

4

José Maurício Schneedorf Ferreira da Silva

JSPLOTLY:

UM APLICATIVO PRÁTICO
E PORTÁVEL PARA SIMULAÇÕES
GRÁFICAS INTERATIVAS E DINÂMICAS
AO ENSINO REPRODUTÍVEL

DOI: 10.31560/pimentacultural/978-85-7221-396-7.4

EDUCAÇÃO 5.0 E DESAFIOS CONTEMPORÂNEOS

A *Educação 5.0* tem sido tema crescente em nossos tempos, ao combinar tecnologias imersivas de realidade aumentada e virtual, aprendizado de máquina, *big data*, internet das coisas (IoT) e inteligência artificial, heranças recentes da Educação 4.0, aos valores de impacto social e às transformações deletérias ao planeta (Vieira *et al.*, 2023). No outro canto do ringue, uma juventude dos assim chamados "*nativos digitais*", jovens focados no uso de dispositivos móveis para toda uma gama de interações sociais mediadas por tecnologia. Não obstante esse matrimônio potencial, reporta-se progressivamente a marcante dicotomia entre o acesso quase irrestrito à informação digital nesses tempos, e a apropriação geracional de ferramentas correlatas (Prakash *et al.*, 2023). Embora qualquer jovem hoje em dia consiga absorver aspectos de mérito individual sobre áudios de redes sociais e plataformas de streaming em velocidade "1.5", raros são os que coadunam com as *competências digitais* recomendadas pela União Europeia desde 2006 para o uso crítico, reflexivo e criativo das novas tecnologias de informação e comunicação, atualizadas no ano corrente pelo Digital Competence Framework for Educators (DigCompEdu) (European Commission, 2025).

ENSINO REPRODUTÍVEL E APLICATIVOS GRÁFICOS

A reprodutibilidade em pesquisa tem sido amplamente discutida há tempos (Most, 1964), tornando-se atualmente um monólito da premissa de qualidade científica com o apoio de *documentos dinâmicos* (Buckley *et al.*, 2025). A mesma preocupação, contudo, não

se transfere ao ensino e aprendizagem, embora haja um movimento inercial de iniciativas seminais a esse respeito (Dogucu, 2024). Não obstante, o ensino reprodutível (ER) modela uma abordagem ativa com uso de linguagem de programação, seu ponto forte à reprodutibilidade garantida do objeto didático ou documento a contemplar, muito embora percebida como uma barreira intransponível, se refletirmos acerca de seu público-alvo, professores e alunos das mais diversas competências (Pownall *et al.*, 2022). O uso de programação ao ER remonta ao sistema *WEB*, desenvolvido por Donald Knuth na Universidade de Stanford nos anos 1984 (1984). Contemplando uma combinação da linguagem de programação *Pascal* com a linguagem para formatação documental *Tex*, o matemático e cientista da computação laureado com o *Turing Award* obteve um *documento dinâmico* para edição manual de textos e códigos para compilação automático ao estudo de números primos.

Um dos desafios para a absorção de conteúdos acadêmicos e escolares diz respeito à interpretação de gráficos presentes em livros-texto e conteúdos afins, em especial aos que se referem a funções matemáticas em áreas não puramente matemáticas, como gráficos científicos das ciências naturais. Essa dificuldade parece repousar na construção mental e nas ações necessárias para a interpretação de quantidades desenhadas implicitamente à visualização e sob mudança das coordenadas de dados nos eixos gráficos (Altindis *et al.*, 2024), um requisito à incorporação de forma (*equação*) ao contexto (*fenômeno*). Não obstante a “era” de maior percepção à construção gráfica em papel quadriculado ou milimetrado, aplicativos web e dispositivos móveis vêm contemplando um exército crescente de admiradores, muito embora, mais por facilidades de uso dos desenvolvedores e da estética do produto final do que por seu cerne conteudista.

