ISSN: 2358-8829

DESENVOLVIMENTO DE JOGO DIDÁTICO NO ROBLOX MEDIADO POR IA GENERATIVA: UM RELATO DE EXPERIÊNCIA DE CRIAÇÃO COM O USO DE LINGUAGEM NATURAL

Alírio Gaia de Lima[1]
Roberto Felicio de Oliveira[2]
Anderson Cavalcante Gonçalves[3]

INTRODUÇÃO

A criação de jogos didáticos digitais permanece distante do cotidiano dos professores da educação básica. Embora o potencial pedagógico dos jogos seja reconhecido, o professor que deseja produzir seu próprio material lúdico encontra a barreira da programação, que dificulta a exploração dos jogos digitais como ferramenta de ensino. Pérez et al. (2023, p. 1) observam que o uso de jogos sérios na educação ainda é limitado porque seu desenvolvimento é um processo caro e complexo, que requer a participação de diferentes especialistas, como educadores, designers e programadores. Portanto essa complexidade coloca os jogos digitais em posição periférica na prática docente.

Uma forma de contornar esse problema seria a utilização de plataformas de jogos do tipo sandbox, em que os próprios usuários constroem as experiências e as disponibilizam para a comunidade de jogadores. Dentre elas, o Roblox tem se destacado como um ambiente amplamente utilizado entre crianças e adolescentes. A própria empresa declara a intenção de permitir a construção de objetos e cenas interativas sem modelagem ou codificação complexa (VINCENT; AYYAR, 2023, p. 2). Na prática, contudo, a criação continua sendo dependente da escrita de scripts na linguagem *Luau*, nativa dessa plataforma, configurando ainda uma barreira para não programadores. Tal barreira foi documentada por Murilo (2024), que, ao elaborar um tutorial destinado a professores leigos em programação, concluiu que tal abordagem não se mostrava viável, pois a criação de experiências na plataforma demanda conhecimentos básicos da lógica de programação e de suas estruturas sintáticas.

[1] Mestrando do Programa de Pós-graduação em Gestão, Educação e Tecnologias da Universidade Estadual de Goiás - UEG, aliriogaia@gmail.com.
[2] Professor coorientador: Doutor em Informática pela Pontifícia Universidade Católica do Rio de Janeiro - PUC-Rio. Professor do Programa de Pós Graduação em Gestão, Educação e Tecnologias da Universidade Estadual de Goiás - UEG, roberto.oliveira@ueg.br.
[3] Professor orientador: Doutor em Ciência da Computação pela Universidade Federal de Goiás - UFG. Professor do Programa de Pós Graduação em Gestão, Educação e Tecnologias da Universidade Estadual de Goiás - UEG, anderson.goncalves@ueg.br.

ISSN: 2358-8829

Com os avanços recentes da inteligência artificial generativa (IAG) uma rota alternativa para criação de jogos digitais tem se apresentado. Os modelos de IAG atualmente no mercado apresentam-se úteis para gerar códigos de diversas linguagens de programação, a partir de comandos em linguagem natural, isto é, na linguagem humana comum, possibilitando ao usuário a possibilidade de descrever o que deseja e obter o artefato correspondente. Diante disso, buscou-se compreender como a IAG, mediada por comandos em linguagem natural, pode viabilizar a criação de jogos didáticos no Roblox por usuários sem conhecimentos de programação. O objetivo deste trabalho é descrever a criação de um jogo didático no Roblox por meio de comandos em linguagem natural, mediada por software de IAG integrada ao *Roblox Studio*, o ambiente de criação de jogos da plataforma, sem a necessidade de programação manual por parte do autor do recurso educacional.

Quanto ao percurso metodológico, adotou-se o relato de experiência, de abordagem qualitativa e caráter exploratório. Utilizou-se o assistente de IAG Claude, da empresa Anthropic, conectado ao *Roblox Studio* através do *Model Context Protocol* (MCP), a ferramenta que possibilita a IAG a interagir com outros softwares, em sessões sucessivas de interação conversacional conduzidas em língua portuguesa, nas quais o pesquisador descreve o que se pretende obter sem especificar como implementá-lo. Construiu-se um jogo do tipo escape room sobre máquinas simples, com foco em alavancas, um conteúdo de ciências da natureza, destinado a alunos do sétimo ano do ensino fundamental.

