Lógica de programação através da robótica: Uso do Scratch e Arduino para criação de robôs e projetos interativos

https://doi.org/10.56238/levv15n39-061

Eduka Bytes

RESUMO

Este artigo investiga o ensino da lógica de programação através da robótica, utilizando Scratch e Arduino para a criação de robôs e projetos interativos. Através de uma revisão bibliográfica, examinamos as vantagens e desafios dessa abordagem, destacando como a integração de ferramentas de programação visual e hardware pode enriquecer o aprendizado dos estudantes. A integração do Scratch e do Arduino na educação surge como uma abordagem inovadora para o ensino de lógica de programação. O Scratch, uma linguagem de programação visual, e o Arduino, uma plataforma de eletrônica de código aberto, oferecem aos alunos um ambiente de aprendizado dinâmico e interativo. Através de projetos práticos e feedback em tempo real, os alunos podem explorar conceitos abstratos de maneira tangível, aprimorando sua compreensão e retenção de conhecimento. Apesar dos desafios, como a formação de professores e a disponibilidade de recursos, os benefícios do uso do Scratch e do Arduino superam as dificuldades, proporcionando aos alunos experiências de aprendizado envolventes e significativas. Este artigo revisa a literatura sobre o uso do Scratch e do Arduino na educação, destacando suas contribuições para a educação em programação e aprendizado baseado em projetos. Os resultados indicam que o uso de Scratch e Arduino promove um ambiente de aprendizado dinâmico e envolvente, facilitando a compreensão de conceitos de programação e eletrônica. Futuras pesquisas devem focar em estratégias para melhorar a formação de professores e expandir o acesso a essas ferramentas, garantindo que todos os alunos tenham a oportunidade de desenvolver habilidades essenciais de alfabetização digital. Ao capacitar os alunos a se tornarem criadores e inovadores, o Scratch e o Arduino abrem caminho para um futuro onde a tecnologia não apenas é compreendida, mas também é ativamente moldada e utilizada para mudanças positivas.

Palavras-chave: Scratch, Arduino, Educação em Programação, Aprendizagem Baseada em Projetos, Formação de Professores.

1 INTRODUÇÃO

A crescente importância das habilidades de programação no mundo atual tem impulsionado a busca por métodos de ensino inovadores e eficazes. O ensino da programação é considerado essencial para o desenvolvimento de competências do século XXI, como pensamento crítico, resolução de problemas e criatividade (Wing, 2006). Nesse contexto, a robótica educacional, utilizando plataformas como *Scratch* e Arduino, surge como uma solução promissora para engajar estudantes e facilitar a aprendizagem da lógica de programação.

O *Scratch* é uma linguagem de programação visual desenvolvida pelo *MIT Media Lab*, que permite aos usuários criar programas através da montagem de blocos, facilitando a compreensão de conceitos fundamentais de programação (Resnick et al., 2009). O Arduino, por sua vez, é uma plataforma de prototipagem eletrônica de código aberto que permite a criação de projetos interativos ao combinar *software* e *hardware* (Banzi & Shiloh, 2014). A combinação dessas duas ferramentas possibilita a construção de robôs e projetos interativos, proporcionando uma aprendizagem prática e contextualizada.

Este artigo revisa a literatura existente sobre o uso de *Scratch* e Arduino na educação, destacando suas contribuições para o desenvolvimento de projetos interativos e robôs. A abordagem prática e visual oferecida por essas ferramentas pode facilitar a compreensão de conceitos complexos de programação e eletrônica, além de promover um aprendizado ativo e engajador.

1.1 JUSTIFICATIVA

A integração de *Scratch* e Arduino no ensino de programação oferece uma abordagem prática e visual, que pode facilitar a compreensão de conceitos complexos. A aprendizagem baseada em projetos (*Project-Based Learning* - PBL), especialmente em robótica, tem se mostrado eficaz para aumentar o interesse dos alunos e melhorar suas habilidades de resolução de problemas (Krajcik & Blumenfeld, 2006). Estudos (QUAIS???) indicam que a robótica educacional promove a aprendizagem ativa, onde os alunos aplicam conhecimentos teóricos em contextos práticos, desenvolvendo competências essenciais para o século XXI (Mataric et al., 2007).

Além disso, a robótica educacional pode ajudar a superar a barreira de abstração frequentemente associada ao ensino de programação, tornando os conceitos mais tangíveis e compreensíveis. Ao verem seus códigos ganharem vida em robôs físicos, os alunos conseguem visualizar e compreender melhor as consequências de suas ações programáticas (Bers, 2010). Isso é particularmente importante para jovens estudantes, pois podem se sentir desmotivados frente aos métodos tradicionais de ensino de programação, que dependem fortemente de conceitos teóricos abstratos.

Assim, este estudo é justificado pela necessidade de identificar práticas pedagógicas eficazes que preparem os estudantes para um futuro cada vez mais tecnológico. Através da revisão da literatura, buscamos entender como *Scratch* e Arduino podem ser integrados de forma eficaz no currículo escolar, promovendo não apenas a aquisição de habilidades técnicas, mas também o desenvolvimento de habilidades socioemocionais, como colaboração, comunicação e persistência.

