganhado destaque com o uso de ferramentas interativas de programação visual. A partir do surgimento dos computadores pessoais, a ideia de utilizá-los para fins educacionais foi se tornando cada vez mais popular. Inicialmente, pensava-se apenas em alfabetização digital mas, com o passar do tempo, a programação passou a se tornar uma habilidade essencial. Esta mudança de paradigma revela uma transição na educação, onde saber linguagens de programação não é visto apenas como uma habilidade técnica, mas como uma forma de pensamento crítico e resolução de problemas, que são habilidades consideradas essenciais no século XXI.

Segundo relatório do Cetic.br (2023, p. 25), “há uma crescente demanda por habilidades de programação no Brasil, impulsionada pela transformação digital em diversos setores econômicos”. Apesar disso, o ensino dessas habilidades ainda é limitado nas escolas brasileiras. Apenas uma pequena porcentagem das instituições de ensino oferece cursos de programação de forma estruturada. Desse modo questão de pesquisa que se coloca é: de que forma as ferramentas de programação visual podem

ser utilizadas no Ensino Médio para melhorar a compreensão conceitual e o engajamento dos alunos no aprendizado da programação? Esse é o problema que esse estudo pretende abordar.

O campo educacional, especialmente para estudantes do Ensino Médio, tem acompanhado a crescente demanda de aprendizado de linguagens de programação, adotando metodologias inovadoras para ensiná-las. No entanto, o processo de ensino-aprendizagem da programação apresenta muitos desafios, principalmente quando se trata de apresentar conceitos complexos, que geralmente são novos aos estudantes. A integração de tecnologias emergentes, como ferramentas de programação visual e plataformas interativas, aparecem como uma solução promissora para tornar o ensino destes conceitos mais acessível e eficaz (Coelho, Melo, Rosa & Farias, 2023).

Uma estratégia inovadora que tem ganhado destaque no ensino de programação é o uso de ferramentas de programação visual, que transformam o processo de aprendizagem em uma experiência mais interativa e acessível. Ferramentas como o *Scratch*¹ permitem que os alunos desenvolvam seus próprios projetos, explorando conceitos de programação como *loops* e condicionais de forma prática e visual. Essas plataformas oferecem um ambiente lúdico e envolvente, permitindo que os alunos testem seus conhecimentos em tempo real. Coelho et al. (2023) destacam que o uso do *Scratch* como metodologia ativa tem sido eficaz para aumentar o engajamento dos alunos e melhorar sua compreensão dos conceitos de programação, tornando a aprendizagem mais dinâmica e adaptada às necessidades dos estudantes do Ensino Médio.

Aplicar tecnologias emergentes no ensino de programação também facilitam a personalização do aprendizado. Em um ambiente que utiliza ferramentas de programação visual, os estudantes podem avançar no seu próprio ritmo, recebendo um feedback imediato sobre o seu desempenho. Isso é especialmente valioso, pois sabemos que estudantes inseridos em uma mesma turma possuem diferentes níveis de conhecimento prévio e, também, diferentes capacidades de aprendizado.

Além disso, ambientes de programação visual, como o *Tynker*², que permite que os estudantes pratiquem conceitos de programação de maneira acessível e interativa, explorando temas como lógica de algoritmos e resolução de problemas de forma prática e o *Alice*³, que é uma plataforma de programação visual para conceitos de programação orientada a objetos por meio de criação de animações e jogos interativos, disponibilizam meios para os alunos aprendam os conceitos de forma interativa e dinâmica. Freitas (2023) destaca que a personalização no ensino de programação orientada a objetos, por meio dessas plataformas, tem demonstrado ser um fator crucial para o sucesso dos alunos, especialmente quando adaptadas às suas necessidades individuais.

Um dos principais obstáculos é a falta de formação adequada para os professores, que muitas vezes não se sentem confortáveis com o uso de tecnologias digitais em suas aulas, seja pela falta de familiaridade com as ferramentas ou pela resistência à adoção de novas metodologias. Uma pesquisa realizada pela Fundação Telefônica Vivo (2022), aponta que, no ano em que a pesquisa foi realizada, apenas 56% dos docentes participaram de algum tipo de formação continuada sobre o uso de tecnologias digitais, uma redução em relação aos 65% registrados no ano anterior. Além disso, outra barreira significativa enfrentada pelos docentes é a infraestrutura tecnológica de muitas escolas, principalmente aquelas em locais com menor acesso à tecnologia. Segundo dados do INEP (2022), nas escolas de Ensino Médio da rede pública, 77,7% das escolas possuem acesso à internet de alta velocidade, porém apenas 66% utilizam este recurso no ensino e na aprendizagem. Sem a utilização de equipamentos adequados ou uma boa conexão de internet, a implementação dessas plataformas torna-se difícil.

Outro desafio enfrentado é a necessidade de equilibrar o uso de elementos de jogos com os objetivos educacionais. Apesar de as ferramentas de programação visual aumentarem o engajamento dos alunos, é de extrema importância que as atividades sejam planejadas de forma a promover o aprendizado significativo. As atividades devem ser pensadas de forma que o aluno compreenda profundamente os conceitos de programação, e não simplesmente a conclusão da tarefa. Freitas (2023) mostra que, no ensino da programação orientada a objetos, é crucial que a gamificação esteja alinhada

¹ Acesso em: <https://scratch.mit.edu/>

² Acesso em: <https://www.tynker.com/>

³ Disponível para instalação em: <https://alice.softonic.com.br/>

com os objetivos pedagógicos, garantindo que os alunos adquiram as competências necessárias para resolver problemas complexos.

No entanto, estudos como o de Hwang, Sung, Chang e Huang (2020) têm demonstrado que o uso de abordagens pedagógicas centradas no aluno, como o aprendizado baseado em projetos e o desenvolvimento de jogos interativos são melhores práticas eficazes na implementação dessas tecnologias, pois permitem que os alunos desenvolvam habilidades práticas e críticas de forma colaborativa e personalizada. Para superar essas barreiras, é necessário um esforço conjunto entre governos, instituições educacionais e comunidades, visando treinamento contínuo dos professores e a melhoria da infraestrutura escolar.

O uso de tecnologias emergentes, como ferramentas de programação visual, no ensino da programação não se limita a plataformas como o *Scratch*, *Tynker* e *Alice*. Outros ambientes de programação visual e simuladores, também têm sido explorados para enriquecer o aprendizado. Estas ferramentas possibilitam ao aluno visualizar o funcionamento de seus códigos em tempo real, o que facilita a compreensão de conceitos abstratos. Além disso, esses ambientes interativos incentivam que o aluno teste diferentes abordagens para resolver um problema, sem medo de cometer erros. Esse pensamento de aprendizado prático é essencial para o desenvolvimento de habilidades de programação, pois promove o pensamento crítico e a resolução criativa de problemas (Coelho et al., 2023).

Apesar dos desafios, existem inúmeras vantagens na integração de ferramentas de programação visual no ensino da programação. Estudantes que utilizam essas ferramentas em atividades práticas, como o desenvolvimento de projetos e resolução de problemas, tendem a melhorar significativamente no desempenho acadêmico. Em contextos onde os alunos têm dificuldades em assimilar os conceitos iniciais de programação, essas tecnologias emergentes oferecem uma maneira prática e eficaz de superar essas barreiras, promovendo uma maior compreensão dos conceitos e mantendo os alunos engajados de forma contínua.

Em sua pesquisa sobre a aplicação de ferramentas de programação visual no ensino de lógica e algoritmos, Freitas (2023) destaca que essas plataformas oferecem um potencial significativo para o aprendizado de conceitos complexos, como a programação orientada a objetos. A capacidade de visualizar os resultados em tempo real ajuda os alunos a entenderem melhor os conceitos e aplicá-los de maneira prática e criativa. Embora a adoção dessas ferramentas ainda esteja em expansão, existe um grande potencial para seu uso em diferentes níveis educacionais, especialmente no Ensino Médio, onde os alunos estão desenvolvendo habilidades de resolução de problemas mais complexo. No entanto, é necessário continuar investindo em pesquisas que explorem os impactos dessas tecnologias em diferentes contextos educacionais, garantindo sua eficácia e adaptabilidade às necessidades dos alunos.

Vistos esses desafios e oportunidades, o presente estudo tem como objetivo identificar e classificar as principais ferramentas de programação visual utilizadas no ensino de programação para melhorar a compreensão e o engajamento dos alunos e analisar os desafios e melhores práticas enfrentados pelos educadores na implementação dessas tecnologias. Este trabalho faz parte de uma pesquisa elaborada no âmbito do mestrado em Tecnologias Emergentes em Educação, desenvolvido em 2024, na Must University, inserindo-se em um contexto acadêmico que busca explorar o impacto das inovações tecnológicas no ensino, especialmente no desenvolvimento de competências essenciais para o século XXI, como a programação. Fundamentado em uma pesquisa bibliográfica, este trabalho busca oferecer uma visão abrangente sobre como essas ferramentas podem ser aplicadas de maneira eficaz no Ensino Médio, ajudando os alunos a melhorar em sua compreensão dos conceitos e a desenvolver habilidades práticas.

Além do objetivo geral mencionado acima, este estudo tem como objetivos específicos: (1) Analisar como essas ferramentas podem melhorar a compreensão dos alunos em conceitos de programação, (2) Avaliar as práticas pedagógicas que integram essas tecnologias de maneira eficaz, e (3) Explorar os desafios e barreiras enfrentados pelos educadores na implementação dessas tecnologias.

A pesquisa bibliográfica será conduzida, conforme orientado por Souza, Silva e Carvalho (2010). Estabeleceu-se como recorte temporal publicações dos últimos 5 anos, que abordam o uso de tecnologias emergentes no ensino da programação, especialmente no Ensino Médio, porém, quando publicações fora deste recorte se mostraram relevantes para este estudo, também foram incluídas. Foram

consultadas fontes de pesquisa confiáveis, como o portal de periódicos da CAPES, Scopus, Scielo e etc. Serão abordados artigos acadêmicos, livros e relatórios governamentais para identificar as melhores práticas e os desafios relacionados à implementação de ferramentas de programação visual no contexto educacional. A metodologia segue orientações para revisões sistemáticas propostas por Kitchenham e Charters (2007) para garantir a abrangência na análise.

Como discutido ao longo deste texto, a utilização de plataformas on-line, ferramentas interativas e ambientes de programação visual oferecem uma oportunidade única para transformar o ensino da programação. À medida que mais pesquisas são conduzidas e novas ferramentas são desenvolvidas, espera-se que o uso dessas tecnologias se torne uma prática cada vez mais comum nas escolas. Com isso, o ensino de programação poderá evoluir para um formato mais inclusivo, acessível e eficaz, preparando os alunos para enfrentar os desafios de um mundo cada vez mais digital, estando munidos dos conhecimentos exigidos pelo século XXI.

METODOLOGIA

A metodologia deste estudo é uma pesquisa bibliográfica, seguindo os princípios de Sousa, Oliveira e Alves (2021), que explicam que a pesquisa bibliográfica é elaborada a partir de materiais já publicados, proporcionando ao pesquisador um contato direto com o conhecimento acumulado. Para este estudo, esse método foi escolhido por ser adequado para analisar as principais ferramentas de programação visual utilizadas no ensino de programação para alunos do Ensino Médio, considerando sua eficácia e os desafios enfrentados pelos educadores.