Diversas características comuns ou específicas desses programas favorecem a reprodutibilidade científica e de ensino-aprendizagem. Entretanto, se por um lado esses sistemas permitem

um estudo avançado das características de modelagem propostas, em contrapartida apresentam:

Características de programas convencionais para construção de gráficos:

1. Necessidade de instalação ou acesso credenciado em nuvem para programa/pacotes;
2. Dependência de licença, ainda que temporária, para uso de programas;
3. Necessidade de conexão à internet para as versões em nuvem dos programas;
4. Preços altos ou versões de demonstração limitadas para os programas pagos;
5. Usabilidade dependente de janelas, menus e abas específicas de cada suíte;
6. Limitação de modelos matemáticos [propostos](#) (embora o conhecimento da sintaxe de cada programa permita a inserção de equações pelo usuário);
7. Presença de diversas janelas e abas individuais para acesso às informações de inserção e visualização gráfica de dados e funções;
8. Programas que usualmente operam por cliques de mouse não permitem ao aprendiz a inserção em pensamento computacional envolvido em linguagens de programação (por exemplo, Prism, Geogebra, Origin e Sigma Plot);
9. Programas que não trabalham com linhas de comando produzem os resultados gráficos em janelas diferentes, exigindo alternância à sua usabilidade;



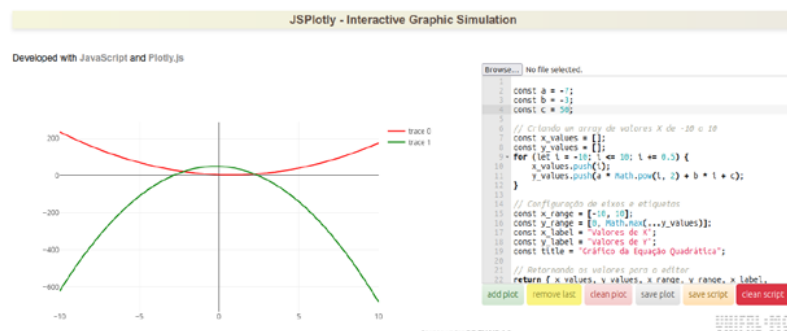
10. Gráficos produzidos podem visualizados somente dentro dos programas;
11. Alguns programas para gráficos não permitem a abertura de mais de uma sessão (por exemplo, Origin e Prism);
12. Uma vasta coleção de funcionalidades de cada suíte, muito além da necessidade básica que envolve um estudo para compreensão de um gráfico;
13. Gráficos renderizados por vezes apenas bidimensionalmente, e não em 3D;
14. Tipos gráficos 2D bastante limitados para alguns programas;
15. Um caminho por vezes detalhado e único junto aos programas, entre janelas e abas, para se buscar apenas a funcionalidade específica à elaboração de um gráfico simulado por uma função matemática;
16. Níveis limitados para ações de "desfazer/refazer" ("undo/redo") para vários programas listados acima;
17. Dependência de hardware de bom desempenho para alguns programas (exemplificando, Origin Lab requer 3 GB de memória de disco e um mínimo de 4 GB de RAM para operar, essa última também necessária ao Matlab);
18. Limitação de compartilhamento para o arquivo original do gráfico a quem não possua o mesmo programa, impossibilitando também a personalização do produto (por exemplo, limites e rótulos de eixos, novos dados, títulos e subtítulos);
19. Compartilhamento do gráfico produzido como arquivo de imagem, ou como arquivo compilável especificamente em cada programa;

20. Produção e compartilhamento de gráficos sem qualquer [interatividade](#) (Power Bi e outros pagos são exceções raras);
21. Apresentação estática dos dados e gráficos envolvidos na confecção (com exceção de alguns pacotes para R, incluindo [plotly](#));
22. Limitação de uso em ambientes virtuais de [aprendizagem](#) (instalação de programas no portal, por exemplo) e, por conseguinte, para algumas modalidades de [ensino](#) (por exemplo, híbrido e EaD).

