

Ensinando *Python* com LEGOTM *Spike Prime*: Uma Abordagem Prática para Jovens de 13 a 18 Anos

Giulia Ramos de Oliveira¹, Rafael Rocha Ribeiro¹, Nina Aguiar Ferreira²,
Miguel Dias de Abreu², Cristina Sayuri Côrtes Ouchi Dusi³

¹Instituto de Ciências Exatas (ICE)
Universidade Federal de Juiz de Fora (UFJF) – Juiz de Fora, MG

{rafael.rocha, giulia.amos}@estudante.ufjf.br

²Departamento de Ciência da Computação (DCC)
Universidade Federal de Juiz de Fora (UFJF) – Juiz de Fora, MG

{nina.aguiar, miguel.dias}@estudante.ufjf.br

³Faculdade de Administração e Ciências Contábeis (FACC)
Universidade Federal de Juiz de Fora (UFJF) – Juiz de Fora, MG

cristina.dusi@ufjf.br

Abstract. *Teaching computational thinking is essential for developing logical reasoning, preparing individuals for future challenges. This article reports on an experience in teaching Python programming integrated with LEGOTM Spike kits, aimed at students aged 13 to 18 with prior experience in block-based programming. The lessons covered basic programming and robotics concepts, culminating in a competition. The here presented methodology promoted the development of cognitive skills, problem-solving, and teamwork, while also stimulating creativity and critical thinking. The results reinforce the value of integrating programming and robotics as an effective and motivating approach to teaching technology and logic.*

Resumo. *O ensino do pensamento computacional é essencial para o desenvolvimento do raciocínio lógico, preparando pessoas para futuros desafios. Este artigo relata uma experiência de ensino de programação em Python junto ao uso de kits LEGOTM Spike com jovens de 13 a 18 anos com experiência prévia em programação com blocos. As aulas abordaram conceitos de programação e robótica, culminando em uma competição. A metodologia aqui apresentada promoveu o desenvolvimento de habilidades cognitivas, resolução de problemas e trabalho em equipe, além de estimular a criatividade e o pensamento crítico. Os resultados reforçam o valor da integração entre programação e robótica como abordagem eficaz e motivadora no ensino de tecnologia e lógica.*

1. Introdução

O ensino de adolescentes representa um desafio complexo, considerando a gama de experiências emocionais e a dificuldade inerente à resolução de problemas típicos dessa fase, fatores que podem desviar o foco dos estudos [Ferreira et al. 2021]. Adicionalmente, a

problemática se agrava com currículos por vezes deficientes, escassez de recursos e despreparo de docentes e gestores escolares, o que contribui para o desinteresse dos estudantes, que, conforme apontam pesquisas, demonstram preferência por metodologias ativas [Andrada et al. 2018]. Nesse contexto, salienta-se a crescente relevância da programação, tanto no cotidiano quanto no âmbito educacional [Tavares et al. 2021]. Ela oferece oportunidades para o desenvolvimento e aprimoramento do pensamento racional e coeso, da lógica computacional e da resolução de problemas [Silva et al. 2023]. Acreditamos que a integração da programação com a robótica e o universo já conhecido pelos estudantes de 13 a 18 anos pode ampliar ainda mais as possibilidades de aprimoramento de suas habilidades. É sob essa perspectiva que se destaca a importância da análise apresentada neste artigo.

Diante desse cenário, o projeto "Minds of The Future" foi criado com o objetivo de fornecer uma base de aprendizado digital e empreendedora, visando o ensino de diferentes linguagens de programação assim como o uso de tecnologias. O público-alvo do projeto são alunos das escolas estaduais de Minas Gerais, porque, além do patrocínio da própria Secretaria de Educação, o projeto espera atender quem não teria acesso a cursos, muitas vezes caros, de programação. Espera-se, também, que o projeto contribua para que estudantes de graduação acessem conhecimentos sobre inovação, empreendedorismo, robótica e uso de metodologias ativas de ensino de maneira facilitada.

