



Evento: XXVI Jornada de Extensão ▾

PROJETO APPGO: INTEGRAÇÃO ENTRE INTERFACE INTUITIVA E ARQUITETURA BACKEND NO ENSINO DE FÍSICA¹

**Cássia Regina Fracaro Polleto², Giordano Bruno Biasi Berwig³, Barbara Gündel⁴,
Edson Luiz Padoin⁵, Patrícia Carolina Pedrali⁶, Taíse Neves Possani⁷**

¹ Projeto de Extensão AppGO da Unijui, trabalho destinado para o Salão do Conhecimento 2025;

² Cássia Regina Fracaro Polleto - Bolsista PIBEX; acadêmica de Engenharia de Software (UNIJUI); cassia.polleto@sou.unijui.edu.br.

³ Giordano Bruno Biasi Berwig - Bolsista PIBEX; acadêmico de Ciências da Computação (UNIJUI); giordano.berwig@sou.unijui.edu.br.

⁴ Professor Extensionista - Curso de Design - barbara.gundel@unijui.edu.br.

⁵ Professor Extensionista - Curso de Ciência da Computação - padoin@unijui.edu.br.

⁶ Professor Extensionista - Curso de Engenharia Mecânica - patricia.pedrali@unijui.edu.br.

⁷ Professor Extensionista - Curso de Letras - taise.possani@unijui.edu.br.

INTRODUÇÃO

A evolução das tecnologias digitais têm impactado significativamente a forma como o conhecimento é produzido e compartilhado. No contexto educacional, surgem cada vez mais soluções tecnológicas voltadas para o aprimoramento do ensino e da aprendizagem, especialmente com o uso de aplicativos interativos e plataformas digitais acessíveis a diferentes públicos. (Gil, 2008)

Nesse cenário, torna-se relevante analisar iniciativas que aliam tecnologia e educação de forma inovadora. A pesquisa desenvolvida no âmbito do Projeto AppGO, por meio do desenvolvimento de softwares como: PortGo, PhysicGo e MathGo, propõe justamente essa integração, buscando compreender como o uso de ferramentas digitais pode favorecer o engajamento, a personalização e o acesso facilitado ao conteúdo educacional (Lakatos; Marconi, 2017).

Este trabalho tem como objetivo apresentar uma análise detalhada do progresso alcançado nas três iniciativas citadas, com foco nos aspectos de usabilidade, arquitetura e implementação das soluções desenvolvidas. Em especial, destaca-se o desenvolvimento do PhysicGo como uma plataforma interativa para o ensino de Física.

Para melhor organização e compreensão dos dados, este artigo foi estruturado em dois eixos principais: (1) Desenvolvimento de Interface Intuitiva para Engajamento no



Aprendizado e (2) Arquitetura e Implementação do *Backend* para o PhysicGo: Fundamentando a Plataforma de Aprendizagem Interativa.

METODOLOGIA

A pesquisa adota uma abordagem qualitativa, voltada para a compreensão dos elementos que compõem os aplicativos desenvolvidos no Projeto AppGO, conforme definição de Denzin e Lincoln (2006). Trata-se também de uma pesquisa descritiva, que, segundo Gil (2008), visa detalhar as características dos objetos estudados, no caso, os aspectos técnicos do PhysicGo. Por fim, tem natureza bibliográfica, com base em fontes teóricas já publicadas, como destacam Lakatos e Marconi (2017).

RESULTADOS E DISCUSSÃO

Esta seção apresenta os resultados obtidos com o desenvolvimento do módulo PhysicGO. A discussão está organizada em duas partes fundamentais: primeiramente, aborda-se a concepção da interface do usuário (frontend), focada em usabilidade e engajamento. Em seguida, detalha-se a arquitetura do sistema de retaguarda (backend), responsável por garantir a performance, segurança e lógica da aplicação, demonstrando como a integração dessas frentes resultou em uma plataforma de aprendizagem funcional e robusta.