JSPLOTLY

No ensejo de contornar algumas das características acima, foi desenvolvido o aplicativo gráfico *JSPlotly*, uma contração léxica de referência às linguagens envolvidas em sua construção. O aplicativo permite a elaboração de gráficos 2D e 3D, tanto para simulações matemáticas quanto para observação de dados, ou mesmo para mapas. Por estar baseado na linguagem [JavaScript](#), pode ser adaptado para ambientes web, incluindo ambientes virtuais de aprendizagem, por ser uma linguagem interpretada em *browsers* modernos (HTML5) e não compilada, como é o caso de *R*, *C/C++* e *Python*. Por ser construído a partir da biblioteca [Plotly.js](#), habilita todas as funcionalidades interativas presentes nessa. E por ser modificável em qualquer editor de texto, confere sua ampla adaptação e compartilhamento a partir de um arquivo de poucos kilobytes de memória. Um exemplo da tela única do *JSPlotly* está apresentada na Figura 1. Pode-se visualizar o potencial do aplicativo e sua interatividade clicando-se na imagem.

Figura 1 – Janela única do JSPlotly¹



Fonte: elaborado pelo autor, 2025.

O aplicativo, o código-fonte *JSPlotly*, algumas orientações de uso e alguns exemplos gráficos estão disponíveis no site [Bioquanti](#), uma iniciativa para disseminar o ensino reproduzível em nível superior e básico (Schneedorf, 2023). Originalmente concebido para abrigar a aplicação de três softwares de distribuição livre ao ensino-aprendizagem para aspectos quantitativos em Bioquímica, o site inclui atualmente e-books e *livros vivos* (programação letrada para a compilação de *chunks* de *R* e ações de *Jmol* diretamente na internet), e três aplicações desenvolvidas na UNIFAL-MG, incluindo *JSPlotly* (além de *ScriptRplot* – gráficos em *R* para *ggplot2* – e *Sisma* – simulação de diagramas dinâmicos). Em atual reformulação, o site abriga objetos didáticos para Bioquímica, Química, Físico-Química, Biologia Molecular, Matemática, Ciências da Natureza, Ciências Humanas e Linguagens.

As características da linguagem e da biblioteca empregadas em conjunto para *JSPlotly* estabelecem algumas habilidades apreciáveis ao aplicativo.

¹ A figura apresenta a sobreposição de equações quadráticas (esquerda), interpretadas a partir da variação de parâmetros da função disposta no editor de códigos (direita). Clique na imagem para acesso on-line do aplicativo.



SUMÁRIO

Características do JSPlotly para gráficos interativos e dinâmicos:

1. Não requer instalação;
2. Não requer requisitos [complementares](#) (por exemplo, bibliotecas .NET Microsoft e Java);
3. Não requer conexão de rede;
4. Não requer configuração de máquina de mínimo [desempenho](#) (por exemplo, memória RAM);
5. Pode ser carregado a partir de um simples visualizador de [HTML](#) (Firefox, Edge, Safari e Chrome);
6. Não requer um editor específico para construção, podendo ser elaborado a partir de um simples bloco de notas;
7. Possui código-fonte e produto contidos em um mesmo arquivo, facilitando armazenamento e compartilhamento;
8. Pode ser carregado a partir de computador, dispositivos móveis (smartphone) ou removíveis (pen-drive);
9. É interpretado a partir de um código de texto simples, utilizando memória física desprezível (**20** kB), ainda que permita a elaboração de gráficos sofisticados, interativos e dinâmicos (atualização por evento de usuário ou em tempo real);
10. Por ser passível de construção em editor simples de texto, possui compartilhamento irrestrito de seu código-fonte;
11. É capaz de produzir gráficos **2D** e **3D** interativos instantaneamente, tanto a partir de equações como de dados inseridos pelo usuário;
12. É capaz de produzir mapas interativos, elevando a extensão de uso para aplicações não matemáticas;