Desde 2023, o projeto "Minds of The Future" tem recebido diversos estudantes das redes estadual e municipal, bem como de algumas escolas particulares, para participação nas aulas interativas com material LEGOTM *Spike Prime*. As turmas se dividem atualmente em dois módulos temáticos: programação em bloco e programação em *Python*. Este artigo apresenta um relato sobre o ensino na turma Módulo 2: LEGOTM e *Python*, que contou com a participação de 20 alunos reincidentes do Módulo 1: LEGOTM e Programação em Blocos.

É importante destacar que as aulas descritas neste artigo utilizaram amplamente com o uso dos materiais didáticos da LEGOTM *for Education*, assim como os kits de montagem *Spike Prime* e materiais desenvolvidos pelos próprios bolsistas-colaboradores do Projeto de Pesquisa. De modo semelhante ao realizado por [Vahldick et al. 2009] com o kit *Mindstorms*, o material foi utilizado para desenvolvimento das competências individuais e coletivas de lógica através de técnicas que vão ao encontro do ensino tradicional, possibilitando liberdade para os alunos e a realização de testes criativos por esses em um ambiente controlado e propício ao aprendizado.

Este artigo está organizado da seguinte forma: a Seção 2 aborda trabalhos relacionados a esta proposta; a Seção 3 descreve os materiais e métodos empregados e por fim, a Seção 4 mostra os resultados iniciais obtidos e a Seção 5 traz as considerações finais.

2. Trabalhos Relacionados

A integração de pensamento computacional, robótica e metodologias ativas no ensino médio tem demonstrado ser uma abordagem promissora para aumentar o engajamento e a permanência dos estudantes nas escolas, além de fomentar o interesse por carreiras tecnológicas [Andriola 2021]. Diversos estudos exploram esses temas, destacando-se pela relevância e alinhamento com a proposta do projeto "Minds of The Future".

Um exemplo é o estudo sobre jogos educacionais como ferramenta de ensino.

No trabalho "BeatTheCode: Um Jogo para Aprendizagem em Programação com Worked Examples"[Leifheit et al. 2024], os autores discutem como a natureza lúdica dos jogos eleva o interesse dos alunos e facilita a compreensão de conceitos de programação. Esse tipo de recurso se aproxima do material LEGOTM utilizado no presente projeto, que também estimula a criatividade e o raciocínio lógico em um ambiente dinâmico e envolvente.

Outro destaque é a oficina "Programando sua Independência Financeira", realizada com alunos do primeiro ano do ensino médio em Porto Alegre. Essa iniciativa promoveu a educação financeira por meio da programação em *Python*, demonstrando como a linguagem pode ser aplicada em contextos práticos e cidadãos [Perosa 2021], de forma similar ao projeto "Minds of The Future". Através da oficina, foi possível mostrar que a programação é uma ferramenta poderosa para o desenvolvimento de habilidades empreendedoras e para a formação de jovens mais autônomos e preparados para o futuro.

O "Education 4.0 Challenge"[de Mira et al. 2023] também é uma contribuição relevante ao explorar como a educação pode ser transformada pela integração de tecnologias digitais, pensamento computacional e habilidades do século XXI. Este estudo enfatiza a necessidade de adaptar o currículo escolar às demandas contemporâneas, promovendo aprendizado baseado em projetos e resolução de problemas. As propostas descritas convergem com as práticas do projeto "Minds of The Future" ao priorizar a interdisciplinaridade, o uso de tecnologia de ponta e a capacitação dos estudantes para enfrentarem desafios futuros.

Adicionalmente, o trabalho [Benitti et al. 2009] sobre o uso do kit *Mindstorms* da LEGOTM é uma referência importante. Nesse estudo, os autores demonstram como a robótica educacional pode ser utilizada para o desenvolvimento de competências lógico-matemáticas e habilidades colaborativas. O uso de materiais como os kits de montagem *Spike Prime* no projeto segue a mesma linha, promovendo o aprendizado prático, a resolução criativa de problemas e a autonomia dos estudantes em um ambiente estruturado, mesmo que ainda não existam numerosos trabalhos com o mesmo kit.