1. Desenvolvimento de Interface Intuitiva para Engajamento no Aprendizado

O PhysicGo foi desenvolvido como uma evolução do sistema MathGo, este último voltado para o ensino de matemática. Inspirados nessa proposta, buscamos aprimorar a experiência do usuário com foco no conteúdo de Física, trazendo melhorias significativas tanto na usabilidade (facilidade de uso) quanto na identidade visual (apresentação gráfica do sistema), conforme orientam os princípios de design centrado no usuário (Nielsen, 1994).

A interface foi planejada cuidadosamente desde o início do projeto. O layout inicial foi desenhado no Figma, uma ferramenta amplamente utilizada por designers para criar protótipos de páginas e aplicativos. A partir da logomarca do projeto, construímos uma identidade visual consistente, garantindo que o sistema mantivesse a mesma linguagem gráfica e comunicacional da proposta pedagógica. Esse protótipo serviu como referência direta para a implementação da interface real, realizada pela programadora responsável pelo



frontend (camada visual do sistema), assegurando fidelidade ao design e uma boa experiência de uso (Mayer, 2001).

O sistema foi construído utilizando tecnologias modernas, organizadas em duas camadas principais: o *backend* (parte interna) e o *frontend* (parte visual). O backend, responsável pelo armazenamento de dados, regras de funcionamento e comunicação com o banco de dados, foi desenvolvido com o Laravel, um framework de desenvolvimento baseado em PHP, amplamente reconhecido por sua sintaxe intuitiva, recursos poderosos e suporte de uma comunidade ativa. O PHP, por sua vez, é uma linguagem de programação muito utilizada na criação de sites e sistemas web. Um framework, nesse contexto, é um conjunto de ferramentas e componentes prontos que facilita e agiliza a construção de aplicações, garantindo também maior organização e segurança no código (Stauffer, 2023).

Na camada visual (*frontend*), utilizamos o Blade, que é um mecanismo de *template* integrado ao Laravel. Ele permite montar páginas dinâmicas, ou seja, páginas que mudam de acordo com os dados de forma organizada, clara e reutilizável. Para definir o estilo das telas, como cores, tamanhos de letras, espaçamentos e posicionamento dos elementos, utilizamos CSS (linguagem de estilo) em conjunto com o Bootstrap, um *framework* de código aberto que oferece blocos prontos para criar interfaces bonitas, consistentes e adaptáveis (Lima, 2021).

Para tornar a interface mais interativa, utilizamos JavaScript, uma linguagem que permite criar ações dinâmicas, como cliques, mudanças de conteúdo sem recarregar a página, animações e validações. Para facilitar essas funcionalidades, empregamos também o jQuery, que é uma biblioteca JavaScript leve e rica em recursos. O jQuery simplifica tarefas comuns como manipular elementos da página, responder a ações do usuário e fazer requisições de dados sem recarregar tudo (como no caso de perguntas que mudam uma a uma), por meio de uma linguagem mais enxuta e compatível com vários navegadores (Welling; Thomson, 2017).

A lógica de navegação foi projetada para ser simples e progressiva: o usuário responde uma pergunta por vez, o que favorece o foco e evita sobrecarga de informações. Essa estrutura também permite otimizar o carregamento de dados, reduzindo o tempo de resposta e contribuindo para um melhor desempenho geral do sistema (Mayer, 2001).

2. Arquitetura e Implementação do *Backend* para o PhysicGo: Fundamentando a Plataforma de Aprendizagem Interativa



Para garantir o desempenho eficiente do PhysicGo, é essencial não apenas uma interface acessível, mas também um *backend* sólido, responsável pelo gerenciamento de dados, lógica do sistema, segurança e performance. A tecnologia escolhida para essa estrutura foi o Laravel.

Como framework baseado no padrão MVC (Model-View-Controller), o Laravel permite uma organização clara do sistema: o Model lida com o acesso e a manipulação dos dados; o View, implementado com Blade, cuida da apresentação visual; e o Controller funciona como intermediário entre as camadas. Essa arquitetura favorece a manutenção do projeto e facilita futuras evoluções da plataforma (Stauffer, 2023).