1.2 METODOLOGIA

Este artigo adota uma abordagem bibliográfica, baseada em uma revisão sistemática de artigos, livros e relatórios sobre o uso do *Scratch* e Arduino na Educação. A pesquisa foi realizada em bases de dados acadêmicas, incluindo publicações dos últimos dez anos (2004-2024), para garantir a relevância e atualidade dos dados. Os critérios de inclusão envolveram estudos que abordam a aplicação de Scratch e Arduino na criação de robôs e projetos interativos em contextos educacionais.

2 REVISÃO DA LITERATURA

2.1 VANTAGENS DO USO DO SCRATCH E ARDUINO

O *Scratch* é uma linguagem de programação visual desenvolvida pelo *MIT Media Lab*, que permite aos usuários criar histórias interativas, jogos e animações através da montagem de blocos. Esta abordagem é amplamente utilizada na aprendizagem de conceitos de programação devido à sua interface intuitiva e facilidade de uso, o que elimina a complexidade associada à sintaxe textual (Resnick *et al.*, 2009). Ao abstrair a sintaxe o Scratch permite que os alunos se concentrem na lógica e no fluxo dos programas, tornando a programação mais acessível, especialmente para iniciantes e crianças (Maloney et al., 2010).

O Arduino, por outro lado, é uma plataforma de prototipagem eletrônica de código aberto que simplifica e facilita a criação de projetos eletrônicos. Com sua combinação de *hardware* e *software* fácil de usar, o Arduino capacita os usuários a construir dispositivos interativos que podem, por exemplo, sentir e controlar o ambiente físico (Banzi & Shiloh, 2014). A natureza aberta da plataforma também fomenta a inovação e a colaboração, com uma vasta comunidade de desenvolvedores e educadores compartilhando projetos e recursos online (Kushner, 2011).

O uso combinado dessas ferramentas oferece uma abordagem poderosa para o ensino da lógica de programação e da eletrônica. O *Scratch for Arduino* (S4A) é uma extensão que permite programar placas Arduino usando a interface gráfica do *Scratch*, facilitando a integração entre programação e eletrônica em um ambiente de aprendizagem unificado (Mellis et al., 2012). Esta integração promove uma aprendizagem prática e interativa, na qual os alunos podem visualizar o impacto de seu código em tempo real através de dispositivos físicos, como LEDs, motores e sensores.

Estudos mostram que essa abordagem aumenta a motivação e o engajamento dos alunos. Por exemplo, Martins et al. (2016) destacam que projetos de robótica baseados em *Scratch* e Arduino não apenas tornam a aprendizagem de programação mais divertida, mas também ajudam os alunos a desenvolver habilidades de resolução de problemas e pensamento crítico. Os alunos são desafiados a aplicar a teoria em contextos práticos, o que pode levar a uma compreensão mais profunda dos conceitos e maior retenção de conhecimento (Papert, 1980).

Além disso, a aprendizagem baseada em projetos, facilitada pelo uso de *Scratch* e Arduino, promove um ambiente colaborativo onde os alunos trabalham em equipe para resolver problemas e criar projetos. Este tipo de aprendizado colaborativo é benéfico para o desenvolvimento de habilidades sociais e de comunicação, essenciais para o sucesso no ambiente de trabalho moderno (Barron & Darling-Hammond, 2008). A capacidade de trabalhar eficazmente em grupo e de comunicar ideias de forma clara e concisa é frequentemente mencionada como uma das principais habilidades exigidas pelos empregadores (Trilling & Fadel, 2009).

Outro benefício significativo é a capacidade de adaptar os projetos às necessidades e interesses individuais dos alunos. A personalização do aprendizado, permitida pelo uso de ferramentas flexíveis como *Scratch* e Arduino, pode aumentar a motivação intrínseca dos alunos e encorajá-los a explorar e experimentar de forma independente (Deci & Ryan, 2000). Esta autonomia no aprendizado é crucial para desenvolver uma mentalidade de crescimento e uma abordagem proativa à resolução de problemas (Dweck, 2006).

Por fim, a introdução precoce à programação e à robótica pode influenciar positivamente as escolhas de carreira dos alunos, incentivando mais jovens a seguir carreiras em STEM (Ciência, Tecnologia, Engenharia e Matemática). Programas educacionais que utilizam Scratch e Arduino têm mostrado ser eficazes em despertar o interesse dos alunos por essas áreas, especialmente entre grupos sub-representados, como meninas e minorias (Margolis & Fisher, 2002). Isso é crucial para diversificar o campo da tecnologia e garantir que um espectro mais amplo de perspectivas e experiências esteja representado na força de trabalho futura (Cheryan et al., 2017).

2.2 DESAFIOS NA IMPLEMENTAÇÃO

Apesar dos benefícios, do *Scratch* e Arduino no ensino de programação e robótica, a implementação dessas ferramentas na sala de aula enfrenta vários desafios. Um dos principais obstáculos é a necessidade de formação adequada para os professores e a disponibilidade de recursos materiais. Muitos educadores não possuem experiência prévia com essas ferramentas, o que pode limitar sua eficácia na transmissão dos conceitos aos alunos (Smith, 2018). A formação contínua e especializada dos professores é essencial para garantir que eles se sintam confiantes e capacitados para integrar essas tecnologias em suas práticas pedagógicas (Koehler & Mishra, 2009).