Esses estudos evidenciam que metodologias ativas de base digital, como as utilizadas no Projeto de Pesquisa aqui relatado, são eficazes no ensino de programação e robótica, podendo proporcionar um aprendizado significativo e alinhado às demandas contemporâneas.

3. Metodologia

Para desenvolvermos os cursos oferecidos no projeto, houve embasamento na metodologia STEAM (*Science, Technology, Engineering, Arts and Mathematics*). Essa proposta consiste na integração da ciência, tecnologia, engenharia, artes e matemática através de recursos interdisciplinares, que têm foco no desenvolvimento de pensamentos inovadores:

"A ideia por trás do STEAM na educação é romper barreiras entre disciplinas. Trata-se da interdisciplinaridade por excelência. As disciplinas STEAM são trabalhadas de forma conjunta permitindo ao estudante a mobilização de habilidades e saberes de forma integrada e concorrendo para uma aprendizagem significativa. Há a ênfase no trabalho em conjunto, que propicia, a cada estudante, o desempenho de funções e ativi-

des que utilizem e desenvolvam suas habilidades e competências contribuindo para a aprendizagem comum.”[Silva et al. 2017]

Tendo isso em mente, a aplicação STEAM possibilitou a conexão necessária entre as disciplinas supracitadas. A ciência conseguiu explicar aos discentes o funcionamento físico das peças, como os sensores de distância e pressão, assim como, através da ciência da computação e suas linguagens, conseguimos introduzir formas de pensamento computacional. Ao mesmo tempo, a tecnologia geral se manifestou no uso dos computadores, muitas vezes desconhecidos aos alunos, e nos conceitos de software e hardware. Já para as engenharias, pode-se pensar na robótica, mecânica e eletrônica, responsáveis pela tradução das peças de hardware e a criação dos robôs, que também envolve criatividade e pensamento coletivo, como nas disciplinas de artes e humanidades. Por fim, a matemática se faz importante por meio do raciocínio lógico e habilidades de pensamento rápido, trabalhando com o *Python* para desenvolver códigos mais complexos com condicionais e comparações de magnitude.

É necessário pontuar que o projeto alvo deste artigo não foi o primeiro a unir o ensino e a STEAM para os alunos, muitas vezes desinteressados no acadêmico, vindos de escolas estaduais. [Campos et al. 2022] realizaram pesquisas para o aprofundamento do que se encontra na literatura atual sobre o uso dos métodos citados anteriormente, e podemos destacar a aplicação citada em um dos tópicos de resultados e análises:

”Pensando no desestímulo dos alunos do ensino médio, Vuerzler (2020), trouxe uma proposta da aprendizagem ativa colaborativa junto a abordagem STEAM, criando uma sequência didática investigativa integrativa com estudantes e professores de uma escola estadual.”[Campos et al. 2022]

Logo, a metodologia escolhida e executada pelo Projeto de Pesquisa já tem se mostrado comprovadamente eficaz e em uma evolução crescente diante dos desafios que o mundo globalizado e tecnológico criou.

3.1. Plano de Aula

Nos últimos anos, o pensamento computacional tem sido amplamente reconhecido como uma habilidade fundamental para a educação. Este envolve a capacidade de resolver problemas de maneira sistemática e lógica, decompondo tarefas complexas em partes menores, identificando padrões e criando soluções eficazes. Quando introduzimos o ensino de programação para jovens de 13 a 18 anos, estamos não apenas ensinando uma nova linguagem, mas também promovendo habilidades cognitivas que serão essenciais em diversas áreas, desde a ciência e tecnologia até a resolução de problemas do cotidiano [Wing 2006].