Os resultados evidenciam que a criação de cenários, mecânicas, animações, questionários e scripts podem ser integralmente implementados a partir de descrições em linguagem natural, sem digitação de código pelo desenvolvedor, uma vez que o assistente gera o código, insere-o no projeto, executa-o e verifica seu comportamento no interior do próprio ambiente de criação. No entanto, o processo exigiu refinamento iterativo, em questões de proporção, posicionamento, animação, dentre outros, além de cuidados com modelos de terceiros, cujos scripts embutidos chegaram a comprometer o funcionamento do jogo. Conclui-se que, na experiência relatada, a mediação por IAG permitiu contornar a barreira técnica da programação, direcionando os esforços para a formulação clara de comandos em linguagem natural e o refinamento iterativo dos resultados obtidos.

Restando investigar, portanto, a efetividade da proposta junto a docentes sem formação na área da programação, o que implicaria avaliar se professores da educação básica conseguem, de modo autônomo, conduzir o processo de criação aqui descrito e atingir resultados funcionais comparáveis. Igualmente, permanece em aberto a avaliação da experiência do jogador, abrangendo engajamento, motivação e ganhos efetivos na

ISSN: 2358-8829

aprendizagem dos conceitos trabalhados, em distintos contextos educacionais e com diferentes perfis de estudantes, faixas etárias e níveis de ensino.

REFERENCIAL TEÓRICO

Jogos digitais e aprendizagem

A abordagem didática apoiada por jogos digitais parte do pressuposto de que o aluno aprende melhor de forma interativa. Prensky (2001) argumenta que os estudantes formados em contato intenso com tecnologias digitais desenvolvem modos próprios de processar informação e demandam abordagens que dialoguem com essas formas de interação. Gee (2009) realizou uma importante contribuição para essa discussão ao identificar nos bons videogames princípios que os aproximam de boas práticas pedagógicas, como a aprendizagem situada, o desafio progressivo, o retorno imediato e a possibilidade de errar sem custo elevado para a progressão do aprendiz e para a manutenção de sua motivação.

Portanto, o mais interessante dos jogos digitais não é a presença de elementos lúdicos, ou o alto engajamento que eles podem gerar, mas a articulação entre a mecânica do jogo e o conteúdo a ser aprendido. Quando essa articulação é bem explorada a ação exigida pelo jogo é a própria avaliação da aprendizagem. Nesse sentido, Ho e Lee (2023, p. 1053) destacam o Roblox como plataforma que possibilita a customização e permite a criação de experiências ajustadas a objetivos de aprendizagem específicos. No entanto, essa abertura demanda do professor competências específicas, para que este possa assumir uma posição ativa na criação e na revisão de seus próprios jogos digitais, como recursos educacionais.

A barreira técnica da programação

A maior dificuldade em se produzir jogos educacionais não se dá pela falta de interesse dos professores, mas é o próprio processo de desenvolvimento que trava o avanço na criação e utilização de jogos digitais na educação. Pérez-Colado et al. (2023, p. 1) observam que jogos sérios exigem trabalho multidisciplinar, envolvendo designers, programadores e educadores. No caso do Roblox, Murilo (2024) desenvolveu objetos virtuais de aprendizagem e elaborou um tutorial voltado a professores sem formação em programação, concluindo pela inviabilidade da abordagem, uma vez que a criação de experiências demanda conhecimentos de programação, sendo indispensável o uso de scripts. O achado é relevante por não se tratar de suposição sobre a dificuldade dos professores, mas de constatação empírica resultante da tentativa concreta de ensiná-los a programar no Roblox, a partir de sua linguagem nativa.