As referências bibliográficas foram selecionadas em dois idiomas: inglês e português, o que permitiu analisar trabalhos de maneira mais abrangente e diversificada, considerando diferentes contextos e abordagens pedagógicas. Segundo Prodanov e Freitas (2013), é essencial considerar publicações em múltiplos idiomas em uma revisão sistemática, uma vez que a inclusão de trabalhos internacionais amplia o espectro de informações e contribui para a qualidade do estudo.

Para orientar as buscas, foram utilizadas palavras-chave específicas, como: “ferramentas de programação visual”, “ensino de programação no Ensino Médio”, “gamificação no ensino da programação” e “tecnologias emergentes na educação”. A separação do material seguiu uma metodologia clara e rigorosa, dividida em três etapas: primeiro, a busca inicial foi realizada nas bases de dados já mencionadas, seguida pela leitura dos títulos e resumos encontrados, para verificar se o estudo teria relevância para o tema. Finalmente, os estudos escolhidos foram lidos para confirmar sua aplicabilidade ao objetivo da pesquisa. Este processo seguiu os princípios de análise qualitativa descritos por Lüdke e André (1986), que destacam a importância de um planejamento cuidadoso e uma análise sistemática dos dados coletados.

Os critérios de inclusão adotados tiveram foco em publicações que abordam o uso de ferramentas de programação visual no Ensino Médio, sem limitar a análise a ferramentas e plataformas específicas. De acordo com Prodanov e Freitas (2013), a definição clara de critérios de inclusão e exclusão é essencial para garantir a relevância e a consistência dos estudos selecionados.

A análise dos dados coletados foi realizada de forma qualitativa, com base na organização dos conteúdos retirados dos artigos selecionados. Seguindo os princípios propostos por Gil (2010), os dados foram organizados em categorias temáticas que permitiram a identificação dos principais desafios e melhores práticas no uso de ferramentas de programação visual no ensino da programação.

A organização envolveu temas como as tecnologias mais eficazes para a melhoria do desempenho dos alunos, o impacto no engajamento dos estudantes e as dificuldades enfrentadas pelos educadores na implementação dessas tecnologias em sala de aula. Esse processo possibilita uma análise mais aprofundada dos detalhes que envolvem a utilização destas ferramentas. Segundo Severino (2016), a categorização de informações facilita a organização e interpretação dos dados, promovendo uma compreensão mais profunda dos fenômenos estudados.

Além disso, outros critérios como relevância, acesso aberto e atualidade, também foram levados em consideração para a escolha das referências bibliográficas. A relevância foi avaliada observando o encaixe do conteúdo no tema central da pesquisa, também foi priorizado o acesso aberto aos documentos utilizados, fazendo uso apenas de publicações disponíveis gratuitamente. A seleção dos

materiais a serem utilizados leva em conta trabalhos publicados entre 2019 e 2024 para, assim, garantir que o material revisado seja atual e mostre as tendências e inovações recentes no campo do ensino de programação com ferramentas visuais, visto que a área tecnológica é bastante mutável, porém, quando necessário, foram utilizados materiais que estão além destes recortes temporais. A atualidade das publicações foi considerada essencial na escolha dos trabalhos utilizados para garantir que fossem considerados os avanços mais recentes no campo da educação tecnológica.

Por fim, este estudo também apresenta desafios, como escolher quais ferramentas de programação visual serão apresentadas, pois existe uma ampla gama de plataformas disponíveis para utilização, gerando uma pesquisa bastante ampla, podendo dificultar a generalização dos resultados, mas, segundo Marconi e Lakatos (2003), toda pesquisa bibliográfica enfrenta o desafio de reunir informações dispersas em diversas fontes, exigindo que o pesquisador mantenha um olhar crítico e focado para garantir a consistência do trabalho.

TECNOLOGIAS EMERGENTES E FERRAMENTAS DE PROGRAMAÇÃO VISUAL NO ENSINO DE PROGRAMAÇÃO

As tecnologias emergentes estão transformando o ensino de programação, oferecendo ferramentas que tornam o aprendizado mais acessível e interativo, caminhando em conjunto com às necessidades dos estudantes do Ensino Médio. Plataformas de programação visual, como o *Scratch* e o *MIT App Inventor*⁴, destacam-se como recursos eficazes para introduzir conceitos de lógica de programação de forma intuitiva e envolvente (Silva, Matos, Libório Filho & Barbosa, 2023). Além disso, essas ferramentas ajudam a esclarecer os conceitos fundamentais de programação e a praticar o pensamento computacional, uma habilidade essencial para os estudantes do século XXI (Shimasaki & Prado, 2021).

Ambientes de programação visual e suas plataformas on-line desempenham um papel crucial na universalização do ensino da programação, permitindo que alunos iniciantes se envolvam no aprendizado de maneira prática e lúdica. A gamificação dessas plataformas, como visto com o uso do *Kahoot!*⁵ e do *Scratch*, tem mostrado resultados positivos na motivação e engajamento dos alunos, contribuindo para uma experiência de aprendizagem mais dinâmica e participativa. Além disso, o uso dessas tecnologias possibilita a adaptação de currículos tradicionais, integrando elementos interativos que facilitam a personalização do aprendizado e atendem às diferentes necessidades dos alunos (Santana & Martins, 2023).

O uso dessas tecnologias possibilita a adaptação dos currículos tradicionais com a inclusão de elementos interativos que personalizam o processo de aprendizagem para atender às diferentes necessidades dos alunos. A acessibilidade dos recursos educacionais se reflete no fato de que as ferramentas de programação visual podem ser utilizadas tanto no ambiente escolar quanto em casa. Essa flexibilidade é crucial para superar obstáculos como a falta de infraestrutura em algumas instituições de ensino e a resistência inicial dos educadores à implementação dessas tecnologias (Santos, Santos, Melo Júnior, Medeiros, Penha, Ramalho, Moniz & Oliveira, 2024b).

Adiante, a discussão aborda as tecnologias emergentes no ensino da programação e discute algumas das ferramentas mais importantes atualmente disponíveis, suas aplicações práticas e o impacto no engajamento e desempenho dos alunos.

Tecnologias Emergentes no Ensino da Programação

O avanço das tecnologias emergentes trouxe um impacto significativo no ensino da programação, criando novas formas de engajamento e aprendizado para os estudantes do Ensino Médio. Durante o processo de ensino, ao introduzir ferramentas como plataformas de programação visual, ambientes gamificados e tecnologias de realidade aumentada é oferecido um ambiente mais interativo e de fácil acesso. As tecnologias emergentes englobam uma série de recursos que, além de facilitar a

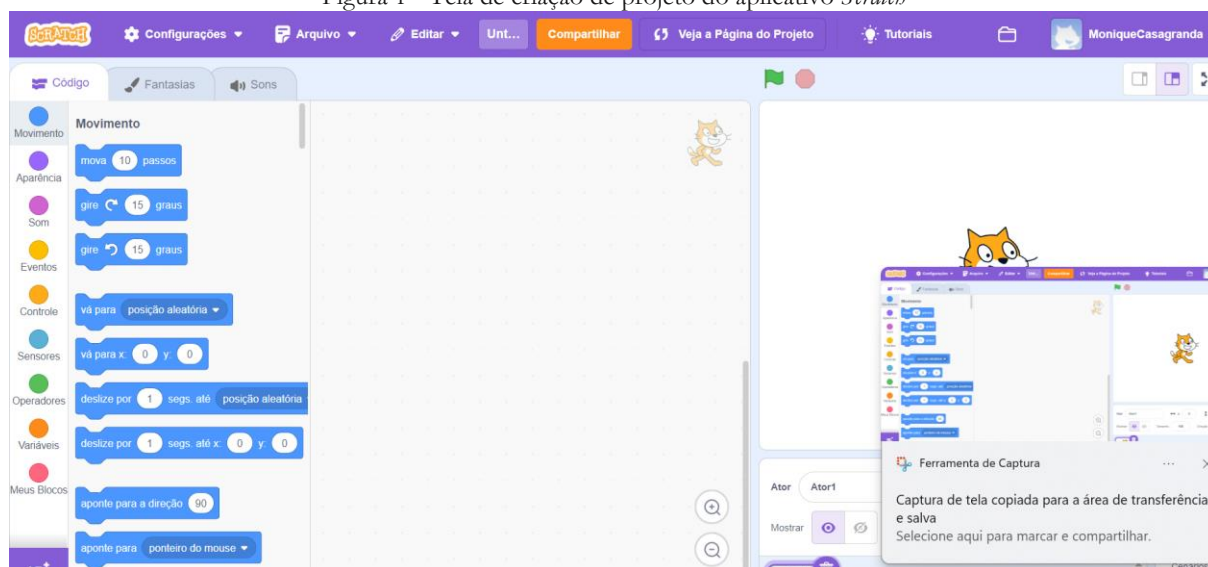
⁴ Acesso em: <https://appinventor.mit.edu/>

⁵ Acesso em: <https://kahoot.com/>

compreensão dos conceitos de programação, também desenvolvem habilidades críticas como o pensamento computacional e a resolução de problemas (Silva et al., 2023).

Um exemplo disso é o uso de plataformas como o *MIT App Inventor*, que permite aos estudantes criarem aplicativos de forma intuitiva, através de um método de blocos de construção visual. A ferramenta tem se mostrado útil nas aulas de lógica de programação, especialmente em nível iniciante, ao proporcionar um espaço de desenvolvimento onde a criatividade se entrelaça com os fundamentos da programação. Estudos mostram que o *MIT App Inventor* não apenas facilita a aprendizagem, mas também aumenta o engajamento dos alunos, ao permitir que eles visualizem imediatamente os resultados de suas criações (Silva et al., 2023). Além disso, a outra plataforma amplamente utilizada, o *Scratch*, permite que o conhecimento seja construído através de atividades lúdicas e interativas, motivando os estudantes a explorar conceitos com entusiasmo e colocá-los em prática (Costa, Lima, Almeida, Magalhães & Almeida, 2022).

Figura 1 - Tela de criação de projeto do aplicativo *Scratch*



Fonte: Elaborado pelas autoras

A gamificação desempenha um papel essencial no ensino da programação, transformando o processo de aprendizagem em uma experiência mais divertida e envolvente. Essas plataformas oferecem pontuações, desafios e recompensas para engajar os estudantes ativamente, tornando o aprendizado mais dinâmico e estimulante. Essa abordagem não só melhora o desempenho acadêmico, mas também promove um ambiente de aprendizagem colaborativo, onde os alunos podem competir e aprender uns com os outros (Santana & Martins, 2023).

Além disso, as novas tecnologias estão em expansão para incluir ferramentas como simuladores e ambientes virtuais, proporcionando aos alunos uma experiência mais rica. Por exemplo, o uso de simuladores permite que o estudante explore conceitos complexos de programação em um ambiente totalmente seguro e controlado, sem o risco de cometer erros críticos. Isso é especialmente útil em disciplinas que exigem uma compreensão profunda de algoritmos e estruturas de dados, proporcionando um espaço para testes e experimentações que seriam inviáveis em ambientes tradicionais de ensino (Menegotto & Torres, 2019).