13. Correlaciona-se diretamente ao uso de linguagens de programação, tal como requerido pela 4a. e 5a. Revolução Industrial, e respectivamente espelhados nas competências digitais da Educação 4.0 e 5.0;
14. Por sua simplicidade como arquivo único em HTML, pode ser incorporado em páginas da web ou ambientes virtuais, permitindo seu uso para qualquer modalidade de ensino-aprendizagem (por exemplo, presencial, híbrido, remoto e EaD);
15. Insere-se nos conceitos de “pesquisa reproduzível”, bem como de “ensino reproduzível”, alicerçados pelo acesso e compartilhamento abertos, facilitados e documentados, para ferramentas digitais direcionadas a conteúdos científico, assim como das matrizes curriculares;
16. Permite que se incorpore no código-fonte outras bibliotecas em JavaScript, complementares à produção e manejo gráficos (ex: *numjs* para computação numérica e álgebra linear, *jsmath* para notações matemáticas complexas e *jStat* para computação estatística);
17. Possui diversas ações interativas por cliques de mouse, e que pontuam um grau lúdico à experimentação de funções matemáticas;
18. Renderiza gráficos a partir de linguagem de programação moderna e largamente utilizada, o que também permite ao aprendiz sua inserção paulatina em técnicas de programação contidos em outras linguagens recorrentes (Python e R), e do mundo geek (Arduino - “C/C++”).

ESTRUTURA E CARACTERÍSTICAS DO APLICATIVO

O *JSPlotly* foi elaborado para facilitar a interpretação de conteúdos matemáticos das ciências em geral, pela simulação do

comportamento gráfico de uma equação ou de dados introduzidos pelo usuário. Para permitir seu uso simplificado e distribuição ampla, foi construído como um arquivo HTML, conferindo-lhe acesso por um simples visualizador (*browser*) e sem necessidade de conexão. Como um todo, utilizou-se *HTML*, *CSS*, *JavaScript*, e *Plotly.js* para a geração de gráficos interativos. Um container flexível criado em HTML foi dividido em duas seções: uma para o *editor de códigos* on-line *Ace.js*, para edição de equações em *Plotly.js* e integração em página HTML, e um *ecrã gráfico*, no qual é plotada a equação ou os dados, com exibição visual e interativa, características daquela biblioteca. O container do simulador foi realizado com *flexbox* e *media queries*, permitindo design responsivo para telas pequenas e dispositivos móveis.

A personalização da interface foi concluída com CSS (*Cascading Style Sheets*), incluindo tamanho, posicionamento, cores, plano de fundo e dimensões dos elementos do container. De modo complementar, engenharia de *prompts* foi empregada junto a assistentes de IA (OpenAI ChatGPT e Anthropic Claude AI), para correção de códigos e para o refinamento da estrutura e de elementos do aplicativo (design, fontes, cores, dimensionamento, posicionamento).

O *JSPlotly* foi também concebido tendo em mente o que um usuário faria para estudar o comportamento gráfico de uma equação. Essas ações são consolidadas por botões simples ao redor do editor de equações. As equações são inseridas ao combinar códigos de *JavaScript*, para definição de variáveis, manipulação de *arrays*, cálculos matemáticos, e configurações de renderização, enquanto os códigos de *Plotly.js* são utilizados para a geração do gráfico interativo.

Ações do usuário para as funcionalidades do JSPlotly:

1. Observar a sintaxe da biblioteca “Plotly.js” por meio de uma equação pré-definida no carregamento do arquivo HTML do **aplicativo** (“função quadrática”);
2. Carregar uma nova função para **plotagem** (“browser”);



3. Editar parâmetros da equação, tanto de sua função matemática e das variáveis inseridas quanto da configuração do plot;
4. Plotar o gráfico ("Add Plot") e observar seus dados interativamente, por *hover* de mouse;
5. Alterar alguma informação no [editor](#) (variáveis, parâmetros, função matemática, eixos, por exemplo), e superimpor um novo gráfico para visualização das diferenças propostas e estudo do comportamento da função ("Add Plot");
6. Alterar a cor do último plot por simples clique de mouse em [ícone](#) ("change color") em barra de ferramentas;
7. Excluir da área gráfica a última equação ("Remove Last"), para novos ajustes e superimposição de outra curva;
8. Limpar a área gráfica ("Clean Plot");
9. Armazenar o gráfico como um arquivo HTML autônomo ("self-contained"), preservando a interatividade da biblioteca "Plotly.js" para estudo posterior ou compartilhamento [do\(s\)](#) gráfico(s) [representados](#) ("Save Plot");
10. Armazenar a equação como um "script" de "JavaScript" e "Plotly.js" para carregamento posterior ou [compartilhamento](#) ("Save Script");
11. Armazenar o plot alternativamete como [PNG](#) (rasterizada – web, documentos) ou [SVG](#) (vetorial – qualidade de publicação, edição) por clique de mouse em ícone da barra;
12. Excluir todos os elementos gráficos e de "script" para uma nova edição ("Clean All");
13. Desfazer/refazer em níveis ilimitados para comandos de edição de um gráfico;