Nesse contexto, o uso do *Python*, aliado ao LEGOTM *Spike Prime*, representa uma ferramenta poderosa para desenvolver o pensamento computacional de forma prática e interativa. O *Python* é uma linguagem acessível e versátil, por ser de alto nível, enquanto o LEGOTM *Spike Prime* permite que os alunos vejam o impacto de seus algoritmos em tempo real e de forma palpável, por meio do controle de robôs que respondem a comandos, sensores e condições. Dessa forma, cada aula não se limita à programação, mas também envolve a criação de estratégias, a adaptação às falhas e a busca por soluções alternativas; competências centrais para o pensamento computacional.

O plano de aula a seguir foi estruturado para permitir que os alunos evoluam de conceitos básicos de *Python*, como condicionais e *loops*, até estruturas mais complexas, como o uso de sensores e motores nos robôs montados por eles próprios. Com os conhecimentos prévios dos bolsistas, foram planejadas aulas baseadas na ordem de disciplinas de base da computação, ou seja, baseadas inteiramente em experiências pessoais conjuntas.

Ao longo de seis encontros de duas horas cada, os alunos utilizaram o pensamento computacional para programar robôs que interagem com o ambiente e solucionam desafios específicos. A atividade finalizou com uma competição, na qual puderam colocar à prova suas habilidades de programação e raciocínio lógico. Essa metodologia prática e estimulante buscou fortalecer o entendimento da lógica de programação, reforçando o pensamento computacional como uma competência essencial em sua formação.

As aulas foram estruturadas de modo a contar com a presença de quatro a cinco monitores, sendo um desses monitores também o professor da aula (com experiência e capacitação prévia para tal tarefa); a Figura 1 ilustra um dos encontros, onde todos que estão de pé são os colaboradores supracitados. Os monitores são os responsáveis em protagonizar os alunos, fazendo com que se sintam confortáveis em tirar dúvidas, escrever códigos inovadores e montarem os robôs do jeito que mais desejam - ou seja, apesar dos colaboradores guardarem os conhecimentos sobre *Python*, os estudantes que são os donos de sua criação e devem desenvolver pensamentos criativos sozinhos.



Figura 1. Momento de uma aula de *Python* em uma das salas do projeto. Imagem advinda do acervo próprio do projeto.

Aula 1 - "Luzes, sons e ação!". Na primeira aula, feita com base na ideia de *pair programming* - que consiste basicamente em organizar os alunos em duplas -, introduziram-se os conceitos básicos de *software* e *hardware*, pensando nos jovens mais leigos no assunto, e foi seguida da introdução ao *Python* como linguagem de programação. Foi feita a importação das bibliotecas necessárias para comunicação com o LEGOTM Spike e com isso feito, os alunos, depois de imprimirem o seu primeiro "Olá mundo!", aprenderam a emitir *beeps*, desenhar na tela com o *hub* e programar motores individuais, o que lhes permitiu dar os primeiros passos no controle do robô.

Aula 2 - "Movendo que se aprende". A segunda aula focou no controle coordenado de motores, capacitando os alunos a programar movimentos sincronizados com uma base motriz feita por eles. Iniciou-se com a diferenciação do funcionamento dos motores

em pares e sozinhos, e dos conceitos de sentido horário e anti-horário, onde os alunos configuraram e testaram movimentos como andar em linha reta e realizar curvas. Essa etapa reforçou não só a leitura de código, mas também o uso do *await* para evitar erros de execução, já que falamos que o código "se atropelaria" sem a presença desse simples comando. Foram propostos desafios práticos, como fazer o robô girar em seu próprio eixo e realizar manobras, preparando os alunos para o uso de sensores em atividades futuras, como a própria competição.

Aula 3 - "Mas e se eu...?". Na terceira aula, os alunos começaram a trabalhar com o sensor de cor e aprenderam a usar estruturas condicionais (*if*, *elif*, *else*) para criar árvores de decisão e usaram um contador com *for*. Foi apresentada uma atividade lúdica chamada "Adivinhe a Cor", em que sons eram associados às cores detectadas pelo sensor, incentivando a compreensão prática das condicionais junto à leitura de código. Os alunos também montaram o robô treinador "Leo", que simulava movimentos repetitivos de academia usando o *for*, consolidando o aprendizado sobre repetição e condicionais.