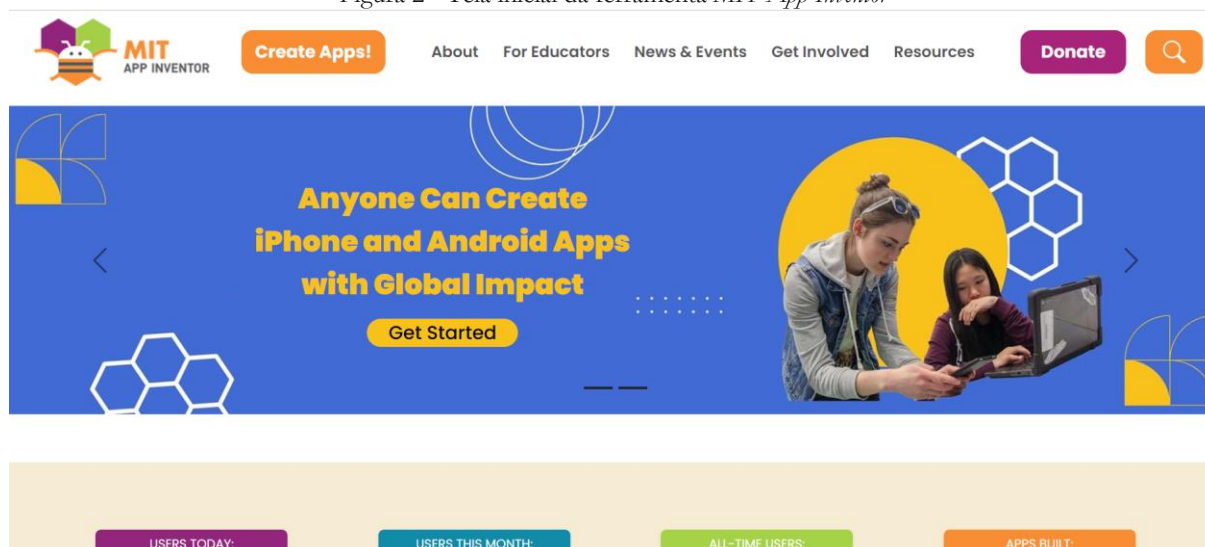
O impacto dessas tecnologias no ensino da programação é amplamente reconhecido pela literatura, que destaca não só a melhoria no engajamento dos alunos, mas também a capacidade dessas ferramentas de democratizar o acesso ao conhecimento tecnológico. O cenário em constante mudança com os avanços tecnológicos exigindo que os educadores adaptem suas metodologias e ferramentas. Segundo Santos, Caetano, Araujo, Costa, Melo Júnior, Graf, Silva e Santos (2024a), é essencial que as instituições de ensino invistam na formação continuada dos professores, para que possam integrar essas tecnologias de forma eficaz em suas práticas pedagógicas. Em resumo, as tecnologias emergentes

representam, no ensino da programação, uma oportunidade valiosa para inovar e melhorar o processo educacional, preparando os alunos para os desafios da era digital.

Ambientes de Programação Visual e Plataformas Interativas

Os ambientes de programação visual e as plataformas interativas têm ganhado destaque no ensino da programação, especialmente no Ensino Médio. Ferramentas como *Tynker* e *MIT App Inventor* tornam o aprendizado mais acessível ao converter conceitos abstratos em atividades visuais e práticas. Boff, Salami e Miotto (2023) ressaltam que utilizando blocos de código que se encaixam como peças de um quebra-cabeça, essas plataformas tornam a programação acessível, permitindo que os alunos compreendam e apliquem conceitos de lógica, algoritmos e estruturas de controle sem a complexidade de sintaxes tradicionais de linguagens de programação.

Figura 2 - Tela inicial da ferramenta *MIT App Inventor*



Fonte: Elaborado pelas autoras

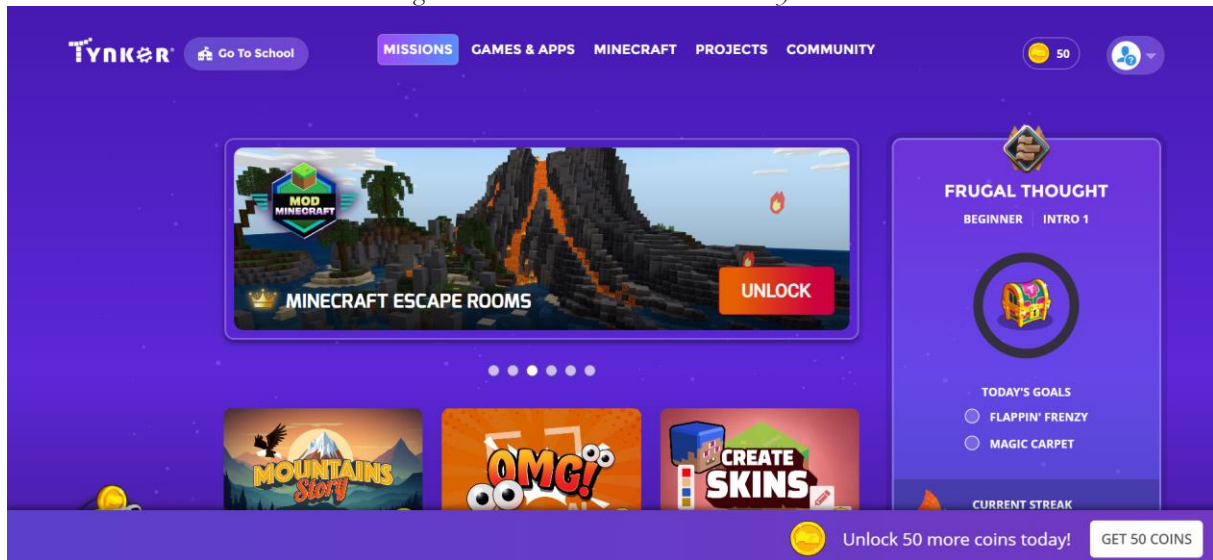
Um exemplo de plataforma que se destaca por sua interface intuitiva e altamente visual é o *Scratch*, que possibilita que os alunos desenvolvam seus próprios projetos de forma colaborativa e interativa. Nesta plataforma, os alunos podem experimentar, errar e aprender com seus erros, o que é essencial para o desenvolvimento de habilidades de resolução de problemas. Além disso, o *Scratch* permite que os alunos compartilhem seus projetos on-line, interajam com outros programadores iniciantes e até mesmo remixem trabalhos existentes, o que amplia as possibilidades de aprendizado e troca de conhecimentos (Lyra, Katakura & Martins, 2022).

Uma das principais vantagens dessas plataformas é que capacidade de personalizar o conteúdo para diferentes níveis de habilidade, isso permite que os alunos avancem em seu próprio ritmo e recebam feedback imediato. Isso é especialmente importante para estudante que precisam de mais tempo para entender conceitos complexos ou que preferem aprender de forma autônoma. As plataformas também oferecem tutoriais guiados e atividades estruturadas que ajudam a consolidar o conhecimento adquirido ao longo das aulas (Menegotto & Torres, 2019). A combinação dessas características promove um ambiente de aprendizagem inclusivo e motivador, que incentiva a prática constante do desenvolvimento contínuo das habilidades dos alunos.

Além do *Scratch*, o *Tynker* é amplamente utilizado para ensinar programação através de jogos e animações, permitindo que os alunos pratiquem lógica de algoritmos de maneira acessível e divertida. Freitas (2023) relata que esta plataforma não só facilita a compreensão dos conceitos de programação, mas também promove a aprendizagem de forma a promover o engajamento ao utilizar histórias interativas e desafios que motivam os alunos a continuar explorando novos níveis de complexidade na programação. De forma semelhante, o *Alice*, uma plataforma voltada para programação orientada a

objetos, permite que os alunos criem animações em 3D e jogos interativos, facilitando a compreensão de conceitos completos por meio da prática visual.

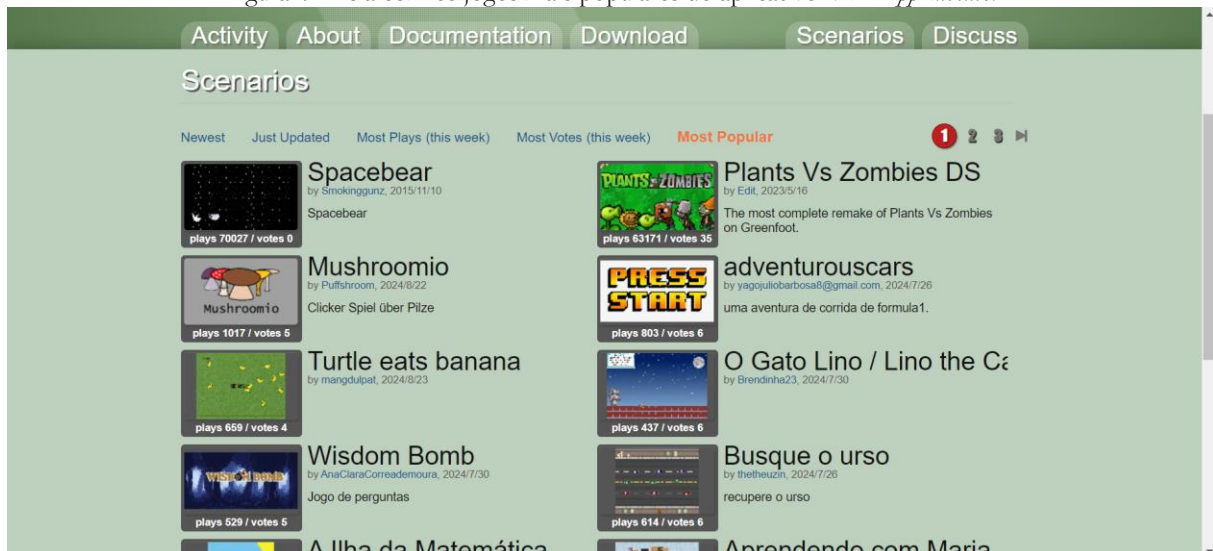
Figura 3 - Tela inicial da ferramenta *Tynker*



Fonte: Elaborado pelas autoras

Ambientes de programação visual e suas plataformas on-line desempenham um papel fundamental na universalização do ensino da programação, permitindo que alunos iniciantes se envolvam no aprendizado de maneira prática e lúdica. O *Roblox Studio*⁶, em particular, promove o desenvolvimento de habilidades de programação em um contexto de criação colaborativa, permitindo que os alunos trabalhem em jogos 3D complexos usando a linguagem de programação Lua. Já o *MIT App Inventor*, voltado para a criação de jogos 2D e 3D, utiliza Java como base e permite aos alunos explorar conceitos de programação orientada a objetos em um ambiente visualmente intuitivo (Santana & Martins, 2023).

Figura 4 - Tela com os jogos mais populares do aplicativo *MIT App Inventor*



Fonte: Elaborado pelas autoras

Essas ferramentas se destacam por suas características específicas: o *Roblox Studio* oferece uma plataforma onde os alunos podem criar e publicar seus próprios jogos, desenvolvendo habilidades não apenas de programação, mas também de design e narrativa digital. Já o *MIT App Inventor* combina

⁶ Acesso em: <https://create.roblox.com/>

programação com visualização de objetos, ideal para alunos que estão começando com Java e desejam ver os resultados de seu código em tempo real, o que facilita a compreensão de conceitos complexos de programação orientada a objetos (Silva et al., 2023).