Uma rota alternativa, que tem se apresentado como uma tendência crescente, é da geração de código auxiliada por assistentes de IAG. No entanto, Feldman e Anderson (2024,

ISSN: 2358-8829

p. 4), em estudo controlado com sessenta e sete participantes sem experiência prévia em programação, identificaram obstáculos que não se resolvem com o acesso à ferramenta. No estudo identificou-se que os leigos não empregam os termos que um programador utilizaria, e os modelos foram treinados a partir de código de programação profissional, e os participantes mostraram-se incapazes de compreender o código gerado e as mensagens de erro, o que os impede de julgar se o resultado atende ao pretendido e de corrigir eventuais falhas de forma autônoma (FELDMAN; ANDERSON, 2024, p. 13).

Outro estudo que também avaliou a utilização de IAG, conduzido por Kohen-Vacs, Usher e Jansen (2025, p. 3177), concluiu que embora as percepções dos participantes sobre essas ferramentas fossem favoráveis, o desempenho na tarefa de corrigir código gerado por modelos de linguagem foi significativamente inferior ao obtido em tarefas tradicionais de programação, invertendo a expectativa de ganho de produtividade.

Portanto, se a criação de jogos no Roblox exige scripts, tutoriais para educadores leigos em programação não são suficientes (MURILO, 2024) e a falta de domínio do vocabulário necessário para solicitá-los às IAGs é uma barreira (FELDMAN; ANDERSON, 2024, p. 13), tampouco uma mediação em que o professor apenas solicita o código e em seguida o transfere manualmente para o ambiente de desenvolvimento é uma solução definitiva, uma vez que erros de scripts precisam ser corrigidos durante o desenvolvimento.

Inteligência artificial generativa na criação de jogos

No caso do Roblox, Vincent e Ayyar (2023, p. 1) apresentaram a proposta de se utilizar linguagem natural para construir objetos e cenas interativas sem modelagem ou codificação complexa, com o objetivo de democratizar a criação no Roblox. Relatam ainda que a eficácia da inteligência artificial nativa da plataforma cresce conforme aumenta o contexto disponível, incluindo o código já existente no projeto, seus metadados e o histórico de interações acumulado (VINCENT; AYYAR, 2023, p. 2).

No campo educacional, Ho e Lee (2023, p. 1052) utilizaram o *ChatGPT*, assistente de IAG da *OpenAI*, no desenvolvimento de um jogo no Roblox para o ensino de máquinas elétricas, para um curso universitário, e relatam que o modelo auxiliou na otimização de scripts, na estruturação das atividades e na correção de falhas apontadas pelos alunos (HO; LEE, 2023, p. 1057). O desenvolvimento, contudo, foi conduzido por pesquisadores da engenharia elétrica, com domínio técnico prévio em programação, e recorreu a ferramentas externas de modelagem tridimensional. Neste caso, a IAG atuou como assistente de um desenvolvedor competente, e não como mediador autônomo do processo de desenvolvimento.

ISSN: 2358-8829

Jeong et al. (2024, p. 17) integraram IAG ao Roblox para desenvolver ambientes voltados à formação de professores, empregando um modelo de linguagem para gerar falas de personagens não jogáveis. O relato mostra que mesmo dispondo de um modelo de IA generativa, o pesquisador, professor da educação básica, precisou desenvolver habilidades com o design do Roblox, recorrendo a tutoriais em vídeo, comunidades e leitura de artigos. A inteligência artificial gerava o conteúdo textual, ao passo que a construção do ambiente tridimensional permaneceu como tarefa essencialmente humana.

A viabilidade da criação conduzida a partir de linguagem natural foi demonstrada por Yang (2025, p. 2), que desenvolveu um jogo de simulação sobre legislação médica na plataforma *Rosebud AI*, comparando o ensino gamificado ao tradicional com duzentos e um estudantes (YANG, 2025, p. 3). O pesquisador afirma que professores sem habilidades de programação podem empregar IAG a partir de linguagem natural para produzir software educacional, mas reconhece que não seria viável a criação de jogos de alta complexidade (YANG, 2025, p. 1). O estudo foi conduzido em uma plataforma de jogos bidimensionais baseados em diálogo, o que delimita o alcance de sua conclusão. Moon et al. (2024, p. 452), em revisão sobre o tema, concluem que a participação ativa dos instrutores no design do jogo é essencial e que sua formação deve abranger aspectos técnicos e pedagógicos.