Aula 4 - "Aperte 1 para motores, 2 para sensores ou 3 para sair". A quarta aula introduziu o uso do sensor de pressão e aprofundou o trabalho com motores e sensores em conjunto, utilizando a criação de funções e comandos como *while True* e *until*. Os alunos aprenderam a programar o robô para interagir de forma autônoma com o ambiente, simulando o comportamento de um aspirador robô que manobra ao detectar um obstáculo. Essa prática consolidou o uso de funções e condicionais para desenvolver autonomia e dinamicidade nos movimentos do robô.

Aula 5 - "Desafiando". Na quinta aula, os alunos foram desafiados a aplicar de maneira autônoma os conceitos aprendidos ao longo do curso, construindo um carrinho de entrega equipado com sensores de distância e cor. Os alunos usaram um glossário de funções para programar o carrinho de forma independente, integrando comandos *if*, *elif* e *until*. Essa atividade teve como foco avaliar a auto adaptação dos alunos para a competição final, estimulando o pensamento crítico e a solução de problemas com a ajuda mínima dos monitores.

Aula 6 - "Competição". A última aula foi dedicada à competição. Os carrinhos foram ajustados para uma batalha amigável, com cada robô tendo um balão acoplado na parte de trás e um palito de churrasco na frente, todos com o objetivo de estourar o balão do adversário dentro de uma arena, como é possível visualizar na Figura 2. A competição estimulou o trabalho em equipe, criatividade e aplicação prática dos conceitos aprendidos. Importante mencionar que todos os palitos e balões foram instalados nos robôs após a montagem dos alunos pelos monitores e após esse momento os jovens não poderiam mais chegar perto dos robôs sem supervisão.

4. Resultados e Discussões

4.1. Aplicação e Desafios

Foi observado que os alunos mais velhos, entre 16 e 18 anos, demonstraram maior facilidade em compreender a lógica de programação, enquanto os alunos mais novos (13 e 15 anos) enfrentaram mais dificuldades, especialmente na transição entre programação em blocos e programação textual. Isso se deve ao fato de que, na programação textual, os alunos precisam interpretar o programa de maneira abstrata e



Figura 2. Momento da competição da aula 6 na sala de aula. Imagem advinda do acervo próprio do projeto.

sem suporte visual, o que pode tornar o processo mais complexo em comparação com a programação em blocos, que apresenta uma interface mais intuitiva e acessível para iniciantes [Weintrop and Wilensky 2015].

Um dos principais desafios percebidos foi o grau de familiaridade com a tecnologia, já que alunos que tinham pouca ou nenhuma familiaridade com informática inicialmente enfrentaram dificuldades para assimilar os conceitos básicos de *Python* e robótica. Outro obstáculo foi a atenção e engajamento, pois alguns alunos demonstraram falta de atenção nas aulas teóricas, o que resultou em desempenho fraco nos desafios práticos, assim como seu nível de autonomia - como a quinta aula foi um "teste de autonomia", os alunos que participaram ativamente das aulas anteriores se destacaram, enquanto aqueles que não acompanharam de perto o conteúdo tiveram dificuldade em completar suas tarefas.

4.2. Benefícios Observados

Por outro lado, tivemos como principais benefícios: a integração de *Python* e robótica - essa integração provou ser uma ferramenta poderosa para o ensino de lógica de programação, tornando o aprendizado mais dinâmico e visual; o desenvolvimento de habilidades cognitivas, de maneira que a competição final incentivou os alunos a pensar criticamente e aplicar os conceitos de maneira criativa e prática. Essa experiência também sugeriu que ajustes no plano de aula seriam necessários para oferecer mais suporte a alunos com menos experiência e para garantir revisões intermediárias nos conceitos voltados à lógica de programação e partes não tão intuitivas do *Python*, como lidar com erros, por exemplo.