A seguir, será apresentada uma tabela comparativa que resume as características principais, benefícios e limitações das ferramentas discutidas. Essa comparação tem por objetivo fornecer uma visão clara das funcionalidades e aplicações de cada ferramenta no ensino da programação, permitindo a visualizar qual ferramenta é mais adequada ao uso em cada contexto educacional.

Tabela 1 - Tabela comparativa das ferramentas de programação visual

Ferramenta	Características Principais	Benefícios	Limitações
<i>Scratch</i>	Programação em blocos, interface amigável, comunidade ativa	Facilita a compreensão de conceitos básicos de programação, promove a criatividade	Limitado para programação avançada, foco em conceitos básicos
<i>Tynker</i>	Programação em blocos, integração com jogos, foco em crianças	Aumenta o engajamento com jogos, oferece desafios progressivos	Custos associados a alguns recursos, menor profundidade técnica
Alice	Programação orientada a objetos, criação de animações 3D e jogos	Ajuda a entender conceitos de programação orientada a objetos de forma visual	Necessita de máquinas com boa capacidade gráfica, curva de aprendizado mais longa
<i>Roblox Studio</i>	Desenvolvimento de jogos 3D, programação com Lua, criação colaborativa	Promove a colaboração entre alunos, excelente para ensinar lógica de jogos complexos	Complexidade elevada para iniciantes, requer conhecimentos prévios de programação
<i>MIT App Inventor</i>	Programação orientada a objetos, ambiente para criação de jogos 2D e 3D, integração com Java	Combina programação com visualização de objetos, ideal para aprender Java	Requer conhecimentos básicos de Java, interface pode ser menos intuitiva

Fonte: Elaborada pelas autoras

Além disso, o uso de plataformas de programação visual em contextos educacionais ajuda a preencher lacunas no currículo tradicional de ciência da computação. Muitas escolas ainda enfrentam desafios na implementação de cursos formais de programação devido à falta de recursos ou formação especializada dos professores. As plataformas como *Scratch* e Alice oferecem soluções práticas para esses problemas, permitindo que as escolas integram conceitos de programação de maneira flexível e acessível.

Outra grande vantagem desses ambientes é a personalização do aprendizado. Neles, o aluno pode avançar em seu próprio ritmo, recebendo feedbacks imediatos sobre seu desempenho. Isso é bastante valioso, pois promove um ambiente inclusivo e adaptável, respeitando os diferentes níveis de conhecimento e capacidades de aprendizado dos estudantes. Além disso, essas plataformas permitem aos professores ajustarem as atividades para atender melhor às necessidades específicas de cada aluno, facilitando a inclusão de estudantes com diferentes estilos de aprendizagem e promovendo a auto eficácia, onde os estudantes sentem que estão no controle de sua jornada de aprendizagem (Menegotto & Torres, 2019).

A integração de gamificação em plataformas como o *Scratch*, Alice e *Tynker* ajuda a tornar o processo de aprendizagem mais envolvente e interativo. Agregar desafios, recompensas e elementos de jogo não apenas motiva os alunos, mas também melhora significativamente a retenção de conhecimento. Essa abordagem faz com que os estudantes se engajem mais e compreendam mais profundamente os

conceitos de programação ao transformar o aprendizado em uma experiência dinâmica que conecta conteúdos teóricos e práticas concretas e lúdicas (Madureira & Schneider, 2021). Desta maneira, esses ambientes contribuem significativamente para a motivação dos estudantes, incentivando-os a explorar, criar e compartilhar suas criações, transformando o processo de aprendizagem em uma atividade divertida e educativa.

Plataformas On-line e Ambientes de Programação Visual

Plataformas on-line e ambientes de programação visual têm se destacado como ferramentas essenciais no ensino da programação, especialmente no contexto educacional do Ensino Médio, devido à sua capacidade de combinar acessibilidade, flexibilidade e aprendizado interativo. Diferente dos ambientes de programação visual que utilizam blocos de código para ensinar de forma lúdica e colaborativa, as plataformas on-line enfatizam a programação em um contexto mais amplo, onde jogos e estratégias são utilizados para engajar os alunos e ensinar conceitos complexos de forma prática.

Figura 5 - Tela de jogos da ferramenta *CodeCombat*



Fonte: Elaborado pelas autoras

Uma plataforma com bastante destaque nesse contexto é o *CodeCombat*⁷, que ensina os alunos a programarem através de um jogo de estratégia baseado em cenários medievais. Nele, os alunos programam o comportamento dos personagens utilizando linguagens como *Python* e *JavaScript*, assim desenvolvendo habilidades de codificação enquanto completam desafios e avançam pelos níveis do jogo. Segundo Madureira e Schneider (2021), a gamificação se mostra uma metodologia adequada para promover a aprendizagem, pois possibilita aos estudantes motivarem-se nos estudos e incentiva a interação com os colegas. O foco está na aplicação dos conceitos, permitindo que os estudantes enfrentem obstáculos e encontrem soluções usando a lógica de programação.

Outra plataforma que merece destaque é o *LightBot*⁸, que utiliza desafios de programação em um formato de quebra-cabeça para ensinar lógica e algoritmos de forma simplificada. Essa plataforma, por ser totalmente on-line, oferece flexibilidade. Assim, os alunos podem aprender em qualquer hora e lugar, não estando delimitados ao ambiente tradicional de sala de aula. A estrutura em níveis e a progressão gradual garantem que os alunos avancem conforme dominam cada conceito, respeitando o ritmo individual de aprendizado (Silva et al., 2023).

Além disso, também proporcionam uma forma de ensino mais autônoma, onde os alunos podem explorar, testar e refinar seus conhecimentos de maneira contínua. Também é interessante, especialmente para alunos que necessitam de um ambiente de aprendizado que se adapte às suas

⁷ Acesso em: <https://codecombat.com/>

⁸ Acesso em: <https://www.lightbot.lu/> ou disponível para download em: <https://lightbot.com/>

necessidades individuais, proporcionando oportunidades para prática adicional e revisão fora do horário escolar regular. As plataformas on-line ajudam a criar uma cultura de aprendizado contínuo e autodirigido, essencial para o desenvolvimento de habilidades críticas na era digital (Santos et al., 2024a).

Essas plataformas on-line se diferenciam dos ambientes de programação visual por oferecerem uma experiência mais acessível e adaptável. Elas não apenas introduzem os conceitos básicos de programação de forma interativa, mas também promovem a autonomia e a flexibilidade do estudante, ao possibilitar que o aprendizado ocorra de maneira autônoma e personalizada. Esse modelo de ensino é particularmente eficaz para alunos que buscam aprender no próprio ritmo e enfrentar desafios estruturados que simulam cenários reais de programação, preparando-os melhor para as demandas do mercado e para o desenvolvimento do pensamento crítico.

Plataformas como o *CodeCombat* e o *LightBot* não apenas preparam os alunos para desafios acadêmicos, mas também para o futuro profissional. Elas são uma ponte entre o aprendizado básico e a aplicação prática, permitindo que os alunos experimentem conceitos de forma tangível. Explorar jogos e estratégias como métodos de ensino mostra como essas ferramentas transformam o aprendizado de programação em uma atividade envolvente e contínua, que estimula a curiosidade e a perseverança dos estudantes. Ao integrar estratégias gamificadas e recursos visuais atrativos, essas plataformas tornam o aprendizado de programação mais acessível para todos, independentemente de sua experiência prévia ou nível de habilidade, promovendo uma inclusão mais ampla no ensino da programação (Santos et al., 2024b).

Inovação e Acessibilidade Tecnológica

A inovação tecnológica no campo da educação tem impulsionado uma série de mudanças significativas, especialmente na forma como as tecnologias digitais são aplicadas para tornar o ensino mais acessível e inclusivo. As Tecnologias Digitais da Informação e Comunicação (TDICs) desempenham um papel fundamental na promoção da equidade educacional, ao ofertar ferramentas que facilitam o acesso ao aprendizado de qualidade para alunos de diferentes contextos socioeconômicos e geográficos.

A acessibilidade é um dos aspectos mais impactantes na adoção das TDICs, especialmente quando levamos em consideração os alunos com deficiências. As tecnologias educacionais podem ser adaptadas para atender a uma variedade de estilos de aprendizagem, permitindo que alunos com deficiências visuais, auditivas ou cognitivas tenham acesso a recursos pedagógicos customizados especialmente para suas necessidades. Plataformas de aprendizagem on-line, como ambientes virtuais de aprendizagem, fornecem recursos multimídia interativos que facilitam o entendimento de conteúdos complexos, proporcionando uma experiência de aprendizado mais imersiva e significativa para todos os estudantes (Santos, Silva, Lima, Costa, Oliveira, Meroto, Narciso & Mendes, 2023b).

Além de promover a inclusão, as TDICs também contribuem para a inovação educacional, oferecendo novas abordagens pedagógicas que priorizam a personalização do ensino. Isso é possível através de ferramentas que permitem que o conteúdo seja ajustado de acordo com o progresso e o perfil de cada aluno, garantindo que eles avancem em seu próprio ritmo. Santos et al. (2024a) ressalta que essa flexibilidade é particularmente importante em um mundo educacional que valoriza cada vez mais o aprendizado centrado no estudante, e não apenas na transmissão de conhecimento pelo professor. A inovação e a acessibilidade tecnológica, portanto, caminham juntas para transformar o cenário educacional, garantindo que o ensino se torne cada vez mais democrático e eficiente.

Após discutir as tecnologias emergentes e as ferramentas de programação visual aplicadas no ensino de programação, é fundamental abordar os desafios e melhores práticas relacionados à implementação dessas tecnologias no ambiente escolar. A transição de uma abordagem teórica para uma aplicação prática enfrenta obstáculos, como a adaptação curricular, a formação docente e a infraestrutura escolar. A seguir, serão explorados esses desafios e as estratégias pedagógicas que podem facilitar a integração eficaz das ferramentas de programação visual, visando otimizar o processo de ensino-aprendizagem e aumentar o engajamento dos alunos.

DESAFIOS E MELHORES PRÁTICAS NA IMPLEMENTAÇÃO DE TECNOLOGIAS NO ENSINO DA PROGRAMAÇÃO

A implementação de tecnologias emergentes e ferramentas de programação visual no ensino da programação enfrenta diversos desafios que impactam tanto o ambiente escolar quanto o desempenho de educadores e alunos. A infraestrutura limitada, a falta de formação específica para os professores e a resistência à mudança são obstáculos recorrentes que dificultam a adoção efetiva dessas tecnologias nas escolas (Silva et al., 2023). Além disso, a adaptação dos currículos tradicionais para integrar essas ferramentas de forma significativa ainda é uma tarefa complexa para muitas instituições de ensino, exigindo mudanças estruturais que considerem as demandas pedagógicas (Santana & Martins, 2023).