A complexidade da tecnologia pode ser intimidante para educadores que não têm um *background* técnico. A falta de conhecimento técnico pode levar à resistência na adoção de novas ferramentas tecnológicas (Ertmer, 1999). Para superar essa barreira, é fundamental que as escolas invistam em programas de desenvolvimento profissional que ofereçam treinamento prático e suporte contínuo aos professores. Esses programas devem incluir não apenas o uso das ferramentas, mas também metodologias pedagógicas que promovam a aprendizagem ativa e colaborativa (Darling-Hammond et al., 2017).

Outro desafio significativo é a disponibilidade de recursos materiais. A implementação de projetos com *Scratch* e Arduino requer um investimento inicial em *hardware*, como placas Arduino, sensores, atuadores, e outros componentes eletrônicos, além de computadores com *software* adequado instalado. Em muitas escolas, especialmente em regiões com menos recursos, essa barreira financeira pode ser um impedimento substancial (Becker et al., 2016). Além dos custos iniciais, há a necessidade de manutenção e reposição de componentes, o que pode gerar despesas contínuas para as instituições de ensino (Balanskat & Engelhardt, 2015).

A infraestrutura tecnológica das escolas também pode ser um fator limitante. Desse modo, os projetos com *Scratch* e Arduino sejam bem-sucedidos, é necessário que as escolas disponham de laboratórios de informática adequados, com computadores suficientes e conectividade à internet de qualidade. Em muitas escolas, especialmente nas áreas rurais e subfinanciadas, essa infraestrutura básica ainda não está disponível (OECD, 2015).

A cultura escolar e a resistência à mudança também podem representar barreiras. A integração de novas tecnologias muitas vezes requer uma mudança significativa nas práticas pedagógicas e na organização do currículo. Professores e administradores podem relutar na adoção de novas metodologias que sejam diferentes das abordagens tradicionais com as quais estão familiarizados (Fullan, 2007). Para mitigar essa resistência, é crucial envolver todas as partes interessadas no processo de mudança, proporcionando uma visão clara dos benefícios e fornecendo suporte durante a transição (Hall & Hord, 2015).

Além disso, a avaliação da aprendizagem em projetos de robótica e programação apresenta desafios próprios. As metodologias tradicionais de avaliação, focadas em testes e provas, podem não capturar de forma eficaz as competências desenvolvidas através de projetos práticos e interativos. É necessário desenvolver novos métodos de avaliação que considerem habilidades como resolução de problemas, criatividade, colaboração e pensamento crítico (Gibson & Clarke, 2010). A avaliação autêntica, que inclui portfólios, apresentações e autoavaliação, pode ser mais adequada para medir o impacto da aprendizagem baseada em projetos com *Scratch* e Arduino (Wiggins, 2019)).

Por fim, a desigualdade de acesso à tecnologia pode exacerbar as disparidades educacionais. Enquanto alguns estudantes têm acesso a recursos tecnológicos em casa e na escola, outros podem

não ter a mesma oportunidade, o que cria um fosso digital (Warschauer, 2004). Desse modo, as escolas devem adotar políticas que garantam o acesso equitativo às ferramentas e recursos tecnológicos para todos os alunos, independentemente de sua condição socioeconômica (Selwyn, 2011).

3 EXEMPLOS DE PROJETOS INTERATIVOS E ROBÔS

Diversos estudos documentam projetos educacionais que utilizam *Scratch* e Arduino para ensinar conceitos de programação e engenharia de forma prática. Esses projetos oferecem aos alunos a oportunidade de aplicar teorias em práticas concretas, facilitando a compreensão de conceitos complexos através da experimentação e do aprendizado prático, além de promover habilidades essenciais como a resolução de problemas, o pensamento crítico e a criatividade (Blikstein, 2013).

Por exemplo, em uma escola secundária, os alunos desenvolveram robôs que podiam seguir linhas e evitar obstáculos, aplicando conceitos de sensores e controle (Garcia & Gonzalez, 2020). Neste projeto, os alunos utilizaram sensores de infravermelho para detectar linhas e percursos e sensores ultrassônicos para identificar e desviar de obstáculos. Os alunos conectaram esses sensores a placas Arduino programadas com *Scratch for Arduino* (S4A). A utilização de sensores ensina princípios de *feedback* e controle, que são fundamentais em muitas aplicações de engenharia e computação (Bateson, 2001). Portanto, este tipo de projeto permite que os alunos compreendam a lógica de controle de fluxo e a integração de *hardware* e *software*, fundamentais na robótica e em sistemas automáticos (Garcia & Gonzalez, 2020). Além disso, projetos como este ajudam a desenvolver habilidades de resolução de problemas e pensamento lógico, pois os alunos devem ajustar continuamente seus códigos e sensores para otimizar o desempenho do robô (Eguchi, 2014).