O estado da arte sobre a temática delimita a lacuna em que este relato de experiência se propõe a inserir-se, partindo de uma agenda de pesquisa que sugere a integração da IAG no interior do ambiente de desenvolvimento, em interação conversacional sucessiva, a partir de comandos em linguagem natural, em que o modelo de IAG encarrega-se de escrever o código, executá-lo, verificar seu funcionamento e corrigi-lo, modalidade que Feldman e Anderson (2024, p. 17) indicaram como prioridade de pesquisas futuras.

METODOLOGIA

Caracterização da pesquisa

Este trabalho configura-se como relato de experiência, de abordagem qualitativa e caráter exploratório, adequado à investigação de um processo de criação ainda pouco documentado, cuja compreensão decorre da análise sistemática da experiência vivenciada pelo pesquisador. Relatos de experiência, para além de uma descrição narrativa, podem constituir uma modalidade de produção de conhecimento quando apresentam sistematização do percurso, dos registros e da análise da experiência (ANTUNES et al., 2024). Neste estudo não se pretende avaliar o desempenho dos estudantes diante da ferramenta, mas analisar o

ISSN: 2358-8829

processo de criação de um recurso educacional mediado por IAG, considerando as possibilidades, limitações e intervenções necessárias ao longo do desenvolvimento.

Ambiente técnico

A experiência foi conduzida no *Roblox Studio*, software oficial de desenvolvimento da plataforma, na versão 0.727.0.7271204. Como assistente de IAG, utilizou-se o *Claude*, da *Anthropic*, em sua aplicação para desktop, com o modelo *Opus 4.8*. A integração entre o assistente e o ambiente de criação foi estabelecida por meio do *Model Context Protocol* (MCP), padrão aberto de comunicação entre modelos de linguagem e aplicações externas. Optou-se pelo servidor integrado ao próprio *Roblox Studio*, disponibilizado pela plataforma e ativado em suas configurações, reduzindo a dependência de componentes externos e ampliando a reprodutibilidade do procedimento por outros pesquisadores.

Diferentemente de abordagens em que o modelo de linguagem opera externamente, gerando código posteriormente transposto pelo usuário, nesta experiência o agente de IAG atuou integrado ao ambiente de desenvolvimento, com acesso à estrutura do projeto e capacidade de criar e modificar objetos, produzir e executar scripts, iniciar testes e verificar os resultados de suas ações. Essa configuração aproxima-se do paradigma de desenvolvimento agêntico, no qual modelos de linguagem podem planejar, executar tarefas, utilizar ferramentas e adaptar suas ações a partir de resultados intermediários, ultrapassando a lógica de geração isolada de código para uma dinâmica de ação, verificação e adaptação contínuas (WANG et al., 2025), paradigma ainda pouco explorado em contextos educacionais.

Procedimento

A construção ocorreu em sessões sucessivas de interação conversacional, conduzidas em língua portuguesa. Em cada sessão, o autor descreve em linguagem natural o resultado pedagógico ou a experiência de jogo pretendida, sem necessariamente especificar sua implementação técnica. Solicitações como a construção de salas conectadas por portas condicionadas à compreensão de determinado conceito ou a criação de uma alavanca capaz de movimentar um objeto eram apresentadas ao assistente, que realizava sua tradução para estruturas tridimensionais e às instruções de código correspondentes.

O processo foi organizado em ciclos de solicitação, implementação, verificação e refinamento. Após cada solicitação, o resultado era examinado por inspeção visual no editor, execução em modo de teste ou verificação funcional no próprio ambiente. Quando identificada divergência entre o resultado esperado e o obtido, o pesquisador descrevia o problema em linguagem natural e uma nova interação era realizada. Os registros das

ISSN: 2358-8829

interações, alterações, erros e correções constituíram o material de análise da experiência, considerando-se a correspondência entre a intenção pedagógica e sua implementação, a necessidade de intervenção humana, os erros produzidos e as limitações e potencialidades observadas no uso do agente de IAG. Essa dinâmica iterativa é coerente com estudos recentes sobre colaboração entre desenvolvedores e agentes de IA, nos quais a interação contínua, o refinamento dos resultados e a verificação durante o processo apresentam papel relevante diante das limitações intrínsecas dos agentes em tarefas complexas e abertas de desenvolvimento de software (KUMAR et al., 2025).