Outro grande desafio é a necessidade de estratégias pedagógicas que aumentem o potencial das tecnologias, promovendo um ambiente de aprendizagem engajador e inclusivo. Miranda, Monteiro e Correia (2022) afirmam que as metodologias ativas, como aprendizagem baseada em projetos e a gamificação, têm aumentado o engajamento e a compreensão dos alunos. No entanto, a falta de preparo dos professores para aplicar essas metodologias com o suporte das tecnologias emergentes continua sendo uma barreira significativa, muitas vezes resultando em uma implementação superficial e pouco eficaz (Costa, Ferreira, Ditzz, Tosta, Silva, Simão & Silveira, 2023).

Esses desafios exigem uma maior compreensão das barreiras estruturais e pedagógicas enfrentadas por educadores e instituições. Superar essas dificuldades depende de investimentos em infraestrutura, formação continuada dos professores e apoio institucional que promova um ambiente colaborativo e aberto à inovação. A adoção de práticas pedagógicas adaptativas e o incentivo ao uso de metodologias ativas são fundamentais para tornar o ensino da programação mais acessível e alinhado às necessidades atuais, aumentando o impacto positivo das tecnologias emergentes no processo de ensino-aprendizagem.

Desafios Enfrentados Pelos Educadores: Infraestrutura e Resistência

Enfrentar os desafios para integrar as tecnologias emergentes e ferramentas de programação visual no ensino da programação é imperativo. A infraestrutura adequada e a resistência dos educadores se mostram os mais significativos. A infraestrutura tecnológica, que inclui desde computadores e dispositivos móveis até uma conexão de internet estável, é um fator determinante para a eficácia das ferramentas de programação visual no ambiente escolar (Santos, Pullen, Rodrigues, Figueirôa & Magalhães, 2023a). Muitos estabelecimentos de ensino, especialmente os públicos, sofrem com a falta de recursos tecnológicos básicos, o que inviabiliza a implementação de plataformas como *Scratch*, *Tynker* e *MIT App Inventor*, que dependem de suporte tecnológico adequado (Silva et al., 2023). Ainda, a manutenção e a atualização desses recursos frequentemente são negligenciadas, resultando em equipamentos obsoletos e que não atendem às necessidades do currículo moderno.

Para superar esses desafios, uma abordagem multifacetada, envolvendo investimento em infraestrutura, desenvolvimento profissional e apoio institucional é necessária. Primeiramente, é essencial que as escolas e os governos invistam na atualização dos equipamentos tecnológicos e na melhoria do acesso à internet, criando um ambiente propício para a integração das tecnologias emergentes no ensino (Costa et al., 2023).

Outro grande desafio é a resistência dos educadores à adoção dessas novas tecnologias. Apesar de as ferramentas de programação visual terem demonstrado potencial para tornar o ensino da programação mais acessível e envolvente, muitos professores ainda se sentem despreparados ou desconfortáveis para utilizá-las em suas aulas (Miranda et al., 2022). Grande parte dessa resistência à mudança se deve à falta de formação adequada. Muitos docentes possuem um histórico de formação focado em metodologias tradicionais e não foram treinados para incorporar tecnologias emergentes no seu dia a dia. Santana e Martins (2023) destacam que a falta de capacitação contínua e o medo do desconhecido levam muitos educadores a manterem práticas conservadoras, evitando o uso de novas ferramentas que poderiam enriquecer o processo de ensino-aprendizagem.

O tempo limitado que os professores têm para aprender e se adaptar a essas tecnologias é outro fator que contribui para a resistência. De acordo com Santos et al. (2023a), muitos educadores

expressam a necessidade de um suporte técnico mais robusto e programas de capacitação contínua que não apenas introduzam as ferramentas, mas também demonstrem como integrá-las de forma eficaz no ensino da programação.

Shimasaki e Prado (2021) ressaltam que as experiências práticas, como oficinas e laboratórios interativos, onde os educadores possam explorar as ferramentas de programação de visual em um ambiente de baixa pressão, são fundamentais para construir confiança e reduzir a resistência. Ainda, a criação de comunidades práticas dentro das escolas, onde os professores podem trocar ideias e dar suporte uns aos outros, pode facilitar a disseminação de melhores práticas e contribuir para a integração bem-sucedida das tecnologias. O apoio administrativo é de extrema importância, oferecendo além de recursos financeiros, mas também o incentivo necessário para que os educadores experimentem e adotem novas abordagens pedagógicas.

Além disso, é de extrema importância que as instituições de ensino ofereçam programas de formação continuada que sejam práticos e contextualizados, abordando não apenas o uso técnico das ferramentas, mas também estratégias pedagógicas que aumentem o impacto dessas tecnologias no aprendizado dos estudantes.

Os desafios enfrentados pelos docentes são grandes, mas não são insuperáveis. Com um investimento adequado em infraestrutura, uma abordagem contínua e colaborativa para a formação docente e o apoio institucional necessário, é possível superar as barreiras atuais e criar um ambiente de aprendizagem mais dinâmico, inclusivo e eficaz para todos os alunos. Reconhecer essas dificuldades e buscar soluções viáveis são passos fundamentais para o avanço na transformação do ensino da programação, preparando melhor os estudantes para as demandas do século XXI.

Adaptação Aos Currículos Tradicionais

A adaptação de currículos tradicionais para incluir tecnologias emergentes e ferramentas de programação visual no ensino da programação é um dos desafios mais complexos enfrentados pelas instituições educacionais. Tradicionalmente, os currículos escolares são estruturados com base em disciplinas convencionais, muitas vezes ignorando a necessidade de desenvolver competências digitais e habilidades de pensamento computacional nos alunos (Santana & Martins, 2023). A inclusão de ferramentas de programação visual, como o *Scratch* e o *MIT App Inventor*, requer uma reestruturação que vá além da simples adição de conteúdos. É necessário transformar a abordagem pedagógica e as estratégias de ensino para que essas tecnologias sejam integradas de forma significativa e não apenas como um complemento.

A carga de trabalho excessiva e a pressão para cumprir currículos extensos deixam pouco espaço para a experimentação com novas metodologias e ferramentas digitais. Muitos professores e administradores escolares ainda seguem abordagens tradicionais, focadas na transmissão de conhecimentos e no ensino expositivo, que muitas vezes não se alinham com as práticas interativas e exploratórias de ferramentas de programação visual (Costa et al., 2023). Para que haja sucesso na adaptação do currículo, é crucial que as instituições adotem uma abordagem abrangente, que inclua a formação contínua dos docentes e a criação de ambientes de aprendizagem que incentivem a experimentação e a inovação.

Miranda et al. (2022) afirma que a integração das tecnologias emergentes no currículo exige que as escolas passem a adotar metodologias ativas que promovam a participação ativa dos alunos no processo de aprendizagem. Estudos demonstram que práticas como a Aprendizagem Baseada em Projetos (ABProj) e o *design thinking* são bastante eficazes para engajar os estudantes e ajudá-los a aplicar os conceitos de programação de maneira prática e contextualizada. Essas abordagens permitem que o aluno desenvolva competências críticas como resolução de problemas, colaboração e criatividade, no entanto, a transição para essas metodologias requer grandes mudanças nos planos de aula, que muitas vezes enfrentam resistência devido à falta de tempo e recursos para desenvolvimento e implementação.

Além das metodologias ativas, a gamificação também tem se mostrado uma estratégia poderosa para integrar ferramentas de programação visual nos currículos tradicionais. A gamificação utiliza elementos de jogos, como pontuação, desafios e recompensas, para tornar o aprendizado mais

dinâmico e motivador. Plataformas como o *CodeCombat* e o *Tynker* têm sido particularmente eficazes em ensinar lógica de programação de maneira envolvente, ensinando conceitos abstratos em experiências interativas que os alunos conseguem compreender e aplicar. Mas, é necessário que os currículos sejam flexibilizados, permitindo essa abordagem mais lúdica, para que essas plataformas sejam utilizadas de maneira eficaz, o que muitas vezes exige negociações e ajustes com órgãos reguladores e com a gestão escolar.

Flexibilizar o currículo também envolve rever os critérios de avaliação, que precisam ser adaptados para refletir as novas competências desenvolvidas pelos alunos. As avaliações tradicionais, que são focadas apenas em testes e provas, não contemplam de maneira eficaz o aprendizado obtido através de atividades práticas e projetos realizados com ferramentas de programação visual. Shimasaki e Prado (2021) mostram que, neste contexto, a avaliação formativa, que considera o processo de aprendizagem e o desenvolvimento contínuo dos alunos se mostra mais adequada. Exemplos de práticas que podem ser integradas para acompanhar o progresso dos estudantes de forma mais alinhada com as metodologias ativas são avaliações baseadas em portfólios, autoavaliações e feedbacks contínuos.

O apoio institucional é outro aspecto fundamental para a adaptação curricular. Escolas que contam com uma gestão proativa e que valorizam a inovação tendem a ter mais sucesso na implementação de mudanças curriculares. É importante que as lideranças educacionais forneçam os recursos necessários, como tempo para planejamento, formação continuada e acesso a plataformas digitais, além de incentivar a cultura de colaboração entre os professores (Silva et al., 2023). Criar comunidades de prática dentro do ambiente escolar pode ser uma das estratégias eficientes para apoiar os educadores na troca de experiências e na busca conjunta por soluções para os desafios enfrentados durante a adaptação curricular.

Adaptar os currículos tradicionais para integrar tecnologias emergentes no ensino da programação é um processo complexo que envolve a superação de barreiras culturais, pedagógicas e estruturais. Embora os desafios sejam significativos, as oportunidades geradas por essas mudanças são igualmente grandes, permitindo que os alunos desenvolvam competências essenciais para o mundo digital. A combinação de metodologias ativas, estratégias de gamificação e avaliações mais inclusivas são passos fundamentais para transformar o ensino da programação e preparar os alunos para os desafios futuros. Formação docente e apoio institucional continuados são, portanto, indispensáveis para o sucesso dessa transformação.

Estratégias Pedagógicas e Melhores Práticas Para o Uso de Tecnologias

A implementação de tecnologias no ensino da programação não se limita apenas à inserção de novas ferramentas, mas envolve a adoção de estratégias pedagógicas inovadoras que potencializam o processo de aprendizagem. Silva et al. (2023) mostra que para o uso de tecnologias emergentes seja eficaz, é essencial que educadores empreguem práticas pedagógicas que favoreçam o engajamento dos alunos e promovam um ambiente de aprendizado colaborativo e interativo.

Uma das abordagens mais eficazes na integração de tecnologias no ensino da programação é o uso de metodologias ativas, como a Aprendizagem Baseada em Projetos (ABProj) e a Aprendizagem Baseada em Problemas (ABProb). Segundo Miranda et al. (2022), essas metodologias incentivam os alunos a trabalhar em projetos reais e a desenvolver projetos práticos, utilizando ferramentas de programação visual para criar soluções. Por exemplo, a utilização de plataformas como o *Scratch* e o *MIT App Inventor* permitem que os alunos construam aplicativos e jogos que resolvam problemas específicos, podendo, assim, aplicar os conceitos de programação de maneira contextualizada e prática. Essas estratégias, além de tornarem o aprendizado significativo, também desenvolvem habilidades críticas, como o pensamento computacional, criatividade e resolução de problemas.