4.3. Estratégias de Apoio e Feedback

Materiais de apoio adicionais foram desenvolvidos para auxiliar alunos que apresentavam dificuldades com programação e interação com tecnologias digitais. Durante as aulas,

muitos alunos necessitaram de assistência contínua, o que resultou em *feedbacks* positivos relatados pelos discentes. Para tornar o conteúdo mais acessível, os bolsistas aproximaram conceitos complexos de programação ao cotidiano dos participantes: utilizaram, por exemplo, corridas esportivas para explicar atividades paralelas e funções assíncronas; times de futebol para criar um ambiente mais familiarizado com o material, e atividades que envolviam tomadas de decisão para introduzir intuitivamente à lógica de algoritmos.

4.4. Percepções em relação aos participantes

Foi essencial considerar as experiências passadas pelos alunos tanto para a realização eficaz das aulas, quanto para uma re-elaboração do material didático, levando em conta o contexto socioeconômico em que os participantes do projeto estavam inseridos. Advindos, em sua grande maioria, da rede pública de ensino, seu dificultado acesso às tecnologias de custosa aquisição se tornou um peso capaz de retardar seu aprendizado, mas que pudemos ver que foi abandonado no caminho [Silva and Hasenbalg 2000].

Como estratégias de registro, os docentes realizaram observações em sala, posteriormente discutidas em reuniões para identificar e resolver questões pedagógicas com a devida atenção. Além disso, foi oferecido um sistema de *feedback* (por meio de um formulário da ferramenta *Google Forms*) ao final do módulo com perguntas sobre o impacto do projeto em suas perspectivas profissionais, a qualidade do material ofertado e o nível de conforto com as tecnologias apresentadas. Essas práticas visavam assegurar a qualidade e aprimorar o planejamento dos módulos futuros. Ao final, recebemos um número muito baixo de respostas para uma análise aprofundada, porém pudemos observar que o projeto ajudou a guiar os alunos para uma trilha universitária, assim como foi apontada a necessidade de dar mais liberdade dentro dos materiais aos alunos.

5. Considerações Finais

Neste artigo, apresentamos um relato de experiência sobre a primeira implementação do módulo de ensino à *Python* no projeto "Minds of The Future", desenvolvido para promover o aprendizado de programação, do pensamento computacional e de robótica em alunos do ensino médio de escolas públicas. A proposta utilizou metodologias ativas e a abordagem STEAM para proporcionar um ambiente de ensino interativo, inovador e alinhado às demandas do mundo moderno.

A experiência destacou os benefícios de combinar a prática da robótica, com seu caráter lúdico, ao ensino de lógica de programação, demonstrando que ferramentas como o kit LEGOTM *Spike Prime* são eficazes para facilitar o aprendizado de linguagens de programação. Ao longo do projeto, os participantes desenvolveram competências essenciais, como raciocínio lógico, resolução de problemas e trabalho em equipe, caminhando para uma competição final que reforçou sua autonomia e criatividade.

Entretanto, o projeto também revelou desafios importantes, como a necessidade de um suporte adicional para alunos com menor conhecimento tecnológico e auxílio na transição para a programação textual. Essa mudança entre a programação em blocos e textual destacou a importância de uma abordagem gradual e contextualizada, especialmente para os estudantes mais jovens. Estratégias de apoio, como o uso de exemplos do cotidiano e analogias, foram fundamentais para reduzir essas dificuldades e devem ser aprimoradas em futuras edições.

Para os organizadores, a experiência trouxe um melhor entendimento sobre planejamento, execução e adaptação de metodologias de ensino em contextos educacionais desafiadores e inovadores. O *feedback* dos alunos e monitores reforçou o impacto positivo do projeto, mas também apontou a importância de uma avaliação contínua e colaborativa para ajustar o planejamento às necessidades específicas de cada turma.

Como próximo passo, o projeto busca ampliar seu alcance, explorando novos contextos e integrando novas tecnologias e metodologias ativas adicionais. Essa expansão pode abrir portas para uma análise mais aprofundada sobre o impacto a longo prazo na formação dos alunos, reafirmando o papel central de práticas educacionais interativas no ensino de Computação.