Para armazenamento e manipulação dos dados, foi utilizado o MySQL, com o auxílio do Eloquent ORM, que possibilita trabalhar com objetos PHP em vez de consultas SQL diretas. Isso contribui para um código mais legível e menos propenso a erros, especialmente ao lidar com informações críticas, como dados de usuários, autenticação e controle de acesso (Welling; Thomson, 2017).

De acordo com Stauffer (2023), a segurança do sistema foi uma preocupação central. O Laravel oferece proteção nativa contra ameaças como SQL Injection, CSRF e XSS, além de gerenciar autenticação com senhas criptografadas, assegurando a integridade e a privacidade das informações armazenadas.

No que diz respeito à experiência do usuário, Mayer (2001) afirma que o *backend* é projetado para entregar as perguntas de forma progressiva, evitando o carregamento massivo de dados e otimizando o desempenho. Essa lógica de entrega sob demanda é gerenciada diretamente pelo servidor.

A comunicação entre o frontend e o backend é realizada por meio de chamadas AJAX e jQuery, permitindo atualizações assíncronas no conteúdo sem a necessidade de recarregar a página. Em síntese, o backend do PhysicGo não é apenas responsável por armazenar dados, mas atua como o núcleo da interatividade da aplicação, promovendo segurança, eficiência e adaptabilidade para diferentes contextos de aprendizagem (Welling; Thomson, 2017).

CONSIDERAÇÕES FINAIS

A partir do objetivo proposto, que consistiu em analisar o desenvolvimento das soluções educacionais do Projeto AppGO, com ênfase na construção do PhysicGo como uma



plataforma voltada ao ensino de Física por meio de tecnologias modernas, é possível compreender a relevância do design de interface e da arquitetura do sistema para o aumento do engajamento dos estudantes e a melhoria da eficiência do processo de aprendizagem.

Os principais resultados apontam que a criação de uma interface responsiva e intuitiva, aliada a um *backend* estruturado com Laravel e ao uso de recursos como jQuery e AJAX, favorece uma navegação fluida e a entrega progressiva de conteúdo. Essa combinação proporciona maior acessibilidade, leveza na experiência do usuário e integração entre os aspectos visuais e funcionais do sistema.

Como contribuição, a pesquisa reforça o potencial das ferramentas digitais no contexto educacional, mostrando que a integração entre pedagogia e tecnologia pode modernizar o processo de ensino, tornando-o mais atrativo, eficiente e adaptado às novas demandas educacionais.

Palavras-chave: PhysicGo. Tecnologia. Software. Educação. Inovação.

REFERÊNCIAS BIBLIOGRÁFICA

DENZIN, Norman K.; LINCOLN, Yvonna S. **O planejamento da pesquisa qualitativa: teorias e abordagens**. 2. ed. Porto Alegre: Artmed, 2006.

GIL, Antonio Carlos. **Métodos e técnicas de pesquisa social**. 6. ed. São Paulo: Atlas, 2008.

LAKATOS, Eva Maria; MARCONI, Marina de Andrade. **Fundamentos de metodologia científica**. 7. ed. São Paulo: Atlas, 2017.

LIMA, Guilherme. **Bootstrap: O que é, Documentação, como e quando usar**. Alura, 20 jul. 2021. Disponível em:

<https://www.alura.com.br/artigos/bootstrap?srsId=AfmBOooL3GUQ-MRGo1bQctXfnh5UknPxS4RyIc-ybvrXGgQkuY7S7RUB>. Acesso em: 26 jun. 2025.

MAYER, R. E. **Multimedia learning**. 2. ed. Cambridge: Cambridge University Press, 2001.

NIELSEN, J. **Usability engineering**. San Francisco: Morgan Kaufmann, 1994.

STAUFFER, M. **Laravel: Up & Running**. 2. ed. Sebastopol: O'Reilly Media, 2023.

WELLING, Luke; THOMSON, Laura. **PHP e MySQL: desenvolvimento web**. 5. ed. São Paulo: Novatec, 2017.