14. Inserir um cursor de coordenadas por linha pontilhada no gráfico ("Toggle Spike Lines") por ícone da barra de ferramentas;
15. Autoescalonar o gráfico por ícone da **barra** ("Autoscale"), ou por clique duplo do mouse no ecrã;
16. Deslocar o gráfico no ecrã por ícone da **barra** ("Pan");
17. Ampliar qualquer área retangular sobre o gráfico por ícone da **barra** ("Zoom");
18. Deslocar o eixo "x" ou "y" por arraste de mouse;
19. Ampliar ou reduzir o gráfico pelo botão "scroll" do **mouse** (rolagem);
20. Inserir rótulos para título e eixos, por simples clique nas respectivas áreas do gráfico;
21. Clicar na legenda para ocultar/apresentar cada linha/dados do gráfico;
22. Editar o gráfico em aplicativo web (Plotly Chart Studio; <https://chart-studio.plotly.com/create/#/>) do desenvolvedor da biblioteca por ícone da barra de ferramentas ("Edit in Chart Studio").

INTERATIVIDADE

Os eventos interativos que agregam valor à representação dinâmica do gráfico são intrínsecos da biblioteca *Plotly.js*, concebida para criação e visualização interativa de dados para mais de 40 tipos em 2D e 3D, e painéis (*dashboards*), tanto para gráficos como para mapas. Assim, cada gráfico gerado apresenta no canto superior direito um conjunto de ícones para armazenamento como arquivo de imagem PNG, ampliação/redução (*zoom*), deslocamento do ecrã, e auto-escalonamento. Além disso, a biblioteca permite o uso de mouse para



1. Efeito de afastamento/aproximação visual geral ou de área específica ("zoom");
2. Arraste do ecrã de visualização ("pan", panorâmica);
3. Deslocamento individual de eixos;
4. Retorno à configuração **original** ("reset");
5. Seleção de dados/pontos individuais ou coletivos **por** ("box" ou "lasso");
6. Informações por passagem de **mouse** ("hover" ou "mouse over"; por exemplo, exibição dinâmica de valores);
7. Dicas customizáveis de **ferramentas** ("tooltips");
8. Legendas interativas;
9. Gráficos dinâmicos (atualização automática em fluxo);
10. Navegação **detalhada** ("drill1.down" e exploração em área específica);
11. Botões personalizados ("buttons");
12. Eventos personalizados por mouse ou ação de usuário ("call-backs", atualização de página por interação);
13. **Deslizadores** ("sliders") e menus para filtragem de informações;
13. Marcação de dados por linhas pontilhadas nos eixos gráficos e "hover" ("Toggle Spike Lines");
14. Menu de ícones para interatividade padrão ("Mode bar");
15. Personalização do "Mode bar" (por exemplo, exportação da imagem em SVG, cópia de dados do gráfico para a área de transferência);

16. Animações em “layout” ou transição de dados;
17. Facetas [interativas](#) (“facets” e gráficos lado a lado);
18. Sincronização entre gráficos [interconectados](#) (“zoom” e seleção de dados);
19. Criação de painéis dinâmicos [interativos](#) (“dashboards”);
20. Integração com bibliotecas [interativas](#) (D3.js e React.js) ou analíticas (por exemplo, *numjs*, *mathjs*, *jstat*).

Parte dessas ações, como *sliders* e *buttons*, não são passíveis de inserção direta nas equações interpretadas pelo aplicativo. Contudo, é possível inseri-las em arquivo adaptado a partir do próprio *JSPlotly*, permitindo ferramentas de interatividade adicionais, tais como as descritas acima. Um exemplo disso está representado na Figura 3, que ilustra um gráfico para titulação de um ácido fraco com recursos interativos adicionais (marcação de ponto nos eixos – *Toggle Spike Lines* – e controle deslizante – *slider*).