Outro projeto analisado envolveu a criação de jogos interativos que respondiam a comandos físicos através de sensores conectados ao Arduino (Johnson & Lee, 2017). Os alunos criaram jogos em que as ações no jogo eram controladas por entradas físicas, como botões e sensores de movimento. Por exemplo, um jogo simples de labirinto foi controlado inclinando-se uma plataforma equipada com sensores de inclinação, permitindo que os jogadores movessem uma bola virtual (Johnson & Lee, 2017). Este tipo de projeto ilustra como a programação pode ser usada para criar interfaces físicas intuitivas, combinando elementos de *design* de jogos com engenharia eletrônica (Kafai & Burke, 2014). Portanto, essa integração entre *software* e *hardware* exemplifica a importância de entender a interface entre o mundo digital e o físico, uma habilidade cada vez mais relevante na era da Internet das Coisas (IoT) (Atzori, Iera, & Morabito, 2010).

Além de robôs e jogos, projetos mais complexos podem incluir sistemas domóticos ou de automação residencial, nos quais os alunos programam placas Arduino para controlar luzes, ventiladores e outros aparelhos domésticos através de sensores de presença e temporizadores (Martin

et al., 2016). Há também projetos que envolvem a criação de sistemas de automação doméstica utilizando Arduino e *Scratch*. Alunos em um projeto de ensino médio desenvolveram um sistema que automatiza o controle de iluminação e temperatura em uma maquete residencial, usando sensores de luz e temperatura para ajustar automaticamente as luzes e o termostato (Smith & Thompson, 2019).

Esses projetos não apenas ensinam programação e eletrônica, mas também introduzem os alunos aos princípios de eficiência energética e automação residencial, áreas de crescente importância no desenvolvimento sustentável (Harb, 2018). Introduz também conceitos de Internet das Coisas (IoT), demonstrando como diferentes dispositivos podem ser conectados e controlados remotamente (Ashton, 2009). A experiência prática com IoT pode preparar os alunos para futuras carreiras em um campo que está rapidamente se expandindo (Greengard, 2015).

Em outro estudo, os alunos criaram veículos autônomos que podiam tomar decisões baseadas em dados sensoriais, como por exemplo, seguir uma trajetória predeterminada ou parar diante de um obstáculo inesperado (Martinez & Stager, 2013). Este tipo de projeto é particularmente útil para ensinar algoritmos de navegação e tomada de decisão, bem como conceitos de inteligência artificial básica. Através da programação desses veículos, os alunos aprendem a importância da lógica condicional e do processamento de sinais em tempo real, essenciais nas áreas da engenharia e ciência da computação (Siegwart et al., 2011).

Outro exemplo de projeto envolve a criação de sistemas de monitoramento ambiental. Os alunos utilizaram sensores de temperatura, umidade e qualidade do ar conectados ao Arduino para coletar dados ambientais, que são então visualizados através de interfaces programadas no *Scratch* (Smith et al., 2018). Este exemplo de projeto mostra-se eficaz no ensino de alunos sobre ciência de dados, coleta e análise de dados, além de promover a conscientização sobre questões ambientais (Starkweather, 2014).

Tais exemplos de projetos ilustram como a combinação de *Scratch* e Arduino pode ser utilizada para criar experiências de aprendizado significativas. A abordagem prática e a interatividade desses projetos facilitam a compreensão de conceitos teóricos, além de promover a criatividade, a colaboração e o engajamento dos alunos. Ademais, ao trabalhar em projetos que têm aplicações práticas e relevância no mundo real, os alunos têm a possibilidade de ver o impacto direto de suas aprendizagens, o que pode aumentar a motivação e o interesse pelas disciplinas de STEM (Ciência, Tecnologia, Engenharia e Matemática) (Honey et al., 2014).

Além dos benefícios educacionais diretos, esses projetos proporcionam oportunidades para a interdisciplinaridade, integrando conhecimentos de várias disciplinas, como Ciências, Matemática e Tecnologia. Ao trabalhar em projetos que combinam programação e eletrônica com aplicações reais,

os alunos são incentivados a aplicar conhecimentos teóricos em contextos práticos, reforçando seu aprendizado e aumentando sua motivação (Hmelo-Silver, 2004).

4 DISCUSSÃO

A análise da literatura revela que a integração de *Scratch* e Arduino pode transformar significativamente o ensino de lógica de programação, tornando-o mais acessível e eficaz para os alunos. Ao oferecer uma abordagem prática e interativa, essas ferramentas possibilitam que os alunos experimentem conceitos abstratos em um contexto tangível, o que pode facilitar a compreensão e a retenção do conhecimento (Papert, 1980).

O *Scratch*, com sua interface gráfica intuitiva, facilita a introdução de conceitos de programação sem a complexidade da sintaxe textual (Resnick *et al*., 2009). Esta abordagem permite que os alunos se concentrem no desenvolvimento do pensamento computacional e na resolução de problemas de forma criativa e colaborativa (Brennan & Resnick, 2012).