Outra estratégia que se destaca no uso de tecnologias no ensino da programação é a personalização do ensino. Santana e Martins (2023) ressaltam que as ferramentas digitais permitem que o aprendizado seja adaptado ao ritmo e ao estilo de cada aluno, oferecendo atividades diferenciadas que atendem às necessidades individuais. Para alcançar a personalização, podemos utilizar plataformas adaptativas, que ajustam o nível de dificuldade dos exercícios baseados no desempenho do aluno, ou por

meio de tarefas personalizadas, onde os alunos escolhem projetos que melhor se encaixam em seus objetivos. Isso permite que cada aluno explore a programação de maneira que faça sentido para ele, respeitando seu ritmo de aprendizado.

Uma das práticas fundamentais no ensino da programação com tecnologias emergentes é o uso de feedback contínuo e formativo. Diferente das avaliações tradicionais, o feedback contínuo permite que os alunos recebam orientações durante o processo de aprendizado, corrigindo erros e aprimorando suas habilidades em tempo real (Shimasaki & Prado, 2021). Plataformas de programação visual frequentemente oferecem feedback imediato, destacando onde o aluno errou e sugerindo caminhos para correção, o que facilita a compreensão dos conceitos de programação e estimula o aprendizado autônomo. Esse tipo de feedback melhora o desempenho acadêmico e fortalece a autoconfiança dos estudantes, ao mostrar que os erros fazem parte do processo de aprendizado e que há sempre oportunidade para melhorar.

Outra estratégia eficaz amplamente utilizada para tornar o ensino da programação mais atraente e dinâmico é a gamificação. Essa estratégia incorpora elementos de jogos, como pontos, níveis e recompensas para motivar os alunos a se envolverem ativamente nas atividades de programação. Ferramentas como o *CodeCombat* e o *Tynker* transformam o aprendizado em uma experiência lúdica e envolvente, utilizando narrativas e desafios baseados em jogos para ensinar programação. Essa abordagem é especialmente eficaz para capturar o interesse dos estudantes que possam inicialmente considerar a programação uma disciplina desafiadora e pouco atrativa.

Para que o uso das tecnologias no ensino da programação seja eficaz, é necessário que a colaboração e o trabalho em equipe sejam incentivados. Costa et al., (2023) afirma que as plataformas digitais frequentemente oferecem recursos que permitem aos alunos trabalhar juntos em projetos, compartilhando ideias e soluções. Colaborar em atividades não é melhora a compreensão dos conceitos de programação, mas também desenvolve habilidades sociais e de comunicação, preparando os estudantes para o trabalho em ambientes de tecnologia que frequentemente envolvem equipes multidisciplinares.

Para aumentar o impacto das tecnologias no ensino da programação, é crucial que os educadores também estejam continuamente atualizados sobre as novas ferramentas e estratégias pedagógicas. A formação continuada e o desenvolvimento profissional são essenciais para capacitar os professores a integrar as tecnologias de maneira eficaz em suas práticas pedagógicas (Silva et al., 2023). Programas de capacitação que envolvem oficinas práticas são particularmente úteis. Além disso, a criação de redes de colaboração entre professores, onde possam compartilhar experiências e recursos, ajuda a disseminar melhores práticas e a superar desafios comuns na implementação das tecnologias.

Visto isso, as estratégias e melhores práticas para o uso das tecnologias no ensino da programação são essenciais para garantir que essas ferramentas sejam utilizadas de maneira eficaz e significativa. Algumas das abordagens que têm se mostrado eficazes na integração das tecnologias no ensino são as metodologias ativas, personalização do ensino, feedback contínuo, gamificação e a colaboração entre os estudantes. O sucesso dessas práticas não depende somente do acesso às ferramentas digitais, mas também do apoio institucional e da formação contínua dos educadores, que devem ser capacitados e incentivados a explorar novas abordagens pedagógicas. Ao adotar essas estratégias, as escolas podem transformar o ensino da programação, tornando-o mais engajador, inclusivo e preparado para atender às demandas da educação no século XXI.

RESULTADOS E DISCUSSÃO

A utilização das tecnologias emergentes no ensino da programação proporciona uma série de benefícios e desafios. Anteriormente, foi discutida a relevância de ferramentas de programação visual para facilitar a compreensão de conceitos complexos de programação por parte dos alunos. Integrar essas ferramentas não promove somente o pensamento computacional, mas também possibilita novas formas de engajamento por meio de plataformas interativas e gamificação (Agner, Thomen, Utzig & Soares, 2021).

A análise dos principais resultados da pesquisa inicia-se com a apresentação dos impactos das ferramentas de programação visual na compreensão conceitual dos alunos, em seguida, discute-se o

efeito dessas tecnologias no engajamento dos estudantes e posteriormente, são abordados os desafios enfrentados pelos educadores na implementação dessas ferramentas e as adaptações necessárias nos currículos escolares. Finalmente, é feita uma síntese dos resultados, pensando em como as ferramentas de programação visual podem melhorar a compreensão conceitual, o desempenho e o engajamento dos alunos no ensino de programação para o Ensino Médio.

Contribuições das Ferramentas de Programação Visual na Compreensão Conceitual dos Alunos

Conforme observado no resultado da pesquisa, as ferramentas de programação visual desempenham um papel fundamental na melhoria da compreensão conceitual dos alunos. Tecnologias como *Scratch* e *Tynker* facilitam o aprendizado de conceitos abstratos de programação, como *loops*, estruturas de dados e controle de fluxo. Isso ocorre porque essas plataformas permitem que os alunos manipulem blocos visuais e vejam, de maneira imediata, os efeitos das instruções que programam. Dessa forma, os alunos podem construir seu próprio conhecimento de forma prática, participando ativamente do processo de aprendizagem.

Conforme a pesquisa de Grover e Pea (2013), o uso de ferramentas de programação visual, como o *Scratch*, facilita a compreensão de conceitos complexos de programação ao permitir que os alunos manipulem blocos visuais, promovendo um ambiente mais acessível para iniciantes. Além disso, as interações visuais, que proporcionam feedback imediato, permitem que os alunos construam seu próprio conhecimento ao experimentar os blocos de código e observar o resultado.

Estudos como os de Resnik, Maloney, Monroy- Hernández, Rusk, Eastmond, Brennan, Millner, Rosenbaum, Silver, Silverman e Kafai (2009), mostram que o aprendizado prático promove uma compreensão mais profunda dos conceitos de programação, particularmente para alunos do Ensino Médio, que estão em fase de desenvolvimento cognitivo e se beneficiam de abordagens interativas. Essas ferramentas reduzem a carga cognitiva associada ao aprendizado inicial de linguagens de programação baseadas em texto, como *Python*, ao fornecer uma abordagem amigável e intuitiva. Isso facilita a internalização dos conceitos e cria um ambiente de aprendizagem no qual os alunos podem se sentir mais confiantes para experimentar e errar, o que é uma parte crucial do aprendizado da programação.

Além disso, a programação visual promove o aprendizado através de experimentação e erro, possibilitando que os alunos testem suas soluções e as ajustem suas possíveis falhas em tempo real. Esse processo reforça o entendimento conceitual dos estudantes, uma vez que eles podem imediatamente verificar o impacto de suas escolhas no resultado final do programa (Resnick et al., 2009).

Logo, os resultados sugerem que as ferramentas de programação visual têm reflexo positivo na compreensão conceitual dos alunos, fornecendo uma base sólida para o aprendizado de programação no Ensino Médio.

Influência das Ferramentas de Programação Visual no Engajamento dos Alunos

Além de melhorar a compreensão dos conceitos, o uso de ferramentas de programação visual tem causado um impacto positivo no engajamento dos alunos. Com as análises mostradas por esse trabalho, é possível ver que plataformas que integram gamificação aumentam o interesse dos estudantes em aprender programação. Ao introduzir conceitos lúdicos e competitivos no processo de aprendizado, a gamificação transforma a programação em uma atividade envolvente e divertida, mantendo os alunos motivados ao longo das aulas, como reforçado por Hamari, Koivisto e Sarsa (2014).

A gamificação se mostrou eficaz para aumentar a motivação dos alunos, incentivando-os a se envolverem mais ativamente no aprendizado e a participar de atividades colaborativas, o que, por sua vez, aumenta a retenção de conhecimento. A competição saudável gerada por essas atividades promove uma maior interação entre os alunos e cria um ambiente de aprendizagem colaborativo, onde eles discutem soluções e estratégias, conforme observado por Barab, Gresalfi e Ingram-Goble (2010). Esse ambiente interativo é essencial para manter os alunos engajados e interessados em temas que, de outra forma, poderiam ser considerados difíceis ou desinteressantes.

Os resultados vão de encontro com as teorias de Gee (2008), que sugere que a gamificação tem a capacidade de transformar o processo de aprendizagem, tornando-o mais dinâmico e envolvente. Quando se trata do ensino de programação, a gamificação ajuda a aumentar a retenção de conhecimento, uma vez que os alunos aplicam os conceitos aprendidos de forma significativa e prática. Além disso, com a promoção da competição saudável e o aprendizado colaborativo, as ferramentas de gamificação incentivam o desenvolvimento de habilidades interpessoais, como a comunicação e o trabalho em equipe.

Logo, conclui-se que as ferramentas de programação visual que integram elementos de gamificação não apenas aumentam o nível de engajamento dos alunos, mas também facilitam a aprendizagem ativa, contribuindo para um ambiente de aprendizado mais estimulante e eficaz.

Desafios Enfrentados Pelos Educadores e Adaptação Curricular

Apesar dos benefícios observados, os educadores enfrentam desafios consideráveis para implementar tecnologias no ensino de programação. Um dos principais obstáculos identificados é a falta de infraestrutura adequada nas escolas, especialmente nas instituições públicas. A ausência de computadores e o acesso limitado à internet são problemas que dificultam a utilização de plataformas de programação visual. Viana e Portela (2019) destacam que, sem os recursos tecnológicos necessários, o potencial dessas ferramentas para transformar o ensino de programação fica severamente limitado.

Além disso, a pesquisa mostra que muitos professores não se sentem preparados para integrar essas tecnologias em suas práticas pedagógicas. A falta de formação continuada e a ausência de políticas educacionais que incentivem o uso de tecnologias emergentes no ensino de programação são fatores que contribuem para essa resistência. A pesquisa de Kafai e Burke (2014) corrobora com essas afirmações, indicando que muitos educadores enfrentam desafios significativos para implementar ferramentas de programação visual nas escolas. Entre os principais obstáculos estão a falta de infraestrutura adequada e a ausência de formação continuada para os professores.

Superar esses desafios requer um investimento significativo por parte das instituições de ensino e dos governos. É essencial que haja uma melhoria na infraestrutura tecnológica das escolas, garantindo que todos os alunos tenham acesso a computadores e à internet de qualidade. Além disso, a formação continuada dos professores deve ser uma prioridade. Programas de treinamento voltados para o uso de tecnologias emergentes no ensino de programação podem ajudar os educadores a se familiarizarem com essas ferramentas e a incorporá-las em suas aulas.

Outro aspecto fundamental é a adaptação dos currículos escolares para incluir disciplinas de programação e promover a interdisciplinaridade. Grover e Pea (2013) observam que muitos currículos escolares ainda não foram adaptados para incluir disciplinas de programação. A falta de uma abordagem interdisciplinar e a resistência à mudança dos métodos tradicionais de ensino são fatores que limitam o uso eficaz dessas tecnologias em sala de aula. Para que essa barreira seja superada, o currículo deve ser revisado para incluir disciplinas que abordem programação de forma significativa, promovendo a interdisciplinaridade e preparando os alunos para os desafios tecnológicos do futuro.