Figura 3 – Titulação de ácido fraco triprótico com controle deslizante para valores de pKa1 e marcação de dados por linhas pontilhadas³



Fonte: elaborado pelo autor, 2025.

- 3 Observe a seleção do valor de pKa1 pelo *slider* e seu posicionamento na curva. A inserção do *slider* foi realizada por adaptação do código-fonte de *JSPlotly*. Clique na imagem para acessar esse e outros gráficos e códigos interativos.

A INSERÇÃO DE COMPETÊNCIAS DIGITAIS E AS NOVAS POLÍTICAS PÚBLICAS

O *JSPlotly* descrito neste texto intenciona esboçar uma contribuição em consonância aos programas públicos de incentivo à educação digital, e as diversas iniciativas globais para a melhoria do ensino-aprendizagem levadas a termos nos últimos anos, como a *Educação 2030: O Futuro da Educação e das Competências* (OECD, 2015). A proposta instituída pela OECD (Organização para a Cooperação e Desenvolvimento Econômico), em 2015, prevê a composição de uma matriz conceitual de aprendizagem para as novas competências (conhecimentos, habilidades, atitudes e valores) a serem inseridas nos currículos escolares (Silva; Fernandes, 2019). Em paralelo, o programa da Unesco (Organização das Nações Unidas para a Educação, a Ciência e a Cultura) para Reimaginar Nossos Futuros Juntos: Um Novo Contrato Social para a Educação (Unesco, 2022), recomenda integrar tecnologias digitais de forma ética e inclusiva (Nunes; Bassani, 2024) frente às transformações ambientais e sociais previstas para os próximos trinta anos.

Também erigidos sobre essa preocupação global, os Objetivos de Desenvolvimento Sustentável (ODS) da Agenda 2030 da Organização das Nações Unidas (ONU, [s.d.]), que, tal como o ODS4, buscam “assegurar a educação inclusiva, equitativa e de qualidade, e promover oportunidades de aprendizagem ao longo da vida para todos” (Shulla *et al.*, 2020). E retornando a solo pátrio, a Cúpula de Líderes do G20 reunida no Rio de Janeiro em novembro de 2024, e que defende em seu Relatório do G20 sobre Educação junto a boas práticas inclusivas e de valorização dos profissionais em educação, a integração tecnológica nas práticas educacionais, e o compartilhamento de conteúdo educacional digital (Group of Twenty, 2024; Relatório [...], 2024).

BRASIL

Com a aprovação das diretrizes do *Novo Ensino Médio* pelo Conselho Nacional de Educação, que enfatizam flexibilização curricular, formação técnica e integração tecnológica, prevê-se o uso de ferramentas digitais e tecnologias emergentes, como programação e softwares educacionais. A *Educação Profissional e Tecnológica* (EPT) também é destacada, integrando capacitações técnicas em áreas como automação e desenvolvimento de software.

Também se agregam a essas iniciativas as propostas federais do *Plano Nacional de Educação* (PNE) 2024–2034 (PL 5665/23), e programas como a *Escola em Tempo Integral* do MEC, a *Estratégia Nacional de Escolas Conectadas* (Enec), o *Programa Mais Ciência na Escola* (MCTI/MEC) e a *Chamada CAPES para produtos de inovação tecnológica na educação*.

A ELABORAÇÃO DESTE CAPÍTULO

A formatação de texto (*RMarkdown*), inserção e referência cruzada de figuras, de hiperlinks, assim como gerenciamento bibliográfico e a conversão do *documento dinâmico* final para compilação a um modelo de arquivo *DOCX* previamente configurado foram exclusivamente conduzidos com a linguagem de programação *R* (versão 4.3.3, fev/2024) em ambiente de desenvolvimento integrado *RStudio* (versão 2024.09.1 Build 394) e pacote *quarto* (versão 1.4.4) como sistema de publicação científica.