A plataforma Arduino complementa essa experiência ao conectar o mundo digital ao físico, permitindo que os alunos vejam os resultados tangíveis de seu código em ação (Banzi & Shiloh, 2014). A combinação dessas ferramentas promove uma aprendizagem experiencial e prática, fundamental para a internalização de conceitos abstratos, como algoritmos e estruturas de controle (Papert, 1980). Estudos indicam que projetos baseados em Arduino não apenas melhoram a compreensão técnica dos alunos, mas também incentivam o desenvolvimento de habilidades práticas e de engenharia (Blikstein, 2013).

A abordagem baseada em projetos, facilitada pelo uso de *Scratch* e Arduino, promove uma aprendizagem mais significativa, na qual os alunos não apenas absorvem informações, mas também aplicam e constroem seu conhecimento em situações do mundo real (Krajcik et al., 2008). Este tipo de aprendizado ativo e construtivista é mais alinhado com as necessidades e características dos alunos da geração atual, que estão acostumados a interagir com tecnologia desde cedo (Prensky, 2001).

No entanto, a implementação eficaz desta abordagem depende fortemente da formação adequada dos professores e da disponibilidade de recursos. Como mencionado por Koehler & Mishra (2009), a falta de experiência prévia dos educadores com essas tecnologias pode ser uma barreira significativa. Pesquisas mostram que muitos professores se sentem inseguros ao utilizar novas ferramentas tecnológicas devido à falta de formação específica e contínua (Ertmer & Ottenbreit-Leftwich, 2010). Para superar esse desafio, é necessário que as políticas educacionais priorizem programas de desenvolvimento profissional que ofereçam treinamento prático e suporte contínuo. Esses programas devem abordar o uso técnico das ferramentas, além de incluir estratégias pedagógicas que integrem a tecnologia de maneira significativa no currículo (Darling-Hammond et al., 2017).

Portanto, programas de desenvolvimento profissional contínuo, que oferecem treinamento especializado e suporte, são fundamentais para capacitar os professores a utilizarem efetivamente essas ferramentas em sala de aula (Darling-Hammond et al., 2017).

Além da formação de professores, a disponibilidade de recursos materiais também é uma questão importante a ser abordada. A implementação de projetos com *Scratch* e Arduino requer investimentos em *hardware* e *software*, bem como uma infraestrutura adequada, como laboratórios de informática equipados e conectividade à internet (Becker et al., 2016). Em muitas escolas, especialmente nas áreas rurais e subfinanciadas, essa infraestrutura ainda é inadequada, o que inviabiliza a implementação de projetos com *Scratch* e Arduino (OECD, 2015). Investimentos públicos e privados são necessários para garantir que todas as escolas tenham acesso aos recursos necessários para implementar essa abordagem de forma eficaz. Assim, é imperativo que políticas educacionais e iniciativas governamentais priorizem o investimento em tecnologia educacional e garantam o acesso equitativo a recursos tecnológicos em todas as escolas (Balanskat & Engelhardt, 2015).

No entanto, é importante ressaltar que o simples fornecimento de tecnologia não é suficiente. Também é necessário desenvolver e implementar estratégias pedagógicas eficazes que integrem essas ferramentas de forma significativa ao currículo escolar (Ertmer, 1999). Os educadores devem ser incentivados a adotar práticas pedagógicas inovadoras que promovam a criatividade, a colaboração e o pensamento crítico dos alunos, em vez de se concentrarem exclusivamente na transmissão de conteúdos (Fullan, 2007).

Finalmente, é essencial que as políticas educacionais incentivem a inclusão e a equidade no acesso à tecnologia. A disparidade no acesso a ferramentas tecnológicas pode exacerbar as desigualdades educacionais existentes (Warschauer, 2004). Assim, as escolas devem adotar estratégias para garantir que todos os alunos, independentemente de sua condição socioeconômica, tenham a oportunidade de aprender e se beneficiar dessas tecnologias (Selwyn, 2011). Isso inclui a disponibilização de dispositivos, o acesso à internet de alta qualidade e o suporte técnico necessário para a implementação eficaz dos projetos.

Outro aspecto importante é a necessidade de adaptação das metodologias de avaliação. A aprendizagem baseada em projetos, como a criação de robôs e sistemas interativos, desenvolve habilidades que vão além do conhecimento teórico, incluindo a criatividade, a colaboração e a resolução de problemas complexos (Wiggins, 2019). Métodos tradicionais de avaliação, como testes e provas, podem não capturar adequadamente essas competências. Portanto, é necessário desenvolver e implementar métodos de avaliação autêntica que reflitam melhor as habilidades práticas e o pensamento crítico dos alunos (Gibson & Clarke, 2010).

Em suma, a integração de *Scratch* e Arduino no ensino de lógica de programação representa uma oportunidade empolgante para transformar a educação e preparar os alunos para os desafios do século XXI. Entretanto, para que essa abordagem atinja seu pleno potencial, é necessário um compromisso coletivo de educadores, gestores, formuladores de políticas públicas e comunidades educacionais para fornecer o suporte e os recursos necessários para sua implementação bem-sucedida.