Portanto, investimentos em infraestrutura tecnológica e em programas de capacitação docente são essenciais para garantir que as ferramentas de programação visual possam ser implementadas de forma eficaz nas escolas.

Efeitos das Ferramentas de Programação Visual no Ensino de Programação

Os resultados obtidos por este estudo indicam que as ferramentas de programação visual, como o *Scratch*, têm o potencial de melhorar significativamente a compreensão conceitual e o engajamento dos alunos no ensino de programação. Ao permitir que os alunos aprendam de forma prática e interativa, essas ferramentas ajudam a internalizar conceitos complexos de programação, ao mesmo tempo que tornam o processo de aprendizagem mais dinâmico e envolvente. No entanto, o sucesso da implementação dessas ferramentas depende fortemente da superação de desafios relacionados à infraestrutura e à capacitação docente.

Ainda, os resultados sugerem que essas ferramentas são eficazes em seus propósitos desde que sejam implementadas em um ambiente com suporte tecnológico adequado e com professores devidamente capacitados. Além disso, a adaptação curricular é essencial para garantir que as ferramentas de programação visual possam ser integradas de forma significativa ao ensino, preparando os alunos para as demandas do século XXI.

Ademais, as ferramentas de programação visual proporcionam um ensino de programação mais inclusivo e acessível ao simplificar a complexidade inicial do aprendizado, permitindo que estudantes de diferentes perfis e níveis de habilidade se envolvam de maneira prática e direta. Ao utilizar blocos visuais em vez de sintaxes complexas, essas plataformas tornam a programação mais intuitiva e menos intimidadora, eliminando barreiras que tradicionalmente afastam iniciantes e estudantes de contexto com menor acesso a recursos tecnológicos. Essa abordagem facilita a entrada de alunos com diferentes necessidades de aprendizado e estilos cognitivos, possibilitando que todos possam progredir e experimentar o desenvolvimento de habilidades computacionais.

CONSIDERAÇÕES FINAIS

Ao longo deste trabalho foi possível analisar o papel das ferramentas de programação visual no ensino de programação para estudantes do Ensino Médio. Essa pesquisa demonstrou que essas tecnologias possuem grande potencial para melhorar a compressão de conceitos, o desempenho acadêmico e o engajamento dos alunos. Ferramentas como o *Scratch*, *Tynker*, *Alice* e outros, facilitaram o aprendizado de conceitos complexos de programação por meio de abordagens visuais e interativas, sendo capazes de traduzir conceitos abstratos em elementos concretos, fator crucial para o sucesso dessas plataformas no ambiente educacional.

Ao utilizar essas ferramentas, os estudantes podem explorar conceitos de forma prática e personalizada. Em um contexto em que os alunos apresentam diferentes níveis de conhecimento e ritmos de aprendizado, essas plataformas se destacam por proporcionar uma experiência adaptativa. A progressão no aprendizado, no ritmo de cada estudante, acompanhada por feedbacks imediatos, contribui para uma maior retenção do conhecimento. Assim, essas ferramentas promovem, além do aumento no engajamento, uma maior autonomia no processo de aprendizado, permitindo que os alunos se tornem mais protagonistas de sua própria educação.

A pesquisa realizada reforçou a importância das ferramentas de programação visual como parte do ensino da programação. Estas tecnologias emergentes permitem aos estudantes construir seu conhecimento de forma colaborativa e prática, rompendo com o modelo tradicional de ensino, que muitas vezes apresenta a programação de maneira mais abstrata e desafiadora. Aplicativos como o *Scratch* e o *Alice*, que utilizam a gamificação, tem esse conceito como elemento chave para aumentar o engajamento dos alunos, transformando o aprendizado em uma experiência lúdica e interativa. Esse ambiente dinâmico possibilita que os estudantes testem e apliquem seus conhecimentos em tempo real, tornando o processo de aprendizado mais acessível, mesmo para aqueles que não possuem familiaridade com os conceitos de programação.

No entanto, apesar de possuir inúmeras vantagens, também existem desafios ao aplicar essas tecnologias nas escolas. A infraestrutura inadequada e a falta de formação específica para educadores são os principais obstáculos encontrados. Muitas instituições, especialmente em regiões com menos acesso à tecnologia, não possuem os equipamentos e recursos necessários para garantir o uso dessas ferramentas. Além disso, muitos professores ainda se sentem desconfortáveis com o uso de tecnologias digitais em sala de aula, seja pela falta de familiaridade com as ferramentas ou pela falta de suporte adequado. A resistência à mudança por parte dos educadores é outro fator que dificulta a integração dessas plataformas no ensino regular.

A adaptação dos currículos escolares tradicionais é outro grande desafio enfrentado na implementação de tecnologias no ensino. A integração de ferramentas de programação visual exige uma revisão dos conteúdos programáticos e mudanças nas práticas pedagógicas. A adoção de metodologias mais interativas e centradas no aluno é essencial para maximizar os benefícios dessas ferramentas. No entanto, essa adaptação exige um esforço mútuo entre educadores, instituições de ensino e governos, para criar um ambiente propício à inovação no ensino da programação.

Para que esses desafios sejam superados, é de extrema importância que se invista em formação continuada para professores, garantindo que eles adquiram as competências necessárias para utilizar essas ferramentas de forma eficaz. Ainda, é essencial que haja um investimento contínuo em infraestrutura tecnológica nas escolas, assegurando o acesso a dispositivos adequados e a uma conexão de internet estável. Esses investimentos são fundamentais para que os estudantes possam usufruir plenamente das vantagens oferecidas pelas ferramentas de programação visual e para que os educadores possam aplicá-las em suas práticas pedagógicas.

Com base neste trabalho, é possível pensar em um futuro onde a educação, cada vez mais, amplia o uso de ferramentas de programação visual, não apenas no Ensino Médio, mas também em outros níveis educacionais. Essas plataformas podem ser adaptadas para diferentes contextos e complexidades, possibilitando que alunos de diferentes idades e graus de conhecimento se beneficiem do aprendizado de programação. Além disso, o desenvolvimento de novas tecnologias que integram inteligência artificial e aprendizado adaptativo pode potencializar ainda mais a personalização do aprendizado, permitindo que os estudantes recebam uma educação personalizada para suas necessidades individuais.

Este trabalho contribui de forma significativa para a compreensão do papel das ferramentas de programação visual no ensino da programação, especialmente no contexto do Ensino Médio. A revisão bibliográfica forneceu uma visão abrangente sobre as principais ferramentas disponíveis, identificando suas melhores práticas e desafios. Ao fornecer uma análise sobre como essas tecnologias podem ser integradas ao currículo escolar, esta pesquisa serve como recurso para educadores que buscam aprimorar suas práticas pedagógicas com o uso de ferramentas interativas e gamificadas.

Em suma, o uso de ferramentas de programação visual no Ensino Médio oferece uma oportunidade única de transformar o processo de aprendizado da programação. Em um mundo cada vez mais digital, no qual as habilidades de programação são essenciais para o sucesso profissional, essas tecnologias emergentes têm o potencial de democratizar o acesso ao ensino de programação, tornando-o mais acessível, interativo e inclusivo. Para que esse potencial seja aplicado, no entanto, é necessário enfrentar os desafios relacionados à infraestrutura, à formação dos professores e à adaptação dos currículos escolares. Ao investir nessas áreas, será possível criar um ambiente educacional mais eficaz e alinhado às demandas do século XXI.

Para pesquisas futuras, recomenda-se o aprofundamento em temas que explorem o impacto de diferentes abordagens pedagógicas combinadas com ferramentas de programação visual, como a gamificação, no desenvolvimento de habilidades de resolução de problemas e pensamento crítico entre estudantes do Ensino Médio. Investigar como essas ferramentas podem ser úteis em contextos diversos, especialmente em escolas com recursos limitados, poderia fornecer insights sobre a adaptabilidade e escalabilidade dessas tecnologias no ensino da programação. Além disso, estudos que analisem o papel da inteligência artificial e do aprendizado adaptativo na personalização do aprendizado de programação visual podem contribuir para o desenvolvimento de novas metodologias que atendam ainda melhor às necessidades individuais dos alunos.

REFERÊNCIAS BIBLIOGRÁFICAS

Agner, L. T. W., Thomen, M. A. F., Utzig, G. M. & Soares, I. W. (2021). Educação em Algoritmos e Estruturas de Dados no Contexto do Ensino Baseado em Jogos Digitais. *Revista Iberoamericana de Tecnología Em Educación Y Educación Em Tecnología*. <https://doi.org/10.24215/18509959.30.e9>

Barab, S. A., Gresalfi, M. & Ingram-Goble, A. (2010). Transformational Play: Using games to position person, content, and context. *Educational Researcher*. <https://doi.org/10.3102/0013189X10386593>

Boff, E., Salami, A. D. G. & Miotto, F. (2023). Programação de Jogos Digitais Educativos com Scratch a Partir do Método Design Thinking. *Caxias do Sul, RS: Scientia cum Industria*. <https://doi.org/10.18226/23185279.e231204>

Cetic.br. (2023). Pesquisa Sobre o Uso das Tecnologias de Informação e Comunicação nas Escolas Brasileiras: TIC educação 2022. Comitê Gestor da Internet no Brasil. Disponível em <https://cetic.br/pt/publicacao/pesquisa-sobre-o-uso-das-tecnologias-de-informacao-e-comunicacao-nas-escolas-brasileiras-tic-educacao-2022/> Acessado em 10 de setembro de 2024.

Coelho, I. R., Melo, G. A., Rosa, A. S. & Farias, A.A. (2023). Gamificação na Sala de Aula: O uso do Scratch como metodologia ativa para o ensino de programação. Joinville, SC: 9º Seminário de Ensino, Pesquisa, Extensão e Inovação do IFSC. Disponível em <https://proceedings.science/sepei-2023/trabalhos/gamificacao-na-sala-de-aula-o-uso-do-Scratch-como-metodologia-ativa-para-o-ensin?lang=pt-br#> Acessado em 05 de setembro de 2024.

Costa, C. F., Ferreira, O. S., Ditzz, Á. J. M., Tosta, C. S. V., Silva, C. A. S., Simão, F. P. & Silveira, P. D. N. (2023). Gamificação como Auxílio ao Desenvolvimento da Lógica de Programação para Alunos de Cursos de Informática. Cuadernos de Educación y Desarrollo. <https://doi.org/10.55905/cuadv15n4-041>

Costa, N. C., Lima, J. R. S., Almeida, G. K. F. C., Magalhães, Y. C. & Almeida, W. R. M. (2022). O Uso da Plataforma Scratch como Ferramenta Facilitadora Durante o Ensino de Lógica de Programação para Alunos do Ensino Médio. Brazilian Journal of Development. <https://doi.org/10.34117/bjdv8n8-286>

Freitas, T. T. (2023). O Uso da Gamificação como Estratégia para o Ensino de Programação Orientada a Objetos: Uma revisão sistemática de literatura. Russas, CE: Universidade Federal do Ceará. Disponível em <https://repositorio.ufc.br/handle/riufc/75490> Acessado em 06 de setembro de 2024.