O artefato

O produto desenvolvido consiste em um jogo do tipo escape room, gênero no qual o jogador resolve desafios sucessivos para avançar por diferentes ambientes. A escolha buscou estabelecer correspondência entre a mecânica do jogo e o conteúdo a ser aprendido, de modo que a progressão dependesse da compreensão dos conceitos trabalhados, aspecto apontado pela literatura como relevante no desenvolvimento de escape rooms educacionais (VELDKAMP et al., 2020). O conteúdo aborda máquinas simples, com foco nas alavancas, destinado ao ensino de Ciências do sétimo ano do ensino fundamental.

O jogo é composto por dez salas ambientadas em um laboratório, cada uma dedicada a um aspecto do conceito, incluindo definição de alavanca, ponto de apoio, potência, resistência e tipos de alavanca. As mecânicas foram concebidas de modo a associar a resolução dos desafios à aplicação dos conceitos, fazendo da interação com o ambiente uma parte constitutiva da experiência de aprendizagem, e não apenas um recurso acessório.

RESULTADOS E DISCUSSÃO

O artefato produzido

A experiência resultou em um jogo funcional, publicado na plataforma e jogável em navegador ou dispositivo móvel. O ambiente reproduz um laboratório de máquinas simples, composto por dez salas sequenciais, com piso e paredes em tons claros, faixas de sinalização, luminárias e mobiliário de laboratório. Cada porta permanece fechada até que o jogador demonstre compreensão do conceito apresentado na sala.

Na primeira sala, uma alavanca manipulável reproduz o funcionamento descrito na teoria: ao acionar a extremidade correspondente à potência, a barra gira em torno do ponto de apoio e ergue uma pedra posicionada na extremidade oposta, com braços de comprimento desigual, de modo que a vantagem mecânica seja perceptível na própria ação. Na sala seguinte, um mural em relevo indica o ponto de apoio em uma alavanca. Na terceira sala, a

ISSN: 2358-8829

porta não apresenta questão a ser respondida, mas uma mensagem que orienta o jogador a utilizar sua força potente para puxar a alavanca instalada ao lado, de maneira que a aplicação do conceito, e não sua verbalização, seja a condição de passagem. Compõe ainda a experiência um personagem que percorre o laboratório e, ao alcançar o jogador, o reconduz ao início, elemento que introduz tensão narrativa sem recorrer a representações de violência.

A construção sem programação manual

O resultado central reside no modo como o jogo foi produzido. Cenários, mecânicas e scripts foram integralmente construídos a partir de descrições em linguagem natural, sem digitação de código pelo autor. O assistente de IAG operou diretamente no interior do próprio *Roblox Studio*, recebendo descrições em linguagem natural, gerando o código correspondente, inserindo-o na estrutura do projeto, executando-o, verificando seu comportamento e construindo toda a ambientação do jogo. O autor não redigiu, copiou ou inseriu manualmente qualquer linha de código, nem alocou objetos tridimensionais de forma manual. Sua participação consistiu em formular o que deveria existir, avaliar o resultado obtido, descrever ao assistente as divergências percebidas e solicitar correções aos erros observados.

Essas constatações dialogam diretamente com as barreiras identificadas na literatura. Diferentemente do que Kohen-Vacs, Usher e Jansen (2025, p. 3177) e Feldman e Anderson (2024, p. 13) apontaram, de que os não-especialistas fracassam por não dominarem o vocabulário técnico e por não interpretarem o código e as mensagens de erro, na experiência aqui relatada nenhuma dessas competências foi exigida do autor. As solicitações foram formuladas na perspectiva de um professor de educação básica, sem formação em programação, e a leitura das mensagens de erro coube ao próprio assistente interpretá-las e resolvê-las. Não se trata de afirmar que o problema foi resolvido, mas de observar que existe uma rota alternativa para não programadores, na qual a competência crítica passa a ser a de avaliar e solicitar o refinamento do resultado gerado pelo modelo de IAG.