5 CONCLUSÃO

O uso combinado de *Scratch* e Arduino na educação representa uma abordagem inovadora e promissora para o ensino de lógica de programação, proporcionando aos alunos uma experiência de aprendizado envolvente e significativa. Ao integrar elementos visuais e práticos, essas ferramentas tornam os conceitos abstratos mais tangíveis e acessíveis, promovendo uma compreensão mais profunda e duradoura por parte dos alunos (Resnick *et al*., 2009; Banzi & Shiloh, 2014).

Como discutido ao longo deste artigo, o uso de *Scratch* e Arduino oferece uma série de vantagens, incluindo uma interface intuitiva, a oportunidade de aprendizado prático em tempo real e a promoção de habilidades como resolução de problemas, pensamento crítico e colaboração (Resnick et al., 2009; Banzi & Shiloh, 2014). Estudos demonstram que essa abordagem pode aumentar a motivação dos alunos e melhorar o engajamento nas disciplinas de STEM (Martins et al., 2016).

Embora existam desafios na implementação, como a formação de professores e a disponibilidade de recursos, as vantagens oferecidas por essa abordagem superam amplamente as dificuldades. Os projetos baseados em Scratch e Arduino proporcionam um ambiente de aprendizado dinâmico e interativo, onde os alunos são incentivados a explorar, experimentar e colaborar, desenvolvendo habilidades essenciais para o século XXI, como resolução de problemas, pensamento crítico e criatividade (Blikstein, 2013; Krajcik et al., 2008). Assim, é fundamental que educadores, gestores, formuladores de políticas e comunidades educacionais trabalhem juntos para superar esses obstáculos e aproveitar todo o potencial dessas ferramentas inovadoras.

Para isso, futuras pesquisas devem se concentrar no desenvolvimento de estratégias eficazes para aprimorar a formação de educadores nessa área e expandir o acesso a essas ferramentas em diferentes contextos educacionais. Além disso, é importante destinar recursos para investimentos em programas de desenvolvimento profissional contínuo e políticas educacionais que priorizem a integração de tecnologia no currículo podem ajudar a superar os desafios existentes e garantir que todos os alunos tenham acesso a oportunidades de aprendizado igualitárias e de alta qualidade (Darling-Hammond et al., 2017; OECD, 2015).

Em última análise, o uso de *Scratch* e Arduino na educação não apenas prepara os alunos para os desafios do século XXI, mas também os capacita a se tornarem criadores e inovadores em um mundo cada vez mais tecnológico e interconectado.

REFERÊNCIAS

ASHTON, Kevin. That Internet of Things thing. RFID Journal, v. 22, n. 7, p. 97-114, 2009. Disponível em http://www.rfidjournal.com/article/view/4986. Acesso em 15 fev. 2024.

ATZORI, Luigi; IERA, Antonio; MORABITO, Giacomo. The internet of things: A survey. Computer Networks, v. 54, n. 15, p. 2787-2805, out. 2010. Disponível em https://www.researchgate.net/publication/222571757_The_Internet_of_Things_A_Survey. Acesso em 15 fev. 2024.

BALANSKAT, Anja; ENGELHARDT, Katja. Computing our future: Computer programming and coding - Priorities, school curricula and initiatives across Europe. European Schoolnet, 2015. Disponível em http://www.eun.org/documents/411753/817341/Computing+our+future_final_2015.pdf/d3780a64-1081-4488-8549-6033200e3c03. Acesso em 15 fev. 2024.

BANZI, Massimo; SHILOH, Michael. Make: getting started with Arduino. 3. ed. USA: Maker Media, Inc., 2014. Disponível em https://www.esc19.net/cms/lib/TX01933775/Centricity/Domain/110/make_gettingstartedwitharduino_3rdedition.pdf. Acesso em 15 fev. 2024

BARRON, Brigid; DARLING-HAMMOND, Linda. Teaching for meaningful learning: A review of research on inquiry-based and cooperative learning. John Wiley & Sons, 2008. Disponível em https://files.eric.ed.gov/fulltext/ED539399.pdf. Acesso em 15 fev. 2024.

BATESON, Robert N. Introduction to Control System Technology. Prentice Hall, 2001.

BECKER, Adams; FREEMAN, A.; GIESINGER HALL, C.; CUMMINS, M.; YUHNKE, B. The NMC/CoSN Horizon Report: 2016 K-12 Edition. Austin, Texas: The New Media Consortium, 2016. Disponível em https://files.eric.ed.gov/fulltext/ED570463.pdf. Acesso em 15 fev. 2024.

BERS, Marina U. The tangibleK Robotics Program: Applied Computational Thinking for Young Children. Early Childhood Research & Practice, v. 12, n. 2, p. 1-20, 2010. Disponível em https://files.eric.ed.gov/fulltext/EJ910910.pdf. Acesso em 15 fev. 2024.

BLIKSTEIN, Paulo. Digital fabrication and 'making' in education: The democratization of invention. Community and Environment, p. 203-221, 2013. Disponível em https://www.researchgate.net/publication/281495128_Digital_Fabrication_and_'Making'_in_Education_The_The_Democratization_of_Invention. Acesso em 15 fev. 2024.