AGRADECIMENTOS

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) – Código de Financiamento 001.

REFERÊNCIAS

- ALTINDIS, N. *et al.* Exploring the role of disciplinary knowledge in students' covariational reasoning during graphical interpretation. **International Journal of STEM Education**, [s. l.], v. 11, n. 1, p. 32, 2024.
- BUCKLEY, Y. M. *et al.* Using dynamic documents to mend cracks in the reproducible research pipeline. **Journal of EcologyWiley Online Library**, 2025.
- DOGUCU, M. Reproducibility in the Classroom. **Annual Review of Statistics and Its Application**, [s. l.], v. 12, 2024.
- EUPOREAN COMISSION. **Digital Competence Framework for Educators (DigCompEdu)**. European Comission, 2025. Disponível em: https://joint-research-centre.ec.europa.eu/digcompedu_en. Acesso em: 17 maio 2025.
- GROUP OF TWENTY. **G20 Rio de Janeiro Leaders' Declaration**. G20, 2024. Disponível em: <https://g20.org/wp-content/uploads/2024/11/G20-Rio-de-Janeiro-Leaders-Declaration-EN.pdf>. Acesso em: 17 maio 2025.
- KNUTH, D. E. Literate programming. **The computer journal**, [s. l.], v. 27, n. 2, p. 97-111, 1984.
- MOST, I. Effective Techniques In Industrial Marketing Research. **Journal of Marketing**, [s. l.], 1964.
- NUNES, J. M. G.; BASSANI, P. S. Reflexões sobre a educação do futuro em contexto Onlife. **Video Journal of Social and Human Research**, [s. l.], v. 3, n. 2, p. 1-9, 2024.
- OECD. **Future of Education and Skills 2030/2040**. Organisation for Economic Co-operation and Development, 2015. Disponível em: <https://www.oecd.org/en/about/projects/future-of-education-and-skills-2030.html>. Acesso em: 17 maio 2025.

ONU. **Sobre o nosso trabalho para alcançar os Objetivos de Desenvolvimento Sustentável no Brasil**. Organização das Nações Unidas, [s.d]. Disponível em: <https://brasil.un.org/pt-br/sdgs>. Acesso em: 17 maio 2025.

POWNALL, M. *et al.* **The impact of open and reproducible scholarship on students' scientific literacy, engagement, and attitudes towards science: a review and synthesis of the evidence**. [s. l.], 2022.

PRAKASH, S. *et al.* **EDUCATION 5.0: REVOLUTIONIZING LEARNING FOR THE FUTURE**. [s. l.], 2023.

UNESCO. **Reimaginar nossos futuros juntos: um novo contrato social para a educação**. Brasília, DF: Comissão Internacional sobre os Futuros da Educação, UNESCO; Boadilla del Monte: Fundación SM, 2022.

RELATÓRIO do G20 sobre Educação prioriza valorização dos profissionais de ensino. **Agência Gov**, 1 nov. 2024. Disponível em: <https://agenciagov.etc.com.br/noticias/202411/relatorio-do-g20-sobre-educacao-prioriza-valorizacao-dos-profissionais-de-ensino-1>. Acesso em: 17 maio 2025.

SCHNEEDORF, J. M. Bioquanti, um website interativo para ensino-aprendizagem em Bioquímica. **Revista de Ensino de Bioquímica**, [s. l.], v. 21, n. 1, p. 110-124, 2023.

SHULLA, K. *et al.* Sustainable development education in the context of the 2030 Agenda for sustainable development. **International Journal of Sustainable Development & World Ecology**, [s. l.], v. 27, n. 5, p. 458-468, 2020.

SILVA, M. A. da; FERNANDES, E. F. O projeto educação 2030 da OCDE: uma bússola para a aprendizagem. **Revista Exitus**, [s. l.], v. 9, n. 5, p. 271-300, 2019.

VIEIRA, R. *et al.* Society 5.0 and Education 5.0: A critical reflection. *In: 2023 18th Iberian Conference on Information Systems and Technologies (CISTI)*. [S. l.]: IEEE, 2023. p. 1-6.