Os pontos em que o processo exigiu intervenção

A construção não transcorreu de forma direta. O jogo resultou de refinamento iterativo sucessivo, e a distância entre o solicitado e o obtido constitui parte relevante dos resultados que esta experiência se propõe a documentar e discutir nesta seção. As divergências mais frequentes foram de natureza espacial e de proporção. Objetos posicionados por descrição verbal apresentavam-se afundados no piso, flutuando acima dele ou orientados de modo inadequado, como bancadas que, embora corretamente posicionadas segundo os valores numéricos, exibiam aparência de peças tombadas.

ISSN: 2358-8829

A causa de alguns dos problemas identificados é que os modelos tridimensionais do acervo da plataforma possuem pontos de referência que não coincidem com seu centro geométrico, de modo que instruções corretas produziam resultados visualmente equivocados, um problema elucidado durante interação com a ferramenta de IAG. Situação análoga ocorreu na animação do personagem não jogável gerado, cujas pernas oscilavam lateralmente, e não à frente, em razão da orientação dos eixos das articulações. Em ambos os casos, a solução exigiu que o pesquisador solicitasse ao assistente a conferência e a correção, procedimento possível apenas por sua capacidade de executar verificações no interior do *Roblox Studio*.

Após a publicação do artefato na plataforma, o acesso online passou a ser interrompido por falha na comunicação com os serviços de moderação da plataforma. A investigação através do assistente revelou que modelos decorativos obtidos gratuitamente no acervo da comunidade continham scripts de terceiros embutidos, alguns dos quais estabeleciam requisições a serviços externos sem relação com as funcionalidades pretendidas. A identificação e remoção desses scripts pelo assistente de IA restabeleceu o funcionamento.

O achado evidencia uma exigência não contemplada na literatura revisada. A mediação por inteligência artificial pode dispensar o desenvolvedor de escrever código, mas este ainda deve compreender que erros podem ocorrer durante o processo e que cabe a ele supervisionar e solicitar a correção através do assistente de IAG. Trata-se de competência de letramento digital distinta da competência em programação. Cabe observar que essas dificuldades convergem com o alerta de Pérez-Colado et al. (2023, p. 8) quanto à variabilidade dos resultados em função de pequenas variações nos comandos, e com a observação de Vincent e Ayyar (2023, p. 2) de que a eficácia da inteligência artificial cresce conforme aumenta o contexto disponível.

Implicações para a autoria docente

A experiência sugere que a barreira identificada por Murilo (2024), sobre o domínio de programação ser indispensável para criar experiências no Roblox, pode ser contornada quando a mediação se dá por IAG operando no interior do ambiente de desenvolvimento. Um tutorial pode auxiliar o professor na tarefa de programar, ao passo que a configuração aqui descrita transfere ao assistente de IAG essa parte operacional. No entanto, isso não assegura a autoria docente. Moon et al. (2024, p. 452) sustentam que a integração efetiva dessas tecnologias exige formação que abranja aspectos técnicos e pedagógicos, e os episódios relatados corroboram esse ponto, uma vez que a experiência exigiu do professor persistência no ciclo de verificação, clareza na formulação das intenções pedagógicas e capacidade de

ISSN: 2358-8829

reconhecer que um resultado aparentemente pronto ainda continha problemas que comprometiam sua funcionalidade pedagógica.

CONSIDERAÇÕES FINAIS

Este relato descreveu a criação de um jogo didático no Roblox por meio de comandos em linguagem natural, mediada por inteligência artificial generativa integrada ao ambiente de desenvolvimento. O percurso resultou em um jogo funcional e contextualizado, do tipo escape room, sobre máquinas simples, cujos cenários, mecânicas e scripts foram integralmente construídos a partir de descrições verbais, sem digitação de código pelo autor.

A experiência indica que a barreira técnica da programação, apontada na literatura como obstáculo central à autoria docente de jogos digitais, pode ser contornada quando a ferramenta de inteligência artificial opera no interior do ambiente de desenvolvimento, encarregando-se de gerar o código, executá-lo, verificá-lo e corrigi-lo. Nessa configuração, as duas competências apontadas como impeditivas para não-especialistas, o domínio do vocabulário técnico e a capacidade de interpretar e corrigir erros, deixam de ser exigidas do professor como pré-requisito para a autoria do recurso educacional.