Este relato reforça a importância de iniciativas que promovam inclusão e inovação no ensino médio, evidenciando como projetos dessa natureza podem transformar a realidade educacional de estudantes do Ensino Médio da rede pública no Brasil e oferecer uma formação mais alinhada às exigências contemporâneas.

Referências

- Andrada, P. C. d., Oliveira, M. C. d., Cruz, P. S. G. d., Correia, C. M. R., and Paiva, M. d. (2018). O desinteresse dos alunos de ensino médio pela escola na atualidade. *Momentum: Revista Eletrônica*, 16(1).
- Andriola, W. B. (2021). Impactos da robótica no ensino básico: estudo comparativo entre escolas públicas e privadas. *Ciência Educação (Bauru)*, 27:e21050.
- Benitti, F. B. V., Vahldick, A., Urban, D. L., Krueger, M. L., and Halma, A. (2009). Experimentação com robótica educativa no ensino médio: ambiente, atividades e resultados. In *Anais do Workshop de Informática na Escola*, volume 1, pages 1811–1820.
- Campos, D. C., Lima, E. J. d., Cintra, D. D., and Moraes, D. V. d. (2022). A abordagem steam e suas tendências pedagógicas e metodológicas. *Research, Society and Development*, 11(15):e190111537148.
- de Mira, I. R. C., Silva, G. K., Siqueira, D. M. R., de Sousa, L. M. C., Faria, L., Pereira, I. P., de Freitas Bueno, M. d. F., di Salvo, A. L. A., et al. (2023). Education 4.0 challenge. *15º JORNADA CIENTÍFICA E TECNOLÓGICA E 12 º SIMPÓSIO DE PÓS-GRADUAÇÃO DO IFSULDEMINAS*, 15(2).
- Ferreira, A. C., Gonzales Martins, L., Soares de Jesus, J., Neves, M. A. P., Arinelli, G. S., and Trevisan de Souza, V. L. (2021). Adolescentes desinteressados? reflexões de estudantes do ensino médio público sobre sua escola. *Revista de Psicologia*, 30(1).
- Leifheit, B., Junior, C. C., Tonhão, S., and Silva, W. (2024). Beatthecode: Um jogo para aprendizagem em programação com worked examples. In *Anais Estendidos do IV Simpósio Brasileiro de Educação em Computação*, pages 25–26, Porto Alegre, RS, Brasil. SBC.
- Perosa, P. (2021). Programação em python no ensino médio: uma proposta em educação financeira.
- Silva, I. O. d., Rosa, J. E. B., Hardoim, E. L., and Guarim-Neto, G. (2017). Educação científica empregando o método steam e um makerspace a partir de uma aula-passeio. *Latin American Journal of Science Education*, 4(2):1–9.

- Silva, L. M., Leite, M. S., Melo, L. A. O. d., Freitas, M. J. B. M., and Rodrigues, R. L. (2023). Avaliação da eficiência das metodologias de problem based learning e rotação por estações em conjunto durante o ensino de robótica. *RENOTE - Revista Novas Tecnologias na Educação*, 21(1):373–382.
- Silva, N. d. V. and Hasenbalg, C. (2000). Tendências da desigualdade educacional no brasil. *Dados*, 43(3):423–445.
- Tavares, M. F. C., Pinto, J. A., and Magalhães, C. S. d. (2021). A utilização da robótica educacional e gamificação empregando o kit ev3 lego: buscando alternativas para o ensino de física em sintonia com os alunos da geração atual. *Revista Valore*, 6:1278–1293.
- Vahldick, A., Barreto, F., Benitti, F., Urban, D., Krueger, M., and Halma, A. (2009). O uso do lego mindstorms no apoio ao ensino de programação de computadores. *ResearchGate*.
- Weintrop, D. and Wilensky, U. (2015). To block or not to block, that is the question: Students’ perceptions of blocks-based programming. In *Proceedings of the 14th International Conference on Interaction Design and Children*, pages 199–208.
- Wing, J. M. (2006). Computational thinking. *Commun. ACM*, 49(3):33–35.