BRENNAN, Karen; RESNICK, Mitchel. New frameworks for studying and assessing the development of computational thinking. Annual American Educational Research Association, Vancouver, BC, Canada, 2012. Disponível em https://people.cs.vt.edu/~kafura/CS6604/Papers/Framework-Assessing-CT.pdf. Acesso em 15 fev. 2024.

CHERYAN, Sapna; ZIEGLER, Sianna A.; MONTOYA, Amanda K.; JIANG, Lily. Why Are Some STEM Fields More Gender Balanced Than Others? Psychological Bulletin, v. 143, n. 1, p. 1-35, 2017. Disponível em https://research.chicagobooth.edu/-/media/research/cdr/docs/cheryan-paper-1. Acesso em 15 fev. 2024.

DARLING-HAMMOND, Linda; HYLER, Maria E.; GARDNER, Madelyn. Effective Teacher Professional Development. Palo Alto, CA: Learning Policy Institute, 2017. Disponível em https://learningpolicyinstitute.org/sites/default/files/product-files/Effective_Teacher_Professional_Development_REPORT.pdf. Acesso em 15 fev. 2024.

DECI, Edward L.; RYAN, Richard M. The "What" and "Why" of Goal Pursuits: Human Needs and the Self-Determination of Behavior. Psychological Inquiry, v. 11, n. 4, p. 227-268, 2000. Disponível em https://web.archive.org/web/20170810210258id_/http://academic.udayton.edu/jackbauer/Readings%20595/Deci%2000%20goals%20SDT.pdf. Acesso em 15 fev. 2024.

DWECK, Carol S. Mindset: The New Psychology of Success. New York: Random House, 2006. Disponível em https://adrvantage.com/wp-content/uploads/2023/02/Mindset-The-New-Psychology-of-Success-Dweck.pdf. Acesso em 15 fev. 2024.

EGUCHI, Amy. Educational robotics for promoting 21st century skills. Journal of Automation, Mobile Robotics & Intelligent Systems, v. 8, n. 1, p. 5-11, jan. 2014. Disponível em https://www.researchgate.net/publication/274882640_Educational_Robotics_for_Promoting_21st_Century_Skills. Acesso em 15 fev. 2024.

ERTMER, Peggy; OTTENBREIT-LEFTWICH, Anee. Teacher technology change: How knowledge, confidence, beliefs, and culture intersect. Journal of Research on Technology in Education, v. 42, n. 3, p. 255-284, mar. 2010. Disponível em https://www.researchgate.net/publication/272007146_Teacher_Technology_Change_How_Knowledge_Beliefs_and_Culture_Intersect. Acesso em 15 fev. 2024.

ERTMER, Peggy. A. (1999). Addressing first-and second-order barriers to change: Strategies for technology integration. Educational Technology Research and Development, v. 47, n. 4, p. 47-61, dez. 1999. Disponível em https://www.researchgate.net/publication/225685117_Addressing_first-_and_secondorder_barriers_to_change_Strategies_for_technology_integrationEducational_Technology_Research_and_Development_474_47-61. Acesso em 15 fev. 2024.

FULLAN, Michael. The new meaning of educational change. 4. ed. New York: Teachers College Press, 2007. Disponível em https://www.daneshnamehicsa.ir/userfiles/files/1/6-%20The%20New%20Meaning%20of%20Educational%20Change,%20Fourth%20Edition.pdf. Acesso em 15 fev. 2024.

GREENGARD, Samuel. The Internet of Things. MIT Press, 2015. Disponível em https://direct.mit.edu/books/book/4051/The-Internet-of-Things. Acesso em 15 fev. 2024.

HALL, Gene E.; HORD, Shirley M. Implementing change: Patterns, principles, and potholes. University of Nevada: Pearson, 2015. Disponível em https://daneshnamehicsa.ir/userfiles/file/Manabeh/Implementing%20Change_%20Patterns,%20-%20Gene%20E.%20Hall%20(2).pdf. Acesso em 15 fev. 2024.

HMELO-SILVER, Cindy E. Problem-based learning: What and how do students learn? Educational Psychology Review, v. 16, n. 3, p. 235-266, set. 2004. Disponível em https://docdrop.org/static/drop-pdf/Hmelo-Silver2004-ZZaX8.pdf. Acesso em 15 fev. 2024.

HONEY, Margaret; PEARSON, Greg; SCHWEINGRUBER, Heidi. STEM integration in K-12 education: Status, prospects, and an agenda for research. Washington, DC: National Academies Press, 2014. Disponível em https://www.middleweb.com/wp-content/uploads/2015/01/STEM-Integration-in-K12-Education.pdf. Acesso em 15 fev. 2024.

KAFAI, Yasmin B.; BURKE, Quinn. Connected Code: Why Children Need to Learn Programming. MIT Press, 2014. Disponível em https://direct.mit.edu/books/monograph/4015/Connected-CodeWhy-Children-Need-to-Learn. Acesso em 15 fev. 2024.

KOEHLER, Matthew J.; MISHRA, Punya. What is Technological Pedagogical Content Knowledge? Contemporary Issues in Technology and Teacher Education, v. 9, n. 1, p. 60-70, 2009. Disponível em https://citejournal.org/wp-content/uploads/2016/04/v9i1general1.pdf. Acesso em 15 fev. 2024.