A experiência evidenciou, contudo, que a mediação não elimina a necessidade de intervenção humana atenta e qualificada. Foram necessários refinamentos iterativos sucessivos, sobretudo em questões de proporção, posicionamento e animação, além de cuidados com modelos de terceiros, cujos scripts embutidos chegaram a comprometer o funcionamento do jogo publicado. O que a experiência sugere não é a eliminação da competência em escrita de scripts, mas elucida que há a necessidade da formulação clara de intenções pedagógicas, a verificação constante e o julgamento crítico para o que a ferramenta produz, bem como noções básicas de como as ferramentas de criação de jogos e de IAG funcionam no contexto de criação de recursos educacionais digitais.

Cabe reconhecer os limites deste relato. Trata-se de experiência singular, com um modelo de linguagem específico e em uma única plataforma, em condições de tempo que não correspondem às da prática docente cotidiana. O jogo não foi submetido à avaliação de sua efetividade como ferramenta didática, de modo que nada pode ser afirmado sobre seu impacto direto na aprendizagem dos estudantes a partir deste relato. Esses limites delineiam os desdobramentos de pesquisas futuras.

Aponta-se como continuidade a este estudo a análise do processo de criação junto a professores sem formação em programação, de modo a verificar a efetividade da proposta. Este desdobramento poderá contribuir para responder à questão da presente experiência, se a

ISSN: 2358-8829

mediação por inteligência artificial no interior do ambiente de desenvolvimento pode contribuir para tornar a autoria de jogos digitais uma prática efetivamente acessível ao professor da educação básica sem formação técnica em programação e em design de jogos digitais. Outro desdobramento é a própria avaliação do jogo, a experiência do jogador e sua contribuição na aprendizagem, em contextos e abordagens diversas.

AGRADECIMENTOS

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001. Os autores agradecem à Universidade Estadual de Goiás (UEG) e ao Programa de Pós-Graduação em Gestão, Educação e Tecnologias pelo suporte institucional que tornou possível o desenvolvimento desta pesquisa e a produção do presente trabalho.

REFERÊNCIAS

ANTUNES, Jeferson; TORRES, Cicero Magerbio Gomes; ALVES, Francione Charapa; QUEIROZ, Zuleide Fernandes de. Como escrever um relato de experiência de forma sistematizada? Contribuições metodológicas. **Práticas Educativas, Memórias e Oralidades - Rev. Pemo**, Fortaleza, v. 6, p. e12517, 2024. Disponível em: https://doi.org/10.47149/pemo.v6.e12517. Acesso em: 24 ago. 2026.

FELDMAN, Molly Q.; ANDERSON, Carolyn Jane. Non-expert programmers in the generative AI future. In: SYMPOSIUM ON HUMAN-COMPUTER INTERACTION FOR WORK, 3., 2024, Newcastle upon Tyne. **Proceedings** [...]. New York: ACM, 2024. p. 1-19. Disponível em: https://doi.org/10.1145/3663384.3663393. Acesso em: 12 jul. 2026.

GEE, James Paul. Bons video games e boa aprendizagem. **Perspectiva**, Florianópolis, v. 27, n. 1, p. 167-178, 2009. Disponível em: https://doi.org/10.5007/2175-795X.2009v27n1p167. Acesso em: 14 jul. 2026.

HO, Won; LEE, Daehyun. Enhancing engineering education in the Roblox metaverse: utilizing chatGPT for game development for electrical machine course. **International Journal on Advanced Science, Engineering and Information Technology**, v. 13, n. 3, p. 1052-1058, 2023. Disponível em: https://doi.org/10.18517/ijaseit.13.3.18458. Acesso em: 16 jul. 2026.

JEONG, Yeil et al. Navigating the creation of immersive learning environments in Roblox: integrating generative AI for enhanced simulation-based learning. **Immersive Learning Research - Practitioner**, v. 1, n. 1, p. 16-19, 2024. Disponível em: https://doi.org/10.56198/5M1RHT9ZJ. Acesso em: 19 jul. 2026.