Fundação Telefônica Vivo. (2022). Pesquisa TIC Educação 2022. Disponível em <https://www.fundacaotelefonicavivo.org.br/noticias/pesquisa-tic-educacao-2022/> Acessado em 21 de outubro de 2024.

Gee, J. P. (2008). Learning and Games. Cambridge, MA: The MIT Press. Disponível em <https://onlineteachered.mit.edu/files/games-and-learning/week-1/Gee-Chapter-Ecology-of-Games.pdf> Acessado em 14 de outubro de 2024.

Gil, A. C. (2010). Métodos e Técnicas de Pesquisa Social. São Paulo: Atlas.

Hwang, G. J., Sung, H. Y., Hung, C. M. & Huang, I. (2020) A Learning Style Perspective to Investigate the Necessity of Developing Adaptive Learning Systems. Educational Technology & Society. Disponível em https://www.researchgate.net/publication/279711154_A_Learning_Style_Perspective_to_Investigate_the_Necessity_of_Developing_Adaptive_Learning_Systems Acessado em 08 de setembro de 2024.

Grover S. & Pea, R. (2013). Computational Thinking in K-12: A review of the state of the field. Educational Researcher. <https://doi.org/10.3102/0013189X12463051>

Hamari, J., Koivisto, J. & Sarsa, H. (2014). Does Gamification Work? A literature review of empirical studies on gamification. Hawaii international conference on system sciences. IEEE. <https://doi.org/10.1109/HICSS.2014.377>

Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP). (2022). Conheça o Panorama da Conectividade na Educação Básica. Ministério da Educação. Disponível em <https://www.gov.br/inep/pt-br/assuntos/noticias/censo-escolar/conheca-o-panorama-da-conectividade-na-educacao-basica> Acessado em 21 de outubro de 2024.

Kafai, Y. B. & Burke, Q. (2014). *Connected Code: Why Children Need to Learn Programming*. The MIT Press. <https://doi.org/10.7551/mitpress/9992.001.0001>

Kitchenham, B. & Charters, S. (2007). *Guidelines for Performing Systematic Literature Reviews in Software Engineering*. EBSE Technical Report. Disponível em https://www.researchgate.net/publication/302924724_Guidelines_for_performing_Systematic_Literature_Reviews_in_Software_Engineering Acessado em 10 de setembro de 2024.

Lüdke, M. & André, M. E. D. A. (1986). *Pesquisa em Educação: Abordagens qualitativas*. São Paulo: EPU. Disponível em https://hugoribeiro.com.br/area-restrita/Ludke_Andre-Pesquisa_Educaca_abordagens_qualitativas.pdf Acessado em 12 de setembro de 2024.

Lyra, P. H. C., Katakura, P. & Martins, A. C. (2022). *A Experiência com Gêmeos Digitais: Integração do BIM com o IoT*. São Caetano do Sul, SP: Anais do ENEBIM 2022 – Encontro Nacional de Ensino de BIM. <https://doi.org/10.46421/enebim.v4i00.1937>

Madureira, J. S. & Schneider, H. N. (2021). *Gamificação no Ensino de Programação de Computadores em Turmas do Ensino Médio: Uma experiência com o software Kahoot!*. *Revista Novas Tecnologias na Educação*. <https://doi.org/10.22456/1679-1916.121191>

Marconi, M. A. & Lakatos, E. M. (2003). *Fundamentos de Metodologia Científica*. São Paulo: Atlas. Disponível em https://docente.ifrn.edu.br/olivianeta/disciplinas/copy_of_historia-i/historia-ii/china-e-india/view Acessado em 28 de setembro de 2024.

Menegotto, J. L. & Torres, J. C. B. (2019). *Integração de Simulador Acústico em Ferramentas de Modelagem IBM*. PARC Pesquisa em Arquitetura e Construção. <https://doi.org/10.20396/parc.v10i0.8653934>

Miranda, M. R. M., Monteiro, W. M. & Correia, F. P. (2022). *Metodologias Ativas: Ressignificando o ensino de programação para estudantes do Ensino Médio*. *Brazilian Journal of Development*. <https://doi.org/10.34117/bjdv8n12-022>

Prodanov, C. C. & Freitas, E. C. (2013). *Metodologia do Trabalho Científico: Métodos e técnicas da pesquisa e do trabalho acadêmico*. Novo Hamburgo, RS: Feevale. Disponível em <https://www.feevale.br/Comum/midias/0163c988-1f5d-496f-b118-a6e009a7a2f9/E-book%20Metodologia%20do%20Trabalho%20Cientifico.pdf> Acessado em 01 de outubro de 2024.

Resnik, M., Maloney, J. H., Monroy-Hernández, A. Rusk, N., Eastmond, E., Brennan, K. A., Millner, A., Rosenbaum, E., Silver, J. S., Silverman, B. S. & Kafai, Y. B. (2009). *Scratch: Programming for all*. *Communications of the ACM*. <https://doi.org/10.1145/1592761.1592779>

Santana, A. G. L. & Martins, I. C. (2023). *A Gamificação como Ferramenta Virtual para Aprendizagem de Estudantes no Ensino Fundamental: Uma revisão*. São José dos Pinhais, PR: *Contribuciones a Las Ciencias Sociales*. <https://doi.org/10.55905/revconv.16n.11-021>

Santos, S. M. A. V., Caetano, A. P. S. G., Araujo, C. S., Costa, E. J., Melo Júnior, H. G., Graf, L., Silva, M. A. & Santos, R. (2024a). *O Uso das Tecnologias Digitais da Informação e Comunicação (TDICS) na Escola Contemporânea*. *Contribuciones a Las Ciencias Sociales*. <https://doi.org/10.55905/revconv.17n.1-274>

Santos, S. M. A. V., Pullen, F. C. S., Rodrigues, F. F., Figueirôa, L. M. & Magalhães, M. S. (2023a). Avaliando o Uso das Tecnologias Digitais na Educação. *Revista Ilustração*. <https://doi.org/10.46550/ilustracao.v4i6.217>

Santos, S. M. A. V., Santos, F. M., Melo Júnior, H. G., Penha, M. C. S. M., Ramalho, R. A., Moniz, S. S. O. R. & Oliveira, Z. M. (2024b). Educação para a Geração Digital: Desafios e estratégias. *Revista Foco*. <https://doi.org/10.54751/revistafoco.v17n1-040>

Santos, S. M. A. V., Silva, C. K., Lima, G. P. T., Costa, J. W. M., Oliveira, J. P., Meroto, M. B. N., Narciso, R. & Mendes, S. A. F. (2023b). O Efeito, as Direções e Abordagens dos Recursos Multimídia na Educação para uma Aprendizagem Imersiva e Individualizada. *Revista Foco*. <https://doi.org/10.54751/revistafoco.v16n11-237>

Severino, A. J. (2013). *Metodologia do Trabalho Científico*. São Paulo: Cortez. Disponível em https://www.ufrb.edu.br/ccaaab/images/AEPE/Divulga%C3%A7%C3%A3o/LIVROS/Metodologia_do_Trabalho_Cient%C3%ADfico_-_1%C2%AA_Edi%C3%A7%C3%A3o_-_Antonio_Joaquim_Severino_-_2014.pdf Acessado em 01 de outubro de 2024.

Shimasaki, R. & Prado, M. E. B. B. (2021). O Ensino da Programação e o Desenvolvimento do Pensamento Lógico: Uma revisão sistemática de literatura. *Revista de Ensino, Educação e Ciências Humanas*. <https://doi.org/10.17921/2447-8733.2021v22n2p197-205>

Silva, T. S., Matos, I. C., Libório Filho, J. M. & Barbosa, L. S. O. (2023). Estágio Supervisionado em Computação II: O uso da plataforma MIT App Inventor como auxílio no aprendizado da lógica de programação nas séries iniciais do Ensino Médio-técnico. São Paulo: *Revista Ibero-Americana de Humanidades*. <https://doi.org/10.51891/rease.v9i9.11515>

Sousa, A. S., Oliveira, G. S. & Alves, L. H. (2021). *A Pesquisa Bibliográfica: Princípios e fundamentos. Cadernos da Fucamp*. Disponível em <https://revistas.fucamp.edu.br/index.php/cadernos/article/view/2336> Acessado em 29 de setembro de 2024.

Souza, M. T., Silva, M. D. & Carvalho, R. (2010). *Revisão integrativa: o que é e como fazer*. São Paulo: Einstein. Disponível em <https://www.scielo.br/j/eins/a/ZQTBkVJZqcWrTt34cXLjtBx/abstract/?lang=pt> Acessado em 10 de setembro de 2024.

Viana, G. A. & Portela, C. S. (2019). O Uso de Softwares Educativos para Introdução de Lógica de Programação no Ensino de Base e Superior. *Informática na Educação: Teoria & Prática*. <https://doi.org/10.22456/1982-1654.86079>

CONTRIBUIÇÃO DAS AUTORAS

Autora 1 – Responsável pela estruturação da pesquisa, redação, revisão e formatação do texto.

Autora 2 – Orientadora de mestrado da 1ª autora, responsável pelas orientações e correções de todas as etapas de construção do texto.

DECLARAÇÃO DE CONFLITO DE INTERESSE

As autoras declaram que não há conflito de interesse com o presente artigo.

Este preprint foi submetido sob as seguintes condições:

- Os autores declaram que estão cientes que são os únicos responsáveis pelo conteúdo do preprint e que o depósito no SciELO Preprints não significa nenhum compromisso de parte do SciELO, exceto sua preservação e disseminação.
- Os autores declaram que os necessários Termos de Consentimento Livre e Esclarecido de participantes ou pacientes na pesquisa foram obtidos e estão descritos no manuscrito, quando aplicável.
- Os autores declaram que a elaboração do manuscrito seguiu as normas éticas de comunicação científica.
- Os autores declaram que os dados, aplicativos e outros conteúdos subjacentes ao manuscrito estão referenciados.
- O manuscrito depositado está no formato PDF.
- Os autores declaram que a pesquisa que deu origem ao manuscrito seguiu as boas práticas éticas e que as necessárias aprovações de comitês de ética de pesquisa, quando aplicável, estão descritas no manuscrito.
- Os autores declaram que uma vez que um manuscrito é postado no servidor SciELO Preprints, o mesmo só poderá ser retirado mediante pedido à Secretaria Editorial do SciELO Preprints, que afixará um aviso de retratação no seu lugar.
- Os autores concordam que o manuscrito aprovado será disponibilizado sob licença [Creative Commons CC-BY](#).
- O autor submissor declara que as contribuições de todos os autores e declaração de conflito de interesses estão incluídas de maneira explícita e em seções específicas do manuscrito.
- Os autores declaram que o manuscrito não foi depositado e/ou disponibilizado previamente em outro servidor de preprints ou publicado em um periódico.
- Caso o manuscrito esteja em processo de avaliação ou sendo preparado para publicação mas ainda não publicado por um periódico, os autores declaram que receberam autorização do periódico para realizar este depósito.
- O autor submissor declara que todos os autores do manuscrito concordam com a submissão ao SciELO Preprints.