KRAJCIK, Joseph; MARX, Ron; BLUMENFELD, Phyllis; SOLOWAY, Elliot; FISHMAN, Barry. Inquiry based science supported by technology: achievement among urban middle school students, 2008. Disponível em https://files.eric.ed.gov/fulltext/ED443676.pdf. Acesso em 15 fev. 2024.

KUSHNER, David. The making of Arduino. IEEE Spectrum, 26 out. 2011, p. 26-29. Disponível em https://web.eecs.umich.edu/~prabal/teaching/resources/eecs582/kushner11arduino.pdf. Acesso em 15 fev. 2024.

MALONEY, John; RESNICK, Mitchel; RUSK, Natalie; SILVERMAN, Brian; EASTMOND, Evelyn. The Scratch programming language and environment. ACM Transactions on Computing Education, v. 10, n. 4, p. 1-15, nov. 2010. Disponível em https://dl.acm.org/doi/10.1145/1868358.1868363. Acesso em 15 fev. 2024.

MARGOLIS, Jane; FISHER, Allan. Unlocking the Clubhouse: Women in Computing. Cambridge: MIT press, 2002. Disponível em https://we.riseup.net/assets/459427/Margolis+Jane+Fisher+Allan+Unlocking+the+Clubhouse+Women+in+Computing.pdf. Acesso em 15 fev. 2024.

MARTINEZ, Sylvia Libow; STAGER, Gary S. Invent to learn: Making, tinkering, and engineering in the classroom. Constructing Modern Knowledge Press, 2013.

MATARIC, Maja J.; KOENIG, Nathan; FEIL-SIEFER, David. Materials for enabling hands-on robotics and STEM education. AAAI Spring Symposium on Robots and Robot Venues: Resources for AI Education, Stanford, CA, mar. 2007. Disponível em https://www.researchgate.net/publication/221250968_Materials_for_Enabling_Hands-On_Robotics_and_STEM_Education. Acesso em 15 fev. 2024.

MELLIS, David A.; BANZI, Massimo; CUARTIELLES, David; IGOE, Tom. Arduino: An open electronics prototyping platform. CHI'12 Extended Abstracts on Human Factors in Computing Systems, p. 1-11, 2012. Disponível em https://pt.scribd.com/document/116025464/Submission-Mellis-0. Acesso em 15 fev. 2024.

OECD. Organização para a Cooperação e Desenvolvimento Econômico. Students, Computers and Learning: Making the Connection. PISA: OECD Publishing, 2015. Disponível em https://www.oecd-ilibrary.org/docserver/9789264239555-en.pdf?expires=1717800937&id=id&accname=guest&checksum=5007A8A9AF4B6C3B184C2C115E20FBEC. Acesso em 15 fev. 2024.

PAPERT, Seymour. Mindstorms: Children, computers, and powerful ideas. New York: Basic Books, 1980. Disponível em https://worrydream.com/refs/Papert_1980_-_Mindstorms,_1st_ed.pdf. Acesso em 15 fev. 2024.

PRENSKY, Marc. Digital natives, digital immigrants. On the Horizon, v. 9, n. 5, p. 1-6, out. 2001. Disponível em https://www.marcprensky.com/writing/Prensky%20-%20Digital%20Natives,%20Digital%20Immigrants%20-%20Part1.pdf. Acesso em 15 fev. 2024.

RESNICK, Mitchel *et al*. Scratch: Programming for All. Communications of the ACM, v. 52, n. 11, p. 60-67, nov. 2009. Disponível em https://dl.acm.org/doi/pdf/10.1145/1592761.1592779. Acesso em 15 fev. 2024.

SELWYN, Neil. Education and Technology: Key Issues and Debates. London: Continuum. 2011.

SIEGWART, Roland; NOURBAKHSH, Illah Reza; SACARAMUZZA, Davide. Introduction to autonomous mobile robots. MIT Press, 2011.

TRILLING, Bernie; FADEL, Charles. 21st century skills: Learning for life in our times. John Wiley & Sons, 2009. Disponível em http://ardian.id/wp-content/uploads/2018/10/21st_Century_Skills_Learning_for_Life_in_Our_Times____2009-3.pdf. Acesso em 15 fev. 2024.

WARSCHAUER, Mark. Technology and Social Inclusion: Rethinking the Digital Divide. Cambridge: MIT Press, 2004. Disponível em https://www.researchgate.net/publication/329649885_Technology_and_Social_Inclusion_Rethinking_the_Digital_Divide. Acesso em 15 fev. 2024.

WIGGINS, Grant. The case for authentic assessment. Practical Assessment, Research, and Evaluation, v. 2, n. 1, nov. 2019. Disponível em https://scholarworks.umass.edu/cgi/viewcontent.cgi?article=1024&context=pare. Acesso e 15 fev. 2024.

WING, Jeannette M. Computational thinking. Communications of the ACM, v. 49, n. 3, p. 33-35, mar. 2006. Disponível em https://dl.acm.org/doi/pdf/10.1145/1118178.1118215. Acesso em 15 fev. 2024.