KOHEN-VACS, Dan; USHER, Maya; JANSEN, Marc. Integrating generative AI into programming education: student perceptions and the challenge of correcting AI errors. **International Journal of Artificial Intelligence in Education**, v. 35, p. 3166-3184, 2025. Disponível em: https://doi.org/10.1007/s40593-025-00496-4. Acesso em: 23 jul. 2026.

ISSN: 2358-8829

KUMAR, Aayush; BAJPAI, Yasharth; GULWANI, Sumit; SOARES, Gustavo; MURPHY-HILL, Emerson R. Sharp Tools: How Developers Wield Agentic AI in Real Software Engineering Tasks. [S. l.]: **arXiv**, 2025. Disponível em: https://doi.org/10.48550/arXiv.2506.12347. Acesso em: 24 ago. 2026.

MOON, Jewoong et al. Generative artificial intelligence in educational game design: nuanced challenges, design implications, and future research. **Technology, Knowledge and Learning**, v. 30, p. 447-459, 2025. Disponível em: https://doi.org/10.1007/s10758-024-09756-z. Acesso em: 26 jul. 2026.

MURILO, Jorge Luís Vieira. **O uso da plataforma Roblox para a criação de objetos virtuais de aprendizagem**: um estudo por meio do desenvolvimento de experiências e elaboração de tutorial. 2024. 62 f. Trabalho de Conclusão de Curso (Bacharelado em Engenharia de Computação) - Instituto Federal de Minas Gerais, Campus Bambuí, Bambuí, 2024. Disponível em: https://repositorio.ifmg.edu.br/items/4a746a59-98e7-421f-bd3a-3a8ce46edd71. Acesso em: 30 jul. 2026.

PÉREZ-COLADO, Iván J. et al. Using new AI-driven techniques to ease serious games authoring. In: IEEE FRONTIERS IN EDUCATION CONFERENCE, 2023, College Station. **Proceedings** [...]. [S. l.]: IEEE, 2023. p. 1-9. Disponível em: https://doi.org/10.1109/FIE58773.2023.10343021. Acesso em: 2 ago. 2026.

PRENSKY, Marc. Digital natives, digital immigrants. **On the Horizon**, v. 9, n. 5, p. 1-5, 2001. Disponível em: https://doi.org/10.1108/10748120110424816. Acesso em: 9 ago. 2026.

VELDKAMP, Alice; VAN DE GRINT, Liesbeth; KNIPPELS, Marie-Christine P. J.; VAN JOOLINGEN, Wouter R. Escape education: A systematic review on escape rooms in education. **Educational Research Review**, v. 31, p. 100364, 2020. Disponível em: https://doi.org/10.1016/j.edurev.2020.100364. Acesso em: 24 ago. 2026.

VINCENT, Brent; AYYAR, Kartik. Roblox generative AI in action. In: SPECIAL INTEREST GROUP ON COMPUTER GRAPHICS AND INTERACTIVE TECHNIQUES CONFERENCE REAL-TIME LIVE!, 2023, Los Angeles. **Proceedings** [...]. New York: ACM, 2023. p. 1-2. Disponível em: https://doi.org/10.1145/3588430.3597250. Acesso em: 5 ago. 2026.

WANG, Huanting; GONG, Jingzhi; ZHANG, Huawei; ZHENG, Wang. AI Agentic Programming: A Survey of Techniques, Challenges, and Opportunities. [S. l.]: **arXiv**, 2025. Disponível em: https://doi.org/10.48550/arXiv.2508.11126. Acesso em: 24 ago. 2026.

YANG, Tianyu. Natural language-driven AI programming: exploring a gamified learning method with broad application potential. In: GUANGDONG-HONG KONG-MACAO GREATER BAY AREA EDUCATION DIGITALIZATION AND COMPUTER SCIENCE INTERNATIONAL CONFERENCE, 2., 2025, Shenzhen. **Proceedings** [...]. New York: ACM, 2025. p. 1-6. Disponível em: https://doi.org/10.1145/3746469.3746472. Acesso em: 6 ago. 2026.